

RZ/G Series, 2nd Generation

User's Manual: Hardware

LSIs for Rich Graphics Applications

arm

Specifications common to RZ/G Series Products

RZ/G2H

RZ/G2M V1.3

RZ/G2M V3.0

RZ/G2N

RZ/G2E

All information contained in these materials, including products and product specifications, represents information on the product at the time of publication and is subject to change by Renesas Electronics Corp. without notice. Please review the latest information published by Renesas Electronics Corp. through various means, including the Renesas Electronics Corp. website (<http://www.renesas.com>).

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.

"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.

"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION ("Vulnerability Issues"). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:

www.renesas.com/contact/.

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Trademarks (continued)

For the "Cortex" notation, it is used as follows;

- Arm® Cortex®-A53
- Arm® Cortex®-A57
- Arm® Cortex®-R7

Note that after this page, they may be noted as Cortex-A53, Cortex-A57, and Cortex-R7 respectively.

Examples of trademark or registered trademark used in the RZ/G series, 2nd Generation User's Manual:

Hardware;

AMBA®: AMBA is a registered trademark of Arm Limited.

CoreLink™: CoreLink is a trademark of Arm Limited.

CoreSight™: CoreSight is a trademark of Arm Limited.

HDMI®: HDMI is a registered trademark of HDMI Licensing, LLC.

HyperFlash™: HyperFlash is a trademark of Cypress Semiconductor Corporation.

Magic Packet™: Magic Packet is a trademark of Advanced Micro Devices, Inc.

MIPI®: MIPI is a registered trademark of MIPI Alliance, Inc.

Neon™: Neon is a trademark of Arm Limited.

OpenGL ES™: OpenGL ES is a trademark of Khronos Group, Inc.

PCIe®: PCIe is a registered trademark of PCI-SIG, Inc.

PCI Express®: PCI Express is a registered trademark of PCI-SIG, Inc.

PowerVR®: PowerVR is a registered trademark of Imagination Technologies Group Plc.

Note that in each section of the Manual, trademark notation of ® and ™ may be omitted.

All other trademarks and registered trademarks are the property of their respective owners.

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.

2. Processing at power-on

The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.

3. Input of signal during power-off state

Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.

4. Handling of unused pins

Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.

5. Clock signals

After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.

6. Voltage application waveform at input pin

Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).

7. Prohibition of access to reserved addresses

Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.

8. Differences between products

Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

How to Use This Manual

1. Purpose and Target Readers

This manual is designed to provide the user with an understanding of the hardware functions of the RZ/G series, 2nd generation products. It is intended for users designing application systems incorporating the RZ/G series, 2nd generation products. A basic knowledge of electric circuits, logical circuits, and MCUs is necessary in order to use this manual.

The manual comprises an overview of the product; descriptions of the CPU, system control functions, peripheral functions for each product of the RZ/G series, 2nd generation.

Particular attention should be paid to the precautionary notes when using the manual. These notes occur within the body of the text, at the end of each section, and in the Usage Notes section.

The revision history summarizes the locations of revisions and additions. It does not list all revisions. Refer to the text of the manual for details.

2. Revision Numbers of User's Manual

The development of the basic line-up of the RZ/G series, 2nd generation has been finished.

Note. Additional new line-up of the RZ/G series, 2nd generation (ex. RZ/G2M V3.0) is still under development.

Note that update information for the development finished product(s) is provided by the Renesas Technical Update.

Following table summarizes the relationship between manual revisions and products.

| RZ/G series, 2nd generation Products | Manual Revision | Remarks |
|--|-----------------|---------|
| | Rev.1.01 | |
| RZ/G2H | Applicable | — |
| RZ/G2M V1.3 | Applicable | — |
| RZ/G2M V3.0 | N/A (Rev.0.5) | — |
| RZ/G2N | Applicable | — |
| RZ/G2E | Applicable | — |

3. Notation of reference document for Realtime Core.

This document described except parts of the Realtime Core of RZ/G Series, 2nd Generation products. Refer to document of RZ/G Series, 2nd Generation User's Manual: Hardware, Additional document for Real Time Core.

Contents

| | | |
|---------|--|------|
| 1. | Overview..... | 1-1 |
| 1.1 | Introduction..... | 1-1 |
| 1.2 | List of Specifications | 1-2 |
| 1.2.1 | Arm Core | 1-2 |
| 1.2.2 | CPU Core Peripherals..... | 1-3 |
| 1.2.3 | External Bus Module | 1-5 |
| 1.2.4 | Internal Bus Module | 1-7 |
| 1.2.5 | Internal Memory | 1-9 |
| 1.2.6 | Graphics Units | 1-10 |
| 1.2.7 | Video Processing | 1-14 |
| 1.2.8 | Sound Interface | 1-19 |
| 1.2.9 | Storage | 1-22 |
| 1.2.10 | Network | 1-24 |
| 1.2.11 | Timer..... | 1-25 |
| 1.2.12 | Peripheral Module..... | 1-26 |
| 1.2.13 | Others..... | 1-28 |
| 2. | Area Map | 2-1 |
| 3. | Mode Pin Settings (RZ/G2H) | 3-1 |
| 4. | Mode Pin Settings (RZ/G2M V1.3, RZ/G2M V3.0) | 4-1 |
| 5. | Mode Pin Settings (RZ/G2N) | 5-1 |
| 6. | Mode Pin Settings (RZ/G2E)..... | 6-1 |
| 7. | AP-System Core..... | 7-1 |
| 7.1 | Overview..... | 7-1 |
| 7.1.1 | Features..... | 7-2 |
| 7.1.1.1 | Cortex-A57 CPU cluster | 7-2 |
| 7.1.1.2 | Cortex-A53 CPU Cluster | 7-3 |
| 7.1.1.3 | Cache Coherent Interconnect (CCI-500)..... | 7-4 |
| 7.1.1.4 | Generic Counter | 7-4 |
| 7.1.2 | Block Diagram..... | 7-5 |
| 7.1.3 | External Pins | 7-7 |
| 7.1.4 | Register Configuration..... | 7-7 |
| 7.1.5 | Connected Module..... | 7-11 |
| 7.2 | Register Description..... | 7-12 |
| 7.3 | Operation | 7-13 |
| 7.3.1 | Boot Address Setting | 7-13 |
| 7.3.2 | Power-Up/Down Mechanism..... | 7-13 |
| 7.3.3 | Switching between AArch32 and AArch64..... | 7-16 |
| 7.3.4 | Maximum Outstanding Setting of CCI-500..... | 7-16 |
| 7.3.5 | System Address Map | 7-16 |
| 7.3.6 | Frequency for Counting Up of Generic Counter..... | 7-17 |
| 7.3.7 | How to Start a Generic Counter | 7-17 |
| 7.4 | Usage Notes | 7-18 |
| 7.4.1 | L2 Cache Setting for Cortex-A57 | 7-18 |
| 7.4.2 | Memory Attribute | 7-18 |

| | | |
|--------|---|-------|
| 8. | Pin Function Controller (PFC)..... | 8-1 |
| 8.1 | Overview..... | 8-1 |
| 8.1.1 | Features..... | 8-1 |
| 8.1.2 | Block Diagram..... | 8-2 |
| 8.1.3 | External Pins..... | 8-2 |
| 8.1.4 | Register Configuration..... | 8-3 |
| 8.1.5 | Connected Module..... | 8-7 |
| 8.2 | Register Description..... | 8-7 |
| 8.2.1 | LSI Multiplexed Pin Setting Mask Register (PMMR)..... | 8-8 |
| 8.2.2 | GPIO/Peripheral Function Select Register 0-7 (GPSR0-7)..... | 8-9 |
| 8.2.3 | Peripheral Function Select Register 0-18 (IPSR0-18)..... | 8-13 |
| 8.2.4 | DRV Control Register 0-24 (DRVCTRL0-24)..... | 8-22 |
| 8.2.5 | POC Control Register 0 (POCCTRL0)..... | 8-34 |
| 8.2.6 | TDSEL Control Register 0 (TDSELCTRL0)..... | 8-36 |
| 8.2.7 | LSI pin pull-enable register 0-6 (PUEN0-6)..... | 8-38 |
| 8.2.8 | LSI pin pull-up/down control Register 0-6 (PUD0-6)..... | 8-43 |
| 8.2.9 | Module Select Register 0-2 (MOD_SEL0-2)..... | 8-47 |
| 8.3 | Operation..... | 8-55 |
| 8.3.1 | Function Setting for Multiplexed Pins..... | 8-55 |
| 8.3.2 | Setting Pull-Up/Down Resistors..... | 8-56 |
| 8.4 | Usage Note..... | 8-57 |
| 8.4.1 | Notes for Using I2C0/3/5..... | 8-57 |
| 9. | Pin Function Controller (PFC)..... | 9-1 |
| 9.1 | Overview..... | 9-1 |
| 9.1.1 | Features..... | 9-1 |
| 9.1.2 | Block Diagram..... | 9-2 |
| 9.1.3 | External Pins..... | 9-2 |
| 9.1.4 | Register Configuration..... | 9-3 |
| 9.1.5 | Connected Module..... | 9-5 |
| 9.2 | Register Description..... | 9-6 |
| 9.2.1 | LSI Multiplexed Pin Setting Mask Register (PMMR)..... | 9-7 |
| 9.2.2 | GPIO/Peripheral Function Select Register 0-6 (GPSR0-6)..... | 9-8 |
| 9.2.3 | Peripheral Function Select Register 0-15 (IPSR0-15)..... | 9-11 |
| 9.2.4 | DRV Control Register 0-24 (DRVCTRL8-10)..... | 9-18 |
| 9.2.5 | POC Control Register 0 (POCCTRL0)..... | 9-21 |
| 9.2.6 | POC Control Register 2 (POCCTRL2)..... | 9-22 |
| 9.2.7 | TDSEL Control Register 0 (TDSELCTRL0)..... | 9-23 |
| 9.2.8 | LSI pin pull-enable register 0-5 (PUEN0-5)..... | 9-25 |
| 9.2.9 | LSI pin pull-up/down control Register 0-5 (PUD0-5)..... | 9-27 |
| 9.2.10 | Module Select Register 0-1 (MOD_SEL0-1)..... | 9-29 |
| 9.3 | Operation..... | 9-37 |
| 9.3.1 | Function Setting for Multiplexed Pins..... | 9-37 |
| 9.3.2 | Setting Pull-Up/Down Resistors..... | 9-38 |
| 10. | General-Purpose Input/Output Ports (GPIO)..... | 10-1 |
| 10.1 | Overview..... | 10-1 |
| 10.1.1 | Features..... | 10-1 |
| 10.1.2 | Block Diagram..... | 10-2 |
| 10.1.3 | External Pins..... | 10-4 |
| 10.1.4 | Register Configuration..... | 10-4 |
| 10.1.5 | Connected Module..... | 10-9 |
| 10.2 | Register Description..... | 10-10 |
| 10.2.1 | General IO/Interrupt Switching Register n (IOINTSEL0 to IOINTSELn)..... | 10-10 |

| | | |
|----------|---|-------|
| 10.2.2 | General Input/Output Switching Register n (INOUTSEL0 to INOUTSELn) | 10-11 |
| 10.2.3 | General Output Register n (OUTDT0 to OUTDTn) | 10-12 |
| 10.2.4 | General Input Register n (INDT0 to INDTn)..... | 10-13 |
| 10.2.5 | Interrupt Display Register n (INTDT0 to INTDTn) | 10-14 |
| 10.2.6 | Interrupt Clear Register n (INTCLR0 to INTCLRn) | 10-16 |
| 10.2.7 | Interrupt Mask Register n (INTMSK0 to INTMSKn) | 10-17 |
| 10.2.8 | Interrupt Mask Clear Register n (MSKCLR0 to MSKCLRn)..... | 10-18 |
| 10.2.9 | Positive/Negative Logic Select Register n (POSNEG0 to POSNEGn) | 10-19 |
| 10.2.10 | Edge/level Select Register n (EDGLEVEL0 to EDGLEVELn) | 10-20 |
| 10.2.11 | Chattering Prevention On/Off Register n (FILONOFF0 to FILONOFFn) | 10-21 |
| 10.2.12 | Interrupt Sub Mask Register n (INTMSKS0 to INTMSKSn)..... | 10-22 |
| 10.2.13 | Interrupt Sub Mask Clear Register n (MSKCLRS0 to MSKCLRSn)..... | 10-23 |
| 10.2.14 | Output Data Select Register n (OUTDTSEL0 to OUTDTSELn) | 10-24 |
| 10.2.15 | Output Data High Register n (OUTDTH0 to OUTDTHn) | 10-25 |
| 10.2.16 | Output Data Low Register n (OUTDTL0 to OUTDTLn)..... | 10-26 |
| 10.2.17 | One Edge/Both Edge Select Register n (BOTHEDGE0 to BOTHEDGE n) | 10-27 |
| 10.3 | Operation | 10-28 |
| 10.3.1 | Port Pin Specifications | 10-28 |
| 10.3.2 | Operations in Each Mode..... | 10-36 |
| 10.3.2.1 | Mode Switching | 10-36 |
| 10.3.2.2 | General Input/Output Mode | 10-36 |
| 10.3.2.3 | Interrupt Input Mode..... | 10-36 |
| 10.3.3 | Handling of Input Signals on Port Pins..... | 10-37 |
| 10.3.3.1 | Chattering..... | 10-37 |
| 10.3.3.2 | Input Signal Synchronization..... | 10-37 |
| 10.3.4 | Interrupt Display Timing Charts | 10-38 |
| 10.3.5 | Using GPIO..... | 10-39 |
| 10.3.5.1 | Setting Edge-Sensitive Interrupt Input Mode..... | 10-39 |
| 10.3.5.2 | Setting Level-Sensitive Interrupt Input Mode..... | 10-40 |
| 10.3.5.3 | Setting General Output Mode | 10-41 |
| 10.3.5.4 | Setting Output data high/Output data low Mode..... | 10-42 |
| 10.3.5.5 | Setting General Input Mode | 10-43 |
| 11. | Clock Pulse Generator (CPG)..... | 11-1 |
| 11.1 | Overview..... | 11-1 |
| 11.1.1 | Features..... | 11-1 |
| 11.1.2 | Block Diagram..... | 11-2 |
| 11.1.3 | External Pins | 11-7 |
| 11.1.4 | List of Clock Outputs..... | 11-9 |
| 11.1.5 | Clock Operating Modes | 11-19 |
| 11.1.6 | Register Configuration..... | 11-23 |
| 11.2 | Register Description..... | 11-25 |
| 11.2.1 | CPG Write Protect Control Register (CPGWPCR) | 11-25 |
| 11.2.2 | CPG Write Protect Register (CPGWPR) | 11-26 |
| 11.2.3 | Frequency Control Register B (FRQCRB) | 11-27 |
| 11.2.4 | Frequency Control Register C (FRQCRC) | 11-30 |
| 11.2.5 | PLL Enable Control Register (PLLECR) | 11-32 |
| 11.2.6 | PLL0 Control Register (PLL0CR)..... | 11-34 |
| 11.2.7 | PLL2 Control Register (PLL2CR)..... | 11-37 |
| 11.2.8 | PLL3 Control Register (PLL3CR)..... | 11-39 |
| 11.2.9 | PLL4 Control Register (PLL4CR)..... | 11-42 |
| 11.2.10 | SD-IFn Clock Frequency Control Register (SDnCKCR, n = 0 to 3 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N], n = 0, 1, 3 [RZ/G2E])..... | 11-44 |
| 11.2.11 | RPC-IF Clock Frequency Control Register (RPCCKCR) | 11-46 |

| | | |
|-----------|--|-------|
| 11.2.12 | CAN-FD Clock Frequency Control Register (CANFDCKCR) | 11-49 |
| 11.2.13 | MSIOF Clock Frequency Control Register (MSOCKCR)..... | 11-50 |
| 11.2.14 | HDMI-IF Clock Frequency Control Register (HDMICKCR) | 11-51 |
| 11.2.15 | CSI0 Clock Frequency Control Register (CSI0CKCR)..... | 11-52 |
| 11.2.16 | RCLK Frequency Control Register (RCKCR)..... | 11-53 |
| 11.2.17 | LV0 Clock Control Register (LV0CKCR)..... | 11-55 |
| 11.2.18 | LV1 Clock Control Register (LV1CKCR)..... | 11-57 |
| 11.2.19 | ZA2 Clock Control Register (ZA2CKCR)..... | 11-58 |
| 11.2.20 | ZA8 Clock Control Register (ZA8CKCR)..... | 11-59 |
| 11.2.21 | Z2D Clock Control Register (Z2DCKCR)..... | 11-60 |
| 11.2.22 | Frequency Control Register D (FRQCRD)..... | 11-61 |
| 11.3 | Operation | 11-62 |
| 11.3.1 | Changing Frequency | 11-62 |
| 11.4 | Usage Notes | 11-63 |
| 11.4.1 | Write protect function for CPG registers | 11-63 |
| 11.4.2 | Notes on Board Designing | 11-64 |
| 12. | Module Standby, Software Reset..... | 12-1 |
| 12.1 | Overview..... | 12-1 |
| 12.1.1 | Features..... | 12-1 |
| 12.1.2 | External Pins | 12-1 |
| 12.1.3 | Register Configuration..... | 12-2 |
| 12.2 | Register Description..... | 12-6 |
| 12.2.1 | Module Stop Status Register (MSTPSRn (n = 1 to 10))..... | 12-6 |
| 12.2.1.1 | Module Stop Status Register 1 (MSTPSR1) | 12-7 |
| 12.2.1.2 | Module Stop Status Register 2 (MSTPSR2) | 12-9 |
| 12.2.1.3 | Module Stop Status Register 3 (MSTPSR3) | 12-11 |
| 12.2.1.4 | Module Stop Status Register 4 (MSTPSR4) | 12-13 |
| 12.2.1.5 | Module Stop Status Register 5 (MSTPSR5) | 12-15 |
| 12.2.1.6 | Module Stop Status Register 6 (MSTPSR6) | 12-17 |
| 12.2.1.7 | Module Stop Status Register 7 (MSTPSR7) | 12-19 |
| 12.2.1.8 | Module Stop Status Register 8 (MSTPSR8) | 12-21 |
| 12.2.1.9 | Module Stop Status Register 9 (MSTPSR9) | 12-23 |
| 12.2.1.10 | Module Stop Status Register 10 (MSTPSR10) | 12-25 |
| 12.2.2 | Module Stop Control Register (RMSTPCRn/SMSTPCRn (n = 1 to 10))..... | 12-27 |
| 12.2.2.1 | Realtime Module Stop Control Register 1 (RMSTPCR1)/System Module Stop Control Register 1 (SMSTPCR1)..... | 12-27 |
| 12.2.2.2 | Realtime Module Stop Control Register 2 (RMSTPCR2)/System Module Stop Control Register 2 (SMSTPCR2)..... | 12-29 |
| 12.2.2.3 | Realtime Module Stop Control Register 3 (RMSTPCR3)/System Module Stop Control Register 3 (SMSTPCR3)..... | 12-31 |
| 12.2.2.4 | Realtime Module Stop Control Register 4 (RMSTPCR4)/System Module Stop Control Register 4 (SMSTPCR4)..... | 12-33 |
| 12.2.2.5 | Realtime Module Stop Control Register 5 (RMSTPCR5)/System Module Stop Control Register 5 (SMSTPCR5)..... | 12-35 |
| 12.2.2.6 | Realtime Module Stop Control Register 6 (RMSTPCR6)/System Module Stop Control Register 6 (SMSTPCR6)..... | 12-37 |
| 12.2.2.7 | Realtime Module Stop Control Register 7 (RMSTPCR7)/System Module Stop Control Register 7 (SMSTPCR7)..... | 12-39 |
| 12.2.2.8 | Realtime Module Stop Control Register 8 (RMSTPCR8)/System Module Stop Control Register 8 (SMSTPCR8)..... | 12-41 |
| 12.2.2.9 | Realtime Module Stop Control Register 9 (RMSTPCR9)/System Module Stop Control Register 9 (SMSTPCR9)..... | 12-43 |

| | | |
|-----------|---|-------|
| 12.2.2.10 | Realtime Module Stop Control Register 10 (RMSTPCR10)/System Module Stop Control Register 10 (SMSTPCR10)..... | 12-45 |
| 12.2.3 | Software Reset Register (SRCRn (n = 1 to 10))..... | 12-46 |
| 12.2.3.1 | Software Reset Register 1 (SRCR1) | 12-47 |
| 12.2.3.2 | Software Reset Register 2 (SRCR2) | 12-49 |
| 12.2.3.3 | Software Reset Register 3 (SRCR3) | 12-51 |
| 12.2.3.4 | Software Reset Register 4 (SRCR4) | 12-53 |
| 12.2.3.5 | Software Reset Register 5 (SRCR5) | 12-55 |
| 12.2.3.6 | Software Reset Register 6 (SRCR6) | 12-57 |
| 12.2.3.7 | Software Reset Register 7 (SRCR7) | 12-59 |
| 12.2.3.8 | Software Reset Register 8 (SRCR8) | 12-61 |
| 12.2.3.9 | Software Reset Register 9 (SRCR9) | 12-63 |
| 12.2.3.10 | Software Reset Register 10 (SRCR10) | 12-65 |
| 12.2.4 | Software Reset Clearing Register n (SRSTCLRn (n = 0 to 11))..... | 12-67 |
| 12.2.5 | Secure Module Stop Control Register n (SCMSTPCRn, n = 0 to 11) | 12-68 |
| 12.2.6 | Secure Software Reset Access Enable Control Register n (SCSRSTECRn, n = 0 to 11)..... | 12-69 |
| 13. | Advanced Power Management Unit for AP-System Core (APMU) | 13-1 |
| 13.1 | Overview..... | 13-1 |
| 13.1.1 | Features..... | 13-1 |
| 13.1.2 | External Pins | 13-1 |
| 13.1.3 | Register Configuration..... | 13-2 |
| 13.1.4 | Connected Module | 13-4 |
| 13.2 | Register Description..... | 13-5 |
| 13.2.1 | Cortex-A53/Cortex-A57 CPU _n Power Status Control Register (CA53CPU _n CR, CA57CPU _n CR (n = 0 to 3))..... | 13-5 |
| 13.2.2 | Cortex-A53/Cortex-A57 Common Power Control Register (CA53CPUCMCR, CA57CPUCMCR) ... | 13-6 |
| 13.2.3 | Cortex-A53/Cortex-A57 CPU Wake Up Control Register (CA53WUPCR, CA57WUPCR) | 13-7 |
| 13.2.4 | Cortex-A53/Cortex-A57 Power Status Register (CA53PSTR, CA57PSTR)..... | 13-10 |
| 13.2.5 | Cortex-A53/Cortex-A57 Debug Resource Reset Control Register (CA53DBGRCR, CA57DBGRCR)..... | 13-15 |
| 13.3 | Operation | 13-17 |
| 13.3.1 | Power Control..... | 13-17 |
| 14. | System Controller (SYSC)..... | 14-1 |
| 14.1 | Overview..... | 14-1 |
| 14.1.1 | Features..... | 14-1 |
| 14.1.2 | Block Diagram..... | 14-2 |
| 14.1.3 | External Pins | 14-4 |
| 14.1.4 | Register Configuration..... | 14-4 |
| 14.1.5 | Connected module | 14-10 |
| 14.2 | Register Description..... | 14-11 |
| 14.2.1 | Common Registers and Arm CPU related registers | 14-11 |
| 14.2.1.1 | SYSC Status Register (SYSCSR) | 14-12 |
| 14.2.1.2 | Interrupt Status Register (SYSCISR) | 14-13 |
| 14.2.1.3 | Interrupt Status Clear Register (SYSCISCR)..... | 14-17 |
| 14.2.1.4 | Interrupt Enable Register (SYSCIER) | 14-20 |
| 14.2.1.5 | Interrupt Mask Register (SYSCIMR) | 14-24 |
| 14.2.1.6 | Cortex-A57 Wake Up Mask Register (WUPMSKCA57) | 14-27 |
| 14.2.1.7 | Cortex-A53 Wake Up Mask Register (WUPMSKCA53)..... | 14-29 |
| 14.2.1.8 | External Event Request Status Register (SYSCEERSR) | 14-31 |
| 14.2.1.9 | External Event Request Status Clear Register (SYSCEERSCR) | 14-33 |
| 14.2.1.10 | External Event Request Status Enable Register (SYSCEERSER)..... | 14-34 |
| 14.2.1.11 | External Event Request Status Register2 (SYSCEERSR2) | 14-35 |

| | | |
|-----------|--|-------|
| 14.2.1.12 | External Event Request Status Clear Register2 (SYSCEERSCR2)..... | 14-37 |
| 14.2.1.13 | External Event Request Status Enable Register2 (SYSCEERSER2)..... | 14-38 |
| 14.2.1.14 | External Request Mask Register (SYSCEXTMASK)..... | 14-39 |
| 14.2.1.15 | Power Status Register n (PWRSRn) (n = 0/6)..... | 14-40 |
| 14.2.1.16 | Power Shutoff Status Register n (PWROFFSRn) (n = 0/6)..... | 14-41 |
| 14.2.1.17 | Power Resume Status Register n (PWRONSRn) (n = 0/6)..... | 14-42 |
| 14.2.1.18 | Power Shutoff/Resume Error Register n (PWRErN) (n = 0/6)..... | 14-43 |
| 14.2.1.19 | Power Pseudo Shutoff Register n (PWRPSEUn) (n = 0/6)..... | 14-44 |
| 14.2.2 | Power Control Registers for SCU of Cortex-A53, SCU of Cortex-A57, Video Processing, and Video Codec..... | 14-45 |
| 14.2.2.1 | Power Status Register n (PWRSRn) (n = 3/4/5/8/9)..... | 14-46 |
| 14.2.2.2 | Power Shutoff Control Register n (PWROFFCRn) (n = 3/4/5/8/9)..... | 14-47 |
| 14.2.2.3 | Power Shutoff Status Register n (PWROFFSRn) (n = 3/4/5/8/9)..... | 14-48 |
| 14.2.2.4 | Power Resume Control Register n (PWRONSCRn) (n = 3/4/5/8/9)..... | 14-49 |
| 14.2.2.5 | Power Resume Status Register n (PWRONSRn) (n = 3/4/5/8/9)..... | 14-50 |
| 14.2.2.6 | Power Shutoff/Resume Error Register n (PWRErN) (n = 3/4/5/8/9)..... | 14-51 |
| 14.2.2.7 | Power Pseudo Shutoff Register n (PWRPSEUn) (n = 3/4/5/8/9)..... | 14-52 |
| 14.2.3 | Power Control Registers for 3D Graphics Engine..... | 14-53 |
| 14.2.3.1 | Power Status Register n (PWRSRn) (n = 2)..... | 14-54 |
| 14.2.3.2 | Power Shutoff Control Register n (PWROFFCRn) (n = 2)..... | 14-55 |
| 14.2.3.3 | Power Shutoff Status Register n (PWROFFSRn) (n = 2)..... | 14-56 |
| 14.2.3.4 | Power Resume Control Register n (PWRONSCRn) (n = 2)..... | 14-57 |
| 14.2.3.5 | Power Resume Status Register n (PWRONSRn) (n = 2)..... | 14-58 |
| 14.2.3.6 | Power Shutoff/Resume Error Register n (PWRErN) (n = 2)..... | 14-59 |
| 14.2.3.7 | Power Pseudo Shutoff Register n (PWRPSEUn) (n = 2)..... | 14-60 |
| 14.2.4 | Power Control Registers for Video Codec..... | 14-61 |
| 14.2.4.1 | Power Status Register n (PWRSRn) (n = 10)..... | 14-61 |
| 14.2.4.2 | Power Shutoff Control Register n (PWROFFCRn) (n = 10)..... | 14-62 |
| 14.2.4.3 | Power Shutoff Status Register n (PWROFFSRn) (n = 10)..... | 14-64 |
| 14.2.4.4 | Power Resume Control Register n (PWRONSCRn) (n = 10)..... | 14-65 |
| 14.2.4.5 | Power Resume Status Register n (PWRONSRn) (n = 10)..... | 14-67 |
| 14.2.4.6 | Power Shutoff/Resume Error Register n (PWRErN) (n = 10)..... | 14-68 |
| 14.2.4.7 | Power Pseudo Shutoff Register n (PWRPSEUn) (n = 10)..... | 14-69 |
| 14.3 | Operation..... | 14-70 |
| 14.3.1 | Power domain structure..... | 14-70 |
| 14.3.2 | Initial state of power domains..... | 14-73 |
| 14.3.3 | Power Control of Arm CPUs..... | 14-73 |
| 14.3.4 | Power Control of non Arm CPU modules..... | 14-74 |
| 14.4 | Usage Notes..... | 14-75 |
| 15. | Thermal Sensor/Chip Internal Voltage Monitor (THS/CIVM)..... | 15-1 |
| 15.1 | Overview..... | 15-1 |
| 15.1.1 | Features..... | 15-1 |
| 15.1.2 | Block Diagram..... | 15-1 |
| 15.1.3 | External Pins..... | 15-3 |
| 15.1.4 | Register Configuration..... | 15-3 |
| 15.1.5 | Connected Module..... | 15-4 |
| 15.2 | Register Description..... | 15-5 |
| 15.2.1 | Interrupt Status Register (IRQSTR)..... | 15-6 |
| 15.2.2 | Interrupt Mask Register (IRQMSK)..... | 15-7 |
| 15.2.3 | Threshold Edge/Level Register (IRQCTL)..... | 15-8 |
| 15.2.4 | Interrupt Enable Register (IRQEN)..... | 15-9 |
| 15.2.5 | Interrupt Temperature 1 Register (IRQTEMP1)..... | 15-10 |
| 15.2.6 | Interrupt Temperature 2 Register (IRQTEMP2)..... | 15-10 |

| | | |
|----------|--|-------|
| 15.2.7 | Interrupt Temperature 3 Register (IRQTEMP3)..... | 15-11 |
| 15.2.8 | Control Register (THCTR) | 15-12 |
| 15.2.9 | Status Register (THSTR) | 15-15 |
| 15.2.10 | Temperature Register (TEMP)..... | 15-16 |
| 15.2.11 | Voltage Register (VOLT) | 15-16 |
| 15.2.12 | THCODE Parameter1 Register (THCODE1) | 15-17 |
| 15.2.13 | THCODE Parameter2 Register (THCODE2) | 15-17 |
| 15.2.14 | THCODE Parameter3 Register (THCODE3) | 15-18 |
| 15.2.15 | PTAT Parameter1 Register (PTAT1)..... | 15-18 |
| 15.2.16 | PTAT Parameter2 Register (PTAT2)..... | 15-19 |
| 15.2.17 | PTAT Parameter3 Register (PTAT3)..... | 15-19 |
| 15.2.18 | Software Correction Parameter Register (THSCP)..... | 15-20 |
| 15.3 | Operation | 15-21 |
| 15.3.1 | The Initial Sequence of Thermal Sensor/Chip Internal Voltage Monitor | 15-21 |
| 15.3.1.1 | Setting of Normal Mode..... | 15-21 |
| 15.3.1.2 | Normal Mode | 15-23 |
| 15.3.2 | Interrupt | 15-24 |
| 15.3.3 | Standby Mode | 15-25 |
| 15.3.4 | Software Reset | 15-25 |
| 15.3.5 | Power Voltage..... | 15-27 |
| 15.3.6 | Input/output Clock | 15-27 |
| 15.4 | Usage Note..... | 15-28 |
| 16. | Thermal Sensor (THS)..... | 16-1 |
| 16.1 | Overview..... | 16-1 |
| 16.1.1 | Features..... | 16-1 |
| 16.1.2 | Block Diagram | 16-1 |
| 16.1.3 | External Pins | 16-2 |
| 16.1.4 | Register Configuration..... | 16-2 |
| 16.1.5 | Connected Module | 16-3 |
| 16.2 | Register Description..... | 16-4 |
| 16.2.1 | Interrupt Status Register (STR)..... | 16-4 |
| 16.2.2 | Interrupt Enable Register (ENR)..... | 16-5 |
| 16.2.3 | Interrupt Mask Register (INT_MASK)..... | 16-6 |
| 16.2.4 | Positive/Negative Logic Select Register (POSNEG) | 16-7 |
| 16.2.5 | THS Control Register (THSCR) | 16-8 |
| 16.2.6 | THS Status Register (THSSR)..... | 16-9 |
| 16.2.7 | Interrupt Control Register (INTCTLR)..... | 16-10 |
| 16.3 | Operation | 16-12 |
| 16.3.1 | Initial Sequence of Thermal Sensor | 16-12 |
| 16.3.2 | Temperature Measurement Using Register (When CPCTL = 0) | 16-13 |
| 16.3.3 | Temperature Measurement Using Register (When CPCTL = 1) | 16-13 |
| 16.3.4 | Interrupt | 16-14 |
| 16.3.5 | Standby Mode | 16-14 |
| 16.3.6 | Software Reset | 16-15 |
| 16.3.7 | Power Voltage..... | 16-16 |
| 16.3.8 | Input/output Clock | 16-16 |
| 16.3.9 | The Temperature Measurement Using External Pins (RZ/G2E) | 16-16 |
| 16.4 | Usage Note..... | 16-19 |
| 17. | Reset (RST)..... | 17-1 |
| 17.1 | Overview..... | 17-1 |
| 17.1.1 | Features..... | 17-1 |
| 17.1.2 | Block Diagram | 17-2 |

| | | |
|---------|--|-------|
| 17.1.3 | External Pins | 17-5 |
| 17.1.4 | Register Configuration..... | 17-6 |
| 17.1.5 | Connected module | 17-11 |
| 17.2 | Register Description..... | 17-12 |
| 17.2.1 | Mode Monitor Register (MODEMR) | 17-12 |
| 17.2.2 | Cortex-A57 Reset Control Register (CA57RESCNT)..... | 17-15 |
| 17.2.3 | Cortex-A53 Reset Control Register (CA53RESCNT)..... | 17-17 |
| 17.2.4 | Watchdog Timer Reset Control Register (WDTRSTCR)..... | 17-19 |
| 17.2.5 | PRESETOUT# Control Register (RSTOUTCR) | 17-21 |
| 17.2.6 | Cortex-A53 Boot Address Register (CA53BAR) | 17-22 |
| 17.2.7 | Cortex-A53 Boot Address Register2 (CA53BAR2)..... | 17-24 |
| 17.2.8 | Cortex-A57 Boot Address Register (CA57BAR) | 17-25 |
| 17.2.9 | Cortex-A57 Boot Address Register2 (CA57BAR2) | 17-27 |
| 17.2.10 | Cortex-A57 CPU _n (n = 0 to 3[RZ/G2H], n = 0 and 1[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]) Boot Address Register for 64-bit mode H (CA57CPU _n BARH) | 17-28 |
| 17.2.11 | Cortex-A57 CPU _n (n = 0 to 3[RZ/G2H], n = 0 and 1[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]) Boot Address Register for 64-bit mode L (CA57CPU _n BARL) | 17-29 |
| 17.2.12 | Cortex-A53 CPU _n (n = 0 to 3[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0], n = 0 and 1[RZ/G2E]) Boot Address Register for 64-bit mode H (CA53CPU _n BARH) | 17-32 |
| 17.2.13 | Cortex-A53 CPU _n (n = 0 to 3[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0], n = 0 and 1[RZ/G2E]) Boot Address Register for 64-bit mode L (CA53CPU _n BARL) | 17-33 |
| 17.2.14 | Standby Flag Register n (STBCHR _n) (n = 0 to 7) | 17-36 |
| 17.2.15 | Soft Power On Reset Control Register (SRESCR) | 17-37 |
| 17.2.16 | SYS Reset Flag Register (SRSTFR) | 17-38 |
| 17.3 | Operation | 17-39 |
| 17.3.1 | Power-On Reset by PRESET# Pin..... | 17-39 |
| 17.3.2 | WDT reset..... | 17-39 |
| 18. | Interrupt Controller (INTC) | 18-1 |
| 18.1 | Overview..... | 18-1 |
| 18.1.1 | Features | 18-1 |
| 18.1.2 | Block Diagram..... | 18-1 |
| 18.1.3 | External Pins | 18-1 |
| 18.1.4 | Connected Module..... | 18-2 |
| 19. | Interrupt Controller (INTC-AP)..... | 19-1 |
| 19.1 | Overview..... | 19-1 |
| 19.1.1 | Features..... | 19-1 |
| 19.1.2 | Block Diagram..... | 19-1 |
| 19.1.3 | External Pins | 19-1 |
| 19.1.4 | Register Configuration..... | 19-1 |
| 19.1.5 | Connected Module..... | 19-2 |
| 19.2 | Register Description..... | 19-3 |
| 19.3 | Operation | 19-4 |
| 19.3.1 | INTC-AP Register Configuration and Function Description..... | 19-4 |
| 19.3.2 | Interrupts Mapping..... | 19-5 |
| 20. | Interrupt Controller (INTC-EX) | 20-1 |
| 20.1 | Overview..... | 20-1 |
| 20.1.1 | Features..... | 20-1 |
| 20.1.2 | Block Diagram..... | 20-1 |
| 20.1.3 | External Pins | 20-2 |
| 20.1.4 | Register Configuration..... | 20-2 |
| 20.1.5 | Connected Module..... | 20-5 |

| | | |
|---------|---|-------|
| 20.2 | Register Description..... | 20-6 |
| 20.2.1 | Interrupt Request Status Register0 (INTREQ_STS0)..... | 20-6 |
| 20.2.2 | Interrupt Enable Status Register0 (INTEN_STS0)..... | 20-7 |
| 20.2.3 | Interrupt Enable Set Register0 (INTEN_SET0)..... | 20-8 |
| 20.2.4 | IRQn Detect Status Register (DETECT_STATUS)..... | 20-9 |
| 20.2.5 | IRQn Signal Level Monitor Register (MONITOR)..... | 20-10 |
| 20.2.6 | IRQn High Level Detect Status Register (HLVL_STS)..... | 20-11 |
| 20.2.7 | IRQn Low Level Detect Status Register (LLVL_STS)..... | 20-12 |
| 20.2.8 | IRQn Sync Rising Edge Detect Status Register (S_R_EDGE_STS)..... | 20-13 |
| 20.2.9 | IRQn Sync Falling Edge Detect Status Register (S_F_EDGE_STS)..... | 20-14 |
| 20.2.10 | IRQn Async Rising Edge Detect Status Register (A_R_EDGE_STS)..... | 20-15 |
| 20.2.11 | IRQn Async Falling Edge Detect Status Register (A_F_EDGE_STS)..... | 20-16 |
| 20.2.12 | IRQn Chattering Reduction Status Register (CHTEN_STS)..... | 20-17 |
| 20.2.13 | IRQn Configuration Register (CONFIG_n)..... | 20-18 |
| 20.2.14 | NMI Request Status Register 0 (NMIREQ_STS0)..... | 20-20 |
| 20.2.15 | NMI Enable Status Register 0 (NMIEN_STS0)..... | 20-22 |
| 20.2.16 | NMI Enable Set Register 0 (NMIEN_SET0)..... | 20-24 |
| 20.2.17 | NMI Detect Status Register (DETECT_STATUS_NMI)..... | 20-26 |
| 20.2.18 | NMI Signal Level Monitor Register (MONITOR_NMI)..... | 20-29 |
| 20.2.19 | NMI High Level Detect Status Register (HLVL_STS_NMI)..... | 20-32 |
| 20.2.20 | NMI Low Level Detect Status Register (LLVL_STS_NMI)..... | 20-34 |
| 20.2.21 | NMI Sync Rising Edge Detect Status Register (S_R_EDGE_STS_NMI)..... | 20-36 |
| 20.2.22 | NMI Sync Falling Edge Detect Status Register (S_F_EDGE_STS_NMI)..... | 20-38 |
| 20.2.23 | NMI Async Rising Edge Detect Status Register (A_R_EDGE_STS_NMI)..... | 20-40 |
| 20.2.24 | NMI Async Falling Edge Detect Status Register (A_F_EDGE_STS_NMI)..... | 20-42 |
| 20.2.25 | NMI Chattering Reduction Status Register (CHTEN_STS_NMI)..... | 20-44 |
| 20.2.26 | NMI Debounce Setting Register (DEB_SET_NMI)..... | 20-45 |
| 20.2.27 | NMI Configuration n Register (CONFIGn_NMI)..... | 20-46 |
| 20.2.28 | NMI Mask Lock Set Register (NMI_LCK)..... | 20-47 |
| 20.2.29 | NMI Lock Code Register (NMI_LCKCODE)..... | 20-47 |
| 20.2.30 | NMI Debug Control Enable Register (NMI_DBG)..... | 20-48 |
| 20.2.31 | NMI Debug Code Register (NMI_DBGCODE)..... | 20-48 |
| 20.3 | Operation..... | 20-49 |
| 20.3.1 | NMI Mask Lock Feature..... | 20-49 |
| 20.3.2 | Procedure of NMI Mask Lock..... | 20-49 |
| 21. | AXI-bus..... | 21-1 |
| 21.1 | Overview..... | 21-1 |
| 21.1.1 | Features..... | 21-1 |
| 21.1.2 | Block Diagram..... | 21-2 |
| 21.1.3 | External Pins..... | 21-6 |
| 21.1.4 | Register Configuration..... | 21-7 |
| 21.2 | Register Description..... | 21-13 |
| 21.2.1 | Address Split Control Register0 (ADSPLCR0)..... | 21-13 |
| 21.2.2 | Address Split Control Register1 (ADSPLCR1)..... | 21-14 |
| 21.2.3 | Address Split Control Register2 (ADSPLCR2)..... | 21-15 |
| 21.2.4 | Address Split Control Register3 (ADSPLCR3)..... | 21-17 |
| 21.2.5 | Decompression Area Configuration Register An (n = 0-15) (DCMPAREACRAn)..... | 21-18 |
| 21.2.6 | Decompression Area Configuration Register Bn (n = 0-15) (DCMPAREACRBn)..... | 21-19 |
| 21.2.7 | Decompression Shadow Area Configuration Register (DCMPSHDWCR)..... | 21-19 |
| 21.2.8 | Main Memory AXI Control Register (MMCR)..... | 21-21 |
| 21.2.9 | DRAM Protected Area Division Register n (n = 0-14) (DPTDIVCRn)..... | 21-22 |
| 21.2.10 | DRAM Protected Area Setting Register n (n = 0-15) (DPTCRn)..... | 21-23 |
| 21.2.11 | System RAM Protected Area Division Register n (n = 0-14) (SPTDIVCRn)..... | 21-25 |

| | | |
|----------|---|-------------|
| 21.2.12 | System RAM Protected Area Setting Register n (n = 0-15) (SPTCRn)..... | 21-26 |
| 21.2.13 | TR3 Control Register (TR3CR)..... | 21-28 |
| 21.2.14 | TR4 Control Register (TR4CR)..... | 21-29 |
| 21.2.15 | Secure Error Register Upper Address Register (SECADDRU_X)..... | 21-30 |
| 21.2.16 | Secure Error Register Lower Address Register (SECADDRL_X)..... | 21-31 |
| 21.2.17 | Secure Error Register (SECERR_X)..... | 21-31 |
| 21.2.18 | QOSCTRL SL INIT Register (QOSCTRL_SL_INIT)..... | 21-32 |
| 21.2.19 | QOSCTRL REF ARS Register (QOSCTRL_REF_ARS)..... | 21-33 |
| 21.2.20 | QOSCTRL REF ENBL Register (QOSCTRL_REF_ENBL)..... | 21-34 |
| 21.2.21 | QOSCTRL STATQC Register (QOSCTRL_STATQC)..... | 21-35 |
| 21.2.22 | QOSCTRL MEMBANK Register (QOSCTRL_MEMBANK)..... | 21-36 |
| 21.2.23 | QOSCTRL RAS Register (QOSCTRL_RAS)..... | 21-37 |
| 21.2.24 | QOSCTRL DANT Register (QOSCTRL_DANT)..... | 21-38 |
| 21.2.25 | QOSCTRL DANN LOW Register (QOSCTRL_DANN_LOW)..... | 21-39 |
| 21.2.26 | QOSCTRL DANN HIGH Register (QOSCTRL_DANN_HIGH)..... | 21-40 |
| 21.2.27 | QOSCTRL EMS LOW Register (QOSCTRL_EMS_LOW)..... | 21-41 |
| 21.2.28 | QOSCTRL EMS HIGH Register (QOSCTRL_EMS_HIGH)..... | 21-43 |
| 21.2.29 | QOSCTRL EARLYR Register (QOSCTRL_EARLYR)..... | 21-45 |
| 21.2.30 | QOSCTRL RACNT0 Register (QOSCTRL_RACNT0)..... | 21-46 |
| 21.2.31 | QOSCTRL INSFC Register (QOSCTRL_INSFC)..... | 21-47 |
| 21.2.32 | QOSCTRL RAEN Register (QOSCTRL_RAEN)..... | 21-48 |
| 21.2.33 | QOSCTRL STATGEN0 Register (QOSCTRL_STATGEN0)..... | 21-49 |
| 21.2.34 | QOSWT WTEN Register (QOSWT_WTEN)..... | 21-50 |
| 21.2.35 | QOSWT WTREF Register (QOSWT_WTREF)..... | 21-51 |
| 21.2.36 | QOSWT WTSET0 Register (QOSWT_WTSET0)..... | 21-52 |
| 21.2.37 | QOSWT WTSET1 Register (QOSWT_WTSET1)..... | 21-53 |
| 21.2.38 | QOSBW FIX QOS BANK0 Register n (QOSBW_FIX_QOS_BANK0)..... | 21-54 |
| 21.2.39 | QOSBW FIX QOS BANK1 Register n (QOSBW_FIX_QOS_BANK1)..... | 21-58 |
| 21.2.40 | QOSBW BE QOS BANK0 Register n..... | 21-62 |
| 21.2.41 | QOSBW BE QOS BANK1 Register n..... | 21-64 |
| 21.2.42 | CPU Domain Active Register n (CPU_ACTn) (n: 0 to 3)..... | 21-66 |
| 21.2.43 | Main Memory Secure Error Status Register (MMSECERRST)..... | 21-67 |
| 21.2.44 | Main Memory Secure Error Address Register0 (MMSECERRADDR0)..... | 21-69 |
| 21.2.45 | Main Memory Secure Error Address Register1 (MMSECERRADDR1)..... | 21-70 |
| 21.3 | Operation..... | 21-71 |
| 21.3.1 | DRAM Split/Linear Address mapping..... | 21-71 |
| 21.3.2 | Decompression of visual near lossless compressed image..... | 21-73 |
| 21.3.3 | QoS arbitration..... | 21-73 |
| 21.3.4 | Access protection for secure/safety regions..... | 21-73 |
| 21.3.4.1 | Memory Protection..... | 21-73 |
| 21.4 | Usage Notes..... | 21-74 |
| 21.4.1 | Change of the initial value of the register..... | 21-74 |
| 22. | IPMMU | 22-1 |
| 22.1 | Overview..... | 22-1 |
| 22.1.1 | Features..... | 22-1 |
| 22.1.2 | Block Diagram..... | 22-2 |
| 22.1.3 | External Pins..... | 22-3 |
| 22.1.4 | Register Configuration..... | 22-3 |
| 22.1.5 | Connected Module..... | 22-8 |
| 22.2 | Register Description..... | 22-9 |
| 22.2.1 | MMU Control Register n (IMCTRn)..... | 22-9 |
| 22.2.2 | MMU Translation Table Base Control Register n (IMTTBCRn)..... | 22-11 |
| 22.2.3 | MMU Translation Table Upper Base Register 0/1n (IMTTUBR0/1n)..... | 22-14 |

| | | |
|-----------|---|-------|
| 22.2.4 | MMU Translation Table Lower Base Register 0/1n (IMTTLBR0/1n) | 22-14 |
| 22.2.5 | MMU Memory Attribute Indirection Register 0n (IMMAIR0n) | 22-15 |
| 22.2.6 | MMU Memory Attribute Indirection Register 1n (IMMAIR1n) | 22-17 |
| 22.2.7 | MMU Error Status Register n (IMSTRn) | 22-19 |
| 22.2.8 | MMU Error Lower Address Register n (IMELARn) | 22-20 |
| 22.2.9 | MMU Error Upper Address Register n (IMEUARn) | 22-20 |
| 22.2.10 | PMB Control Register (IMPCTR) | 22-21 |
| 22.2.11 | PMB Address Array n (IMPMBAn) | 22-22 |
| 22.2.12 | PMB Data Array n (IMPMBDn) | 22-23 |
| 22.2.13 | PMB Status Register (IMPSTR) | 22-24 |
| 22.2.14 | PMB Error Address Register (IMPEAR) | 22-25 |
| 22.2.15 | uTLB Control Register n (IMUCTRn) | 22-26 |
| 22.2.16 | uTLB ASID Register n (IMUASIDn) | 22-27 |
| 22.2.17 | MMU System Control Register (IMSCTLR) | 22-28 |
| 22.2.18 | MMU Auxiliary Control Register (IMSAUXCTLR) | 22-30 |
| 22.2.19 | MMU Interrupt Status Register (IMSSTR) | 22-31 |
| 22.2.20 | MMU Performance Monitor Control Register (IMPFMCTR) | 22-35 |
| 22.2.21 | MMU Performance Monitor Total Translation Counter (IMPFMTOTAL) | 22-37 |
| 22.2.22 | MMU Performance Monitor Hit Counter (IMPFMHIT) | 22-38 |
| 22.2.23 | MMU Performance Monitor L3 Miss Counter (IMPFML3MISS) | 22-39 |
| 22.2.24 | MMU Performance Monitor L2 Miss Counter (IMPFML2MISS) | 22-39 |
| 22.2.25 | MMU Performance Monitor Miss Counter (IMPFMMISS) | 22-40 |
| 22.3 | Operation | 22-41 |
| 22.3.1 | Address Translation Sequence in micro-TLBs | 22-41 |
| 22.3.2 | Address Translation Sequence in the IPMMU | 22-42 |
| 22.3.3 | Address Space | 22-44 |
| 22.3.4 | Initialization Sequence | 22-48 |
| 22.3.5 | TLB Maintenance | 22-49 |
| 22.3.6 | Security Support | 22-50 |
| 22.3.7 | OS-ID support | 22-50 |
| 22.3.8 | Error Handling | 22-51 |
| 22.3.9 | Performance monitor | 22-52 |
| 22.4 | micro-TLB Assignment | 22-53 |
| 22.4.1 | micro-TLB Assignment | 22-53 |
| 22.5 | Usage Notes | 22-62 |
| 22.5.1 | IPMMU naming | 22-62 |
| 22.5.2 | IPMMU configuration | 22-63 |
| 22.5.2.1 | IPMMU configuration for FCPCS [RZ/G2H, RZ/G2M V3.0] | 22-63 |
| 22.5.2.2 | IPMMU configuration for 3DG [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] | 22-64 |
| 22.5.3 | Restrictions for IPMMU | 22-65 |
| 22.5.3.1 | Page size and Section size usage in VMSAv8-32 | 22-65 |
| 22.5.3.2 | 4KB page usage in VMSAv8-64 | 22-65 |
| 22.5.3.3 | Error Handling in VMSAv8-64 | 22-65 |
| 22.5.3.4 | Start Level for Page Table Walk | 22-66 |
| 22.5.3.5 | Virtual address space range | 22-68 |
| 22.5.3.6 | Treatments of ASID feature | 22-68 |
| 22.5.3.7 | Disabling the address translation by IPMMU | 22-68 |
| 22.5.3.8 | Restriction for DMAC transfer [RZ/G2M V1.3 only] | 22-68 |
| 22.5.3.9 | Restriction for OS-ID support [RZ/G2M V1.3 only] | 22-68 |
| 22.5.3.10 | Restriction for TLB location [RZ/G2M V1.3 only] | 22-68 |
| 22.5.3.11 | Restriction for TLB flush operation [RZ/G2M V1.3 only] | 22-68 |
| 22.5.3.12 | Restriction for IPMMU cache [RZ/G2M V1.3, RZ/G2N, RZ/G2E] | 22-68 |
| 22.5.3.13 | Restriction for PMB | 22-68 |
| 22.5.3.14 | Restriction for PMB with Stage 2 Translation [RZ/G2M V1.3 only] | 22-69 |

| | | |
|---------|--|-------|
| 23. | Direct Memory Access Controller for System (SYS-DMAC) | 23-1 |
| 23.1 | Overview | 23-1 |
| 23.1.1 | Features | 23-1 |
| 23.1.2 | External Pins | 23-3 |
| 23.1.3 | Register Configuration | 23-4 |
| 23.1.4 | Connected Module | 23-35 |
| 23.2 | Register Description | 23-36 |
| 23.2.1 | DMA Interrupt Status Register for channels 0 to 15 (DMAISTA_0) | 23-36 |
| 23.2.2 | DMA Interrupt Status Register for channels 16 to 31 (DMAISTA_1) | 23-38 |
| 23.2.3 | DMA Interrupt Status Register for channels 32 to 47 (DMAISTA_2) | 23-40 |
| 23.2.4 | DMA Secure Control Register for channels 0 to 15 (DMASEC_0) | 23-42 |
| 23.2.5 | DMA Secure Control Register for channels 16 to 31 (DMASEC_1) | 23-44 |
| 23.2.6 | DMA Secure Control Register for channels 32 to 47 (DMASEC_2) | 23-46 |
| 23.2.7 | DMA Operation Register for channels 0 to 15 (DMAOR_0) | 23-48 |
| 23.2.8 | DMA Operation Register for channels 16 to 31 (DMAOR_1) | 23-50 |
| 23.2.9 | DMA Operation Register for channels 32 to 47 (DMAOR_2) | 23-52 |
| 23.2.10 | DMA Channel Clear Register for channels 0 to 15 (DMACHCLR_0) | 23-54 |
| 23.2.11 | DMA Channel Clear Register for channels 16 to 31 (DMACHCLR_1) | 23-55 |
| 23.2.12 | DMA Channel Clear Register for channels 32 to 47 (DMACHCLR_2) | 23-56 |
| 23.2.13 | DPRAM Secure Control Register for channels 0 to 15 (DMADPSEC_0) | 23-57 |
| 23.2.14 | DPRAM Secure Control Register for channels 16 to 31 (DMADPSEC_1) | 23-58 |
| 23.2.15 | DPRAM Secure Control Register for channels 32 to 47 (DMADPSEC_2) | 23-59 |
| 23.2.16 | DMA Source Address Registers 0 to 47 (DMASAR_0 to DMASAR_47) | 23-60 |
| 23.2.17 | DMA Destination Address Registers 0 to 47 (DMADAR_0 to DMADAR_47) | 23-60 |
| 23.2.18 | DMA Transfer Count Registers 0 to 47 (DMATCR_0 to DMATCR_47) | 23-61 |
| 23.2.19 | DMA Transfer Count Registers B_0 to 47 (DMATCRB_0 to DMATCRB_47) | 23-61 |
| 23.2.20 | DMA Transfer Size Registers 0 to 47 (DMATSR_0 to DMATSR_47) | 23-62 |
| 23.2.21 | DMA Transfer Size Registers B_0 to 47 (DMATSRB_0 to DMATSRB_47) | 23-62 |
| 23.2.22 | DMA Channel Control Registers 0 to 47 (DMACHCR_0 to DMACHCR_47) | 23-63 |
| 23.2.23 | DMA Channel Control Register B_0 to 47 (DMACHCRB_0 to DMACHCRB_47) | 23-67 |
| 23.2.24 | DMA Buffer Control Registers 0 to 47 (DMABUFCR_0 to DMABUFCR_47) | 23-68 |
| 23.2.25 | DMA Extended Resource Selectors 0 to 47 (DMARS_0 to DMARS_47) | 23-69 |
| 23.2.26 | DMA Descriptor Base Address Registers 0 to 47 (DMADPBASE_0 to DMADPBASE_47) | 23-70 |
| 23.2.27 | DMA Descriptor Control Registers 0 to 47 (DMADPCR_0 to DMADPCR_47) | 23-71 |
| 23.2.28 | DMA Fixed Source Address Registers 0 to 47 (DMAFIXSAR_0 to DMAFIXSAR_47) | 23-71 |
| 23.2.29 | DMA Fixed Destination Address Registers 0 to 47 (DMAFIXDAR_0 to DMAFIXDAR_47) | 23-72 |
| 23.2.30 | DMA Fixed Descriptor Base Address Registers 0 to 47 (DMAFIXDPBASE_0 to DMAFIXDPBASE_47) | 23-72 |
| 23.2.31 | Descriptor Memory (DescriptorMEM) | 23-72 |
| 23.2.32 | Secure Function Secure Status Register for channel 0 to 15 (DMASES_0) | 23-73 |
| 23.2.33 | Secure Function Secure Status Register for channel 16 to 31 (DMASES_1) | 23-74 |
| 23.2.34 | Secure Function Secure Status Register for channel 32 to 47 (DMASES_2) | 23-75 |
| 23.2.35 | Secure Function Slave Error Address Register for channel 0 to 15 (DMASEA_0) | 23-76 |
| 23.2.36 | Secure Function Slave Error Address Register for channel 16 to 31 (DMASEA_1) | 23-76 |
| 23.2.37 | Secure Function Slave Error Address Register for channel 32 to 47 (DMASEA_2) | 23-77 |
| 23.2.38 | Secure Function Error Master ID Register for channel 0 to 15 (DMAEMID_0) | 23-77 |
| 23.2.39 | Secure Function Error Master ID Register for channel 16 to 31 (DMAEMID_1) | 23-78 |
| 23.2.40 | Secure Function Error Master ID Register for channel 32 to 47 (DMAEMID_2) | 23-78 |
| 23.3 | Operation | 23-79 |
| 23.3.1 | DMA Transfer Requests | 23-79 |
| 23.3.2 | Channel Priority | 23-84 |
| 23.3.3 | Slow Speed Mode | 23-84 |
| 23.3.4 | Descriptor Memory | 23-85 |
| 23.3.5 | Transmission Flow | 23-93 |

| | | |
|------------|---|-------------|
| 23.3.6 | Total Size Transmission..... | 23-95 |
| 23.3.7 | Rate access control..... | 23-95 |
| 23.4 | Secure function | 23-96 |
| 23.4.1 | Secure function | 23-96 |
| 23.5 | Usage Notes | 23-97 |
| 24. | Booting..... | 24-1 |
| 24.1 | Overview..... | 24-1 |
| 24.1.1 | Features..... | 24-1 |
| 24.1.2 | External Pins..... | 24-2 |
| 24.1.2.1 | MD Pins | 24-2 |
| 24.1.2.2 | HyperFlash Boot Pins | 24-4 |
| 24.1.2.3 | Serial Flash Boot Pins | 24-5 |
| 24.1.2.4 | SCIF and HSCIF Download Mode Pins..... | 24-5 |
| 24.1.2.5 | eMMC Boot Pins..... | 24-6 |
| 24.1.2.6 | USB Download Mode Pins | 24-6 |
| 24.1.3 | Connected Module..... | 24-7 |
| 24.2 | Operation | 24-8 |
| 24.2.1 | Boot Operation Overview | 24-8 |
| 24.2.2 | Boot Address Control | 24-9 |
| 24.2.3 | Boot Parameters | 24-10 |
| 24.2.4 | Hyper/Serial ROM Using DMA Boot Sequence | 24-11 |
| 24.2.5 | eMMC Using DMA Boot Sequence | 24-12 |
| 24.2.6 | SCIF Download Mode | 24-15 |
| 24.2.7 | USB download mode..... | 24-17 |
| 24.2.8 | Boot ROM..... | 24-18 |
| 24.3 | Usage Note..... | 24-19 |
| 24.3.1 | IPL Address and Data Size Limitation..... | 24-19 |
| 24.3.2 | Cleanup | 24-19 |
| 24.3.3 | Pull-up/down settings..... | 24-20 |
| 24.3.4 | Dual channel Serial Flash(Dual QSPI) boot | 24-20 |
| 25. | External Bus Controller for EX-Bus (LBSC)..... | 25-1 |
| 25.1 | Overview..... | 25-1 |
| 25.1.1 | Features..... | 25-1 |
| 25.1.2 | Block Diagram..... | 25-2 |
| 25.1.3 | LBSC Areas | 25-3 |
| 25.1.3.1 | LBSC Support Areas..... | 25-3 |
| 25.1.3.2 | Functionality Supported in Each Area | 25-4 |
| 25.1.4 | Register Configuration..... | 25-4 |
| 25.2 | Register Description..... | 25-6 |
| 25.2.1 | Area 0 Control Register (CS0CTRL)..... | 25-7 |
| 25.2.2 | Area 1 Control Register (CS1CTRL)..... | 25-8 |
| 25.2.3 | Area 0 RD/WE Pulse Control Register (CSWCR0) | 25-9 |
| 25.2.4 | Area 1 RD/WE Pulse Control Register (CSWCR1) | 25-11 |
| 25.2.5 | Area 0 External Wait Control Register (CSPWCR0) | 25-13 |
| 25.2.6 | Area 1 External Wait Control Register (CSPWCR1) | 25-14 |
| 25.2.7 | External Wait Input Control Register (EXWTSYNC)..... | 25-15 |
| 25.2.8 | Area 0 Burst Control Register (CS0BSTCTL) | 25-16 |
| 25.2.9 | Area 0 Burst Pitch Set Register (CS0BTPH)..... | 25-17 |
| 25.2.10 | Area 1 Guard Setting Register (CS1GDST) | 25-18 |
| 25.2.11 | BSC Interrupt Source Status Register (BCINTSR)..... | 25-19 |
| 25.2.12 | BSC Interrupt Source Clear Register (BCINTCR) | 25-20 |
| 25.2.13 | BSC Interrupt Enable Register (BCINTMR)..... | 25-21 |

| | | |
|----------|--|-------|
| 25.2.14 | External Wait Status Register (EXWTSTS) | 25-22 |
| 25.2.15 | EX-BUS Wait Timeout Detection Base Counter Register (EXBCT) | 25-23 |
| 25.2.16 | EX-BUS Wait Timeout Detection Counter Register (EXTCT) | 25-24 |
| 25.2.17 | EX-BUS Wait Timeout Detection Access Source Indication Register (EXTSR)..... | 25-25 |
| 25.2.18 | EX-BUS Wait Timeout Detection Address Indication Register (EXTADR) | 25-26 |
| 25.3 | Operation | 25-27 |
| 25.3.1 | SRAM Interface (Basic Functionality) | 25-27 |
| 25.3.2 | CPU (AXI Bus) → Burst ROM Interface | 25-33 |
| 25.3.3 | CPU (AXI Bus) → Byte Control SRAM Interface | 25-35 |
| 25.3.4 | Wait Timeout | 25-35 |
| 25.4 | Usage Notes | 25-36 |
| 25.4.1 | Pin Multiplexing | 25-36 |
| 26. | External Bus Controller for LPDDR4/DDR3L SDRAM (DBSC4) | 26-1 |
| 26.1 | Overview | 26-1 |
| 26.1.1 | Definition of Terms and Symbols | 26-2 |
| 26.1.2 | Block Diagram | 26-4 |
| 26.2 | Operation | 26-7 |
| 26.2.1 | Activation Sequence | 26-7 |
| 26.2.1.1 | Overview | 26-7 |
| 26.2.2 | Self-Refresh Sequence | 26-8 |
| 26.2.2.1 | Overview | 26-8 |
| 26.2.3 | SDRAM Power-Supply Backup Mode | 26-8 |
| 26.2.3.1 | Overview | 26-8 |
| 26.2.4 | Recovery from SDRAM Power-Supply Backup Mode | 26-9 |
| 26.2.4.1 | Overview | 26-9 |
| 26.3 | Clock Supply Path..... | 26-10 |
| 27. | System RAM..... | 27-1 |
| 27.1 | Overview..... | 27-1 |
| 27.1.1 | Features | 27-1 |
| 27.1.2 | Block Diagram | 27-1 |
| 27.2 | Operation | 27-2 |
| 27.2.1 | Address of System RAM | 27-2 |
| 28. | 3D Graphics Engine (3DGE) | 28-1 |
| 28.1 | Overview..... | 28-1 |
| 28.1.1 | Features | 28-1 |
| 28.1.2 | Block Diagram | 28-2 |
| 28.1.3 | Register Configuration..... | 28-4 |
| 28.1.4 | Connected Module | 28-4 |
| 28.2 | Operation | 28-5 |
| 28.2.1 | Initial Setting..... | 28-5 |
| 28.2.1.1 | Power and Clock Supply | 28-5 |
| 29. | CSI2 | 29-1 |
| 29.1 | Overview..... | 29-1 |
| 29.1.1 | Features | 29-2 |
| 29.1.2 | Block Diagram | 29-2 |
| 29.1.3 | External Pins | 29-6 |
| 29.1.4 | Register Configuration..... | 29-7 |
| 29.1.5 | Connected Module | 29-12 |
| 29.2 | Register Description..... | 29-13 |
| 29.2.1 | Control Timing Select Register (TREF) | 29-13 |

| | | |
|---------|--|-------|
| 29.2.2 | Software Reset Register (SRST)..... | 29-14 |
| 29.2.3 | PHY Operation Control Register (PHYCNT)..... | 29-15 |
| 29.2.4 | Checksum Control Register (CHKSUM)..... | 29-16 |
| 29.2.5 | Channel Data Type Select Register (VCDT)..... | 29-17 |
| 29.2.6 | Channel Data Type Select Register 2 (VCDT2)..... | 29-19 |
| 29.2.7 | Frame Data Type Select Register (FRDT)..... | 29-21 |
| 29.2.8 | Field Detection Control Register (FLD)..... | 29-22 |
| 29.2.9 | Automatic Standby Control Register (ASTBY)..... | 29-24 |
| 29.2.10 | Long Data Type Setting Register 0 (LNGDT0)..... | 29-26 |
| 29.2.11 | Long Data Type Setting Register 1 (LNGDT1)..... | 29-27 |
| 29.2.12 | Interrupt Enable Register (INTEN)..... | 29-28 |
| 29.2.13 | Interrupt Source Mask Register (INTCLOSE)..... | 29-31 |
| 29.2.14 | Interrupt Status Monitor Register (INTSTATE)..... | 29-34 |
| 29.2.15 | Interrupt Error Status Monitor Register (INTERRSTATE)..... | 29-41 |
| 29.2.16 | Short Packet Data Register (SHPDAT)..... | 29-46 |
| 29.2.17 | Short Packet Count Register (SHPCNT)..... | 29-47 |
| 29.2.18 | LINK Operation Control Register (LINKCNT)..... | 29-48 |
| 29.2.19 | Lane Swap Register (LSWAP)..... | 29-49 |
| 29.2.20 | PHY Test Interface Write Register (PHTW)..... | 29-51 |
| 29.2.21 | PHY Test Interface Read Register (PHTR)..... | 29-52 |
| 29.2.22 | PHY Test Interface Clear Register (PHTC)..... | 29-53 |
| 29.2.23 | PHY Frequency Control Register (PHYPLL)..... | 29-54 |
| 29.2.24 | PHY ESC Error Monitor Register (PHEERM)..... | 29-55 |
| 29.2.25 | PHY Clock Lane Monitor Register (PHCLM)..... | 29-57 |
| 29.2.26 | PHY Data Lane Monitor Register (PHDLM)..... | 29-58 |
| 29.2.27 | Packet Header 0 Monitor Register 0 (PH0M0)..... | 29-60 |
| 29.2.28 | Packet Header 0 Monitor Register 1 (PH0M1)..... | 29-61 |
| 29.2.29 | Packet Header 1 Monitor Register 0 (PH1M0)..... | 29-62 |
| 29.2.30 | Packet Header 1 Monitor Register 1 (PH1M1)..... | 29-63 |
| 29.2.31 | Packet Header 2 Monitor Register 0 (PH2M0)..... | 29-64 |
| 29.2.32 | Packet Header 2 Monitor Register 1 (PH2M1)..... | 29-65 |
| 29.2.33 | Packet Header 3 Monitor Register 0 (PH3M0)..... | 29-66 |
| 29.2.34 | Packet Header 3 Monitor Register 1 (PH3M1)..... | 29-67 |
| 29.2.35 | Packet Header R Monitor Register 0 (PHRM0)..... | 29-67 |
| 29.2.36 | Packet Header R Monitor Register 1 (PHRM1)..... | 29-68 |
| 29.2.37 | Packet Header R Monitor Register 2 (PHRM2)..... | 29-68 |
| 29.2.38 | Packet Header C Monitor Register 0 (PHCM0)..... | 29-69 |
| 29.2.39 | Packet Header C Monitor Register 1 (PHCM1)..... | 29-69 |
| 29.2.40 | CRC Monitor Register 0 (CRCM0)..... | 29-70 |
| 29.2.41 | CRC Monitor Register 1 (CRCM1)..... | 29-70 |
| 29.2.42 | SOT Error Count Register (SERRCNT)..... | 29-71 |
| 29.2.43 | SOTSYNC Error Count Register (SSERRCNT)..... | 29-72 |
| 29.2.44 | ECC_CRCT Count Register (ECCCM)..... | 29-73 |
| 29.2.45 | ECC_ERR Count Register (ECECM)..... | 29-74 |
| 29.2.46 | CRC_ERR Count Register (CRCECM)..... | 29-74 |
| 29.2.47 | Line Register (LCNT)..... | 29-75 |
| 29.2.48 | Line Register 2 (LCNT2)..... | 29-75 |
| 29.2.49 | Line Monitor Register (LCNTM)..... | 29-76 |
| 29.2.50 | Line Monitor Register 2 (LCNTM2)..... | 29-76 |
| 29.2.51 | Frame Count Monitor Register (FCNTM)..... | 29-77 |
| 29.2.52 | Frame Count Monitor Register 2 (FCNTM2)..... | 29-77 |
| 29.2.53 | PHY Data IN Monitor Register (PHYDIM)..... | 29-78 |
| 29.2.54 | PHY Input Monitor Register (PHYIM)..... | 29-79 |
| 29.2.55 | VIN Data Monitor Register (VINDM)..... | 29-81 |

| | | |
|---------|--|--------|
| 29.2.56 | VIN Signal Monitor Register 1 (VINSM1)..... | 29-82 |
| 29.2.57 | VIN Signal Monitor Register 2 (VINSM2)..... | 29-83 |
| 29.2.58 | VIN Signal Monitor Register 3 (VINSM3)..... | 29-84 |
| 29.2.59 | PHY Output Monitor Register (PHYOM) | 29-85 |
| 29.2.60 | Packet Header Monitor Registers 1 to 8 (PHM1 to PHM8)..... | 29-86 |
| 29.2.61 | Packet Period Monitor Registers 1 to 8 (PPM1 to PPM8) | 29-87 |
| 29.2.62 | CSI0CLK Frequency Configuration Preset Register (CSI0CLK)..... | 29-88 |
| 29.3 | Operation | 29-89 |
| 29.3.1 | Transfer Rate..... | 29-89 |
| 29.3.2 | ECC 1-Bit Error Correction and 2-Bit or More Error Detection of a Packet Header | 29-89 |
| 29.3.3 | CRC Error Detection of a Payload Data Part..... | 29-90 |
| 29.3.4 | Generation of VD (vertical sync), HD (horizontal sync), and FLD (field) Signals | 29-91 |
| 29.3.5 | Four-Channel Output | 29-92 |
| 29.3.6 | Interrupts..... | 29-93 |
| 29.3.7 | Lane Swapping | 29-97 |
| 29.3.8 | Initial Setting of PHY | 29-98 |
| 29.3.9 | PHY Control and Monitoring through Register Setting..... | 29-100 |
| 29.3.10 | Interrupt Generation by ULP Status, and Reception Error Status..... | 29-111 |
| 29.3.11 | Monitoring Function | 29-112 |
| 29.3.12 | Standby Mode..... | 29-113 |
| 29.3.13 | Software Reset | 29-113 |
| 29.3.14 | Input Clock Signals..... | 29-115 |
| 29.4 | Usage Notes | 29-116 |
| 29.4.1 | MIPI CSI-2 Transfer Rate..... | 29-116 |
| 29.4.2 | PHY Operation Control Register (PHYCNT)..... | 29-116 |
| 29.4.3 | Channel Data Type Select Register VCDT and Channel Data Type Select Register VCDT2 | 29-116 |
| 29.4.4 | Synchronization Error during HS (Hi-Speed) Reception..... | 29-116 |
| 29.4.5 | FLD Signal..... | 29-116 |
| 29.4.6 | INT_LESS_THAN_WC | 29-117 |
| 29.4.7 | Transfer Rate of Interleaved Transmission | 29-117 |
| 29.4.8 | Lane Setting | 29-118 |
| 29.4.9 | V-blanking Period..... | 29-118 |
| 29.4.10 | Notes on data | 29-118 |
| 30. | Video Input Module (VIN) | 30-1 |
| 30.1 | Overview..... | 30-1 |
| 30.1.1 | Features..... | 30-2 |
| 30.1.2 | Block Diagram..... | 30-8 |
| 30.1.3 | External Pins | 30-15 |
| 30.1.4 | Register Configuration..... | 30-19 |
| 30.1.5 | Connected Module | 30-47 |
| 30.2 | Register Description..... | 30-48 |
| 30.2.1 | Video n Main Control Register (VnMC) | 30-49 |
| 30.2.2 | Video n Module Status Register (VnMS)..... | 30-54 |
| 30.2.3 | Video n Frame Capture Register (VnFC) | 30-56 |
| 30.2.4 | Video n Start Line Pre-Clip Register (VnSLPrC)..... | 30-57 |
| 30.2.5 | Video n End Line Pre-Clip Register (VnELPrC)..... | 30-58 |
| 30.2.6 | Video n Start Pixel Pre-Clip Register (VnSPPrC) | 30-59 |
| 30.2.7 | Video n End Pixel Pre-Clip Register (VnEPPrC) | 30-60 |
| 30.2.8 | Video n CSI2 Interface Mode Register (VnCSI_IFMD) | 30-61 |
| 30.2.9 | Video n Image Stride Register (VnIS)..... | 30-68 |
| 30.2.10 | Video n Memory Base 1 Register (VnMB1) | 30-70 |
| 30.2.11 | Video n Memory Base 2 Register (VnMB2) | 30-71 |
| 30.2.12 | Video n Memory Base 3 Register (VnMB3) | 30-72 |

| | | |
|-----------|---|--------|
| 30.2.13 | Video n Line Count Register (VnLC)..... | 30-73 |
| 30.2.14 | Video n Interrupt Enable Register (VnIE)..... | 30-74 |
| 30.2.15 | Video n Interrupt Status Register (VnINTS)..... | 30-76 |
| 30.2.16 | Video n Scanline Interrupt Register (VnSI)..... | 30-79 |
| 30.2.17 | Video n Data Mode Register (VnDMR)..... | 30-80 |
| 30.2.18 | Video n Data Mode Register 2 (VnDMR2)..... | 30-84 |
| 30.2.19 | Video n UV Address Offset Register (VnUVAOF)..... | 30-86 |
| 30.2.20 | YC-RGB Conversion Coefficient Registers..... | 30-87 |
| 30.2.20.1 | Video n Color Space Change Coefficient 1 Register (VnCSCC1)..... | 30-88 |
| 30.2.20.2 | Video n Color Space Change Coefficient 2 Register (VnCSCC2)..... | 30-89 |
| 30.2.20.3 | Video n Color Space Change Coefficient 3 Register (VnCSCC3)..... | 30-90 |
| 30.2.20.4 | Video n YC → RGB Calculation Setting Extension Register 1 (VnCSCE1)..... | 30-91 |
| 30.2.20.5 | Video n YC → RGB Calculation Setting Extension Register 2 (VnCSCE2)..... | 30-92 |
| 30.2.20.6 | Video n YC → RGB Calculation Setting Extension Register 3 (VnCSRZ/G2E)..... | 30-93 |
| 30.2.20.7 | Video n YC → RGB Calculation Setting Extension Register 4 (VnCSCE4)..... | 30-94 |
| 30.2.21 | UDS Control Registers..... | 30-95 |
| 30.2.21.1 | Video n Scaling Control Register (VnUDS_CTRL)..... | 30-95 |
| 30.2.21.2 | Video n Scaling Factor Register (VnUDS_SCALE)..... | 30-97 |
| 30.2.21.3 | Video n Passband Register (VnUDS_PASS_BWIDTH)..... | 30-101 |
| 30.2.21.4 | Video n UDS Output Size Clipping Register (VnUDS_CLIP_SIZE)..... | 30-103 |
| 30.2.22 | Video n Lookup Table Pointer (VnLUTP)..... | 30-105 |
| 30.2.23 | Video n Lookup Table Data Register (VnLUTD)..... | 30-106 |
| 30.2.24 | RGB-YC Conversion Coefficient Registers..... | 30-107 |
| 30.2.24.1 | Video n RGB → Y Calculation Setting Register 1 (VnYCCR1)..... | 30-107 |
| 30.2.24.2 | Video n RGB → Y Calculation Setting Register 2 (VnYCCR2)..... | 30-108 |
| 30.2.24.3 | Video n RGB → Y Calculation Setting Register 3 (VnYCCR3)..... | 30-109 |
| 30.2.24.4 | Video n RGB → Cb Calculation Setting Register 1 (VnCBCCR1)..... | 30-110 |
| 30.2.24.5 | Video n RGB → Cb Calculation Setting Register 2 (VnCBCCR2)..... | 30-111 |
| 30.2.24.6 | Video n RGB → Cb Calculation Setting Register 3 (VnCBCCR3)..... | 30-112 |
| 30.2.24.7 | Video n RGB → Cr Calculation Setting Register 1 (VnCRCCR1)..... | 30-113 |
| 30.2.24.8 | Video n RGB → Cr Calculation Setting Register 2 (VnCRCCR2)..... | 30-114 |
| 30.2.24.9 | Video n RGB → Cr Calculation Setting Register 3 (VnCRCCR3)..... | 30-115 |
| 30.3 | Operation..... | 30-116 |
| 30.3.1 | Input Interface..... | 30-116 |
| 30.3.2 | Error Correction..... | 30-125 |
| 30.3.3 | Capture Mode..... | 30-125 |
| 30.3.4 | Size Clipping..... | 30-126 |
| 30.3.5 | Vertical Scaling..... | 30-126 |
| 30.3.6 | Horizontal Scaling..... | 30-126 |
| 30.3.7 | Color Conversion Function..... | 30-127 |
| 30.3.8 | Image Data Format Conversion Functions..... | 30-130 |
| 30.3.9 | Internal Field Signal Generation..... | 30-133 |
| 30.3.10 | Output Data Format..... | 30-134 |
| 30.3.11 | Endian Conversion..... | 30-145 |
| 30.3.12 | Module Standby Mode..... | 30-147 |
| 30.3.13 | Transition to Module Standby Mode..... | 30-147 |
| 30.3.14 | Cancellation of Module Standby Mode and Restarting of Video Input Module..... | 30-147 |
| 30.3.15 | Interrupts..... | 30-147 |
| 30.3.16 | Input and Output of YCbCr-422 Format..... | 30-149 |
| 30.3.17 | Initialize Procedure..... | 30-156 |
| 30.3.18 | Capture Stop Procedure..... | 30-158 |
| 30.3.19 | Capture Restarting procedure..... | 30-159 |
| 30.4 | Usage Notes..... | 30-161 |
| 30.4.1 | Specification..... | 30-161 |

| | | |
|-----------|---|--------|
| 30.4.2 | Limitations on Usage | 30-167 |
| 31. | Video Decoding Processor for inter-device video transfer (iVDPIC) | 31-1 |
| 31.1 | Overview..... | 31-1 |
| 31.1.1 | Features..... | 31-2 |
| 31.1.2 | Connected Module..... | 31-4 |
| 31.2 | Usage Notes | 31-5 |
| 31.2.1 | Limitations on Software Reset, Module Standby and power-shutoff | 31-5 |
| 31.2.2 | Note on input bitstream for decoding..... | 31-5 |
| 32. | Video Codec Processor (VCP4)..... | 32-1 |
| 32.1 | Overview..... | 32-1 |
| 32.1.1 | Features..... | 32-4 |
| 32.1.2 | Connected Module..... | 32-11 |
| 32.2 | Usage Notes | 32-12 |
| 32.2.1 | Limitations on Software Reset, Module Standby and power-shutoff | 32-12 |
| 32.2.2 | Note on input bitstream for decoding..... | 32-12 |
| 33. | Video Signal Processor (VSP2)..... | 33-1 |
| 33.1 | Overview..... | 33-1 |
| 33.1.1 | Features..... | 33-1 |
| 33.1.1.1 | Features of VSPI | 33-2 |
| 33.1.1.2 | Features of VSPBD [RZ/G2H] | 33-3 |
| 33.1.1.3 | Features of VSPBC [RZ/G2H]..... | 33-3 |
| 33.1.1.4 | Features of VSPB [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] | 33-4 |
| 33.1.1.5 | Features of VSPD..... | 33-5 |
| 33.1.1.6 | Features of VSPDL [RZ/G2H, RZ/G2N]..... | 33-5 |
| 33.1.2 | Block diagram..... | 33-7 |
| 33.1.2.1 | Block diagram of each VSP | 33-7 |
| 33.1.2.2 | Memory Access Unit (MAU)..... | 33-13 |
| 33.1.2.3 | Command Transfer Unit (CTU)..... | 33-13 |
| 33.1.2.4 | Read Pixel Formatter (RPF)..... | 33-13 |
| 33.1.2.5 | Data Path Router (DPR)..... | 33-14 |
| 33.1.2.6 | Super Resolution Unit (SRU)..... | 33-14 |
| 33.1.2.7 | Up Down Scalar (UDS) | 33-14 |
| 33.1.2.8 | Look Up Table (LUT)..... | 33-14 |
| 33.1.2.9 | Blend ROP Unit (BRU) | 33-15 |
| 33.1.2.10 | Blend ROP Sub Unit (BRS)..... | 33-16 |
| 33.1.2.11 | Write Pixel Formatter (WPF) | 33-17 |
| 33.1.2.12 | Cubic Look Up table (CLU)..... | 33-17 |
| 33.1.2.13 | Hue Saturation value Transform (HST) | 33-17 |
| 33.1.2.14 | Hue Saturation value Inverse transform (HSI)..... | 33-17 |
| 33.1.2.15 | Histogram Generator -One dimension (HGO) | 33-17 |
| 33.1.2.16 | Histogram Generator-Two dimension (HGT) | 33-18 |
| 33.1.2.17 | Display Unit InterFace (LIF)..... | 33-18 |
| 33.1.2.18 | SHarPness (SHP) | 33-18 |
| 33.1.2.19 | InterLeaVer (ILV)..... | 33-18 |
| 33.1.2.20 | Detail Function..... | 33-18 |
| 33.1.3 | External Pins | 33-24 |
| 33.1.4 | Register Configuration..... | 33-24 |
| 33.1.5 | Connected Module..... | 33-38 |
| 33.2 | Register Description..... | 33-39 |
| 33.2.1 | Notational Conventions for Registers and Bit Fields..... | 33-39 |
| 33.2.2 | Register Classification | 33-39 |

| | | |
|-----------|--|--------|
| 33.2.3 | Restrictions on Access to Registers and Lookup Tables..... | 33-40 |
| 33.2.4 | General Control Registers..... | 33-41 |
| 33.2.4.1 | VSP2 Start Registers n (VI6_CMDn: n = 0, 1)..... | 33-41 |
| 33.2.4.2 | Clock Control Register 0 (VI6_CLK_CTRL0)..... | 33-44 |
| 33.2.4.3 | Clock Control Register 1 (VI6_CLK_CTRL1)..... | 33-45 |
| 33.2.4.4 | Dynamic Clock Stop Control Register (VI6_CLK_DCSWT)..... | 33-46 |
| 33.2.4.5 | Dynamic Clock Stop Disable Register 0 (VI6_CLK_DCSM0)..... | 33-47 |
| 33.2.4.6 | Dynamic Clock Stop Disable Register 1 (VI6_CLK_DCSM1)..... | 33-48 |
| 33.2.4.7 | Software Reset Register (VI6_SRESET)..... | 33-49 |
| 33.2.4.8 | Module Reset Enable Register 0 (VI6_MRESET_ENB0)..... | 33-50 |
| 33.2.4.9 | Module Reset Enable Register 1 (VI6_MRESET_ENB1)..... | 33-51 |
| 33.2.4.10 | Module Reset Issuing Register (VI6_MRESET)..... | 33-52 |
| 33.2.4.11 | Operating Status Register (VI6_STATUS)..... | 33-53 |
| 33.2.4.12 | WPFn Interrupt Enable Registers (VI6_WPFn_IRQ_ENB: n = 0, 1)..... | 33-54 |
| 33.2.4.13 | WPFn Interrupt Status Registers (VI6_WPFn_IRQ_STA: n = 0, 1)..... | 33-55 |
| 33.2.4.14 | Display-n Interrupt Enable Register (VI6_DISPn_IRQ_ENB: n = 0, 1)..... | 33-57 |
| 33.2.4.15 | Display-n Interrupt Status Register (VI6_DISPn_IRQ_STA: n = 0, 1)..... | 33-58 |
| 33.2.5 | Display List Control Registers..... | 33-59 |
| 33.2.5.1 | Display List Control Register (VI6_DL_CTRL)..... | 33-59 |
| 33.2.5.2 | Display List-n Header Address Register (VI6_DL_HDR_ADDRn: n = 0, 1)..... | 33-61 |
| 33.2.5.3 | Display List-n Data Swapping Register (VI6_DL_SWAPn: n = 0, 1)..... | 33-62 |
| 33.2.5.4 | Extended Display List-n Control Register (VI6_DL_EXT_CTRLn: n = 0, 1)..... | 33-64 |
| 33.2.5.5 | Display List Body Size Register-n (VI6_DL_BODY_SIZE_n: n = 0)..... | 33-65 |
| 33.2.5.6 | Display List-n Header Reference Address Register (VI6_DL_HDR_REF_ADDRn: n = 0, 1)..... | 33-66 |
| 33.2.6 | RPF Control Registers..... | 33-67 |
| 33.2.6.1 | RPFn Basic Read Size Registers (VI6_RPFn_SRC_BSIZE)..... | 33-67 |
| 33.2.6.2 | RPFn Extended Read Size Registers (VI6_RPFn_SRC_ESIZE)..... | 33-69 |
| 33.2.6.3 | RPFn Input Format Registers (VI6_RPFn_INFMT)..... | 33-70 |
| 33.2.6.4 | RPFn Data Swapping Registers (VI6_RPFn_DSWAP)..... | 33-80 |
| 33.2.6.5 | RPFn Display Location Registers (VI6_RPFn_LOC)..... | 33-82 |
| 33.2.6.6 | RPFn α Plane Selection Control Registers (VI6_RPFn_ALPH_SEL)..... | 33-84 |
| 33.2.6.7 | RPFn Virtual Plane Color Information Registers (VI6_RPFn_VRTCOL_SET)..... | 33-91 |
| 33.2.6.8 | RPFn Mask Control Registers (VI6_RPFn_MSKCTRL)..... | 33-92 |
| 33.2.6.9 | RPFn IROP-SRC Input Value Registers 0 (VI6_RPFn_MSKSET0)..... | 33-94 |
| 33.2.6.10 | RPFn IROP-SRC Input Value Registers 1 (VI6_RPFn_MSKSET1)..... | 33-95 |
| 33.2.6.11 | RPFn Color Keying Control Registers (VI6_RPFn_CKEY_CTRL)..... | 33-96 |
| 33.2.6.12 | RPFn Color Keying Color Setting Registers-m (VI6_RPFn_CKEY_SETm: m = 0, 1)..... | 33-98 |
| 33.2.6.13 | RPFn Source Picture Memory Stride Setting Registers (VI6_RPFn_SRCM_PSTRIDE)..... | 33-100 |
| 33.2.6.14 | RPFn Source α Memory Stride Setting Registers (VI6_RPFn_SRCM_ASTRIDE)..... | 33-102 |
| 33.2.6.15 | RPFn Source Y/RGB Address Registers (VI6_RPFn_SRCM_ADDR_Y)..... | 33-103 |
| 33.2.6.16 | RPFn Source Chroma Address Registers 0 (VI6_RPFn_SRCM_ADDR_C0)..... | 33-104 |
| 33.2.6.17 | RPFn Source Chroma Address Registers 1 (VI6_RPFn_SRCM_ADDR_C1)..... | 33-105 |
| 33.2.6.18 | RPFn Source α Address Registers (VI6_RPFn_SRCM_ADDR_AI)..... | 33-106 |
| 33.2.6.19 | RPFn Bus Access Control Registers (VI6_RPFn_BAC)..... | 33-107 |
| 33.2.6.20 | RPFn Multiple Alpha Control (VI6_RPFn_MULT_ALPH)..... | 33-108 |
| 33.2.7 | WPF Control Registers..... | 33-110 |
| 33.2.7.1 | WPFn-Source-RPF Registers (VI6_WPFn_SRCRPF)..... | 33-110 |
| 33.2.7.2 | WPFn Horizontal Input Size Clipping Registers (VI6_WPFn_HSZCLIP)..... | 33-112 |
| 33.2.7.3 | WPFn Vertical Input Size Clipping Registers (VI6_WPFn_VSZCLIP)..... | 33-114 |
| 33.2.7.4 | WPFn Output Format Registers (VI6_WPFn_OUTFMT)..... | 33-116 |
| 33.2.7.5 | WPFn Data Swapping Registers (VI6_WPFn_DSWAP)..... | 33-125 |
| 33.2.7.6 | WPFn Rounding Control Registers (VI6_WPFn_RNDCTRL)..... | 33-126 |
| 33.2.7.7 | WPF0 Rotation Control Registers (VI6_WPF0_ROT_CTRL)..... | 33-128 |

| | | |
|------------|--|--------|
| 33.2.7.8 | WPFn Destination Y Plane Memory Stride Registers (VI6_WPFn_DSTM_STRIDE_Y) . | 33-129 |
| 33.2.7.9 | WPFn Destination C Plane Memory Stride Registers (VI6_WPFn_DSTM_STRIDE_C) . | 33-130 |
| 33.2.7.10 | WPFn Destination Y/RGB Address Registers (VI6_WPFn_DSTM_ADDR_Y) | 33-131 |
| 33.2.7.11 | WPFn Destination Chroma Address Registers 0 (VI6_WPFn_DSTM_ADDR_C0)..... | 33-134 |
| 33.2.7.12 | WPFn Destination Chroma Address Registers 1 (VI6_WPFn_DSTM_ADDR_C1)..... | 33-135 |
| 33.2.7.13 | WPFn LIF Write Back Control Registers (VI6_WPFn_WRBCK_CTRL: n = 0, 1) | 33-136 |
| 33.2.8 | DPR Control Registers..... | 33-137 |
| 33.2.8.1 | Concept of DPR Settings..... | 33-137 |
| 33.2.8.2 | RPFn Routing Register (VI6_DPR_RPFn_ROUTE : n = 0, 1, 2, 3, 4)..... | 33-141 |
| 33.2.8.3 | WPFn Timing Control Register (VI6_DPR_WPFn_FPORCH: n = 0, 1) | 33-142 |
| 33.2.8.4 | {mod} Routing Register (VI6_DPR_{mod}_ROUTE: {mod} = ILV_BRS, SRU, UDS, LUT, CLU, HST, HSI, BRU, SHP) | 33-143 |
| 33.2.8.5 | {mod} Sampling Point Register (VI6_DPR_{mod}_SMPPT: {mod} = HGO, HGT)..... | 33-144 |
| 33.2.9 | SRU Control Registers..... | 33-145 |
| 33.2.9.1 | Super Resolution Control Register 0 (VI6_SRU_CTRL0) | 33-145 |
| 33.2.9.2 | Super Resolution Control Register 1 (VI6_SRU_CTRL1) | 33-147 |
| 33.2.9.3 | Super Resolution Control Register 2 (VI6_SRU_CTRL2) | 33-148 |
| 33.2.10 | UDS Control Registers | 33-149 |
| 33.2.10.1 | Scaling Control Registers (VI6_UDS_CTRL)..... | 33-149 |
| 33.2.10.2 | Scaling Factor Registers (VI6_UDS_SCALE) | 33-152 |
| 33.2.10.3 | α Data Threshold Setting Registers (VI6_UDS_ALPTH) | 33-155 |
| 33.2.10.4 | α Data Replacing Value Setting Registers (VI6_UDS_ALPVAL) | 33-156 |
| 33.2.10.5 | Passband Registers (VI6_UDS_PASS_BWIDTH)..... | 33-157 |
| 33.2.10.6 | Scaling Filter Horizontal Phase Registers (VI6_UDS_HPHASE)..... | 33-158 |
| 33.2.10.7 | 2D IPC Setting Register (VI6_UDS_IPC) | 33-159 |
| 33.2.10.8 | UDS Horizontal Input Clipping Registers (VI6_UDS_HSZCLIP)..... | 33-161 |
| 33.2.10.9 | UDS Output Size Clipping Registers (VI6_UDS_CLIP_SIZE) | 33-162 |
| 33.2.10.10 | Color Fill Register (VI6_UDS_FILL_COLOR) | 33-164 |
| 33.2.11 | LUT Control Register | 33-165 |
| 33.2.11.1 | LUT Control Register (VI6_LUT_CTRL)..... | 33-165 |
| 33.2.12 | CLU Control Register | 33-168 |
| 33.2.12.1 | CLU Control Register (VI6_CLU_CTRL) | 33-168 |
| 33.2.13 | HST Control Register | 33-172 |
| 33.2.13.1 | HST Control Register (VI6_HST_CTRL) | 33-172 |
| 33.2.14 | HSI Control Register..... | 33-173 |
| 33.2.14.1 | HSI Control Register (VI6_HSI_CTRL) | 33-173 |
| 33.2.15 | BRU Control Registers | 33-174 |
| 33.2.15.1 | BRU Input Control Register (VI6_BRU_INCTRL) | 33-174 |
| 33.2.15.2 | Size Register of BRU Input Virtual RPF (VI6_BRU_VIRRRPF_SIZE) | 33-178 |
| 33.2.15.3 | Display Location Register of BRU Input Virtual RPF (VI6_BRU_VIRRRPF_LOC) | 33-179 |
| 33.2.15.4 | Color Information Register of BRU Input Virtual RPF (VI6_BRU_VIRRRPF_COL) | 33-180 |
| 33.2.15.5 | BRU Control Registers (VI6_BRUm_CTRL: m = A, B, C, D, E) | 33-182 |
| 33.2.15.6 | BRU Blend Control Registers (VI6_BRUm_BLD: m = A, B, C, D, E) | 33-184 |
| 33.2.15.7 | BRU Raster Operation Control Register (VI6_BRU_ROP) | 33-186 |
| 33.2.16 | HGO Control Registers..... | 33-187 |
| 33.2.16.1 | HGO Detection Window Offset Register (VI6_HGO_OFFSET) | 33-187 |
| 33.2.16.2 | HGO Detection Window Size Register (VI6_HGO_SIZE) | 33-189 |
| 33.2.16.3 | HGO Mode Register (VI6_HGO_MODE) | 33-190 |
| 33.2.16.4 | HGO LB Detection Threshold Register (VI6_HGO_LB_TH) | 33-194 |
| 33.2.16.5 | HGO Horizontal Position Register for LB Detection Zone-n (VI6_HGO_LBn_H: n = 0, 1, 2, 3) | 33-195 |
| 33.2.16.6 | HGO Vertical Position Register for LB Detection Zone-n (VI6_HGO_LBn_V: n = 0, 1, 2, 3) | 33-197 |

| | | |
|------------|---|--------|
| 33.2.16.7 | HGO Component-m Histogram Register (VI6_HGO_m_HISTO_n: n = 0 to 63: m = R, G, B) | 33-198 |
| 33.2.16.8 | HGO Component-m Min/Max Value Register (VI6_HGO_m_MAXMIN: m = R, G, B) | 33-200 |
| 33.2.16.9 | HGO Component-m Sum Register (VI6_HGO_m_SUM: m = R, G, B) | 33-201 |
| 33.2.16.10 | HGO Component-m LB Detection Result Register (VI6_HGO_m_LB_DET: m = R, G, B) | 33-202 |
| 33.2.16.11 | HGO Y/MaxRGB Extended Histogram Address Register (VI6_HGO_EXT_HIST_ADDR) | 33-203 |
| 33.2.16.12 | HGO Y/MaxRGB Extended Histogram Data Register (VI6_HGO_EXT_HIST_DATA) | 33-204 |
| 33.2.16.13 | HGO Write Buffer Side (VI6_HGO_WBUFS) | 33-205 |
| 33.2.16.14 | HGO Read Buffer Side (VI6_HGO_RBUFS) | 33-206 |
| 33.2.16.15 | HGO Histogram Storing Address Register (VI6_HGO_HISTADD) | 33-207 |
| 33.2.16.16 | HGO Histogram Swapping Register (VI6_HGO_HSWAP) | 33-208 |
| 33.2.16.17 | HGO Parameter Register Reset (VI6_HGO_REGRST) | 33-209 |
| 33.2.17 | HGT Control Registers | 33-210 |
| 33.2.17.1 | HGT Detection Window Offset Register (VI6_HGT_OFFSET) | 33-210 |
| 33.2.17.2 | HGT Detection Window Size Register (VI6_HGT_SIZE) | 33-212 |
| 33.2.17.3 | HGT Mode Register (VI6_HGT_MODE) | 33-213 |
| 33.2.17.4 | HGT Hue Area Register (VI6_HGT_HUE_AREA_n: n = 0 to 5) | 33-216 |
| 33.2.17.5 | HGT LB Detection Threshold Register (VI6_HGT_LB_TH) | 33-218 |
| 33.2.17.6 | HGT Horizontal Position Register for LB Detection Zone-n (VI6_HGT_LBn_H : n = 0, 1, 2, 3) | 33-219 |
| 33.2.17.7 | HGT Vertical Position Register for LB Detection Zone-n (VI6_HGT_LBn_V: n = 0, 1, 2, 3) | 33-221 |
| 33.2.17.8 | HGT Histogram Register (VI6_HGT_HISTO_m_n: m = 0 to 5, n = 0 to 31) | 33-222 |
| 33.2.17.9 | HGT Max/Min Value Register (VI6_HGT_MAXMIN) | 33-223 |
| 33.2.17.10 | HGT Sum Register (VI6_HGT_SUM) | 33-224 |
| 33.2.17.11 | HGT LB Detection Result Register (VI6_HGT_LB_DET) | 33-225 |
| 33.2.17.12 | HGT Parameter Register Reset (VI6_HGT_REGRST) | 33-226 |
| 33.2.17.13 | HGT Write Buffer Side (VI6_HGT_WBUFS) | 33-227 |
| 33.2.17.14 | HGT Read Buffer Side (VI6_HGT_RBUFS) | 33-228 |
| 33.2.17.15 | HGT Histogram Storing Address Register (VI6_HGT_HISTADD) | 33-229 |
| 33.2.17.16 | HGT Histogram Swapping Register (VI6_HGT_HSWAP) | 33-230 |
| 33.2.18 | ILV Control Register | 33-231 |
| 33.2.18.1 | ILV Control Register (VI6_ILV_CTRL) | 33-231 |
| 33.2.19 | BRS Control Registers | 33-233 |
| 33.2.19.1 | BRS Input Control Register (VI6_BRS_INCTRL) | 33-233 |
| 33.2.19.2 | Size Register of BRS Input Virtual RPF (VI6_BRS_VIRRRPF_SIZE) | 33-236 |
| 33.2.19.3 | Display Location Register of BRS Input Virtual RPF (VI6_BRS_VIRRRPF_LOC) | 33-237 |
| 33.2.19.4 | Color Information Register of BRS Input Virtual RPF (VI6_BRS_VIRRRPF_COL) | 33-238 |
| 33.2.19.5 | BRS Control Registers (VI6_BRSm_CTRL: m = A, B) | 33-239 |
| 33.2.19.6 | BRS Blend Control Registers (VI6_BRSm_BLD: m = A, B) | 33-241 |
| 33.2.20 | LIF Control Registers | 33-243 |
| 33.2.20.1 | LIFn Control Register (VI6_LIFn_CTRL) | 33-243 |
| 33.2.20.2 | LIFn Clock Stop Buffer Control Register (VI6_LIFn_CSBTH) | 33-245 |
| 33.2.20.3 | LIF0 Buffer Attribute Register (VI6_LIF0_LBA) | 33-246 |
| 33.2.21 | SHP control registers | 33-247 |
| 33.2.21.1 | SHP Control Register0 (VI6_SHP_CTRL0) | 33-247 |
| 33.2.21.2 | SHP Control Register1 (VI6_SHP_CTRL1) | 33-250 |
| 33.2.21.3 | SHP Control Register2 (VI6_SHP_CTRL2) | 33-251 |
| 33.2.22 | VSP internal registers | 33-252 |
| 33.2.22.1 | IP Version Register (VI6_IP_VERSION) | 33-252 |
| 33.2.23 | RPFn OSD-CLUT Table | 33-253 |
| 33.2.24 | LUT table | 33-253 |

| | | |
|----------|--|--------|
| 33.2.25 | CLU table address register and table data register | 33-253 |
| 33.3 | Operation | 33-254 |
| 33.3.1 | Registers to set fixed value | 33-254 |
| 33.3.2 | Concept of VSP2 Operation Starting and Stopping | 33-254 |
| 33.3.3 | Display List | 33-257 |
| 33.3.3.1 | Functional Description | 33-257 |
| 33.3.3.2 | Normal Display List Mode | 33-259 |
| 33.3.3.3 | Header-less Display List Mode | 33-265 |
| 33.3.3.4 | Restrictions on Display List Usage | 33-265 |
| 33.3.4 | Interrupt Processing | 33-265 |
| 33.3.5 | Lookup Table Settings | 33-266 |
| 33.3.5.1 | OSD-CLUT or LUT | 33-266 |
| 33.3.5.2 | CLU Table | 33-267 |
| 33.3.6 | Linked with DU | 33-270 |
| 33.3.6.1 | Operation flow of VSPD and DU | 33-270 |
| 33.3.6.2 | Controlling Two Register Planes Using Display Lists | 33-273 |
| 33.3.6.3 | Auto display function (AUTO-DISP) | 33-275 |
| 33.3.6.4 | Auto field function (AUTO-FLD) | 33-283 |
| 33.3.7 | Image partition for VSPI processing | 33-288 |
| 33.3.7.1 | Necessity of image partition | 33-288 |
| 33.3.7.2 | Auto execution by using display list | 33-290 |
| 33.3.7.3 | Start Reservation for VSPI | 33-292 |
| 33.3.7.4 | Reading overlapped pixels and clipping discontinuous line | 33-296 |
| 33.3.7.5 | The number of taps for each function's filter in horizontal direction | 33-297 |
| 33.3.7.6 | Horizontal size clipping registers for image partition | 33-305 |
| 33.3.7.7 | Histogram of HGO/HGT | 33-306 |
| 33.3.7.8 | Restriction when VSPI image partition is used | 33-307 |
| 33.3.8 | Start Reservation for VSPBD, VSPBC, VSPBS or VSPB | 33-308 |
| 33.3.9 | The step to read HGO histogram registers of 256 step mode by indirect addressing | 33-310 |
| 33.4 | Usage Notes | 33-311 |
| 33.4.1 | Assignment in Memory Space | 33-311 |
| 33.4.2 | Limitations on Software Reset and Module Standby | 33-311 |
| 33.4.3 | Input Image Size | 33-311 |
| 33.4.4 | Output Image Size | 33-314 |
| 33.4.5 | Configuration of VSP2 | 33-316 |
| 33.4.6 | Performance of VSP2 | 33-321 |
| 33.4.6.1 | Performance of VSPI | 33-321 |
| 33.4.6.2 | Performance of VSPBD, VSPBC or VSPB | 33-321 |
| 33.4.6.3 | Performance of VSPD or VSPDL | 33-322 |
| 33.4.7 | Legend of terms | 33-323 |
| 34. | Fine Display Processor (FDP1) | 34-1 |
| 34.1 | Overview | 34-1 |
| 34.1.1 | Features | 34-1 |
| 34.1.2 | Block Diagram | 34-3 |
| 34.1.2.1 | ABI (AXI Bus Interface) | 34-3 |
| 34.1.2.2 | RPF (Read Pixel Formatter) | 34-3 |
| 34.1.2.3 | IPC (Interlace to Progressive Converter) | 34-4 |
| 34.1.2.4 | WPF (Write Pixel Formatter) | 34-4 |
| 34.1.2.5 | Input/output Data Format | 34-4 |
| 34.1.3 | Register Configuration | 34-6 |
| 34.1.4 | Connected Module | 34-10 |
| 34.2 | Register Description | 34-11 |
| 34.2.1 | Access Restrictions | 34-11 |

| | | |
|-----------|--|-------|
| 34.2.2 | Register Type Definitions | 34-11 |
| 34.2.3 | Notational Conventions for Registers and Bit Fields | 34-11 |
| 34.2.4 | Register Classification | 34-11 |
| 34.2.5 | Restrictions on Access to Registers and Lookup Tables..... | 34-12 |
| 34.2.6 | General Control Registers..... | 34-13 |
| 34.2.6.1 | FDP1 Start Register (FD1_CTL_CMD) | 34-13 |
| 34.2.6.2 | Sync Generator Register (FD1_CTL_SGCMD) | 34-14 |
| 34.2.6.3 | Register Set End Register (FD1_CTL_REGEND) | 34-15 |
| 34.2.6.4 | Channel Activation Register (FD1_CTL_CHACT)..... | 34-16 |
| 34.2.6.5 | Operation Mode Register (FD1_CTL_OPMODE) | 34-17 |
| 34.2.6.6 | V-Period Register (FD1_CTL_VPERIOD) | 34-21 |
| 34.2.6.7 | Clock Control Register (FD1_CTL_CLKCTRL) | 34-22 |
| 34.2.6.8 | Software Reset Register (FD1_CTL_SRESET)..... | 34-23 |
| 34.2.6.9 | Operating Status Register (FD1_CTL_STATUS)..... | 34-24 |
| 34.2.6.10 | V-Cycles Status Register (FD1_CTL_VCYCLE_STAT)..... | 34-25 |
| 34.2.6.11 | Interrupt Enable Register (FD1_CTL_IRQENB)..... | 34-26 |
| 34.2.6.12 | Interrupt Status Register (FD1_CTL_IRQSTA) | 34-27 |
| 34.2.6.13 | Interrupt Control Register (FD1_CTL_IRQFSET)..... | 34-28 |
| 34.2.7 | RPF Control Registers | 34-29 |
| 34.2.7.1 | Source Picture Size Register (FD1_RPF_SIZE) | 34-29 |
| 34.2.7.2 | Source Picture Format Register (FD1_RPF_FORMAT) | 34-30 |
| 34.2.7.3 | Source Picture Stride Register (FD1_RPF_PSTRIDE)..... | 34-36 |
| 34.2.7.4 | RPFn Source Component-m Address Register (FD1_RPFn_ADDR_m: {n,m} is one of {0,Y},{1,Y},{1,C0},{1,C1},{2,Y}) | 34-37 |
| 34.2.7.5 | Still Mask Address Register (FD1_RPF_SMSK_ADDR) | 34-38 |
| 34.2.7.6 | RPF Data Swap Register (FD1_RPF_SWAP) | 34-40 |
| 34.2.8 | WPF Control Registers | 34-42 |
| 34.2.8.1 | Destination Picture Format Register (FD1_WPF_FORMAT) | 34-42 |
| 34.2.8.2 | Destination Picture Rounding Control Register (FD1_WPF_RNDCTL) | 34-45 |
| 34.2.8.3 | Destination Picture Stride Register (FD1_WPF_PSTRIDE) | 34-46 |
| 34.2.8.4 | Destination Component-m Address Register (FD1_WPF_ADDR_m: m = Y, C0, C1) | 34-47 |
| 34.2.8.5 | WPF Data Swap Register (FD1_WPF_SWAP)..... | 34-48 |
| 34.2.9 | IPC Control Registers | 34-49 |
| 34.2.9.1 | IPC Mode Register (FD1_IPC_MODE) | 34-50 |
| 34.2.9.2 | Still Mask Threshold Register (FD1_IPC_SMSK_THRESH)..... | 34-51 |
| 34.2.9.3 | Comb Detection Parameter Register (FD1_IPC_COMB_DET)..... | 34-52 |
| 34.2.9.4 | Motion Decision Parameter Register (FD1_IPC_MOTDEC)..... | 34-53 |
| 34.2.9.5 | DLI Blend Parameter Register (FD1_IPC_DLI_BLEND) | 34-54 |
| 34.2.9.6 | DLI Horizontal Frequency Gain Register (FD1_IPC_DLI_HGAIN) | 34-55 |
| 34.2.9.7 | DLI Suppression Parameter register (FD1_IPC_DLI_SPRS)..... | 34-56 |
| 34.2.9.8 | DLI Angle Parameter Register (FD1_IPC_DLI_ANGLE)..... | 34-57 |
| 34.2.9.9 | DLI Isolated Pixel Parameter Register0 (FD1_IPC_DLI_ISOPIX0)..... | 34-58 |
| 34.2.9.10 | DLI Isolated Pixel Parameter Register1 (FD1_IPC_DLI_ISOPIX1)..... | 34-59 |
| 34.2.9.11 | IPC Sensor Threshold Register0 (FD1_IPC_SENSOR_TH0)..... | 34-60 |
| 34.2.9.12 | IPC Sensor Threshold Register1 (FD1_IPC_SENSOR_TH1)..... | 34-61 |
| 34.2.9.13 | Sensor Control Register0 (FD1_SENSOR_CTL0) | 34-62 |
| 34.2.9.14 | Sensor Control Register1 (FD1_SENSOR_CTL1) | 34-63 |
| 34.2.9.15 | Sensor Control Register2 (FD1_SENSOR_CTL2)..... | 34-64 |
| 34.2.9.16 | Sensor Control Register3 (FD1_SENSOR_CTL3)..... | 34-65 |
| 34.2.9.17 | Sensor Register (FD1_SENSOR_m: m = 0, 1...17) | 34-66 |
| 34.2.9.18 | Line Memory Pixel Number Register (FD1_IPC_LMEM) | 34-67 |
| 34.2.9.19 | IP Internal Data (FD1_IP_INTDATA) | 34-68 |
| 34.2.10 | LUT..... | 34-69 |
| 34.2.10.1 | LUT DIF_ADJ | 34-69 |

| | | |
|-----------|--|-------|
| 34.2.10.2 | LUT SAD_ADJ..... | 34-69 |
| 34.2.10.3 | LUT BLD_GAIN..... | 34-69 |
| 34.2.10.4 | LUT DIF_GAIN..... | 34-69 |
| 34.2.10.5 | LUT MDET..... | 34-69 |
| 34.3 | Operation | 34-70 |
| 34.3.1 | Procedure of FDP1 Start and Stop | 34-70 |
| 34.3.1.1 | Register usages for FDP1 Start and Stop | 34-70 |
| 34.3.1.2 | Register setting order for FDP1 Start and Stop..... | 34-72 |
| 34.3.2 | Look Up Table (LUT) Setting | 34-73 |
| 34.4 | Usage Notes | 34-92 |
| 34.4.1 | Assignment in Memory Space..... | 34-92 |
| 34.4.2 | FCPF connection..... | 34-92 |
| 34.4.3 | Manipulate FDP with module stop, software reset of CPG | 34-92 |
| 34.4.4 | Terminology..... | 34-92 |
| 35. | Frame Compression Processor (FCP)..... | 35-1 |
| 35.1 | Overview..... | 35-1 |
| 35.2 | FCP for Codec (FCPC)..... | 35-2 |
| 35.2.1 | Features..... | 35-3 |
| 35.2.2 | Connected Module..... | 35-3 |
| 35.2.3 | Usage Notes | 35-4 |
| 35.2.3.1 | Limitations on Software Reset, Module Standby and power-shutoff..... | 35-4 |
| 35.3 | FCP for VSP (FCPV)..... | 35-5 |
| 35.3.1 | Features..... | 35-5 |
| 35.3.2 | Block Diagram..... | 35-7 |
| 35.3.3 | IP Set and Bus Access Functions..... | 35-7 |
| 35.3.4 | Register Configuration..... | 35-8 |
| 35.3.5 | Connected Module..... | 35-9 |
| 35.3.6 | Register Description..... | 35-10 |
| 35.3.6.1 | FCP_VCR: FCP Version Control Register | 35-10 |
| 35.3.6.2 | FCP_CFG0: FCPV Configuration Register 0 | 35-11 |
| 35.3.6.3 | FCP_RST: FCP Reset Register | 35-12 |
| 35.3.6.4 | FCP_STA: FCP Status Register | 35-13 |
| 35.3.7 | Operation | 35-14 |
| 35.3.7.1 | Operation Control Setting IP set VSPI+FCPVI | 35-14 |
| 35.3.7.2 | Operation Control Setting for IP set VSPB+FCPVB | 35-15 |
| 35.3.7.3 | Operation Control Setting for IP set VSPD+FCPVD, VSPDL+FCPVD..... | 35-16 |
| 35.3.7.4 | Reset Operation..... | 35-17 |
| 35.3.7.5 | Register Settings and Restrictions on Connected VSP..... | 35-17 |
| 35.3.8 | Memory Layout of FCNL Image Data..... | 35-21 |
| 35.3.8.1 | In Cases of VI6_WPF0_OUTFMT.ROT [2:0] = 0, 1, 2, 3, 6, 7 | 35-21 |
| 35.3.8.2 | In Cases of VI6_WPF0_OUTFMT.ROT [2:0] = 4, 5 | 35-22 |
| 35.3.9 | Usage Notes | 35-23 |
| 35.3.9.1 | Manipulating FCPV+VSP with CPG module stop, software reset | 35-23 |
| 35.4 | FCP for FDP (FCPF) | 35-24 |
| 35.4.1 | Features..... | 35-24 |
| 35.4.2 | Block Diagram..... | 35-25 |
| 35.4.3 | IP set and bus access functions | 35-25 |
| 35.4.4 | Register Configuration..... | 35-26 |
| 35.4.5 | Connected Module..... | 35-28 |
| 35.4.6 | Register Descriptions..... | 35-29 |
| 35.4.6.1 | FCP_VCR: FCP Version Control Register | 35-29 |
| 35.4.6.2 | FCP_RST: FCP Reset Register | 35-30 |
| 35.4.6.3 | FCP_STA: FCP Status Register..... | 35-31 |

| | | |
|------------|---|-------------|
| 35.4.6.4 | FCP_TL_CTRL: FCP Tile/Linear Control Register | 35-32 |
| 35.4.6.5 | FCP_PICINFO1: FCP Picture Information 1 | 35-33 |
| 35.4.6.6 | FCP_BA_ANC_m {m: Y0, Y1, Y2, C}: FCP Base Address Ancillary information of plane m | 35-34 |
| 35.4.6.7 | FCP_BA_REF_m {m: Y0, Y1, Y2, C}: FCP Base Address Reference picture of plane m.. | 35-35 |
| 35.4.7 | Operation | 35-36 |
| 35.4.7.1 | Operation Control Setting for the IP set FCPF+FDP | 35-36 |
| 35.4.7.2 | Reset Operation | 35-37 |
| 35.4.7.3 | Register settings and restrictions on connected IP (FDP) | 35-38 |
| 35.4.7.4 | Decode data..... | 35-40 |
| 35.4.7.5 | FCNL image data | 35-45 |
| 35.4.7.6 | Operation status..... | 35-45 |
| 35.4.8 | Usage Notes | 35-46 |
| 35.4.8.1 | Manipulating FCPF+FDP with CPG module stop, software reset..... | 35-46 |
| 36. | Display Unit (DU)..... | 36-1 |
| 36.1 | Overview..... | 36-1 |
| 36.1.1 | Features..... | 36-1 |
| 36.1.2 | Block Diagram..... | 36-4 |
| 36.1.3 | External Pins..... | 36-7 |
| 36.1.4 | Register Configuration..... | 36-13 |
| 36.1.5 | Connected Module | 36-27 |
| 36.2 | Register Description..... | 36-28 |
| 36.2.1 | Display Unit System Control Registers | 36-29 |
| 36.2.2 | Display Timing Generation Registers | 36-80 |
| 36.2.3 | Display Attribute Registers..... | 36-92 |
| 36.2.4 | Display Plane Registers | 36-97 |
| 36.2.5 | Dual Display Output Control Registers | 36-106 |
| 36.2.6 | External Synchronization Control Registers | 36-117 |
| 36.2.7 | YC-RGB Conversion Coefficient Registers | 36-124 |
| 36.3 | Operation | 36-133 |
| 36.3.1 | Configuration of Output Screen..... | 36-133 |
| 36.3.2 | Display On/Off | 36-134 |
| 36.3.3 | Plane Parameter | 36-135 |
| 36.3.4 | Data Format for Input | 36-136 |
| 36.3.5 | Data Formats for Output | 36-138 |
| 36.3.6 | Superimposing of Planes..... | 36-139 |
| 36.3.7 | Sync Mode | 36-140 |
| 36.3.8 | Dual Display Output | 36-142 |
| 36.3.9 | YC Format Display | 36-146 |
| 36.3.10 | Display Timing Generation..... | 36-149 |
| 36.3.11 | CSYNC [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] | 36-151 |
| 36.3.12 | Scan Method | 36-152 |
| 36.3.13 | Color Detection..... | 36-156 |
| 36.3.14 | External Sync Control [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]..... | 36-157 |
| 36.3.15 | Output Signal Timing Adjustment..... | 36-158 |
| 36.4 | Usage Notes | 36-159 |
| 36.4.1 | Module Standby Mode..... | 36-159 |
| 36.4.2 | Transition to Module Standby Mode | 36-159 |
| 36.4.3 | Release from Module Standby Mode and Restarting Display | 36-159 |
| 36.4.4 | Acquisition of External Sync Signal [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]..... | 36-159 |
| 36.4.5 | Restrictions on Changing the Synchronization of the External SYNC Signal [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]..... | 36-160 |
| 36.4.6 | Note on Register Settings..... | 36-160 |

| | | |
|------------|--|-------------|
| 36.4.7 | Transition to Display operation stop by DRES bit in DSYSRn..... | 36-160 |
| 37. | LVDS Interface..... | 37-1 |
| 37.1 | Overview..... | 37-1 |
| 37.1.1 | Features..... | 37-1 |
| 37.1.2 | Block Diagram..... | 37-2 |
| 37.1.3 | External Pins..... | 37-4 |
| 37.1.4 | Register Configuration..... | 37-5 |
| 37.1.5 | Connected Module..... | 37-6 |
| 37.2 | Register Description..... | 37-7 |
| 37.2.1 | LVDS-IF Control Register 0 (LVDCR0 / LVD1CR0)..... | 37-7 |
| 37.2.2 | LVDS-IF Control Register 1 (LVDCR1 / LVD1CR1)..... | 37-9 |
| 37.2.3 | PLL Control Register (LVDPLLCR / LVD1PLLCR)..... | 37-10 |
| 37.2.4 | CTR Control Register (LVDCTRCR / LVD1CTRCR)..... | 37-13 |
| 37.2.5 | CH Control Register (LVDCHCR / LVD1CHCR)..... | 37-15 |
| 37.2.6 | Vertical Stripe Control Register (LVDSTRIPE / LVD1STRIPE)..... | 37-17 |
| 37.2.7 | SSCG Control Register (LVDSCR / LVD1SCR)..... | 37-18 |
| 37.2.8 | LVDS DIVIDER Register (LVDDIV/ LVD1DIV)..... | 37-20 |
| 37.3 | Operation..... | 37-21 |
| 37.3.1 | Mode Selection..... | 37-21 |
| 37.3.2 | Ctrl Signal Selection..... | 37-23 |
| 37.3.3 | CH Selection..... | 37-24 |
| 37.3.4 | Vertical stripe output [RZ/G2E]..... | 37-25 |
| 37.3.5 | Dot clock Settings..... | 37-26 |
| 37.3.6 | PLL Setting [RZ/G2E]..... | 37-28 |
| 37.3.7 | Setting Procedure..... | 37-30 |
| 37.4 | Usage Notes..... | 37-35 |
| 38. | HDMI..... | 38-1 |
| 38.1 | Overview..... | 38-1 |
| 38.1.1 | Features..... | 38-1 |
| 39. | AUDIO..... | 39-1 |
| 39.1 | Overview..... | 39-1 |
| 39.1.1 | Features..... | 39-1 |
| 39.1.2 | Block Diagram..... | 39-2 |
| 39.1.3 | External Pins..... | 39-4 |
| 39.1.4 | Register Configuration..... | 39-4 |
| 39.1.5 | Connected Module..... | 39-5 |
| 39.2 | Register Description..... | 39-5 |
| 39.3 | Operation..... | 39-6 |
| 39.3.1 | List of Routings..... | 39-6 |
| 39.3.2 | Audio-DMAC..... | 39-7 |
| 39.3.3 | Data Format on Audio Local Bus..... | 39-10 |
| 39.3.4 | Rearranging the Order of Data..... | 39-11 |
| 39.3.5 | Transfer Flow..... | 39-15 |
| 39.3.6 | Module Standby Function..... | 39-17 |
| 39.4 | Usage Notes..... | 39-21 |
| 39.4.1 | Notes on Route Switching and Setting Changes..... | 39-21 |
| 39.4.2 | Use of SSI3 and SSI4..... | 39-21 |
| 39.4.3 | Notes on CMD Block Usage..... | 39-21 |
| 39.4.4 | Note when SSI0, SSI1, and SSI2, or SSI0, SSI1, SSI2, and SSI9 are Used for Single-Source Audio Data..... | 39-21 |
| 39.4.5 | Note on Transfer..... | 39-21 |

| | | |
|------------|--|-------------|
| 39.4.6 | Note on Usage of the SCU Module..... | 39-22 |
| 39.4.7 | Note on Usage of the SSIU Module [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]..... | 39-22 |
| 39.4.8 | Note on software reset during data transfer | 39-22 |
| 40. | Serial Sound Interface Unit (SSIU) | 40-1 |
| 40.1 | Overview..... | 40-1 |
| 40.1.1 | Features..... | 40-1 |
| 40.1.2 | Block Diagram..... | 40-3 |
| 40.1.3 | External Pins..... | 40-8 |
| 40.1.4 | Register Configuration..... | 40-10 |
| 40.1.5 | Connected Module..... | 40-23 |
| 40.2 | Register Description..... | 40-24 |
| 40.2.1 | SSIn BUSIF Mode Register (SSIn_BUSIF_MODE)..... | 40-25 |
| 40.2.2 | SSIm BUSIF Audio Information Register (SSIm_BUSIF_ADINR)..... | 40-27 |
| 40.2.3 | SSIr BUSIF Audio Information Register (SSIr_BUSIF_ADINR)..... | 40-31 |
| 40.2.4 | SSIQ BUSIF Data Align Register (SSIQ_BUSIF_DALIGN)..... | 40-32 |
| 40.2.5 | SSIr BUSIF Data Align Register (SSIr_BUSIF_DALIGN)..... | 40-40 |
| 40.2.6 | SSIs BUSIF Data Align Register (SSIs_BUSIF_DALIGN)..... | 40-42 |
| 40.2.7 | SSIt BUSIF Data Align Register (SSIt_BUSIF_DALIGN)..... | 40-45 |
| 40.2.8 | SSIU BUSIF Data Align Register (SSIU_BUSIF_DALIGN)..... | 40-50 |
| 40.2.9 | SSIV BUSIF Data Align Register (SSIV_BUSIF_DALIGN)..... | 40-53 |
| 40.2.10 | SSIW BUSIF Data Align Register (SSIW_BUSIF_DALIGN)..... | 40-60 |
| 40.2.11 | SSIX BUSIF Data Align Register (SSIX_BUSIF_DALIGN)..... | 40-61 |
| 40.2.12 | SSIY BUSIF Data Align Register (SSIY_BUSIF_DALIGN)..... | 40-64 |
| 40.2.13 | SSIQ Mode Register (SSIQ_MODE)..... | 40-65 |
| 40.2.14 | SSIQ Control Register (SSIQ_CONTROL)..... | 40-67 |
| 40.2.15 | SSIR Control Register (SSIR_CONTROL)..... | 40-70 |
| 40.2.16 | SSIQ Status Register (SSIQ_STATUS)..... | 40-71 |
| 40.2.17 | SSIR Status Register (SSIR_STATUS)..... | 40-74 |
| 40.2.18 | SSIQ Interrupt Enable Register (SSIQ_INT_ENABLE_MAIN)..... | 40-77 |
| 40.2.19 | SSIR Interrupt Enable Register (SSIR_INT_ENABLE_MAIN)..... | 40-79 |
| 40.2.20 | SSI Mode Register 0 (SSI_MODE0)..... | 40-81 |
| 40.2.21 | SSI Mode Register 1 (SSI_MODE1)..... | 40-84 |
| 40.2.22 | SSI Mode Register 2 (SSI_MODE2)..... | 40-86 |
| 40.2.23 | SSI Mode Register 3 (SSI_MODE3)..... | 40-88 |
| 40.2.24 | SSI Control Register (SSI_CONTROL)..... | 40-89 |
| 40.2.25 | SSI Control Register 2 (SSI_CONTROL2)..... | 40-90 |
| 40.2.26 | SSI SYSTEM Status Register 0 (SSI_SYSTEM_STATUS0)..... | 40-91 |
| 40.2.27 | SSI SYSTEM Status Register 1 (SSI_SYSTEM_STATUS1)..... | 40-94 |
| 40.2.28 | SSI SYSTEM Status Register 2 (SSI_SYSTEM_STATUS2)..... | 40-95 |
| 40.2.29 | SSI SYSTEM Status Register 3 (SSI_SYSTEM_STATUS3)..... | 40-98 |
| 40.2.30 | SSI SYSTEM Interrupt Enable Register 0 (SSI_SYSTEM_INT_ENABLE0)..... | 40-99 |
| 40.2.31 | SSI SYSTEM Interrupt Enable Register 1 (SSI_SYSTEM_INT_ENABLE1)..... | 40-101 |
| 40.2.32 | SSI SYSTEM Interrupt Enable Register 2 (SSI_SYSTEM_INT_ENABLE2)..... | 40-102 |
| 40.2.33 | SSI SYSTEM Interrupt Enable Register 3 (SSI_SYSTEM_INT_ENABLE3)..... | 40-104 |
| 40.2.34 | SSI SYSTEM Status Register 4 (SSI_SYSTEM_STATUS4)..... | 40-105 |
| 40.2.35 | SSI SYSTEM Status Register 5 (SSI_SYSTEM_STATUS5)..... | 40-108 |
| 40.2.36 | SSI SYSTEM Status Register 6 (SSI_SYSTEM_STATUS6)..... | 40-109 |
| 40.2.37 | SSI SYSTEM Status Register 7 (SSI_SYSTEM_STATUS7)..... | 40-112 |
| 40.2.38 | SSI SYSTEM Interrupt Enable Register 4 (SSI_SYSTEM_INT_ENABLE4)..... | 40-113 |
| 40.2.39 | SSI SYSTEM Interrupt Enable Register 5 (SSI_SYSTEM_INT_ENABLE5)..... | 40-115 |
| 40.2.40 | SSI SYSTEM Interrupt Enable Register 6 (SSI_SYSTEM_INT_ENABLE6)..... | 40-116 |
| 40.2.41 | SSI SYSTEM Interrupt Enable Register 7 (SSI_SYSTEM_INT_ENABLE7)..... | 40-118 |
| 40.2.42 | HDMI0 Select Register (HDMI0_SEL)..... | 40-119 |

| | | |
|---------|--|--------|
| 40.2.43 | HDMI1 Select Register (HDMI1_SEL)..... | 40-122 |
| 40.2.44 | SSIm BUSIF Data Align2 Register (SSIm_BUSIF_DALIGN2)..... | 40-125 |
| 40.2.45 | SSIQ Mode2 Register (SSIQ_MODE2)..... | 40-131 |
| 40.2.46 | SSIQ Status2 Register (SSIQ_STATUS2)..... | 40-132 |
| 40.2.47 | SSIQ Interrupt Enable2 Register (SSIQ_INT_ENABLE_MAIN2)..... | 40-134 |
| 40.2.48 | SSIn_BUSIF Data Register (SSIn_BUSIF)..... | 40-135 |
| 40.3 | Operation | 40-136 |
| 40.3.1 | Module Specifications | 40-136 |
| 40.3.2 | Basic Configuration | 40-138 |
| 40.3.3 | Configuration with Multiple SSI Modules..... | 40-140 |
| 40.3.4 | TDM Format Extension Function (TDM Extend Mode) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]..... | 40-142 |
| 40.3.5 | TDM Format Extension Function (TDM Extend Mode) [RZ/G2E] | 40-144 |
| 40.3.6 | TDM Format Split Function (TDM Split Mode) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]..... | 40-146 |
| 40.3.7 | TDM Format Split Function (TDM Split Mode) [RZ/G2E]..... | 40-148 |
| 40.3.8 | TDM Format 16ch Function (TDM 16ch Mode) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]..... | 40-150 |
| 40.3.9 | TDM Format Extend Split Function (TDM Ex-Split Mode) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]..... | 40-151 |
| 40.3.10 | TDM Format Extend Split Function (TDM Ex-Split Mode) [RZ/G2E] | 40-154 |
| 40.3.11 | SSI Pin Connections | 40-156 |
| 40.3.12 | HDMI Signal Selection [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]..... | 40-167 |
| 40.3.13 | Procedure of SSIU Transfer..... | 40-168 |
| 40.3.14 | Interrupts of SSI Setting..... | 40-172 |
| 40.4 | Usage Notes | 40-178 |
| 40.4.1 | Note on Transfer | 40-178 |
| 41. | Serial Sound Interface (SSI) | 41-1 |
| 41.1 | Overview..... | 41-1 |
| 41.1.1 | Features..... | 41-1 |
| 41.1.2 | Block Diagram..... | 41-2 |
| 41.1.3 | External Pins | 41-3 |
| 41.1.4 | Register Configuration..... | 41-3 |
| 41.1.5 | Connected Module | 41-6 |
| 41.2 | Register Description..... | 41-7 |
| 41.2.1 | Control Register (SSICRn) | 41-7 |
| 41.2.2 | Status Register (SSISRn) | 41-13 |
| 41.2.3 | Transmit Data Register (SSITDRn)..... | 41-17 |
| 41.2.4 | Receive Data Register (SSIRDRn) | 41-18 |
| 41.2.5 | WS Mode Register (SSIWSRn)..... | 41-19 |
| 41.2.6 | FS Mode Register (SSIFMRn)..... | 41-21 |
| 41.2.7 | FS Status Register (SSIFSRn)..... | 41-22 |
| 41.2.8 | Control Register Extend (SSICREm) (m = 0 to 4, 9)..... | 41-23 |
| 41.3 | Operation | 41-24 |
| 41.3.1 | Bus Format..... | 41-24 |
| 41.3.2 | Operating Modes..... | 41-41 |
| 41.3.3 | Transmit Operation | 41-42 |
| 41.3.4 | Receive Operation..... | 41-45 |
| 41.3.5 | Serial Bit Clock Control..... | 41-48 |
| 41.4 | Usage Notes | 41-49 |
| 41.4.1 | Limitations from Overflow during Receive DMA Operation..... | 41-49 |
| 41.4.2 | Limitations on Combinations of Modes Related to Common Pins to SSI7 and SSI8..... | 41-49 |
| 41.4.3 | Limitations on Slave Mode Operation..... | 41-50 |

| | | |
|------------|---|-------------|
| 41.4.4 | Limitations on Changes to Settings..... | 41-50 |
| 41.4.5 | Stopping or Resuming Transmission | 41-50 |
| 41.4.6 | Stable period of SCK and WS..... | 41-51 |
| 41.4.7 | About CLK_FS input clock | 41-51 |
| 42. | Audio Clock Generator (ADG)..... | 42-1 |
| 42.1 | Overview..... | 42-1 |
| 42.1.1 | Features..... | 42-1 |
| 42.1.2 | Block Diagram | 42-2 |
| 42.1.3 | External Pins | 42-4 |
| 42.1.4 | Register Configuration..... | 42-5 |
| 42.1.5 | Connected Module | 42-7 |
| 42.2 | Register Description..... | 42-8 |
| 42.2.1 | BRGA Baud Rate Setting Register (BRRA)..... | 42-9 |
| 42.2.2 | BRGB Baud Rate Setting Register (BRRB) | 42-10 |
| 42.2.3 | BRG Clock Select Register (BRGCKR)..... | 42-11 |
| 42.2.4 | Audio Clock Select Register 0 (AUDIO_CLK_SEL0)..... | 42-13 |
| 42.2.5 | Audio Clock Select Register 1 (AUDIO_CLK_SEL1)..... | 42-17 |
| 42.2.6 | Audio Clock Select Register 2 (AUDIO_CLK_SEL2)..... | 42-21 |
| 42.2.7 | Timing Signal Enable Register (TIM_EN)..... | 42-23 |
| 42.2.8 | SRC Input Timing Select Register 0 (SRCIN_TIMSEL0) | 42-24 |
| 42.2.9 | SRC Input Timing Select Register 1 (SRCIN_TIMSEL1) | 42-28 |
| 42.2.10 | SRC Input Timing Select Register 2 (SRCIN_TIMSEL2) | 42-32 |
| 42.2.11 | SRC Input Timing Select Register 3 (SRCIN_TIMSEL3) | 42-36 |
| 42.2.12 | SRC Input Timing Select Register 4 (SRCIN_TIMSEL4) | 42-40 |
| 42.2.13 | SRC Output Timing Select Register 0 (SRCOUT_TIMSEL0)..... | 42-44 |
| 42.2.14 | SRC Output Timing Select Register 1 (SRCOUT_TIMSEL1)..... | 42-48 |
| 42.2.15 | SRC Output Timing Select Register 2 (SRCOUT_TIMSEL2)..... | 42-52 |
| 42.2.16 | SRC Output Timing Select Register 3 (SRCOUT_TIMSEL3)..... | 42-56 |
| 42.2.17 | SRC Output Timing Select Register 4 (SRCOUT_TIMSEL4)..... | 42-60 |
| 42.2.18 | CMD Output Timing Select Register (CMDOUT_TIMSEL)..... | 42-64 |
| 42.2.19 | AVB Clock Select Register (AVBCKR) | 42-69 |
| 42.2.20 | AVB sync select Register 0 (AVB_SYNC_SEL0)..... | 42-71 |
| 42.2.21 | AVB sync select Register 1 (AVB_SYNC_SEL1)..... | 42-72 |
| 42.2.22 | AVB sync select Register 2 (AVB_SYNC_SEL2)..... | 42-73 |
| 42.2.23 | AVB sync divide Register 0 (AVB_SYNC_DIV0)..... | 42-74 |
| 42.2.24 | AVB sync divide Register 1 (AVB_SYNC_DIV1)..... | 42-75 |
| 42.2.25 | AVB clock divider comparison value register 0 (AVB_CLK_DIV0)..... | 42-76 |
| 42.2.26 | AVB clock divider comparison value register 1 (AVB_CLK_DIV1) | 42-77 |
| 42.2.27 | AVB clock divider comparison value register 2 (AVB_CLK_DIV2) | 42-78 |
| 42.2.28 | AVB clock divider comparison value register 3 (AVB_CLK_DIV3) | 42-79 |
| 42.2.29 | AVB clock divider comparison value register 4 (AVB_CLK_DIV4) | 42-80 |
| 42.2.30 | AVB clock divider comparison value register 5 (AVB_CLK_DIV5) | 42-81 |
| 42.2.31 | AVB clock divider comparison value register 6 (AVB_CLK_DIV6)..... | 42-82 |
| 42.2.32 | AVB clock divider comparison value register 7 (AVB_CLK_DIV7) | 42-83 |
| 42.2.33 | AVB clock divider configuration register (AVB_CLK_CONFIG) | 42-84 |
| 42.3 | Operation | 42-85 |
| 42.4 | Usage Notes | 42-89 |
| 42.4.1 | Note on using AUDIO_CLK and Input/Output timing signal in stable condition..... | 42-89 |
| 42.4.2 | Note on using ZA2φ..... | 42-89 |
| 43. | Sampling Rate Converter Unit (SCU) | 43-1 |
| 43.1 | Overview..... | 43-1 |
| 43.1.1 | Features..... | 43-1 |

| | | |
|---------|--|-------|
| 43.1.2 | Block Diagram..... | 43-3 |
| 43.1.3 | External Pins..... | 43-3 |
| 43.1.4 | Register Configuration..... | 43-4 |
| 43.1.5 | Connected Module..... | 43-13 |
| 43.2 | Register Description..... | 43-14 |
| 43.2.1 | SRCm(in/out)_BUSIF_MODE Register (SRCm(in/out)_BUSIF_MODE)..... | 43-15 |
| 43.2.2 | SRCm_BUSIF_DALIGN Register (SRCm_BUSIF_DALIGN)..... | 43-17 |
| 43.2.3 | SRCn_BUSIF_DALIGN Register (SRCn_BUSIF_DALIGN)..... | 43-21 |
| 43.2.4 | SRCm_MODE Register (SRCm_MODE)..... | 43-22 |
| 43.2.5 | SRCm Control Register (SRCm_CONTROL)..... | 43-23 |
| 43.2.6 | SRCm Status Register (SRCm_STATUS)..... | 43-24 |
| 43.2.7 | SRCn Status Register (SRCn_STATUS)..... | 43-26 |
| 43.2.8 | SRCm Interrupt Enable Register 0 (SRCm_INT_ENABLE0)..... | 43-27 |
| 43.2.9 | SRCn Interrupt Enable Register 0 (SRCn_INT_ENABLE0)..... | 43-29 |
| 43.2.10 | CMDn out_BUSIF_MODE Register (CMDn out_BUSIF_MODE)..... | 43-30 |
| 43.2.11 | CMDn_BUSIF_DALIGN Register (CMDn_BUSIF_DALIGN)..... | 43-32 |
| 43.2.12 | CMDn_ROUTE_SELECT Register (CMDn_ROUTE_SELECT)..... | 43-35 |
| 43.2.13 | CMDn Control Register (CMDn_CONTROL)..... | 43-37 |
| 43.2.14 | SCU_SYSTEM Status Register 0 (SCU_SYSTEM_STATUS0)..... | 43-38 |
| 43.2.15 | SCU_SYSTEM Interrupt Enable Register 0 (SCU_SYSTEM_INT_ENABLE0)..... | 43-41 |
| 43.2.16 | SCU_SYSTEM Status Register 1 (SCU_SYSTEM_STATUS1)..... | 43-43 |
| 43.2.17 | SCU_SYSTEM Interrupt Enable Register 1 (SCU_SYSTEM_INT_ENABLE1)..... | 43-46 |
| 43.2.18 | SRCm Software Reset Register (SRCm_SWRSR)..... | 43-48 |
| 43.2.19 | SRCm SRC Initialization Register (SRCm_SRCIR)..... | 43-49 |
| 43.2.20 | SRCm Audio Information Register (SRCm_ADINR)..... | 43-50 |
| 43.2.21 | SRCm IFS Control Register (SRCm_IFSCR)..... | 43-52 |
| 43.2.22 | SRCm IFS Value Setting Register (SRCm_IFSVR)..... | 43-53 |
| 43.2.23 | SRCm SRC Control Register (SRCm_SRCCR)..... | 43-54 |
| 43.2.24 | SRCm Buffer Size DATA RAM Setting Register (SRCm_BSDSR)..... | 43-55 |
| 43.2.25 | SRCm Buffer Size DATA RAM Setting Register (SRCm_BSDSR)..... | 43-56 |
| 43.2.26 | SRCm Buffer Size IJEC RAM Setting Register (SRCm_BSISR)..... | 43-57 |
| 43.2.27 | CTUn Software Reset Register (CTUn_SWRSR)..... | 43-58 |
| 43.2.28 | CTUn CTU Initialization Register (CTUn_CTUIR)..... | 43-59 |
| 43.2.29 | CTUn Audio Information Register (CTUn_ADINR)..... | 43-60 |
| 43.2.30 | CTUn CTU Pass Mode Register (CTUn_CPMDR)..... | 43-61 |
| 43.2.31 | CTUn Scale Mode Register (CTUn_SCMDR)..... | 43-66 |
| 43.2.32 | CTUn Scale Value e00 Register (CTUn_SV00R)..... | 43-67 |
| 43.2.33 | CTUn Scale Value e01 Register (CTUn_SV01R)..... | 43-68 |
| 43.2.34 | CTUn Scale Value e02 Register (CTUn_SV02R)..... | 43-69 |
| 43.2.35 | CTUn Scale Value e03 Register (CTUn_SV03R)..... | 43-70 |
| 43.2.36 | CTUn Scale Value e04 Register (CTUn_SV04R)..... | 43-71 |
| 43.2.37 | CTUn Scale Value e05 Register (CTUn_SV05R)..... | 43-72 |
| 43.2.38 | CTUn Scale Value e06 Register (CTUn_SV06R)..... | 43-73 |
| 43.2.39 | CTUn Scale Value e07 Register (CTUn_SV07R)..... | 43-74 |
| 43.2.40 | CTUn Scale Value e10 Register (CTUn_SV10R)..... | 43-75 |
| 43.2.41 | CTUn Scale Value e11 Register (CTUn_SV11R)..... | 43-76 |
| 43.2.42 | CTUn Scale Value e12 Register (CTUn_SV12R)..... | 43-77 |
| 43.2.43 | CTUn Scale Value e13 Register (CTUn_SV13R)..... | 43-78 |
| 43.2.44 | CTUn Scale Value e14 Register (CTUn_SV14R)..... | 43-79 |
| 43.2.45 | CTUn Scale Value e15 Register (CTUn_SV15R)..... | 43-80 |
| 43.2.46 | CTUn Scale Value e16 Register (CTUn_SV16R)..... | 43-81 |
| 43.2.47 | CTUn Scale Value e17 Register (CTUn_SV17R)..... | 43-82 |
| 43.2.48 | CTUn Scale Value e20 Register (CTUn_SV20R)..... | 43-83 |
| 43.2.49 | CTUn Scale Value e21 Register (CTUn_SV21R)..... | 43-84 |

| | | |
|---------|---|--------|
| 43.2.50 | CTUn Scale Value e22 Register (CTUn_SV22R) | 43-85 |
| 43.2.51 | CTUn Scale Value e23 Register (CTUn_SV23R) | 43-86 |
| 43.2.52 | CTUn Scale Value e24 Register (CTUn_SV24R) | 43-87 |
| 43.2.53 | CTUn Scale Value e25 Register (CTUn_SV25R) | 43-88 |
| 43.2.54 | CTUn Scale Value e26 Register (CTUn_SV26R) | 43-89 |
| 43.2.55 | CTUn Scale Value e27 Register (CTUn_SV27R) | 43-90 |
| 43.2.56 | CTUn Scale Value e30 Register (CTUn_SV30R) | 43-91 |
| 43.2.57 | CTUn Scale Value e31 Register (CTUn_SV31R) | 43-92 |
| 43.2.58 | CTUn Scale Value e32 Register (CTUn_SV32R) | 43-93 |
| 43.2.59 | CTUn Scale Value e33 Register (CTUn_SV33R) | 43-94 |
| 43.2.60 | CTUn Scale Value e34 Register (CTUn_SV34R) | 43-95 |
| 43.2.61 | CTUn Scale Value e35 Register (CTUn_SV35R) | 43-96 |
| 43.2.62 | CTUn Scale Value e36 Register (CTUn_SV36R) | 43-97 |
| 43.2.63 | CTUn Scale Value e37 Register (CTUn_SV37R) | 43-98 |
| 43.2.64 | MIXp Software Reset Register (MIXp_SWRSR) | 43-99 |
| 43.2.65 | MIXp MIX Initialization Register (MIXp_MIXIR) | 43-100 |
| 43.2.66 | MIXp Audio Information Register (MIXp_ADINR)..... | 43-101 |
| 43.2.67 | MIXp MIX Mode Register (MIXp_MIXMR) | 43-102 |
| 43.2.68 | MIXp MIX Volume Period Register (MIXp_MVPDR) | 43-103 |
| 43.2.69 | MIXp MIX Decibel A Register (MIXp_MDBAR)..... | 43-105 |
| 43.2.70 | MIXp MIX Decibel B Register (MIXp_MDBBR) | 43-106 |
| 43.2.71 | MIXp MIX Decibel C Register (MIXp_MDBCR) | 43-107 |
| 43.2.72 | MIXp MIX Decibel D Register (MIXp_MDBDR)..... | 43-108 |
| 43.2.73 | MIXp MIX Decibel Enable Register (MIXp_MDBER)..... | 43-109 |
| 43.2.74 | MIXp MIX Status Register (MIXp_MIXSR) | 43-110 |
| 43.2.75 | DVCp Software Reset Register (DVCp_SWRSR) | 43-111 |
| 43.2.76 | DVCp DVU Initialization Register (DVCp_DVUIR) | 43-112 |
| 43.2.77 | DVCp Audio Information Register (DVCp_ADINR) | 43-113 |
| 43.2.78 | DVCp DVU Control Register (DVCp_DVUCR) | 43-115 |
| 43.2.79 | DVCp Zero Cross Mute Control Register (DVCp_ZCMCR)..... | 43-116 |
| 43.2.80 | DVCp Volume Ramp Control Register (DVCp_VRCTR) | 43-118 |
| 43.2.81 | DVCp Volume Ramp Period Register (DVCp_VRPDR)..... | 43-120 |
| 43.2.82 | DVCp Volume Ramp Decibel Register (DVCp_VRDBR) | 43-122 |
| 43.2.83 | DVCp Volume Ramp Wait Time Register (DVCp_VRWTR) | 43-123 |
| 43.2.84 | DVCp Volume Value Setting 0 Register (DVCp_VOL0R) | 43-124 |
| 43.2.85 | DVCp Volume Value Setting 1 Register (DVCp_VOL1R) | 43-125 |
| 43.2.86 | DVCp Volume Value Setting 2 Register (DVCp_VOL2R) | 43-126 |
| 43.2.87 | DVCp Volume Value Setting 3 Register (DVCp_VOL3R) | 43-127 |
| 43.2.88 | DVCp Volume Value Setting 4 Register (DVCp_VOL4R) | 43-128 |
| 43.2.89 | DVCp Volume Value Setting 5 Register (DVCp_VOL5R) | 43-129 |
| 43.2.90 | DVCp Volume Value Setting 6 Register (DVCp_VOL6R) | 43-130 |
| 43.2.91 | DVCp Volume Value Setting 7 Register (DVCp_VOL7R) | 43-131 |
| 43.2.92 | DVCp DVU Enable Register (DVCp_DVUER) | 43-132 |
| 43.2.93 | DVCp DVU Status Register (DVCp_DVUSR)..... | 43-133 |
| 43.2.94 | DVCp Interrupt Enable Register (DVCp_DVIER)..... | 43-135 |
| 43.2.95 | SRCm in Write Data Register (SRCm_in_BUSIF) | 43-137 |
| 43.2.96 | SRCm out Read Data Register (SRCm_out_BUSIF) | 43-137 |
| 43.2.97 | CMDn out Read Data Register (CMDn_out_BUSIF) | 43-138 |
| 43.3 | Operation | 43-139 |
| 43.3.1 | Procedure for Initializing the SCU..... | 43-139 |
| 43.3.2 | Procedure for Starting and Stopping Transfer by the SCU | 43-142 |
| 43.3.3 | Data Format for Data Transfer BUSIF..... | 43-143 |
| 43.3.4 | Changing Data Order in Channel Units | 43-144 |
| 43.3.5 | SRC Block | 43-147 |

| | | |
|---------|--|--------|
| 43.3.6 | CMD Block | 43-157 |
| 43.3.7 | Functional Blocks in CMD | 43-158 |
| 43.3.8 | Input Data Timing and Output Data Timing | 43-179 |
| 43.4 | Usage Notes | 43-180 |
| 43.4.1 | Notes on Fixed Value related to SCU registers | 43-180 |
| 43.4.2 | Note on generation and selection of timing signal | 43-180 |
| 44. | Direct Memory Access Controller for Audio (Audio-DMAC) | 44-1 |
| 44.1 | Overview | 44-1 |
| 44.1.1 | Features | 44-1 |
| 44.1.2 | External Pins | 44-2 |
| 44.1.3 | Register Configuration | 44-2 |
| 44.1.4 | Connected Modules | 44-19 |
| 44.2 | Register Description | 44-20 |
| 44.2.1 | DMA Interrupt Status Register for Lower-Numbered Channels (DMAISTA_L) | 44-20 |
| 44.2.2 | DMA Interrupt Status Register for Higher-Numbered Channels (DMAISTA_U) | 44-22 |
| 44.2.3 | DMA Secure Control Register for Lower-Numbered Channels (DMASEC_L) | 44-24 |
| 44.2.4 | DMA Secure Control Register Higher-Numbered Channels (DMASEC_H) | 44-26 |
| 44.2.5 | DMA Operation Register for Lower-Numbered Channels (DMAOR_L) | 44-28 |
| 44.2.6 | DMA Operation Register for Higher-Numbered Channels (DMAOR_U) | 44-30 |
| 44.2.7 | DMA Channel Clear Register for Lower-Numbered Channels (DMACHCLR_L) | 44-32 |
| 44.2.8 | DMA Channel Clear Register for Higher-Numbered Channels (DMACHCLR_U) | 44-33 |
| 44.2.9 | DPRAM Secure Control Register for Lower-Numbered Channels (DMADPSEC_L) | 44-34 |
| 44.2.10 | DPRAM Secure Control Register for Higher-Numbered Channels (DMADPSEC_U) | 44-35 |
| 44.2.11 | DMA Source Address Registers 0 to 31 (DMASAR_0 to DMASAR_31) | 44-36 |
| 44.2.12 | DMA Destination Address Registers 0 to 31 (DMADAR_0 to DMADAR_31) | 44-37 |
| 44.2.13 | DMA Transfer Count Registers 0 to 31 (DMATCR_0 to DMATCR_31) | 44-38 |
| 44.2.14 | DMA Transfer Count Registers B_0 to 31 (DMATCRB_0 to DMATCRB_31) | 44-39 |
| 44.2.15 | DMA Transfer Size Registers 0 to 31 (DMATSR_0 to DMATSR_31) | 44-39 |
| 44.2.16 | DMA Transfer Size Registers B_0 to 31 (DMATSRB_0 to DMATSRB_31) | 44-39 |
| 44.2.17 | DMA Channel Control Registers 0 to 31 (DMACHCR_0 to DMACHCR_31) | 44-41 |
| 44.2.18 | DMA Channel Control Register B_0 to 31 (DMACHCRB_0 to DMACHCRB_31) | 44-45 |
| 44.2.19 | DMA Buffer Control Registers 0 to 31 (DMABUFCR_0 to DMABUFCR_31) | 44-47 |
| 44.2.20 | DMA Extended Resource Selectors 0 to 31 (DMARS_0 to DMARS_31) | 44-48 |
| 44.2.21 | DMA Descriptor Base Address Registers 0 to 31 (DMADPBASE_0 to DMADPBASE_31) | 44-49 |
| 44.2.22 | DMA Descriptor Control Registers 0 to 31 (DMADPCR_0 to DMADPCR_31) | 44-50 |
| 44.2.23 | DMA Fixed Source Address Registers 0 to 31 (DMAFIXSAR_0 to DMAFIXSAR_31) | 44-51 |
| 44.2.24 | DMA Fixed Destination Address Registers 0 to 31 (DMAFIXDAR_0 to DMAFIXDAR_31) | 44-51 |
| 44.2.25 | DMA Fixed Descriptor Base Address Registers 0 to 31 (DMAFIXDPBASE_0 to DMAFIXDPBASE_31) | 44-52 |
| 44.2.26 | Descriptor Memory (DescriptorMEM) | 44-52 |
| 44.2.27 | Secure function Secure Status register for channel 0 to 15 (DMASES_L) | 44-53 |
| 44.2.28 | Secure function Secure Status register for channel 16 to 31 (DMASES_U) | 44-54 |
| 44.2.29 | Secure function Salve Error Address register for channel 0 to 15 (DMASEDDR_L) | 44-55 |
| 44.2.30 | Secure function Salve Error Address register for channel 16 to 31 (DMASEDDR_U) | 44-55 |
| 44.2.31 | Secure function Error Master ID register for channel 0 to 15 (DMASEMID_L) | 44-56 |
| 44.2.32 | Secure function Error Master ID register for channel 16 to 31 (DMASEMID_U) | 44-56 |
| 44.3 | Operation | 44-57 |
| 44.3.1 | DMA Transfer Requests | 44-57 |
| 44.3.2 | Channel Priority | 44-65 |
| 44.3.3 | Slow Speed Mode | 44-65 |
| 44.3.4 | Descriptor Memory | 44-65 |
| 44.3.5 | Transmission Flow | 44-74 |
| 44.3.6 | Total Size Transmission | 44-75 |

| | | |
|---------|---|-------|
| 44.4 | Usage Notes | 44-77 |
| 45. | Audio-DMAC-Peripheral-Peripheral (Audio-DMACpp)..... | 45-1 |
| 45.1 | Overview..... | 45-1 |
| 45.1.1 | Features..... | 45-1 |
| 45.1.2 | Block Diagram..... | 45-2 |
| 45.1.3 | External Pins..... | 45-3 |
| 45.1.4 | Register Configuration..... | 45-4 |
| 45.1.5 | Connected Module..... | 45-5 |
| 45.2 | Register Description..... | 45-6 |
| 45.2.1 | PDMA Source Address n Register (PDMASARn)..... | 45-6 |
| 45.2.2 | PDMA Destination Address n Register (PDMADARn)..... | 45-7 |
| 45.2.3 | PDMA Channel Control n Register (PDMACHCRn)..... | 45-8 |
| 45.2.4 | PDMA Source Address n Register (extended) (PDMASAREn)..... | 45-9 |
| 45.2.5 | PDMA Destination Address n Register (extended) (PDMADAREn)..... | 45-10 |
| 45.2.6 | PDMA Channel Control n Register (extended) (PDMACHCREn)..... | 45-11 |
| 45.3 | Operation..... | 45-12 |
| 45.3.1 | Register Settings..... | 45-12 |
| 45.3.2 | Setting Flow..... | 45-19 |
| 45.4 | Usage Notes..... | 45-20 |
| 46. | EthernetAVB-IF..... | 46-1 |
| 46.1 | Overview..... | 46-1 |
| 46.1.1 | Features..... | 46-1 |
| 46.1.2 | Block Diagram..... | 46-2 |
| 46.1.3 | External Pins..... | 46-4 |
| 46.1.4 | Register Configuration..... | 46-5 |
| 46.1.5 | Connected Module..... | 46-10 |
| 46.2 | Register Description..... | 46-11 |
| 46.2.1 | AVB-DMAC Mode Register (CCC)..... | 46-11 |
| 46.2.2 | Descriptor Base Address Table Register (DBAT)..... | 46-13 |
| 46.2.3 | Descriptor Base Address Load Request Register (DLR)..... | 46-14 |
| 46.2.4 | AVB-DMAC Status Register (CSR)..... | 46-17 |
| 46.2.5 | Current Descriptor Address Register q (CDARq) (q = 0 to 21)..... | 46-20 |
| 46.2.6 | Error Status Register (ESR)..... | 46-21 |
| 46.2.7 | AVB-DMAC Product Specific Register (APSR)..... | 46-23 |
| 46.2.8 | Receive Configuration Register (RCR)..... | 46-24 |
| 46.2.9 | Receive Queue Configuration Register i (RQCi) (i = 0 to 4)..... | 46-27 |
| 46.2.10 | Receive Padding Configuration Register (RPC)..... | 46-32 |
| 46.2.11 | Reception Truncation Configuration register (RTC)..... | 46-33 |
| 46.2.12 | Unread frame counter warning level register (UFCW)..... | 46-34 |
| 46.2.13 | Unread frame counter stop level register (UFCS)..... | 46-35 |
| 46.2.14 | Unread Frame Counter Register i (UFCVi) (i = 0 to 4)..... | 46-36 |
| 46.2.15 | Unread Frame Counter Decrement Register i (UFCDi) (i = 0 to 4)..... | 46-38 |
| 46.2.16 | Separation Filter Offset Register (SFO)..... | 46-39 |
| 46.2.17 | Separation Filter Pattern Register i (SFPi) (i = 0 to 31)..... | 46-40 |
| 46.2.18 | Separation Filter Value register i (SFVi) (i = 0 to 1)..... | 46-41 |
| 46.2.19 | Separation Filter Mask Register i (SFMi) (i = 0 or 1)..... | 46-42 |
| 46.2.20 | Separation Filter Load register (SFL)..... | 46-43 |
| 46.2.21 | Current Incremental Address Register r (CIARr) (r = 0 to 17)..... | 46-44 |
| 46.2.22 | Last Incremental Address Register r (LIARr) (r = 0 to 17)..... | 46-45 |
| 46.2.23 | Transmit Configuration Register (TGC)..... | 46-46 |
| 46.2.24 | Transmit Configuration Control Register (TCCR)..... | 46-48 |
| 46.2.25 | Transmit Status Register (TSR)..... | 46-51 |

| | | |
|---------|--|--------|
| 46.2.26 | E-MAC status FIFO Access register (MFA)..... | 46-54 |
| 46.2.27 | Time Stamp FIFO Access Register 0 (TFA0)..... | 46-56 |
| 46.2.28 | Time Stamp FIFO Access Register 1 (TFA1)..... | 46-57 |
| 46.2.29 | Time Stamp FIFO Access Register 2 (TFA2)..... | 46-58 |
| 46.2.30 | Version and Release Register (VRR)..... | 46-59 |
| 46.2.31 | CBS Increment Value Register c (CIVRc) (c = 0 or 1)..... | 46-60 |
| 46.2.32 | CBS Decrement Value Register c (CDVRc) (c = 0 or 1)..... | 46-61 |
| 46.2.33 | CBS Upper Limit Register c (CULc) (c = 0 or 1)..... | 46-62 |
| 46.2.34 | CBS Lower Limit Register c (CLLc) (c = 0 or 1)..... | 46-63 |
| 46.2.35 | Descriptor Interrupt Control Register (DIC)..... | 46-64 |
| 46.2.36 | Descriptor Interrupt Status Register (DIS)..... | 46-66 |
| 46.2.37 | Error Interrupt Control Register (EIC)..... | 46-68 |
| 46.2.38 | Error Interrupt Status Register (EIS)..... | 46-71 |
| 46.2.39 | Receive Interrupt Control Register 0 (RIC0)..... | 46-76 |
| 46.2.40 | Receive Interrupt Status Register 0 (RIS0)..... | 46-78 |
| 46.2.41 | Receive Interrupt Control Register 1 (RIC1)..... | 46-80 |
| 46.2.42 | Receive Interrupt Status Register 1 (RIS1)..... | 46-84 |
| 46.2.43 | Receive Interrupt Control Register 2 (RIC2)..... | 46-91 |
| 46.2.44 | Receive Interrupt Status Register 2 (RIS2)..... | 46-95 |
| 46.2.45 | Transmit Interrupt Control Register (TIC)..... | 46-98 |
| 46.2.46 | Transmit Interrupt Status Register (TIS)..... | 46-101 |
| 46.2.47 | Interrupt Summary Status Register (ISS)..... | 46-105 |
| 46.2.48 | Common Interrupt Enable register (CIE)..... | 46-109 |
| 46.2.49 | Reception Interrupt Control register 3 (RIC3)..... | 46-111 |
| 46.2.50 | Reception Interrupt Status register 3 (RIS3)..... | 46-115 |
| 46.2.51 | gPTP Configuration Control Register (GCCR)..... | 46-120 |
| 46.2.52 | gPTP Maximum Transit Time Configuration Register (GTTT)..... | 46-123 |
| 46.2.53 | gPTP Presentation Time Comparison Register (GPTC)..... | 46-124 |
| 46.2.54 | gPTP Timer Increment Configuration Register (GTI)..... | 46-125 |
| 46.2.55 | gPTP timer offset register i (GTOi) (i = 0 to 2)..... | 46-126 |
| 46.2.56 | gPTP Interrupt Control Register (GIC)..... | 46-127 |
| 46.2.57 | gPTP interrupt status register (GIS)..... | 46-132 |
| 46.2.58 | gPTP Captured Presentation Time register (GCPT)..... | 46-139 |
| 46.2.59 | gPTP Timer Capture Register i (GCTi) (i = 0 to 2)..... | 46-140 |
| 46.2.60 | gPTP Status Register (GSR)..... | 46-141 |
| 46.2.61 | gPTP Interrupt Enable register (GIE)..... | 46-146 |
| 46.2.62 | gPTP Interrupt Disable register (GID)..... | 46-150 |
| 46.2.63 | gPTP Interrupt Line selection register (GIL)..... | 46-154 |
| 46.2.64 | gPTP Avtp Capture Prescaler register (GACP)..... | 46-158 |
| 46.2.65 | gPTP Presentation Time FIFO register i (GPTFi) (i = 0 to 3)..... | 46-159 |
| 46.2.66 | gPTP Captured Avtp Time register i (GCATi) (i = 0 to 15)..... | 46-160 |
| 46.2.67 | Descriptor Interrupt Line selection register (DIL)..... | 46-161 |
| 46.2.68 | Error Interrupt Line selection register (EIL)..... | 46-163 |
| 46.2.69 | Transmission Interrupt Line selection register (TIL)..... | 46-165 |
| 46.2.70 | Descriptor Interrupt Enable register (DIE)..... | 46-166 |
| 46.2.71 | Descriptor Interrupt Disable register (DID)..... | 46-169 |
| 46.2.72 | Error Interrupt Enable register (EIE)..... | 46-172 |
| 46.2.73 | Error Interrupt Disable register (EID)..... | 46-174 |
| 46.2.74 | Reception Interrupt Enable register 0 (RIE0)..... | 46-176 |
| 46.2.75 | Reception Interrupt Disable register 0 (RID0)..... | 46-179 |
| 46.2.76 | Reception Interrupt Enable register 1 (RIE1)..... | 46-182 |
| 46.2.77 | Reception Interrupt Disable register 1 (RID1)..... | 46-185 |
| 46.2.78 | Reception Interrupt Enable register 2 (RIE2)..... | 46-188 |
| 46.2.79 | Reception Interrupt Disable register 2 (RID2)..... | 46-191 |

| | | |
|----------|--|--------|
| 46.2.80 | Transmission Interrupt Enable register (TIE) | 46-194 |
| 46.2.81 | Transmission Interrupt Disable register (TID)..... | 46-196 |
| 46.2.82 | Reception Interrupt Enable register 3 (RIE3) | 46-198 |
| 46.2.83 | Reception Interrupt Disable register 3 (RID3)..... | 46-201 |
| 46.2.84 | E-MAC Mode Register (ECMR) | 46-204 |
| 46.2.85 | Receive Frame Length Register (RFLR) | 46-207 |
| 46.2.86 | E-MAC Status Register (ECSR) | 46-208 |
| 46.2.87 | E-MAC Interrupt Permission Register (ECSIPR)..... | 46-210 |
| 46.2.88 | PHY Interface Register (PIR) | 46-211 |
| 46.2.89 | PHY Status Register (PSR)..... | 46-212 |
| 46.2.90 | PHY_INT Polarity Register (PIPR)..... | 46-213 |
| 46.2.91 | Automatic PAUSE frame register (APR) | 46-214 |
| 46.2.92 | Manual PAUSE Frame Register (MPR) | 46-215 |
| 46.2.93 | PAUSE Frame Transmit Counter (PFTCR)..... | 46-216 |
| 46.2.94 | PAUSE Frame Receive Counter (PFRCR) | 46-217 |
| 46.2.95 | Automatic PAUSE frame retransmit count register (TPAUSER)..... | 46-218 |
| 46.2.96 | PAUSE frame transmit times counter (PFTTCR)..... | 46-219 |
| 46.2.97 | E-MAC Mode Register 2 (GECMR) | 46-220 |
| 46.2.98 | E-MAC Address High Register (MAHR)..... | 46-221 |
| 46.2.99 | E-MAC Address Low Register (MALR)..... | 46-222 |
| 46.2.100 | Transmit retry over counter register (TROCR)..... | 46-223 |
| 46.2.101 | CRC Error Frame Receive Counter Register (CEFCR)..... | 46-224 |
| 46.2.102 | Frame Receive Error Counter Register (FRECR)..... | 46-225 |
| 46.2.103 | Too-Short Frame Receive Counter Register (TSFRCR)..... | 46-226 |
| 46.2.104 | Too-Long Frame Receive Counter Register (TLFRCR)..... | 46-227 |
| 46.2.105 | Residual-Bit Frame Receive Counter Register (RFCR)..... | 46-228 |
| 46.2.106 | Multicast Address Frame Receive Counter Register (MAFCR)..... | 46-229 |
| 46.3 | Operation | 46-230 |
| 46.3.1 | AVB-DMAC Operating Modes | 46-231 |
| 46.3.2 | Common Control for Transmission and Reception..... | 46-235 |
| 46.3.3 | Descriptors | 46-243 |
| 46.3.4 | Control in Reception | 46-255 |
| 46.3.5 | Transmission Control..... | 46-270 |
| 46.3.6 | CBS (Credit-Based Shaping) | 46-285 |
| 46.3.7 | Time Synchronization..... | 46-291 |
| 46.3.8 | Support for IEEE 1722..... | 46-294 |
| 46.3.9 | Flow Control | 46-305 |
| 46.3.10 | Magic Packet Detection | 46-306 |
| 46.3.11 | Interrupts..... | 46-307 |
| 46.3.12 | Flows of Operations..... | 46-313 |
| 46.3.13 | Connection to PHY-LSI..... | 46-326 |
| 46.3.14 | Usage Notes | 46-331 |
| 46.4 | Usage Notes | 46-332 |
| 46.4.1 | Checksum Calculation of Ethernet Frames..... | 46-332 |
| 46.4.2 | Data Transfer Function Stops after Read Access Error | 46-333 |
| 46.4.3 | Power Down Request..... | 46-334 |
| 47. | Controller Area Network Interface (CAN interface) | 47-1 |
| 47.1 | Overview..... | 47-1 |
| 47.1.1 | Features..... | 47-1 |
| 47.1.2 | Block Diagram | 47-3 |
| 47.1.3 | External Pins | 47-5 |
| 47.1.4 | Connected Module | 47-5 |
| 47.1.5 | Register Configuration..... | 47-6 |

| | | |
|---------|---|-------|
| 47.2 | Register Description..... | 47-10 |
| 47.2.1 | CANi Control Register (CiCTLR) (i = 0, 1)..... | 47-10 |
| 47.2.2 | CANi Clock Select Register (CiCLKR) (i = 0, 1)..... | 47-15 |
| 47.2.3 | CANi Bit Configuration Register (CiBCR) (i = 0, 1)..... | 47-16 |
| 47.2.4 | CANi Mask Register k (CiMKRk) (i = 0, 1; k = 0 to 9)..... | 47-18 |
| 47.2.5 | CANi FIFO Received ID Compare Registers (CiFIDCR0 and CiFIDCR1) (i = 0, 1)..... | 47-20 |
| 47.2.6 | CANi Mask Invalid Registers (CiMKIVLR0 and CiMKIVLR1) (i = 0, 1)..... | 47-22 |
| 47.2.7 | CANi Mailbox Register j (CiMBj) (i = 0, 1; j = 0 to 63)..... | 47-24 |
| 47.2.8 | CANi Mailbox Interrupt Enable Registers (CiMIER0 and CiMIER1) (i = 0, 1)..... | 47-28 |
| 47.2.9 | CANi Message Control Register j (CiMCTLj) (i = 0, 1; j = 0 to 63)..... | 47-31 |
| 47.2.10 | CANi Receive FIFO Control Register (CiRFCR) (i = 0, 1)..... | 47-36 |
| 47.2.11 | CANi Receive FIFO Pointer Control Register (CiRFPCR) (i = 0, 1)..... | 47-39 |
| 47.2.12 | CANi Transmit FIFO Control Register (CiTFCR) (i = 0, 1)..... | 47-40 |
| 47.2.13 | CANi Transmit FIFO Pointer Control Register (CiTFPCR) (i = 0, 1)..... | 47-43 |
| 47.2.14 | CANi Status Register (CiSTR) (i = 0, 1)..... | 47-44 |
| 47.2.15 | CANi Mailbox Search Mode Register (CiMSMR) (i = 0, 1)..... | 47-47 |
| 47.2.16 | CANi Mailbox Search Status Register (CiMSSR) (i = 0, 1)..... | 47-48 |
| 47.2.17 | CANi Channel Search Support Register (CiCSSR) (i = 0, 1)..... | 47-50 |
| 47.2.18 | CANi Acceptance Filter Support Register (CiAFSR) (i = 0, 1)..... | 47-51 |
| 47.2.19 | CANi Error Interrupt Enable Register (CiEIER) (i = 0, 1)..... | 47-52 |
| 47.2.20 | CANi Error Interrupt Factor Judge Register (CiEIFR) (i = 0, 1)..... | 47-54 |
| 47.2.21 | CANi Receive Error Count Register (CiRECR) (i = 0, 1)..... | 47-58 |
| 47.2.22 | CANi Transmit Error Count Register (CiTECR) (i = 0, 1)..... | 47-59 |
| 47.2.23 | CANi Error Code Store Register (CiECSR) (i = 0, 1)..... | 47-60 |
| 47.2.24 | CANi Time Stamp Register (CiTSR) (i = 0, 1)..... | 47-62 |
| 47.2.25 | CANi Test Control Register (CiTCR) (i = 0, 1)..... | 47-63 |
| 47.2.26 | CANi Interrupt Status Register (CiISR) (i = 0, 1)..... | 47-66 |
| 47.2.27 | CANi Interrupt Enable Register (CiIER) (i = 0, 1)..... | 47-68 |
| 47.2.28 | CANi Mailbox Search Mask Register (CiMBSMR) (i = 0, 1)..... | 47-69 |
| 47.2.29 | CANi Parity Error Control Register (CiPECR) (i = 0, 1)..... | 47-70 |
| 47.2.30 | CANi Parity Error Address Capture Register (CiPEACR) (i = 0, 1)..... | 47-72 |
| 47.3 | Operating Mode..... | 47-73 |
| 47.3.1 | CAN Reset Mode..... | 47-74 |
| 47.3.2 | CAN Halt Mode..... | 47-75 |
| 47.3.3 | CAN Sleep Mode..... | 47-76 |
| 47.3.4 | CAN Operation Mode (Excluding Bus-Off State)..... | 47-76 |
| 47.3.5 | CAN Operation Mode (Bus-Off State)..... | 47-77 |
| 47.4 | CAN Communication Speed Configuration..... | 47-78 |
| 47.4.1 | CAN Clock Configuration..... | 47-78 |
| 47.4.2 | Bit Timing Configuration..... | 47-78 |
| 47.4.3 | Bit-rate..... | 47-79 |
| 47.5 | Mailbox and Mask Register Structure..... | 47-80 |
| 47.6 | Acceptance Filtering and Masking Function..... | 47-82 |
| 47.7 | Reception and Transmission..... | 47-85 |
| 47.7.1 | Reception..... | 47-86 |
| 47.7.2 | Transmission..... | 47-88 |
| 47.8 | CAN Interrupt..... | 47-89 |
| 47.9 | CAN module can support Parity Check..... | 47-93 |
| 47.9.1 | Overview..... | 47-93 |
| 47.9.2 | Recommend Usage..... | 47-93 |
| 47.9.3 | Failure Control..... | 47-93 |
| 48. | CAN-FD..... | 48-1 |
| 48.1 | Overview..... | 48-1 |

| | | |
|----------|---|-------|
| 48.1.1 | Functional Overview..... | 48-1 |
| 48.1.2 | Interface Mode..... | 48-5 |
| 48.2 | Block Diagram..... | 48-6 |
| 48.3 | External Pins..... | 48-7 |
| 48.4 | Connected module..... | 48-7 |
| 48.5 | Features of RS-CANFD..... | 48-8 |
| 48.5.1 | Register Base Address..... | 48-8 |
| 48.6 | Registers Configuration (classical CAN mode)..... | 48-9 |
| 48.6.1 | List of Registers..... | 48-9 |
| 48.7 | Register Description (classical CAN mode)..... | 48-15 |
| 48.7.1 | Details of Interface Mode Related Registers..... | 48-15 |
| 48.7.1.1 | RSCFDnCFDGRMCFG - Global Interface Mode Select Register..... | 48-15 |
| 48.7.2 | Details of Channel Related Registers..... | 48-16 |
| 48.7.2.1 | RSCFDnCmCFG — Channel Configuration Register (m = 0, 1)..... | 48-16 |
| 48.7.2.2 | RSCFDnCmCTR — Channel Control Register (m = 0 or 1)..... | 48-18 |
| 48.7.2.3 | RSCFDnCmSTS — Channel Status Register (m = 0 or 1)..... | 48-23 |
| 48.7.2.4 | RSCFDnCmERFL — Channel Error Flag Register (m = 0 or 1)..... | 48-25 |
| 48.7.3 | Details of Global Related Registers..... | 48-29 |
| 48.7.3.1 | RSCFDnGCFG - Global Configuration Register..... | 48-29 |
| 48.7.3.2 | RSCFDnGCTR — Global Control Register..... | 48-32 |
| 48.7.3.3 | RSCFDnGSTS — Global Status Register..... | 48-34 |
| 48.7.3.4 | RSCFDnGERFL — Global Error Flag Register..... | 48-35 |
| 48.7.3.5 | RSCFDnGTSC — Global Timestamp Counter Register..... | 48-37 |
| 48.7.3.6 | RSCFDnGTINTSTS0 — Global TX Interrupt Status Register 0..... | 48-38 |
| 48.7.3.7 | RSCFDnGTINTSTS1 — Global TX Interrupt Status Register 1..... | 48-41 |
| 48.7.4 | Details of Receive Rule Related Registers..... | 48-42 |
| 48.7.4.1 | RSCFDnGAFLECTR — Receive Rule Entry Control Register..... | 48-42 |
| 48.7.4.2 | RSCFDnGAFLCFG0 — Receive Rule Configuration Register 0..... | 48-43 |
| 48.7.4.3 | RSCFDnGAFLCFG1 — Receive Rule Configuration Register 1..... | 48-44 |
| 48.7.4.4 | RSCFDnGAFLIDj — Receive Rule ID Register (j = 0 to 15)..... | 48-45 |
| 48.7.4.5 | RSCFDnGAFLMj — Receive Rule Mask Register (j = 0 to 15)..... | 48-47 |
| 48.7.4.6 | RSCFDnGAFLP0_j — Receive Rule Pointer 0 Register (j = 0 to 15)..... | 48-48 |
| 48.7.4.7 | RSCFDnGAFLP1_j — Receive Rule Pointer 1 Register (j = 0 to 15)..... | 48-50 |
| 48.7.5 | Details of Receive Buffer Related Registers..... | 48-51 |
| 48.7.5.1 | RSCFDnRMNB - Receive Buffer Number Register..... | 48-51 |
| 48.7.5.2 | RSCFDnRMNDy — Receive Buffer New Data Register (y = 0)..... | 48-52 |
| 48.7.5.3 | RSCFDnRMIDq — Receive Buffer ID Register (q = 0 to 31)..... | 48-53 |
| 48.7.5.4 | RSCFDnRMPTRq — Receive Buffer Pointer Register (q = 0 to 31)..... | 48-54 |
| 48.7.5.5 | RSCFDnRMDf0_q — Receive Buffer Data Field 0 Register (q = 0 to 31)..... | 48-55 |
| 48.7.5.6 | RSCFDnRMDf1_q — Receive Buffer Data Field 1 Register (q = 0 to 31)..... | 48-56 |
| 48.7.6 | Details of Receive FIFO Buffer Related Registers..... | 48-57 |
| 48.7.6.1 | RSCFDnRFCCx — Receive FIFO Buffer Configuration and Control Register (x = 0 to 7)..... | 48-57 |
| 48.7.6.2 | RSCFDnRFSTsx — Receive FIFO Buffer Status Register (x = 0 to 7)..... | 48-59 |
| 48.7.6.3 | RSCFDnRFPCTRx — Receive FIFO Buffer Pointer Control Register (x = 0 to 7)..... | 48-61 |
| 48.7.6.4 | RSCFDnRFIDx — Receive FIFO Buffer Access ID Register (x = 0 to 7)..... | 48-62 |
| 48.7.6.5 | RSCFDnRFPTRx — Receive FIFO Buffer Access Pointer Register (x = 0 to 7)..... | 48-63 |
| 48.7.6.6 | RSCFDnRFDf0_x — Receive FIFO Buffer Access Data Field 0 Register (x = 0 to 7)..... | 48-64 |
| 48.7.6.7 | RSCFDnRFDf1_x — Receive FIFO Buffer Access Data Field 1 Register (x = 0 to 7)..... | 48-65 |
| 48.7.7 | Details of Transmit/Receive FIFO Buffer Related Registers..... | 48-66 |
| 48.7.7.1 | RSCFDnCFCCk — Transmit/receive FIFO Buffer Configuration and Control Register k (k = 0 to 5)..... | 48-66 |
| 48.7.7.2 | RSCFDnCFSTSk — Transmit/receive FIFO Buffer Status Register (k = 0 to 5)..... | 48-70 |
| 48.7.7.3 | RSCFDnCFPCTRk — Transmit/receive FIFO Buffer Pointer Control Register (k = 0 to 5)..... | 48-74 |

| | | |
|-----------|---|--------|
| 48.7.7.4 | RSCFDnCFIDk — Transmit/receive FIFO Buffer Access ID Register (k = 0 to 5)..... | 48-76 |
| 48.7.7.5 | RSCFDnCFPTRk — Transmit/receive FIFO Buffer Access Pointer Register (k = 0 to 5) .. | 48-78 |
| 48.7.7.6 | RSCFDnCFDF0_k — Transmit/receive FIFO Buffer Access Data Field 0 Register (k = 0 to 5)..... | 48-80 |
| 48.7.7.7 | RSCFDnCFDF1_k — Transmit/receive FIFO Buffer Access Data Field 1 Register (k = 0 to 5)..... | 48-81 |
| 48.7.8 | Details of FIFO Status Related Registers..... | 48-82 |
| 48.7.8.1 | RSCFDnFESTS — FIFO Empty Status Register | 48-82 |
| 48.7.8.2 | RSCFDnFFSTS — FIFO Full Status Register..... | 48-84 |
| 48.7.8.3 | RSCFDnFMSTS — FIFO Message Lost Status Register | 48-86 |
| 48.7.8.4 | RSCFDnRFISTS — Receive FIFO Buffer Interrupt Flag Status Register | 48-88 |
| 48.7.8.5 | RSCFDnCFRISTS — Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register..... | 48-89 |
| 48.7.8.6 | RSCFDnCFRISTS — Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register..... | 48-90 |
| 48.7.9 | Details of Transmit Buffer Related Registers | 48-91 |
| 48.7.9.1 | RSCFDnTMCp — Transmit Buffer Control Register (p = 0 to 31)..... | 48-91 |
| 48.7.9.2 | RSCFDnTMSTSp — Transmit Buffer Status Register (p = 0 to 31)..... | 48-93 |
| 48.7.9.3 | RSCFDnTMIDp — Transmit Buffer ID Register (p = 0 to 31)..... | 48-94 |
| 48.7.9.4 | RSCFDnTMPTRp — Transmit Buffer Pointer Register (p = 0 to 31)..... | 48-96 |
| 48.7.9.5 | RSCFDnTMDf0_p — Transmit Buffer Data Field 0 Register (p = 0 to 31)..... | 48-98 |
| 48.7.9.6 | RSCFDnTMDf1_p — Transmit Buffer Data Field 1 Register (p = 0 to 31)..... | 48-99 |
| 48.7.9.7 | RSCFDnTMIECy — Transmit Buffer Interrupt Enable Configuration Register (y = 0).... | 48-100 |
| 48.7.10 | Details of Transmit Buffer Status Related Registers | 48-102 |
| 48.7.10.1 | RSCFDnTMTRSTSy — Transmit Buffer Transmit Request Status Register (y = 0)..... | 48-102 |
| 48.7.10.2 | RSCFDnTMTARSTSy — Transmit Buffer Transmit Abort Request Status Register (y = 0)..... | 48-104 |
| 48.7.10.3 | RSCFDnTMTcSTSy — Transmit Buffer Transmit Complete Status Register (y = 0)..... | 48-106 |
| 48.7.10.4 | RSCFDnTMTASTSy — Transmit Buffer Transmit Abort Status Register (y = 0)..... | 48-108 |
| 48.7.11 | Details of Transmit Queue Related Registers | 48-110 |
| 48.7.11.1 | RSCFDnTXQCCm — Transmit Queue Configuration and Control Register (m = 0 or 1) | 48-110 |
| 48.7.11.2 | RSCFDnTXQSTSm — Transmit Queue Status Register (m = 0 or 1) | 48-112 |
| 48.7.11.3 | RSCFDnTXQPCTrm — Transmit Queue Pointer Control Register (m = 0 or 1) | 48-114 |
| 48.7.12 | Details of Transmit History Related Registers..... | 48-115 |
| 48.7.12.1 | RSCFDnTHLCCm — Transmit History Configuration and Control Register (m = 0 or 1) | 48-115 |
| 48.7.12.2 | RSCFDnTHLSTSm — Transmit History Status Register (m = 0 or 1)..... | 48-117 |
| 48.7.12.3 | RSCFDnTHLPCTrm — Transmit History Pointer Control Register (m = 0 or 1) | 48-119 |
| 48.7.12.4 | RSCFDnTHLACcm — Transmit History Access Register (m = 0 or 1) | 48-120 |
| 48.7.13 | Details of Test Related Registers | 48-121 |
| 48.7.13.1 | RSCFDnGTSTCFG — Global Test Configuration Register | 48-121 |
| 48.7.13.2 | RSCFDnGTSTCTR — Global Test Control Register | 48-122 |
| 48.7.13.3 | RSCFDnGLOCKK — Global Lock Key Register..... | 48-123 |
| 48.7.13.4 | RSCFDnRPGACCr — RAM Test Page Access Register (r = 0 to 63)..... | 48-124 |
| 48.8 | Registers Configuration (CAN FD mode) | 48-125 |
| 48.8.1 | List of Registers | 48-125 |
| 48.9 | Register Description (CAN FD mode)..... | 48-131 |
| 48.9.1 | Details of Interface Mode Related Registers..... | 48-131 |
| 48.9.1.1 | RSCFDnCFDGRMCFG — Global Interface Mode Select Register | 48-131 |
| 48.9.2 | Details of Channel Related Registers..... | 48-132 |
| 48.9.2.1 | RSCFDnCFDCmNCFG — Channel Normal Bit Rate Configuration Register (m = 0 or 1)..... | 48-132 |
| 48.9.2.2 | RSCFDnCFDCmCTR — Channel Control Register (m = 0 or 1)..... | 48-134 |

| | | |
|----------|--|--------|
| 48.9.2.3 | RSCFDnCFDCmSTS — Channel Status Register (m = 0 or 1) | 48-139 |
| 48.9.2.4 | RSCFDnCFDCmERFL — Channel Error Flag Status Register (m = 0 or 1) | 48-142 |
| 48.9.2.5 | RSCFDnCFDCmDCFG — Channel Data Bit Rate Configuration Register (m = 0 or 1)... .. | 48-146 |
| 48.9.2.6 | RSCFDnCFDCmFDCFG — Channel CAN FD Configuration Register (m = 0 or 1) | 48-148 |
| 48.9.2.7 | RSCFDnCFDCmFDCTR — Channel CAN FD Control Register (m = 0 or 1)..... | 48-152 |
| 48.9.2.8 | RSCFDnCFDCmFDSTS — Channel CAN FD Status Register (m = 0 or 1)..... | 48-153 |
| 48.9.2.9 | RSCFDnCFDCmFDCRC — Channel CAN FD CRC Register (m = 0 or 1)..... | 48-155 |
| 48.9.3 | Details of Global Related Registers | 48-156 |
| 48.9.3.1 | RSCFDnCFDGCFG — Global Configuration Register | 48-156 |
| 48.9.3.2 | RSCFDnCFDGCTR — Global Control Register..... | 48-159 |
| 48.9.3.3 | RSCFDnCFDGSTS — Global Status Register..... | 48-161 |
| 48.9.3.4 | RSCFDnCFDGERFL — Global Error Flag Register | 48-163 |
| 48.9.3.5 | RSCFDnCFDGTSC — Global Timestamp Counter Register | 48-165 |
| 48.9.3.6 | RSCFDnCFDGTINTSTS0 — Global TX Interrupt Status Register 0..... | 48-166 |
| 48.9.3.7 | RSCFDnCFDGTINTSTS1 — Global TX Interrupt Status Register 1..... | 48-169 |
| 48.9.4 | Details of Receive Rule Related Registers..... | 48-170 |
| 48.9.4.1 | RSCFDnCFDGAFLECTR — Receive Rule Entry Control Register | 48-170 |
| 48.9.4.2 | RSCFDnCFDGAFLCFG0 — Receive Rule Configuration Register 0..... | 48-171 |
| 48.9.4.3 | RSCFDnCFDGAFLCFG1 — Receive Rule Configuration Register 1..... | 48-172 |
| 48.9.4.4 | RSCFDnCFDGAFIDj — Receive Rule ID Register (j = 0 to 15)..... | 48-173 |
| 48.9.4.5 | RSCFDnCFDGAFMj — Receive Rule Mask Register (j = 0 to 15)..... | 48-175 |
| 48.9.4.6 | RSCFDnCFDGAFLP0_j — Receive Rule Pointer 0 Register (j = 0 to 15)..... | 48-176 |
| 48.9.4.7 | RSCFDnCFDGAFLP1_j — Receive Rule Pointer 1 Register (j = 0 to 15)..... | 48-178 |
| 48.9.5 | Details of Receive Buffer Related Registers..... | 48-179 |
| 48.9.5.1 | RSCFDnCFDRMNB — Receive Buffer Number Register | 48-179 |
| 48.9.5.2 | RSCFDnCFDRMNDy — Receive Buffer New Data Register (y = 0) | 48-180 |
| 48.9.5.3 | RSCFDnCFDRMIDq — Receive Buffer ID Register (q = 0 to 31)..... | 48-181 |
| 48.9.5.4 | RSCFDnCFDRMPTRq — Receive Buffer Pointer Register (q = 0 to 31) | 48-182 |
| 48.9.5.5 | RSCFDnCFDRMFDSTSq — Receive Buffer CAN FD Status Register (q = 0 to 31)..... | 48-184 |
| 48.9.5.6 | RSCFDnCFDRMDFb_q — Receive Buffer Data Field b Register (b = 0 to 4, q = 0 to 31)..... | 48-185 |
| 48.9.6 | Details of Receive FIFO Buffer Related Registers | 48-186 |
| 48.9.6.1 | RSCFDnCFDRFCCx — Receive FIFO Buffer Configuration and Control Register (x = 0 to 7)..... | 48-186 |
| 48.9.6.2 | RSCFDnCFDRFSTSx — Receive FIFO Buffer Status Register (x = 0 to 7) | 48-188 |
| 48.9.6.3 | RSCFDnCFDRFPCTR _x — Receive FIFO Buffer Pointer Control Register (x = 0 to 7) ... | 48-190 |
| 48.9.6.4 | RSCFDnCFDRFID _x — Receive FIFO Buffer Access ID Register (x = 0 to 7)..... | 48-191 |
| 48.9.6.5 | RSCFDnCFDRFPTR _x — Receive FIFO Buffer Access Pointer Register (x = 0 to 7)..... | 48-192 |
| 48.9.6.6 | RSCFDnCFDRFFDSTS _x — Receive FIFO CAN FD Status Register (x = 0 to 7)..... | 48-194 |
| 48.9.6.7 | RSCFDnCFDRFDFd_x — Receive FIFO Buffer Access Data Field d Register (d = 0 to 15, x = 0 to 7)..... | 48-195 |
| 48.9.7 | Details of Transmit/Receive FIFO Buffer Related Registers..... | 48-196 |
| 48.9.7.1 | RSCFDnCFDCFCCK — Transmit/receive FIFO Buffer Configuration and Control Register k (k = 0 to 5)..... | 48-196 |
| 48.9.7.2 | RSCFDnCFDCFSTSk — Transmit/receive FIFO Buffer Status Register (k = 0 to 5)..... | 48-200 |
| 48.9.7.3 | RSCFDnCFDCFPCTR _k — Transmit/receive FIFO Buffer Pointer Control Register (k = 0 to 5)..... | 48-204 |
| 48.9.7.4 | RSCFDnCFDCFID _k — Transmit/receive FIFO Buffer Access ID Register (k = 0 to 5)..... | 48-206 |
| 48.9.7.5 | RSCFDnCFDCFPTR _k — Transmit/receive FIFO Buffer Access Pointer Register (k = 0 to 5)..... | 48-208 |
| 48.9.7.6 | RSCFDnCFDCFFDCSTSk — Transmit/receive FIFO CAN FD Configuration and Status Register (k = 0 to 5) | 48-211 |

| | | |
|-----------|--|--------|
| 48.9.7.7 | RSCFDnCFDCFDf_k — Transmit/receive FIFO Buffer Access Data Field d Register (d = 0 to 15, k = 0 to 5)..... | 48-213 |
| 48.9.8 | Details of FIFO Status Related Registers..... | 48-214 |
| 48.9.8.1 | RSCFDnCFDFESTS — FIFO Empty Status Register..... | 48-214 |
| 48.9.8.2 | RSCFDnCFDFFSTS — FIFO Full Status Register..... | 48-216 |
| 48.9.8.3 | RSCFDnCFDFMSTS — FIFO Message Lost Status Register..... | 48-218 |
| 48.9.8.4 | RSCFDnCFDRFISTS — Receive FIFO Buffer Interrupt Flag Status Register..... | 48-220 |
| 48.9.8.5 | RSCFDnCFDCFRISTS — Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register..... | 48-221 |
| 48.9.8.6 | RSCFDnCFDCFTISTS — Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register..... | 48-222 |
| 48.9.9 | Details of FIFO DMA Related Registers..... | 48-223 |
| 48.9.9.1 | RSCFDnCFDCDTCT — DMA Enable Register..... | 48-223 |
| 48.9.9.2 | RSCFDnCFDCDTSTS — DMA Status Register..... | 48-225 |
| 48.9.10 | Details of Transmit Buffer Related Registers..... | 48-227 |
| 48.9.10.1 | RSCFDnCFDTMCp — Transmit Buffer Control Register (p = 0 to 31)..... | 48-227 |
| 48.9.10.2 | RSCFDnCFDTMSTSp — Transmit Buffer Status Register (p = 0 to 31)..... | 48-229 |
| 48.9.10.3 | RSCFDnCFDTMIDp — Transmit Buffer ID Register (p = 0 to 31)..... | 48-231 |
| 48.9.10.4 | RSCFDnCFDTMPTRp — Transmit Buffer Pointer Register (p= 0 to 31)..... | 48-233 |
| 48.9.10.5 | RSCFDnCFDTMFDCTRp — Transmit Buffer CAN FD Configuration Register (p = 0 to 31)..... | 48-235 |
| 48.9.10.6 | RSCFDnCFDTMDFb_p — Transmit Buffer Data Field b Register (b = 0 to 4, p = 0 to 31)..... | 48-237 |
| 48.9.10.7 | RSCFDnCFDTMIECy — Transmit Buffer Interrupt Enable Configuration Register (y = 0)..... | 48-238 |
| 48.9.11 | Details of Transmit Buffer Status Related Registers..... | 48-240 |
| 48.9.11.1 | RSCFDnCFDTMTRSTSy — Transmit Buffer Transmit Request Status Register (y = 0)..... | 48-240 |
| 48.9.11.2 | RSCFDnCFDTMTARSTSy — Transmit Buffer Transmit Abort Request Status Register (y = 0)..... | 48-242 |
| 48.9.11.3 | RSCFDnCFDTMTCSTSy — Transmit Buffer Transmit Complete Status Register (y = 0)..... | 48-244 |
| 48.9.11.4 | RSCFDnCFDTMTASTSy — Transmit Buffer Transmit Abort Status Register (y = 0).... | 48-246 |
| 48.9.12 | Details of Transmit Queue Related Registers..... | 48-248 |
| 48.9.12.1 | RSCFDnCFDTXQCCm — Transmit Queue Configuration and Control Register (m = 0 or 1)..... | 48-248 |
| 48.9.12.2 | RSCFDnCFDTXQSTSm — Transmit Queue Status Register (m = 0 or 1)..... | 48-250 |
| 48.9.12.3 | RSCFDnCFDTXQPCTRm — Transmit Queue Pointer Control Register (m = 0 or 1).... | 48-252 |
| 48.9.13 | Details of Transmit History Related Registers..... | 48-253 |
| 48.9.13.1 | RSCFDnCFDTHLCCm — Transmit History Configuration and Control Register (m = 0 or 1)..... | 48-253 |
| 48.9.13.2 | RSCFDnCFDTHLSTSm — Transmit History Status Register (m = 0 or 1)..... | 48-255 |
| 48.9.13.3 | RSCFDnCFDTHLPCTRm — Transmit History Pointer Control Register (m = 0 or 1).... | 48-257 |
| 48.9.13.4 | RSCFDnCFDTHLACCM — Transmit History Access Register (m = 0 or 1)..... | 48-258 |
| 48.9.14 | Details of Test Related Registers..... | 48-259 |
| 48.9.14.1 | RSCFDnCFDGTSTCFG — Global Test Configuration Register..... | 48-259 |
| 48.9.14.2 | RSCFDnCFDGTSTCTR — Global Test Control Register..... | 48-260 |
| 48.9.14.3 | RSCFDnCFDGLOCKK — Global Lock Key Register..... | 48-261 |
| 48.9.14.4 | RSCFDnCFDRPGACCr — RAM Test Page Access Register (r = 0 to 63)..... | 48-262 |
| 48.10 | Interrupt Sources and DMA Triggers..... | 48-263 |
| 48.10.1 | Interrupt Sources..... | 48-263 |
| 48.10.2 | DMA Triggers (CAN FD Mode Only)..... | 48-266 |
| 48.11 | CAN Modes..... | 48-267 |
| 48.11.1 | Global Modes..... | 48-267 |

| | | |
|-----------|--|--------|
| 48.11.1.1 | Global Stop Mode | 48-268 |
| 48.11.1.2 | Global Reset Mode..... | 48-268 |
| 48.11.1.3 | Global Test Mode..... | 48-269 |
| 48.11.1.4 | Global Operating Mode..... | 48-269 |
| 48.11.2 | Channel Modes | 48-270 |
| 48.11.2.1 | Channel Stop Mode..... | 48-271 |
| 48.11.2.2 | Channel Reset Mode | 48-271 |
| 48.11.2.3 | Channel Halt Mode | 48-271 |
| 48.11.2.4 | Channel Communication Mode | 48-272 |
| 48.11.2.5 | Bus Off State | 48-273 |
| 48.11.3 | Registers Initialized by Transition to CAN Mode | 48-274 |
| 48.12 | Reception Function | 48-276 |
| 48.12.1 | Data Processing Using the Receive Rule Table | 48-276 |
| 48.12.1.1 | Acceptance Filter Processing | 48-278 |
| 48.12.1.2 | DLC Filter Processing..... | 48-278 |
| 48.12.1.3 | Routing Processing..... | 48-279 |
| 48.12.1.4 | Label Addition Processing | 48-279 |
| 48.12.1.5 | Mirror Function Processing..... | 48-279 |
| 48.12.1.6 | Timestamp..... | 48-279 |
| 48.13 | Transmission Function..... | 48-281 |
| 48.13.1 | Transmit Priority Determination..... | 48-282 |
| 48.13.2 | Transmission Using Transmit Buffers | 48-282 |
| 48.13.2.1 | Transmit Abort Function..... | 48-282 |
| 48.13.2.2 | One-Shot Transmission Function (Retransmission Disabling Function) | 48-282 |
| 48.13.2.3 | Transmit Buffer Merge Mode (CAN FD Mode Only)..... | 48-283 |
| 48.13.3 | Transmission Using FIFO Buffers..... | 48-283 |
| 48.13.3.1 | Interval Transmission Function..... | 48-283 |
| 48.13.4 | Transmission Using Transmit Queues | 48-286 |
| 48.13.5 | Padding Transmit Data (CAN FD Mode Only) | 48-286 |
| 48.13.6 | Transmit History Function..... | 48-286 |
| 48.14 | Gateway Function | 48-288 |
| 48.14.1 | CAN-CAN FD Gateway (CAN FD Mode Only)..... | 48-288 |
| 48.15 | Test Function | 48-289 |
| 48.15.1 | Standard Test Mode | 48-289 |
| 48.15.2 | Listen-Only Mode..... | 48-289 |
| 48.15.3 | Self-Test Mode (Loopback Mode)..... | 48-290 |
| 48.15.3.1 | Self-Test Mode 0 (External Loopback Mode)..... | 48-290 |
| 48.15.3.2 | Self-Test Mode 1 (Internal Loopback Mode)..... | 48-290 |
| 48.15.4 | Limited Operation Mode (CAN FD Mode Only) | 48-291 |
| 48.15.5 | RAM Test | 48-291 |
| 48.15.6 | Inter-Channel Communication Test..... | 48-291 |
| 48.15.6.1 | CRC Error Test | 48-292 |
| 48.16 | RS-CANFD Setting Procedure | 48-293 |
| 48.16.1 | Initial Settings..... | 48-293 |
| 48.16.1.1 | Setting a Clock | 48-294 |
| 48.16.1.2 | Setting Bit Timing..... | 48-294 |
| 48.16.1.3 | Setting Communication Speed | 48-296 |
| 48.16.1.4 | Setting Receive Rules | 48-298 |
| 48.16.1.5 | Setting Buffers | 48-299 |
| 48.16.1.6 | Transmit Delay Correction (CAN FD Mode Only)..... | 48-302 |
| 48.16.2 | Reception Procedure | 48-303 |
| 48.16.2.1 | Receive Buffer Reading Procedure | 48-303 |
| 48.16.2.2 | FIFO Buffer Reading Procedure | 48-304 |
| 48.16.2.3 | Procedure for Reading FIFO Buffers by the DMA Transfer..... | 48-307 |

| | | |
|------------|---|-------------|
| 48.16.3 | Transmission Procedure..... | 48-308 |
| 48.16.3.1 | Procedure for Transmission from Transmit Buffers..... | 48-308 |
| 48.16.3.2 | Procedure for Transmission from Transmit/Receive FIFO Buffers..... | 48-311 |
| 48.16.3.3 | Procedure for Transmission from the Transmit Queue..... | 48-315 |
| 48.16.3.4 | Transmit History Buffer Reading Procedure..... | 48-315 |
| 48.16.4 | Test Settings..... | 48-316 |
| 48.16.4.1 | Self-Test Mode Setting Procedure..... | 48-316 |
| 48.16.4.2 | Procedure for Releasing the Protection..... | 48-316 |
| 48.16.4.3 | RAM Test Setting Procedure..... | 48-318 |
| 48.16.4.4 | Inter-Channel Communication Test Setting Procedure..... | 48-319 |
| 48.17 | Usage Notes..... | 48-320 |
| 49. | PCIE Controller..... | 49-1 |
| 49.1 | Overview..... | 49-1 |
| 49.1.1 | Features..... | 49-1 |
| 49.1.2 | Block Diagram..... | 49-2 |
| 49.1.3 | External Pins..... | 49-3 |
| 49.1.4 | Register Configuration..... | 49-3 |
| 49.1.5 | Connected Module..... | 49-14 |
| 49.2 | Register Description..... | 49-15 |
| 49.2.1 | Configuration Transmission Address Register (PCIECAR)..... | 49-16 |
| 49.2.2 | Configuration Transmission Control Register (PCIECCTLR)..... | 49-17 |
| 49.2.3 | Configuration Transmission Data Register (PCIECDR)..... | 49-19 |
| 49.2.4 | Mode Setting Register (PCIEMSR)..... | 49-20 |
| 49.2.5 | Unlock Transmission Control Register (PCIEUNLOCKCR)..... | 49-21 |
| 49.2.6 | INTx Register (PCIEINTXR)..... | 49-22 |
| 49.2.7 | Message Reception Status Register (PCIERMSGR)..... | 49-24 |
| 49.2.8 | Message Reception Interrupt Enable Register (PCIERMSGIER)..... | 49-25 |
| 49.2.9 | Power Management Message Transmission Control Register (PCIEPMMSGCR)..... | 49-26 |
| 49.2.10 | Physical Layer Status Register (PCIEPHYSR)..... | 49-27 |
| 49.2.11 | MSI Transmission Register (PCIEMSITXR)..... | 49-28 |
| 49.2.12 | Transfer Control Register (PCIETCTLR)..... | 49-30 |
| 49.2.13 | Transfer Status Register (PCIETSTR)..... | 49-31 |
| 49.2.14 | Interrupt Register (PCIEINTR)..... | 49-32 |
| 49.2.15 | Interrupt Enable Register (PCIEINTER)..... | 49-35 |
| 49.2.16 | Error Factor Register (PCIEERRFR)..... | 49-37 |
| 49.2.17 | Error Interrupt Enable Register (PCIEERRFER)..... | 49-39 |
| 49.2.18 | Error Factor Register 2 (PCIEERRFR2)..... | 49-41 |
| 49.2.19 | Transfer Interrupt Enable Register (PCIETIER)..... | 49-42 |
| 49.2.20 | Power Management State Status Register (PCIEPMSR)..... | 49-43 |
| 49.2.21 | Power Management State Interrupt Enable Register (PCIEPMSCIER)..... | 49-44 |
| 49.2.22 | MSIF Register (PCIEMSIFR)..... | 49-45 |
| 49.2.23 | MSI Address Lower Register (PCIEMSIALR)..... | 49-46 |
| 49.2.24 | MSI Address Upper Register (PCIEMSIAUR)..... | 49-47 |
| 49.2.25 | MSI Interrupt Enable Register (PCIEMSIIER)..... | 49-48 |
| 49.2.26 | PCI Express Root Port Address Register 0 to 5 (PCIEPRAR0 to PCIEPRAR5)..... | 49-49 |
| 49.2.27 | Local Address Register 0 to 5 (PCIELAR0 to PCIELAR5)..... | 49-50 |
| 49.2.28 | Local Address Mask Register 0 to 5 (PCIELAMR0 to PCIELAMR5)..... | 49-51 |
| 49.2.29 | PCIEC Address Lower Register 0 to 3 (PCIEPALR0 to PCIEPALR3)..... | 49-53 |
| 49.2.30 | PCIEC Address Upper Register 0 to 3 (PCIEPAUR0 to PCIEPAUR3)..... | 49-54 |
| 49.2.31 | PCIEC Address Mask Register 0 to 3 (PCIEPAMR0 to PCIEPAMR3)..... | 49-55 |
| 49.2.32 | PCIEC Conversion Control Register 0 to 3 (PCIEPTCTLR0 to PCIEPTCTLR3)..... | 49-56 |
| 49.2.33 | PCIEC DMAC DMA Operation Register (PCIEDMAOR)..... | 49-58 |
| 49.2.34 | PCIEC DMAC PCIEC Address Lower Register 0 to 7 (PCIEDMPALR0 to PCIEDMPALR 7)..... | 49-59 |

| | | |
|---------|---|--------|
| 49.2.35 | PCIEC DMAC PCIEC Address Upper Register 0 to 7 (PCIEDMPAUR0 to PCIEDMPAUR7)..... | 49-59 |
| 49.2.36 | PCIEC DMAC Internal Bus Address Register 0 to 7 (PCIEDMIAR0 to PCIEDMIAR7) | 49-60 |
| 49.2.37 | PCIEC DMAC Byte Count Register 0 to 7 (PCIEDMBCNTR0 to PCIEDMBCNTR7)..... | 49-61 |
| 49.2.38 | PCIEC DMAC Command Chain Address Register 0 to 7 (PCIEDMCCAR0 to PCIEDMCCAR7)... | 49-62 |
| 49.2.39 | PCIEC DMAC Channel Control Register 0 to 7 (PCIEDMCHCR0 to PCIEDMCHCR7) | 49-63 |
| 49.2.40 | PCIEC DMAC Channel Status Register 0 to 7 (PCIEDMCHSR0 to PCIEDMCHSR7)..... | 49-65 |
| 49.2.41 | PCIEC DMAC Channel Control 2 Register 0 to 7 (PCIEDMCHC2R0 to PCIEDMCHC2R7) | 49-68 |
| 49.2.42 | PCI Configuration Register 0 (PCICONF0) | 49-69 |
| 49.2.43 | PCI Configuration Register 1 (PCICONF1) | 49-73 |
| 49.2.44 | PCI Configuration Register 2 (PCICONF2) | 49-77 |
| 49.2.45 | PCI Configuration Register 3 (PCICONF3) | 49-78 |
| 49.2.46 | PCI Configuration Register 4 (PCICONF4) | 49-80 |
| 49.2.47 | PCI Configuration Register 5 (PCICONF5) | 49-81 |
| 49.2.48 | PCI Configuration Register 6 (PCICONF6) | 49-82 |
| 49.2.49 | PCI Configuration Register 7 (PCICONF7) | 49-84 |
| 49.2.50 | PCI Configuration Register 8 (PCICONF8) | 49-87 |
| 49.2.51 | PCI Configuration Register 9 (PCICONF9) | 49-89 |
| 49.2.52 | PCI Configuration Register 10 (PCICONF10) | 49-91 |
| 49.2.53 | PCI Configuration Register 11 (PCICONF11) | 49-92 |
| 49.2.54 | PCI Configuration Register 12 (PCICONF12) | 49-94 |
| 49.2.55 | PCI Configuration Register 13 (PCICONF13) | 49-96 |
| 49.2.56 | PCI Configuration Register 14 (PCICONF14) | 49-97 |
| 49.2.57 | PCI Configuration Register 15 (PCICONF15) | 49-98 |
| 49.2.58 | PCI Power Management Capability Register 0 (PMCAP0) | 49-101 |
| 49.2.59 | PCI Power Management Capability Register 1 (PMCAP1) | 49-104 |
| 49.2.60 | MSI Capability Register 0 (MSICAP0) | 49-106 |
| 49.2.61 | MSI Capability Register 1 (MSICAP1) | 49-108 |
| 49.2.62 | MSI Capability Register 2 (MSICAP2) | 49-109 |
| 49.2.63 | MSI Capability Register 3 (MSICAP3) | 49-110 |
| 49.2.64 | MSI Capability Register 4 (MSICAP4) | 49-111 |
| 49.2.65 | MSI Capability Register 5 (MSICAP5) | 49-112 |
| 49.2.66 | PCI Express Capability Register 0 (EXPCAP0)..... | 49-113 |
| 49.2.67 | PCI Express Capability Register 1 (EXPCAP1)..... | 49-115 |
| 49.2.68 | PCI Express Capability Register 2 (EXPCAP2)..... | 49-118 |
| 49.2.69 | PCI Express Capability Register 3 (EXPCAP3)..... | 49-121 |
| 49.2.70 | PCI Express Capability Register 4 (EXPCAP4)..... | 49-124 |
| 49.2.71 | PCI Express Capability Register 5 (EXPCAP5)..... | 49-128 |
| 49.2.72 | PCI Express Capability Register 6 (EXPCAP6)..... | 49-131 |
| 49.2.73 | PCI Express Capability Register 7 (EXPCAP7)..... | 49-134 |
| 49.2.74 | PCI Express Capability Register 8 (EXPCAP8)..... | 49-136 |
| 49.2.75 | PCI Express Capability Register 9 (EXPCAP9)..... | 49-138 |
| 49.2.76 | PCI Express Capability Register 10 (EXPCAP10)..... | 49-140 |
| 49.2.77 | PCI Express Capability Register 11 (EXPCAP11)..... | 49-142 |
| 49.2.78 | PCI Express Capability Register 12 (EXPCAP12)..... | 49-143 |
| 49.2.79 | PCI Express Capability Register 13 (EXPCAP13)..... | 49-146 |
| 49.2.80 | PCI Express Capability Register 14 (EXPCAP14)..... | 49-147 |
| 49.2.81 | VC Capability Register 0 (VCCAP0) | 49-148 |
| 49.2.82 | VC Capability Register 1 (VCCAP1) | 49-149 |
| 49.2.83 | VC Capability Register 2 (VCCAP2) | 49-150 |
| 49.2.84 | VC Capability Register 3 (VCCAP3) | 49-151 |
| 49.2.85 | VC Capability Register 4 (VCCAP4) | 49-152 |
| 49.2.86 | VC Capability Register 5 (VCCAP5) | 49-154 |
| 49.2.87 | VC Capability Register 6 (VCCAP6) | 49-156 |
| 49.2.88 | Device Serial Number Capability Register 0 (SERNUMCAP0)..... | 49-157 |

| | | |
|----------|---|--------|
| 49.2.89 | Device Serial Number Capability Register 1 (SERNUMCAP1) | 49-158 |
| 49.2.90 | Device Serial Number Capability Register 2 (SERNUMCAP2) | 49-158 |
| 49.2.91 | ID Setting Register 0 (IDSETR0) | 49-159 |
| 49.2.92 | ID Setting Register 1 (IDSETR1) | 49-160 |
| 49.2.93 | SUBID Setting Register (SUBIDSETR) | 49-161 |
| 49.2.94 | Device Serial Number Setting Register 0 (DSERSETR0) | 49-162 |
| 49.2.95 | Device Serial Number Setting Register 1 (DSERSETR1) | 49-162 |
| 49.2.96 | TL Control Register (TLCTLR) | 49-163 |
| 49.2.97 | MAC Status Register (MACSR) | 49-164 |
| 49.2.98 | MAC Control Register (MACCTLR) | 49-167 |
| 49.2.99 | PM Status Register (PMSR) | 49-169 |
| 49.2.100 | PM Control Register (PMCTLR) | 49-170 |
| 49.2.101 | MAC Interrupt Enable Register (MACINTENR) | 49-171 |
| 49.2.102 | PM Interrupt Enable Register (PMINTENR) | 49-173 |
| 49.2.103 | MAC Status Register 2 (MACS2R) | 49-174 |
| 49.2.104 | MAC Control Register 2 (MACCTL2R) | 49-175 |
| 49.2.105 | MAC Speed Change Setting Register (MACCGSPSETR) | 49-176 |
| 49.2.106 | AXI Cache Control Register (AXI_CACHE_REG) | 49-177 |
| 49.2.107 | PHY clock Register (PHY_CLK) | 49-180 |
| 49.3 | Operation | 49-181 |
| 49.3.1 | Initialization | 49-181 |
| 49.3.2 | Configuration Cycle | 49-184 |
| 49.3.3 | Messages and MSI | 49-188 |
| 49.3.4 | Overview of Transmission and Reception | 49-194 |
| 49.3.5 | PIO Transfer (Data Transfer from Internal Bus → PCIEC → External Device) | 49-195 |
| 49.3.6 | Target Transfer (Data Transfer from External Device → PCIEC → Internal Bus) | 49-200 |
| 49.3.7 | DMA Transfer | 49-207 |
| 49.3.8 | Transfer Priority | 49-212 |
| 49.3.9 | Link Function | 49-215 |
| 49.3.10 | Power Management | 49-218 |
| 49.3.11 | Error Handling | 49-223 |
| 49.3.12 | Interrupts | 49-229 |
| 49.3.13 | Compliance Test | 49-234 |
| 49.3.14 | Internal Reference Clock Supply | 49-234 |
| 49.4 | Usage Notes | 49-235 |
| 49.4.1 | Write/Read Attributes of Configuration Registers | 49-235 |
| 49.4.2 | Notes on Accessing Configuration Registers | 49-236 |
| 49.4.3 | Registers Which Affect Hardware Internal Operation in Root Port Mode | 49-236 |
| 49.4.4 | Power Management States | 49-239 |
| 49.4.5 | State Transition | 49-239 |
| 50. | Serial Communication Interface with FIFO (SCIF) | 50-1 |
| 50.1 | Overview | 50-1 |
| 50.1.1 | Features | 50-3 |
| 50.1.2 | Block Diagram | 50-4 |
| 50.1.3 | External Pins | 50-5 |
| 50.1.4 | Register Configuration | 50-6 |
| 50.1.5 | Connected Module | 50-7 |
| 50.2 | Register Description | 50-8 |
| 50.2.1 | Receive Shift Register (SCRSR) | 50-8 |
| 50.2.2 | Receive FIFO Data Register (SCFRDR) | 50-9 |
| 50.2.3 | Transmit Shift Register (SCTSR) | 50-9 |
| 50.2.4 | Transmit FIFO Data Register (SCFTDR) | 50-10 |
| 50.2.5 | Serial Mode Register (SCSMR) | 50-11 |

| | | |
|----------|--|-------|
| 50.2.6 | Serial Control Register (SCSCR)..... | 50-13 |
| 50.2.7 | Serial Status Register (SCFSR)..... | 50-17 |
| 50.2.8 | Bit Rate Register (SCBRR)..... | 50-22 |
| 50.2.9 | FIFO Control Register (SCFCR) | 50-23 |
| 50.2.10 | FIFO Data Count Register (SCFDR)..... | 50-25 |
| 50.2.11 | Serial Port Register (SCSPTR) | 50-26 |
| 50.2.12 | Line Status Register (SCLSR) | 50-31 |
| 50.3 | Operation | 50-33 |
| 50.3.1 | Operation in Asynchronous Mode | 50-33 |
| 50.3.2 | Operation in Clock Synchronous Mode..... | 50-41 |
| 50.3.3 | Baud Rate Generator for External Clock (BRG) | 50-49 |
| 50.3.4 | SCIF Interrupt Sources and the DMAC | 50-54 |
| 50.4 | Usage Notes | 50-55 |
| 50.4.1 | Break Detection and Operation..... | 50-55 |
| 50.4.2 | Sending a Break Signal..... | 50-55 |
| 50.4.3 | Data Sampling Timing and Reception Margin in Asynchronous Mode | 50-55 |
| 50.4.4 | Reception Margin and Baud Rate Error..... | 50-56 |
| 51. | High Speed Serial Communication Interface with FIFO (HSCIF)..... | 51-1 |
| 51.1 | Overview..... | 51-1 |
| 51.1.1 | Features..... | 51-1 |
| 51.1.2 | Block Diagram | 51-3 |
| 51.1.3 | External Pins | 51-4 |
| 51.1.4 | Register Configuration..... | 51-5 |
| 51.2 | Register Description..... | 51-6 |
| 51.2.1 | Receive Shift Register (HSRSR) | 51-6 |
| 51.2.2 | Receive FIFO Data Register (HSFRDR) | 51-7 |
| 51.2.3 | Transmit Shift Register (HSTSR)..... | 51-7 |
| 51.2.4 | Transmit FIFO Data Register (HSFTDR)..... | 51-8 |
| 51.2.5 | Serial Mode Register (HSSMR) | 51-9 |
| 51.2.6 | Serial Control Register (HSSCR) | 51-11 |
| 51.2.7 | Serial Status Register (HSFSR) | 51-14 |
| 51.2.8 | Bit Rate Register (HSBRR) | 51-19 |
| 51.2.9 | FIFO Control Register (HSFCR) | 51-20 |
| 51.2.10 | FIFO Data Count Register (HSFDR)..... | 51-21 |
| 51.2.11 | Serial Port Register (HSSPTR)..... | 51-22 |
| 51.2.12 | Line Status Register (HLSR) | 51-24 |
| 51.2.13 | Sampling Rate Register (HSSRR) | 51-26 |
| 51.2.14 | Serial Error Register (HSRER)..... | 51-27 |
| 51.2.15 | RTS Output Active Trigger Register (HSRTGR)..... | 51-28 |
| 51.2.16 | Receive FIFO Data Count Trigger Register (HSRTRGR)..... | 51-28 |
| 51.2.17 | Transmit FIFO Data Count Trigger Register (HSTTRGR) | 51-29 |
| 51.3 | Operation | 51-30 |
| 51.3.1 | Operation in Asynchronous Serial Communication Mode | 51-30 |
| 51.3.2 | HSCIF Interrupt Sources and the DMAC | 51-39 |
| 51.4 | Usage Notes | 51-40 |
| 51.5 | Baud Rate Generator for External Clock (BRG)..... | 51-44 |
| 51.5.1 | Overview..... | 51-44 |
| 51.5.2 | Block Diagram | 51-44 |
| 51.5.3 | BRG Register Configuration..... | 51-45 |
| 51.5.3.1 | Frequency Division Register (DL)..... | 51-45 |
| 51.5.3.2 | Clock Select Register (CKS)..... | 51-47 |
| 51.5.4 | Restrictions in BRG | 51-48 |

| | | |
|---------|--|-------|
| 52. | I2C Bus Interface (I2C) | 52-1 |
| 52.1 | Overview..... | 52-1 |
| 52.1.1 | Features..... | 52-1 |
| 52.1.2 | External Pins..... | 52-3 |
| 52.1.3 | Register Configuration..... | 52-4 |
| 52.2 | Register Description..... | 52-6 |
| 52.2.1 | Slave Control Register (ICSCR)..... | 52-7 |
| 52.2.2 | Slave Status Register (ICSSR)..... | 52-8 |
| 52.2.3 | Slave Interrupt Enable Register (ICSIER)..... | 52-10 |
| 52.2.4 | Slave Address Register (ICSAR)..... | 52-11 |
| 52.2.5 | Master Control Register (ICMCR)..... | 52-12 |
| 52.2.6 | Master Status Register (ICMSR)..... | 52-14 |
| 52.2.7 | Master Interrupt Enable Register (ICMIER)..... | 52-16 |
| 52.2.8 | Master Address Register (ICMAR)..... | 52-17 |
| 52.2.9 | Clock Control Register (ICCCR)..... | 52-18 |
| 52.2.10 | Receive and Transmit Data Registers (ICRXD and ICTXD)..... | 52-20 |
| 52.2.11 | Clock Control Register 2 (ICCCR2)..... | 52-21 |
| 52.2.12 | SCL Mask Control Register (ICMPR)..... | 52-23 |
| 52.2.13 | SCL HIGH Control Register (ICHPR)..... | 52-24 |
| 52.2.14 | SCL LOW Control Register (ICLPR)..... | 52-25 |
| 52.2.15 | DMA Enable Register (ICDMAER)..... | 52-26 |
| 52.2.16 | First Bit Setup Cycle Register (ICFBSCR)..... | 52-28 |
| 52.3 | Operation | 52-29 |
| 52.3.1 | Data and Clock Filters | 52-29 |
| 52.3.2 | Clock Generator..... | 52-29 |
| 52.3.3 | Master/Slave Interfaces..... | 52-29 |
| 52.3.4 | Software Status Interlocking..... | 52-29 |
| 52.3.5 | I2C Bus Data Format | 52-30 |
| 52.3.6 | 7-Bit Address Format..... | 52-31 |
| 52.3.7 | 10-Bit Address Format..... | 52-32 |
| 52.3.8 | Master Transmit Operation | 52-33 |
| 52.3.9 | Master Receive Operation..... | 52-37 |
| 52.4 | Programming Examples..... | 52-39 |
| 52.4.1 | Master Transmitter..... | 52-39 |
| 52.4.2 | Master Receiver | 52-42 |
| 52.4.3 | Master Transmitter—Repeated START—Master Receiver | 52-45 |
| 52.4.4 | Slave Transmitter | 52-49 |
| 52.4.5 | Slave Receiver | 52-51 |
| 52.4.6 | DMA Mode - Master Transmit Operation | 52-53 |
| 52.4.7 | DMA Mode – Master Receive Operation..... | 52-56 |
| 52.4.8 | DMA Mode – Slave Transmit Operation..... | 52-60 |
| 52.4.9 | DMA Mode – Slave Receive Operation | 52-62 |
| 52.5 | Usage Notes | 52-64 |
| 52.5.1 | Usage note for Multi-Master..... | 52-64 |
| 52.5.2 | Usage note for SCL frequency calculation equation..... | 52-64 |
| 52.5.3 | Usage note for Pull-up resistance..... | 52-64 |
| 52.5.4 | Usage note for the transmission and reception procedure..... | 52-65 |
| 52.5.5 | Usage note for DMA mode of Receive Operation..... | 52-65 |
| 53. | IIC Bus Interface for PMIC (IIC for PMIC)..... | 53-1 |
| 53.1 | Overview..... | 53-1 |
| 53.1.1 | Features..... | 53-1 |
| 53.1.2 | External Pins..... | 53-3 |
| 53.1.3 | Register Configuration..... | 53-4 |

| | | |
|---------|---|-------|
| 53.2 | Register Description..... | 53-9 |
| 53.2.1 | IIC Bus Data Register (ICDR)..... | 53-9 |
| 53.2.2 | IIC Bus Control Register (ICCR)..... | 53-10 |
| 53.2.3 | IIC Bus Status Register (ICSR)..... | 53-12 |
| 53.2.4 | IIC Interrupt Control Register (ICIC)..... | 53-15 |
| 53.2.5 | IIC Clock Control Register Low (ICCL)..... | 53-17 |
| 53.2.6 | IIC Clock Control Register High (ICCH)..... | 53-17 |
| 53.2.7 | IIC Automatic Transmission interrupt Mask Register (ICIMSK)..... | 53-18 |
| 53.2.8 | IIC Automatic Transmission Interrupt Flag Register (ICINT)..... | 53-19 |
| 53.2.9 | IIC Automatic Transmission Processing Enabling Register (ICACE)..... | 53-20 |
| 53.2.10 | IIC Automatic Transmission Timer Control Register 1 (ICTMC1)..... | 53-21 |
| 53.2.11 | IIC Automatic Transmission Timer Control Register 2 (ICTMC2)..... | 53-22 |
| 53.2.12 | IIC Automatic Transmission Wait Control Register (ICTMCW)..... | 53-23 |
| 53.2.13 | IIC Automatic Transmission Transmit Start Register (ICSTART)..... | 53-24 |
| 53.2.14 | IIC Automatic Transmission Transmit Control Register (ICATFR)..... | 53-25 |
| 53.2.15 | IIC Automatic Transmission Transmit Time Register 1 (ICATSET1)..... | 53-26 |
| 53.2.16 | IIC Automatic Transmission Transmit Time Register 2 (ICATSET2)..... | 53-27 |
| 53.2.17 | IIC Automatic Transmission Receive Time Register 1 (ICARSET1)..... | 53-28 |
| 53.2.18 | IIC Automatic Transmission Receive Time Register 2 (ICARSET2)..... | 53-29 |
| 53.2.19 | IIC Automatic Transmission Transmit Data 00 to 09 (ICATD00 to ICATD09)..... | 53-29 |
| 53.2.20 | IIC Automatic Transmission Transmit Data 10 to 19 (ICATD10 to ICATD19)..... | 53-29 |
| 53.2.21 | IIC Automatic Transmission Receipt Data 00 to 09 (ICARD00 to ICARD09)..... | 53-30 |
| 53.2.22 | IIC Automatic Transmission Receipt Data 10 to 19 (ICARD10 to ICARD19)..... | 53-30 |
| 53.2.23 | Transfer Rate..... | 53-31 |
| 53.3 | Operation..... | 53-33 |
| 53.3.1 | I2C Bus Data Format..... | 53-33 |
| 53.3.2 | Master Transmit Operation (Using WAIT Function)..... | 53-35 |
| 53.3.3 | Master Receive Operation (Using WAIT Function)..... | 53-38 |
| 53.3.4 | IIC Continuous Transmission by DMA..... | 53-43 |
| 53.3.5 | IIC Continuous Receive by DMA..... | 53-45 |
| 53.3.6 | Automatic transfer mode..... | 53-47 |
| 53.3.7 | Automatic transfer mode by DMA..... | 53-49 |
| 53.3.8 | Synchronizing SCL Line..... | 53-51 |
| 53.3.9 | Noise Canceller..... | 53-52 |
| 53.3.10 | Arbitration Lost Operation..... | 53-53 |
| 53.3.11 | Non-Acknowledge Operation..... | 53-55 |
| 53.3.12 | Wait Operation..... | 53-57 |
| 53.4 | Usage Notes..... | 53-58 |
| 54. | Clock-Synchronized Serial Interface with FIFO (MSIOF)..... | 54-1 |
| 54.1 | Overview..... | 54-1 |
| 54.1.1 | Features..... | 54-1 |
| 54.1.2 | Block diagram..... | 54-2 |
| 54.1.3 | External Pins..... | 54-3 |
| 54.1.4 | Register Configuration..... | 54-5 |
| 54.1.5 | Connected Module..... | 54-6 |
| 54.2 | Register Description..... | 54-7 |
| 54.2.1 | MSIOF Transmit Mode Register 1 (SITMDR1)..... | 54-8 |
| 54.2.2 | MSIOF Transmit Mode Register 2 (SITMDR2)..... | 54-11 |
| 54.2.3 | MSIOF Transmit Mode Register 3 (SITMDR3)..... | 54-12 |
| 54.2.4 | MSIOF Receive Mode Register 1 (SIRMDR1)..... | 54-13 |
| 54.2.5 | MSIOF Receive Mode Register 2 (SIRMDR2)..... | 54-15 |
| 54.2.6 | MSIOF Receive Mode Register 3 (SIRMDR3)..... | 54-16 |
| 54.2.7 | MSIOF Control Register (SICTR)..... | 54-17 |

| | | |
|---------|---|-------|
| 54.2.8 | MSIOF Transmit Clock Select Register (SITSCR) | 54-21 |
| 54.2.9 | MSIOF Transmit FIFO Data Register (SITFDR) | 54-22 |
| 54.2.10 | MSIOF Receive FIFO Data Register (SIRFDR)..... | 54-23 |
| 54.2.11 | MSIOF Status Register (SISTR)..... | 54-24 |
| 54.2.12 | MSIOF Interrupt Enable Register (SIIER) | 54-28 |
| 54.2.13 | MSIOF FIFO Control Register (SIFCTR) | 54-30 |
| 54.3 | Operation | 54-32 |
| 54.3.1 | Operating Mode | 54-32 |
| 54.3.2 | Serial Clocks | 54-32 |
| 54.3.3 | Serial Timing | 54-33 |
| 54.3.4 | Transfer Data Allocation to Registers..... | 54-35 |
| 54.3.5 | FIFO..... | 54-36 |
| 54.3.6 | Transmit and Receive Procedures..... | 54-38 |
| 54.3.7 | Interrupts..... | 54-45 |
| 54.3.8 | Transmit and Receive Timing..... | 54-46 |
| 54.3.9 | SPI..... | 54-48 |
| 54.3.10 | IIS | 54-50 |
| 54.4 | Usage Notes | 54-51 |
| 54.4.1 | Notes for Using SPI Mode..... | 54-51 |
| 54.4.2 | Notes for Using SPI slave Mode..... | 54-51 |
| 54.4.3 | Notes for Transmitting Audio data via MSIOF | 54-52 |
| 54.4.4 | Notes on Fixed Value related to MSIOF registers | 54-52 |
| 55. | PWM..... | 55-1 |
| 55.1 | Overview..... | 55-1 |
| 55.1.1 | Features..... | 55-1 |
| 55.1.2 | Block Diagram..... | 55-2 |
| 55.1.3 | External Pins | 55-3 |
| 55.1.4 | Register Configuration..... | 55-4 |
| 55.2 | Register Description..... | 55-5 |
| 55.2.1 | PWM Control Register (PWMCR) | 55-6 |
| 55.2.2 | PWM Count Register (PWMCNT)..... | 55-8 |
| 55.3 | Operation | 55-9 |
| 55.4 | Usage Notes | 55-11 |
| 56. | SPI Multi I/O Bus Controller (RPC-IF)..... | 56-1 |
| 56.1 | Overview..... | 56-1 |
| 56.1.1 | Features..... | 56-1 |
| 56.1.2 | Block Diagram..... | 56-2 |
| 56.1.3 | External Pins | 56-3 |
| 56.1.4 | Register Configuration..... | 56-4 |
| 56.2 | Register Description..... | 56-7 |
| 56.2.1 | Common Control Register (CMNCR) | 56-7 |
| 56.2.2 | SSL Delay Register (SSLDR)..... | 56-10 |
| 56.2.3 | Data Read Control Register (DRCR)..... | 56-12 |
| 56.2.4 | Data Read Command Setting Register (DRCMR)..... | 56-14 |
| 56.2.5 | Data Read Extended Address Setting Register (DREAR) | 56-15 |
| 56.2.6 | Data Read Option Setting Register (DROPR) | 56-16 |
| 56.2.7 | Data Read Enable Setting Register (DRENr)..... | 56-17 |
| 56.2.8 | Manual Mode Control Register (SMCR)..... | 56-20 |
| 56.2.9 | Manual Mode Command Setting Register (SMCMR)..... | 56-22 |
| 56.2.10 | Manual Mode Address Setting Register (SMADR)..... | 56-23 |
| 56.2.11 | Manual Mode Option Setting Register (SMOPR) | 56-24 |
| 56.2.12 | Manual Mode Enable Setting Register (SMENR) | 56-25 |

| | | |
|---------|--|--------|
| 56.2.13 | Manual Mode Read Data Register 0 (SMRDR0)..... | 56-29 |
| 56.2.14 | Manual Mode Read Data Register 1 (SMRDR1)..... | 56-30 |
| 56.2.15 | Manual Mode Write Data Register 0 (SMWDR0)..... | 56-31 |
| 56.2.16 | Manual Mode Write Data Register 1 (SMWDR1)..... | 56-32 |
| 56.2.17 | Common Status Register (CMNSR)..... | 56-33 |
| 56.2.18 | Data Read Dummy Cycle Setting Register (DRDMCR)..... | 56-34 |
| 56.2.19 | Data Read DDR Enable Register (DRDRENR)..... | 56-35 |
| 56.2.20 | Manual Mode Dummy Cycle Setting Register (SMDMCR)..... | 56-37 |
| 56.2.21 | Manual Mode DDR Enable Register (SMDRENR)..... | 56-38 |
| 56.2.22 | PHY Control Register (PHYCNT)..... | 56-40 |
| 56.2.23 | PHY Timing Offset Register 1 (PHYOFFSET1)..... | 56-43 |
| 56.2.24 | Timing Offset Register 2 (PHYOFFSET2)..... | 56-44 |
| 56.2.25 | PHY Interrupt Register (PHYINT)..... | 56-45 |
| 56.2.26 | Address Division Register 1 (ADD_DIV1)..... | 56-47 |
| 56.2.27 | Address Division Register 2 (ADD_DIV2)..... | 56-48 |
| 56.2.28 | Address Division Register 3 (ADD_DIV3)..... | 56-49 |
| 56.2.29 | Secure Configuration Register (SEC_CONF)..... | 56-50 |
| 56.2.30 | Access Right Register (ARIGHT)..... | 56-52 |
| 56.2.31 | Secure Command Register (SEC_CMDn (n = 0 to 15))..... | 56-53 |
| 56.2.32 | Non-Secure Command Register (NON_SEC_CMDn (n = 0 to 15))..... | 56-54 |
| 56.2.33 | Erase Command List Register (ERASELISTn (n = 1, 2))..... | 56-55 |
| 56.2.34 | Write Command List Register (WRITELISTn (n = 1, 2))..... | 56-56 |
| 56.3 | Operation..... | 56-57 |
| 56.3.1 | System Configuration..... | 56-57 |
| 56.3.2 | Address Map..... | 56-61 |
| 56.3.3 | 32-Bit Address Area in the Serial Flash Memory..... | 56-62 |
| 56.3.4 | Operating Modes..... | 56-63 |
| 56.3.5 | External Address Space Read Mode..... | 56-64 |
| 56.3.6 | Read Cache..... | 56-69 |
| 56.3.7 | Manual Mode..... | 56-70 |
| 56.3.8 | Command Sequence..... | 56-73 |
| 56.3.9 | Data Pin Control..... | 56-81 |
| 56.3.10 | QSPIn_SSL Pin Control..... | 56-83 |
| 56.3.11 | QSPIn_SPCLK Pin Control..... | 56-83 |
| 56.3.12 | Flags..... | 56-84 |
| 56.3.13 | Write Buffer Operation..... | 56-84 |
| 56.3.14 | Data Alignment with Octal-SPI flash memory..... | 56-86 |
| 56.3.15 | Supported Serial Flash Protocol..... | 56-87 |
| 56.3.16 | Supported Octal-SPI Flash Protocol..... | 56-87 |
| 56.3.17 | DDR operation..... | 56-88 |
| 56.3.18 | Security function..... | 56-89 |
| 56.3.19 | PHY register interface..... | 56-91 |
| 56.4 | Usage Notes..... | 56-104 |
| 56.4.1 | Transfer to Read Data while the Signal on the QSPIn_SSL Pin is Negated..... | 56-104 |
| 56.4.2 | Transfer to Read Data while the Signal on the QSPIn_SSL Pin is Asserted..... | 56-104 |
| 56.4.3 | Notes on Starting Transfer from the QSPIn_SSL Retained State in Manual Mode..... | 56-104 |
| 56.4.4 | RPC_RESET# operation..... | 56-104 |
| 56.4.5 | Software Reset after Module Stop operation..... | 56-105 |
| 56.4.6 | DMA transfer..... | 56-105 |
| 56.4.7 | Frequency change..... | 56-105 |
| 56.4.8 | QSPI1 pins switched to high impedance [RZ/G2M V1.3] (not RZ/G2M V3.0)..... | 56-105 |
| 57. | SD Card/MMC Interfaces..... | 57-1 |
| 57.1 | Overview..... | 57-1 |

| | | |
|---------|--|-------|
| 57.1.1 | Features..... | 57-1 |
| 57.1.2 | Block Diagram..... | 57-2 |
| 57.1.3 | External Pins..... | 57-3 |
| 57.1.4 | Register Configuration..... | 57-5 |
| 57.2 | Usage Notes..... | 57-6 |
| 58. | rawNAND Controller (R-NANDC)..... | 58-1 |
| 58.1 | Overview..... | 58-1 |
| 58.1.1 | Features..... | 58-1 |
| 58.1.2 | Block Diagram..... | 58-3 |
| 58.1.3 | External Pins..... | 58-5 |
| 58.1.4 | Register Configuration..... | 58-6 |
| 58.1.5 | Connected Module..... | 58-8 |
| 58.2 | Register Description..... | 58-9 |
| 58.2.1 | Controller commands register (COMMAND)..... | 58-9 |
| 58.2.2 | Main configurations register (CONTROL)..... | 58-10 |
| 58.2.3 | GENERIC_SEQ register (GEN_SEQ_CTRL)..... | 58-13 |
| 58.2.4 | Controller status register (STATUS)..... | 58-15 |
| 58.2.5 | LUN per device status register (LUN_STATUS_0)..... | 58-17 |
| 58.2.6 | Interrupts mask register (INT_MASK)..... | 58-18 |
| 58.2.7 | Interrupts status register (INT_STATUS)..... | 58-20 |
| 58.2.8 | ECC module control register (ECC_CTRL)..... | 58-22 |
| 58.2.9 | ECC module status register (ECC_STAT)..... | 58-23 |
| 58.2.10 | ECC offset in the spare area register (ECC_OFFSET)..... | 58-24 |
| 58.2.11 | ECC error level counter register (ECC_CNT)..... | 58-25 |
| 58.2.12 | Column/row address registers (ADDR[1:0]_COL, ADDR[1:0]_ROW)..... | 58-26 |
| 58.2.13 | Page size value register (DATA_SIZE)..... | 58-28 |
| 58.2.14 | FIFO module interface register (FIFO_DATA)..... | 58-30 |
| 58.2.15 | Bad block management (BBM) module control register (BBM_CTRL)..... | 58-31 |
| 58.2.16 | Records table pointer register (DEV0_PTR)..... | 58-32 |
| 58.2.17 | Records table size register (DEV0_SIZE)..... | 58-33 |
| 58.2.18 | DMA base address - least significant part register (DMA_ADDR_L)..... | 58-34 |
| 58.2.19 | DMA counter initial value register (DMA_CNT)..... | 58-35 |
| 58.2.20 | DMA control register (DMA_CTRL)..... | 58-36 |
| 58.2.21 | DMA trigger level value register (DMA_TRIG_TLVL)..... | 58-37 |
| 58.2.22 | Mask register for the READ STATUS commands (STATUS_MASK)..... | 58-38 |
| 58.2.23 | Command sequence timing configuration register 0 (TIME_SEQ_0)..... | 58-39 |
| 58.2.24 | Command sequence timing configuration register 1 (TIME_SEQ_1)..... | 58-40 |
| 58.2.25 | Generic command sequence timing configuration register 0 (TIME_GEN_SEQ_0)..... | 58-41 |
| 58.2.26 | Generic command sequence timing configuration register 1 (TIME_GEN_SEQ_1)..... | 58-43 |
| 58.2.27 | Generic command sequence timing configuration register 2 (TIME_GEN_SEQ_2)..... | 58-45 |
| 58.2.28 | Generic command sequence timing configuration register 3 (TIME_GEN_SEQ_3)..... | 58-47 |
| 58.2.29 | Timing configuration register (TIMINGS_ASYN)..... | 58-48 |
| 58.2.30 | Data register (DATA_REG)..... | 58-49 |
| 58.2.31 | Data register size selection register (DATA_REG_SIZE)..... | 58-50 |
| 58.2.32 | FIFO module control register (FIFO_INIT)..... | 58-51 |
| 58.2.33 | FIFO module status register (FIFO_STATE)..... | 58-52 |
| 58.2.34 | MLUN register (MLUN)..... | 58-53 |
| 58.2.35 | CMD ID initial value register (CMD_MARK)..... | 58-54 |
| 58.2.36 | Memory device control register (MEM_CTRL)..... | 58-55 |
| 58.3 | Operation..... | 58-56 |
| 58.3.1 | Command Generation..... | 58-56 |
| 58.3.2 | Generic Command Sequence..... | 58-67 |
| 58.3.3 | Instructions..... | 58-70 |

| | | |
|------------|---|-------------|
| 58.3.4 | Multi LUN Work Mode | 58-90 |
| 58.3.5 | Remapping Mechanism | 58-90 |
| 58.3.6 | Interrupts Mechanism | 58-92 |
| 58.3.7 | Setup and Configuration | 58-94 |
| 58.4 | Functional Details | 58-104 |
| 58.4.1 | Block Diagram | 58-104 |
| 58.4.2 | DMA Module..... | 58-105 |
| 58.4.3 | ECC Module | 58-111 |
| 58.4.4 | BCH Algorithm Implementation | 58-112 |
| 58.5 | Usage Notes | 58-113 |
| 58.5.1 | Initial Setting..... | 58-113 |
| 58.5.2 | Clock Stop..... | 58-114 |
| 59. | Serial-ATA Gen3 | 59-1 |
| 59.1 | Overview..... | 59-1 |
| 59.1.1 | Features..... | 59-1 |
| 59.1.2 | Block Diagram..... | 59-1 |
| 59.1.3 | External Pins | 59-2 |
| 59.1.4 | Register Configuration..... | 59-3 |
| 59.1.5 | Connected Module..... | 59-5 |
| 59.2 | Register Description..... | 59-6 |
| 59.2.1 | Note on the Access to the Registers Allocated to H'1018 to H'10A8 | 59-6 |
| 59.2.2 | ATAPI Control Register (ATAPI_CONTROL1) | 59-7 |
| 59.2.3 | ATAPI Status Register (ATAPI_STATUS) | 59-9 |
| 59.2.4 | Interrupt Enable Register (ATAPI_INT_ENABLE)..... | 59-11 |
| 59.2.5 | Descriptor Table Base Address Register (ATAPI_DTB_ADR)..... | 59-12 |
| 59.2.6 | DMA Start Address Register (ATAPI_DMA_START_ADR)..... | 59-15 |
| 59.2.7 | DMA Transfer Count Register (ATAPI_DMA_TRANS_CNT) | 59-16 |
| 59.2.8 | ATAPI Control 2 Register (ATAPI_CONTROL2) | 59-17 |
| 59.2.9 | ATAPI Signal Status Register (ATAPI_SIG_ST) | 59-18 |
| 59.2.10 | Byte Swap Register (ATAPI_BYTE_SWAP)..... | 59-19 |
| 59.2.11 | BIST Config Register (BISTCONF)..... | 59-20 |
| 59.2.12 | Shadow Data Register (DATA) (SDATA) | 59-21 |
| 59.2.13 | Shadow Error Register (SERR[Read Mode]) (SSERR[Read Mode]) | 59-22 |
| 59.2.14 | Shadow Features Register (SFEATURES[Write Mode]) (SSFEATURES[Write Mode])..... | 59-23 |
| 59.2.15 | Shadow Sector CNT Register (SECCNT) (SSECCNT) | 59-24 |
| 59.2.16 | Shadow LBA Low Register (LBALOW) (SLBALOW)..... | 59-25 |
| 59.2.17 | Shadow LBA Mid Register (LBAMID) (SLBAMID)..... | 59-26 |
| 59.2.18 | Shadow LBA High Register (LBAHIGH) (SLBAHIGH) | 59-27 |
| 59.2.19 | Shadow Device/Head Register (DEVHEAD) (SDEVHEAD)..... | 59-28 |
| 59.2.20 | Shadow Status Register (SSTATUS [Read Mode]) (SSSTATUS [Read Mode])..... | 59-29 |
| 59.2.21 | Shadow Command Register (SCOM [Write Mode]) (SSCOM [Write Mode])..... | 59-30 |
| 59.2.22 | Shadow alternate status register (SALTSTS [Read Mode]) (SSALTSTS [Read Mode])..... | 59-31 |
| 59.2.23 | Shadow Device Control Register (SDEVCON [Write Mode]) (SSDEVCON [Write Mode]) | 59-32 |
| 59.2.24 | SATA Extend ICC Register (SATAEICCR) | 59-33 |
| 59.2.25 | SATA Extend Auxiliary Register (SATAEAUXR)..... | 59-33 |
| 59.2.26 | SATA Extend DEVSLP Register (SATAEDEVSLPR) | 59-34 |
| 59.2.27 | SCR SStatus Register (SCRSSSTS)..... | 59-35 |
| 59.2.28 | SCR SError Register (SCRSEERR) | 59-36 |
| 59.2.29 | SCR SControl Register (SCRSCON)..... | 59-38 |
| 59.2.30 | SCR SActive Register (SCRSACT)..... | 59-40 |
| 59.2.31 | SATA INT Status Register (SATAINTSTAT)..... | 59-41 |
| 59.2.32 | SATA INT Mask Register (SATAINTMASK) | 59-43 |
| 59.2.33 | PHY STOP Register (PHYSTOP) | 59-44 |

| | | |
|-----------|--|-------|
| 59.2.34 | Rx DMA Setup FIS Dword0 Register (DMADW0) | 59-45 |
| 59.2.35 | Rx DMA Setup FIS Dword1 Register (DMADW1) | 59-46 |
| 59.2.36 | Rx DMA Setup FIS Dword2 Register (DMADW2) | 59-46 |
| 59.2.37 | Rx DMA Setup FIS Dword3 Register (DMADW3) | 59-47 |
| 59.2.38 | Rx DMA Setup FIS Dword4 Register (DMADW4) | 59-47 |
| 59.2.39 | Rx DMA Setup FIS Dword5 Register (DMADW5) | 59-48 |
| 59.2.40 | Rx DMA Setup FIS Dword6 Register (DMADW6) | 59-48 |
| 59.3 | Operation | 59-49 |
| 59.3.1 | Power Management Mode | 59-49 |
| 59.3.2 | Device Sleep | 59-49 |
| 59.3.3 | Initialization | 59-50 |
| 59.3.4 | Interrupt Modes | 59-51 |
| 59.3.5 | Interrupt Processing Flow | 59-53 |
| 59.3.6 | Endian | 59-58 |
| 59.3.7 | SATA BIST Modes | 59-59 |
| 59.3.8 | Operation Flow | 59-61 |
| 59.3.9 | Resets | 59-63 |
| 59.3.10 | Extended ATA Registers | 59-64 |
| 59.3.11 | Signature of the Shadow Register (INFORMATIVE) | 59-64 |
| 59.3.12 | Reference Clock Source | 59-65 |
| 59.4 | Usage Notes | 59-66 |
| 59.4.1 | Host Termination | 59-66 |
| 59.4.2 | Clock Input in the Partial and Slumber States | 59-66 |
| 60. | USB2.0 Host (EHCI/OHCI) | 60-1 |
| 60.1 | Overview | 60-1 |
| 60.1.1 | Features | 60-1 |
| 60.1.1.1 | OTG Function | 60-2 |
| 60.1.1.2 | Battery Charging Function | 60-2 |
| 60.1.1.3 | Core Setting | 60-2 |
| 60.1.1.4 | Host/Peripheral Switching Specifications | 60-2 |
| 60.1.2 | External Pins | 60-3 |
| 60.1.3 | Register Configuration | 60-4 |
| 60.1.3.1 | Register Attributes | 60-4 |
| 60.1.3.2 | Address Mappings | 60-4 |
| 60.1.3.3 | Register Configuration | 60-6 |
| 60.1.4 | Connected Module | 60-8 |
| 60.2 | Register Description | 60-9 |
| 60.2.1 | OHCI Operational Registers | 60-9 |
| 60.2.1.1 | HcRevision Register (offset: H'000) | 60-9 |
| 60.2.1.2 | HcControl Register (offset: H'004) | 60-10 |
| 60.2.1.3 | HcCommandStatus Register (offset: H'008) | 60-13 |
| 60.2.1.4 | HcInterruptStatus Register (offset: H'00C) | 60-15 |
| 60.2.1.5 | HcInterruptEnable Register (offset: H'010) | 60-17 |
| 60.2.1.6 | HcInterruptDisable Register (offset: H'014) | 60-20 |
| 60.2.1.7 | HcHCCA Register (offset: H'018) | 60-22 |
| 60.2.1.8 | HcPeriodCurrentED Register (offset: H'01C) | 60-23 |
| 60.2.1.9 | HcControlHeadED Register (offset: H'020) | 60-24 |
| 60.2.1.10 | HcControlCurrentED Register (offset: H'024) | 60-25 |
| 60.2.1.11 | HcBulkHeadED Register (offset: H'028) | 60-26 |
| 60.2.1.12 | HcBulkCurrentED Register (offset: H'02C) | 60-27 |
| 60.2.1.13 | HcDoneHead Register (offset: H'030) | 60-28 |
| 60.2.1.14 | HcFmInterval Register (offset: H'034) | 60-29 |
| 60.2.1.15 | HcFmRemaining Register (offset: H'038) | 60-30 |

| | | |
|-----------|---|-------|
| 60.2.1.16 | HcFmNumber Register (offset: H'03C) | 60-31 |
| 60.2.1.17 | HcPeriodicStart Register (offset: H'040)..... | 60-32 |
| 60.2.1.18 | HcLSThreshold Register (offset: H'044)..... | 60-33 |
| 60.2.1.19 | HcRhDescriptorA Register (offset: H'048)..... | 60-34 |
| 60.2.1.20 | HcRhDescriptorB Register (offset: H'04C)..... | 60-36 |
| 60.2.1.21 | HcRhStatus Register (offset: H'050)..... | 60-37 |
| 60.2.1.22 | HcRhPortStatus[1] Register (offset: H'054)..... | 60-40 |
| 60.2.2 | EHCI Controller Capability Registers | 60-44 |
| 60.2.2.1 | HCVERSION/CAPLENGTH Register (offset: H'100) | 60-44 |
| 60.2.2.2 | HCSPARAMS Register (offset: H'104)..... | 60-45 |
| 60.2.2.3 | HCCPARAMS Register (offset: H'108)..... | 60-46 |
| 60.2.2.4 | HCSP-PORTROUTE Register (offset: H'10C)..... | 60-48 |
| 60.2.3 | EHCI Operational Registers..... | 60-49 |
| 60.2.3.1 | USBCMD Register (offset: H'120) | 60-49 |
| 60.2.3.2 | USBSTS Register (offset: H'124) | 60-52 |
| 60.2.3.3 | USBINTR Register (offset: H'128)..... | 60-55 |
| 60.2.3.4 | FRINDEX Register (offset: H'12C) | 60-57 |
| 60.2.3.5 | CTRLDSSEGMENT Register (offset: H'130)..... | 60-58 |
| 60.2.3.6 | PERIODICLISTBASE Register (offset: H'134) | 60-59 |
| 60.2.3.7 | ASYNCLISTADDR Register (offset: H'138)..... | 60-60 |
| 60.2.3.8 | CONFIGFLAG Register (offset: H'160)..... | 60-61 |
| 60.2.3.9 | PORTSC[1] Register (offset: H'164) | 60-62 |
| 60.2.4 | AHB Bridge Registers | 60-68 |
| 60.2.4.1 | INT_ENABLE Register (offset: H'200)..... | 60-68 |
| 60.2.4.2 | INT_STATUS Register (offset: H'204) | 60-69 |
| 60.2.4.3 | AHB_BUS_CTR Register (offset: H'208) | 60-70 |
| 60.2.4.4 | USBCTR Register (offset: H'20C)..... | 60-72 |
| 60.2.5 | Core Defined Registers | 60-73 |
| 60.2.5.1 | REVID Register (offset: H'300)..... | 60-73 |
| 60.2.5.2 | Register Enable/Clock Gating Control Register (offset: H'304) | 60-74 |
| 60.2.5.3 | Suspend Control Register (offset: H'308) | 60-75 |
| 60.2.5.4 | Suspend/Resume Timer Setting Register (offset: H'30C) | 60-77 |
| 60.2.5.5 | Overcurrent Detection Timer Setting Register (offset: H'310) | 60-78 |
| 60.2.5.6 | SBRN, FLADJ, PORTWAKECAP Register (offset: H'314) | 60-79 |
| 60.2.6 | UCOM Registers..... | 60-80 |
| 60.2.6.1 | Common Control Register (offset: H'800) | 60-80 |
| 60.2.6.2 | OTG-BC Interrupt Status Register (offset: H'804)..... | 60-81 |
| 60.2.6.3 | OTG-BC Interrupt Enable Register (offset: H'808) | 60-83 |
| 60.2.6.4 | VBUS Control Register (offset: H'80C)..... | 60-85 |
| 60.2.6.5 | Line Control Port1 Register (offset: H'810) | 60-87 |
| 60.2.6.6 | BC Control Port1 Register (offset: H'820)..... | 60-89 |
| 60.2.6.7 | ADP Control Register (offset: H'830)..... | 60-91 |
| 60.3 | Clock Specifications | 60-93 |
| 60.3.1 | Clock Gating Specifications..... | 60-93 |
| 60.3.1.1 | Overview | 60-93 |
| 60.3.1.2 | NONUSE_CLK_MSK Function Specifications | 60-93 |
| 60.3.1.3 | PERI_CLK_MSK and HOST_CLK_MSK Function Specifications | 60-94 |
| 60.4 | Reset Specifications | 60-95 |
| 60.4.1 | Input Reset Specifications..... | 60-95 |
| 60.5 | Interrupt Specifications | 60-96 |
| 60.5.1 | Interrupt Signal List | 60-96 |
| 60.5.2 | Interrupt Sources and Control..... | 60-97 |
| 60.5.2.1 | U2H_INT Assertion Source and Control | 60-97 |
| 60.5.2.2 | U2H_OHCI_INT Assertion Sources and Control | 60-97 |

| | | |
|----------|--|--------|
| 60.5.2.3 | U2H_EHCI_INT Assertion Sources and Control | 60-99 |
| 60.5.2.4 | U2H_WAKEON_INT Assertion Sources and Control | 60-100 |
| 60.5.2.5 | U2H_OBINT Assertion Sources and Control | 60-101 |
| 60.5.3 | Interrupt Signal Deassertion Timing | 60-103 |
| 60.5.3.1 | Connection of Interrupt Signal Lines | 60-103 |
| 60.6 | Power Management | 60-104 |
| 60.6.1 | Controlling SUSPENDM and SLEEPM Pins of UTMI+ PHY | 60-104 |
| 60.6.2 | Controlling Clock Gating Function..... | 60-104 |
| 60.7 | OTG Control | 60-105 |
| 60.7.1 | Conditions for Using This Module for OTG..... | 60-105 |
| 60.7.1.1 | Core Pin and define Item Settings..... | 60-105 |
| 60.7.1.2 | ID Pin Connections | 60-105 |
| 60.7.1.3 | Port Power (VBUS) Control and Connection with External Power-Supply IC | 60-105 |
| 60.7.1.4 | OTG System Configuration Example | 60-106 |
| 60.7.2 | Initial Setting Procedure | 60-107 |
| 60.7.2.1 | OTG Initial Setting Procedure..... | 60-107 |
| 60.7.3 | A-Device/B-Device Recognition Procedure | 60-108 |
| 60.7.3.1 | A-Device/B-Device Recognition Procedure | 60-108 |
| 60.7.4 | Disconnect Detection Procedure | 60-109 |
| 60.7.4.1 | Disconnect Detection Procedure | 60-109 |
| 60.7.5 | HNP Control Procedure (Conforming to OTG Revision 2.0)..... | 60-110 |
| 60.7.5.1 | HNP Control Procedure for A-Device (Switching from A-Host to A-Peripheral)..... | 60-110 |
| 60.7.5.2 | HNP Control Procedure for A-Device (Switching from A-Peripheral to A-Host)..... | 60-112 |
| 60.7.5.3 | HNP Control Procedure for B-Device (Switching from B-Peripheral to B-Host) | 60-113 |
| 60.7.5.4 | HNP Control Procedure for B-Device (Switching from B-Host to B-Peripheral) | 60-114 |
| 60.7.6 | SRP Control Procedure (Conforming to OTG Revision 2.0)..... | 60-116 |
| 60.7.6.1 | SRP Control Procedure for A-Device | 60-116 |
| 60.7.6.2 | SRP Control Procedure for B-Device | 60-117 |
| 60.8 | Battery Charging Control..... | 60-118 |
| 60.8.1 | Conditions for Using Battery Charging Function in this Module..... | 60-118 |
| 60.8.1.1 | Limitations on Support of Charging Ports | 60-118 |
| 60.8.1.2 | Portable Device Control..... | 60-118 |
| 60.8.2 | Controlling Method..... | 60-119 |
| 60.8.2.1 | How to Control..... | 60-119 |
| 60.9 | Operation | 60-128 |
| 60.9.1 | Power control and Initialization..... | 60-128 |
| 60.9.2 | Transfer Overview | 60-128 |
| 61. | USB High-Speed Module (HS-USB) | 61-1 |
| 61.1 | Overview..... | 61-1 |
| 61.1.1 | Features | 61-1 |
| 61.1.2 | External Pins | 61-3 |
| 61.1.3 | Register Configuration..... | 61-4 |
| 61.1.4 | Connected Module..... | 61-9 |
| 61.2 | Register Description..... | 61-10 |
| 61.2.1 | System Configuration Control Register (SYSCFG) | 61-11 |
| 61.2.2 | CPU Bus Wait Register (BUSWAIT)..... | 61-13 |
| 61.2.3 | System Configuration Status Register (SYSSTS)..... | 61-14 |
| 61.2.4 | Device State Control Register (DVSTCTR) | 61-15 |
| 61.2.5 | Test Mode Register (TESTMODE) | 61-16 |
| 61.2.6 | CFIFO Port Register (CFIFO) | 61-17 |
| 61.2.7 | FIFO Port Select Registers (CFIFOSEL, D0FIFOSEL, D1FIFOSEL, D2FIFOSEL, D3FIFOSEL)..... | 61-18 |

| | | |
|---------|---|--------|
| 61.2.8 | FIFO Port Control Registers (CFIFOCTR, D0FIFOCTR, D1FIFOCTR, D2FIFOCTR, D3FIFOCTR)..... | 61-24 |
| 61.2.9 | Interrupt Enable Register 0 (INTENB0)..... | 61-26 |
| 61.2.10 | BRDY Interrupt Enable Register (BRDYENB)..... | 61-28 |
| 61.2.11 | NRDY Interrupt Enable Register (NRDYENB)..... | 61-30 |
| 61.2.12 | BEMP Interrupt Enable Register (BEMPENB)..... | 61-32 |
| 61.2.13 | SOF Output Configuration Register (SOFCFG)..... | 61-34 |
| 61.2.14 | Interrupt Status Register 0 (INTSTS0)..... | 61-35 |
| 61.2.15 | BRDY Interrupt Status Register (BRDYSTS)..... | 61-38 |
| 61.2.16 | NRDY Interrupt Status Register (NRDYSTS)..... | 61-40 |
| 61.2.17 | BEMP Interrupt Status Register (BEMPSTS)..... | 61-42 |
| 61.2.18 | Frame Number Register (FRMNUM)..... | 61-44 |
| 61.2.19 | μ Frame Number Register (UFRMNUM)..... | 61-45 |
| 61.2.20 | USB Address Register (USBADDR)..... | 61-46 |
| 61.2.21 | USB Request Type Register (USBREQ)..... | 61-47 |
| 61.2.22 | USB Request Value Register (USBVAL)..... | 61-47 |
| 61.2.23 | USB Request Index Register (USBINDEX)..... | 61-48 |
| 61.2.24 | USB Request Length Register (USBLENG)..... | 61-48 |
| 61.2.25 | DCP Maximum Packet Size Register (DCPMAXP)..... | 61-49 |
| 61.2.26 | DCP Control Register (DCPCTR)..... | 61-50 |
| 61.2.27 | Pipe Window Select Register (PIPESEL)..... | 61-53 |
| 61.2.28 | Pipe Configuration Register (PIPECFG)..... | 61-54 |
| 61.2.29 | Pipe Buffer Register (PIPEBUF)..... | 61-58 |
| 61.2.30 | Pipe Maximum Packet Size Register (PIPEMAXP)..... | 61-60 |
| 61.2.31 | Pipe Cycle Control Register (PIPEPERI)..... | 61-61 |
| 61.2.32 | Pipe n Control Register (PIPEnCTR) (n = 1 to F)..... | 61-62 |
| 61.2.33 | Pipe n Transaction Counter Enable Register (PIPEnTRE) (n = 1 to 5, 9 to F)..... | 61-70 |
| 61.2.34 | Pipe n Transaction Counter Register (PIPEnTRN) (n = 1 to 5, 9 to F)..... | 61-71 |
| 61.2.35 | Low Power Status register (LPSTS)..... | 61-72 |
| 61.2.36 | Battery Charging Control Register (BCCTRL)..... | 61-73 |
| 61.2.37 | USB General Control Register (UGCTRL)..... | 61-75 |
| 61.2.38 | USB General Control Register 2 (UGCTRL2)..... | 61-76 |
| 61.2.39 | USB General Status Register (UGSTS)..... | 61-77 |
| 61.2.40 | USB XTAL/EXTTAL control register (USB20_CLKSET0)..... | 61-78 |
| 61.3 | Operation..... | 61-79 |
| 61.3.1 | System Control and Oscillation Control..... | 61-79 |
| 61.3.2 | Interrupt Function..... | 61-82 |
| 61.3.3 | Pipe Control..... | 61-92 |
| 61.3.4 | FIFO Buffer Memory..... | 61-98 |
| 61.3.5 | Control Transfer (DCP)..... | 61-105 |
| 61.3.6 | Bulk Transfer (Pipes 1 to 5, 9 to F)..... | 61-106 |
| 61.3.7 | Interrupt Transfer (Pipes 6 to 8)..... | 61-107 |
| 61.3.8 | Isochronous Transfer (Pipes 1, 2)..... | 61-107 |
| 61.3.9 | SOF Interpolation Function..... | 61-113 |
| 61.4 | Guidelines for Designing USB2.0 Hi-Speed Boards..... | 61-114 |
| 61.4.1 | USB Transmission Line..... | 61-114 |
| 61.4.2 | Power Supply and Ground Pattern..... | 61-114 |
| 61.4.3 | Oscillation..... | 61-114 |
| 61.4.4 | VBUS..... | 61-115 |
| 61.4.5 | TXRTUNE _n /USB_RREF Pin..... | 61-115 |
| 61.5 | Battery Charging Control..... | 61-116 |
| 62. | USB High-Speed DMAC (USB-DMAC)..... | 62-1 |
| 62.1 | Overview..... | 62-1 |

| | | |
|-----------|---|-------|
| 62.1.1 | Features..... | 62-1 |
| 62.1.2 | Block Diagram..... | 62-3 |
| 62.1.3 | Register Configuration..... | 62-4 |
| 62.2 | Register Description..... | 62-6 |
| 62.2.1 | DMA _n VCR Register (USBDMAN_VCR)..... | 62-6 |
| 62.2.2 | DMA _n Software Reset Register (USBDMAN_SWR)..... | 62-7 |
| 62.2.3 | DMA _n Interrupt Source Register (USBDMAN_DMICR)..... | 62-8 |
| 62.2.4 | DMA _n Source Address Registers 0 and 1 (USBDMAN_SAR_0/1)..... | 62-10 |
| 62.2.5 | DMA _n Destination Address Registers 0 and 1 (USBDMAN_DAR_0/1)..... | 62-10 |
| 62.2.6 | DMA _n Transfer Count Registers 0 and 1 (USBDMAN_TCR_0/1)..... | 62-11 |
| 62.2.7 | DMA _n Timeout Count Registers 0 and 1 (USBDMAN_TOCNTR_0/1)..... | 62-12 |
| 62.2.8 | DMA _n Timeout Constant Registers 0 and 1 (USBDMAN_TOCSTR_0/1)..... | 62-13 |
| 62.2.9 | DMA _n Channel Control Registers 0 and 1 (USBDMAN_CHCR_0/1)..... | 62-14 |
| 62.2.10 | DMA _n Final Transaction Valid Data Transfer Enable Registers 0 and 1 (USBDMAN_TEND_0/1)..... | 62-17 |
| 62.2.11 | DMA _n Operation Register (USBDMAN_DMAOR)..... | 62-18 |
| 62.3 | Operation..... | 62-20 |
| 62.3.1 | Interrupt/Error Detection Function..... | 62-20 |
| 62.3.2 | DMA Transfer Function..... | 62-28 |
| 62.4 | DDM (Descriptor DMAC)..... | 62-38 |
| 62.4.1 | DDM Register Configuration..... | 62-39 |
| 62.4.2 | DDM Register Description..... | 62-42 |
| 62.4.2.1 | DDM Interrupt Source Mask Register (DDIREQMSK)..... | 62-42 |
| 62.4.2.2 | DDM Interrupt Source Register (DDIREQSTA)..... | 62-45 |
| 62.4.2.3 | DDM CH _m Start Source Mask Registers (DDINTMSK _{m1})..... | 62-48 |
| 62.4.2.4 | DDM CH _m Descriptor Pointer (DDPTR _m)..... | 62-49 |
| 62.4.2.5 | DDM CH _m Control Register (DDCTRL _m)..... | 62-50 |
| 62.4.3 | Internal RAM..... | 62-51 |
| 62.4.4 | Principles of Operation..... | 62-51 |
| 63. | USB3.0 Host Controller..... | 63-1 |
| 63.1 | Overview..... | 63-1 |
| 63.1.1 | Features..... | 63-2 |
| 63.1.2 | External Pins..... | 63-4 |
| 63.1.3 | Register Configuration..... | 63-5 |
| 63.1.3.1 | Register Attributes..... | 63-5 |
| 63.1.3.2 | Register Overview..... | 63-5 |
| 63.1.3.3 | USB3.0 Host Register Configuration..... | 63-6 |
| 63.1.3.4 | USB3.0 Peripheral Register Configuration..... | 63-11 |
| 63.1.3.5 | PHY Register Configuration..... | 63-14 |
| 63.1.4 | Connected Module..... | 63-15 |
| 63.2 | Register Description..... | 63-16 |
| 63.2.1 | USB3.0 Host Register..... | 63-16 |
| 63.2.1.1 | Host Controller Capability Register (HCSPARAMS0)..... | 63-16 |
| 63.2.1.2 | Host Controller Capability Register (HCSPARAMS1)..... | 63-17 |
| 63.2.1.3 | Host Controller Capability Register (HCSPARAMS2)..... | 63-18 |
| 63.2.1.4 | Host Controller Capability Register (HCSPARAMS3)..... | 63-19 |
| 63.2.1.5 | Host Controller Capability Register (HCCPARAMS)..... | 63-20 |
| 63.2.1.6 | Host Controller Capability Register (DBOFF)..... | 63-22 |
| 63.2.1.7 | Host Controller Capability Register (RSTOFF)..... | 63-23 |
| 63.2.1.8 | Host Controller Operational Register (USBCMD)..... | 63-24 |
| 63.2.1.9 | Host Controller Operational Register (USBSTS)..... | 63-26 |
| 63.2.1.10 | Host Controller Operational Register (PAGESIZE)..... | 63-28 |
| 63.2.1.11 | Host Controller Operational Register (DNCTRL)..... | 63-28 |

| | | |
|-----------|--|-------|
| 63.2.1.12 | Host Controller Operational Register (CRCRL) | 63-29 |
| 63.2.1.13 | Host Controller Operational Register (CRCRH)..... | 63-30 |
| 63.2.1.14 | Host Controller Operational Register (DCBAAPL)..... | 63-30 |
| 63.2.1.15 | Host Controller Operational Register (DCBAAPH)..... | 63-31 |
| 63.2.1.16 | Host Controller Operational Register (CONFIG) | 63-31 |
| 63.2.1.17 | AXI Host Control Register (AXH_CON) | 63-32 |
| 63.2.1.18 | AXI Host Control Status Register (AXH_STA) | 63-33 |
| 63.2.1.19 | Pseudo PCI Class Code Register (CLASS_CODE)..... | 63-34 |
| 63.2.1.20 | Frame Length Adjustment Register (FLADJ)..... | 63-34 |
| 63.2.1.21 | Host Interrupt Register (HOST_INT) | 63-35 |
| 63.2.1.22 | Host Interrupt Enable Register (INT_ENA)..... | 63-36 |
| 63.2.1.23 | Latency Tolerance Reporting Register (LTR)..... | 63-37 |
| 63.2.1.24 | Battery Charging Register (BTR_CHRG)..... | 63-38 |
| 63.2.1.25 | FW Version Register (FW_VER) | 63-39 |
| 63.2.1.26 | FW Download Control & Status Register (DL_CTRL)..... | 63-40 |
| 63.2.1.27 | FW Data0 Register (FW_DATA0) | 63-41 |
| 63.2.1.28 | FW Data1 Register (FW_DATA1) | 63-41 |
| 63.2.1.29 | Core Control 3 Register (CORE_CTRL3)..... | 63-42 |
| 63.2.1.30 | UTMI Plus Vendor Control Interface Set Register (VENDOR_CON_SET)..... | 63-43 |
| 63.2.1.31 | UTMI Plus Vendor Control Interface Status Register (VENDOR_CON_STA)..... | 63-43 |
| 63.2.1.32 | Port1 Status and Control Register (PORTSC1)..... | 63-44 |
| 63.2.1.33 | Port1 Status and Control Register (PORTPMSC1)..... | 63-48 |
| 63.2.1.34 | Port1 Status and Control Register (PORTPLI) | 63-49 |
| 63.2.1.35 | Port2 Status and Control Register (PORTSC2)..... | 63-50 |
| 63.2.1.36 | Port2 Status and Control Register (PORTPMSC2)..... | 63-54 |
| 63.2.1.37 | xHCI Extended Capabilities Register (USBLEGSUP) | 63-56 |
| 63.2.1.38 | xHCI Extended Capabilities Register (USBLEGCTLSTS) | 63-57 |
| 63.2.1.39 | xHCI Extended Capabilities Register (HCEXTDCAP1) | 63-58 |
| 63.2.1.40 | xHCI Extended Capabilities Register (HCEXTDCAP2) | 63-58 |
| 63.2.1.41 | xHCI Extended Capabilities Register (HCEXTDCAP3) | 63-59 |
| 63.2.1.42 | xHCI Extended Capabilities Register (HCEXTDCAP4) | 63-60 |
| 63.2.1.43 | xHCI Extended Capabilities Register (HCEXTDCAP5) | 63-61 |
| 63.2.1.44 | xHCI Extended Capabilities Register (HCEXTDCAP6)..... | 63-61 |
| 63.2.1.45 | xHCI Extended Capabilities Register (HCEXTDCAP7) | 63-62 |
| 63.2.1.46 | xHCI Extended Capabilities Register (HCEXTDCAP8) | 63-63 |
| 63.2.1.47 | xHCI Extended Capabilities Register (HCEXTDCAP9) | 63-64 |
| 63.2.1.48 | xHCI Extended Capabilities Register (HCEXTDCAP10) | 63-65 |
| 63.2.1.49 | xHCI Extended Power Management Capability Register (HCEXTDPWMNGCAP1) | 63-66 |
| 63.2.1.50 | xHCI Extended Power Management Capability Register (HCEXTDPWMNGCAP2) | 63-67 |
| 63.2.1.51 | xHCI Extended Power Management Capability Register (HCEXTDPWMNGCAP3) | 63-68 |
| 63.2.1.52 | xHCI Extended Power Management Capability Register (HCEXTDPWMNGCAP4) | 63-68 |
| 63.2.1.53 | xHCI Extended Power Management Capability Register (DCID)..... | 63-69 |
| 63.2.1.54 | xHCI Extended Power Management Capability Register (DCDB) | 63-70 |
| 63.2.1.55 | xHCI Extended Power Management Capability Register (DCERSTSZ)..... | 63-70 |
| 63.2.1.56 | xHCI Extended Power Management Capability Register (DCERSTBA0)..... | 63-71 |
| 63.2.1.57 | xHCI Extended Power Management Capability Register (DCERSTBA1)..... | 63-71 |
| 63.2.1.58 | xHCI Extended Power Management Capability Register (DCERDP0)..... | 63-72 |
| 63.2.1.59 | xHCI Extended Power Management Capability Register (DCERDP1)..... | 63-72 |
| 63.2.1.60 | xHCI Extended Power Management Capability Register (DCCTRL)..... | 63-73 |
| 63.2.1.61 | xHCI Extended Power Management Capability Register (DCST) | 63-74 |
| 63.2.1.62 | xHCI Extended Power Management Capability Register (DCPORTSC)..... | 63-75 |
| 63.2.1.63 | xHCI Extended Power Management Capability Register (DCCP0) | 63-76 |
| 63.2.1.64 | xHCI Extended Power Management Capability Register (DCCP1) | 63-77 |
| 63.2.1.65 | Host Controller Runtime Register (MFINDEX) | 63-77 |

| | | |
|-----------|---|--------|
| 63.2.1.66 | Host Controller Runtime Register (IMAN)..... | 63-78 |
| 63.2.1.67 | Host Controller Runtime Register (IMOD)..... | 63-79 |
| 63.2.1.68 | Host Controller Runtime Register (ERSTSZ)..... | 63-79 |
| 63.2.1.69 | Host Controller Runtime Register (ERSTBA_L)..... | 63-80 |
| 63.2.1.70 | Host Controller Runtime Register (ERSTBA_H)..... | 63-80 |
| 63.2.1.71 | Host Controller Runtime Register (ERDP_L)..... | 63-81 |
| 63.2.1.72 | Host Controller Runtime Register (ERDP_H)..... | 63-81 |
| 63.2.1.73 | Host Controller Doorbell Register (DOORBELL0)..... | 63-82 |
| 63.2.1.74 | Host Controller Doorbell Register (DOORBELL1-32)..... | 63-83 |
| 63.2.2 | USB3.0 Peripheral Register..... | 63-84 |
| 63.2.2.1 | AXI Control Register (AXI_CON)..... | 63-84 |
| 63.2.2.2 | AXI Status Register (AXI_STA)..... | 63-86 |
| 63.2.2.3 | AXI Interrupt Status Register (AXI_INT_STA)..... | 63-88 |
| 63.2.2.4 | AXI Interrupt Enable Register (AXI_INT_ENA)..... | 63-90 |
| 63.2.2.5 | DMA Interrupt Status Register (DMA_INT_STA)..... | 63-92 |
| 63.2.2.6 | DMA Interrupt Enable Register (DMA_INT_ENA)..... | 63-93 |
| 63.2.2.7 | Data Enable Status Register (DATAEN_STA)..... | 63-94 |
| 63.2.2.8 | AXI Common Input Register (AXI_COM_IN)..... | 63-94 |
| 63.2.2.9 | AXI Common Output Register (AXI_COM_OUT)..... | 63-95 |
| 63.2.2.10 | DMA Ch0 Control Register1 (DMA_Ch0_CON1)..... | 63-96 |
| 63.2.2.11 | DMA Ch0 PRD Address Register1 (DMA_Ch0_PRD_ADR1)..... | 63-97 |
| 63.2.2.12 | DMA Ch0 Control Register2 (DMA_Ch0_CON2)..... | 63-98 |
| 63.2.2.13 | DMA Ch0 PRD Address Register2 (DMA_Ch0_PRD_ADR2)..... | 63-99 |
| 63.2.2.14 | DMA Ch0 Control Register3 (DMA_Ch0_CON3)..... | 63-100 |
| 63.2.2.15 | DMA Ch0 PRD Address Register3 (DMA_Ch0_PRD_ADR3)..... | 63-101 |
| 63.2.2.16 | DMA Ch0 Control Register4 (DMA_Ch0_CON4)..... | 63-102 |
| 63.2.2.17 | DMA Ch0 PRD Address Register4 (DMA_Ch0_PRD_ADR4)..... | 63-103 |
| 63.2.2.18 | USB Common Control Register (USB_COM_CON)..... | 63-104 |
| 63.2.2.19 | USB20 Control Register (USB20_CON)..... | 63-106 |
| 63.2.2.20 | USB30 Control Register (USB30_CON)..... | 63-109 |
| 63.2.2.21 | USB Status Register (USB_STA)..... | 63-112 |
| 63.2.2.22 | USB20 Frame No. Register (USB20_FRAME)..... | 63-113 |
| 63.2.2.23 | DRD Common Control Register (DRD_CON)..... | 63-114 |
| 63.2.2.24 | DRD Common Control 2 Register (DRD_CON2)..... | 63-117 |
| 63.2.2.25 | USB Interrupt Status 1 Register (USB_INT_STA_1)..... | 63-119 |
| 63.2.2.26 | USB Interrupt Status 2 Register (USB_INT_STA_2)..... | 63-121 |
| 63.2.2.27 | USB Interrupt Enable 1 Register (USB_INT_ENA_1)..... | 63-122 |
| 63.2.2.28 | USB Interrupt Enable 2 Register (USB_INT_ENA_2)..... | 63-123 |
| 63.2.2.29 | Setup Data 0 Register (STUP_DAT_0)..... | 63-124 |
| 63.2.2.30 | Setup Data 1 Register (STUP_DAT_1)..... | 63-124 |
| 63.2.2.31 | USB30 TP_Send Data 0 Register (USB3_TPDAT_0)..... | 63-125 |
| 63.2.2.32 | USB30 TP_Send Data 1 Register (USB3_TPDAT_1)..... | 63-125 |
| 63.2.2.33 | USB30 TP_Send Data 2 Register (USB3_TPDAT_2)..... | 63-126 |
| 63.2.2.34 | USB20 LPM Status Register (USB20_LPM_STA)..... | 63-127 |
| 63.2.2.35 | USB30 Vendor Device Test Register (USB30_VND_DEV)..... | 63-128 |
| 63.2.2.36 | USB30 Vendor Defined Data 0 Register (USB30_VND_DAT_0)..... | 63-128 |
| 63.2.2.37 | USB30 Vendor Defined Data 1 Register (USB30_VND_DAT_1)..... | 63-129 |
| 63.2.2.38 | EPC Version Register (USB_VER)..... | 63-129 |
| 63.2.2.39 | Scratch Pad Register (USB_SCRATCH)..... | 63-130 |
| 63.2.2.40 | USB OTG Status Register (USB_OTG_STA)..... | 63-130 |
| 63.2.2.41 | USB OTG Interrupt Status Register (USB_OTG_INT_STA)..... | 63-131 |
| 63.2.2.42 | USB OTG Interrupt Enable Register (USB_OTG_INT_ENA)..... | 63-132 |
| 63.2.2.43 | PIPE0 Mode Setting Register (P0_MOD)..... | 63-133 |
| 63.2.2.44 | PIPE0 Control Register (P0_CON)..... | 63-134 |

| | | |
|-----------|--|--------|
| 63.2.2.45 | PIPE0 Status Register (P0_STA) | 63-136 |
| 63.2.2.46 | PIPE0 Interrupt Status Register (P0_INT_STA)..... | 63-137 |
| 63.2.2.47 | PIPE0 Interrupt Enable Register (P0_INT_ENA)..... | 63-139 |
| 63.2.2.48 | UTMI Plus Vendor Control Interface Set Register (VENDOR_CON_SET)..... | 63-140 |
| 63.2.2.49 | UTMI Plus Vendor Control Interface Status Register (VENDOR_CON_STA)..... | 63-140 |
| 63.2.2.50 | PIPE0 Length Register (P0_LNG)..... | 63-141 |
| 63.2.2.51 | PIPE0 Read Register (P0_READ) | 63-142 |
| 63.2.2.52 | PIPE0 Write Register (P0_WRITE)..... | 63-143 |
| 63.2.2.53 | DRD Power Control Register (DRD_PW_CON) | 63-143 |
| 63.2.2.54 | PIPE Common Setting Register (PIPE_COM) | 63-144 |
| 63.2.2.55 | PIPEn Mode Setting Register (Pn_MOD)..... | 63-145 |
| 63.2.2.56 | PIPEn RAM Mapping Register (Pn_RAMMAP) | 63-146 |
| 63.2.2.57 | PIPEn Control Register (Pn_CON)..... | 63-148 |
| 63.2.2.58 | PIPEn Status Register (Pn_STA) | 63-151 |
| 63.2.2.59 | PIPEn Interrupt Status Register (Pn_INT_STA)..... | 63-152 |
| 63.2.2.60 | PIPEn Interrupt Enable Register (Pn_INT_ENA)..... | 63-155 |
| 63.2.2.61 | PIPEn Reserved Packet Register (Pn_RSVPKT)..... | 63-157 |
| 63.2.2.62 | PIPEn Length Register (Pn_LNG)..... | 63-158 |
| 63.2.2.63 | PIPEn Read Register (Pn_READ) | 63-159 |
| 63.2.2.64 | PIPEn Write Register (Pn_WRITE)..... | 63-160 |
| 63.2.2.65 | PIPEn Stream ID Register (Pn_STREAM)..... | 63-161 |
| 63.2.2.66 | PIPEn NRDY Stream ID Register (Pn_NRDYSTRM)..... | 63-162 |
| 63.2.2.67 | Port Status and Control Register (PORTSC)..... | 63-163 |
| 63.2.2.68 | Port PM Status and Control Register (PORTPMSC) | 63-165 |
| 63.2.2.69 | Port Link Information Register (PORTLI)..... | 63-168 |
| 63.2.2.70 | SSIF Command Register (SSIFCMD) | 63-169 |
| 63.2.2.71 | SSIF Link Setting 2 Register (SSIFLINKSET2)..... | 63-171 |
| 63.2.2.72 | SSIF Link Setting 3 Register (SSIFLINKSET3)..... | 63-172 |
| 63.2.2.73 | SSIF Link Setting 4 Register (SSIFLINKSET4)..... | 63-173 |
| 63.2.3 | PHY Register | 63-174 |
| 63.2.3.1 | USB30 Clock Setting Register (USB30_CLKSET0)..... | 63-174 |
| 63.2.3.2 | USB30 Clock Setting Register (USB30_CLKSET1)..... | 63-175 |
| 63.2.3.3 | USB30 SSC Setting Register (USB30_SSC_SET)..... | 63-176 |
| 63.2.3.4 | USB30 PHY Enable Register (USB30_PHY_ENABLE)..... | 63-177 |
| 63.2.3.5 | USB30 VBUS Enable Register (USB30_VBUS_EN)..... | 63-177 |
| 63.3 | Operation | 63-178 |
| 63.3.1 | Reset | 63-178 |
| 63.3.1.1 | Host Controller core..... | 63-179 |
| 63.3.1.2 | Peripheral Controller core | 63-179 |
| 63.3.1.3 | PHY Reset..... | 63-179 |
| 63.4 | Description of Host Functions | 63-181 |
| 63.4.1 | FW Download..... | 63-181 |
| 63.4.1.1 | Download Data Format | 63-181 |
| 63.4.1.2 | System Interface..... | 63-183 |
| 63.4.2 | Battery Charging | 63-185 |
| 63.4.2.1 | A-Device side (Charging Port Device) | 63-185 |
| 63.4.3 | Power Management | 63-188 |
| 63.4.3.1 | Power Management Capability | 63-188 |
| 63.4.3.2 | Low Power Management of Receiver Detection..... | 63-188 |
| 63.4.3.3 | Power Control for Each State..... | 63-189 |
| 63.4.3.4 | Clock Termination by USB3PHY and LTSSM State | 63-190 |
| 63.4.3.5 | Flow of System Suspend and Resume | 63-191 |
| 63.4.4 | Power Control..... | 63-192 |
| 63.4.4.1 | VBUS On | 63-192 |

| | | |
|-----------|---|--------|
| 63.4.4.2 | VBUS Off | 63-192 |
| 63.4.4.3 | Over Current Detection | 63-192 |
| 63.4.5 | Structure of Notification for Elements of Interrupt..... | 63-193 |
| 63.4.6 | Implementation Constraint of Host Core | 63-194 |
| 63.5 | Example of Transaction | 63-195 |
| 63.5.1 | Bulk OUT Transaction..... | 63-195 |
| 63.5.2 | Bulk IN Transaction..... | 63-196 |
| 63.6 | Power IC Interface Connection..... | 63-197 |
| 63.6.1 | Usable Power IC type | 63-197 |
| 63.7 | Descriptions of functions | 63-198 |
| 63.7.1 | VBUS Connection | 63-198 |
| 63.7.2 | Functions of EPC | 63-200 |
| 63.7.2.1 | Buffering for transmission and reception on USB | 63-200 |
| 63.7.2.2 | Handshake..... | 63-201 |
| 63.7.3 | USB3.0/USB2.0 connection | 63-203 |
| 63.7.4 | Power Management Control | 63-205 |
| 63.7.4.1 | USB3.0 Power Control..... | 63-205 |
| 63.7.4.2 | USB3.0 Power State Control..... | 63-207 |
| 63.7.4.3 | USB2.0 Power State Control..... | 63-214 |
| 63.7.5 | AXI DMA Transfer Control | 63-217 |
| 63.7.5.1 | OUT Transfer | 63-217 |
| 63.7.5.2 | IN Transfer | 63-218 |
| 63.7.5.3 | Two PRD tables for one group of transfer | 63-220 |
| 63.7.5.4 | DMA Transfer using EXTPRD_EN..... | 63-221 |
| 63.7.5.5 | Example of transfer using bus master function | 63-223 |
| 63.7.5.6 | Finishing the process for PRD table and its conditions..... | 63-224 |
| 63.7.5.7 | Case of error..... | 63-225 |
| 63.7.5.8 | Ordering of transactions..... | 63-225 |
| 63.7.5.9 | Write Transfer with all zero strobe..... | 63-226 |
| 63.7.5.10 | Interrupts | 63-227 |
| 63.7.6 | List of Interrupt factors | 63-228 |
| 63.7.7 | Zero Length Packet (ZLP) | 63-232 |
| 63.7.7.1 | Definition of Zero Length Packet..... | 63-232 |
| 63.7.7.2 | Transfer of ZLP support mode | 63-232 |
| 63.7.7.3 | In the configuration with AXI-IF..... | 63-232 |
| 63.7.8 | Physical Region Descriptor (PRD) Table | 63-234 |
| 63.8 | Example of Software flow | 63-238 |
| 63.8.1 | Outline of software flow | 63-238 |
| 63.8.2 | Initialization | 63-239 |
| 63.8.3 | Interrupt process..... | 63-241 |
| 63.8.3.1 | Interrupt handler process..... | 63-241 |
| 63.8.3.2 | USB Connection..... | 63-243 |
| 63.8.3.3 | USB Disconnection..... | 63-246 |
| 63.8.3.4 | Warm Reset Process..... | 63-247 |
| 63.8.3.5 | Hot Reset Process..... | 63-250 |
| 63.8.3.6 | USB2.0 Bus Reset Process..... | 63-252 |
| 63.8.4 | Control Transfer..... | 63-257 |
| 63.8.4.1 | No Data Control Process | 63-259 |
| 63.8.4.2 | Control Write Process | 63-261 |
| 63.8.4.3 | Control Read Process | 63-263 |
| 63.8.5 | Bulk Transfer | 63-265 |
| 63.8.5.1 | Bulk-IN Process | 63-266 |
| 63.8.5.2 | Bulk-OUT Process | 63-268 |
| 63.8.5.3 | Stream Bulk Process | 63-270 |

| | | |
|------------|---|-------------|
| 63.8.5.4 | SuperSpeed UASP Non-Data Transfer | 63-276 |
| 63.8.5.5 | SuperSpeed UASP Data-IN transfer | 63-277 |
| 63.8.5.6 | SuperSpeed UASP Data-OUT transfer | 63-278 |
| 63.8.6 | Interrupt Transfer | 63-279 |
| 63.8.6.1 | Interrupt IN Transfer | 63-280 |
| 63.8.7 | DMA Transfer | 63-282 |
| 63.8.7.1 | Settings for DMA transfer | 63-282 |
| 63.8.8 | Power Management | 63-284 |
| 63.8.8.1 | Suspend in USB3.0 (transition of state from U0 to U3)..... | 63-284 |
| 63.8.8.2 | Resume in USB3.0 (from U3 to U0: Initiated by host)..... | 63-286 |
| 63.8.8.3 | Resume in USB3.0 (from U3 to U0: Initiated by device) | 63-287 |
| 63.8.8.4 | Function Suspend..... | 63-288 |
| 63.8.8.5 | Function Remote Wakeup..... | 63-289 |
| 63.8.8.6 | USB3.0-PHY (PLL) Wakeup Process..... | 63-291 |
| 63.8.8.7 | Suspend in USB2.0 | 63-292 |
| 63.8.8.8 | Resume in USB2.0 (Initiated by host)..... | 63-293 |
| 63.8.8.9 | Resume in USB2.0 (Initiated by device)..... | 63-294 |
| 63.8.8.10 | USB2.0 PHY (PLL) Wakeup Process..... | 63-295 |
| 63.8.8.11 | Process of transition from U0 to U1/U2 in USB3.0 (Initiated by device)..... | 63-296 |
| 63.8.9 | Device Requests..... | 63-298 |
| 63.8.9.1 | Clear Feature | 63-298 |
| 63.8.9.2 | Get Configuration..... | 63-300 |
| 63.8.9.3 | Get Descriptor | 63-300 |
| 63.8.9.4 | Get Interface..... | 63-300 |
| 63.8.9.5 | Get Status | 63-300 |
| 63.8.9.6 | Set Address | 63-303 |
| 63.8.9.7 | Set Configuration | 63-303 |
| 63.8.9.8 | Set Descriptor..... | 63-303 |
| 63.8.9.9 | Set Feature..... | 63-304 |
| 63.8.9.10 | Set Interface | 63-305 |
| 63.8.9.11 | Set Isochronous Delay..... | 63-306 |
| 63.8.9.12 | Set SEL | 63-306 |
| 63.8.9.13 | Synch Frame | 63-306 |
| 63.8.10 | BOT Control | 63-307 |
| 63.8.10.1 | Overall..... | 63-307 |
| 63.8.10.2 | BOT assistance mode..... | 63-307 |
| 63.8.10.3 | BOT Process | 63-308 |
| 63.9 | Instructions and Directions for Use..... | 63-310 |
| 63.9.1 | Method of Specifying RAM Mapping | 63-310 |
| 63.9.2 | Example of Setting..... | 63-311 |
| 63.9.3 | Register Access | 63-312 |
| 64. | RCLK Watchdog Timer (RWDT) | 64-1 |
| 64.1 | Overview..... | 64-1 |
| 64.1.1 | Features..... | 64-1 |
| 64.1.2 | Block Diagram | 64-2 |
| 64.1.3 | Register Configuration..... | 64-3 |
| 64.2 | Register Description..... | 64-4 |
| 64.2.1 | RCLK Watchdog Timer Counter (RWCNT)..... | 64-4 |
| 64.2.2 | RCLK Watchdog Timer Control/Status Register A (RWCSRA)..... | 64-5 |
| 64.2.3 | RCLK Watchdog Timer Control/Status Register B (RWCSR) | 64-7 |
| 64.2.4 | Notes on Register Access..... | 64-9 |
| 64.3 | Operation | 64-10 |
| 64.3.1 | Control of System Runaway | 64-10 |

| | | |
|---------|---|-------|
| 64.3.2 | Module Stop..... | 64-10 |
| 64.3.3 | Debugging Operation..... | 64-10 |
| 64.3.4 | Formula to calculate overflow time | 64-11 |
| 65. | System Watchdog Timer (SWDT)..... | 65-1 |
| 65.1 | Overview..... | 65-1 |
| 65.1.1 | Features..... | 65-1 |
| 65.1.2 | Block Diagram..... | 65-1 |
| 65.1.3 | Register Configuration..... | 65-2 |
| 65.2 | Register Description..... | 65-3 |
| 65.2.1 | System Watchdog Timer Counter (SWTCNT)..... | 65-3 |
| 65.2.2 | System Watchdog Timer Control/Status Register A (SWTCSRA) | 65-4 |
| 65.2.3 | System Watchdog Timer Control/Status Register B (SWTCSR)..... | 65-6 |
| 65.2.4 | Notes on Register Access..... | 65-8 |
| 65.3 | Operation | 65-9 |
| 65.3.1 | Control of System Runaway | 65-9 |
| 65.3.2 | Debugging Operation..... | 65-9 |
| 65.3.3 | Formula to calculate overflow time | 65-9 |
| 66. | 16-Bit Timer Pulse Unit (TPU)..... | 66-1 |
| 66.1 | Overview..... | 66-1 |
| 66.1.1 | Features..... | 66-1 |
| 66.1.2 | Block Diagram..... | 66-3 |
| 66.1.3 | External Pins | 66-4 |
| 66.1.4 | Register Configuration..... | 66-5 |
| 66.2 | Register Description..... | 66-8 |
| 66.2.1 | Timer Control Register (TCR)..... | 66-8 |
| 66.2.2 | Timer Mode Register (TMDR)..... | 66-10 |
| 66.2.3 | Timer I/O Control Register (TIOR)..... | 66-11 |
| 66.2.4 | Timer Interrupt Enable Register (TIER)..... | 66-12 |
| 66.2.5 | Timer Status Registers (TSR) | 66-14 |
| 66.2.6 | Timer Counter (TCNT)..... | 66-16 |
| 66.2.7 | Timer General Register A to D (TGRA/TGRB/TGRC/TGRD) | 66-17 |
| 66.2.8 | Timer Start Register (TSTR)..... | 66-18 |
| 66.2.9 | Motor Control Setting Register (TMIR)..... | 66-19 |
| 66.2.10 | Motor Deceleration (Stop) Transition Register (TMRR)..... | 66-20 |
| 66.2.11 | Motor Control Status Register (TMSR)..... | 66-21 |
| 66.2.12 | Motor Operation Pattern Storing Register 0 (TMMPR0) | 66-22 |
| 66.2.13 | Motor Operation Pattern Storing Register 1 (TMMPR1) | 66-23 |
| 66.2.14 | Motor Stop Pattern Storing Register 0 (TMSPR0) | 66-24 |
| 66.2.15 | Motor Stop Pattern Storing Register 1 (TMSPR1) | 66-25 |
| 66.2.16 | Motor Output Pattern Storing Register (TMOPR)..... | 66-26 |
| 66.2.17 | Motor Acceleration Step Count Register (TMASR)..... | 66-26 |
| 66.2.18 | Motor Normal the Number of Steps Register (TMTSR) | 66-27 |
| 66.2.19 | Motor Deceleration the Number of Steps Register (TMRSR) | 66-27 |
| 66.2.20 | Motor Control Sequence Counter Register (TMSCR)..... | 66-28 |
| 66.2.21 | Motor Control Normal Counter Register (TMTCR)..... | 66-28 |
| 66.3 | Operation | 66-29 |
| 66.3.1 | Overview..... | 66-29 |
| 66.3.2 | Basic Functions..... | 66-30 |
| 66.3.3 | Buffered Operation | 66-33 |
| 66.3.4 | PWM Mode..... | 66-35 |
| 66.3.5 | Stepping-Motor Control Mode..... | 66-37 |

| | | |
|---------|---|-------|
| 67. | Compare Match Timer Type0 (CMT0) | 67-1 |
| 67.1 | Overview | 67-1 |
| 67.1.1 | Features | 67-1 |
| 67.1.2 | Block Diagram | 67-1 |
| 67.1.3 | Register Configuration | 67-2 |
| 67.2 | Register Description | 67-3 |
| 67.2.1 | CLK Enable Register (CMCLKE) | 67-3 |
| 67.2.2 | Compare Match Timer Start Registers 0 and 1 (CMSTR0, CMSTR1) | 67-4 |
| 67.2.3 | Compare Match Timer Control/Status Registers 0 and 1 (CMCSR0, CMCSR1) | 67-5 |
| 67.2.4 | Compare Match Timer Counters 0 and 1 (CMCNT0, CMCNT1) | 67-7 |
| 67.2.5 | Compare Match Timer Constant Registers 0 and 1 (CMCOR0, CMCOR1) | 67-8 |
| 67.3 | Operation | 67-9 |
| 67.3.1 | Counter Operation | 67-9 |
| 67.3.2 | Counter Size | 67-11 |
| 67.3.3 | Timing for Counting by CMCNTn | 67-11 |
| 67.3.4 | Internal Interrupt Request to CPU | 67-11 |
| 67.3.5 | CMT0 Register Access | 67-11 |
| 67.3.6 | Compare Match Flag Set/Clear Timing | 67-12 |
| 67.3.7 | CMT0 Usage | 67-12 |
| 67.3.8 | Module Stop/ CMCLKE Setting | 67-12 |
| 68. | Compare Match Timer Type1 (CMT1) | 68-1 |
| 68.1 | Overview | 68-1 |
| 68.1.1 | Features | 68-1 |
| 68.1.2 | Block Diagram | 68-2 |
| 68.1.3 | Register Configuration | 68-3 |
| 68.2 | Register Description | 68-11 |
| 68.2.1 | CLK Enable Register (CMCLKE) | 68-11 |
| 68.2.2 | Compare Match Timer Start Registers 0 to 7 (CMSTRn) | 68-13 |
| 68.2.3 | Compare Match Timer Control/Status Registers 0 to 7 (CMCSRn) | 68-16 |
| 68.2.4 | Compare Match Timer Control/Status Registers H0 to 4 (CMCSRHn) | 68-23 |
| 68.2.5 | Compare Match Timer Counters 0 to 7 (CMCNTn) | 68-24 |
| 68.2.6 | Compare Match Timer Counters H 0 to 4 (CMCNTHn) | 68-25 |
| 68.2.7 | Compare Match Timer Constant Registers 0 to 7 (CMCORn) | 68-26 |
| 68.2.8 | Compare Match Timer Constant Registers H0 to 4 (CMCORHn) | 68-27 |
| 68.2.9 | Compare Match Timer Counter 3 Backup 0 (CMCNT3BK0) | 68-28 |
| 68.2.10 | Compare Match Timer Counter 3 Backup 1 (CMCNT3BK1) | 68-28 |
| 68.2.11 | Compare Match Timer Match Control/Status Registers 0 and 3 (CMCSRm) | 68-29 |
| 68.2.12 | Compare Match Timer Match Counters 0 and 3 (CMCNTMn) | 68-30 |
| 68.3 | Operation | 68-31 |
| 68.3.1 | Counter Operation | 68-31 |
| 68.3.2 | Counter Size | 68-33 |
| 68.3.3 | Timing for Counting by CMCNTn | 68-33 |
| 68.3.4 | Internal Interrupt Request to CPU | 68-33 |
| 68.3.5 | Register Access | 68-33 |
| 68.3.6 | Compare Match Flag Set/Clear Timing | 68-34 |
| 68.3.7 | RCLK-Synchronous Counter Start/Stop Mode in Channel 0 | 68-34 |
| 68.3.8 | Pseudo 32-kHz Counter | 68-38 |
| 68.3.9 | CMT1 Usage | 68-38 |
| 68.3.10 | Module Stop/ CMCLKE Setting | 68-38 |
| 69. | Timer Unit (TMU) | 69-1 |
| 69.1 | Overview | 69-1 |
| 69.1.1 | Features | 69-2 |

| | | |
|--------|---|-------|
| 69.1.2 | Block Diagram | 69-4 |
| 69.1.3 | External Pins | 69-5 |
| 69.1.4 | Register Configuration..... | 69-6 |
| 69.2 | Register Description..... | 69-11 |
| 69.2.1 | Timer Start Registers (TSTRn) (n = 0 to 4)..... | 69-11 |
| 69.2.2 | Timer Constant Registers (TCORn) (n = 0 to 14)..... | 69-14 |
| 69.2.3 | Timer Counters (TCNTn) (n = 0 to 14) | 69-14 |
| 69.2.4 | Timer Control Registers (TCRn) (n = 0 to 14) | 69-15 |
| 69.2.5 | Input Capture Registers n (TCPRn) | 69-18 |
| 69.3 | Operation | 69-19 |
| 69.3.1 | Counter Operation..... | 69-19 |
| 69.3.2 | Input Capture Function | 69-23 |
| 69.3.3 | Interrupt | 69-24 |
| 69.4 | Usage Notes | 69-25 |
| 69.4.1 | Writing to Registers | 69-25 |
| 69.4.2 | Reading TCNT Register | 69-25 |
| 69.4.3 | External Clock Frequency..... | 69-25 |
| 70. | System Timer (SCMT) | 70-1 |
| 70.1 | Overview..... | 70-1 |
| 70.1.1 | Features..... | 70-1 |
| 70.1.2 | Block Diagram | 70-2 |
| 70.1.3 | Register Configuration..... | 70-2 |
| 70.2 | Register Description..... | 70-4 |
| 70.2.1 | Compare Match Timer Start Register (CMSSTR)..... | 70-4 |
| 70.2.2 | Compare Match Timer Control/Status Register (CMSCSR) | 70-5 |
| 70.2.3 | Compare Match Timer Counter (CMSCNT) | 70-7 |
| 70.2.4 | Compare Match Timer Constant Register (CMSCOR) | 70-8 |
| 70.3 | Operation | 70-9 |
| 70.3.1 | Counter Operation..... | 70-9 |
| 70.3.2 | Counter Size..... | 70-10 |
| 70.3.3 | Timing for Counting by CMSCNT | 70-10 |
| 70.3.4 | Internal Interrupt Request to CPU | 70-10 |
| 70.3.5 | Register Access..... | 70-10 |
| 70.3.6 | Compare Match Flag Set/Clear Timing | 70-11 |
| 70.3.7 | System Timer Usage | 70-11 |
| 71. | System Up-Time Clock (SUCMT) | 71-1 |
| 71.1 | Overview..... | 71-1 |
| 71.1.1 | Features..... | 71-1 |
| 71.1.2 | Block Diagram | 71-2 |
| 71.1.3 | Register Configuration..... | 71-2 |
| 71.2 | Register Description..... | 71-4 |
| 71.2.1 | Compare Match Timer Start Register (CMUSTR) | 71-4 |
| 71.2.2 | Compare Match Timer Control/Status Register (CMUCSR) (T.B.C.) | 71-5 |
| 71.2.3 | Compare Match Timer Counter (CMUCNT)..... | 71-8 |
| 71.2.4 | Compare Match Timer Constant Register (CMUCOR)..... | 71-9 |
| 71.3 | Operation | 71-10 |
| 71.3.1 | Counter Operation..... | 71-10 |
| 71.3.2 | Counter Size..... | 71-11 |
| 71.3.3 | Timing for Counting by CMUCNT | 71-11 |
| 71.3.4 | Internal Interrupt Request to CPU | 71-11 |
| 71.3.5 | Register Access..... | 71-12 |
| 71.3.6 | Compare Match Flag Set/Clear Timing | 71-12 |

| | | |
|-----------|---|-------|
| 71.3.7 | System Up-Tim Clock Usage | 71-13 |
| 72. | Debug and Trace | 72-1 |
| 72.1 | Overview..... | 72-1 |
| 72.1.1 | Features..... | 72-1 |
| 72.1.1.1 | RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 features | 72-1 |
| 72.1.1.2 | RZ/G2N features | 72-2 |
| 72.1.1.3 | RZ/G2E features..... | 72-3 |
| 72.1.2 | Block Diagram | 72-4 |
| 72.1.3 | External Pins | 72-8 |
| 72.1.3.1 | JTAG Pins | 72-8 |
| 72.1.3.2 | Trace Pins..... | 72-11 |
| 72.1.3.3 | System Trace Pins | 72-12 |
| 72.1.4 | Register Configuration..... | 72-12 |
| 72.1.4.1 | Debug resource configurations..... | 72-13 |
| 72.1.4.2 | CSreg registers configurations | 72-15 |
| 72.1.4.3 | TDBG registers configurations | 72-16 |
| 72.1.5 | Connected module | 72-16 |
| 72.1.6 | Part number and TARGETID | 72-17 |
| 72.2 | Register Description..... | 72-18 |
| 72.2.1 | CoreSight and Debug registers | 72-18 |
| 72.2.2 | CSreg registers | 72-19 |
| 72.2.2.1 | ATCLKCR_CA57 Register | 72-19 |
| 72.2.2.2 | CSREG_E00 Register | 72-20 |
| 72.2.2.3 | LOCKACCESS Register..... | 72-21 |
| 72.2.2.4 | LOCKSTATUS Register..... | 72-21 |
| 72.2.2.5 | Peripheral ID4 Register..... | 72-22 |
| 72.2.2.6 | Peripheral ID0 Register..... | 72-22 |
| 72.2.2.7 | Peripheral ID1 Register..... | 72-23 |
| 72.2.2.8 | Peripheral ID2 Register..... | 72-23 |
| 72.2.2.9 | Peripheral ID3 Register..... | 72-24 |
| 72.2.2.10 | Component ID0 Register..... | 72-24 |
| 72.2.2.11 | Component ID1 Register..... | 72-25 |
| 72.2.2.12 | Component ID2 Register..... | 72-25 |
| 72.2.2.13 | Component ID3 Register..... | 72-26 |
| 72.2.3 | TDBG registers | 72-27 |
| 72.2.3.1 | Debug register 1 (DBGREG1) | 72-27 |
| 72.2.3.2 | Debug register 4 (DBGREG4) | 72-30 |
| 72.2.3.3 | Debug register 9 (DBGREG9) | 72-32 |
| 72.3 | Operation | 72-33 |
| 72.3.1 | JTAG selection | 72-33 |
| 72.3.2 | Trace output selection..... | 72-35 |
| 72.3.3 | STM output selection..... | 72-35 |
| 72.3.4 | STM Master ID mapping | 72-35 |
| 72.3.5 | STM AXI Slave Interface | 72-36 |
| 72.3.5.1 | STM FIXED AWBURST [RZ/G2E] | 72-36 |
| 72.3.6 | Cross Trigger Connection | 72-37 |
| 72.3.7 | ATB Funnel information..... | 72-42 |
| 72.4 | Usage Notes | 72-43 |
| 72.4.1 | EDPRSR.PU polling | 72-43 |
| 72.4.2 | TRST control in no JTAG selection..... | 72-43 |
| 72.4.3 | Pseudo Power down..... | 72-43 |
| 72.4.4 | Important notice for soft power-on reset | 72-44 |

| | |
|---|--------|
| 73. Electrical Characteristics | 73-1 |
| 73.1 Absolute Maximum Ratings | 73-1 |
| 73.2 Power Supply | 73-6 |
| 73.3 Sequence of Turning On/Off Power Supplies | 73-10 |
| 73.3.1 Sequence of Turning On/Off Power Supplies for [RZ/G2H] [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] | 73-10 |
| 73.3.2 Sequence of Turning On/Off Power Supplies for [RZ/G2E] | 73-14 |
| 73.3.3 Wave form definition for power sequence | 73-16 |
| 73.3.4 Power On and Power Off Wave Form | 73-17 |
| 73.4 DC Characteristics | 73-18 |
| 73.5 Clock and Reset Timings | 73-44 |
| 73.6 EXTAL Clock Input/output Timing | 73-45 |
| 73.6.1 EXTAL Clock Input Timing | 73-45 |
| 73.6.2 CLKOUT Clock Output Timing | 73-46 |
| 73.7 EXTAL Clock Input/output Timing | 73-47 |
| 73.7.1 EXTAL Clock Input Timing | 73-47 |
| 73.8 PLL Characteristics | 73-48 |
| 73.9 EXTALR Input Timing | 73-49 |
| 73.10 LBSC | 73-50 |
| 73.10.1 Normal Read/Write, Burst ROM Read | 73-50 |
| 73.11 CSI2 | 73-52 |
| 73.12 Video Input Module (VIN) | 73-53 |
| 73.13 Display Unit (DU) | 73-56 |
| 73.14 LVDS-IF | 73-64 |
| 73.15 SSI Interface | 73-68 |
| 73.16 Ethernet AVB-IF Module Signal Timing | 73-71 |
| 73.17 CAN, CAN-FD | 73-78 |
| 73.18 PCIE Controller | 73-79 |
| 73.19 SCIF | 73-80 |
| 73.20 HSCIF | 73-81 |
| 73.21 I2C | 73-82 |
| 73.22 IIC Bus Interface | 73-83 |
| 73.23 MSIOF | 73-85 |
| 73.24 RPC Interface | 73-88 |
| 73.25 SD Host Interface (SDHI) | 73-93 |
| 73.25.1 SDHI (3.3V) Electrical Characteristics | 73-93 |
| 73.25.2 SDHI (1.8V) Electrical Characteristics | 73-95 |
| 73.26 MMC Interface | 73-97 |
| 73.26.1 MMC (3.3V) Electrical Characteristics [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E] | 73-97 |
| 73.26.2 MMC (1.8V) Electrical Characteristics [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E] | 73-98 |
| 73.27 Serial ATA (Gen3) | 73-101 |
| 73.28 USB Signal Timing | 73-102 |
| 73.29 USB 3.0 (Super-Speed) | 73-104 |
| 73.30 TMU | 73-105 |
| 73.31 R-NANDC | 73-106 |
| 73.32 DBSC4 Access Timing | 73-109 |
| 73.33 Debug and Trace | 73-119 |
| 73.33.1 JTAG IF | 73-119 |
| 73.34 GPIO | 73-121 |
| 73.35 Internal PLL Characteristics (reference only) | 73-121 |
| Appendix A . Product Register (PRR) | A-1 |

| | | |
|--|---|-----|
| App.A.1 | Overview..... | A-1 |
| App.A.1.1 | Register Configuration..... | A-1 |
| App.A.2 | Register Description..... | A-1 |
| App.A.2.1 | Product register | A-1 |
| Appendix B . | Sequence of Activation for the RZ/G series, 2nd generation products | B-1 |
| Main Revisions and Additions in this Edition | Revision History-1 | |

1. Overview

1.1 Introduction

The RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E includes:

RZ/G2H

- Four 1.5-GHz Arm® Cortex®-A57 MPCore™ cores,
- Four 1.2-GHz Arm® Cortex®-A53 MPCore™ cores,
- Memory controller for LPDDR4-3200 with 32 bits × 2 channels,
- 1 channel for HDMI1.4b output (option) and 1 channel for RGB888 output and 1channel for LVDS,
- 2 channels MIPI-CSI2 Video Input, 2 channels digital Video Input,
- Serial ATA interface,
- USB3.0 × 1 channel and USB2.0 × 2 channels,

RZ/G2M V1.3 RZ/G2M V3.0

- Two 1.5-GHz Arm® Cortex®-A57 MPCore™ cores,
- Four 1.2-GHz Arm® Cortex®-A53 MPCore™ cores,
- Memory controller for LPDDR4-3200 with 32 bits × 2 channels,
- 1 channel for HDMI1.4b output (option) and 1channel for RGB888 output and 1channel for LVDS,
- 2 channels MIPI-CSI2 Video Input, 2channels digital Video Input,
- USB3.0 × 1 channel and USB2.0 × 2 channels,

RZ/G2N

- Two 1.5-GHz Arm® Cortex®-A57 MPCore™ cores,
- Memory controller for LPDDR4-3200 with 32 bits × 1 channel,
- 1 channel for HDMI1.4b output (option) and 1channel for RGB888 output and 1channel for LVDS,
- 2 channels MIPI-CSI2 Video Input, 2 channels digital Video Input,
- Serial ATA interface,
- USB3.0 × 1 channel and USB2.0 × 2 channels,

RZ/G2E

- Two 1.2-GHz Arm® Cortex®-A53 MPCore™ cores,
- Memory controller for DDR3L-1856 * with 32 bits × 1 channel,
- 1 channel for RGB888 output and 2 channels for LVDS,
- 1 channel MIPI-CSI2 Video Input, 2 channels digital Video Input,
- USB3.0 × 1 channel and USB2.0 × 1 channel,

RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E

- 800-MHz Arm® Cortex®-R7 core (option),
- Three-dimensional graphics engines,
- Sound processing units,
- SD host interface,
- PCI Express interface,
- Video processing units,
- CAN interface, and

- EthernetAVB interface.

* : The “DDR3L-1856” indicates the DDR3L-SDRAM bus interface which operates at a frequency of 928 MHz in this manual.

Note: Arm and Cortex are registered trademark of Arm Limited. All other brands or product names are the property of their respective holders.

Remarks: For items noted as "option", please contact a Renesas Electronics sales representative.

1.2 List of Specifications

1.2.1 Arm Core

| Item | Description |
|---|--|
| System CPU Cortex-A57 RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N | <ul style="list-style-type: none"> • RZ/G2H • Arm Cortex-A57 Quad MPCore 1.5 GHz • L2 cache 2 Mbytes (ECC) • RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N • Arm Cortex-A57 Dual MPCore 1.5 GHz • L2 cache 1 Mbytes (ECC) • RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N • L1 I/D cache 48/32 Kbytes (Parity/ECC) • NEON™/VFPv4 supported • Security extension supported • Virtualization supported • Armv8 architecture |
| System CPU Cortex-A53 RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2E | <ul style="list-style-type: none"> • RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 • Arm Cortex-A53 Quad MPCore 1.2 GHz • L1 I/D cache 32/32 Kbytes (Parity/ECC), L2 cache 512 Kbytes (ECC) • RZ/G2E • Arm Cortex-A53 Dual MPCore 1.2 GHz • L1 I/D cache 32/32 Kbytes (Parity/ECC), L2 cache 256 Kbytes (ECC) • RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2E • NEON™/VFPv4 supported • Security extension supported • Virtualization supported • Armv8 architecture |
| Debug and Trace (For All Products) | <ul style="list-style-type: none"> • JTAG/SWD I/F supported • RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 • ETM-A57/A53 supported (each CPU) • RZ/G2N • ETM-A57 supported • RZ/G2E • ETM-A53 supported • RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E • ETF 16 Kbytes for program flow trace (each cluster) |

1.2.2 CPU Core Peripherals

| Item | Description |
|---|---|
| Clock Pulse Generator (CPG) (For All Products) | <ul style="list-style-type: none"> Generates the clocks from external clock (EXTAL). RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N <ul style="list-style-type: none"> Maximum Cortex-A57 clock: 1.5 GHz RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2E Maximum Cortex-A53 clock: 1.2 GHz RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E Maximum 3DGE clock: 600 MHz RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N Maximum AXI-bus clock: 400 MHz Maximum SDRAM bus clock: 1600 MHz (LPDDR4-3200) Maximum media clock: 400 MHz RZ/G2E <ul style="list-style-type: none"> Maximum AXI-bus clock: 266 MHz Maximum SDRAM bus clock: 928 MHz (DDR3L-1856) Maximum media clock: 266 MHz RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E System-CPU shut down mode control supported Module-standby mode supported Includes module reset registers to control reset operation of individual on-chip peripheral modules |
| System Controller (SYSC) (For All Products) | <ul style="list-style-type: none"> Shuts down and restores power to target modules Target modules: RZ/G2H <ul style="list-style-type: none"> Cortex-A57 (with independent shutting down of CPUs 0, 1, 2, 3, and SCU+L2 cache) * RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N Cortex-A57 (with independent shutting down of CPUs 0, 1 and SCU+L2 cache) * RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 Cortex-A53 (with independent shutting down of CPUs 0, 1, 2, 3, and SCU+L2 cache) * 3DGE VCP RZ/G2E <ul style="list-style-type: none"> Cortex-A53 (with independent shutting down of CPUs 0, 1 and SCU+L2 cache) * *: SCU and L2 cache are treated as one power-domain. When CPU is working, SCU+L2 cache cannot be powered off. Low leakage standby mode supported |
| RESET (For All Products) | <ul style="list-style-type: none"> Includes one reset-signal external output port for external modules Includes Boot Address Register, etc. |

| Item | Description |
|---|--|
| Pin function controller (PFC) (For All Products) | <ul style="list-style-type: none"> Setting multiplexed pin functions for LSI pins Function of the LSI pin selectable by setting the registers in the PFC module Module selection Enable and disable the functions of LSI pins to which pin functions from multiple pin groups are assigned by setting the registers in the PFC module. Pull-up/down control for each LSI pin On/off and up/down of the pull register on each LSI pin can be controlled by setting the registers in the PFC module. Control of SDIO functions SDIO functions, including the driving ability of pins for the SDIF, can be controlled by setting registers of the PFC. |
| General-purpose I/O (GPIO) (For All Products) | <ul style="list-style-type: none"> General-purpose I/O ports Supports GPIO interrupts |
| Thermal sensor / Chip Internal Voltage Monitor (THS/CIVM *) (For All Products) | <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <ul style="list-style-type: none"> 3 channels of thermal sensor <p>RZ/G2E</p> <ul style="list-style-type: none"> 1 channel of thermal sensor <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E</p> <ul style="list-style-type: none"> Programmable 3 temperature level for the sensor, to indicate the temperature level Interrupt when the temperature reaches programmed <p>* : RZ/G2E does not support CIVM.</p> |

1.2.3 External Bus Module

| Item | Description |
|---|--|
| External Bus Controller for EX-Bus (LBSC) (For All Products) | <ul style="list-style-type: none"> EX-BUS interface: max. 16-bit bus RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N Frequency: 66 MHz or 44.4 MHz RZ/G2E Frequency: 66 MHz RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E External area divided into several areas and managed <ul style="list-style-type: none"> Allocation to space of area 0, area 1 Area 0 supports 1-MByte memory space (startup mode). I/F settings, bus width settings, and wait state insertion are possible for each area. SRAM interface <ul style="list-style-type: none"> Wait states can be inserted through register settings. Period of waiting is set in cycle unit, and the maximum value is 15. EX_WAIT pin can be used for wait state insertion. Connectable bus widths: 16 bits or 8 bits Supports external buffer enable/direction control Supports Burst ROM interface Supports Byte-control SRAM interface |
| External Flash Controller (For All Products) | <ul style="list-style-type: none"> Supports RPC-IF (Reduced Pin Count interface) flash memory or QSPI flash memory RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N Max. Frequency 160 MHz (320MB/s) for RPC-IF, 80 MHz (80MB/s)* for QSPI (QSPI0) RZ/G2E Max. Frequency 150 MHz (300MB/s) for RPC-IF, 80 MHz (80MB/s) for QSPI (QSPI0) RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E Dual QSPI operation for two 4-bit serial flash memories is also available; 80 MHz (160MB/s)* for Dual QSPI (QSPI0+QSPI1). Note. *: For RZ/G2M V1.3, 40MB/s for QSPI, 80MB/s for dual QSPI. The RZ/G2M V1.3 does not support QSPI-DDR operation. |

| Item | Description |
|---|---|
| External Bus Controller for LPDDR4/DDR3L SDRAM (DBSC4) (For All Products) | <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <ul style="list-style-type: none"> LPDDR4-3200 can be connected directly. Note. The LPDDR4X (JESD209-4-1) is not supported. <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0</p> <ul style="list-style-type: none"> 2 channels (32-bit bus mode) Memory Size: Up to 8GB * <p>RZ/G2N</p> <ul style="list-style-type: none"> 1 channel (32-bit bus mode) Memory Size: Up to 4GB * <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <p>* : LPDDR4-SDRAM compliant with JEDEC JESD209-4B. (Supports memory with sizes from 4 Gbits to 16 Gbits per channel of DBSC4.)</p> <p>RZ/G2E</p> <ul style="list-style-type: none"> DDR3L-1856 can be connected directly. 1 channel (32-bit bus mode) Memory Size: Up to 2 GB <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E</p> <ul style="list-style-type: none"> Auto Refresh/Self Refresh/Partial Array Self Refresh supported Auto Pre-charge Mode DDR Back Up supported Cache memory for DDR-Memory access efficiency Memory access protection for secure/safety regions Decompression of visual near lossless compressed image ECC supported |

1.2.4 Internal Bus Module

| Item | Description |
|---|--|
| AXI-bus (For All Products) | <ul style="list-style-type: none"> • On-chip main bus <ul style="list-style-type: none"> — Bus protocol: AXI3 with QoS control RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N — Frequency: 400 MHz — Bus width: 512 bits/256 bits/128 bits RZ/G2E — Frequency: 266 MHz — Bus width: 256 bits/128 bits/64 bits • On-chip CPU bus RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 <ul style="list-style-type: none"> — CoreLink™ CCI-Kipling Cache Coherent Interconnect — Bus protocol: AMBA®4 ACE™ and ACE-Lite™ — Frequency: 800 MHz — Bus width: 128 bits RZ/G2N — Bus protocol: AXI3 — Frequency: 800 MHz — Bus width: 128 bits RZ/G2E — Bus protocol: AXI3 — Frequency: 533 MHz — Bus width: 128 bits |
| Direct Memory Access Controller for System (SYS-DMAC) (For All Products) | <ul style="list-style-type: none"> • 16 channels for PeriW domain (SYDM0) • 32 channels for PeriE domain (SYDM1, 2) • Address space: 4 GBytes on architecture • Data transfer length: Byte, word (2 Bytes), longword (4 Bytes), 8 Bytes, 16 Bytes, 32 Bytes and 64 Bytes • Maximum number of transfer times: 16,777,215 times • Transfer request: Selectable from on-chip peripheral module request and auto request • Bus mode: Selectable from normal mode and slow mode • Priority: Selectable from fixed channel priority mode and round-robin mode • Interrupt request: Supports interrupt request to CPU at the end of data transfer • Repeat function: Automatically resets the transfer source, destination, and count at the end of DMA transfer (by descriptor function) • Descriptor function (each channel) supported • MMU (each channel) supported • Channel bandwidth arbiter (each channel) |

| Item | Description |
|--|---|
| Boot (For All Products) | <ul style="list-style-type: none"> System startup with selectable boot mode at power-on reset Either external ROM boot (area 0) or on-chip ROM boot can be selected through MD pin on development chip. In on-chip ROM boot, RPC-IF or QSPI serial ROM boot is supported. Program downloaded to internal memory (System RAM) Autorun function for the downloaded program About detail information of BOOT, refer to Section 24 (Boot) and Appendix B (Active sequence). |
| Direct Memory Access Controller for Audio (Audio-DMAC) (For All Products) | <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <ul style="list-style-type: none"> 32 channels asdm0: 16 channels asdm1: 16 channels <p>RZ/G2E</p> <ul style="list-style-type: none"> 16 channels asdm0: 16 channels <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E</p> <ul style="list-style-type: none"> Address space: 4 GBytes on architecture Data transfer length: Byte, word (2 Bytes), longword (4 Bytes), 8 Bytes, 16 Bytes, 32 Bytes and 64 Bytes Maximum number of transfer times: 16,777,215 times Transfer request: Selectable from on-chip peripheral module request and auto request Bus mode: Selectable from normal mode and slow mode Priority: Selectable from fixed channel priority mode and round-robin mode Interrupt request: Supports interrupt request to CPU at the end of data transfer Repeat function: Automatically resets the transfer source, destination, and count at the end of DMA transfer (by descriptor function) Descriptor function (each channel) supported MMU (each channel) supported Channel bandwidth arbiter (each channel) |
| Audio-DMAC-Peripheral-Peripheral (For All Products) | <ul style="list-style-type: none"> Audio-DMAC (for transfer from Peripheral to Peripheral) <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <ul style="list-style-type: none"> 29 channels + 29 (extended) channels for audio domain <p>RZ/G2E</p> <ul style="list-style-type: none"> 29 (extended) channels for audio domain <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <ul style="list-style-type: none"> Data transfer length: longword (4 Bytes) Transfer count: Transfer count is not specified (DMA transfer is made from the transfer-start to transfer-stop settings.). Transfer request: Selectable from on-chip audio peripheral module request Priority: round-robin mode Interrupt request: not supports interrupt request to CPU at the end of data transfer |

| Item | Description |
|--|---|
| IPMMU (For All Products) | <ul style="list-style-type: none"> An IPMMU is a memory management unit (MMU) which provides address translation and access protection functionalities to processing units and interconnect networks. |
| Interrupt Controller (INTC) (For All Products) | <p>INTC-AP (For All Products; for AP-System core Cortex-A57/Cortex-A53)</p> <ul style="list-style-type: none"> 7 interrupt pins which can detect external interrupts Max. 480 shared peripheral interrupts supported Fall/rise/high level/low level detection is selectable On-chip peripheral interrupts: Priority can be specified for each module 16 software interrupts that have been generated and 6 private peripheral interrupts supported 32-level priority selectable Trust Zone supported |

1.2.5 Internal Memory

| Item | Description |
|----------------------------------|---|
| System RAM (For All Products) | <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RAM of 384 KBytes</p> <p>RZ/G2E RAM of 128 KBytes</p> |

1.2.6 Graphics Units

| Item | Description |
|---------------------------|--|
| 3D Graphics Engine (3DGE) | RZ/G2H |
| | <ul style="list-style-type: none"> Imagination Technologies PowerVR Series6XT GX6650 Max. Frequency. 600 MHz Drastically performance improvements for sophisticated graphics and GPU computer Reducing power consumption even further through advanced power saving mechanisms Lowest memory bandwidth in the industry with compression technologies Ultra HD deep color GPU Support APIs: OpenGL ES 3.1, (OpenCL 1.2 EP) 7.2 Gpix/s, 300 Mpoly/s, 288 GFLOPS |
| | RZ/G2M V1.3 RZ/G2M V3.0 |
| | <ul style="list-style-type: none"> Imagination Technologies PowerVR Series 6XT GX6250 Max. Frequency 600 MHz Drastically performance improvements for sophisticated graphics and GPU computer Reducing power consumption even further through advanced power saving mechanisms Lowest memory bandwidth in the industry with compression technologies Ultra HD deep color GPU Support APIs: OpenGL ES 3.1, (OpenCL 1.2 EP) 2.4 Gpix/s, 300 Mpoly/s, 96 GFLOPS |
| | RZ/G2N |
| | <ul style="list-style-type: none"> Imagination Technologies PowerVR Series 7XE GE7800 Max. Frequency. 600 MHz Drastically performance improvements for sophisticated graphics and GPU computer Reducing power consumption even further through advanced power saving mechanisms Lowest memory bandwidth in the industry with compression technologies Ultra HD deep color GPU Support APIs: OpenGL ES 3.1, (OpenCL 1.2 EP) 2.4 Gpix/s, 200 Mpoly/s, 38.4 GFLOPS |
| | RZ/G2E |
| | <ul style="list-style-type: none"> Imagination Technologies PowerVR Series 8XE GE8300 Max. Frequency. 600 MHz Drastically performance improvements for sophisticated graphics and GPU computer Reducing power consumption even further through advanced power saving mechanisms Lowest memory bandwidth in the industry with compression technologies Ultra HD deep color GPU Support APIs: OpenGL ES 3.1, (OpenCL 1.2 EP) 2.4 Gpix/s, 200 Mpoly/s, 19.2 GFLOPS |

| Item | Description | |
|---|--|--|
| Display Unit (DU) (For All Products) | Display channel | RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N 3 independently controllable channels RZ/G2E 2 independently controllable channels |
| | Interface | RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N <ul style="list-style-type: none"> HDMI 1 channel (option) LVDS 1 channel RZ/G2E <ul style="list-style-type: none"> LVDS 2 channels RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E <ul style="list-style-type: none"> Digital RGB 1 channel (8-bit precision for each RGB color) |
| | LVDS interface (per channel) | <ul style="list-style-type: none"> Output: compliant with TIA/EIA-644; five pairs of differential output (four pairs of data and one pair of clock) Operating frequency: Dotclk 148.5 MHz |
| | HDMI (option) | RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N <ul style="list-style-type: none"> Support HDMI 1.4b class transfer rate, up to 3D format 1080p60/4Kp30 Dotclk 297 MHz |
| | Screen size and number of composite planes per channel | RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N <ul style="list-style-type: none"> Maximum screen size: 3840 × 2160 RZ/G2E <ul style="list-style-type: none"> Maximum screen size: 1920 × 1080 (also depends on frame rate) RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E <ul style="list-style-type: none"> Number of planes specifiable: 5 (VSP2 processing) Number of planes specifiable: 1 (DU) |
| | CRT scanning method | Non-interlaced |
| | Synchronization method | Master |
| | Internal color palette (VSP2) | <ul style="list-style-type: none"> Includes four color palette planes which can display 256 of 260 thousands colors at the same time. |
| | Output display numbers | RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N <ul style="list-style-type: none"> Three output channels (resolutions for different displays) RZ/G2E <ul style="list-style-type: none"> Two output channels RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N <ul style="list-style-type: none"> Output on rising and falling edges of the synchronizing signal (resolution for the same display) RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E <ul style="list-style-type: none"> 8-bit precision for each RGB color |
| | Blending ratio settings (VSP2) | Number of color palette planes with blending ratio: 4 |
| | Dot clock | Switchable between external input and internal clock |

| Item | Description | |
|--|---------------------|---|
| Video Input Module (VIN) (For All Products) | MIPI-CSI2 interface | <div style="display: flex; justify-content: space-between;"> <div style="width: 48%;"> <p>RZ/G2H</p> <ul style="list-style-type: none"> • 2 channels (4 lane × 1 channel, 2 lane × 1 channel) • Interleaving by 4 VC (virtual channel) supported • Filtering by DT (data type) supported • YUV422 8/10bit, RGB888, Embedded 8bit, User Defined 8bit are supported • 1.5 Gbps/Lane </div> <div style="width: 48%;"> <p>RZ/G2M V1.3</p> <p>RZ/G2M V3.0</p> <p>RZ/G2N</p> </div> </div> <div style="display: flex; justify-content: space-between;"> <div style="width: 48%;"> <p>RZ/G2E</p> <ul style="list-style-type: none"> • 1 channel (2 lane × 1 channel) • Interleaving by 2 VC (virtual channel) supported • Filtering by DT (data type) supported • YUV422 8/10bit, RGB888, Embedded 8bit, User Defined 8bit are supported • 1.1 Gbps/Lane </div> <div style="width: 48%;"></div> </div> |

| Item | Description | |
|--|--------------------|--|
| Video Input Module (VIN) (For All Products) | digital interface | RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N <ul style="list-style-type: none"> • 2 channels (RGB/YCbCr) • Dotclk 100 MHz • ITU-R BT.601 interface: 8-, 10- (same size only (not scaling)), 12- (same size only (not scaling)) 16-, 20- (same size only (not scaling)) or 24-bit (same size only (not scaling)), YCbCr422, 18-bit RGB666, 24-bit RGB888 • ITU-R BT.656 interface: 8-, 10- (same size only (not scaling)), or 12-bit (same size only (not scaling)) YCbCr422 • ITU-R BT.1358 interface: 16-, 20- (same size only (not scaling)), or 24-bit (same size only (not scaling)) YCbCr422 • ITU-R BT.709 interface: 8-, 10- (same size only (not scaling)), 12- (same size only (not scaling)) 16-, 20- (same size only (not scaling)) or 24-bit (same size only (not scaling)) YCbCr422, 18-bit RGB666, 24-bit RGB888 <p>About Digital RGB channel usage combination, Refer as follows cases. CASE1 VIN-A8bit + VIN-B8/12/16 CASE2 VIN-A12bit + VIN-B8/12/16 CASE3 VIN-A16bit + VIN-B8/12</p> |
| | | RZ/G2E <ul style="list-style-type: none"> • 2 channels (RGB/YCbCr) • Dotclk 100 MHz • ITU-R BT.601 interface: 8-, 10- (same size only (not scaling)), or 12-bit (same size only (not scaling)) YCbCr422, RGB666, RGB888 • ITU-R BT.656 interface: 8-, 10- (same size only (not scaling)), or 12-bit (same size only (not scaling)) YCbCr422 • ITU-R BT.1358 interface: 16-, 20- (same size only (not scaling)), or 24-bit (same size only (not scaling)) YCbCr422 • ITU-R BT.709 interface: 8-, 10- (same size only (not scaling)), or 12-bit (same size only (not scaling)) YCbCr422, RGB666, RGB888 |
| Video Input Module (VIN) (For All Products) | Capturing function | RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N Up to 8 input images can be captured (using VC, DT filtering) |
| | | RZ/G2E Up to 2 input images can be captured (using VC, DT filtering) |
| | Clipping function | Up to 4096 × 4096 |
| | Horizontal scaling | Up to two times, but only scaling down is possible for HD1080i or HD1080p data. (one input only) |
| | Vertical scaling | Up to three times, but only scaling down is possible for HD1080i or HD1080p data. (one input only) |
| | Output format | RGB-565, ARGB-1555, ARGB8888, YCbCr422, RGB888, YCbCr420 YC separation, and extraction of the Y component |

1.2.7 Video Processing

| Item | Description |
|---|---|
| Video Signal Processor (VSPi) (For All Products) | <p>RZ/G2H</p> <p>VSPi has the following features. 2 sets of VSPi are integrated. 500 Mpix/s process rate per 1 VSPi Supports 4K (3840 pixels x 2160 lines) processing</p> <p>RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <p>VSPi has the following features. 1 set of VSPi is integrated. 500 Mpix/s process rate per 1 VSPi Supports 4K (3840 pixels x 2160 lines) processing</p> <p>RZ/G2E</p> <p>VSPi has the following features. 1 set of VSPi is integrated. 125 Mpix/s process rate per 1 VSPi Supports Full HD (1920 pixels x 1080 lines) processing</p> <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E</p> <p>(1) Supports Various Data Formats and Conversion</p> <ul style="list-style-type: none"> — Supports YCbCr444/422/420, RGB, aRGB, aplane — Color space conversion and changes to the number of colors by dithering — Color keying — Supports combination between pixel alpha and global alpha — Supports generating pre multiplied alpha <p>(2) Video Processing</p> <ul style="list-style-type: none"> — Up and down scaling with arbitrary scaling ratio — Super resolution processing — Image rotation/reversal function: Reverses an image vertically/horizontally or rotates it by 90°/270° <p>(3) Picture Quality/Color Correction with 1D/3D Look Up Table (LUT)</p> <ul style="list-style-type: none"> — Hue, brightness, and saturation adjustment — 1D and 2D histogram <p>Following functions will be supported by Renesas software portfolio.</p> <ul style="list-style-type: none"> — Dynamic γ correction and gain correction — Correction of color (to adjust skin tones or colors in memory) <p>(4) Visual near lossless image compression supported</p> <ul style="list-style-type: none"> — 50% of bandwidth is diminished |

| Item | Description |
|---|---|
| Video Signal Processor (VSPB) (For All Products) | <p>RZ/G2H VSPB has the following features. 2 sets of VSPB are integrated. 500 Mpix/s process rate (output rate) per 1 VSPB. Supports 4K (3840 pixels × 2160 lines) processing</p> <p>RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N VSPB has the following features. 1 set of VSPB is integrated. 500 Mpix/s process rate (output rate) per 1 VSPB. Supports 4K (3840 pixels × 2160 lines) processing</p> <p>RZ/G2E VSPB has the following features. 1 set of VSPB is integrated. 125 Mpix/s process rate (output rate) per 1 VSPB. Supports Full HD (1920 pixels × 1080 lines) processing</p> <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E</p> <p>(1) Supports Various Data Formats and Conversion</p> <ul style="list-style-type: none"> — Supports YCbCr444/422/420, RGB, αRGB, αplane — Color space conversion and changes to the number of colors by dithering — Color keying — Supports combination between pixel alpha and global alpha — Supports generating pre multiplied alpha <p>(2) Video Processing</p> <ul style="list-style-type: none"> — Blending of 5 picture layers and raster operations (ROPs) — Vertical flipping <p>(3) Picture Quality/Color Correction with 1D/3D Look Up Table (LUT)</p> <ul style="list-style-type: none"> — 1D and 2D histogram <p>Following functions will be supported by Renesas software portfolio.</p> <ul style="list-style-type: none"> — Dynamic γ correction and gain correction — Correction of color (to adjust skin tones or colors in memory) <p>(4) Visual near lossless image compression supported</p> <ul style="list-style-type: none"> — 50% of bandwidth is diminished |

| Item | Description |
|--|---|
| Video Signal Processor (VSPD) (For All Products) | <p>RZ/G2H VSPD has the following features. 1 set of VSPD is integrated. Supports 4K (3840 pixels × 2160 lines) resolution.</p> <p>RZ/G2M V1.3 RZ/G2M V3.0 VSPD has the following features. 3 sets of VSPD are integrated. Supports 4K (3840 pixels × 2160 lines) resolution.</p> <p>RZ/G2N VSPD has the following features. 1 set of VSPD is integrated. Supports 4K (3840 pixels × 2160 lines) resolution.</p> <p>RZ/G2E VSPD has the following features. 2 sets of VSPD are integrated. Supports Full HD (1920 pixels × 1080 lines) resolution.</p> <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E</p> <p>(1) Supports Various Data Formats and Conversion</p> <ul style="list-style-type: none"> — Supports YCbCr444/422/420, RGB, αRGB, αplane — Color space conversion and changes to the number of colors by dithering — Color keying — Supports combination between pixel alpha and global alpha — Supports generating pre multiplied alpha <p>(2) Video processing</p> <ul style="list-style-type: none"> — Blending of 5 picture layers and raster operations (ROPs) — Vertical flipping in case of output to memory <p>(3) Direct connection to display module</p> <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <ul style="list-style-type: none"> — Supports 4096 pixels in horizontal direction <p>RZ/G2E</p> <ul style="list-style-type: none"> — Supports 2048 pixels in horizontal direction <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E</p> <ul style="list-style-type: none"> — Writing back image data which is transferred to Display Unit (DU) to memory |
| Video Signal Processor (VSPDL) RZ/G2H RZ/G2N | <p>VSPDL has the following features. 1 set of VSPDL is integrated.</p> <p>(1) Supports various data formats and conversion</p> <ul style="list-style-type: none"> — Supports YCbCr444/422/420, RGB, αRGB, αplane — Color space conversion and changes to the number of colors by dithering — Color keying — Supports combination between pixel alpha and global alpha — Supports generating pre multiplied alpha <p>(2) Video processing</p> <ul style="list-style-type: none"> — Blending of five picture layers and raster operations (ROPs) <p>(3) Direct connection to display module</p> <ul style="list-style-type: none"> — Supports two display output interfaces — Supports 2048 pixels in horizontal direction — Writing back image data which is transferred to Display Unit (DU) to memory |

| Item | Description |
|--|---|
| Video Codec Processor (VCP4) (For All Products) | <p>The VCP4 is a multi-codec module which provides encoding and decoding capabilities on the basis of multiple video coding schemes, e.g., H.265/HEVC, H.264/AVC. This IP (Intellectual Property) is a multi codec that processes the frame or each field by controlling software for VCP4 executed on host CPU.</p> <ul style="list-style-type: none"> The VCP4 has the following features: Support for multiple codecs H.265/HEVC MP (Main Profile) decoding H.264/MPEG-4 AVC HP (High Profile) and MVC SHP (Stereo High Profile) encoding and decoding Support for up to 4K resolutions (H.265 and H.264) Multiple channel processing: RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N When iVDP1C is used: (H.264/H.265 1920 × 1080p × 120 fps) When iVDP1C is not used: (H.265 1920 × 1080p × 120 fps) + (H.264 1920 × 1080p × 120 fps) RZ/G2E When iVDP1C is not used: (H.264/H.265 1920 × 1080p × 60 fps) When iVDP1C is used: (H.264/H.265 1920 × 1080p × 30 fps) RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E Note: “1920 × 1080p × 120 fps” can be replaced as “3840 × 2160p × 30 fps”. “1920 × 1080p × 60 fps” can be replaced as “1280 × 720p × 120 fps”. “1920 × 1080p × 30 fps” can be replaced as “1280 × 720p × 60 fps”. Maximum performance will change with securable bus bandwidth. RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E Data handling on a picture-by-picture basis Encodes/decodes data one picture (frame or field) at a time. High picture quality Supports the H.264 high-efficiency coding tools (CABAC, 8 × 8 frequency conversion, and quantization matrix). High-efficiency motion vector detection by a combination of discrete search and trace search Optimal-mode selection by Rate-Distortion (RD) cost evaluation Picture quality control based on activity analysis results which match visual models Low power dissipation Dynamically disables the clocks for the entire VCP4. Dynamically disables the clocks for individual submodules. Includes its own reference data cache Lossless image compression for reference picture is supported Use the software from Renesas to handle VCP4 functions. |

| Item | Description |
|---|--|
| Video Decoding Processor for inter-device video transfer (iVDP1C) (For All Products) | <ul style="list-style-type: none"> • Low-latency decoder H.264/AVC, JPEG • Color format 4:2:0/4:2:2 • Bit depth 8/10/12bits • Performance: <ul style="list-style-type: none"> RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N 1280 pixels × 960 lines × 30 frames/second × 4 channels (regardless of when VCP4 is used or not) RZ/G2E 1280 pixels × 960 lines × 30 frames/second × 2 channels (when VCP4 is not used) 1280 pixels × 960 lines × 30 frames/second × 1 channel (when VCP4 is used) RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E • Max. resolution: <ul style="list-style-type: none"> 1920 pixels × 1088 lines • Includes its own reference data cache • Lossless image compression for reference picture is supported if bit depth is 8 bits <p>Use the software from Renesas to handle iVDP1C functions.</p> |
| Fine Display Processor (FDP1) (For All Products) | <p>The FDP1 is the de-interlacing module which converts the interlaced video to progressive video, and has the following features.</p> <p>RZ/G2H</p> <p>(1) Supports 2 channels 500 Mpix/s for output performance per 1 FDP1 RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <p>(1) Supports 1 channel 500 Mpix/s for output performance per 1 FDP1 RZ/G2E</p> <p>(1) Supports 1 channel 125 Mpix/s for output performance per 1 FDP1 RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E</p> <p>(2) Supports various data formats</p> <ul style="list-style-type: none"> — Input: YCbCr444/422/420 — Output: YCbCr444/422/420 and RGB/αRGB <p>(3) 8190 pixels x 8190 lines video processing performance</p> <p>(4) High image quality de-interlacing algorithm</p> <ul style="list-style-type: none"> — Motion adaptive de-interlacing — Accurate still detection — Diagonal line interpolation (DLI) <p>(5) Visual near lossless image compression supported 50% of bandwidth is diminished</p> |

1.2.8 Sound Interface

| Item | Description | |
|--|-------------------------------------|---|
| Sampling Rate Converter Unit (SCU) (For All Products) | Overall specification | <ul style="list-style-type: none"> • Includes ten SRC modules <ul style="list-style-type: none"> — Supports the quality suitable for audio sound (High-sound-quality type) (THD+N -132dB): six modules — Supports the quality suitable for voice sound (general-sound-quality type) (THD+N -96dB): four modules • The SRC module is capable of correcting phase change and delay (timing jitter) generated during data transfer over external memories or external devices. • The channel count conversion unit (CTU), mixer (MIX), and digital mute and volume function (DVC) can be used on two fixed output channels. |
| | Sampling rate conversion (SRC) | <ul style="list-style-type: none"> • Capable of asynchronous sampling rate conversion • Supports resolutions up to 24 bits • Two kinds of filter type for SRC. <ul style="list-style-type: none"> — Supports the quality suitable for audio sound (High-sound-quality type) (THD+N -132dB): Realized the filter by passband -1dB@0.4575FS, cutoff -18dB@0.5FS. — Supports the quality suitable for voice sound (general-sound-quality type) (THD+N -96dB): Realized the filter by passband -1dB@0.4561FS, cutoff -72dB@0.5FS. (Characteristics of each filter is written in the equivalent/up-sampling cases.) • Automatically generates antialiasing filter coefficients • For monaural to eight-channel sound sources |
| Sampling Rate Converter Unit (SCU) (For All Products) | Channel count conversion unit (CTU) | <ul style="list-style-type: none"> • Downmixing and splitter functions <ul style="list-style-type: none"> — Conversion of eight input channels into four output channels — Conversion of six input channels into two output channels — Conversion of two input channels into four sets of two output channels — Conversion of one input channel into eight sets of one output channel — No conversion |
| | Mixer (MIX) | <ul style="list-style-type: none"> • Mixing (adds) two to four sources into one • Ratio for adding sources is selectable • Ratio is dynamically changeable • Mixing with volume ramp is available (ramp period is selectable) |

| Item | Description |
|--|--|
| Digital volume and mute function (DVC) | <ul style="list-style-type: none">• Volume control function including digital volume, volume ramp, and zero-crossing mute• The digital volume function is specified by a 24-bit fixed-point value within the range from 0 to 8 times (mute, or -120 to 18 dB)• The volume ramp function can be used for soft mute, fade-in, fade-out, or desired volume adjustment• The volume ramp period can be changed within the sampling range from the 0th to 23rd power of 2• The zero-crossing mute function silences the sound at the zero-crossing point of the audio data |

| Item | Description | |
|--|--|---|
| Serial Sound Interface Unit (SSIU) (For All Products) | Overall specification | <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E</p> <ul style="list-style-type: none"> Includes ten SSI modules functioning as interfaces with external devices. <ul style="list-style-type: none"> — Supports short and long formats for monaural — Supports TDM format (six modules of ten modules can be used for this function) <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <ul style="list-style-type: none"> Max. 16 independent monaural sound sources in a TDM format can be in TDM format. <p>RZ/G2E</p> <ul style="list-style-type: none"> Max. 8 independent monaural sound sources in a TDM format can be in TDM format. |
| | Serial sound interface (SSI) | <ul style="list-style-type: none"> 10 channels Max. SCK frequency 15.1 MHz (for slave input) or 12.5 MHz (for master output) Operating mode: non-compressed mode (Not support compressed mode) Supports versatile serial audio formats (I2S/left justified/right justified) Supports master/slave functions Programmable word clock, bit clock generation functions Multichannel format functions (up to four channels) Supports 8-/16-/18-/20-/22-/24-bit data formats Supports TDM mode Supports WS continue mode The DMA controller or interrupts control the transfer of data to and from the SSI module. Supports short and long frames for monaural data (valid data lengths are 8 and 16 bits) Up to nine independent clock signals can be input. |
| Audio Clock Generator (ADG) (For All Products) | Selection or division of audio clock signals | |

1.2.9 Storage

| Item | Description | | | | | | | | | | | | | | | |
|---|--|------------|-------------|------------|------------|-----------|------|---|---|---|---|------------|---|---|---|---|
| USB2.0 Host (For All Products) | <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <ul style="list-style-type: none"> • 2 channels (Host only 1 channel/Host-Function 1 channel) • USB Host (EHCI/OHCI) 2 LINK <p>RZ/G2E</p> <ul style="list-style-type: none"> • 1 channel (Host-Function 1 channel) • USB Host (EHCI/OHCI) 1 LINK <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <ul style="list-style-type: none"> • USB3.0 module also can be used as USB2.0 • Compliance with USB2.0 • USB Function 1 LINK • Supports On-The-Go (OTG) function Rev2.0 complying with 2 protocols: <ul style="list-style-type: none"> + Session Request Protocol (SRP). + Host Negotiation Protocol (HNP). <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E</p> <ul style="list-style-type: none"> • Compliance with USB2.0 (High-Speed) • Interrupt request • Internal dedicated DMA • Compliance with Battery Charging function Rev1.2: <ul style="list-style-type: none"> + Charging Port (Host): CDP, SDP are supported (Not support DCP). + Portable Device (Function) is supported. | | | | | | | | | | | | | | | |
| USB3.0 (For All Products) | <ul style="list-style-type: none"> • USB 3.0 DRD 1 channel • This module can be use as USB2.0 as follows <table border="1" data-bbox="491 1240 1426 1361"> <thead> <tr> <th>Core</th> <th>Super Speed</th> <th>High Speed</th> <th>Full Speed</th> <th>Low Speed</th> </tr> </thead> <tbody> <tr> <td>Host</td> <td>√</td> <td>√</td> <td>√</td> <td>√</td> </tr> <tr> <td>Peripheral</td> <td>√</td> <td>√</td> <td>√</td> <td>—</td> </tr> </tbody> </table> • Supports SS/HS/FS/LS. xHCI | Core | Super Speed | High Speed | Full Speed | Low Speed | Host | √ | √ | √ | √ | Peripheral | √ | √ | √ | — |
| Core | Super Speed | High Speed | Full Speed | Low Speed | | | | | | | | | | | | |
| Host | √ | √ | √ | √ | | | | | | | | | | | | |
| Peripheral | √ | √ | √ | — | | | | | | | | | | | | |
| Serial-ATA Gen3 RZ/G2H RZ/G2N | <ul style="list-style-type: none"> • Serial ATA Standard Rev3.2 supported • 6.0-Gbps (Gen3) transfer rate supported PHY is shared with PCIE Controller | | | | | | | | | | | | | | | |

| Item | Description |
|---|---|
| SD host Interface (For All Products) | <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <ul style="list-style-type: none"> 4 channels <p>RZ/G2E</p> <ul style="list-style-type: none"> 3 channels <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E</p> <ul style="list-style-type: none"> Supports SDR104 class transfer rate Does not support CPRM Supports SD memory/SDIO interface Error check function: CRC7 (command/response), CRC16 (data) Max. Frequency 200 MHz Card detection function Supports write protection SD-binding function <ul style="list-style-type: none"> Compliant with Content Protection for Recordable Media Specification in revision 0.92 of the SD-Binding Part of the SD Memory Card Book SD-SD content protection <ul style="list-style-type: none"> Compliant with Content Protection for Recordable Media Specification in revision 0.92 of the SD-SD part of the SD Memory Card Book |
| Multimedia Card Interface (MMC) (For All Products) | <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <ul style="list-style-type: none"> 2 channels eMMC 5.0 base, Support HS400 class transfer rate <p>RZ/G2E</p> <ul style="list-style-type: none"> 1 channel eMMC 5.0 base, Support HS400 class transfer rate <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E</p> <ul style="list-style-type: none"> eMMC controllable Data bus: 1/4/8-bit MMC mode (not support SPI mode) Support block transfer (not support stream transfer) Block size in multiple block transfer: 512 Bytes |
| rawNAND Controller | <ul style="list-style-type: none"> The NAND Flash Memory Interface controller implements the function of a high level interface to one NAND flash device. It supports the functionality of the high speed NAND Flash devices described in the ONFI 1.x specifications. |

1.2.10 Network

| Item | Description |
|---|--|
| Controller Area Network Interface (CAN interface) (For All Products) | <ul style="list-style-type: none"> • 2 channels • Supports CAN specification 2.0B • Maximum bit rate: 1 Mbps • Message box <ul style="list-style-type: none"> — Normal mode: 32 receive-only mailboxes and 32 mailboxes for transmission/reception — FIFO mode: <ul style="list-style-type: none"> 32 receive-only mailboxes and 24 mailboxes for transmission/reception, 4-stage FIFO for transmission, and 4-stage FIFO for reception • Reception <ul style="list-style-type: none"> — Data frame and remote frame can be received. — Selectable receiving ID format — Selectable overwrite mode (message overwritten) or overrun mode (message discarded) • Acceptance filter <ul style="list-style-type: none"> — Mask can be enabled or disabled for each mailbox. • Transmission <ul style="list-style-type: none"> — Data frame and remote frame can be transmitted. — Selectable transmitting ID format (only standard ID, only extended ID, or both IDs) — Selectable ID priority mode or mailbox number priority mode • Sleep mode for reducing power consumption |
| CAN-FD (For All Products) | <ul style="list-style-type: none"> • 2 channels • 8 Mbps (CAN clock 40 MHz) |
| PCIE Controller (For All Products) | <ul style="list-style-type: none"> • PCI Express Base Specification Revision 2.0 • PHY integrated <ul style="list-style-type: none"> RZ/G2H RZ/G2N • 1 Lane x 2 channels (one of PHY is shared with Serial ATA) <ul style="list-style-type: none"> RZ/G2M V1.3 RZ/G2M V3.0 • 1 Lane x 2 channels <ul style="list-style-type: none"> RZ/G2E • 1 Lane x 1 channel |
| EthernetAVB-IF (For All Products) | <ul style="list-style-type: none"> • Supports IEEE802.1BA, IEEE802.1AS, IEEE802.1Qav and IEEE1722 functions • Magic packet detection • Supports Reception Filtering to separate streaming frames from different sources • Supports interface conforming to IEEE802.3 PHY RGMII (Reduced Gigabit Media Independent Interface) • RGMII v1.3 • Supports transfer at 1000 Mbps and 100 Mbps. |

1.2.11 Timer

| Item | Description |
|--|--|
| RCLK Watchdog Timer (For All Products) | <ul style="list-style-type: none"> • 1 channel • Internal 16-bit watchdog timer operated by RCLK • Programmable overflow time period: more than 1 hour count capable |
| 16-Bit Timer Pulse Unit (TPU) (For All Products) | <ul style="list-style-type: none"> • 4-channels 16-bit timers • Each channel outputs PWM |
| System Watchdog Timer (For All Products) | <ul style="list-style-type: none"> • 1 channel • Internal 16-bit watchdog timer • Programmable overflow time period: more than 1 hour count capable initial counter value 171[s] |
| Compare Match Timer Type0 (CMT0) (For All Products) | <ul style="list-style-type: none"> • 2 channels • 32-bit timer (16 bits/32 bits can be selected) • Source clock: RCLK clock • Compare match function provided • Interrupt requests |
| Compare Match Timer Type1 (CMT1) (For All Products) | <ul style="list-style-type: none"> • 8 channels • 48-bit timer (16 bits/32 bits/48 bits can be selected) • Source clock: RCLK/system clock • Compare match function provided • Interrupt requests |
| Compare match timer 2 (CMT2) (For All Products) | (same as CMT1) |
| Compare match timer 3 (CMT3) (For All Products) | (same as CMT1) |
| System Timer (For All Products) | <ul style="list-style-type: none"> • 32-bit timer, 1 channel (16 bits/32 bits can be selected) • Compare match function provided • Interrupt requests |
| System up-time clock (For All Products) | <ul style="list-style-type: none"> • 1 channel • Internal 32-bit timer • Programmable overflow time period: maximum 24 hours |
| Timer Unit (TMU) (For All Products) | <ul style="list-style-type: none"> • 15 channels • 32-bit timer • Auto-reload type 32-bit down counter • Internal prescaler • Interrupt request • 2 channels for input capture |

1.2.12 Peripheral Module

| Item | Description |
|---|---|
| IIC Bus Interface for PMIC (IIC for PMIC) (For All Products) | <ul style="list-style-type: none"> • 1 PMIC channel for dedicated buffer • Supports single master transmission/reception • Interrupt request • Automatic transfer by wakeup control |
| I2C Bus Interface (I2C) (For All Products) | <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N</p> <ul style="list-style-type: none"> • 7 channels • 4 channels for buffers with a slew rate (channel 0, 3, 4, 5 for dedicated buffers) • 3 channels for LVTTTL buffers (channels 1, 2, 6 for ordinary buffers) <p>RZ/G2E</p> <ul style="list-style-type: none"> • 8 channels • 2 channels for buffers with a slew rate (channel 0, 3 for dedicated buffers) • 6 channels for LVTTTL buffers (channels 1, 2, 4, 5, 6, 7 for ordinary buffers) <p>RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E</p> <ul style="list-style-type: none"> • NXP I2C bus interface method supported • Master/slave functions • Multi-master functions • Transfer rate up to 400 kbps supported • Programmable clock generation from the system clock • Master and Slave function DMA supported |

| Item | Description | |
|---|-----------------------|---|
| Serial communication interface with FIFO (SCIF) (For All Products) | Overall specification | <ul style="list-style-type: none"> • 6 channels • Asynchronous, clock-synchronized modes • Asynchronous serial communication mode <p>The SCIF performs serial data communication based on a character-by-character asynchronous system. This feature enables serial data communication with standard asynchronous communication chips that support Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communication Interface Adapter (ACIA). There is a choice of eight serial data transfer formats.</p> <ul style="list-style-type: none"> — Data length: 7 bits or 8 bits — Stop bits: 1 bit or 2 bits — Parity: Even/odd/none — Receive error detection: Parity, framing, and overrun errors — Break detection: <p>A break is detected when a framing error lasts for more than 1 frame length at Space 0 (low level).</p> <p>When a framing error occurs, a break can also be detected by reading the RX pin level directly from the serial port register (SCSPTR).</p> • Clock synchronous serial communication mode <p>The SCIF performs serial data communication synchronized with a clock. This feature enables serial data communication with other LSIs that support synchronous communication. There is a single serial data communication format for clock synchronous serial communication.</p> <ul style="list-style-type: none"> — Data length: 8 bits — Receive error detection: Overrun errors • Full-duplex communication capability <p>The SCIF has an independent transmitter and receiver that enable simultaneous transmission and reception. The transmitter and receiver both have a 16-stage FIFO buffer structure, enabling continuous serial data transmission and reception.</p> • On-chip baud rate generator, enabling any bit rate to be selected <p>The SCIF enables choice of a clock source for transmission/reception: a clock from the on-chip baud rate generator based on the internal clock or an external clock.</p> • Eight interrupt sources <p>The SCIF has eight types of interrupt sources. receive-data-ready, receive-FIFO-data-full, break, transmit-FIFO-data-empty, transmit-end, receive-error, overrun-error and time-out and enables any of them to be requested independently.</p> • DMA data transfer <p>When the transmit FIFO register is empty or the receive FIFO register has received data, issuing a DMA transfer request activates the DMA controller (DMAC) to execute a data transfer.</p> • In asynchronous mode using channels 0, 1, 3, and 4, modem control functions (RTS and CTS) are stored. • RTS and CTS are not implemented for SCIF2 and SCIF5. |

| Item | Description |
|---|---|
| Serial Communication Interface with FIFO (SCIF) (For All Products) | <p>Overall specification</p> <ul style="list-style-type: none"> The amount of data in the transmit/receive FIFO registers and the number of receive errors in receive data in the receive FIFO register are available. In asynchronous mode, a receive data ready (DR) or a timeout error (TO) can be detected during reception. |
| Clock-Synchronized Serial Interface with FIFO (MSIOF) (For All Products) | <ul style="list-style-type: none"> 4 channels Internal 32-bit × 64-stage transmit FIFOs/internal 32-bit × 256-stage receive FIFOs Supports master and slave modes Internal prescaler <p>RZ/G2H</p> <ul style="list-style-type: none"> Supports serial formats: SPI (master and slave modes) <p>RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E</p> <ul style="list-style-type: none"> Supports serial formats: IIS, SPI (master and slave modes) Interrupt request, DMAC request |
| High Speed Serial Communication Interface with FIFO (HSCIF) (For All Products) | <ul style="list-style-type: none"> 5 channels Asynchronous serial communication mode Capable of full-duplex communication On-chip baud rate generator, enabling any bit rate to be selected Eight interrupt sources DMA data transfer Modem control functions (HRTS# and HCTS#) are stored. The amount of data in the transmit/receive FIFO registers and the number of receive errors in receive data in the receive FIFO register are available. A receive data ready (DR) or a timeout error (TO) can be detected during reception. |
| PWM (For All Products) | <ul style="list-style-type: none"> 7 channels High-level width (10 bits) of PWM output can be set. Output cycle periods (10 bits) of PWM can be set. Periods in the range from two to $2^{24} \times 1023$ cycles of the Pϕ clock can be set. Continuous pulse or single pulse output selectable |

1.2.13 Others

| Item | Description |
|---|---|
| Boundary Scan (For All Products: option) | <ul style="list-style-type: none"> Boundary scan based on IEEE 1149.1 via JTAG interface is supported. Note that some module pins are not available on this boundary scan. |

2. Area Map

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

[Legend]

√: Support for the product

—: Not support for the product (the area is equivalent to a reserved for the product)

Table 2.1 Address Space (Over 4-Gbyte Space, RZ/G2H)

| Address | Default mode (Gen2) | Gen1 mode | Second Generation RZ/G Series Products | | |
|----------------------------------|------------------------|------------------------|--|-------|-----------|
| | | | RZ/G2H (memory area) | | |
| | | | no-swap* | swap* | Gen1 mode |
| H'00_0000_0000 to H'00_FFFF_FFFF | Legacy (DDR0 shadow A) | Legacy (DDR0 shadow A) | √ | √ | √ |
| H'01_0000_0000 to H'01_FFFF_FFFF | DDR1 shadow A | DDR0 shadow A | — | √ | √ |
| H'02_0000_0000 to H'02_FFFF_FFFF | DDR2 shadow A | DDR1 shadow A | √ | — | — |
| H'03_0000_0000 to H'03_FFFF_FFFF | DDR3 shadow A | DDR1 shadow D | — | — | — |
| H'04_0000_0000 to H'04_FFFF_FFFF | DDR0 | DDR0 | √ | √ | √ |
| H'05_0000_0000 to H'05_FFFF_FFFF | DDR1 | DDR1 | — | √ | — |
| H'06_0000_0000 to H'06_FFFF_FFFF | DDR2 | DDR2 | √ | — | √ |
| H'07_0000_0000 to H'07_FFFF_FFFF | DDR3 | DDR3 | — | — | — |
| H'08_0000_0000 to H'08_FFFF_FFFF | DDR0 shadow B | DDR0 shadow B | √ | √ | √ |
| H'09_0000_0000 to H'09_FFFF_FFFF | DDR1 shadow B | DDR1 shadow B | — | √ | — |
| H'0A_0000_0000 to H'0A_FFFF_FFFF | DDR2 shadow B | DDR2 shadow B | √ | — | √ |
| H'0B_0000_0000 to H'0B_FFFF_FFFF | DDR3 shadow B | DDR3 shadow B | — | — | — |
| H'0C_0000_0000 to H'0C_FFFF_FFFF | DDR0 shadow C | DDR0 shadow C | √ | √ | √ |
| H'0D_0000_0000 to H'0D_FFFF_FFFF | DDR1 shadow C | DDR1 shadow C | — | √ | — |
| H'0E_0000_0000 to H'0E_FFFF_FFFF | DDR2 shadow C | DDR2 shadow C | √ | — | √ |
| H'0F_0000_0000 to H'0F_FFFF_FFFF | DDR3 shadow C | DDR3 shadow C | — | — | — |
| H'10_0000_0000 to H'10_FFFF_FFFF | CCI-Kipling (legacy) | CCI-Kipling (legacy) | √ | √ | √ |
| H'11_0000_0000 to H'11_FFFF_FFFF | CCI-Kipling (DDR0) | CCI-Kipling (DDR0) | √ | √ | √ |
| H'12_0000_0000 to H'12_FFFF_FFFF | CCI-Kipling (DDR1) | CCI-Kipling (DDR1) | — | √ | — |
| H'13_0000_0000 to H'13_FFFF_FFFF | CCI-Kipling (DDR2) | CCI-Kipling (DDR2) | √ | — | √ |
| H'14_0000_0000 to H'14_FFFF_FFFF | CCI-Kipling (DDR3) | CCI-Kipling (DDR3) | — | — | — |
| H'15_0000_0000 to H'17_FFFF_FFFF | CCI-Kipling | CCI-Kipling | √ | √ | √ |
| H'18_0000_0000 to H'1B_FFFF_FFFF | uTLB (IPMMU slave) ch0 | uTLB (IPMMU slave) ch0 | √ | √ | √ |
| H'1C_0000_0000 to H'1F_FFFF_FFFF | uTLB (IPMMU slave) ch1 | uTLB (IPMMU slave) ch1 | √ | √ | √ |
| H'20_0000_0000 to H'2F_FFFF_FFFF | IPMMU/ICB | IPMMU/ICB | √ | √ | √ |
| H'30_0000_0000 to H'FE_FFFF_FFFF | Reserved | Reserved | √ | √ | √ |
| H'FF_0000_0000 to H'FF_FFFF_FFFF | Reserved | Reserved | √ | √ | √ |

Note. Memory area no-swap mode: DRAM Linear Address mapping (AXI-bus) can be performed.

Memory area swap mode: DRAM Split Address mapping (2CH) (AXI-bus) can be performed by Memory Channel Swap Mode (DBSC4).

Table 2.2 Address Space (Over 4-Gbyte Space, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E)

| Address | Default mode (Gen2) | Gen1 mode | Second Generation RZ/G Series Products | | |
|----------------------------------|------------------------|------------------------|--|--------|--------|
| | | | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| H'00_0000_0000 to H'00_FFFF_FFFF | Legacy (DDR0 shadow A) | Legacy (DDR0 shadow) | √ | √ | √ |
| H'01_0000_0000 to H'01_FFFF_FFFF | Reserved | DDR0 shadow A | √ | √ | — |
| H'02_0000_0000 to H'02_FFFF_FFFF | DDR1 shadow A | DDR1 shadow A | √ | — | — |
| H'03_0000_0000 to H'03_FFFF_FFFF | Reserved | DDR1 shadow D | √ | — | — |
| H'04_0000_0000 to H'04_FFFF_FFFF | DDR0 | DDR0 | √ | √ | √ |
| H'05_0000_0000 to H'05_FFFF_FFFF | Reserved | Reserved | √ | √ | √ |
| H'06_0000_0000 to H'06_FFFF_FFFF | DDR1 | DDR1 | √ | — | — |
| H'07_0000_0000 to H'07_FFFF_FFFF | Reserved | Reserved | √ | √ | √ |
| H'08_0000_0000 to H'08_FFFF_FFFF | DDR0 shadow B | DDR0 shadow B | √ | √ | √ |
| H'09_0000_0000 to H'09_FFFF_FFFF | Reserved | Reserved | √ | √ | √ |
| H'0A_0000_0000 to H'0A_FFFF_FFFF | DDR1 shadow B | DDR1 shadow B | √ | — | — |
| H'0B_0000_0000 to H'0B_FFFF_FFFF | Reserved | Reserved | √ | √ | √ |
| H'0C_0000_0000 to H'0C_FFFF_FFFF | DDR0 shadow C | DDR0 shadow C | √ | √ | √ |
| H'0D_0000_0000 to H'0D_FFFF_FFFF | Reserved | Reserved | √ | √ | √ |
| H'0E_0000_0000 to H'0E_FFFF_FFFF | DDR1 shadow C | DDR1 shadow C | √ | — | — |
| H'0F_0000_0000 to H'0F_FFFF_FFFF | Reserved | Reserved | √ | √ | √ |
| H'10_0000_0000 to H'10_FFFF_FFFF | CCI-Kipling (legacy) | CCI-Kipling (legacy) | √ | — | — |
| H'11_0000_0000 to H'11_FFFF_FFFF | CCI-Kipling (DDR0) | CCI-Kipling (DDR0) | √ | — | — |
| H'12_0000_0000 to H'12_FFFF_FFFF | CCI-Kipling (DDR1) | CCI-Kipling (DDR1) | √ | — | — |
| H'13_0000_0000 to H'13_FFFF_FFFF | CCI-Kipling (DDR2) | CCI-Kipling (DDR2) | √ | — | — |
| H'14_0000_0000 to H'14_FFFF_FFFF | CCI-Kipling (DDR3) | CCI-Kipling (DDR3) | √ | — | — |
| H'15_0000_0000 to H'17_FFFF_FFFF | CCI-Kipling | CCI-Kipling | √ | — | — |
| H'18_0000_0000 to H'1B_FFFF_FFFF | uTLB (IPMMU slave) ch0 | uTLB (IPMMU slave) ch0 | √ | √ | √ |
| H'1C_0000_0000 to H'1F_FFFF_FFFF | uTLB (IPMMU slave) ch1 | uTLB (IPMMU slave) ch1 | √ | — | — |
| H'20_0000_0000 to H'2F_FFFF_FFFF | IPMMU/ICB | IPMMU/ICB | √ | — | — |
| H'30_0000_0000 to H'FE_FFFF_FFFF | Reserved | Reserved | √ | √ | √ |
| H'FF_0000_0000 to H'FF_FFFF_FFFF | Reserved | Reserved | √ | √ | √ |

Table 2.3 Address Space (Legacy 4-Gbyte Space)

| Address | Space | Description | Second Generation RZ/G Series Products | | | | |
|----------------------------------|-------------------|-----------------------------------|--|-------------|-------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| H'00_0000_0000 to H'00_03FF_FFFF | LBSC | Area0 asynchronous memory | √ | √ | √ | √ | √ |
| H'00_0400_0000 to H'00_07FF_FFFF | | Area1 asynchronous memory | √ | √ | √ | √ | √ |
| H'00_0800_0000 to H'00_0BFF_FFFF | | RPC-IF memory space | √ | √ | √ | √ | √ |
| H'00_0C00_0000 to H'00_0FFF_FFFF | | Reserved | √ | √ | √ | √ | √ |
| H'00_1000_0000 to H'00_13FF_FFFF | | Reserved | √ | √ | √ | √ | √ |
| H'00_1400_0000 to H'00_15FF_FFFF | | Reserved | √ | √ | √ | √ | √ |
| H'00_1600_0000 to H'00_17FF_FFFF | | Reserved | √ | √ | √ | √ | √ |
| H'00_1800_0000 to H'00_19FF_FFFF | | Reserved | √ | √ | √ | √ | √ |
| H'00_1A00_0000 to H'00_1BFF_FFFF | | Reserved | √ | √ | √ | √ | √ |
| H'00_1C00_0000 to H'00_1FFF_FFFF | | Reserved | √ | √ | √ | √ | √ |
| H'00_2000_0000 to H'00_2FFF_FFFF | | Reserved | √ | √ | √ | √ | √ |
| H'00_3000_0000 to H'00_37FF_FFFF | PCIEC | PCIE0 memory2 | √ | √ | √ | √ | √ |
| H'00_3800_0000 to H'00_3FFF_FFFF | | PCIE0 memory3 | √ | √ | √ | √ | √ |
| H'00_4000_0000 to H'00_5FFF_FFFF | DBSC4 | SDRAM Area | √ | √ | √ | √ | √ |
| H'00_6000_0000 to H'00_7FFF_FFFF | | (over4G mirror space) | √ | √ | √ | √ | √ |
| H'00_8000_0000 to H'00_BFFF_FFFF | | | √ | √ | √ | √ | √ |
| H'00_C000_0000 to H'00_C7FF_FFFF | PCIEC | PCIE1 memory2 | √ | √ | √ | √ | — |
| H'00_C800_0000 to H'00_CFFF_FFFF | | PCIE1 memory3 | √ | √ | √ | √ | — |
| H'00_D000_0000 to H'00_DFFF_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_E000_0000 to H'00_E3FF_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_E400_0000 to H'00_E4FF_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_E500_0000 to H'00_E54F_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_E550_0000 to H'00_E56F_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_E570_0000 to H'00_E58F_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_E590_0000 to H'00_E5FF_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_E600_0000 to H'00_E61F_FFFF | C5 APB | System-Domain Peripheral bus (CP) | √ | √ | √ | √ | √ |
| H'00_E620_0000 to H'00_E62F_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_E630_0000 to H'00_E63F_FFFF | DBSC4 | System RAM space | √ | √ | √ | √ | √ |
| H'00_E640_0000 to H'00_E642_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_E643_0000 to H'00_E64F_FFFF | HC-APB | AXHC Register Space | √ | √ | √ | √ | √ |
| H'00_E650_0000 to H'00_E656_FFFF | PeriE APB (APSE2) | HSCIF, I2C | √ | √ | √ | √ | √ |
| H'00_E6570000 to H'00_E65F_FFFF | HC-APB | USB register space | √ | √ | √ | √ | √ |
| | | PCIEC register space | √ | √ | √ | √ | √ |
| | | SATA register space | √ | — | √ | — | — |
| H'00_E660_0000 to H'00_E664_FFFF | RC Hier APB | (secure) | √ | √ | √ | √ | √ |

| | | | | Second Generation RZ/G Series Products | | | | |
|----------------|----|----------------|------------------------|--|----------------------------|--------|--------|--|
| Address | | Space | Description | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| H'00_E665_0000 | to | H'00_E667_FFFF | Reserved | √ | √ | √ | √ | |
| H'00_E668_0000 | to | H'00_E677_FFFF | PeriW APB (APSW2) | √ | √ | √ | √ | |
| H'00_E678_0000 | to | H'00_E67F_FFFF | DDR Hier | √ | √ | √ | √ | |
| H'00_E680_0000 | to | H'00_E6A1_FFFF | PeriW (EthernetAVB-IF) | √ | √ | √ | √ | |
| H'00_E6B0_0000 | to | H'00_E6B2_FFFF | Reserved | √ | √ | √ | √ | |
| H'00_E6C0_0000 | to | H'00_E6E3_FFFF | PeriW APB | √ | √ | √ | — | |
| H'00_E6E4_0000 | to | H'00_E6E8_FFFF | PeriE APB (APSE1) | √ | √ | √ | √ | |
| H'00_E6E9_0000 | to | H'00_E6EA_FFFF | PeriE APB (APSE3) | √ | √ | √ | √ | |
| H'00_E6EB_0000 | to | H'00_E6EB_FFFF | Reserved | √ | √ | √ | √ | |
| H'00_E6EC_0000 | to | H'00_E6EC_FFFF | Reserved | √ | √ | √ | √ | |
| H'00_E6ED_0000 | to | H'00_E6EE_FFFF | Reserved | √ | √ | √ | √ | |
| H'00_E6EF_0000 | to | H'00_E6EF_FFFF | VIO-APB | √ | √ | √ | √ | |
| H'00_E6F0_0000 | to | H'00_E6FF_FFFF | PeriE | √ | √ | √ | √ | |
| H'00_E700_0000 | to | H'00_E790_FFFF | | √ | √ | √ | √ | |
| H'00_E791_0000 | to | H'00_E79F_FFFF | Reserved | √ | √ | √ | √ | |
| H'00_E7A0_0000 | to | H'00_E7BF_FFFF | Reserved | √ | √ | √ | √ | |
| H'00_E7C0_0000 | to | H'00_E7FF_FFFF | Reserved | √ | √ | √ | √ | |
| H'00_E800_0000 | to | H'00_E807_FFFF | ICB | √ | √ | √ | — | |
| H'00_E808_0000 | to | H'00_E827_FFFF | Reserved | √ | √ | √ | √ | |
| H'00_E828_0000 | to | H'00_E8FF_FFFF | Reserved | √ | √ | √ | √ | |

**Second Generation
RZ/G Series
Products**

| Address | Space | Description | RZ/G2H | RZ/G2M V1.3 RZG2M V3.0 | RZ/G2N | RZ/G2E |
|----------------------------------|---------------------|-----------------------------|--------|---------------------------|--------|--------|
| H'00_E900_0000 to H'00_E9FF_FFFF | CDBGTOP STM | CoreSight STM | √ | √ | √ | √ |
| H'00_EA00_0000 to H'00_EAFF_FFFF | CoreSightAPB Bridge | 16MB space | √ | √ | √ | √ |
| H'00_EB00_0000 to H'00_EB0F_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EB10_0000 to H'00_EB10_FFFF | (on-chip memory) | | √ | √ | √ | √ |
| H'00_EB11_0000 to H'00_EB11_FFFF | | | √ | √ | √ | √ |
| H'00_EB12_0000 to H'00_EB12_FFFF | | | √ | — | √ | √ |
| H'00_EB13_0000 to H'00_EB13_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EB14_0000 to H'00_EB1F_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EB20_0000 to H'00_EB2F_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EB30_0000 to H'00_EB3F_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EB40_0000 to H'00_EB7F_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EB80_0000 to H'00_EBBF_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EBC0_0000 to H'00_EBFF_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EC00_0000 to H'00_EC0F_FFFF | MP-APB1 | SRC/CMD | √ | √ | √ | √ |
| H'00_EC10_0000 to H'00_EC1F_FFFF | MP-APB2 | SSI | √ | √ | √ | √ |
| H'00_EC20_0000 to H'00_EC2F_FFFF | MP-APB3 | SSI0-2 | √ | √ | √ | √ |
| | | SSI3-4 | √ | √ | √ | √ |
| | | SSI5-9 | √ | √ | √ | √ |
| H'00_EC30_0000 to H'00_EC3F_FFFF | MP-APB4 | SRC/CMD | √ | √ | √ | √ |
| H'00_EC40_0000 to H'00_EC4F_FFFF | MP-APB5 | SSI | √ | √ | √ | √ |
| H'00_EC50_0000 to H'00_EC5F_FFFF | MP-APB6 | SRC/CMD/SSI/ADG | √ | √ | √ | √ |
| H'00_EC60_0000 to H'00_EC9F_FFFF | MP-APB7 | Audio-DMAC/ Audio-DMACpp | √ | √ | √ | √ |
| H'00_ECA0_0000 to H'00_ECBF_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_ECC0_0000 to H'00_ECDF_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_ECE0_0000 to H'00_ECFE_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_ED00_0000 to H'00_EDFF_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EE00_0000 to H'00_EE01_FFFF | USB3 | USB3.0 ch0 Host | √ | √ | √ | √ |
| H'00_EE02_0000 to H'00_EE03_FFFF | USB3-0 | USB3.0 ch0 Peripheral | √ | √ | √ | √ |
| H'00_EE04_0000 to H'00_EE05_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EE06_0000 to H'00_EE07_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EE08_0000 to H'00_EE09_FFFF | USB2-0 | USB (EHCI) | √ | √ | √ | √ |
| H'00_EE0A_0000 to H'00_EE0B_FFFF | USB2-1 | | √ | √ | √ | — |
| H'00_EE0C_0000 to H'00_EE0D_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EE0E_0000 to H'00_EE0F_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EE10_0000 to H'00_EE1F_FFFF | SDHI0-3/ eMMC0-1 | SDHI/eMMC | √ | √ | √ | √ |

**Second Generation
RZ/G Series
Products**

| Address | Space | Description | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|----------------------------------|---------------------------------------|--|--------|----------------------------|--------|--------|
| H'00_EE20_0000 to H'00_EE2F_FFFF | PeriW (RPC/BSC) | RPC-IF register space LBSC register space | √ | √ | √ | √ |
| H'00_EE30_0000 to H'00_EE4F_FFFF | SATA0 | SATA | √ | — | √ | — |
| H'00_EE50_0000 to H'00_EE6F_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EE70_0000 to H'00_EE7F_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_EE80_0000 to H'00_EE8F_FFFF | PCIEC | PCIE1 Config | √ | √ | √ | — |
| H'00_EE90_0000 to H'00_EE9F_FFFF | | PCIE1 Memory0 | √ | √ | √ | — |
| H'00_EEA0_0000 to H'00_EEBF_FFFF | | PCIE1 Memory1 | √ | √ | √ | — |
| H'00_EEC0_0000 to H'00_EFFF_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_F000_0000 to H'00_F000_1FFF | Reserved | | √ | √ | √ | √ |
| H'00_F000_2000 to H'00_F000_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_F001_0000 to H'00_F0FF_FFFF | Reserved | (H'F010_0000 to H'F010_0FFF) | √ | √ | √ | √ |
| H'00_F100_0000 to H'00_F10F_FFFF | INTC-AP | (H'F101_0000 to H'F107_0FFF) | √ | √ | √ | √ |
| H'00_F110_0000 to H'00_F11F_FFFF | Reserved | (H'F111_0000 to H'F117_0FFF) | √ | √ | √ | √ |
| H'00_F120_0000 to H'00_F12F_FFFF | CCI-500 | | √ | √ | — | — |
| H'00_F130_0000 to H'00_F13F_FFFF | AXI-Bus [Leaf Arbitor for CPU domain] | CPU_ACT0: H'F130_0800 | √ | √ | √ | √ |
| | | CPU_ACT1: H'F134_0800 | √ | | | |
| | | CPU_ACT2: H'F138_0800 | √ | | | |
| | | CPU_ACT3: H'F13C_0800 | √ | | | |
| H'00_F140_0000 to H'00_FBFF_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_FC00_0000 to H'00_FCFE_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_FD00_0000 to H'00_FD03_FFFF | 3DGE | 3D accelerator Control Register (32bit) | √ | √ | √ | √ |
| H'00_FD04_0000 to H'00_FD7F_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_FD80_0000 to H'00_FD94_FFFF | 3DGE | Bus, IPMMU | √ | √ | √ | √ |
| H'00_FD95_0000 to H'00_FDFF_FFFF | IPMMU-PV1 | (H'FD95_0000 to H'FD95_FFFF) | √ | √ | — | — |
| H'00_FE00_0000 to H'00_FE0F_FFFF | PCIEC | PCIE0 Config | √ | √ | √ | √ |
| H'00_FE10_0000 to H'00_FE1F_FFFF | | PCIE0 memory0 | √ | √ | √ | √ |
| H'00_FE20_0000 to H'00_FE3F_FFFF | | PCIE0 memory1 | √ | √ | √ | √ |
| H'00_FE40_0000 to H'00_FE5F_FFFF | Reserved | | √ | √ | √ | √ |
| H'00_FE60_0000 to H'00_FE67_FFFF | VC | Video Codecs register | √ | √ | √ | √ |
| H'00_FE68_0000 to H'00_FE6F_FFFF | | | √ | √ | √ | √ |
| H'00_FE70_0000 to H'00_FE7F_FFFF | | | √ | √ | √ | √ |
| H'00_FE80_0000 to H'00_FE91_FFFF | | | √ | √ | √ | √ |
| H'00_FE92_0000 to H'00_FE9F_FFFF | VP-APB | Video Processing control register | √ | √ | √ | √ |

**Second Generation
RZ/G Series
Products**

| Address | | Space | Description | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|----------------|-------------------|-----------|--|--------|-------------|-------------|--------|--------|
| H'00_FEA0_0000 | to H'00_FEBF_FFFF | VIO-APB | Video Input Output Function control register | √ | √ | √ | √ | √ |
| H'00_FEC0_0000 | to H'00_FEFF_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_FF00_0000 | to H'00_FF3F_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_FF40_0000 | to H'00_FF5F_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_FF60_0000 | to H'00_FF6F_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_FF70_0000 | to H'00_FF7F_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_FF80_0000 | to H'00_FF87_FFFF | Slave Bus | Slave Bus register | √ | √ | √ | √ | √ |
| H'00_FF88_0000 | to H'00_FF8F_FFFF | IR-APB | IR domain peripheral bus | √ | √ | √ | √ | √ |
| H'00_FF90_0000 | to H'00_FF9F_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_FFA0_0000 | to H'00_FFBF_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_FFC0_0000 | to H'00_FFEF_FFFF | Reserved | | √ | √ | √ | √ | √ |
| H'00_FFF0_0000 | to H'00_FFFF_FFFF | Reserved | | √ | √ | √ | √ | √ |

Table 2.4 List of Module Base Address in each Domain (CP, HP, MP and other Clock Domains)

| Domain | Address[31:24] | Address[23:0] | Module | Second Generation RZ/G Series Products | | | | |
|---------------------------------------|----------------|---------------|----------------------------------|---|----------------------------|--------|--------|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| C5-APB (cpck) | H'E6 | H'00_0000 | Reserved | √ | √ | √ | √ | |
| | | H'00_1000 | Reserved | √ | √ | √ | √ | |
| | | H'02_0000 | RWDT | √ | √ | √ | √ | |
| | | H'03_0000 | SWDT | √ | √ | √ | √ | |
| | | H'04_0000 | SCMT | √ | √ | √ | √ | |
| PeriWest APSW1 | H'E6 | H'05_0000 | GPIO0 | √ | √ | √ | √ | |
| | | H'05_1000 | GPIO1 | √ | √ | √ | √ | |
| | | H'05_2000 | GPIO2 | √ | √ | √ | √ | |
| PeriEast APSE1 | H'E6 | H'05_3000 | GPIO3 | √ | √ | √ | √ | |
| | | H'05_4000 | GPIO4 | √ | √ | √ | √ | |
| | | H'05_5000 | GPIO5 | √ | √ | √ | √ | |
| | | H'05_5400 | GPIO6 | √ | √ | √ | √ | |
| | | H'05_5800 | GPIO7 | √ | √ | √ | — | |
| PeriWest APSW1 | H'E6 | H'06_0000 | PFC | √ | √ | √ | √ | |
| C5-APB (cpck) *: APSW1 [RZ/G2N] | H'E6 | H'08_0000 | Arm Generic Counter (armgcnt) | √ | √ | √ | √ | |
| | | H'0A_0000 | Reserved | √ | √ | √ | √ | |
| | | H'0B_0000* | IIC (PMIC) | √ | √ | √ | √ | |
| | | H'0F_0000 | CMT0 | √ | √ | √ | √ | |
| | | H'10_0000 | DBE | √ | √ | √ | √ | |
| | | H'11_0000 | Reserved | √ | √ | √ | √ | |
| | | H'12_0000 | Reserved | √ | √ | √ | √ | |
| | | H'13_0000 | CMT1 | √ | √ | √ | √ | |
| | | H'14_0000 | CMT2 | √ | √ | √ | √ | |
| | | H'14_8000 | CMT3 | √ | √ | √ | √ | |
| | | H'15_0000 | CPG | √ | √ | √ | √ | |
| | | H'16_0000 | RST | √ | √ | √ | √ | |
| | | H'18_0000 | SYSC | √ | √ | √ | √ | |
| | | H'19_0000 | THS/CIVM | — | — | — | √ | |
| | | H'19_8000 | TSC1 | √ | √ | √ | — | |
| | | H'1A_0000 | TSC2 | √ | √ | √ | — | |
| H'1A_8000 | TSC3 | √ | √ | √ | — | | | |
| H'1C_0000 | IRQC | √ | √ | √ | √ | | | |
| H'1D_0000 | SUCMT | √ | √ | √ | √ | | | |
| H'1E_0000 | TMU0 | √ | √ | √ | √ | | | |

**Second Generation
RZ/G Series Products**

| Domain | Address[31:24] | Address[23:0] | Module | Second Generation RZ/G Series Products | | | | |
|---------------|----------------|---------------|---|--|----------------------------|--------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| C5-APB (cpck) | H'E6 | H'1F_0000 | Reserved | √ | √ | √ | √ | |
| | | H'1F_2000 | Reserved | √ | √ | √ | √ | |
| | | H'1F_4000 | Reserved | √ | √ | √ | √ | |
| | | H'1F_6000 | Reserved | √ | √ | √ | √ | |
| | | H'1F_8000 | Reserved | √ | √ | √ | √ | |
| | | H'20_0000 | Reserved | √ | √ | √ | √ | |
| | | H'21_0000 | Reserved | √ | √ | √ | √ | |
| | | H'22_0000 | Reserved | √ | √ | √ | √ | |
| | | H'23_0000 | Reserved | √ | √ | √ | √ | |
| | | H'24_0000 | Reserved | √ | √ | √ | √ | |
| | | H'25_0000 | Reserved | √ | √ | √ | √ | |
| | | H'26_0000 | Reserved | √ | √ | √ | √ | |
| | | H'28_0000 | Reserved | √ | √ | √ | √ | |
| DDR Hier | H'E6 | H'30_0000 | System RAM (384KB: [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 128KB: [RZ/G2E]) | √ | √ | √ | √ | |
| AXI Slave | H'E6 | H'36_0000 | Reserved | √ | √ | √ | √ | |
| | | H'38_0000 | Reserved | √ | √ | √ | √ | |
| | | H'39_0000 | Reserved | √ | √ | √ | √ | |
| | | H'3A_0000 | System RAM Mirror (H'E633_0000 to H'E635_FFFF) | √ | √ | √ | — | |
| | | H'3D_0000 | Reserved | √ | √ | √ | √ | |
| | | H'3E_0000 | Reserved | √ | √ | √ | √ | |
| | | H'3F_0000 | Reserved | √ | √ | √ | √ | |
| | | H'41_0000 | Reserved | √ | √ | √ | √ | |
| | | H'42_0000 | Reserved | √ | √ | √ | √ | |
| | | H'42_8000 | Reserved | √ | √ | √ | √ | |
| HC-APB | H'E6 | H'43_0000 | HC LSTAT | √ | √ | — | — | |
| | | H'44_0000 | HC AXISTAT | √ | √ | — | — | |
| | | H'45_0000 | Reserved | √ | √ | √ | √ | |
| | | H'46_0000 | Reserved | √ | √ | √ | √ | |
| | | H'47_0000 | Reserved | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Domain | Address[31:24] | Address[23:0] | Module | Second Generation RZ/G Series Products | | | | | | |
|-----------------------------|-------------------|---------------|---------------|--|----------------------------|--------|--------|---|---|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | | | |
| PeriE APB (APSE2) 133MHz | H'E6 | H'50_0000 | I2C0 | √ | √ | √ | √ | | | |
| | | H'50_8000 | I2C1 | √ | √ | √ | √ | | | |
| | | H'51_0000 | I2C2 | √ | √ | √ | √ | | | |
| | | H'51_8000 | Reserved | √ | √ | √ | √ | | | |
| | | H'52_0000 | Reserved | √ | √ | √ | √ | | | |
| | | H'53_0000 | Reserved | √ | √ | √ | √ | | | |
| | | H'53_8000 | Reserved | √ | √ | √ | √ | | | |
| | | H'54_0000 | HSCIF0 | √ | √ | √ | √ | | | |
| | | H'55_0000 | HSCIF1 | √ | √ | √ | √ | | | |
| | | H'56_0000 | HSCIF2 | √ | √ | √ | √ | | | |
| | | HC-APB | H'E6 | H'57_0000 | IPMMU-HC | √ | √ | √ | √ | |
| | | | | H'58_0000 | HC AXI Router | √ | √ | √ | √ | |
| | | | | H'59_0000 | HS-USB ch0 | √ | √ | √ | √ | |
| H'59_4000 | Reserved | | | √ | √ | √ | √ | | | |
| H'59_8000 | Reserved | | | √ | √ | √ | √ | | | |
| H'59_C000 | Reserved | | | √ | √ | √ | √ | | | |
| H'5A_0000 | USB-DMAC | | | √ | √ | √ | — | | | |
| H'5B_0000 | USB-DMAC | | | √ | √ | √ | — | | | |
| H'5C_0000 | DDM | | | √ | √ | √ | √ | | | |
| H'5D_0000 | PCIEC (PCIE0 PHY) | | | √ | √ | √ | — | | | |
| H'5D_8000 | PCIEC (PCIE1 PHY) | | | √ | √ | √ | — | | | |
| H'5E_0000 | SATA | | | √ | — | √ | — | | | |
| H'5E_E000 | USB3.0 PHY | | | √ | √ | √ | — | | | |
| H'5F_0000 | Reserved | | | √ | √ | √ | √ | | | |
| H'5F_8000 | Reserved | | | √ | √ | √ | √ | | | |
| RC Hier APB | H'E6 | | | H'60_4000 | Reserved | √ | √ | √ | √ | |
| | | | | H'61_0000 | Reserved | √ | √ | √ | √ | |
| | | H'62_0000 | RC AXI Router | √ | √ | √ | √ | | | |
| | | H'63_0000 | Reserved | √ | √ | √ | √ | | | |
| | | H'64_0000 | Reserved | √ | √ | √ | √ | | | |
| | | H'65_0000 | Reserved | √ | √ | √ | √ | | | |

**Second Generation
RZ/G Series Products**

| Domain | Address[31:24] | Address[23:0] | Module | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------------------------------|----------------------|--------------------------------|--------------------------|-----------|----------------------------|--------|--------|
| PeriW APB (APSW2) 133MHz | H'E6 | H'68_0000 | Reserved | √ | √ | √ | √ |
| | | H'69_0000 | I2C7 | — | — | — | √ |
| | | H'6A_0000 | HSCIF3 | √ | √ | √ | √ |
| | | H'6B_0000 | HSCIF4 | √ | √ | √ | √ |
| | | H'6C_0000 | CAN-FD | √ | √ | √ | √ |
| | | H'6D_0000 | I2C3 | √ | √ | √ | √ |
| | | H'6D_8000 | I2C4 | √ | √ | √ | √ |
| | | H'6E_0000 | I2C5 | √ | √ | √ | √ |
| | | H'6E_8000 | I2C6 | √ | √ | √ | √ |
| | | H'6F_0000 | Peri WEST DMA AXI Router | √ | √ | √ | √ |
| | | H'70_0000 | SYS-DMAC0 | √ | √ | √ | √ |
| | | H'71_0000 | Reserved | √ | √ | √ | √ |
| | | PeriW APB (APSW2) 133MHz | H'E6 | H'72_0000 | Reserved | √ | √ |
| H'73_0000 | Reserved | | | √ | √ | √ | √ |
| H'74_0000 | IPMMU-DS0 | | | √ | √ | √ | √ |
| H'75_0000 | Peri WEST AXI Router | | | √ | √ | √ | √ |
| H'76_0000 | Reserved | | | √ | √ | √ | √ |
| H'77_0000 | Reserved | | | √ | √ | √ | √ |
| DDR Hier | H'E6 | | | H'78_0000 | AXI Bus | √ | √ |
| | | H'79_0000 | DBSC4 | √ | √ | √ | √ |
| | | H'7A_0000 | Reserved | √ | √ | — | — |
| | | H'7B_0000 | IPMMU-MM | √ | √ | √ | √ |
| | | H'7C_0000 | MM Slave AXI Router | √ | √ | √ | √ |
| | | H'7D_0000 | MM AXI Router | √ | √ | √ | √ |
| | | H'7E_0000 | AXI-Bus [QoS Register] | √ | √ | √ | √ |
| | | H'7F_0000 | AXI-Bus [QoS Register] | √ | √ | √ | √ |
| | | H'7F_2000 | Reserved | √ | √ | √ | √ |
| | | H'7F_3000 | Reserved | √ | √ | √ | √ |
| | | H'7E_4000 | Reserved | √ | √ | √ | √ |
| | | H'7E_5000 | Reserved | √ | √ | √ | √ |
| | | H'7E_6000 | Reserved | √ | √ | √ | √ |
| | | H'7E_7000 | Reserved | √ | √ | √ | √ |
| | | H'7F_8000 | Reserved | √ | √ | √ | √ |
| | | H'7F_E000 | Reserved | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Domain | Address[31:24] | Address[23:0] | Module | Second Generation RZ/G Series Products | | | | |
|---|----------------|---------------------------|----------------|--|----------------------------|--------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| PeriW AXI Slave EthernetAVB-IF, PeriW APB [RZ/G2E] | H'E6 | H'80_0000 to H'9F_FFFF | EthernetAVB-IF | √ | √ | √ | √ | |
| | | H'A0_0000 to H'A1_FFFF | Reserved | √ | √ | √ | √ | |
| RPC-APB 133MHz | H'E6 | H'B0_0000 | Reserved | √ | √ | √ | √ | |
| | | H'B1_0000 | Reserved | √ | √ | √ | √ | |
| | | H'B2_0000 | Reserved | √ | √ | √ | √ | |
| PeriW APB (APSW3) (mpck) 133MHz | H'E6 | H'C0_0000 | MSIOF2 | √ | √ | √ | √ | |
| | | H'C1_0000 | MSIOF3 | √ | √ | √ | √ | |
| 66.6MHz clock specified module PeriW APB (APSW1) | H'E6 | H'C2_8000 | Reserved | √ | √ | √ | √ | |
| | | H'C3_0000 | CAN0 | √ | √ | √ | √ | |
| | | H'C3_8000 | CAN1 | √ | √ | √ | √ | |
| | | H'C4_0000 | SCIF4 | √ | √ | √ | √ | |
| | | H'C4_8000 | Reserved | √ | √ | √ | √ | |
| | | H'C5_0000 | SCIF3 | √ | √ | √ | √ | |
| 66.6MHz clock specified module PeriW APB (APSW1) | H'E6 | H'C5_8000 | Reserved | √ | √ | √ | √ | |
| | | H'C6_0000 | Reserved | √ | √ | √ | √ | |
| | | H'C7_0000 | Reserved | √ | √ | √ | √ | |
| | | H'C8_0000 | Reserved | √ | √ | √ | √ | |
| | | H'CA_0000 | Reserved | √ | √ | √ | √ | |
| | | H'CE_0000 | Reserved | √ | √ | √ | √ | |
| | | H'CF_0000 | Reserved | √ | √ | √ | √ | |
| | | H'D0_0000 | Reserved | √ | √ | √ | √ | |
| | | H'D4_0000 | Reserved | √ | √ | √ | √ | |
| | | H'D5_0000 | Reserved | √ | √ | √ | √ | |
| | | H'D6_0000 | Reserved | √ | √ | √ | √ | |
| | | H'D7_0000 | Reserved | √ | √ | √ | √ | |
| | | H'E0_0000 | Reserved | √ | √ | √ | √ | |
| | | H'E1_0000 | Reserved | √ | √ | √ | √ | |
| | | H'E2_0000 | Reserved | √ | √ | √ | √ | |
| | | H'E3_0000 | PWM0 | √ | √ | √ | √ | |
| | | H'E3_1000 | PWM1 | √ | √ | √ | √ | |
| H'E3_2000 | PWM2 | √ | √ | √ | √ | | | |
| H'E3_3000 | PWM3 | √ | √ | √ | √ | | | |

**Second Generation
RZ/G Series Products**

| Domain | Address[31:24] | Address[23:0] | Module | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---|----------------|---------------|----------|--------|----------------------------|--------|--------|
| 66.6MHz clock specified module PeriW APB (APSW1) | H'E6 | H'E3_4000 | PWM4 | √ | √ | √ | √ |
| | | H'E3_5000 | PWM5 | √ | √ | √ | √ |
| | | H'E3_6000 | PWM6 | √ | √ | √ | √ |
| | | H'E3_7000 | Reserved | √ | √ | √ | √ |
| PeriE APB (APSE1) 66MHz | H'E6 | H'E4_0000 | Reserved | √ | √ | √ | √ |
| | | H'E5_0000 | Reserved | √ | √ | √ | √ |
| | | H'E5_1000 | Reserved | √ | √ | √ | √ |
| | | H'E5_2000 | Reserved | √ | √ | √ | √ |
| | | H'E5_3000 | Reserved | √ | √ | √ | √ |
| | | H'E5_4000 | Reserved | √ | √ | √ | √ |
| | | H'E5_5000 | Reserved | √ | √ | √ | √ |
| | | H'E5_6000 | Reserved | √ | √ | √ | √ |
| | | H'E5_7000 | Reserved | √ | √ | √ | √ |
| | | H'E6_0000 | SCIF0 | √ | √ | √ | √ |
| | | H'E6_8000 | SCIF1 | √ | √ | √ | √ |
| | | H'E7_0000 | Reserved | √ | √ | √ | √ |
| | | H'E8_0000 | TPU0 | √ | √ | √ | √ |
| | | H'E8_8000 | SCIF2 | √ | √ | √ | √ |
| PeriE APB (APSRZ/G2E) mpck *: APSW1 [RZ/G2N] | H'E6 | H'E9_0000 | MSIOF0 | √ | √ | √ | √ |
| | | H'EA_0000 | MSIOF1 | √ | √ | √ | √ |
| | | H'EB_0000 | Reserved | √ | √ | √ | √ |
| | | H'EC_0000* | Reserved | √ | √ | √ | √ |
| VIO-APB 100MHz | H'E6 | H'EF_0000 | VIN0 | √ | √ | √ | — |
| | | H'EF_1000 | VIN1 | √ | √ | √ | — |
| | | H'EF_2000 | VIN2 | √ | √ | √ | — |
| | | H'EF_3000 | VIN3 | √ | √ | √ | — |
| | | H'EF_4000 | VIN4 | √ | √ | √ | √ |
| | | H'EF_5000 | VIN5 | √ | √ | √ | √ |
| | | H'EF_6000 | VIN6 | √ | √ | √ | — |
| | | H'EF_7000 | VIN7 | √ | √ | √ | — |
| | | H'EF_8000 | Reserved | √ | √ | √ | √ |
| | | H'EF_9000 | Reserved | √ | √ | √ | √ |
| | | H'EF_A000 | Reserved | √ | √ | √ | √ |
| | | H'EF_B000 | Reserved | √ | √ | √ | √ |
| | | H'EF_C000 | Reserved | √ | √ | √ | √ |
| | | H'EF_D000 | Reserved | √ | √ | √ | √ |
| | | H'EF_E000 | Reserved | √ | √ | √ | √ |
| | | H'EF_F000 | Reserved | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Domain | Address[31:24] | Address[23:0] | Module | Second Generation RZ/G Series Products | | | | |
|---|----------------|---------------|-----------------------|--|----------------------------|--------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| VIO-APB 100MHz | H'E6 | H'F0_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F1_0000 | Reserved | √ | √ | √ | √ | |
| PeriE APB (APSE1) 66MHz | H'E6 | H'F2_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F3_0000 | SCIF5 | √ | √ | √ | √ | |
| PeriE APB (APSE2) 133MHz | H'E6 | H'F4_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F5_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F6_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F7_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F8_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F9_0000 | Reserved | √ | √ | √ | √ | |
| | | H'FA_0000 | Reserved | √ | √ | √ | √ | |
| | | H'FB_0000 | Reserved | √ | √ | √ | √ | |
| | | H'FC_0000 | TMU1 | √ | √ | √ | √ | |
| | | H'FD_0000 | TMU2 | √ | √ | √ | √ | |
| | | H'FE_0000 | TMU3 | √ | √ | √ | √ | |
| | | H'FF_0000 | Reserved | √ | √ | √ | √ | |
| | | H'E7 | H'00_0000 | Reserved | √ | √ | √ | √ |
| | | | H'01_0000 | Reserved | √ | √ | √ | √ |
| PeriE APB (APSE2) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] (APSE6) [RZ/G2N] | H'E7 | H'30_0000 | SYS-DMAC1_East1 | √ | √ | √ | √ | |
| | | H'31_0000 | SYS-DMAC2_East2 | √ | √ | √ | √ | |
| | | H'32_0000 | Reserved | √ | √ | √ | √ | |
| | | H'37_0000 | Reserved | √ | √ | √ | √ | |
| | | H'38_0000 | Reserved | √ | √ | √ | √ | |
| PeriE APB | H'E7 | H'73_0000 | IPMMU Peri East (SY0) | √ | √ | — | √ | |

| | | | | Second Generation RZ/G Series Products | | | | |
|---|----------------|---------------|--------------------------|---|----------------------------|--------|--------|--|
| Domain | Address[31:24] | Address[23:0] | Module | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| PeriE APB (APSE2) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] (APSE6) [RZ/G2N] 133MHz | H'E7 | H'74_0000 | IPMMU-DS1 | √ | √ | √ | √ | |
| | | H'75_0000 | Peri EAST AXI Router | √ | √ | √ | √ | |
| | | H'76_0000 | Reserved | √ | √ | √ | √ | |
| | | H'77_0000 | Reserved | √ | √ | √ | √ | |
| | | H'78_0000 | Peri EAST DMA AXI Router | √ | √ | √ | √ | |
| | | H'79_0000 | Reserved | √ | √ | √ | √ | |
| | | H'90_0000 | Reserved | √ | √ | √ | √ | |
| | | H'91_0000 | Reserved | √ | √ | √ | √ | |
| MP-APB1 | H'EC | H'00_0000 | SRC_CMD_BUSIF (window) | √ | √ | √ | √ | |
| | | H'02_0000 | Reserved | √ | √ | √ | √ | |
| MP-APB2 | H'EC | H'10_0000 | SSI_BUSIF (window) | √ | √ | √ | √ | |
| MP-APB3 | H'EC | H'20_0000 | Reserved | √ | √ | √ | √ | |
| | | H'24_0000 | SSI0~SSI2 (window) | √ | √ | √ | √ | |
| | | | SSI3~SSI4 (window) | √ | √ | √ | √ | |
| SSI5~SSI9 (window) | √ | | √ | √ | √ | | | |
| MP-APB4 | H'EC | H'30_0000 | SRC_CMD_BUSIF (window) | √ | √ | √ | √ | |
| | | H'32_0000 | Reserved | √ | √ | √ | √ | |
| MP-APB5 | H'EC | H'40_0000 | SSI_BUSIF (window) | √ | √ | √ | √ | |
| | | H'44_0000 | Reserved | √ | √ | √ | √ | |
| | | H'46_0000 | Reserved | √ | √ | √ | √ | |

| | | | | Second Generation RZ/G Series Products | | | | | |
|-----------|----------------|---------------|-----------------------|---|----------------------------|--------|--------|---|---|
| Domain | Address[31:24] | Address[23:0] | Module | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | | |
| MP-APB6 | H'EC | H'50_0000 | SRC0~4, CMD0, 1 (apb) | √ | √ | √ | √ | | |
| | | | SRC5~6, CMD0, 1 (apb) | √ | √ | √ | √ | | |
| | | | SRC7~9, CMD0, 1 (apb) | √ | √ | √ | √ | | |
| | | H'52_0000 | Reserved | √ | √ | √ | √ | | |
| | | H'54_0000 | SSI0~SSI2 (apb) | √ | √ | √ | √ | | |
| | | | SSI3~SSI4 (apb) | √ | √ | √ | √ | | |
| | | | SSI5~SSI9 (apb) | √ | √ | √ | √ | | |
| | | H'58_0000 | Reserved | √ | √ | √ | √ | | |
| | | H'5A_0000 | ADG (apb) | √ | √ | √ | √ | | |
| | MP-APB7 | H'EC | H'60_0000 | MP AXI Router | √ | √ | √ | √ | |
| H'61_0000 | | | MP DMA AXI Router | √ | √ | √ | √ | | |
| H'62_0000 | | | Reserved | √ | √ | √ | √ | | |
| H'63_0000 | | | Reserved | √ | √ | √ | √ | | |
| H'64_0000 | | | Reserved | √ | √ | √ | √ | | |
| H'67_0000 | | | IPMMU-MP | √ | √ | √ | √ | | |
| H'68_0000 | | | Reserved | √ | — | — | — | | |
| H'6F_0000 | | | Reserved | √ | √ | √ | √ | | |
| H'70_0000 | | | Audio-DMAC Lower | √ | √ | √ | √ | | |
| H'72_0000 | | | Audio-DMAC Higher | √ | √ | √ | √ | | |
| H'74_0000 | | | Audio-DMACpp | √ | √ | √ | — | | |
| H'76_0000 | | | Audio-DMACpp Extended | √ | √ | √ | √ | | |
| H'80_0000 | | | Reserved | √ | √ | √ | √ | | |
| H'C0_0000 | | | Reserved | √ | √ | √ | √ | | |
| H'E0_0000 | | | Reserved | √ | √ | √ | √ | | |
| USB3.0 | | | H'EE | H'00_0000 | USB3-0 Host | √ | √ | √ | √ |
| | | | | H'02_0000 | USB3-0 Peripheral | √ | √ | √ | √ |
| | H'04_0000 | Reserved | | √ | √ | √ | √ | | |
| | H'06_0000 | Reserved | | √ | √ | √ | √ | | |
| SDHI | H'EE | H'10_0000 | SDHI0 | √ | √ | √ | √ | | |
| | | H'12_0000 | SDHI1 | √ | √ | √ | √ | | |
| | | H'14_0000 | SDHI2/MMC0 | √ | √ | √ | — | | |
| | | H'16_0000 | SDHI3/MMC1 | √ | √ | √ | √ | | |
| | | H'18_0000 | RAW NAND FLASH | — | √ | √ | √ | | |
| RPC/BSC | H'EE | H'20_0000 | RPC-IF | √ | √ | √ | √ | | |
| | | H'22_0000 | BSC | √ | √ | √ | √ | | |
| RGX | H'FD | H'00_0000 | 3DGE | √ | √ | √ | √ | | |
| | | H'04_0000 | Reserved | √ | √ | √ | √ | | |

**Second Generation
RZ/G Series Products**

| Domain | Address[31:24] | Address[23:0] | Module | Second Generation RZ/G Series Products | | | | | | |
|-----------|--------------------|---------------|-----------------|--|----------------------------|--------|--------|---|---|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | | | |
| RGX-APB | H'FD | H'80_0000 | IPMMU-PV0 | √ | √ | √ | √ | | | |
| | | H'81_0000 | 3DGE AXI Router | √ | √ | √ | √ | | | |
| | | H'82_0000 | Reserved | √ | √ | √ | √ | | | |
| | | H'83_0000 | Reserved | √ | √ | √ | √ | | | |
| | | H'84_0000 | Reserved | √ | √ | √ | √ | | | |
| | | H'90_0000 | Reserved | √ | √ | √ | √ | | | |
| | | H'91_0000 | Reserved | √ | √ | √ | √ | | | |
| | | H'92_0000 | Reserved | √ | √ | √ | √ | | | |
| | | H'93_0000 | Reserved | √ | √ | √ | √ | | | |
| | | H'94_0000 | Reserved | √ | √ | √ | √ | | | |
| | | H'95_0000 | IPMMU-PV1 | √ | √ | — | — | | | |
| | | H'96_0000 | IPMMU-PV2 | √ | — | — | — | | | |
| | | H'97_0000 | IPMMU-PV3 | √ | — | — | — | | | |
| | | H'98_0000 | Reserved | √ | √ | √ | √ | | | |
| | | VC Hier APB | H'FE | H'60_0000 | Reserved | √ | √ | √ | √ | |
| | | | | H'61_0000 | Reserved | √ | √ | √ | √ | |
| | | | | H'68_0000 | VC AXI Router0 | √ | √ | √ | √ | |
| | | | | H'69_0000 | Reserved | √ | √ | √ | √ | |
| H'6A_0000 | Reserved | | | √ | √ | √ | √ | | | |
| H'6B_0000 | IPMMU-VC0 | | | √ | √ | √ | √ | | | |
| H'6C_0000 | Reserved | | | √ | √ | √ | √ | | | |
| H'6D_0000 | Reserved | | | √ | √ | √ | √ | | | |
| H'6E_0000 | Reserved | | | √ | √ | √ | √ | | | |
| H'6F_0000 | IPMMU-VC1 | | | √ | — | — | — | | | |
| H'70_0000 | Reserved | | | √ | √ | √ | √ | | | |
| H'86_0000 | Reserved | | | √ | √ | √ | √ | | | |
| H'87_0000 | Reserved | | | √ | √ | √ | √ | | | |
| H'88_0000 | Reserved | | | √ | √ | √ | √ | | | |
| H'89_0000 | Reserved | | | √ | √ | √ | √ | | | |
| H'8A_0000 | Reserved | | | √ | √ | √ | √ | | | |
| H'8B_0000 | Reserved | | | √ | √ | √ | √ | | | |
| H'8D_0000 | VLC iVDP1C | | | √ | √ | √ | √ | | | |
| H'8D_0200 | CE for iVDP1C | | | √ | √ | √ | √ | | | |
| H'8D_F000 | FCPCI (for iVDP1C) | | | √ | √ | — | — | | | |

**Second Generation
RZ/G Series Products**

| Domain | Address[31:24] | Address[23:0] | Module | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | | |
|-------------|-----------------------|---------------|--------------------------|-----------|----------------------------|--------|--------|---|---|
| VC Hier APB | H'FE | H'8E_0000 | Reserved | √ | √ | √ | √ | | |
| | | H'8F_0000 | Reserved | √ | √ | √ | √ | | |
| | | H'8F_0200 | Reserved | √ | √ | √ | √ | | |
| | | H'8F_F000 | Reserved | √ | √ | √ | √ | | |
| | | H'90_0000 | VLC for H.265 | √ | √ | √ | √ | | |
| | | H'90_0200 | CE for H.265 | √ | √ | √ | √ | | |
| | | H'90_8000 | Reserved | √ | √ | √ | √ | | |
| | | H'90_F000 | FCPCS | √ | √ | √ | √ | | |
| | | H'91_0000 | VLC for VCPLF | √ | √ | √ | √ | | |
| | | H'91_0200 | CE for VCPLF | √ | √ | √ | √ | | |
| | | H'91_8000 | Reserved | √ | √ | √ | √ | | |
| | | VIO (APB) | H'FE | H'92_0000 | Reserved | √ | √ | √ | √ |
| | | | | H'92_F000 | Reserved | √ | √ | √ | √ |
| H'93_0000 | Reserved | | | √ | √ | √ | √ | | |
| VC (APB) | H'FE | H'94_0000 | FDP0 | √ | √ | √ | √ | | |
| | | H'94_4000 | FDP1 | √ | — | — | — | | |
| | | H'94_8000 | Reserved | √ | √ | √ | √ | | |
| | | H'94_B000 | Reserved | √ | √ | √ | √ | | |
| | | H'95_0000 | FCP for FDP0 (4K) | √ | √ | √ | √ | | |
| | | H'95_1000 | FCP for FDP1 (4K) | √ | — | — | — | | |
| | | H'95_2000 | Reserved | √ | √ | √ | √ | | |
| | | H'95_3000 | Reserved | √ | √ | √ | √ | | |
| VIO (APB) | H'FE | H'96_0000 | VSP-BLD_DRC0 | √ | √ | √ | — | | |
| | | | VSPBS/VSPB | — | — | — | √ | | |
| | | H'96_F000 | FCP for VSP-BLD-DRC (4K) | √ | √ | √ | — | | |
| | | | FCPVB0 | — | — | — | √ | | |
| | | H'97_0000 | Reserved | √ | √ | √ | √ | | |
| | | H'97_F000 | Reserved | √ | √ | √ | √ | | |
| | | H'98_0000 | IPMMU-VP1 | √ | — | — | — | | |
| | | H'99_0000 | IPMMU-VP0 | √ | — | √ | √ | | |
| | | VC (APB) | H'FE | H'9A_0000 | VSP-IMG0 | √ | √ | √ | √ |
| | | | | H'9A_F000 | FCP for VSP-IMG0 (4K) | √ | √ | √ | — |
| H'9B_0000 | VSP-IMG1 | | | √ | — | — | — | | |
| H'9B_F000 | FCP for VSP-IMG1 (4K) | | | √ | — | — | — | | |
| H'9C_0000 | Reserved | | | √ | √ | √ | √ | | |
| H'9C_F000 | Reserved | | | √ | √ | √ | √ | | |
| H'9D_0000 | VP AXI Router | | | √ | — | √ | √ | | |
| H'9E_0000 | Reserved | | | √ | √ | √ | √ | | |
| H'9E_8000 | Reserved | √ | √ | √ | √ | | | | |

**Second Generation
RZ/G Series Products**

| Domain | Address[31:24] | Address[23:0] | Module | Second Generation RZ/G Series Products | | | | |
|-----------|----------------|---------------|-----------------|--|----------------------------|--------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| VC (APB) | H'FE | H'9F_0000 | Reserved | √ | √ | √ | √ | √ |
| VIO Hier | H'FE | H'A0_0000 | Reserved | √ | √ | √ | √ | √ |
| | | H'A1_0000 | Reserved | √ | √ | √ | √ | √ |
| | | H'A2_0000 | VSPD0 (VSPD0) | √ | √ | √ | √ | √ |
| | | H'A2_7000 | FCP for VSPDU0 | √ | √ | √ | √ | √ |
| | | H'A2_8000 | VSPD1 (VSPD1) | √ | √ | √ | √ | √ |
| | | H'A2_F000 | FCP for VSPDU1 | √ | √ | √ | √ | √ |
| | | H'A3_0000 | VSPD2 (VSPD2) | — | √ | — | — | — |
| | | H'A3_7000 | FCP for VSPDU2 | — | √ | — | — | — |
| | | H'A3_8000 | Reserved | √ | √ | √ | √ | √ |
| | | H'A3_F000 | Reserved | √ | √ | √ | √ | √ |
| | | H'A4_0000 | Reserved | √ | √ | √ | √ | √ |
| | | H'A5_0000 | Reserved | √ | √ | √ | √ | √ |
| | | H'A6_0000 | Reserved | √ | √ | √ | √ | √ |
| | | H'A7_0000 | Reserved | √ | √ | √ | √ | √ |
| | | H'A8_0000 | CSI2LNK0 (16KB) | √ | √ | √ | — | — |
| | | H'A9_0000 | Reserved | √ | √ | √ | √ | √ |
| | | H'AA_0000 | CSI4LNK0 (16KB) | √ | √ | √ | √ | √ |
| | | H'AB_0000 | Reserved | √ | √ | √ | √ | √ |
| | | H'AC_0000 | Reserved | √ | √ | √ | √ | √ |
| | | H'AD_0000 | HDMI0_CNT | √ | √ | √ | — | — |
| | | H'AD_8000 | HDMI0_ESM | — | √ | √ | — | — |
| | | H'AE_0000 | HDMI1_CNT | √ | — | — | — | — |
| | | H'AE_8000 | Reserved | √ | √ | √ | √ | √ |
| | | H'B0_0000 | DU0, DU1 | √ | √ | √ | √ | √ |
| | | H'B4_0000 | DU2 | — | √ | √ | — | — |
| | | H'B7_0000 | DU3 | √ | — | √ | — | — |
| | | H'B7_F000 | Reserved | √ | √ | √ | √ | √ |
| | | H'B8_0000 | Reserved | √ | √ | √ | √ | √ |
| | | H'B8_1000 | Reserved | √ | √ | √ | √ | √ |
| | | H'B8_2000 | Reserved | √ | √ | √ | √ | √ |
| | | H'B8_3000 | Reserved | √ | √ | √ | √ | √ |
| | | H'B8_4000 | Reserved | √ | √ | √ | √ | √ |
| H'B8_5000 | Reserved | √ | √ | √ | √ | √ | | |
| H'B8_8000 | VIO AXI Router | √ | √ | — | — | — | | |
| H'B8_9000 | Reserved | √ | √ | √ | √ | √ | | |
| H'B8_D000 | Reserved | √ | √ | √ | √ | √ | | |
| H'B8_E000 | Reserved | √ | √ | √ | √ | √ | | |
| H'B8_C000 | Reserved | √ | √ | √ | √ | √ | | |

**Second Generation
RZ/G Series Products**

| Domain | Address[31:24] | Address[23:0] | Module | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | | |
|-----------|-----------------------|---------------|-------------------------|-----------|----------------------------|--------|--------|---|---|
| VIO Hier | H'FE | H'B9_0000 | LVDS0 | √ | √ | √ | √ | | |
| | | H'B9_4000 | Reserved | √ | √ | √ | √ | | |
| | | H'B9_8000 | Reserved | √ | √ | √ | √ | | |
| | | H'BA_0000 | Reserved | √ | √ | √ | √ | | |
| | | H'BB_8000 | Reserved | √ | √ | √ | √ | | |
| | | H'BD_0000 | IPMMU-VI0 | √ | √ | √ | √ | | |
| | | H'BE_0000 | IPMMU-VI1 | √ | — | — | — | | |
| | | H'BF_0000 | AXI-Bus[VIO AXI Router] | √ | √ | √ | √ | | |
| | | H'CO_0000 | Reserved | √ | √ | √ | √ | | |
| | | H'D0_0000 | Reserved | √ | √ | √ | √ | | |
| | | H'D1_0000 | Reserved | √ | √ | √ | √ | | |
| | | H'D2_0000 | Reserved | √ | √ | √ | √ | | |
| | | H'E0_0000 | Reserved | √ | √ | √ | √ | | |
| | | | H'FF | H'00_0000 | Reserved | √ | √ | √ | √ |
| | | Slave Bus | H'FF | H'80_0000 | AXI Slave Bus West | √ | √ | √ | √ |
| | | | | H'81_0000 | AXI Slave Bus East | √ | √ | √ | √ |
| H'82_0000 | Reserved | | | √ | √ | √ | √ | | |
| H'83_0000 | Reserved | | | √ | √ | √ | √ | | |
| H'84_0000 | AXI-Bus[EDC Ctrl] | | | √ | √ | √ | √ | | |
| H'85_0000 | CCI-AXI Reg Block | | | √ | √ | — | — | | |
| H'86_0000 | AXI Slave Bus Central | | | √ | √ | √ | √ | | |
| H'87_0000 | Reserved | | | √ | √ | √ | √ | | |
| H'87_1000 | Reserved | | | √ | √ | √ | √ | | |
| H'87_2000 | Reserved | | | √ | √ | √ | √ | | |
| H'87_3000 | Reserved | | | √ | √ | √ | √ | | |
| H'87_4000 | Reserved | | | √ | √ | √ | √ | | |
| H'87_5000 | Reserved | | | √ | √ | √ | √ | | |
| H'87_6000 | Reserved | | | √ | √ | √ | √ | | |
| H'87_7000 | Reserved | | | √ | √ | √ | √ | | |
| H'87_8000 | Reserved | | | √ | √ | √ | √ | | |
| H'87_9000 | Reserved | | | √ | √ | √ | √ | | |
| H'87_A000 | Reserved | | | √ | √ | √ | √ | | |
| H'87_B000 | Reserved | | | √ | √ | √ | √ | | |
| H'87_C000 | Reserved | | | √ | √ | √ | √ | | |
| H'87_D000 | Reserved | | | √ | √ | √ | √ | | |
| IR-APB | H'FF | | | H'88_0000 | IR AXI Router0 | √ | √ | √ | √ |
| | | | | H'89_0000 | Reserved | √ | √ | √ | √ |
| | | H'8A_0000 | Reserved | √ | √ | √ | √ | | |

**Second Generation
RZ/G Series Products**

| Domain | Address[31:24] | Address[23:0] | Module | Second Generation RZ/G Series Products | | | | |
|--------|----------------|---------------|----------------|--|----------------------------|--------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| IR-APB | H'FF | H'8B_0000 | IPMMU-IR | √ | √ | √ | √ | |
| | | H'8C_0000 | Reserved | √ | √ | √ | √ | |
| | | H'8D_0000 | Reserved | √ | √ | √ | √ | |
| | | H'8D_1000 | Reserved | √ | √ | √ | √ | |
| | | H'8D_2000 | Reserved | √ | √ | √ | √ | |
| | | H'8D_3000 | Reserved | √ | √ | √ | √ | |
| | | H'8D_4000 | Reserved | √ | √ | √ | √ | |
| | | H'8D_5000 | Reserved | √ | √ | √ | √ | |
| | | H'8D_6000 | Reserved | √ | √ | √ | √ | |
| | | H'8D_7000 | Reserved | √ | √ | √ | √ | |
| | | H'8D_8000 | Reserved | √ | √ | √ | √ | |
| | | H'8E_0000 | IR AXI Router1 | √ | √ | √ | √ | |
| | | H'8F_0000 | Reserved | √ | √ | √ | √ | |
| | | H'90_0000 | Reserved | √ | √ | √ | √ | |
| | | H'A0_0000 | Reserved | √ | √ | √ | √ | |
| | | H'B0_0000 | Reserved | √ | √ | √ | √ | |
| | | H'BE_0000 | Reserved | √ | √ | √ | √ | |
| | | H'BF_0000 | Reserved | √ | √ | √ | √ | |
| | | H'C0_0000 | Reserved | √ | √ | √ | √ | |
| | | H'C1_0000 | Reserved | √ | √ | √ | √ | |
| | | H'C2_0000 | Reserved | √ | √ | √ | √ | |
| | | H'C3_0000 | Reserved | √ | √ | √ | √ | |
| | | H'C4_0000 | Reserved | √ | √ | √ | √ | |
| | | H'C5_0000 | Reserved | √ | √ | √ | √ | |
| | | H'C6_0000 | Reserved | √ | √ | √ | √ | |
| | | H'C7_0000 | Reserved | √ | √ | √ | √ | |
| | | H'C8_0000 | Reserved | √ | √ | √ | √ | |
| | | H'C9_0000 | Reserved | √ | √ | √ | √ | |
| | | H'CA_0000 | Reserved | √ | √ | √ | √ | |
| | | H'CB_0000 | Reserved | √ | √ | √ | √ | |
| | | H'CC_0000 | Reserved | √ | √ | √ | √ | |
| | | H'CD_0000 | Reserved | √ | √ | √ | √ | |
| | | H'CE_0000 | Reserved | √ | √ | √ | √ | |
| | | H'CF_0000 | Reserved | √ | √ | √ | √ | |
| | | H'D0_0000 | Reserved | √ | √ | √ | √ | |
| | | H'D1_0000 | Reserved | √ | √ | √ | √ | |
| | | H'D2_0000 | Reserved | √ | √ | √ | √ | |
| | | H'D3_0000 | Reserved | √ | √ | √ | √ | |
| | | H'D4_0000 | Reserved | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Domain | Address[31:24] | Address[23:0] | Module | Second Generation RZ/G Series Products | | | | |
|--------|----------------|---------------|----------|--|----------------------------|--------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| IR-APB | H'FF | H'D5_0000 | Reserved | √ | √ | √ | √ | |
| | | H'D6_0000 | Reserved | √ | √ | √ | √ | |
| | | H'D7_0000 | Reserved | √ | √ | √ | √ | |
| | | H'E8_0000 | Reserved | √ | √ | √ | √ | |
| | | H'E8_1000 | Reserved | √ | √ | √ | √ | |
| | | H'E8_2000 | Reserved | √ | √ | √ | √ | |
| | | H'E8_3000 | Reserved | √ | √ | √ | √ | |
| | | H'E8_4000 | Reserved | √ | √ | √ | √ | |
| | | H'E8_5000 | Reserved | √ | √ | √ | √ | |
| | | H'E8_6000 | Reserved | √ | √ | √ | √ | |
| | | H'E8_7000 | Reserved | √ | √ | √ | √ | |
| | | H'E8_8000 | Reserved | √ | √ | √ | √ | |
| | | H'E8_9000 | Reserved | √ | √ | √ | √ | |
| | | H'E8_A000 | Reserved | √ | √ | √ | √ | |
| | | H'E8_B000 | Reserved | √ | √ | √ | √ | |
| | | H'E9_0000 | Reserved | √ | √ | √ | √ | |
| | | H'EA_0000 | Reserved | √ | √ | √ | √ | |
| | | H'EF_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F0_0044 | PRR | √ | √ | √ | √ | |
| | | H'F1_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F2_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F3_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F4_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F5_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F6_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F7_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F8_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F8_1000 | Reserved | √ | √ | √ | √ | |
| | | H'F9_0000 | Reserved | √ | √ | √ | √ | |
| | | H'F9_1000 | Reserved | √ | √ | √ | √ | |
| | | H'F9_2000 | Reserved | √ | √ | √ | √ | |
| | | H'F9_3000 | Reserved | √ | √ | √ | √ | |
| | | H'FA_0000 | Reserved | √ | √ | √ | √ | |
| | | H'FB_0000 | Reserved | √ | √ | √ | √ | |
| | | H'FC_0000 | Reserved | √ | √ | √ | √ | |
| | | H'FD_0000 | Reserved | √ | √ | √ | √ | |
| | | H'FE_0000 | Reserved | √ | √ | √ | √ | |
| | | H'FF_0000 | Reserved | √ | √ | √ | √ | |

3. Mode Pin Settings (RZ/G2H)

RZ/G2H

RZ/G2M V1.3

RZ/G2M V3.0

RZ/G2N

RZ/G2E

Table 3.1 TEST2 and TEST1 Settings

| Test2 | Test1 | Mode Select |
|-------|-------|--------------------|
| 0 | 0 | Normal operation |
| 0 | 1 | Setting prohibited |
| 1 | 0 | Setting prohibited |
| 1 | 1 | Setting prohibited |

Table 3.2 BSMODE Settings

| BSMODE | JTAG pin Operating Mode Switching |
|--------|---|
| 0 | Normal operation |
| 1 | Operates in boundary scan mode. When PRESET# is at low level, pin IO control is disabled. |

Table 3.3 MD0 Settings

| MD0 | Reserved, fixed to 0 |
|-----|----------------------|
| | |

Table 3.4 MD4, MD3, MD2, MD1 Settings

| MD4 | MD3 | MD2 | MD1 | Boot Device Selection |
|-----|-----|-----|-----|-----------------------|
| — | — | — | — | See the Boot section. |

Table 3.5 MD5 Settings

| MD5 | Reserved (See the Boot section.) |
|-----|----------------------------------|
| 0 | Setting prohibited |
| 1 | Normal boot |

Table 3.6 MD7 and MD6 Settings

| MD7 | MD6 | Master Boot Processor Selection (See the Boot section.) |
|-----|-----|---|
| 0 | 0 | Cortex-A57 boot |
| 0 | 1 | Cortex-A53 boot |
| 1 | 0 | Setting prohibited |
| 1 | 1 | Setting prohibited |

Table 3.7 MD8 Settings

| MD8 | EXBUS Data Bus Width |
|-----|----------------------|
| 0 | 8-bit |
| 1 | 16-bit |

Table 3.8 MD9 Settings

| MD9 | EXTAL/XTAL Pin Setting (See the Clock Pulse Generator section.) |
|-----|---|
| 0 | Inputs an external clock to the EXTAL pin. |
| 1 | Connects the crystal resonator to the EXTAL and XTAL pin |

Table 3.9 MD21, MD20, MD11, MD10 and MDT[1:0] Settings

| MD21 | MD20 | MD11 | MD10 | MDT[1:0] | JTAG Selection |
|------|------|------|------|----------|----------------------------------|
| — | — | — | — | — | See the Debug and Trace section. |

Table 3.10 MD14 and MD13 Settings

| MD14 | MD13 | Clock Frequency Selection |
|------|------|--|
| — | — | See the Clock Pulse Generator section. |

Table 3.11 MD15 Settings

| MD15 | Initial ARMv8 Execution State Selection (See the Boot section.) |
|------|---|
| 0 | AArch32 |
| 1 | AArch64 |

Table 3.12 MD16 Settings

| MD16 | Reserved, fixed to 1 |
|------|----------------------|
| | |

Table 3.13 MD18 Settings

| MD18 | External Bus Clock Frequency Setting (See the Clock Pulse Generator section.) |
|------|---|
| 0 | Division ratio = 1/24 of the PLL1 oscillation frequency |
| 1 | Division ratio = 1/36 of the PLL1 oscillation frequency |

Table 3.14 MD19 and MD17 Settings

| MD19 | MD17 | LPDDR4 Data Transfer Rate |
|------|------|---------------------------|
| | | LPDDR4 |
| 0 | 0 | DDR3200 |
| 0 | 1 | DDR2800 |
| 1 | 0 | Setting prohibited |
| 1 | 1 | DDR1600 |

Table 3.15 MD27 and MD22 Settings

| MD27 | MD22 | DDR Phy Pin-mux Setting |
|------|------|-------------------------|
| 0 | 0 | LPDDR4 |
| 0 | 1 | Setting prohibited |
| 1 | 0 | Setting prohibited |
| 1 | 1 | Setting prohibited |

Table 3.16 MD25 Settings

| MD25 | Reserved, fixed to 0 |
|------|----------------------|
| | |

Table 3.17 MD23 Settings

| MD23 | Reserved, fixed to 0 |
|------|----------------------|
| | |

Table 3.18 MD26 Settings

| MD26 | Reserved, fixed to 0 |
|------|----------------------|
| | |

Table 3.19 MD28 Settings

| MD28 | Select RCLK Source |
|------|--------------------|
| 0 | CLK from inside |
| 1 | from EXTALR |

Table 3.20 MD12 Settings

| MD12 | Select PCIE or SATA |
|------|---------------------|
| 0 | PCl express |
| 1 | SATA |

4. Mode Pin Settings (RZ/G2M V1.3, RZ/G2M V3.0)



Table 4.1 TEST2 and TEST1 Settings

| TEST2 | TEST1 | Mode Select |
|-------|-------|--------------------|
| 0 | 0 | Normal operation |
| 0 | 1 | Setting prohibited |
| 1 | 0 | Setting prohibited |
| 1 | 1 | Setting prohibited |

Table 4.2 BSMODE Settings

| BSMODE | JTAG Pin Operating Mode Switching |
|--------|--|
| 0 | Normal operation |
| 1 | Operates in boundary scan mode. When PRESET# is at low level, pin IO control is disabled |

Table 4.3 MD0 Settings

| MD0 | Reserved, fixed to 0 |
|-----|----------------------|
| | |

Table 4.4 MD4, MD3, MD2 and MD1 Settings

| MD4 | MD3 | MD2 | MD1 | Boot Device Selection |
|-----|-----|-----|-----|-----------------------|
| — | — | — | — | See the Boot section |

Table 4.5 MD5 Settings

| MD5 | Reserved (See the Boot section.) |
|-----|----------------------------------|
| 0 | Setting prohibited |
| 1 | Normal boot |

Table 4.6 MD7 and MD6 Settings

| MD7 | MD6 | Master Boot Processor Selection (See the Boot section.) |
|-----|-----|---|
| 0 | 0 | Cortex-A57 boot |
| 0 | 1 | Cortex-A53 boot |
| 1 | 0 | Setting prohibited |
| 1 | 1 | Setting prohibited |

Table 4.7 MD8 Settings

| MD8 | EXBUS Data Bus Width |
|-----|----------------------|
| 0 | 8-bit |
| 1 | 16-bit |

Table 4.8 MD9 Settings

| MD9 | EXTAL/XTAL Pin Setting (See the Clock Pulse Generator section.) |
|-----|---|
| 0 | Inputs an external clock to the EXTAL pin |
| 1 | Connects a crystal resonator to the EXTAL/XTAL pin |

Table 4.9 MD21, MD20, MD11, MD10 and MDT[1:0] Settings

| MD21 | MD20 | MD11 | MD10 | MDT[1:0] | JTAG Selection |
|------|------|------|------|----------|---------------------------------|
| — | — | — | — | — | See the Debug and Trace section |

Table 4.10 MD12 Settings

| MD12 | Reserved, fixed to 0 |
|------|----------------------|
| — | — |

Table 4.11 MD14 and MD13 Settings

| MD14 | MD13 | Clock Frequency Selection |
|------|------|---------------------------------------|
| — | — | See the Clock Pulse Generator section |

Table 4.12 MD15 Settings

| MD15 | Initial Armv8 Execution State Selection (See the Boot section.) |
|------|---|
| 0 | AArch32 |
| 1 | AArch64 |

Table 4.13 MD16 Settings

| MD16 | Reserved, fixed to 1 |
|------|----------------------|
| — | — |

Table 4.14 MD18 Settings

| MD18 | External Bus Clock Frequency Setting (See the Clock Pulse Generator section.) |
|------|---|
| 0 | Division ratio = 1/24 of the PLL1 oscillation frequency |
| 1 | Division ratio = 1/36 of the PLL1 oscillation frequency |

Table 4.15 MD19 and MD17 Settings

| | | LPDDR4 Data Transfer Rate |
|------|------|---------------------------|
| MD19 | MD17 | LPDDR4 |
| 0 | 0 | DDR3200 |
| 0 | 1 | DDR2800 |
| 1 | 0 | Setting prohibited |
| 1 | 1 | DDR1600 |

Table 4.16 MD27 and MD22 Settings

| MD27 | MD22 | DDR Phy Pin-mux Setting |
|------|------|-------------------------|
| 0 | 0 | LPDDR4 |
| 0 | 1 | Setting prohibited |
| 1 | 0 | Setting prohibited |
| 1 | 1 | Setting prohibited |

Table 4.17 MD25 Settings

| | |
|-------------|-----------------------------|
| MD25 | Reserved, fixed to 0 |
|-------------|-----------------------------|

Table 4.18 MD23 Settings

| | |
|-------------|-----------------------------|
| MD23 | Reserved, fixed to 0 |
|-------------|-----------------------------|

Table 4.19 MD26 Settings

| | |
|-------------|-----------------------------|
| MD26 | Reserved, fixed to 0 |
|-------------|-----------------------------|

Table 4.20 MD28 Settings

| MD28 | Select RCLK Source |
|------|--------------------|
| 0 | CLK from inside |
| 1 | from EXTALR |

5. Mode Pin Settings (RZ/G2N)

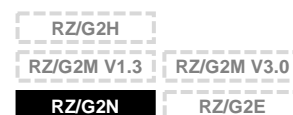


Table 5.1 TEST2 and TEST1 Settings

| Test2 | Test1 | Mode Select |
|-------|-------|--------------------|
| 0 | 0 | Normal operation |
| 0 | 1 | Setting prohibited |
| 1 | 0 | Setting prohibited |
| 1 | 1 | Setting prohibited |

Table 5.2 BSMODE Settings

| BSMODE | JTAG Pin Operating Mode Switching |
|--------|--|
| 0 | Normal operation |
| 1 | Operates in boundary scan mode. When PRESET# is at low level, pin IO control is disabled |

Table 5.3 MD0 Settings

| MD0 | Reserved, fixed to 0 |
|-----|----------------------|
| | |

Table 5.4 MD4, MD3, MD2 and MD1 Settings

| MD4 | MD3 | MD2 | MD1 | Boot Device Selection |
|-----|-----|-----|-----|-----------------------|
| — | — | — | — | See the Boot section. |

Table 5.5 MD5 Settings

| MD5 | Reserved (See the Boot section.) |
|-----|----------------------------------|
| 0 | Setting prohibited |
| 1 | Normal boot |

Table 5.6 MD7 and MD6 Settings

| MD7 | MD6 | Master Boot Processor Selection (See the Boot section.) |
|-----|-----|---|
| 0 | 0 | Cortex-A57 boot |
| 0 | 1 | Setting prohibited |
| 1 | 0 | Setting prohibited |
| 1 | 1 | Setting prohibited |

Table 5.7 MD8 Settings

| MD8 | EXBUS Data Bus Width |
|-----|----------------------|
| 0 | 8-bit |
| 1 | 16-bit |

Table 5.8 MD9 Settings

| MD9 | EXTAL/XTAL Pin Setting (See the Clock Pulse Generator section.) |
|-----|---|
| 0 | Inputs an external clock to the EXTAL pin |
| 1 | Connects the crystal resonator to the EXTAL and XTAL pin |

Table 5.9 MD21, MD20, MD11, MD10 and MDT[1:0] Settings

| MD21 | MD20 | MD11 | MD10 | MDT[1:0] | JTAG Selection |
|------|------|------|------|----------|----------------------------------|
| — | — | — | — | — | See the Debug and Trace section. |

Table 5.10 MD12 Settings

| MD12 | Select PCIE or SATA |
|------|---------------------|
| 0 | PCIexpress |
| 1 | SATA |

Table 5.11 MD14 and MD13 Settings

| MD14 | MD13 | Clock Frequency Selection |
|------|------|--|
| — | — | See the Clock Pulse Generator section. |

Table 5.12 MD15 Settings

| MD15 | Initial ARMv8 Execution State Selection (See the Boot section.) |
|------|---|
| 0 | AArch32 |
| 1 | AArch64 |

Table 5.13 MD16 Settings

| MD16 | Reserved, fixed to 1 |
|------|----------------------|
| — | — |

Table 5.14 MD18 Settings

| MD18 | External Bus Clock Frequency Setting (See the Clock Pulse Generator section.) |
|------|---|
| 0 | Division ratio = 1/24 of the PLL1 oscillation frequency |
| 1 | Division ratio = 1/36 of the PLL1 oscillation frequency |

Table 5.15 MD19 and MD17 Settings

| MD19 | MD17 | LPDDR4 Data Transfer Rate |
|------|------|---------------------------|
| 0 | 0 | DDR3200 |
| 0 | 1 | DDR2800 |
| 1 | 0 | Setting prohibited |
| 1 | 1 | DDR1600 |

Table 5.16 MD27 and MD22 Settings

| MD27 | MD22 | DDR Phy Pin-mux Setting |
|------|------|-------------------------|
| 0 | 0 | LPDDR4 |
| 0 | 1 | Setting prohibited |
| 1 | 0 | Setting prohibited |
| 1 | 1 | Setting prohibited |

Table 5.17 MD25 Settings

| MD25 | Reserved, fixed to 0 |
|------|----------------------|
| | |

Table 5.18 MD23 Settings

| MD23 | Reserved, fixed to 0 |
|------|----------------------|
| | |

Table 5.19 MD26 Settings

| MD26 | Reserved, fixed to 0 |
|------|----------------------|
| | |

Table 5.20 MD28 Settings

| MD28 | Select RCLK Source |
|------|--------------------|
| 0 | CLK from inside |
| 1 | from EXTALR |

6. Mode Pin Settings (RZ/G2E)



Table 6.1 TEST2 and TEST1 Settings

| Test2 | Test1 | Mode Select |
|-------|-------|--------------------|
| 0 | 0 | Use this setting |
| 0 | 1 | Setting prohibited |
| 1 | 0 | Setting prohibited |
| 1 | 1 | Setting prohibited |

Table 6.2 BSMODE Settings

| BSMODE | JTAG pin mode selection |
|--------|---|
| 0 | Normal operation |
| 1 | Boundary scan operation. Pin IO control during PRESET#=Low level is disabled. |

Table 6.3 MD0 Settings

| MD0 | Reserved, fixed to 0 |
|-----|----------------------|
|-----|----------------------|

Table 6.4

MD4, MD3, MD2 and MD1 Settings

| MD4 | MD3 | MD2 | MD1 | Boot Device Selection |
|-----|-----|-----|-----|-----------------------|
| — | — | — | — | See the Boot section |

Table 6.5 MD5 Settings

| MD5 | Reserved |
|-----|--------------------|
| 0 | Setting prohibited |
| 1 | Normal boot |

Table 6.6 MD7 and MD6 Settings

| MD7 | MD6 | Description |
|-----|-----|-----------------------------------|
| 0 | 0 | Setting prohibited |
| 0 | 1 | Booted through CPU0 in Cortex-A53 |
| 1 | 0 | Setting prohibited |
| 1 | 1 | Setting prohibited |

Table 6.7 MD8 Settings

| MD8 | Reserved, fixed to 1 |
|-----|----------------------|
| | |

Table 6.8 MD9 Settings

| MD9 | EXTAL/XTAL Pin Settings |
|-----|--|
| 0 | Inputs an external clock to the EXTAL pin. |
| 1 | Connects the crystal resonator to the EXTAL and XTAL pins. |

Table 6.9 MD12 Settings

| MD12 | System Clock SSCG/nonSSCG select |
|------|----------------------------------|
| 0 | SSCG OFF |
| 1 | SSCG ON |

Table 6.10 MD13 Settings

| MD13 | VDDQ_QSPI voltage indication (should meet with input voltage) |
|------|---|
| 0 | 1.8V |
| 1 | 3.3V |

Table 6.11 MD15 Settings

| MD15 | Description |
|------|-------------|
| 0 | AArch32 |
| 1 | AArch64 |

Table 6.12 MD16 Settings

| MD16 | Reserved, fixed to 1 |
|------|----------------------|
| | |

Table 6.13 MD17 Settings

| MD17 | Reserved, fixed to 0 |
|------|----------------------|
| 0 | Use this setting |
| 1 | Reserved |

Table 6.14 MD19 Settings

| MD19 | EXTAL Input Frequency [MHz] | Divider Setting of EXTAL Input | PLL Multiplication Ratio | | | |
|------|--------------------------------------|---|--------------------------|--------------------|---------------|--------------|
| | | | PLL0 (nonSSCG) | PLL1 (SYS SSCG) | PLL3 (DDR) | |
| 0 | 48 | × 1 | × 100.0 | × 33.3 | × 33.3 | DDR:1600Mbps |
| 1 | 48 | × 1 | × 100.0 | × 33.3 | × 19.3 | DDR:1856Mbps |

Table 6.15 MD23 Settings

| | |
|-------------|-----------------------------|
| MD23 | Reserved, fixed to 0 |
|-------------|-----------------------------|

Table 6.16 MD25 Settings

| | |
|-------------|-----------------------------|
| MD25 | Reserved, fixed to 0 |
|-------------|-----------------------------|

Table 6.17 MD26 Settings

| | |
|-------------|-----------------------------|
| MD26 | Reserved, fixed to 0 |
|-------------|-----------------------------|

7. AP-System Core



| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

7.1 Overview

RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0

The AP-System core is a core block equipped with Arm Cortex-A53 and Arm Cortex-A57. For details on the functions of Cortex-A53 and Cortex-A57, see the relevant Technical Reference Manual.

RZ/G2E

The AP-System core is a core block equipped with Arm Cortex-A53. For details on the functions of Cortex-A53, see the relevant Technical Reference Manual.

RZ/G2N

The AP-System core is a core block equipped with Arm Cortex-A57. For details on the functions of Cortex-A57, see the relevant Technical Reference Manual.

7.1.1 Features

7.1.1.1 Cortex-A57 CPU cluster

Table 7.1 shows the feature of Cortex-A57 CPU cluster in AP-System Core.

Table 7.1 Feature of the AP-System Core Block (Cortex-A57)

| | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----------------------------------|--|--|--|--------|
| Version | r1p3 | r1p3 | r1p3 | — |
| Cluster Number | Cluster 0 | Cluster 0 | Cluster 0 | — |
| Number of CPUs | Quad CPU | Dual CPU | Dual CPU | — |
| L1 cache | I/ 48 KB (16 KB × 3 way) [w/Parity] D/32 KB (16 KB × 2 way) [w/ECC] | I/ 48 KB (16 KB × 3 way) [w/Parity] D/32 KB (16 KB × 2 way) [w/ECC] | I/ 48 KB (16 KB × 3 way) [w/Parity] D/32 KB (16 KB × 2 way) [w/ECC] | — |
| L2 cache | 2 MB (128 KB × 16 way) [w/ECC] | 1MB (64KB × 16 way) [w/ECC] | 1MB (64KB × 16 way) [w/ECC] | — |
| ECC | Supported | Supported | Supported | — |
| MMU | Supported | Supported | Supported | — |
| NEON/VFP | NEON supported (Advanced SIMD) VFP supported | | NEON supported (Advanced SIMD) VFP supported | — |
| Cryptography Extension (Optional) | Supported | Supported | Supported | — |
| Operating clock | CPU core: Zφ L2-Tag RAMs: 3-divided clock of Zφ L2-Data RAMs: 4-divided clock of Zφ ACE bus: ZXφ | CPU core: Zφ | CPU core: Zφ L2-Tag RAMs: 3-divided clock of Zφ L2-Data RAMs: 4-divided clock of Zφ ACE bus: ZXφ | — |
| Low power mode | Sleep mode (each CPU) Core standby mode (each CPU) AP-system core power down mode (CPUs and L2 cache shutdown: A2BS = OFF) | | Sleep mode (each CPU) Core standby mode (each CPU) AP-system core power down mode (CPUs and L2 cache shutdown: A2BS = OFF) | — |
| Private peripherals | Arm Generic Timer supported | | Arm Generic Timer supported | — |

7.1.1.2 Cortex-A53 CPU Cluster

Table 7.2 shows the feature of Cortex-A53 CPU cluster in AP-System Core.

Table 7.2 Features of the AP-System Core Block (Cortex-A53)

| | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--|--|---|--------|--|
| Version | r0p4 | r0p4 | — | r0p4 |
| Cluster number | Cluster 1 | Cluster 1 | — | Cluster 0 |
| Number of CPUs | Quad CPU | Quad CPU | — | Dual CPU |
| L1 cache (per CPU) | I/ 32 KB (16 KB × 2 way) [w/Parity] D/32 KB (8 KB × 4 way) [w/ECC] | I/ 32 KB (16 KB × 2 way) [w/Parity] D/32 KB (8 KB × 4 way) [w/ECC] | — | I/ 32 KB (16 KB × 2 way) [w/Parity] D/32 KB (8 KB × 4 way) [w/ECC] |
| L2 cache | 512 KB (32 KB × 16 way) [w/ECC] | 512 KB (32 KB × 16 way) [w/ECC] | — | 256 KB (16 KB × 16 way) [w/ECC] |
| ECC | Supported | Supported | — | Supported |
| MMU | Supported | Supported | — | Supported |
| NEON/VFP | NEON supported (Advanced D) VFP supported | | — | NEON supported (Advanced D) VFP supported |
| Cryptography Extension (Optional) | Supported | Supported | — | Supported |
| Operating clock | CPU core: Z2φ L2-Tag RAMs: Z2φ L2-Data RAMs: 2-divided clock of Z2φ ACE bus: ZXφ | CPU core: Z2φ ACE bus: ZXφ | — | CPU core: Z2φ L2-Tag RAMs: Z2φ L2-Data RAMs: 2-divided clock of Z2φ AXI4 bus: ZXφ |
| Low power mode | Sleep mode (each CPU) Core standby mode (each CPU) AP-system core power down mode (CPUs and L2 cache shutdown: A2LS = OFF) | | — | Sleep mode (each CPU) Core standby mode (each CPU) AP-system core power down mode (CPUs and L2 cache shutdown: A2LS = OFF) |
| Private peripherals | Arm Generic Timer supported | | — | Arm Generic Timer supported |

7.1.1.3 Cache Coherent Interconnect (CCI-500)

RZ/G2H **RZ/G2M V1.3** **RZ/G2M V3.0**

Cache coherent interconnect combines interconnect and coherency function. It can connect two ACE masters, one ACE-Lite masters and three AXI4 slaves.

Table 7.3 shows the module connection of each ports. In order to improve the performance of the coherency access, snoop filter is implemented. For details, refer to the following document.

RZ/G2H

“CoreLink CCI-500 Cache Coherent Interconnect Revision r1p0 Technical Reference Manual”

(corelink_cci500_cache_coherent_interconnect_technical_reference_manual_100023_0100_00_en.pdf). 3.4 Address map

RZ/G2M V1.3 **RZ/G2M V3.0**

“CoreLink CCI-500 Cache Coherent Interconnect Revision r0p1 Technical Reference Manual”

(corelink_cci500_cache_coherent_interconnect_technical_reference_manual_100023_0001_00_en.pdf). 3.4 Address map

RZ/G2N **RZ/G2E**

The bus interconnect is connected to Cortex-A53.

Table 7.3 module connection of each ports

| | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|----------------------|------------------------|--|---------------|---------------|
| Slave port 0 | Main Memory Domain AXI | Main Memory Domain AXI | — | — |
| Slave port 1 | Cortex-A57 | Cortex-A57 | — | — |
| Slave port 2 | Cortex-A53 | Cortex-A53 | — | — |
| Slave port 3 | — | — | — | — |
| Master port 0 | Peripheral Domain AXI | Peripheral Domain AXI | — | — |
| Master port 1 | Main Memory Domain AXI | Main Memory Domain AXI | — | — |
| Master port 2 | Main Memory Domain AXI | Main Memory Domain AXI | — | — |

7.1.1.4 Generic Counter

For the general explanation about Generic Counter, and the details of the relation between Generic Counter and Generic Timer, refer to the Arm manual. In this chapter, it is explained that the frequency for count-up of the Generic Counter. The Generic Counter serves as the CoreSight Timestamp function as well as Generic Timer.

7.1.2 Block Diagram

Figure 7.1, Figure 7.2 and Figure 7.3 show the block diagram of AP-System Core.

RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0

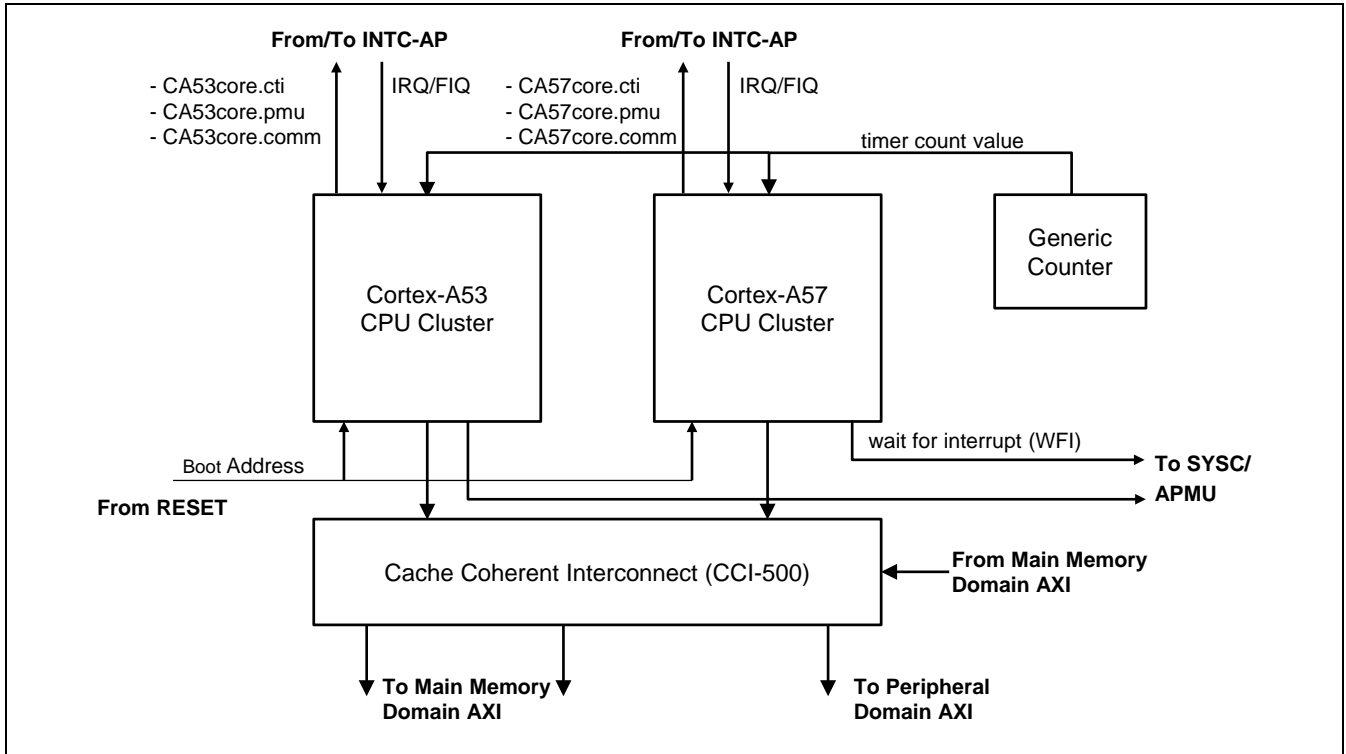


Figure 7.1 AP-System Core block diagram with Cortex-A57, Cortex-A53 and CCI-500

RZ/G2E

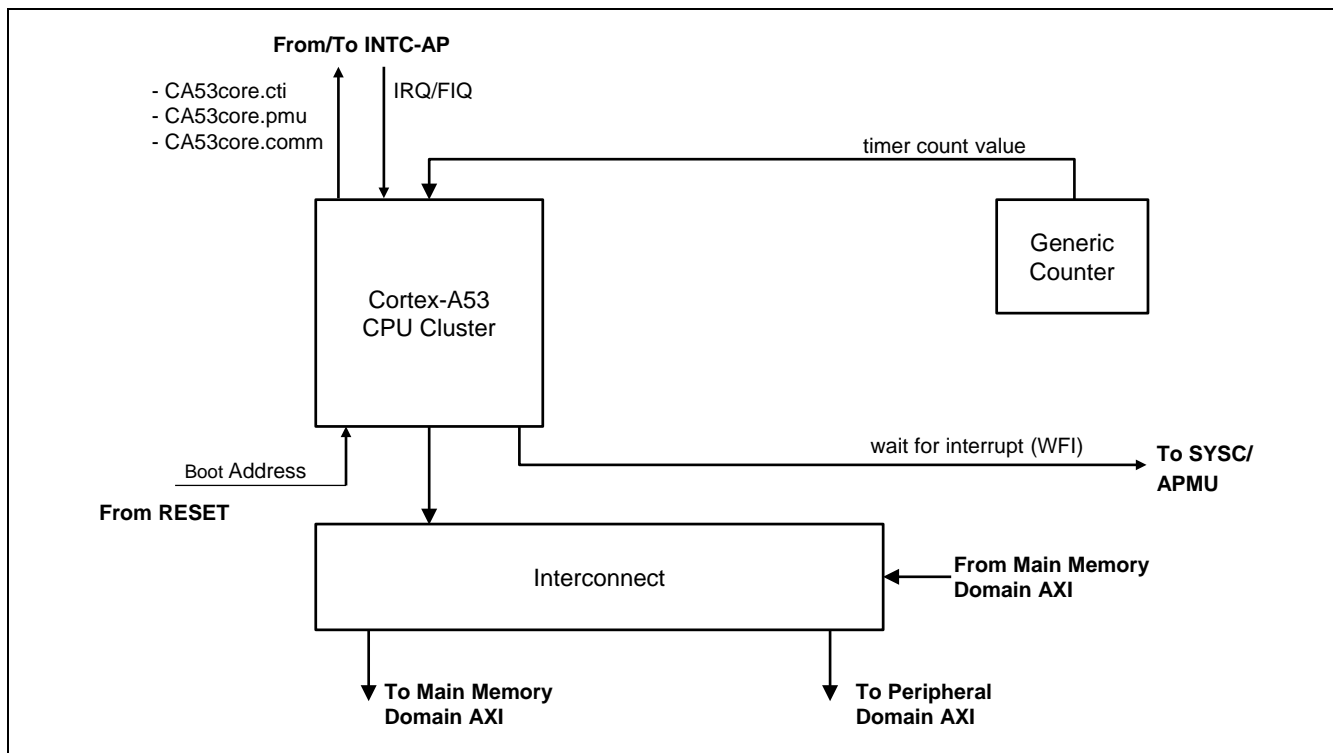


Figure 7.2 AP-System Core block diagram with only Cortex-A53

RZ/G2N

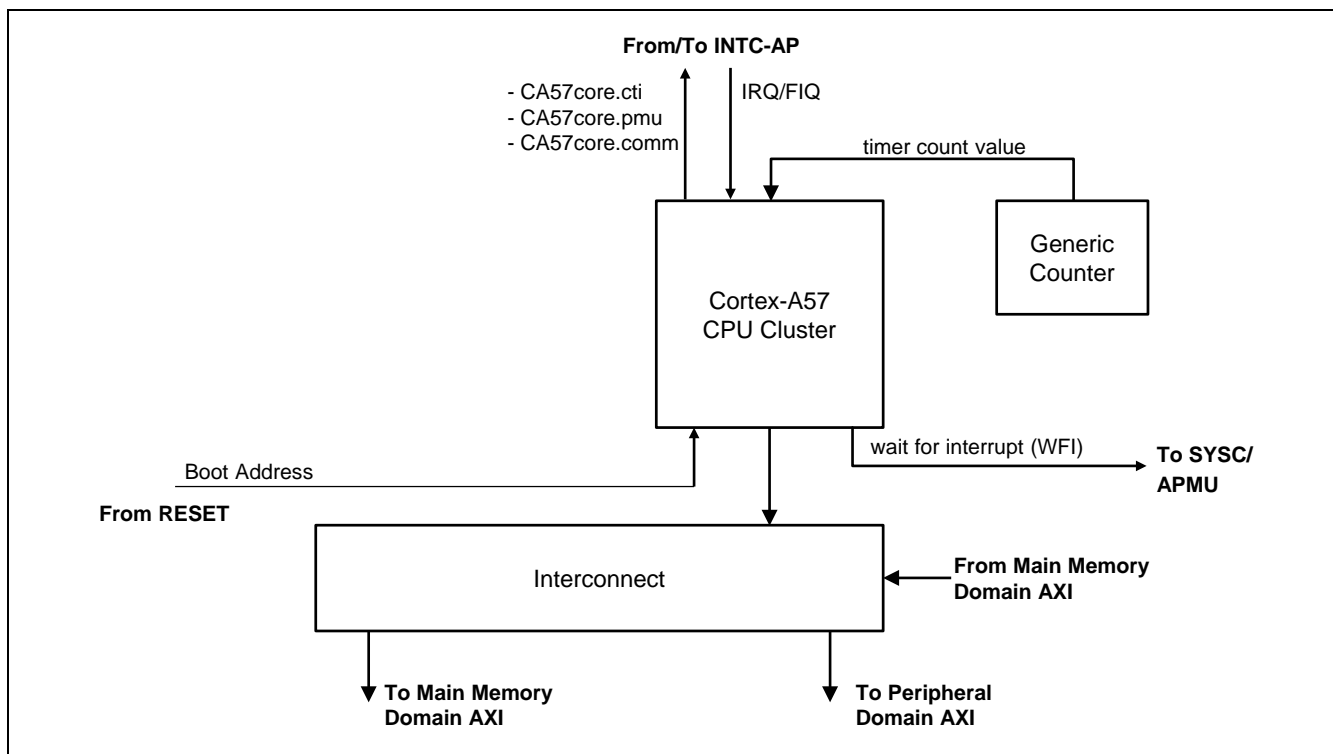


Figure 7.3 AP-System Core block diagram with only Cortex-A57

7.1.3 External Pins

In AP-System Core, there are no external pins.

7.1.4 Register Configuration

Table 7.4, Table 7.5 and Table 7.6 show the register configuration of CCI-500 and Generic Counter.

Table 7.4 Register Configuration (CCI-500)

| Name | Abbreviation | R/W | Address | Initial Value | Size | Second Generation RZ/G Series Products | | | |
|---|--------------|-----|-----------------------------|---|------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Control Override Register | — | R/W | H'F120_0000 | H'0000_0000 | 32 | √ | √ | — | — |
| Secure Access Register | — | —/W | H'F120_0008 | H'0000_0000 | 32 | √ | √ | — | — |
| Status Register | — | R | H'F120_000C | H'0000_0000 | 32 | √ | √ | — | — |
| Imprecise Error Register | — | R/W | H'F120_0010 | H'0000_0000 | 32 | √ | √ | — | — |
| Performance Monitor Control Register | — | R/W | H'F120_0100 | H'0000_4000 | 32 | √ | √ | — | — |
| Interface Monitor Control Register | — | R/W | H'F120_0104 | H'0000_0000 | 32 | √ | √ | — | — |
| Peripheral ID 0 | — | R | H'F120_0FE0 | H'0000_0022 | 32 | √ | √ | — | — |
| Peripheral ID 1 | — | R | H'F120_0FE4 | H'0000_00B4 | 32 | √ | √ | — | — |
| Peripheral ID 2 | — | R | H'F120_0FE8 | H'0000_000B | 32 | √ | √ | — | — |
| Peripheral ID 3 | — | R | H'F120_0FEC | H'0000_0000 | 32 | √ | √ | — | — |
| Peripheral ID 4 | — | R | H'F120_0FD0 | H'0000_0084 | 32 | √ | √ | — | — |
| Peripheral ID 5 | — | R | H'F120_0FD4 | H'0000_0000 | 32 | √ | √ | — | — |
| Peripheral ID 6 | — | R | H'F120_0FD8 | H'0000_0000 | 32 | √ | √ | — | — |
| Peripheral ID 7 | — | R | H'F120_0FDC | H'0000_0000 | 32 | √ | √ | — | — |
| Component ID 0 | — | R | H'F120_0FF0 | H'0000_000D | 32 | √ | √ | — | — |
| Component ID 1 | — | R | H'F120_0FF4 | H'0000_00F0 | 32 | √ | √ | — | — |
| Component ID 2 | — | R | H'F120_0FF8 | H'0000_0005 | 32 | √ | √ | — | — |
| Component ID 3 | — | R | H'F120_0FFC | H'0000_00B1 | 32 | √ | √ | — | — |
| Snoop Control Register SIn*1 | — | R/W | H'F120_1000 + H'1000 * n | SI0: H'0000_0000 SI1: H'8000_0000 SI2: H'C000_0000 | 32 | √ | — | — | — |
| Snoop Control Register SIn*1 | — | R/W | H'F120_1000 + H'1000 * n | SI0: H'0000_0000 SI1/2: H'C000_0000 | 32 | — | √ | — | — |
| Shareable Override Register SIn*1 | — | R/W | H'F120_1004 + H'1000 * n | H'0000_0000 | 32 | √ | √ | — | — |
| Read QoS Override Register SIn*1 | — | R/W | H'F120_1100 + H'1000 * n | H'0000_0000 | 32 | √ | √ | — | — |

| Name | Abbreviation | R/W | Address | Initial Value | Size | Second Generation RZ/G Series Products | | | |
|-----------------------------------|--------------|-----|-------------------------------|---------------|------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Write QoS Override Register SIn*1 | — | R/W | H'F120_1104 + H'1000 * n | H'0000_0000 | 32 | √ | √ | — | — |
| MAX OT Register SIn*1 | — | R/W | H'F120_1110 + H'1000 * n | H'0000_0020 | 32 | √ | √ | — | — |
| Event Select n*1 | — | R/W | H'F121_0000 + H'1_0000 * n | H'0000_0000 | 32 | √ | √ | — | — |
| Event Count n*1 | — | R/W | H'F121_0004 + H'1_0000 * n | H'0000_0000 | 32 | √ | √ | — | — |
| Count Control n*1 | — | R/W | H'F121_0008 + H'1_0000 * n | H'0000_0000 | 32 | √ | √ | — | — |
| Count Overflow n*1 | — | R/W | H'F121_000C + H'1_0000 * n | H'0000_0000 | 32 | √ | √ | — | — |
| Interface Monitor Register SIn*1 | — | R | H'F129_0000 + H'0004 * n | H'0000_0000 | 32 | √ | √ | — | — |
| Interface Monitor Register MIn*2 | — | R | H'F129_0100 + H'0004 * n | H'0000_0000 | 32 | √ | √ | — | — |

Notes: 1. SIn: (n = 0-2)

2. MIn (n = 0-2)

For details, refer to CoreLink CCI-500 Cache Coherent Interconnect Revision r0p0 (for RZ/G2M V1.3, RZ/G2M V3.0) and r1p0 (for RZ/G2H) Technical Reference Manual (corelink_cci500_cache_coherent_interconnect_technical_reference_manual_100023_0000_01_en.pdf).

Table 7.5 Register Configuration (Generic Counter – Control Base)

| Name | Abbreviation | R/W | Address | Initial Value | Size | Second Generation RZ/G Series Products | | | |
|------------------------------------|--------------|-----|-------------|---------------|------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Counter Control Register | CNTCR | R/W | H'E608_0000 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Counter Status Register | CNTSR | R | H'E608_0004 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Current Count Value Lower register | CNTCVL | R/W | H'E608_0008 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Current Count Value Upper register | CNTCVU | R/W | H'E608_000C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Base Frequency ID register | CNTFID0 | R/W | H'E608_0020 | H'00C6_5D40 | 32 | √ | √ | √ | √ |
| Counter ID Register 0 | CounterID0 | R | H'E608_0FD0 | H'0000_0004 | 32 | √ | √ | √ | √ |
| Counter ID Register 1 | CounterID1 | R | H'E608_0FD4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Counter ID Register 2 | CounterID2 | R | H'E608_0FD8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Counter ID Register 3 | CounterID3 | R | H'E608_0FDC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Counter ID Register 4 | CounterID4 | R | H'E608_0FE0 | H'0000_0001 | 32 | √ | √ | √ | √ |
| Counter ID Register 5 | CounterID5 | R | H'E608_0FE4 | H'0000_00B1 | 32 | √ | √ | √ | √ |
| Counter ID Register 6 | CounterID6 | R | H'E608_0FE8 | H'0000_001B | 32 | √ | √ | √ | √ |
| Counter ID Register 7 | CounterID7 | R | H'E608_0FEC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Counter ID Register 8 | CounterID8 | R | H'E608_0FF0 | H'0000_000D | 32 | √ | √ | √ | √ |
| Counter ID Register 9 | CounterID9 | R | H'E608_0FF4 | H'0000_00F0 | 32 | √ | √ | √ | √ |
| Counter ID Register 10 | CounterID10 | R | H'E608_0FF8 | H'0000_0005 | 32 | √ | √ | √ | √ |
| Counter ID Register 11 | CounterID11 | R | H'E608_0FFC | H'0000_00B1 | 32 | √ | √ | √ | √ |

Notes: 1. For details, refer to DDI0487A_b_armv8_arm.pdf, Appendix I1 System Level Implementation of the Generic Timer.

2. Registers in Control Base region can be accessed only from secure world.

Table 7.6 Register Configuration (Generic Counter – Read Base)

| Name | Abbreviation | R/W | Address | Initial Value | Size | Second Generation RZ/G Series Products | | | |
|---------------------------------------|--------------|-----|-------------|---------------|------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Current Count Value Lower register | CNTCVL | R/W | H'E608_1000 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Current Count Value Upper register | CNTCVU | R/W | H'E608_1004 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Counter ID Register 0 | CounterID0 | R | H'E608_1FD0 | H'0000_0004 | 32 | √ | √ | √ | √ |
| Counter ID Register 1 | CounterID1 | R | H'E608_1FD4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Counter ID Register 2 | CounterID2 | R | H'E608_1FD8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Counter ID Register 3 | CounterID3 | R | H'E608_1FDC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Counter ID Register 4 | CounterID4 | R | H'E608_1FE0 | H'0000_0001 | 32 | √ | √ | √ | √ |
| Counter ID Register 5 | CounterID5 | R | H'E608_1FE4 | H'0000_00B1 | 32 | √ | √ | √ | √ |
| Counter ID Register 6 | CounterID6 | R | H'E608_1FE8 | H'0000_001B | 32 | √ | √ | √ | √ |
| Counter ID Register 7 | CounterID7 | R | H'E608_1FEC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Counter ID Register 8 | CounterID8 | R | H'E608_1FF0 | H'0000_000D | 32 | √ | √ | √ | √ |
| Counter ID Register 9 | CounterID9 | R | H'E608_1FF4 | H'0000_00F0 | 32 | √ | √ | √ | √ |
| Counter ID Register 10 | CounterID10 | R | H'E608_1FF8 | H'0000_0005 | 32 | √ | √ | √ | √ |
| Counter ID Register 11 | CounterID11 | R | H'E608_1FFC | H'0000_00B1 | 32 | √ | √ | √ | √ |

Note: For details, refer to DDI0487A_b_armv8_arm.pdf, Appendix I1 System Level Implementation of the Generic Timer.

7.1.5 Connected Module

Table 7.7 shows the connected modules to AP-System Core. As for the feature of the connected modules, refer to the chapter of each modules.

Table 7.7 Connected Modules

| Module Name | Connected Module Name | Connected Module Function |
|--------------------|------------------------------|--|
| AP-System Core | APMU | Power On/Off sequence Clock supply |
| | SYSC | Power On/Off sequence |
| | RST | Boot address mode setting CPU reset control |
| | INTC-AP | Interrupt handling |
| | Debug and Trace | Debug function |

7.2 Register Description

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

As for register description of AP-System Core, refer to the following documents.

- **RZ/G2H** CoreLink CCI-500 Cache Coherent Interconnect Revision r1p0 Technical Reference Manual
- **RZ/G2M V1.3** **RZ/G2M V3.0** CoreLink CCI-500 Cache Coherent Interconnect Revision r0p0 Technical Reference Manual
- Arm Architecture Reference Manual Armv8, for Armv8-A architecture profile

7.3 Operation

7.3.1 Boot Address Setting

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0

In AArch32 mode, boot address can be controlled by CA57/53BAR and CA57/53BAR2 register in RESET(RST). In AArch64 mode, boot address can also be controlled by CA57/53CPUnBARH/L registers in RST. About the initial value of these registers, refer to section 17, RESET(RST).

RZ/G2E

In AArch32 mode, boot address can be controlled by CA53BAR and CA53BAR2 register in RESET(RST). In AArch64 mode, boot address can also be controlled by CA53CPUnBARH/L registers in RST. About the initial value of these registers, refer to section 17, RESET(RST).

RZ/G2N

In AArch32 mode, boot address can be controlled by CA57BAR and CA57BAR2 register in RESET(RST). In AArch64 mode, boot address can also be controlled by CA57CPUnBARH/L registers in RST. About the initial value of these registers, refer to section 17, RESET(RST).

7.3.2 Power-Up/Down Mechanism

The AP-System core supports the following five power modes.

- AP-System core power down mode

RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0

(CPUs and L2 Cache shutdown: A2BS = OFF (Cortex-A57)/A2LS = OFF (Cortex-A53))

RZ/G2E

(CPUs and L2 Cache shutdown: A2LS = OFF (Cortex-A53))

RZ/G2N

(CPUs and L2 Cache shutdown: A2BS = OFF (Cortex-A57))

- Core standby mode (each CPU)
- Sleep mode (clock stop, each CPU)
- L2 shutdown mode
- Normal Operation mode

RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E

In AP-System core power down mode, the power of the AP-System core area is shut down. This mode is intended to be used at long-time wait, and achieves the reduction of power consumption including leakage current cut-off. The system boots up from the reset vector at its return from AP-System core power down mode. For detail of the AP-system core power down control, see section 14, System Controller (SYSC).

In Core standby mode, all memories including L1 cache of the relevant CPU do not keep its contents, but the power of the SRAM of level 2 cache is supplied to keep its contents.

In L2 shutdown mode, all power of CPUs and L2 in cluster is shut down. So contents of both L1s and L2 are not retained.

For details of Core standby mode and L2 shutdown mode, see section 13, Advanced Power Management Unit for AP-System Core (APMU).

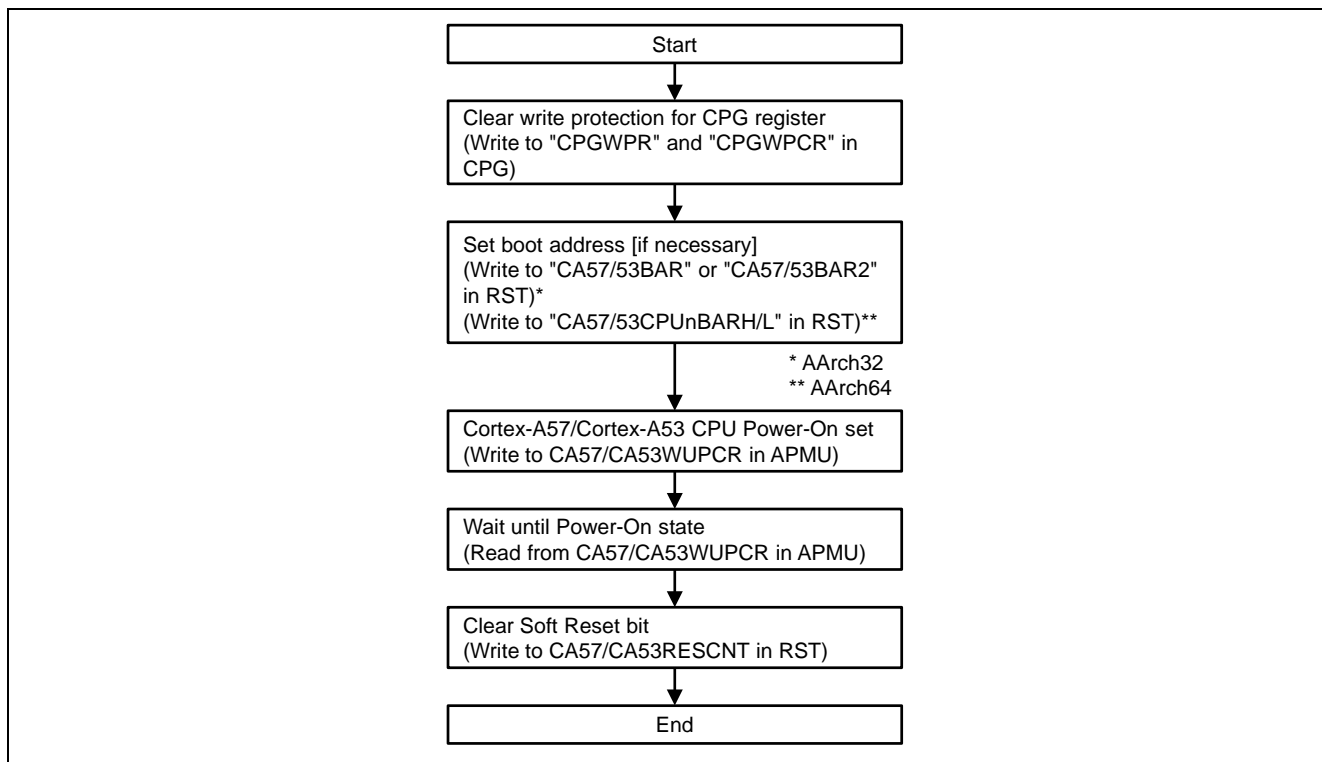


Figure 7.4 Cortex-A57/Cortex-A53 CPU core power-on sequence

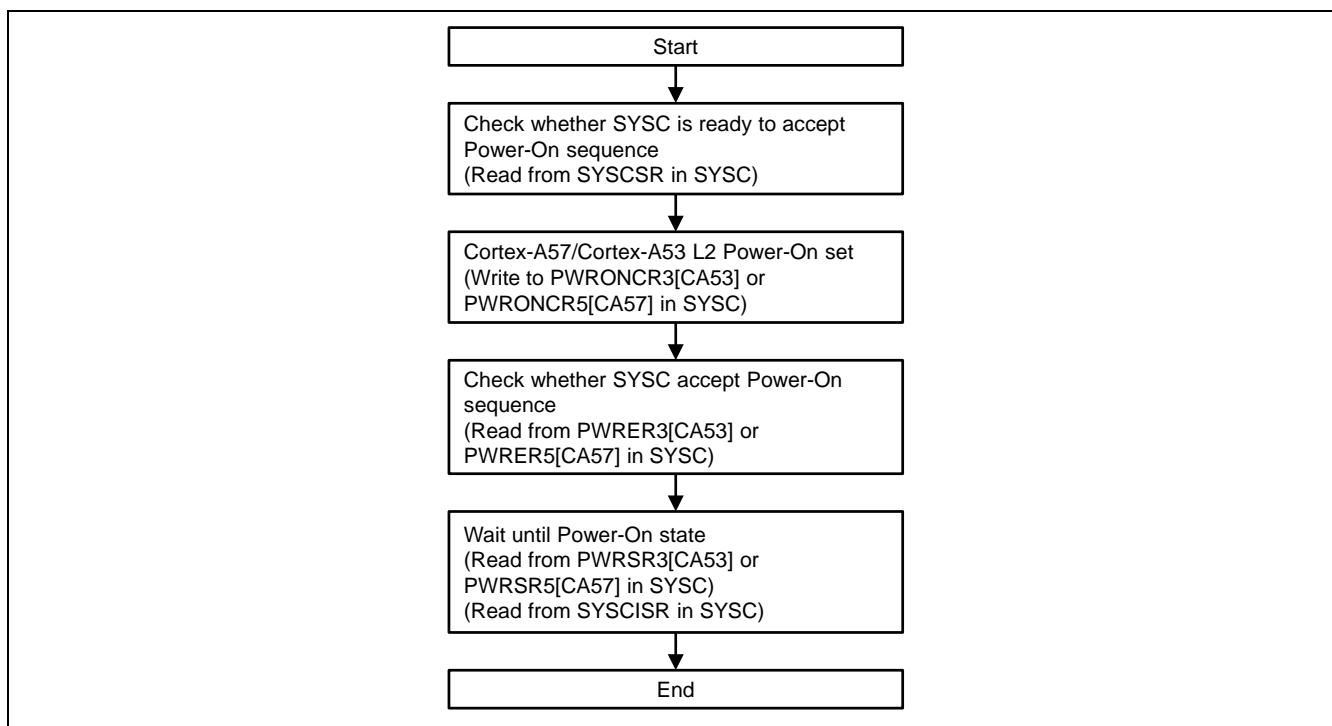


Figure 7.5 Cortex-A57/Cortex-A53 L2 power-on sequence

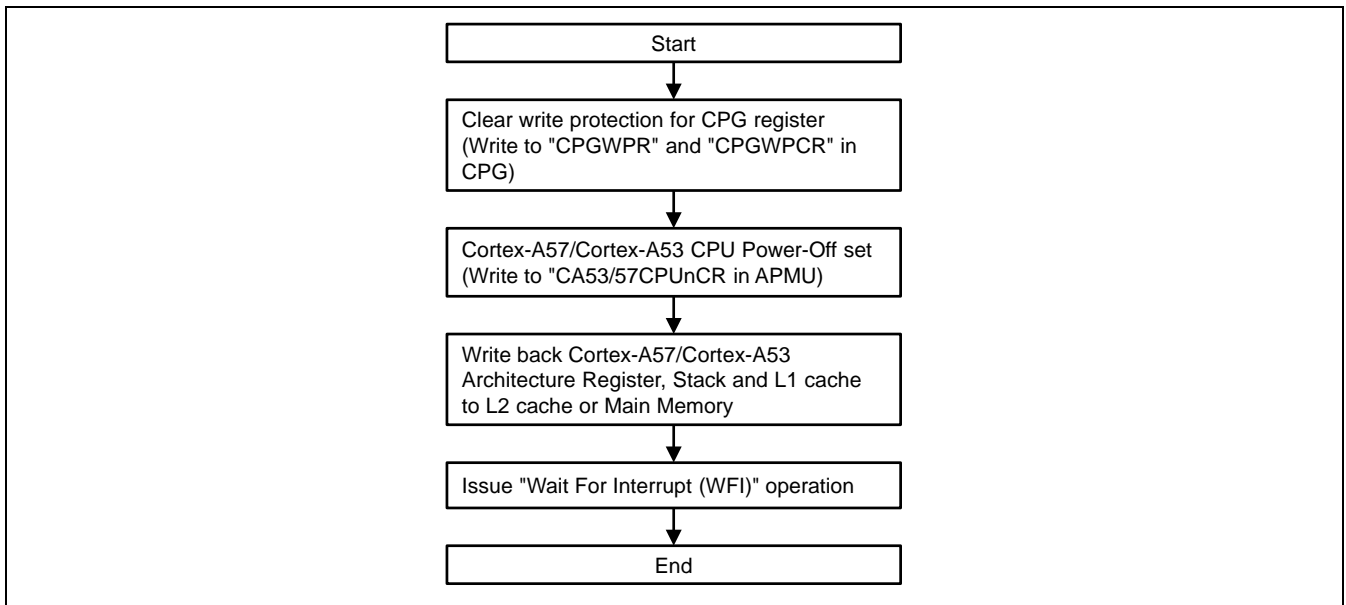


Figure 7.6 Cortex-A57/Cortex-A53 CPU core power-off sequence

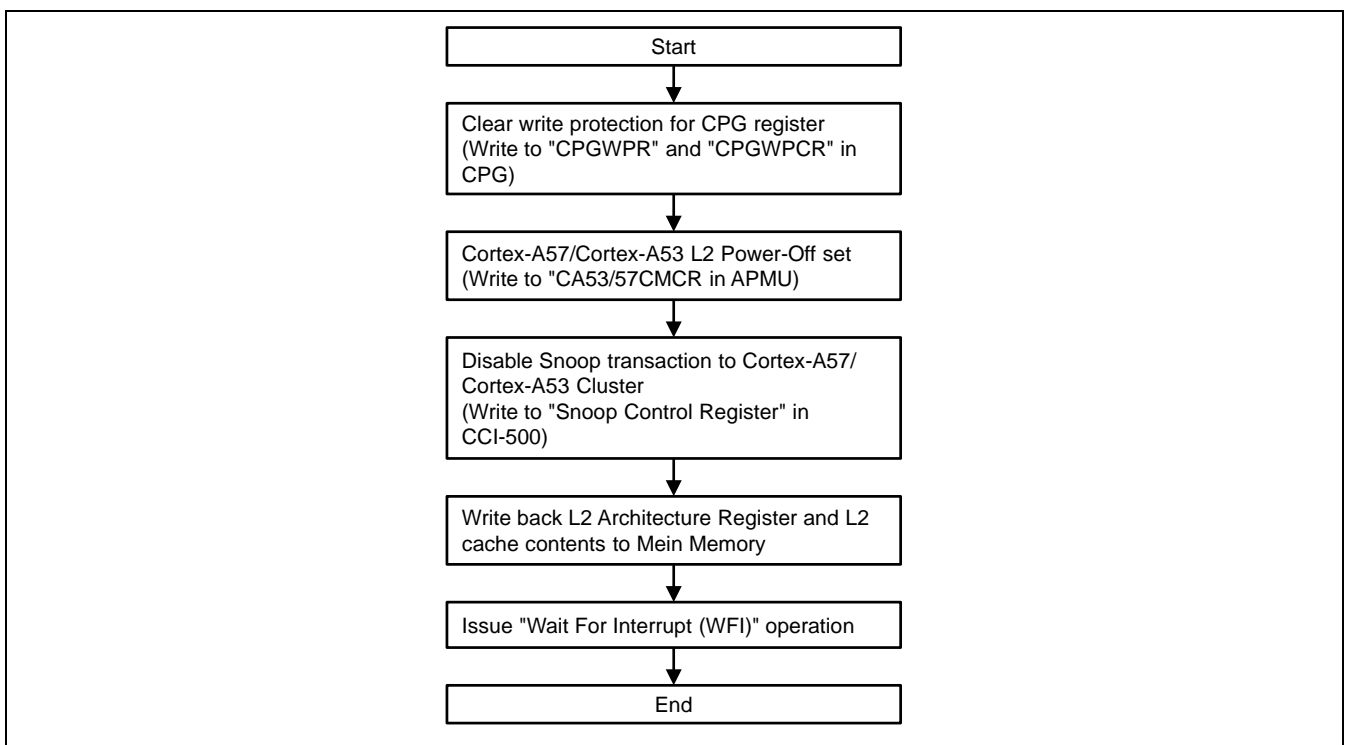


Figure 7.7 Cortex-A57/Cortex-A53 L2 power-off sequence

7.3.3 Switching between AArch32 and AArch64

Each CPU in AP-System core can be switched between AArch32 state and AArch64 state using warm reset. In order to switch between AArch32 and AArch64 safely, it is recommended to use the below sample code. In addition, interrupts and debug requests for this core should be disabled in advance.

```
MOV   Wy, #3           ; #3 : for AArch64, #2 : for AArch32, y : any register
DSB                               ; ensure all memory transactions are completed
MSR   RMR_ELx, Wy      ; request the reset
ISB                               ; synchronization for changing to RMR
loop
WFI                               ; in order to issue warm reset
B     loop
```

Note: Refer to the “Arm Architecture reference manual” in detail.
In case of Cortex-A53 and Cortex-A57, RMR_ELx is RMR_EL3

7.3.4 Maximum Outstanding Setting of CCI-500

CCI-500 can adjust maximum number of outstanding request as shown in Table 7.8. Set the same or lower value listed in Table 7.8.

Table 7.8 Maximum Number of Outstanding Request (CCI-500)

| | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------------|------------|--------|----------------------------|--------|--------|
| Slave port 0 | Read/Write | 32 | 32 | — | — |
| Slave port 1 | Read/Write | 32 | 32 | — | — |
| Slave port 2 | Read/Write | 32 | 32 | — | — |
| Slave port 3 | Read/Write | 32 | — | — | — |

7.3.5 System Address Map

RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0

The CCI-500 has the system address map. The address spaces defined in this map are split into 11 regions. The connection to each master port is defined by the value of ADDRMAP[32:0] in advance.

Table 7.9 shows System Address MAP of CCI-500. For the detail of address map, see the following document.

RZ/G2H

“CoreLink CCI-500 Cache Coherent Interconnect Revision r1p0 Technical Reference Manual”

(corelink_cci500_cache_coherent_interconnect_technical_reference_manual_100023_0100_00_en.pdf). 3.4 Address map

RZ/G2M V1.3 RZ/G2M V3.0

“CoreLink CCI-500 Cache Coherent Interconnect Revision r0p1 Technical Reference Manual”

(corelink_cci500_cache_coherent_interconnect_technical_reference_manual_100023_0001_00_en.pdf). 3.4 Address map

Table 7.9 System Address Map for CCI-500

| Address Region | Address Range | Destination |
|----------------|----------------------------------|---|
| REGION 0 | H'00_0000_0000 to H'00_3FFF_FFFF | Master port 0 (Peripheral) |
| REGION 1 | H'00_4000_0000 to H'00_BFFF_FFFF | Master port 1/2 (Main Memory) [4KB split] |
| REGION 2 | H'00_C000_0000 to H'00_DFFF_FFFF | Master port 0 (Peripheral) |
| REGION 3 | H'00_E000_0000 to H'00_E62F_FFFF | Master port 0 (Peripheral) |
| REGION 4 | H'00_E630_0000 to H'00_E63F_FFFF | Master port 1/2 (System RAM) [4KB split] |
| REGION 5 | H'00_E640_0000 to H'00_FFFF_FFFF | Master port 0 (Peripheral) |
| REGION 6 | H'01_0000_0000 to H'0F_FFFF_FFFF | Master port 1/2 (Main Memory) [4KB split] |
| REGION 7 | H'10_0000_0000 to H'14_FFFF_FFFF | Master port 0 (Peripheral) |
| REGION 8 | H'15_0000_0000 to H'1E_FFFF_FFFF | Master port 0 (Peripheral) |
| REGION 9 | H'1F_0000_0000 to H'1F_FFFF_FFFF | Master port 0 (Peripheral) |
| REGION 10 | H'20_0000_0000 to H'FF_FFFF_FFFF | Master port 0 (Peripheral) |

7.3.6 Frequency for Counting Up of Generic Counter

The generic timer counts up by one every clock edge of the following frequency continuously.

Table 7.10 Frequency for count-up

| | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------------------------|-------------|----------------------------|-------------|-------------|
| Count-up clock frequency | EXTAL × 1/2 | EXTAL × 1/2 | EXTAL × 1/2 | EXTAL × 1/2 |

The operating frequency of EXTAL is decided by mode pin setting. For details, see section 8.1.5 Clock Operating Modes, PLL Initial Multiplication Ratio description.

7.3.7 How to Start a Generic Counter

Set bit 0 in CNTCR to 1 to start the Generic Counter when the CPU is placed in the secure mode.

7.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

7.4.1 L2 Cache Setting for Cortex-A57

Notes for L2 cache setting for Cortex-A57 are follows:

- The default value of “Tag RAM latency” is 2 cycles. You must change it to 3 cycles by accessing L2CTLR.
- The default value of “Data RAM setup” is 0 cycles. In case of RZ/G2H, you must change it to 1 cycle by accessing L2CTLR.
- The default value of “Data RAM latency” is 2 cycles. You must change it to 4 cycles by accessing L2CTLR.

Table 7.11 shows the L2 cache setting for Cortex-A57

Table 7.11 L2 Cache Setting for Cortex-A57

| | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------------------------|--------|----------------------------|--------|--------|
| Tag RAM latency (cycle) | 3 | 3 | 3 | — |
| Data RAM latency (cycle) | 4 | 4 | 4 | — |
| Data RAM setup (cycle) | 1 | 0 | 0 | — |

7.4.2 Memory Attribute

The following is restriction on the memory attributes in the case of the memory write access. If you ignore the restriction, the write access will cause unexpected results. There is not restriction in case of the read access.

External memories accessed via LBSC

The memory attribute must be the below.

- Device-nGnRnE(default)({TEX,C,B} = B'0_0000)(*).
- Shareable/Non-Shareable Device-nGnRE({TEX,C,B} = B'0_0001 or B'0_1000)(*).

Note: * In case that “Memory Region Remap” is enabled in OS, the bit pattern of {TEX,C,B} is different.

8. Pin Function Controller (PFC)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

8.1 Overview

The pin function controller (PFC) is a module that consists of registers for selecting the function of the multiplexed pins and controlling the pull-up resistor on each LSI pin.

8.1.1 Features

- Setting multiplexed pin functions for LSI pins

Function of the RZ/G 2nd Generation pin selectable by setting the registers in the PFC module

(The function of the LSI pin can be selected by the GPIO/peripheral function select registers 0 to 7 (GPSR0 to GPSR7) and peripheral function select registers 0 to 18 (IPSR0 to IPSR18) in the PFC module. For details, see sections 8.2.2, GPIO/Peripheral Function Select Register (GPSR0-7) through 8.2.3, Peripheral Function Select Register 0-18 (IPSR0-18).)

- DRV control for each LSI pin

DRV control resistors can control the driving abilities of pins.

DRV control resistors on each LSI pin can be controlled by setting the registers in the PFC module. (Selection is handled by the output drive select register (DRVCTRL0-24). For details, see sections 8.2.4, DRV Control Register 0-24 (DRVCTRL0-24).

- POC control for each LSI pin

POC control registers must be set according to IO voltage level that is supplied to the pin.

POC control resistors on each LSI pin can be controlled by setting the registers in the PFC module. (Selection is handled by the IO voltage level select register (POCCTRL0). For details, see sections 8.2.5, POC Control Register 0 (POCCTRL0).

- TDSEL control for each LSI pin

TDSEL control registers can control the driving abilities of pins in use for the SDHI.

TDSEL control resistors on each LSI pin can be controlled by setting the registers in the PFC module. (Selection is handled by the return path for SDHI clock drive select register (TDSELCTRL0). For details, see sections 8.2.6, TDSEL Control Register 0 (TDSELCTRL0).

- Pull-up/down control for each LSI pin.

PUEN registers can on/off control of the pull resistors.

On/off of the pull resistors on each LSI pin can be controlled by setting the registers in the PFC module.

(Selection is handled by the Pull-on/off select register (PUEN0-6). For details, see sections 8.2.7, LSI pin pull-enable register 0-6 (PUEN0-6).

PUD registers can pull-up/pull-down control of the pull resistors.

Pull-up/Pull-down control resistors on each LSI pin can be controlled by setting the registers in the PFC module.

(Selection is handled by the Pull-up/down select register (PUD0-6). For details, see sections 8.2.8, LSI pin pull-up/down control Register 0-6 (PUD0-6).

- Module selection

Module Select Register can select the group for multiple LSI pins with multiplexed pin functions.

Enable and disable the functions of RZ/G 2nd Generation LSI pins to which pin functions from multiple pin groups are assigned by setting the registers in the PFC module.

(Selection is handled by the module select register (MOD_SEL0-2). For details, see sections 8.2.9, Module Select Register 0-2 (MOD_SEL0-2).

- Notes on configuring multiplexed pin functions
 The multiplexed LSI pins (MOD_SEL0-2, GPSR0-7, IPSR0-18) must be set in the initial sequence (*).
 Switching multiplexed LSI pins during operation is not guaranteed.
 *: The initial sequence is Appendix B.(3)
- Module clock
 Module clock is S3D4φ.

8.1.2 Block Diagram

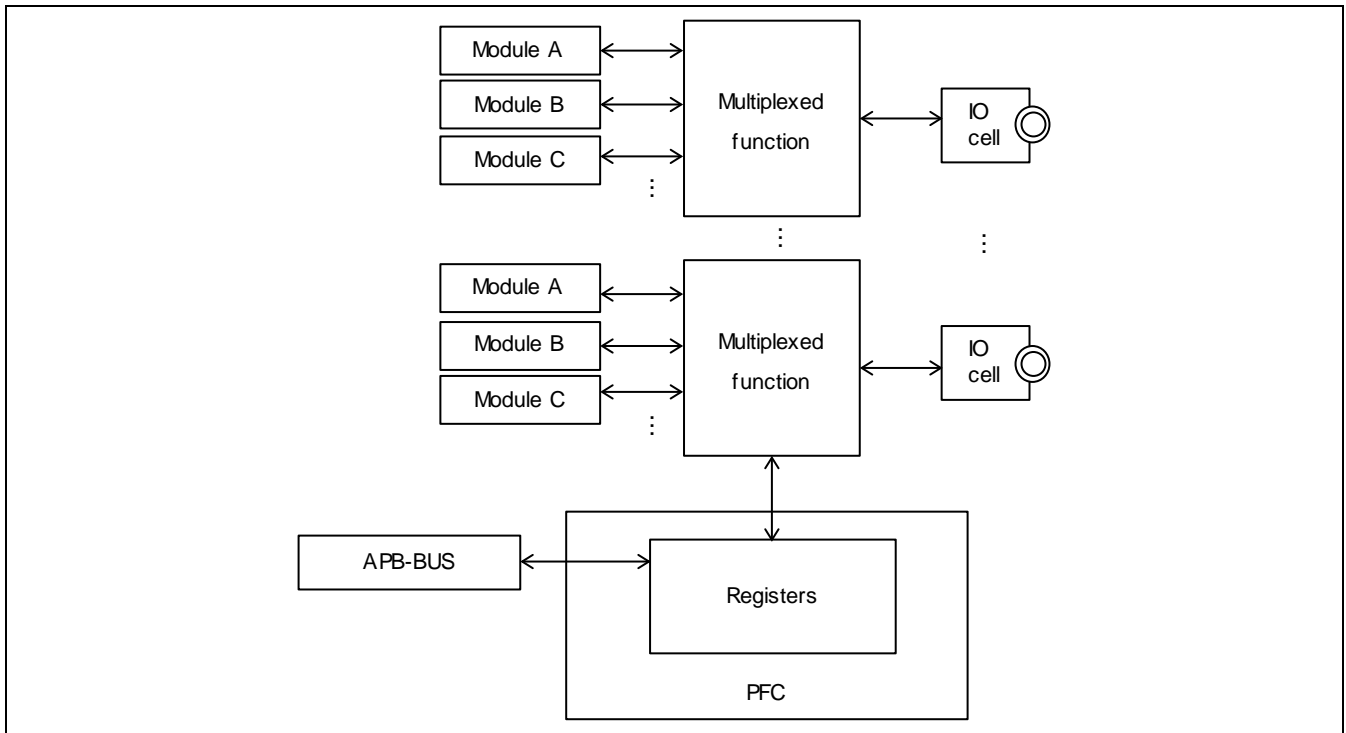


Figure 8.1 PFC Block Configuration

8.1.3 External Pins

PFC does not have External Pins.

8.1.4 Register Configuration

All the registers in the PFC are mapped into the APB bus space. Table 8.1 shows the configuration of the registers provided in the PFC. Details on each register in the PFC are given in sections 8.2.1 to 8.2.9.

Table 8.1 Configuration of Registers in PFC

| Name | Abbr. | R/W | Initial Value | | | Address | Access Size | Condition |
|--|-------|-----|---|---|---|-------------|-------------|---|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | | | |
| LSI Multiplexed Pin Setting Mask Register | PMMR | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0000 | 32 | — |
| GPIO/Peripheral Function Select register 0 | GPSR0 | R/W | H'0000_FFFF (when md[4:1] = B'0000), H'0000_0000 (when md[4:1] ≠ B'0000) | H'0000_FFFF (when md[4:1] = B'0000), H'0000_0000 (when md[4:1] ≠ B'0000) | H'0000_FFFF (when md[4:1] = B'0000), H'0000_0000 (when md[4:1] ≠ B'0000) | H'E606_0100 | 32 | — |
| GPIO/Peripheral Function Select register 1 | GPSR1 | R/W | H'1EFF_FFFF (when md[4:1] = B'0000), H'1000_0000 (when md[4:1] ≠ B'0000) | H'1EFF_FFFF (when md[4:1] = B'0000), H'1000_0000 (when md[4:1] ≠ B'0000) | H'1EFF_FFFF (when md[4:1] = '0000), H'1000_0000 (when md[4:1] ≠ B'0000) | H'E606_0104 | 32 | The values of bits 15 to 0 in the power-on reset state are retained for 50 μs after release from the power-on reset state. When bits 15 to 0 are to be used for the output of addresses, external handling of the pins is required because these bits are hi-z in the power-on reset state. |
| GPIO/Peripheral Function Select register 2 | GPSR2 | R/W | H'0000_03C0 (when md[4:1] = B'0000), H'0000_0200 (when md[4:1] ≠ B'0000) | H'0000_03C0 (when md[4:1] = B'0000), H'0000_0200 (when md[4:1] ≠ B'0000) | H'0000_03C0 (when md[4:1] = B'0000), H'0000_0200 (when md[4:1] ≠ B'0000) | H'E606_0108 | 32 | — |
| GPIO/Peripheral Function Select register 3 | GPSR3 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_010C | 32 | — |
| GPIO/Peripheral Function Select register 4 | GPSR4 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0110 | 32 | — |
| GPIO/Peripheral Function Select register 5 | GPSR5 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0114 | 32 | — |
| GPIO/Peripheral Function Select register 6 | GPSR6 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0118 | 32 | — |
| GPIO/Peripheral Function Select register 7 | GPSR7 | R/W | H'0000_0003 | H'0000_0003 | H'0000_0003 | H'E606_011C | 32 | — |

| Name | Abbr. | R/W | Initial Value | | | Address | Access Size | Condition |
|---|--------------|-----|---|---|---|-------------|----------------|-----------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | | | |
| Peripheral Function Select register 0 | IPSR0 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0200 | 32 | — |
| Peripheral Function Select register 1 | IPSR1 | R/W | H'0222_0000 (when md[4:1] = B'0000), H'0000_0000 (when md[4:1] ≠ B'0000) | H'0222_0000 (when md[4:1] = B'0000), H'0000_0000 (when md[4:1] ≠ B'0000) | H'0222_0000 (when md[4:1] = B'0000), H'0000_0000 (when md[4:1] ≠ B'0000) | H'E606_0204 | 32 | — |
| Peripheral Function Select register 2 | IPSR2 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0208 | 32 | — |
| Peripheral Function Select register 3 | IPSR3 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_020C | 32 | — |
| Peripheral Function Select register 4 | IPSR4 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0210 | 32 | — |
| Peripheral Function Select register 5 | IPSR5 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0214 | 32 | — |
| Peripheral Function Select register 6 | IPSR6 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0218 | 32 | — |
| Peripheral Function Select register 7 | IPSR7 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_021C | 32 | — |
| Peripheral Function Select register 8 | IPSR8 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0220 | 32 | — |
| Peripheral Function Select register 9 | IPSR9 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0224 | 32 | — |
| Peripheral Function Select register 10 | IPSR10 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0228 | 32 | — |
| Peripheral Function Select register 11 | IPSR11 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_022C | 32 | — |
| Peripheral Function Select register 12 | IPSR12 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0230 | 32 | — |
| Peripheral Function Select register 13 | IPSR13 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0234 | 32 | — |
| Peripheral Function Select register 14 | IPSR14 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0238 | 32 | — |
| Peripheral Function Select register 15 | IPSR15 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_023C | 32 | — |
| Peripheral Function Select register 16 | IPSR16 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0240 | 32 | — |
| Peripheral Function Select register 17 | IPSR17 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0244 | 32 | — |
| Peripheral Function Select register 18 | IPSR18 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0248 | 32 | — |
| DRV control register0 | DRVCTR L0 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0300 | 32 | — |
| DRV control register1 | DRVCTR L1 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0304 | 32 | — |
| DRV control register2 | DRVCTR L2 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0308 | 32 | — |
| DRV control register3 | DRVCTR L3 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_030C | 32 | — |

| Name | Abbr. | R/W | Initial Value | | | Address | Access Size | Condition |
|--------------------------------|--------------|-----|---------------|-------------|-------------|-------------|-------------|-----------|
| | | | RZ/G2H | RZ/G2M V1.3 | | | | |
| | | | | RZ/G2M V3.0 | RZ/G2N | | | |
| DRV control register4 | DRVCTR L4 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0310 | 32 | — |
| DRV control register5 | DRVCTR L5 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0314 | 32 | — |
| DRV control register6 | DRVCTR L6 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0318 | 32 | — |
| DRV control register7 | DRVCTR L7 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_031C | 32 | — |
| DRV control register8 | DRVCTR L8 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0320 | 32 | — |
| DRV control register9 | DRVCTR L9 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0324 | 32 | — |
| DRV control register10 | DRVCTR L10 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0328 | 32 | — |
| DRV control register11 | DRVCTR L11 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_032C | 32 | — |
| DRV control register12 | DRVCTR L12 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0330 | 32 | — |
| DRV control register13 | DRVCTR L13 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0334 | 32 | — |
| DRV control register14 | DRVCTR L14 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0338 | 32 | — |
| DRV control register15 | DRVCTR L15 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_033C | 32 | — |
| DRV control register16 | DRVCTR L16 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0340 | 32 | — |
| DRV control register17 | DRVCTR L17 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0344 | 32 | — |
| DRV control register18 | DRVCTR L18 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0348 | 32 | — |
| DRV control register19 | DRVCTR L19 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_034C | 32 | — |
| DRV control register20 | DRVCTR L20 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0350 | 32 | — |
| DRV control register21 | DRVCTR L21 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0354 | 32 | — |
| DRV control register22 | DRVCTR L22 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0358 | 32 | — |
| DRV control register23 | DRVCTR L23 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_035C | 32 | — |
| DRV control register24 | DRVCTR L24 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0360 | 32 | — |
| POC control register0 | POCCTR L0 | R/W | H'3FFF_FFFF | H'3FFF_FFFF | H'3FFF_FFFF | H'E606_0380 | 32 | — |
| TDSEL control register0 | TDSELECT RL0 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_03C0 | 32 | — |
| LSI pin pull-enable register 0 | PUEN0 | R/W | H'CFFF_9000 | H'CFFF_9000 | H'CFFF_9000 | H'E606_0400 | 32 | — |

| Name | Abbr. | R/W | Initial Value | | | Address | Access Size | Condition |
|---|--------------|-----|---------------|-------------|-------------|-------------|----------------|-----------|
| | | | RZ/G2H | RZ/G2M V1.3 | | | | |
| | | | | RZ/G2M V3.0 | RZ/G2N | | | |
| LSI pin pull-enable register 1 | PUEN1 | R/W | H'00B2_4FFF | H'00B2_4FFF | H'00B2_4FFF | H'E606_0404 | 32 | — |
| LSI pin pull-enable register 2 | PUEN2 | R/W | H'FF00_05E6 | H'FF00_05E6 | H'FF00_05E6 | H'E606_0408 | 32 | — |
| LSI pin pull-enable register 3 | PUEN3 | R/W | H'0000_02F7 | H'0000_02F5 | H'0000_02F5 | H'E606_040C | 32 | — |
| LSI pin pull-enable register 4 | PUEN4 | R/W | H'FFFF_FF00 | H'FFFF_FF00 | H'FFFF_FF00 | H'E606_0410 | 32 | — |
| LSI pin pull-enable register 5 | PUEN5 | R/W | H'FF7F_FF87 | H'FF7F_FF87 | H'FF7F_FF87 | H'E606_0414 | 32 | — |
| LSI pin pull-enable register 6 | PUEN6 | R/W | H'0000_007F | H'0000_00FF | H'0000_007F | H'E606_0418 | 32 | — |
| LSI pin pull-up/down control register 0 | PUD0 | R/W | H'3000_7FFF | H'3000_7FFF | H'3000_7FFF | H'E606_0440 | 32 | — |
| LSI pin pull-up/down control register 1 | PUD1 | R/W | H'FF7F_BFFE | H'FF7F_BFFE | H'FF7F_BFFE | H'E606_0444 | 32 | — |
| LSI pin pull-up/down control register 2 | PUD2 | R/W | H'33FF_FBFF | H'33FF_FBFF | H'33FF_FBFF | H'E606_0448 | 32 | — |
| LSI pin pull-up/down control register 3 | PUD3 | R/W | H'FFFF_FEEC | H'FFFF_FEEC | H'FFFF_FEEC | H'E606_044C | 32 | — |
| LSI pin pull-up/down control register 4 | PUD4 | R/W | H'FFFF_FFFF | H'FFFF_FFFF | H'FFFF_FFFF | H'E606_0450 | 32 | — |
| LSI pin pull-up/down control register 5 | PUD5 | R/W | H'7FDF_FFFF | H'7F5F_FFFF | H'7FDF_FFFF | H'E606_0454 | 32 | — |
| LSI pin pull-up/down control register 6 | PUD6 | R/W | H'0000_0055 | H'0000_0055 | H'0000_0055 | H'E606_0458 | 32 | — |
| Module select register 0 | MOD_SE L0 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0500 | 32 | — |
| Module select register 1 | MOD_SE L1 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0504 | 32 | — |
| Module select register 2 | MOD_SE L2 | R/W | H'0000_0000 | H'0000_0000 | H'0000_0000 | H'E606_0508 | 32 | — |

8.1.5 Connected Module

Table 8.2 Connected module

| Module name | Connected module name | Function of connected module |
|-------------|-----------------------|------------------------------|
| PFC | AP-System Core | Access the Registers |
| | CPG | Output clocks |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |

8.2 Register Description

[Legend]

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R: Read-only. The write value should always be 0.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

W: Write-only. Reading this bit is prohibited. When the bit is reserved, the write value should always be 0.

—/W: Write-only. The read value is undefined.

All the bits are active high unless otherwise specified, and deactivated on reset.

All access to registers is made in longword units.

The write value to a reserved bit should always be 0.

8.2.1 LSI Multiplexed Pin Setting Mask Register (PMMR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: PMMR enables/disables writing to the multiplexed pin setting registers.

| | | | | | | | | | | | | | | | | |
|----------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MPM [31] | MPM [30] | MPM [29] | MPM [28] | MPM [27] | MPM [26] | MPM [25] | MPM [24] | MPM [23] | MPM [22] | MPM [21] | MPM [20] | MPM [19] | MPM [18] | MPM [17] | MPM [16] |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MPM [15] | MPM [14] | MPM [13] | MPM [12] | MPM [11] | MPM [10] | MPM [9] | MPM [8] | MPM [7] | MPM [6] | MPM [5] | MPM [4] | MPM [3] | MPM [2] | MPM [1] | MPM [0] |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | MPM[31:0] | H'0000_0000 | R/W | <p>Multiplexed Pin Setting Mask</p> <p>Writing a value to any register from among the GPIO/peripheral function select registers GPSR0 to GPSR7, peripheral function select registers IPSR0 to IPSR18, DRV control registers DRVCTRL0-24, TDSEL control registers TDSELCTRL0, POC control register POCCTRL0 and Module select register MOD_SEL0-2 is enabled by writing the inverse of the value to this register.</p> |

Note: This register must be set before setting each of the GPIO/peripheral function select registers GPSR0 to GPSR7, peripheral function select registers IPSR0 to IPSR18, DRV control registers DRVCTRL0-24, TDSEL control registers TDSELCTRL0, POC control register POCCTRL0 and Module select register MOD_SEL0-2.

8.2.2 GPIO/Peripheral Function Select Register 0-7 (GPSR0-7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: GPSR0-7 selects the functions of the multiplexed LSI pins.

| | | | | | | | | | | | | | | | | |
|------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | GPn* [31] | GPn* [30] | GPn* [29] | GPn* [28] | GPn* [27] | GPn* [26] | GPn* [25] | GPn* [24] | GPn* [23] | GPn* [22] | GPn* [21] | GPn* [20] | GPn* [19] | GPn* [18] | GPn* [17] | GPn* [16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | GPn* [15] | GPn* [14] | GPn* [13] | GPn* [12] | GPn* [11] | GPn* [10] | GPn* [9] | GPn* [8] | GPn* [7] | GPn* [6] | GPn* [5] | GPn* [4] | GPn* [3] | GPn* [2] | GPn* [1] | GPn* [0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * n = 0-7

| Bit | Bit Name | R/W | Description |
|---------|-------------|-----|-----------------------------------|
| 31 to 0 | GP0-7[31:0] | R/W | 0: GPIO 1: Peripheral function |

The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the multiplexed pin setting mask register (PMMR) immediately before setting this register.

Table 8.3 Configuration of Registers in GPSR0-3

| | Initial value (md[4:1] =0000) | Initial value (md[4:1] ≠0000) | GPSR1 | Initial value (md[4:1] =0000) | Initial value (md[4:1] ≠0000) | GPSR2 | Initial value (md[4:1] =0000) | Initial value (md[4:1] ≠0000) | GPSR3 | Initial value (md[4:1] =0000) | Initial value (md[4:1] ≠0000) |
|-------|-------------------------------|-------------------------------|------------|-------------------------------|-------------------------------|--------------------|-------------------------------|-------------------------------|--------|-------------------------------|-------------------------------|
| bit31 | — | — | — | — | — | — | — | — | — | — | — |
| bit30 | — | — | — | — | — | — | — | — | — | — | — |
| bit29 | — | — | — | — | — | — | — | — | — | — | — |
| bit28 | — | — | CLKOUT | 1 | 1 | — | — | — | — | — | — |
| bit27 | — | — | EX_WAIT0_A | 1 | 0 | — | — | — | — | — | — |
| bit26 | — | — | WE1# | 1 | 0 | — | — | — | — | — | — |
| bit25 | — | — | WE0# | 1 | 0 | — | — | — | — | — | — |
| bit24 | — | — | RD/WR# | 0 | 0 | — | — | — | — | — | — |
| bit23 | — | — | RD# | 1 | 0 | — | — | — | — | — | — |
| bit22 | — | — | BS# | 1 | 0 | — | — | — | — | — | — |
| bit21 | — | — | CS1# | 1 | 0 | — | — | — | — | — | — |
| bit20 | — | — | CS0# | 1 | 0 | — | — | — | — | — | — |
| bit19 | — | — | A19 | 1 | 0 | — | — | — | — | — | — |
| bit18 | — | — | A18 | 1 | 0 | — | — | — | — | — | — |
| bit17 | — | — | A17 | 1 | 0 | — | — | — | — | — | — |
| bit16 | — | — | A16 | 1 | 0 | — | — | — | — | — | — |
| bit15 | D15 | 1 | A15 | 1 | 0 | — | — | SD1_WP | 0 | 0 | 0 |
| bit14 | D14 | 1 | A14 | 1 | 0 | AVB_AVTP_CAPTURE_A | 0 | 0 | SD1_CD | 0 | 0 |

| GPSR0 | | Initial value (md[4:1] =0000) | Initial value (md[4:1] ≠0000) | GPSR1 | Initial value (md[4:1] =0000) | Initial value (md[4:1] ≠0000) | GPSR2 | Initial value (md[4:1] =0000) | Initial value (md[4:1] ≠0000) | GPSR3 | Initial value (md[4:1] =0000) | Initial value (md[4:1] ≠0000) |
|-------|-----|-------------------------------|-------------------------------|-------|-------------------------------|-------------------------------|------------------|-------------------------------|-------------------------------|----------|-------------------------------|-------------------------------|
| bit13 | D13 | 1 | 0 | A13 | 1 | 0 | AVB_AVTP_MATCH_A | 0 | 0 | SD0_WP | 0 | 0 |
| bit12 | D12 | 1 | 0 | A12 | 1 | 0 | AVB_LINK | 0 | 0 | SD0_CD | 0 | 0 |
| bit11 | D11 | 1 | 0 | A11 | 1 | 0 | AVB_PHY_INT | 0 | 0 | SD1_DAT3 | 0 | 0 |
| bit10 | D10 | 1 | 0 | A10 | 1 | 0 | AVB_MAGIC | 0 | 0 | SD1_DAT2 | 0 | 0 |
| bit9 | D9 | 1 | 0 | A9 | 1 | 0 | AVB_MDC | 1 | 1 | SD1_DAT1 | 0 | 0 |
| bit8 | D8 | 1 | 0 | A8 | 1 | 0 | PWM2_A | 1 | 0 | SD1_DAT0 | 0 | 0 |
| bit7 | D7 | 1 | 0 | A7 | 1 | 0 | PWM1_A | 1 | 0 | SD1_CMD | 0 | 0 |
| bit6 | D6 | 1 | 0 | A6 | 1 | 0 | PWM0 | 1 | 0 | SD1_CLK | 0 | 0 |
| bit5 | D5 | 1 | 0 | A5 | 1 | 0 | IRQ5 | 0 | 0 | SD0_DAT3 | 0 | 0 |
| bit4 | D4 | 1 | 0 | A4 | 1 | 0 | IRQ4 | 0 | 0 | SD0_DAT2 | 0 | 0 |
| bit3 | D3 | 1 | 0 | A3 | 1 | 0 | IRQ3 | 0 | 0 | SD0_DAT1 | 0 | 0 |
| bit2 | D2 | 1 | 0 | A2 | 1 | 0 | IRQ2 | 0 | 0 | SD0_DAT0 | 0 | 0 |
| bit1 | D1 | 1 | 0 | A1 | 1 | 0 | IRQ1 | 0 | 0 | SD0_CMD | 0 | 0 |
| bit0 | D0 | 1 | 0 | A0 | 1 | 0 | IRQ0 | 0 | 0 | SD0_CLK | 0 | 0 |

Table 8.4 Configuration of Registers in GPSR4-7

| bit | GPSR4 | Initial value | Initial value | GPSR5 | Initial value | Initial value | GPSR6 | Initial value | Initial value | GPSR7 | Initial value | Initial value |
|-------|----------|----------------|----------------|-----------------|----------------|----------------|---|----------------|----------------|-------|----------------|----------------|
| | | (md[4:1]=0000) | (md[4:1]≠0000) | | (md[4:1]=0000) | (md[4:1]≠0000) | | (md[4:1]=0000) | (md[4:1]≠0000) | | (md[4:1]=0000) | (md[4:1]≠0000) |
| bit31 | — | — | — | — | — | — | [RZ/G2H] [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] GP6_31* | 0 | 0 | — | — | — |
| bit30 | — | — | — | — | — | — | [RZ/G2H] [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] GP6_30* | 0 | 0 | — | — | — |
| bit29 | — | — | — | — | — | — | USB30_ OVC | 0 | 0 | — | — | — |
| bit28 | — | — | — | — | — | — | USB30_ PWEN | 0 | 0 | — | — | — |
| bit27 | — | — | — | — | — | — | USB1_OVC | 0 | 0 | — | — | — |
| bit26 | — | — | — | — | — | — | USB1_ PWEN | 0 | 0 | — | — | — |
| bit25 | — | — | — | MLB_DAT | 0 | 0 | USB0_OVC | 0 | 0 | — | — | — |
| bit24 | — | — | — | MLB_SIG | 0 | 0 | USB0_ PWEN | 0 | 0 | — | — | — |
| bit23 | — | — | — | MLB_CLK | 0 | 0 | AUDIO_ CLKB_B | 0 | 0 | — | — | — |
| bit22 | — | — | — | MSIOF0_ RXD | 0 | 0 | AUDIO_ CLKA_A | 0 | 0 | — | — | — |
| bit21 | — | — | — | MSIOF0_ SS2 | 0 | 0 | SSI_ SDATA9_A | 0 | 0 | — | — | — |
| bit20 | — | — | — | MSIOF0_ TXD | 0 | 0 | SSI_ SDATA8 | 0 | 0 | — | — | — |
| bit19 | — | — | — | MSIOF0_ SS1 | 0 | 0 | SSI_ SDATA7 | 0 | 0 | — | — | — |
| bit18 | — | — | — | MSIOF0_ SYNC | 0 | 0 | SSI_WS78 | 0 | 0 | — | — | — |
| bit17 | SD3_DS | 0 | 0 | MSIOF0_ SCK | 0 | 0 | SSI_SCK78 | 0 | 0 | — | — | — |
| bit16 | SD3_DAT7 | 0 | 0 | HRTS0# | 0 | 0 | SSI_ SDATA6 | 0 | 0 | — | — | — |
| bit15 | SD3_DAT6 | 0 | 0 | HCTS0# | 0 | 0 | SSI_WS6 | 0 | 0 | — | — | — |
| bit14 | SD3_DAT5 | 0 | 0 | HTX0 | 0 | 0 | SSI_SCK6 | 0 | 0 | — | — | — |
| bit13 | SD3_DAT4 | 0 | 0 | HRX0 | 0 | 0 | SSI_SDAT A5 | 0 | 0 | — | — | — |
| bit12 | SD3_DAT3 | 0 | 0 | HACK0 | 0 | 0 | SSI_WS5 | 0 | 0 | — | — | — |
| bit11 | SD3_DAT2 | 0 | 0 | RX2_A | 0 | 0 | SSI_SCK5 | 0 | 0 | — | — | — |
| bit10 | SD3_DAT1 | 0 | 0 | TX2_A | 0 | 0 | SSI_SDAT A4 | 0 | 0 | — | — | — |

| GPSR4 | | Initial value (md[4:1] =0000) | Initial value (md[4:1] ≠0000) | GPSR5 | Initial value (md[4:1] =0000) | Initial value (md[4:1] ≠0000) | GPSR6 | Initial value (md[4:1] =0000) | Initial value (md[4:1] ≠0000) | GPSR7 | Initial value (md[4:1] =0000) | Initial value (md[4:1] ≠0000) |
|-------------|----------|-------------------------------|-------------------------------|-------|-------------------------------|-------------------------------|---------------|-------------------------------|-------------------------------|--------|-------------------------------|-------------------------------|
| bit9 | SD3_DAT0 | 0 | 0 | SCK2 | 0 | 0 | SSL_WS4 | 0 | 0 | — | — | — |
| bit8 | SD3_CMD | 0 | 0 | RTS1# | 0 | 0 | SSL_SCK4 | 0 | 0 | — | — | — |
| bit7 | SD3_CLK | 0 | 0 | CTS1# | 0 | 0 | SSL_ SDATA3 | 0 | 0 | — | — | — |
| bit6 | SD2_DS | 0 | 0 | TX1_A | 0 | 0 | SSL_WS349 | 0 | 0 | — | — | — |
| bit5 | SD2_DAT3 | 0 | 0 | RX1_A | 0 | 0 | SSL_ SCK349 | 0 | 0 | — | — | — |
| bit4 | SD2_DAT2 | 0 | 0 | RTS0# | 0 | 0 | SSL_ SDATA2_A | 0 | 0 | — | — | — |
| bit3 | SD2_DAT1 | 0 | 0 | CTS0# | 0 | 0 | SSL_ SDATA1_A | 0 | 0 | GP7_03 | 0 | 0 |
| bit2 | SD2_DAT0 | 0 | 0 | TX0 | 0 | 0 | SSL_ SDATA0 | 0 | 0 | GP7_02 | 0 | 0 |
| bit1 | SD2_CMD | 0 | 0 | RX0 | 0 | 0 | SSL_ WS01239 | 0 | 0 | AVS2 | 1 | 1 |
| bit0 | SD2_CLK | 0 | 0 | SCK0 | 0 | 0 | SSL_ SCK01239 | 0 | 0 | AVS1 | 1 | 1 |

Note: * Different for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N

8.2.3 Peripheral Function Select Register 0-18 (IPSR0-18)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: IPSR0-18 selects the functions of the multiplexed LSI pins.

| | | | | | | | | | | | | | | | | |
|------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | IPn* [31] | IPn* [30] | IPn* [29] | IPn* [28] | IPn* [27] | IPn* [26] | IPn* [25] | IPn* [24] | IPn* [23] | IPn* [22] | IPn* [21] | IPn* [20] | IPn* [19] | IPn* [18] | IPn* [17] | IPn* [16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | IPn* [15] | IPn* [14] | IPn* [13] | IPn* [12] | IPn* [11] | IPn* [10] | IPn* [9] | IPn* [8] | IPn* [7] | IPn* [6] | IPn* [5] | IPn* [4] | IPn* [3] | IPn* [2] | IPn* [1] | IPn* [0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * n = 0-18

| Bit | R/W | Description |
|---------|-----|--|
| 31 to 0 | R/W | The functions of the LSI pins are selected according to the table below. For the initial values of these registers, refer to Table 8.1, and change the initial values if necessary. |

Note: To enable this register to be set, appropriately set the multiplexed pin setting mask register (PMMR) immediately before setting this register.

IPSR register must not set "—", "Reserved" and IPSR = H'E, F.

Table 8.5 Configuration of Registers in IPSR

| Register | IPSR = H'0 | IPSR = H'1 | IPSR = H'2 | IPSR = H'3 | IPSR = H'4 | IPSR = H'5 | IPSR = H'6 | IPSR = H'7 | IPSR = H'8 | IPSR = H'9 | IPSR = H'A | IPSR = H'B | IPSR = H'C | IPSR = H'D |
|------------|--------------------------------|----------------------|-------------------|---|---------------------|---------------|-----------------|-------------------|---------------|---------------|---------------|---------------|---------------|---------------|
| IP0[3:0] | AVB_ MDC | — | MSIOF2_ SS2_C | — | — | — | — | — | — | — | — | — | — | — |
| IP0[7:4] | AVB_ MAGIC | — | MSIOF2_ SS1_C | SCK4_A | — | — | — | — | — | — | — | — | — | — |
| IP0[11:8] | AVB_ PHY_ INT | — | MSIOF2_ SYNC_C | RX4_A | — | — | — | — | — | — | — | — | — | — |
| IP0[15:12] | AVB_ LINK | — | MSIOF2_ SCK_C | TX4_A | — | — | — | — | — | — | — | — | — | — |
| IP0[19:16] | AVB_ AVTP_M ATCH_A | — | MSIOF2_ RXD_C | CTS4#_A | — | — | — | — | — | — | — | — | — | — |
| IP0[23:20] | AVB_ AVTP_ CAPTUR E_A | — | MSIOF2_ TXD_C | RTS4#A | — | — | — | — | — | — | — | — | — | — |
| IP0[27:24] | IRQ0 | — | — | DU_CDE | VI4_ DATA0_ B | CAN0_ TX_B | CANFD0 _TX_B | MSIOF3_ SS2_E | — | — | — | — | — | — |
| IP0[31:28] | IRQ1 | — | — | DU_DISP | VI4_ DATA1_ B | CAN0_R X_B | CANFD0 _RX_B | MSIOF3_ SS1_E | — | — | — | — | — | — |
| IP1[3:0] | IRQ2 | — | — | DU_EXO DDF/ DU_ ODDF/ DISP / CDE | VI4_ DATA2_ B | — | — | MSIOF3_ SYNC_E | — | PWM3_B | — | — | — | — |
| IP1[7:4] | IRQ3 | — | — | DU_DOT CLKOUT 1 | VI4_ DATA3_ B | — | — | MSIOF3_ SCK_E | — | PWM4_B | — | — | — | — |
| IP1[11:8] | IRQ4 | — | — | DU_EXH SYNC/ DU_ HSYNC | VI4_ DATA4_ B | — | — | MSIOF3_ RXD_E | — | PWM5_B | — | — | — | — |
| IP1[15:12] | IRQ5 | — | — | DU_EXV SYNC/ DU_ VSYNC | VI4_DAT A5_B | — | — | MSIOF3_ TXD_E | — | PWM6_B | — | — | — | — |
| IP1[19:16] | PWM0 | AVB_ AVTP_ PPS | — | — | VI4_ DATA6_ B | — | — | — | — | — | — | — | — | — |
| IP1[23:20] | PWM1_A | — | — | HRX3_D | VI4_ DATA7_ B | — | — | — | — | — | — | — | — | — |
| IP1[27:24] | PWM2_A | — | — | HTX3_D | — | — | — | — | — | — | — | — | — | — |
| IP1[31:28] | A0 | — | MSIOF3_ SYNC_B | — | VI4_ DATA8 | — | DU_DB0 | — | — | PWM3_A | — | — | — | — |
| IP2[3:0] | A1 | — | MSIOF3_ TXD_B | — | VI4_ DATA9 | — | DU_DB1 | — | — | PWM4_A | — | — | — | — |
| IP2[7:4] | A2 | — | MSIOF3_ SCK_B | — | VI4_ DATA10 | — | DU_DB2 | — | — | PWM5_A | — | — | — | — |

| Register | IPSR = H'0 | IPSR = H'1 | IPSR = H'2 | IPSR = H'3 | IPSR = H'4 | IPSR = H'5 | IPSR = H'6 | IPSR = H'7 | IPSR = H'8 | IPSR = H'9 | IPSR = H'A | IPSR = H'B | IPSR = H'C | IPSR = H'D |
|------------|---------------|---------------|-------------------|---------------|----------------|----------------|---------------|----------------|--------------------------------|-----------------|---------------|---------------|---------------|---------------|
| IP2[11:8] | A3 | — | MSIOF3_ RXD_B | — | VI4_ DATA11 | — | DU_DB3 | — | — | PWM6_A | — | — | — | — |
| IP2[15:12] | A4 | — | MSIOF3_ SS1_B | — | VI4_ DATA12 | VI5_ DATA12 | DU_DB4 | — | — | — | — | — | — | — |
| IP2[19:16] | A5 | — | MSIOF3_ SS2_B | SCK4_B | VI4_ DATA13 | VI5_ DATA13 | DU_DB5 | — | — | — | — | — | — | — |
| IP2[23:20] | A6 | — | MSIOF2_ SS1_A | RX4_B | VI4_ DATA14 | VI5_ DATA14 | DU_DB6 | — | — | — | — | — | — | — |
| IP2[27:24] | A7 | — | MSIOF2_ SS2_A | TX4_B | VI4_ DATA15 | VI5_ DATA15 | DU_DB7 | — | — | — | — | — | — | — |
| IP2[31:28] | A8 | RX3_B | MSIOF2_ SYNC_A | HRX4_B | — | — | — | SDA6_A | AVB_ AVTP_ MATCH_ B | PWM1_B | — | — | — | — |
| IP3[3:0] | A9 | — | MSIOF2_ SCK_A | CTS4#_B | — | VI5_ VSYNC# | — | — | — | — | — | — | — | — |
| IP3[7:4] | A10 | — | MSIOF2_ RXD_A | RTS4#_B | — | VI5_ HSYNC# | — | — | — | — | — | — | — | — |
| IP3[11:8] | A11 | TX3_B | MSIOF2_ TXD_A | HTX4_B | HCK4 | VI5_ FIELD | — | SCL6_A | AVB_ AVTP_ CAPTUR E_B | PWM2_B | — | — | — | — |
| IP3[15:12] | A12 | — | MSIOF3_ SCK_C | — | HRX4_A | VI5_ DATA8 | DU_DG4 | — | — | — | — | — | — | — |
| IP3[19:16] | A13 | — | MSIOF3_ SYNC_C | — | HTX4_A | VI5_ DATA9 | DU_DG5 | — | — | — | — | — | — | — |
| IP3[23:20] | A14 | — | MSIOF3_ RXD_C | — | HCTS4# | VI5_ DATA10 | DU_DG6 | — | — | — | — | — | — | — |
| IP3[27:24] | A15 | — | MSIOF3_ TXD_C | — | HRTS4# | VI5_ DATA11 | DU_DG7 | — | — | — | — | — | — | — |
| IP3[31:28] | A16 | — | — | — | VI4_ FIELD | — | DU_DG0 | — | — | — | — | — | — | — |
| IP4[3:0] | A17 | — | — | — | VI4_ VSYNC# | — | DU_DG1 | — | — | — | — | — | — | — |
| IP4[7:4] | A18 | — | — | — | VI4_ HSYNC# | — | DU_DG2 | — | — | — | — | — | — | — |
| IP4[11:8] | A19 | — | — | — | VI4_ CLKENB | — | DU_DG3 | — | — | — | — | — | — | — |
| IP4[15:12] | CS0# | — | — | — | — | VI5_ CLKENB | — | — | — | — | — | — | — | — |
| IP4[19:16] | CS1# | — | — | — | — | VI5_CLK | — | EX_ WAIT0_B | — | — | — | — | — | — |
| IP4[23:20] | BS# | — | MSIOF3_ SCK_D | SCK3 | HCK3 | — | — | — | CAN1_ TX | CANFD1 _TX | — | — | — | — |
| IP4[27:24] | RD# | — | MSIOF3_ SYNC_D | RX3_A | HRX3_A | — | — | — | CAN0_ TX_A | CANFD0 _TX_A | — | — | — | — |
| IP4[31:28] | RD/WR# | — | MSIOF3_ RXD_D | TX3_A | HTX3_A | — | — | — | CAN0_ RX_A | CANFD0 _RX_A | — | — | — | — |
| IP5[3:0] | WE0# | — | MSIOF3_ TXD_D | CTS3# | HCTS3# | — | — | SCL6_B | CAN_ CLK | — | — | — | — | — |

| Register | IPSR = H'0 | IPSR = H'1 | IPSR = H'2 | IPSR = H'3 | IPSR = H'4 | IPSR = H'5 | IPSR = H'6 | IPSR = H'7 | IPSR = H'8 | IPSR = H'9 | IPSR = H'A | IPSR = H'B | IPSR = H'C | IPSR = H'D |
|------------------|----------------|-------------------|-------------------|---------------|---------------------|---------------|-----------------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| IP5[7:4] | WE1# | — | MSIOF3_ SS1_D | RTS3# | HRTS3# | — | — | SDA6_B | CAN1_ RX | CANFD1 _RX | — | — | — | — |
| IP5[11:8] | EX_ WAIT0_A | — | — | — | VI4_CLK | — | DU_ DOTCLK OUT0 | — | — | — | — | — | — | — |
| IP5[15:12] | D0 | MSIOF2_ SS1_B | MSIOF3_ SCK_A | — | VI4_ DATA16 | VI5_ DATA0 | — | — | — | — | — | — | — | — |
| IP5[19:16] | D1 | MSIOF2_ SS2_B | MSIOF3_ SYNC_A | — | VI4_ DATA17 | VI5_ DATA1 | — | — | — | — | — | — | — | — |
| IP5[23:20] | D2 | — | MSIOF3_ RXD_A | — | VI4_ DATA18 | VI5_ DATA2 | — | — | — | — | — | — | — | — |
| IP5[27:24] | D3 | — | MSIOF3_ TXD_A | — | VI4_ DATA19 | VI5_ DATA3 | — | — | — | — | — | — | — | — |
| IP5[31:28] | D4 | MSIOF2_ SCK_B | — | — | VI4_ DATA20 | VI5_ DATA4 | — | — | — | — | — | — | — | — |
| IP6[3:0] | D5 | MSIOF2_ SYNC_B | — | — | VI4_ DATA21 | VI5_ DATA5 | — | — | — | — | — | — | — | — |
| IP6[7:4] | D6 | MSIOF2_ RXD_B | — | — | VI4_ DATA22 | VI5_ DATA6 | — | — | — | — | — | — | — | — |
| IP6[11:8] | D7 | MSIOF2_ TXD_B | — | — | VI4_ DATA23 | VI5_ DATA7 | — | — | — | — | — | — | — | — |
| IP6[15:12] | D8 | — | MSIOF2_ SCK_D | SCK4_C | VI4_ DATA0_ A | — | DU_DR0 | — | — | — | — | — | — | — |
| IP6[19:16] | D9 | — | MSIOF2_ SYNC_D | — | VI4_ DATA1_ A | — | DU_DR1 | — | — | — | — | — | — | — |
| IP6[23:20] | D10 | — | MSIOF2_ RXD_D | HRX3_B | VI4_ DATA2_ A | CTS4#_C | DU_DR2 | — | — | — | — | — | — | — |
| IP6[27:24] | D11 | — | MSIOF2_ TXD_D | HTX3_B | VI4_ DATA3_ A | RTS4#_C | DU_DR3 | — | — | — | — | — | — | — |
| IP6[31:28] | D12 | — | MSIOF2_ SS1_D | RX4_C | VI4_ DATA4_ A | — | DU_DR4 | — | — | — | — | — | — | — |
| IP7[3:0] | D13 | — | MSIOF2_ SS2_D | TX4_C | VI4_ DATA5_ A | — | DU_DR5 | — | — | — | — | — | — | — |
| IP7[7:4] | D14 | — | MSIOF3_ SS1_A | HRX3_C | VI4_ DATA6_ A | — | DU_DR6 | SCL6_C | — | — | — | — | — | — |
| IP7[11:8] | D15 | — | MSIOF3_ SS2_A | HTX3_C | VI4_ DATA7_ A | — | DU_DR7 | SDA6_C | — | — | — | — | — | — |
| IP7[15:12] *2 | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| IP7[19:16] | SD0_ CLK | — | MSIOF1_ SCK_E | — | — | — | — | — | — | — | — | — | — | — |
| IP7[23:20] | SD0_ CMD | — | MSIOF1_ SYNC_E | — | — | — | — | — | — | — | — | — | — | — |

| Register | IPSR = H'0 | IPSR = H'1 | IPSR = H'2 | IPSR = H'3 | IPSR = H'4 | IPSR = H'5 | IPSR = H'6 | IPSR = H'7 | IPSR = H'8 | IPSR = H'9 | IPSR = H'A | IPSR = H'B | IPSR = H'C | IPSR = H'D |
|----------------|---------------|---------------|-------------------|----------------|---------------|---------------|---------------|---------------|--|---------------|---------------|---------------|---------------|---------------|
| IP7[27:24] | SD0_ DAT0 | — | MSIOF1_ RXD_E | — | — | — | — | — | — | — | — | — | — | — |
| IP7[31:28] | SD0_ DAT1 | — | MSIOF1_ TXD_E | — | — | — | — | — | — | — | — | — | — | — |
| IP8[3:0] | SD0_ DAT2 | — | MSIOF1_ SS1_E | — | — | — | — | — | — | — | — | — | — | — |
| IP8[7:4] | SD0_ DAT3 | — | MSIOF1_ SS2_E | — | — | — | — | — | — | — | — | — | — | — |
| IP8[11:8] | SD1_ CLK | — | MSIOF1_ SCK_G | — | — | — | — | — | — | — | — | — | — | — |
| IP8 [15:12] | SD1_ CMD | — | MSIOF1_ SYNC_G | NFCE#_ B | — | — | — | — | — | — | — | — | — | — |
| IP8[19:16] | SD1_DA T0 | SD2_DA T4 | MSIOF1_ RXD_G | NFWP#_ B | — | — | — | — | — | — | — | — | — | — |
| IP8[23:20] | SD1_ DAT1 | SD2_ DAT5 | MSIOF1_ TXD_G | NFDATA 14_B | — | — | — | — | — | — | — | — | — | — |
| IP8[27:24] | SD1_ DAT2 | SD2_DA T6 | MSIOF1_ SS1_G | NFDATA 15_B | — | — | — | — | — | — | — | — | — | — |
| IP8[31:28] | SD1_ DAT3 | SD2_DA T7 | MSIOF1_ SS2_G | NFRB#_ B | — | — | — | — | — | — | — | — | — | — |
| IP9[3:0] | SD2_ CLK | — | NFDATA 8 | — | — | — | — | — | — | — | — | — | — | — |
| IP9[7:4] | SD2_ CMD | — | NFDATA 9 | — | — | — | — | — | — | — | — | — | — | — |
| IP9[11:8] | SD2_ DAT0 | — | NFDATA 10 | — | — | — | — | — | — | — | — | — | — | — |
| IP9[15:12] | SD2_ DAT1 | — | NFDATA 11 | — | — | — | — | — | — | — | — | — | — | — |
| IP9[19:16] | SD2_ DAT2 | — | NFDATA 12 | — | — | — | — | — | — | — | — | — | — | — |
| IP9[23:20] | SD2_DA T3 | — | NFDATA 13 | — | — | — | — | — | — | — | — | — | — | — |
| IP9[27:24] | SD2_DS | — | NFALE | — | — | — | — | — | [RZ/G2H] [RZ/G2N] SATA_ DEVSLP _B *1 [RZ/G2M V1.3] [RZ/G2M V3.0] — *1 | — | — | — | — | — |
| IP9[31:28] | SD3_ CLK | — | NFWE# | — | — | — | — | — | — | — | — | — | — | — |
| IP10[3:0] | SD3_ CMD | — | NFRE# | — | — | — | — | — | — | — | — | — | — | — |
| IP10[7:4] | SD3_ DAT0 | — | NFDATA 0 | — | — | — | — | — | — | — | — | — | — | — |
| IP10[11:8] | SD3_ DAT1 | — | NFDATA 1 | — | — | — | — | — | — | — | — | — | — | — |

| Register | IPSR = H'0 | IPSR = H'1 | IPSR = H'2 | IPSR = H'3 | IPSR = H'4 | IPSR = H'5 | IPSR = H'6 | IPSR = H'7 | IPSR = H'8 | IPSR = H'9 | IPSR = H'A | IPSR = H'B | IPSR = H'C | IPSR = H'D |
|-------------|---------------|---------------|----------------|---------------|---------------|---------------|---------------|---------------|-----------------|---------------|---------------|---------------|---------------|---------------|
| IP10[15:12] | SD3_ DAT2 | — | NFDATA 2 | — | — | — | — | — | — | — | — | — | — | — |
| IP10[19:16] | SD3_ DAT3 | — | NFDATA 3 | — | — | — | — | — | — | — | — | — | — | — |
| IP10[23:20] | SD3_ DAT4 | SD2_ CD_A | NFDATA 4 | — | — | — | — | — | — | — | — | — | — | — |
| IP10[27:24] | SD3_ DAT5 | SD2_ WP_A | NFDATA 5 | — | — | — | — | — | — | — | — | — | — | — |
| IP10[31:28] | SD3_ DAT6 | SD3_CD | NFDATA 6 | — | — | — | — | — | — | — | — | — | — | — |
| IP11[3:0] | SD3_ DAT7 | SD3_WP | NFDATA 7 | — | — | — | — | — | — | — | — | — | — | — |
| IP11[7:4] | SD3_DS | — | NFCLE | — | — | — | — | — | — | — | — | — | — | — |
| IP11[11:8] | SD0_CD | — | NFDATA 14_A | — | SCL2_B | — | — | — | — | — | — | — | — | — |
| IP11[15:12] | SD0_WP | — | NFDATA 15_A | — | SDA2_B | — | — | — | — | — | — | — | — | — |
| IP11[19:16] | SD1_CD | — | NFRB#_A | — | — | — | — | — | — | — | — | — | — | — |
| IP11[23:20] | SD1_WP | — | NFCE#_A | — | — | — | — | — | — | — | — | — | — | — |
| IP11[27:24] | SCK0 | HSCK1_B | MSIOF1_ SS2_B | AUDIO_ CLKC_B | SDA2_A | — | — | — | — | — | SCK5_B | — | — | — |
| IP11[31:28] | RX0 | HRX1_B | — | — | — | — | — | — | — | — | — | — | — | — |
| IP12[3:0] | TX0 | HTX1_B | — | — | — | — | — | — | — | — | — | — | — | — |
| IP12[7:4] | CTS0# | HCTS1#_B | MSIOF1_ SYNC_B | — | — | — | — | — | AUDIO_ CLKOUT_C | — | — | — | — | — |
| IP12[11:8] | RTS0# | HRTS1#_B | MSIOF1_ SS1_B | AUDIO_ CLKA_B | SCL2_A | — | — | — | — | — | — | — | — | — |
| IP12[15:12] | RX1_A | HRX1_A | — | — | — | — | — | — | — | — | — | — | — | — |
| IP12[19:16] | TX1_A | HTX1_A | — | — | — | — | — | — | — | — | — | — | — | — |
| IP12[23:20] | CTS1# | HCTS1#_A | MSIOF1_ RXD_B | — | — | — | — | — | — | — | — | — | — | — |
| IP12[27:24] | RTS1# | HRTS1#_A | MSIOF1_ TXD_B | — | — | — | — | — | — | — | — | — | — | — |
| IP12[31:28] | SCK2 | SCIF_ CLK_B | MSIOF1_ SCK_B | — | — | — | — | — | — | — | — | — | — | — |
| IP13[3:0] | TX2_A | — | — | SD2_CD_B | SCL1_A | — | — | — | — | — | — | — | — | — |
| IP13[7:4] | RX2_A | — | — | SD2_WP_B | SDA1_A | — | — | — | — | — | — | — | — | — |
| IP13[11:8] | HSCK0 | — | MSIOF1_ SCK_D | AUDIO_ CLKB_A | SSI_ SDATA1_B | — | — | — | — | — | RX5_B | — | — | — |
| IP13[15:12] | HRX0 | — | MSIOF1_ RXD_D | — | SSI_ SDATA2_B | — | — | — | — | — | — | — | — | — |

| Register | IPSR = H'0 | IPSR = H'1 | IPSR = H'2 | IPSR = H'3 | IPSR = H'4 | IPSR = H'5 | IPSR = H'6 | IPSR = H'7 | IPSR = H'8 | IPSR = H'9 | IPSR = H'A | IPSR = H'B | IPSR = H'C | IPSR = H'D |
|-----------------|----------------------|---------------|-------------------|------------------|----------------------|---------------|---------------|---------------|-------------------------|---------------|---------------|---------------|---------------|---------------|
| IP13[19:16] | HTX0 | — | MSIOF1_ TXD_D | — | SSL_ SDATA9 _B | — | — | — | — | — | — | — | — | — |
| IP13[23:20] | HCTS0# | RX2_B | MSIOF1_ SYNC_D | — | SSL_ SCK9_A | — | — | — | AUDIO_ CLKOUT 1_A | — | — | — | — | — |
| IP13[27:24] | HRTS0# | TX2_B | MSIOF1_ SS1_D | — | SSL_WS9 _A | — | — | — | AUDIO_ CLKOUT 2_A | — | — | — | — | — |
| IP13[31:28] | MSIOF0_ SYNC | — | — | — | — | — | — | — | AUDIO_ CLKOUT _A | — | TX5_B | — | — | — |
| IP14[3:0] | MSIOF0_ SS1 | RX5_A | NFWP#_ A | AUDIO_ CLKA_C | SSL_ SCK2_A | — | — | — | AUDIO_ CLKOUT 3_A | — | TCLK1_B | — | — | — |
| IP14[7:4] | MSIOF0_ SS2 | TX5_A | MSIOF1_ SS2_D | AUDIO_ CLKC_A | SSL_WS2 _A | — | — | — | AUDIO_ CLKOUT _D | — | — | — | — | — |
| IP14[11:8] | MLB_CL K | — | MSIOF1_ SCK_F | — | SCL1_B | — | — | — | — | — | — | — | — | — |
| IP14[15:12] | MLB_SIG | RX1_B | MSIOF1_ SYNC_F | — | SDA1_B | — | — | — | — | — | — | — | — | — |
| IP14 [19:16] | MLB_ DAT | TX1_B | MSIOF1_ RXD_F | — | — | — | — | — | — | — | — | — | — | — |
| IP14[23:20] | SSL_ SCK0123 9 | — | MSIOF1_ TXD_F | — | — | — | — | — | — | — | — | — | — | — |
| IP14[27:24] | SSL_ WS01239 | — | MSIOF1_ SS1_F | — | — | — | — | — | — | — | — | — | — | — |
| IP14[31:28] | SSL_ SDATA0 | — | MSIOF1_ SS2_F | — | — | — | — | — | — | — | — | — | — | — |
| IP15[3:0] | SSL_ SDATA1 _A | — | — | — | — | — | — | — | — | — | — | — | — | — |
| IP15[7:4] | SSL_ SDATA2 _A | — | — | — | SSL_ SCK1_B | — | — | — | — | — | — | — | — | — |
| IP15[11:8] | SSL_ SCK349 | — | MSIOF1_ SS1_A | — | — | — | — | — | — | — | — | — | — | — |
| IP15[15:12] | SSL_ WS349 | HCTS2#_ A | MSIOF1_ SS2_A | — | — | — | — | — | — | — | — | — | — | — |
| IP15[19:16] | SSL_ SDATA3 | HRTS2#_ A | MSIOF1_ TXD_A | — | — | — | — | — | — | — | — | — | — | — |
| IP15[23:20] | SSL_ SCK4 | HRX2_A | MSIOF1_ SCK_A | — | — | — | — | — | — | — | — | — | — | — |
| IP15[27:24] | SSL_WS4 | HTX2_A | MSIOF1_ SYNC_A | — | — | — | — | — | — | — | — | — | — | — |
| IP15[31:28] | SSL_ SDATA4 | HSCK2_ A | MSIOF1_ RXD_A | — | — | — | — | — | — | — | — | — | — | — |

| Register | IPSR = H'0 | IPSR = H'1 | IPSR = H'2 | IPSR = H'3 | IPSR = H'4 | IPSR = H'5 | IPSR = H'6 | IPSR = H'7 | IPSR = H'8 | IPSR = H'9 | IPSR = H'A | IPSR = H'B | IPSR = H'C | IPSR = H'D |
|-------------|----------------------|--|-------------------|------------------------|----------------|---------------|---------------|---------------|--|---------------|---------------|---------------|---------------|---------------|
| IP16[3:0] | SSL_ SCK6 | [RZ/G2H] USB2_ PWEN *1 [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] — *1 | — | — | — | — | — | — | — | — | — | — | — | — |
| IP16[7:4] | SSI_WS6 | [RZ/G2H] USB2_ OVC *1 [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] — *1 | — | — | — | — | — | — | — | — | — | — | — | — |
| IP16[11:8] | SSI_ SDATA6 | — | — | — | — | — | — | — | [RZ/G2H] [RZ/G2N] SATA_ DEVSLP _A *1 [RZ/G2M V1.3] [RZ/G2M V3.0] — *1 | — | — | — | — | — |
| IP16[15:12] | SSI_ SCK78 | HRX2_B | MSIOF1_ SCK_C | — | — | — | — | — | — | — | — | — | — | — |
| IP16[19:16] | SSI_ WS78 | HTX2_B | MSIOF1_ SYNC_C | — | — | — | — | — | — | — | — | — | — | — |
| IP16[23:20] | SSI_ SDATA7 | HCTS2#_ B | MSIOF1_ RXD_C | — | — | — | — | — | — | — | TCLK2_A | — | — | — |
| IP16[27:24] | SSI_ SDATA8 | HRTS2#_ B | MSIOF1_ TXD_C | — | — | — | — | — | — | — | — | — | — | — |
| IP16[31:28] | SSI_ SDATA9 _A | HCK2_ B | MSIOF1_ SS1_C | HCK1_ A | SSI_WS1 _B | SCK1 | — | SCK5_A | — | — | — | — | — | — |
| IP17[3:0] | AUDIO_ CLKA_A | — | — | — | — | — | — | — | — | — | — | — | — | — |
| IP17[7:4] | AUDIO_ CLKB_B | SCIF_ CLK_A | — | — | — | — | — | — | — | — | TCLK1_A | — | — | — |
| IP17[11:8] | USB0_ PWEN | — | — | — | — | — | — | — | — | — | — | — | — | HCK2_ C |
| IP17[15:12] | USB0_ OVC | — | — | — | — | — | — | — | — | — | — | — | — | HRX2_C |
| IP17[19:16] | USB1_ PWEN | — | — | — | SSI_ SCK1_A | — | — | — | — | — | — | — | — | HTX2_C |
| IP17[23:20] | USB1_ OVC | — | MSIOF1_ SS2_C | — | SSI_ WS1_A | — | — | — | — | — | — | — | — | HCTS2#_ C |
| IP17[27:24] | USB30_ PWEN | — | — | AUDIO_ CLKOUT _B | SSI_ SCK2_B | — | — | — | — | — | TCLK2_B | TPU0T0 0 | — | HRTS2#_ C |

| Register | IPSR = H'0 | IPSR = H'1 | IPSR = H'2 | IPSR = H'3 | IPSR = H'4 | IPSR = H'5 | IPSR = H'6 | IPSR = H'7 | IPSR = H'8 | IPSR = H'9 | IPSR = H'A | IPSR = H'B | IPSR = H'C | IPSR = H'D |
|------------------|--|---------------|---------------|-------------------------|----------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| IP17[31:28] | USB30_ OVC | — | — | AUDIO_ CLKOUT 1_B | SSI_ WS2_B | — | — | — | — | — | FSO_ TOE# | TPU0TO 1 | — | — |
| IP18[3:0] | [RZ/G2H] [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] GP6_30 *1 | — | — | AUDIO_ CLKOUT 2_B | SSI_ SCK9_B | — | — | — | — | — | — | TPU0TO 2 | — | — |
| IP18[7:4] | [RZ/G2H] [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] GP6_31 *1 | — | — | AUDIO_ CLKOUT 3_B | SSI_ WS9_B | — | — | — | — | — | — | TPU0TO 3 | — | — |
| IP18[31:8] *2 | — | — | — | — | — | — | — | — | — | — | — | — | — | — |

Note 1 Different for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N

2 These bits are reserved. There is no effect setting any value from B'0000(H'0) to B'1101(H'D) to the bits.

8.2.4 DRV Control Register 0-24 (DRVCTRL0-24)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: DRVCTRL0-24 controls the driving abilities of pins. This setting is related to only the output.

| Output buffer | | | Drive capability | | | | | Output buffer | |
|---------------|------|------|---------------------|-----------|------|------|------------------|---------------|--|
| DRV1 | DRV2 | DRV3 | Operating condition | | DRV1 | DRV2 | Drive capability | | |
| | | | for 1.8 V | for 3.3 V | | | | | |
| L | L | L | 2/8 | 2/8 | L | L | 1/4 | | |
| H | L | L | 2/8 | 2/8 | H | L | 2/4 | | |
| L | H | L | 3/8 | 3/8 | L | H | 3/4 | | |
| H | H | L | 4/8 | 4/8 | H | H | Full | | |
| L | L | H | 5/8 | 5/8 | | | | | |
| H | L | H | 6/8 | 6/8 | | | | | |
| L | H | H | 7/8 | 7/8 | | | | | |
| H | H | H | Full | Full | | | | | |

The value of these bits must be 11 in the case of 2bit (DRV2, DRV1).

The value of these bits must be 111 in the case of 3bit (DRV3, DRV2, DRV1).

| | | | | | | | | | | | | | | | | |
|------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | DRVn* [31] | DRVn* [30] | DRVn* [29] | DRVn* [28] | DRVn* [27] | DRVn* [26] | DRVn* [25] | DRVn* [24] | DRVn* [23] | DRVn* [22] | DRVn* [21] | DRVn* [20] | DRVn* [19] | DRVn* [18] | DRVn* [17] | DRVn* [16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DRVn* [15] | DRVn* [14] | DRVn* [13] | DRVn* [12] | DRVn* [11] | DRVn* [10] | DRVn* [9] | DRVn* [8] | DRVn* [7] | DRVn* [6] | DRVn* [5] | DRVn* [4] | DRVn* [3] | DRVn* [2] | DRVn* [1] | DRVn* [0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * n = 0-24

Note: To enable this register to be set, appropriately set the multiplexed pin setting mask register (PMMR) immediately before setting this register.

Table 8.6 Configuration of Registers in DRVCTRL0-3

| bit | DRVCTRL0 | Initial value | | | | DRVCTRL1 | Initial value | | | | DRVCTRL2 | Initial value | | | | DRVCTRL3 | Initial value | | | |
|-------|---------------------|---------------|-------------|-------------|---------------------|----------|---------------|-------------|-----------------|--------|----------|---------------|--------------------|-------------|--------|----------|---------------|-------------|-------------|--------|
| | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N |
| bit31 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | | | |
| bit30 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | DRV3_AVB_RXC | 1 | 1 | 1 | DRV3_AVB_TD1 | 1 | 1 | 1 | | | | |
| bit29 | DRV2_QSPI0_SPCLK | 1 | 1 | 1 | DRV2_QSPI1_MISO/IO1 | 1 | 1 | 1 | DRV2_AVB_RXC | 1 | 1 | 1 | DRV2_AVB_TD1 | 1 | 1 | 1 | | | | |
| bit28 | DRV1_QSPI0_SPCLK | 1 | 1 | 1 | DRV1_QSPI1_MISO/IO1 | 1 | 1 | 1 | DRV1_AVB_RXC | 1 | 1 | 1 | DRV1_AVB_TD1 | 1 | 1 | 1 | | | | |
| bit27 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | | | |
| bit26 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | DRV3_AVB_RD0 | 1 | 1 | 1 | DRV3_AVB_TD2 | 1 | 1 | 1 | | | | |
| bit25 | DRV2_QSPI0_MOSI/IO0 | 1 | 1 | 1 | DRV2_QSPI1_IO2 | 1 | 1 | 1 | DRV2_AVB_RD0 | 1 | 1 | 1 | DRV2_AVB_TD2 | 1 | 1 | 1 | | | | |
| bit24 | DRV1_QSPI0_MOSI/IO0 | 1 | 1 | 1 | DRV1_QSPI1_IO2 | 1 | 1 | 1 | DRV1_AVB_RD0 | 1 | 1 | 1 | DRV1_AVB_TD2 | 1 | 1 | 1 | | | | |
| bit23 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | | | |
| bit22 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | DRV3_AVB_RD1 | 1 | 1 | 1 | DRV3_AVB_TD3 | 1 | 1 | 1 | | | | |
| bit21 | DRV2_QSPI0_MISO/IO1 | 1 | 1 | 1 | DRV2_QSPI1_IO3 | 1 | 1 | 1 | DRV2_AVB_RD1 | 1 | 1 | 1 | DRV2_AVB_TD3 | 1 | 1 | 1 | | | | |
| bit20 | DRV1_QSPI0_MISO/IO1 | 1 | 1 | 1 | DRV1_QSPI1_IO3 | 1 | 1 | 1 | DRV1_AVB_RD1 | 1 | 1 | 1 | DRV1_AVB_TD3 | 1 | 1 | 1 | | | | |
| bit19 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | | | |
| bit18 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | DRV3_AVB_RD2 | 1 | 1 | 1 | DRV3_AVB_TXCREFLCK | 1 | 1 | 1 | | | | |
| bit17 | DRV2_QSPI0_IO2 | 1 | 1 | 1 | DRV2_QSPI1_SSL | 1 | 1 | 1 | DRV2_AVB_RD2 | 1 | 1 | 1 | DRV2_AVB_TXCREFLCK | 1 | 1 | 1 | | | | |
| bit16 | DRV1_QSPI0_IO2 | 1 | 1 | 1 | DRV1_QSPI1_SSL | 1 | 1 | 1 | DRV1_AVB_RD2 | 1 | 1 | 1 | DRV1_AVB_TXCREFLCK | 1 | 1 | 1 | | | | |
| bit15 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | | | |
| bit14 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | DRV3_AVB_RD3 | 1 | 1 | 1 | DRV3_AVB_MDIO | 1 | 1 | 1 | | | | |
| bit13 | DRV2_QSPI0_IO3 | 1 | 1 | 1 | DRV2_RPC_INT# | 1 | 1 | 1 | DRV2_AVB_RD3 | 1 | 1 | 1 | DRV2_AVB_MDIO | 1 | 1 | 1 | | | | |
| bit12 | DRV1_QSPI0_IO3 | 1 | 1 | 1 | DRV1_RPC_INT# | 1 | 1 | 1 | DRV1_AVB_RD3 | 1 | 1 | 1 | DRV1_AVB_MDIO | 1 | 1 | 1 | | | | |
| bit11 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | | | |
| bit10 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | DRV3_AVB_TX_CTL | 1 | 1 | 1 | DRV3_AVB_MDC | 1 | 1 | 1 | | | | |
| bit9 | DRV2_QSPI0_SSL | 1 | 1 | 1 | DRV2_RPC_WP# | 1 | 1 | 1 | DRV2_AVB_TX_CTL | 1 | 1 | 1 | DRV2_AVB_MDC | 1 | 1 | 1 | | | | |
| bit8 | DRV1_QSPI0_SSL | 1 | 1 | 1 | DRV1_RPC_WP# | 1 | 1 | 1 | DRV1_AVB_TX_CTL | 1 | 1 | 1 | DRV1_AVB_MDC | 1 | 1 | 1 | | | | |
| bit7 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | | | |

| | | Initial value | | | | Initial value | | | | Initial value | | | | Initial value | | | | | | | | | | |
|----------|---------------------|---------------|-------------|-------------|--------|-----------------|---|--------|-------------|---------------|--------|----------|--------------|---------------|-------------|-------------|--------|----------|---|--------|-------------|-------------|--------|---|
| DRVCTRL0 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL1 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL2 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL3 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | |
| bit6 | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit5 | DRV2_QSPI1_SPCLK | 1 | 1 | 1 | — | DRV2_RPC_RESET# | — | 1 | 1 | 1 | — | — | DRV2_AVB_TXC | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit4 | DRV1_QSPI1_SPCLK | 1 | 1 | 1 | — | DRV1_RPC_RESET# | — | 1 | 1 | 1 | — | — | DRV1_AVB_TXC | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit3 | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit2 | — | 1 | 1 | 1 | — | DRV3_AVB_RX_CTL | — | 1 | 1 | 1 | — | — | DRV3_AVB_TD0 | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit1 | DRV2_QSPI1_MOSI/IO0 | 1 | 1 | 1 | — | DRV2_AVB_RX_CTL | — | 1 | 1 | 1 | — | — | DRV2_AVB_TD0 | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit0 | DRV1_QSPI1_MOSI/IO0 | 1 | 1 | 1 | — | DRV1_AVB_RX_CTL | — | 1 | 1 | 1 | — | — | DRV1_AVB_TD0 | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |

Table 8.7 Configuration of Registers in DRVCTRL4-7

| | | Initial value | | | | Initial value | | | | Initial value | | | | Initial value | | | | | | | | | | |
|----------|-----------------------|---------------|-------------|-------------|--------|---------------|---|--------|-------------|---------------|--------|----------|---------|---------------|-------------|-------------|--------|----------|---|--------|-------------|-------------|--------|---|
| DRVCTRL4 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL5 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL6 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL7 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | |
| bit31 | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit30 | DRV3_AVB_LINK | 1 | 1 | 1 | — | DRV3_IRQ5 | — | 1 | 1 | 1 | — | — | DRV3_A4 | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit29 | DRV2_AVB_LINK | 1 | 1 | 1 | — | DRV2_IRQ5 | — | 1 | 1 | 1 | — | — | DRV2_A4 | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit28 | DRV1_AVB_LINK | 1 | 1 | 1 | — | DRV1_IRQ5 | — | 1 | 1 | 1 | — | — | DRV1_A4 | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit27 | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit26 | DRV3_AVB_AVTP_MATCH | 1 | 1 | 1 | — | DRV3_PWM0 | — | 1 | 1 | 1 | — | — | DRV3_A5 | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit25 | DRV2_AVB_AVTP_MATCH | 1 | 1 | 1 | — | DRV2_PWM0 | — | 1 | 1 | 1 | — | — | DRV2_A5 | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit24 | DRV1_AVB_AVTP_MATCH | 1 | 1 | 1 | — | DRV1_PWM0 | — | 1 | 1 | 1 | — | — | DRV1_A5 | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit23 | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit22 | DRV3_AVB_AVTP_CAPTURE | 1 | 1 | 1 | — | DRV3_PWM1 | — | 1 | 1 | 1 | — | — | DRV3_A6 | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit21 | DRV2_AVB_AVTP_CAPTURE | 1 | 1 | 1 | — | DRV2_PWM1 | — | 1 | 1 | 1 | — | — | DRV2_A6 | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit20 | DRV1_AVB_AVTP_CAPTURE | 1 | 1 | 1 | — | DRV1_PWM1 | — | 1 | 1 | 1 | — | — | DRV1_A6 | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |
| bit19 | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — | — | 1 | 1 | 1 | — | — |

| DRVCTRL4 | | Initial value | | | | DRVCTRL5 | | | | DRVCTRL6 | | | | DRVCTRL7 | | | |
|----------|-----------|---------------|----------------------------|--------|-----------|----------|---|---|----------|----------|---|---|----------|----------|---|---|--|
| | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | | | | | | | | | | | | | |
| bit18 | DRV3_IRQ0 | 1 | 1 | 1 | DRV3_PWM2 | 1 | 1 | 1 | DRV3_A7 | 1 | 1 | 1 | DRV3_A15 | 1 | 1 | 1 | |
| bit17 | DRV2_IRQ0 | 1 | 1 | 1 | DRV2_PWM2 | 1 | 1 | 1 | DRV2_A7 | 1 | 1 | 1 | DRV2_A15 | 1 | 1 | 1 | |
| bit16 | DRV1_IRQ0 | 1 | 1 | 1 | DRV1_PWM2 | 1 | 1 | 1 | DRV1_A7 | 1 | 1 | 1 | DRV1_A15 | 1 | 1 | 1 | |
| bit15 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | |
| bit14 | DRV3_IRQ1 | 1 | 1 | 1 | DRV3_A0 | 1 | 1 | 1 | DRV3_A8 | 1 | 1 | 1 | DRV3_A16 | 1 | 1 | 1 | |
| bit13 | DRV2_IRQ1 | 1 | 1 | 1 | DRV2_A0 | 1 | 1 | 1 | DRV2_A8 | 1 | 1 | 1 | DRV2_A16 | 1 | 1 | 1 | |
| bit12 | DRV1_IRQ1 | 1 | 1 | 1 | DRV1_A0 | 1 | 1 | 1 | DRV1_A8 | 1 | 1 | 1 | DRV1_A16 | 1 | 1 | 1 | |
| bit11 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | |
| bit10 | DRV3_IRQ2 | 1 | 1 | 1 | DRV3_A1 | 1 | 1 | 1 | DRV3_A9 | 1 | 1 | 1 | DRV3_A17 | 1 | 1 | 1 | |
| bit9 | DRV2_IRQ2 | 1 | 1 | 1 | DRV2_A1 | 1 | 1 | 1 | DRV2_A9 | 1 | 1 | 1 | DRV2_A17 | 1 | 1 | 1 | |
| bit8 | DRV1_IRQ2 | 1 | 1 | 1 | DRV1_A1 | 1 | 1 | 1 | DRV1_A9 | 1 | 1 | 1 | DRV1_A17 | 1 | 1 | 1 | |
| bit7 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | |
| bit6 | DRV3_IRQ3 | 1 | 1 | 1 | DRV3_A2 | 1 | 1 | 1 | DRV3_A10 | 1 | 1 | 1 | DRV3_A18 | 1 | 1 | 1 | |
| bit5 | DRV2_IRQ3 | 1 | 1 | 1 | DRV2_A2 | 1 | 1 | 1 | DRV2_A10 | 1 | 1 | 1 | DRV2_A18 | 1 | 1 | 1 | |
| bit4 | DRV1_IRQ3 | 1 | 1 | 1 | DRV1_A2 | 1 | 1 | 1 | DRV1_A10 | 1 | 1 | 1 | DRV1_A18 | 1 | 1 | 1 | |
| bit3 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | |
| bit2 | DRV3_IRQ4 | 1 | 1 | 1 | DRV3_A3 | 1 | 1 | 1 | DRV3_A11 | 1 | 1 | 1 | DRV3_A19 | 1 | 1 | 1 | |
| bit1 | DRV2_IRQ4 | 1 | 1 | 1 | DRV2_A3 | 1 | 1 | 1 | DRV2_A11 | 1 | 1 | 1 | DRV2_A19 | 1 | 1 | 1 | |
| bit0 | DRV1_IRQ4 | 1 | 1 | 1 | DRV1_A3 | 1 | 1 | 1 | DRV1_A11 | 1 | 1 | 1 | DRV1_A19 | 1 | 1 | 1 | |

Table 8.8 Configuration of Registers in DRVCTRL8-11

| DRVCTRL8 | | Initial value | | | | DRVCTRL9 | | | | DRVCTRL10 | | | | DRVCTRL11 | | | |
|----------|-------------|---------------|----------------------------|--------|------------------|----------|---|---|---------|-----------|---|---|-----------|-----------|---|---|--|
| | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | | | | | | | | | | | | | |
| bit31 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | |
| bit30 | DRV3_CLKOUT | 1 | 1 | 1 | DRV3_EX_WAIT0 | 1 | 1 | 1 | DRV3_D6 | 1 | 1 | 1 | DRV3_D14 | 1 | 1 | 1 | |
| bit29 | DRV2_CLKOUT | 1 | 1 | 1 | DRV2_EX_WAIT0 | 1 | 1 | 1 | DRV2_D6 | 1 | 1 | 1 | DRV2_D14 | 1 | 1 | 1 | |
| bit28 | DRV1_CLKOUT | 1 | 1 | 1 | DRV1_EX_WAIT0 | 1 | 1 | 1 | DRV1_D6 | 1 | 1 | 1 | DRV1_D14 | 1 | 1 | 1 | |
| bit27 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | |
| bit26 | DRV3_CS0# | 1 | 1 | 1 | DRV3_ PRESETOUT# | 1 | 1 | 1 | DRV3_D7 | 1 | 1 | 1 | DRV3_D15 | 1 | 1 | 1 | |
| bit25 | DRV2_CS0# | 1 | 1 | 1 | DRV2_ PRESETOUT# | 1 | 1 | 1 | DRV2_D7 | 1 | 1 | 1 | DRV2_D15 | 1 | 1 | 1 | |
| bit24 | DRV1_CS0# | 1 | 1 | 1 | DRV1_ PRESETOUT# | 1 | 1 | 1 | DRV1_D7 | 1 | 1 | 1 | DRV1_D15 | 1 | 1 | 1 | |
| bit23 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | |
| bit22 | DRV3_CS1# | 1 | 1 | 1 | DRV3_D0 | 1 | 1 | 1 | DRV3_D8 | 1 | 1 | 1 | DRV3_AVS1 | 1 | 1 | 1 | |
| bit21 | DRV2_CS1# | 1 | 1 | 1 | DRV2_D0 | 1 | 1 | 1 | DRV2_D8 | 1 | 1 | 1 | DRV2_AVS1 | 1 | 1 | 1 | |
| bit20 | DRV1_CS1# | 1 | 1 | 1 | DRV1_D0 | 1 | 1 | 1 | DRV1_D8 | 1 | 1 | 1 | DRV1_AVS1 | 1 | 1 | 1 | |

| DRVCTRL8 | Initial value | | | | DRVCTRL9 | Initial value | | | | DRVCTRL10 | Initial value | | | | DRVCTRL11 | Initial value | | | |
|----------|---------------|-------------|-------------|--------|----------|---------------|-------------|-------------|----------|-----------|---------------|-------------|-------------------|--------|-----------|---------------|-------------|-------------|--------|
| | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N |
| bit19 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | | |
| bit18 | DRV3_BS# | 1 | 1 | 1 | DRV3_D1 | 1 | 1 | 1 | DRV3_D9 | 1 | 1 | 1 | DRV3_AVS2 | 1 | 1 | 1 | | | |
| bit17 | DRV2_BS# | 1 | 1 | 1 | DRV2_D1 | 1 | 1 | 1 | DRV2_D9 | 1 | 1 | 1 | DRV2_AVS2 | 1 | 1 | 1 | | | |
| bit16 | DRV1_BS# | 1 | 1 | 1 | DRV1_D1 | 1 | 1 | 1 | DRV1_D9 | 1 | 1 | 1 | DRV1_AVS2 | 1 | 1 | 1 | | | |
| bit15 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | | |
| bit14 | DRV3_RD# | 1 | 1 | 1 | DRV3_D2 | 1 | 1 | 1 | DRV3_D10 | 1 | 1 | 1 | DRV3_GP7_02 | 1 | 1 | 1 | | | |
| bit13 | DRV2_RD# | 1 | 1 | 1 | DRV2_D2 | 1 | 1 | 1 | DRV2_D10 | 1 | 1 | 1 | DRV2_GP7_02 | 1 | 1 | 1 | | | |
| bit12 | DRV1_RD# | 1 | 1 | 1 | DRV1_D2 | 1 | 1 | 1 | DRV1_D10 | 1 | 1 | 1 | DRV1_GP7_02 | 1 | 1 | 1 | | | |
| bit11 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | | |
| bit10 | DRV3_RD/WR# | 1 | 1 | 1 | DRV3_D3 | 1 | 1 | 1 | DRV3_D11 | 1 | 1 | 1 | DRV3_GP7_03 | 1 | 1 | 1 | | | |
| bit9 | DRV2_RD/WR# | 1 | 1 | 1 | DRV2_D3 | 1 | 1 | 1 | DRV2_D11 | 1 | 1 | 1 | DRV2_GP7_03 | 1 | 1 | 1 | | | |
| bit8 | DRV1_RD/WR# | 1 | 1 | 1 | DRV1_D3 | 1 | 1 | 1 | DRV1_D11 | 1 | 1 | 1 | DRV1_GP7_03 | 1 | 1 | 1 | | | |
| bit7 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | | |
| bit6 | DRV3_WE0# | 1 | 1 | 1 | DRV3_D4 | 1 | 1 | 1 | DRV3_D12 | 1 | 1 | 1 | — | 1 | 1 | 1 | | | |
| bit5 | DRV2_WE0# | 1 | 1 | 1 | DRV2_D4 | 1 | 1 | 1 | DRV2_D12 | 1 | 1 | 1 | DRV2_DU_DOTCLKIN0 | 1 | 1 | 1 | | | |
| bit4 | DRV1_WE0# | 1 | 1 | 1 | DRV1_D4 | 1 | 1 | 1 | DRV1_D12 | 1 | 1 | 1 | DRV1_DU_DOTCLKIN0 | 1 | 1 | 1 | | | |
| bit3 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | | |
| bit2 | DRV3_WE1# | 1 | 1 | 1 | DRV3_D5 | 1 | 1 | 1 | DRV3_D13 | 1 | 1 | 1 | — | 1 | 1 | 1 | | | |
| bit1 | DRV2_WE1# | 1 | 1 | 1 | DRV2_D5 | 1 | 1 | 1 | DRV2_D13 | 1 | 1 | 1 | DRV2_DU_DOTCLKIN1 | 1 | 1 | 1 | | | |
| bit0 | DRV1_WE1# | 1 | 1 | 1 | DRV1_D5 | 1 | 1 | 1 | DRV1_D13 | 1 | 1 | 1 | DRV1_DU_DOTCLKIN1 | 1 | 1 | 1 | | | |

Table 8.9 Configuration of Registers in DRVCTRL12-15

| DRVCTRL12 | Initial value | | | | DRVCTRL13 | Initial value | | | | DRVCTRL14 | Initial value | | | | DRVCTRL15 | Initial value | | | |
|-----------|--|-------------|-------------|--------|-----------|---------------|-------------|-------------|--------------|-----------|---------------|-------------|---------------|--------|-----------|---------------|-------------|-------------|--------|
| | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N |
| bit31 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | | |
| bit30 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | DRV3_SD1_CLK | 1 | 1 | 1 | DRV3_SD2_DAT0 | 1 | 1 | 1 | | | |
| bit29 | [RZ/G2M V1.3] [RZ/G2M V3.0] DRV2_DU_DOTCLKIN2 [RZ/G2H] [RZ/G2N] — | 1 | 1 | 1 | DRV2_TDO | 1 | 1 | 1 | DRV2_SD1_CLK | 1 | 1 | 1 | DRV2_SD2_DAT0 | 1 | 1 | 1 | | | |

| | Initial value | Initial value | | | | Initial value | Initial value | | | | | | | | | |
|------------------|--|---------------|-------------|-------------|-------------------|---------------|---------------|-------------|-------------------|--------|--------|-------------|-------------------|--------|---|---|
| | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | |
| DRVCTRL12 | | | | | DRVCTRL13 | | | | DRVCTRL14 | | | | DRVCTRL15 | | | |
| bit28 | [RZ/G2M V1.3] [RZ/G2M V3.0] DRV1_DU_ DOTCLKIN2 [RZ/G2H] [RZ/G2N] — | 1 | 1 | 1 | DRV1_TDO | 1 | 1 | 1 | DRV1_SD1_ CLK | 1 | 1 | 1 | DRV1_SD2_ DAT0 | 1 | 1 | 1 |
| bit27 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit26 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | DRV3_SD1_ CMD | 1 | 1 | 1 | DRV3_SD2_ DAT1 | 1 | 1 | 1 |
| bit25 | [RZ/G2H] [RZ/G2N] DRV2_DU_ DOTCLKIN3 [RZ/G2M V1.3] [RZ/G2M V3.0] — | 1 | 1 | 1 | DRV2_ASEBR K | 1 | 1 | 1 | DRV2_SD1_ CMD | 1 | 1 | 1 | DRV2_SD2_ DAT1 | 1 | 1 | 1 |
| bit24 | [RZ/G2H] [RZ/G2N] DRV1_DU_ DOTCLKIN3 [RZ/G2M V1.3] [RZ/G2M V3.0] — | 1 | 1 | 1 | DRV1_ASEBR K | 1 | 1 | 1 | DRV1_SD1_ CMD | 1 | 1 | 1 | DRV1_SD2_ DAT1 | 1 | 1 | 1 |
| bit23 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit22 | — | 1 | 1 | 1 | DRV3_SD0_ CLK | 1 | 1 | 1 | DRV3_SD1_ DAT0 | 1 | 1 | 1 | DRV3_SD2_ DAT2 | 1 | 1 | 1 |
| bit21 | — | 1 | 1 | 1 | DRV2_SD0_ CLK | 1 | 1 | 1 | DRV2_SD1_ DAT0 | 1 | 1 | 1 | DRV2_SD2_ DAT2 | 1 | 1 | 1 |
| bit20 | — | 1 | 1 | 1 | DRV1_SD0_ CLK | 1 | 1 | 1 | DRV1_SD1_ DAT0 | 1 | 1 | 1 | DRV1_SD2_ DAT2 | 1 | 1 | 1 |
| bit19 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit18 | — | 1 | 1 | 1 | DRV3_SD0_ CMD | 1 | 1 | 1 | DRV3_SD1_ DAT1 | 1 | 1 | 1 | DRV3_SD2_ DAT3 | 1 | 1 | 1 |
| bit17 | — | 1 | 1 | 1 | DRV2_SD0_ CMD | 1 | 1 | 1 | DRV2_SD1_ DAT1 | 1 | 1 | 1 | DRV2_SD2_ DAT3 | 1 | 1 | 1 |
| bit16 | — | 1 | 1 | 1 | DRV1_SD0_ CMD | 1 | 1 | 1 | DRV1_SD1_ DAT1 | 1 | 1 | 1 | DRV1_SD2_ DAT3 | 1 | 1 | 1 |
| bit15 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit14 | — | 1 | 1 | 1 | DRV3_SD0_ DAT0 | 1 | 1 | 1 | DRV3_SD1_ DAT2 | 1 | 1 | 1 | DRV3_SD2_ DS | 1 | 1 | 1 |
| bit13 | — | 1 | 1 | 1 | DRV2_SD0_ DAT0 | 1 | 1 | 1 | DRV2_SD1_ DAT2 | 1 | 1 | 1 | DRV2_SD2_ DS | 1 | 1 | 1 |
| bit12 | — | 1 | 1 | 1 | DRV1_SD0_ DAT0 | 1 | 1 | 1 | DRV1_SD1_ DAT2 | 1 | 1 | 1 | DRV1_SD2_ DS | 1 | 1 | 1 |
| bit11 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit10 | — | 1 | 1 | 1 | DRV3_SD0_ — | 1 | 1 | 1 | DRV3_SD1_ — | 1 | 1 | 1 | DRV3_SD3_ — | 1 | 1 | 1 |

| | | Initial value | | | | Initial value | | | | Initial value | | | | Initial value | | | |
|-----------|----------|---------------|-------------|-------------|--------|---------------|-------------|-------------|--------|---------------|-------------|-------------|--------|---------------|-------------|-------------|--------|
| DRVCTRL12 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL13 | | | | DRVCTRL14 | | | | DRVCTRL15 | | | |
| | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N |
| | | | | | | DAT1 | | | | DAT3 | | | | CLK | | | |
| bit9 | — | 1 | 1 | 1 | 1 | DRV2_SD0_ | 1 | 1 | 1 | DRV2_SD1_ | 1 | 1 | 1 | DRV2_SD3_ | 1 | 1 | 1 |
| | | | | | | DAT1 | | | | DAT3 | | | | CLK | | | |
| bit8 | — | 1 | 1 | 1 | 1 | DRV1_SD0_ | 1 | 1 | 1 | DRV1_SD1_ | 1 | 1 | 1 | DRV1_SD3_ | 1 | 1 | 1 |
| | | | | | | DAT1 | | | | DAT3 | | | | CLK | | | |
| bit7 | — | 1 | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit6 | — | 1 | 1 | 1 | 1 | DRV3_SD0_ | 1 | 1 | 1 | DRV3_SD2_ | 1 | 1 | 1 | DRV3_SD3_ | 1 | 1 | 1 |
| | | | | | | DAT2 | | | | CLK | | | | CMD | | | |
| bit5 | DRV2_TMS | 1 | 1 | 1 | 1 | DRV2_SD0_ | 1 | 1 | 1 | DRV2_SD2_ | 1 | 1 | 1 | DRV2_SD3_ | 1 | 1 | 1 |
| | | | | | | DAT2 | | | | CLK | | | | CMD | | | |
| bit4 | DRV1_TMS | 1 | 1 | 1 | 1 | DRV1_SD0_ | 1 | 1 | 1 | DRV1_SD2_ | 1 | 1 | 1 | DRV1_SD3_ | 1 | 1 | 1 |
| | | | | | | DAT2 | | | | CLK | | | | CMD | | | |
| bit3 | — | 1 | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit2 | — | 1 | 1 | 1 | 1 | DRV3_SD0_ | 1 | 1 | 1 | DRV3_SD2_ | 1 | 1 | 1 | DRV3_SD3_ | 1 | 1 | 1 |
| | | | | | | DAT3 | | | | CMD | | | | DAT0 | | | |
| bit1 | — | 1 | 1 | 1 | 1 | DRV2_SD0_ | 1 | 1 | 1 | DRV2_SD2_ | 1 | 1 | 1 | DRV2_SD3_ | 1 | 1 | 1 |
| | | | | | | DAT3 | | | | CMD | | | | DAT0 | | | |
| bit0 | — | 1 | 1 | 1 | 1 | DRV1_SD0_ | 1 | 1 | 1 | DRV1_SD2_ | 1 | 1 | 1 | DRV1_SD3_ | 1 | 1 | 1 |
| | | | | | | DAT3 | | | | CMD | | | | DAT0 | | | |

Table 8.10 Configuration of Registers in DRVCTRL16-19

| | | Initial value | | | | Initial value | | | | Initial value | | | | Initial value | | | |
|-----------|-----------|---------------|-------------|-------------|--------|---------------|-------------|-------------|--------|---------------|-------------|-------------|--------|---------------|-------------|-------------|--------|
| DRVCTRL16 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL17 | | | | DRVCTRL18 | | | | DRVCTRL19 | | | |
| | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N |
| bit31 | — | 1 | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit30 | DRV3_SD3_ | 1 | 1 | 1 | 1 | DRV3_SD0_ | 1 | 1 | 1 | DRV3_RTS0# | 1 | 1 | 1 | DRV3_HSCK0 | 1 | 1 | 1 |
| | | | | | | DAT1 | | | | CD | | | | | | | |
| bit29 | DRV2_SD3_ | 1 | 1 | 1 | 1 | DRV2_SD0_ | 1 | 1 | 1 | DRV2_RTS0# | 1 | 1 | 1 | DRV2_HSCK0 | 1 | 1 | 1 |
| | | | | | | DAT1 | | | | CD | | | | | | | |
| bit28 | DRV1_SD3_ | 1 | 1 | 1 | 1 | DRV1_SD0_ | 1 | 1 | 1 | DRV1_RTS0# | 1 | 1 | 1 | DRV1_HSCK0 | 1 | 1 | 1 |
| | | | | | | DAT1 | | | | CD | | | | | | | |
| bit27 | — | 1 | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit26 | DRV3_SD3_ | 1 | 1 | 1 | 1 | DRV3_SD0_ | 1 | 1 | 1 | DRV3_RX1 | 1 | 1 | 1 | DRV3_HRX0 | 1 | 1 | 1 |
| | | | | | | DAT2 | | | | WP | | | | | | | |
| bit25 | DRV2_SD3_ | 1 | 1 | 1 | 1 | DRV2_SD0_ | 1 | 1 | 1 | DRV2_RX1 | 1 | 1 | 1 | DRV2_HRX0 | 1 | 1 | 1 |
| | | | | | | DAT2 | | | | WP | | | | | | | |
| bit24 | DRV1_SD3_ | 1 | 1 | 1 | 1 | DRV1_SD0_ | 1 | 1 | 1 | DRV1_RX1 | 1 | 1 | 1 | DRV1_HRX0 | 1 | 1 | 1 |
| | | | | | | DAT2 | | | | WP | | | | | | | |
| bit23 | — | 1 | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit22 | DRV3_SD3_ | 1 | 1 | 1 | 1 | DRV3_SD1_ | 1 | 1 | 1 | DRV3_TX1 | 1 | 1 | 1 | DRV3_HTX0 | 1 | 1 | 1 |
| | | | | | | DAT2 | | | | | | | | | | | |

| | | Initial value | | | | Initial value | | | | Initial value | | | | Initial value | | | | | | | |
|------------------|-------------------|---------------|-------------|-------------|-----------------|---------------|--------|-------------|-------------|---------------|-----------|--------|-------------|---------------|--------|-----------|--------|-------------|-------------|--------|--|
| | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL17 | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL18 | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL19 | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | |
| DRVCTRL16 | | | | | | | | | | | | | | | | | | | | | |
| | DAT3 | | | | CD | | | | | | | | | | | | | | | | |
| bit21 | DRV2_SD3_ DAT3 | 1 | 1 | 1 | DRV2_SD1_ CD | 1 | 1 | 1 | DRV2_TX1 | 1 | 1 | 1 | DRV2_HTX0 | 1 | 1 | 1 | | | | | |
| bit20 | DRV1_SD3_ DAT3 | 1 | 1 | 1 | DRV1_SD1_ CD | 1 | 1 | 1 | DRV1_TX1 | 1 | 1 | 1 | DRV1_HTX0 | 1 | 1 | 1 | | | | | |
| bit19 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | | | | |
| bit18 | DRV3_SD3_ DAT4 | 1 | 1 | 1 | DRV3_SD1_ WP | 1 | 1 | 1 | DRV3_CTS1# | 1 | 1 | 1 | DRV3_HCTS0# | 1 | 1 | 1 | | | | | |
| bit17 | DRV2_SD3_ DAT4 | 1 | 1 | 1 | DRV2_SD1_ WP | 1 | 1 | 1 | DRV2_CTS1# | 1 | 1 | 1 | DRV2_HCTS0# | 1 | 1 | 1 | | | | | |

| DRVCTRL16 | | Initial value | | | | Initial value | | | | Initial value | | | | Initial value | | | | |
|--------------|---------------|---------------|-------------|-------------|-------------|---------------|--------|-------------|-------------|---------------|-----------|--------|------------------|---------------|--------|-----------|--------|-------------|
| | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL17 | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL18 | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL19 | RZ/G2H | RZ/G2M V1.3 |
| bit16 | DRV1_SD3_DAT4 | 1 | 1 | 1 | DRV1_SD1_WP | 1 | 1 | 1 | DRV1_CTS1# | 1 | 1 | 1 | DRV1_HCTS0# | 1 | 1 | 1 | | |
| bit15 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | |
| bit14 | DRV3_SD3_DAT5 | 1 | 1 | 1 | DRV3_SCK0 | 1 | 1 | 1 | DRV3_RTS1# | 1 | 1 | 1 | DRV3_HRTS0# | 1 | 1 | 1 | | |
| bit13 | DRV2_SD3_DAT5 | 1 | 1 | 1 | DRV2_SCK0 | 1 | 1 | 1 | DRV2_RTS1# | 1 | 1 | 1 | DRV2_HRTS0# | 1 | 1 | 1 | | |
| bit12 | DRV1_SD3_DAT5 | 1 | 1 | 1 | DRV1_SCK0 | 1 | 1 | 1 | DRV1_RTS1# | 1 | 1 | 1 | DRV1_HRTS0# | 1 | 1 | 1 | | |
| bit11 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | |
| bit10 | DRV3_SD3_DAT6 | 1 | 1 | 1 | DRV3_RX0 | 1 | 1 | 1 | DRV3_SCK2 | 1 | 1 | 1 | DRV3_MSIOF0_SCK | 1 | 1 | 1 | | |
| bit9 | DRV2_SD3_DAT6 | 1 | 1 | 1 | DRV2_RX0 | 1 | 1 | 1 | DRV2_SCK2 | 1 | 1 | 1 | DRV2_MSIOF0_SCK | 1 | 1 | 1 | | |
| bit8 | DRV1_SD3_DAT6 | 1 | 1 | 1 | DRV1_RX0 | 1 | 1 | 1 | DRV1_SCK2 | 1 | 1 | 1 | DRV1_MSIOF0_SCK | 1 | 1 | 1 | | |
| bit7 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | |
| bit6 | DRV3_SD3_DAT7 | 1 | 1 | 1 | DRV3_TX0 | 1 | 1 | 1 | DRV3_TX2 | 1 | 1 | 1 | DRV3_MSIOF0_SYNC | 1 | 1 | 1 | | |
| bit5 | DRV2_SD3_DAT7 | 1 | 1 | 1 | DRV2_TX0 | 1 | 1 | 1 | DRV2_TX2 | 1 | 1 | 1 | DRV2_MSIOF0_SYNC | 1 | 1 | 1 | | |
| bit4 | DRV1_SD3_DAT7 | 1 | 1 | 1 | DRV1_TX0 | 1 | 1 | 1 | DRV1_TX2 | 1 | 1 | 1 | DRV1_MSIOF0_SYNC | 1 | 1 | 1 | | |
| bit3 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | | |
| bit2 | DRV3_SD3_DS | 1 | 1 | 1 | DRV3_CTS0# | 1 | 1 | 1 | DRV3_RX2 | 1 | 1 | 1 | DRV3_MSIOF0_SS1 | 1 | 1 | 1 | | |
| bit1 | DRV2_SD3_DS | 1 | 1 | 1 | DRV2_CTS0# | 1 | 1 | 1 | DRV2_RX2 | 1 | 1 | 1 | DRV2_MSIOF0_SS1 | 1 | 1 | 1 | | |
| bit0 | DRV1_SD3_DS | 1 | 1 | 1 | DRV1_CTS0# | 1 | 1 | 1 | DRV1_RX2 | 1 | 1 | 1 | DRV1_MSIOF0_SS1 | 1 | 1 | 1 | | |

Table 8.11 Configuration of Registers in DRVCTRL20-24

| bit | DRVCTRL20 | Initial value | | | DRVCTRL21 | Initial value | | | DRVCTRL22 | Initial value | | | DRVCTRL23 | Initial value | | | DRVCTRL24 | Initial value | | |
|-------|-------------------|---------------|-------------|-------------|-------------------|---------------|--------|-------------|------------------|---------------|--------|--------|------------------|---------------|-------------|--------|-------------------|---------------|-------------|-------------|
| | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | | RZ/G2N | RZ/G2H | RZ/G2M V1.3 | | RZ/G2M V3.0 | RZ/G2N | RZ/G2H | | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 |
| bit31 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit30 | DRV3_ MSIOF0_ TXD | 1 | 1 | 1 | DRV3_SSI_ WS01239 | 1 | 1 | 1 | DRV3_SSI_ WS4 | 1 | 1 | 1 | DRV3_SSI_ SCK78 | 1 | 1 | 1 | DRV3_ USB0_OVC | 1 | 1 | 1 |
| bit29 | DRV2_ MSIOF0_ TXD | 1 | 1 | 1 | DRV2_SSI_ WS01239 | 1 | 1 | 1 | DRV2_SSI_ WS4 | 1 | 1 | 1 | DRV2_SSI_ SCK78 | 1 | 1 | 1 | DRV2_ USB0_OVC | 1 | 1 | 1 |
| bit28 | DRV1_ MSIOF0_ TXD | 1 | 1 | 1 | DRV1_SSI_ WS01239 | 1 | 1 | 1 | DRV1_SSI_ WS4 | 1 | 1 | 1 | DRV1_SSI_ SCK78 | 1 | 1 | 1 | DRV1_ USB0_OVC | 1 | 1 | 1 |
| bit27 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit26 | DRV3_ MSIOF0_ SS2 | 1 | 1 | 1 | DRV3_SSI_ SDATA0 | 1 | 1 | 1 | DRV3_SSI_ SDATA4 | 1 | 1 | 1 | DRV3_SSI_ WS78 | 1 | 1 | 1 | DRV3_ USB1_ PWEN | 1 | 1 | 1 |
| bit25 | DRV2_ MSIOF0_ SS2 | 1 | 1 | 1 | DRV2_SSI_ SDATA0 | 1 | 1 | 1 | DRV2_SSI_ SDATA4 | 1 | 1 | 1 | DRV2_SSI_ WS78 | 1 | 1 | 1 | DRV2_ USB1_ PWEN | 1 | 1 | 1 |
| bit24 | DRV1_ MSIOF0_ SS2 | 1 | 1 | 1 | DRV1_SSI_ SDATA0 | 1 | 1 | 1 | DRV1_SSI_ SDATA4 | 1 | 1 | 1 | DRV1_SSI_ WS78 | 1 | 1 | 1 | DRV1_ USB1_ PWEN | 1 | 1 | 1 |
| bit23 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit22 | DRV3_ MSIOF0_ RXD | 1 | 1 | 1 | DRV3_SSI_ SDATA1 | 1 | 1 | 1 | DRV3_SSI_ SCK5 | 1 | 1 | 1 | DRV3_SSI_ SDATA7 | 1 | 1 | 1 | DRV3_ USB1_OVC | 1 | 1 | 1 |
| bit21 | DRV2_ MSIOF0_ RXD | 1 | 1 | 1 | DRV2_SSI_ SDATA1 | 1 | 1 | 1 | DRV2_SSI_ SCK5 | 1 | 1 | 1 | DRV2_SSI_ SDATA7 | 1 | 1 | 1 | DRV2_ USB1_OVC | 1 | 1 | 1 |
| bit20 | DRV1_ MSIOF0_ RXD | 1 | 1 | 1 | DRV1_SSI_ SDATA1 | 1 | 1 | 1 | DRV1_SSI_ SCK5 | 1 | 1 | 1 | DRV1_SSI_ SDATA7 | 1 | 1 | 1 | DRV1_ USB1_OVC | 1 | 1 | 1 |
| bit19 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit18 | DRV3_ MLB_CLK | 1 | 1 | 1 | DRV3_SSI_ SDATA2 | 1 | 1 | 1 | DRV3_SSI_ WS5 | 1 | 1 | 1 | DRV3_SSI_ SDATA8 | 1 | 1 | 1 | DRV3_ USB30_ PWEN | 1 | 1 | 1 |
| bit17 | DRV2_ MLB_CLK | 1 | 1 | 1 | DRV2_SSI_ SDATA2 | 1 | 1 | 1 | DRV2_SSI_ WS5 | 1 | 1 | 1 | DRV2_SSI_ SDATA8 | 1 | 1 | 1 | DRV2_ USB30_ PWEN | 1 | 1 | 1 |
| bit16 | DRV1_ MLB_CLK | 1 | 1 | 1 | DRV1_SSI_ SDATA2 | 1 | 1 | 1 | DRV1_SSI_ WS5 | 1 | 1 | 1 | DRV1_SSI_ SDATA8 | 1 | 1 | 1 | DRV1_ USB30_ PWEN | 1 | 1 | 1 |
| bit15 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit14 | DRV3_ MLB_SIG | 1 | 1 | 1 | DRV3_SSI_ SCK349 | 1 | 1 | 1 | DRV3_SSI_ SDATA5 | 1 | 1 | 1 | DRV3_SSI_ SDATA9 | 1 | 1 | 1 | DRV3_ USB30_ OVC | 1 | 1 | 1 |

| | | Initial value | | | Initial value | | | Initial value | | | Initial value | | | Initial value | | | | | | |
|--------------|--------------|---------------|-------------|-------------|-----------------|-------------|-------------|---------------|-----------------|-------------|---------------|-------------|-----------------|---------------|-------------|-------------|--|---|---|---|
| DRVCTRL20 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | | | | |
| bit13 | DRV2_MLB_SIG | 1 | 1 | 1 | DRV2_SSI_SCK349 | 1 | 1 | 1 | DRV2_SSI_SDATA5 | 1 | 1 | 1 | DRV2_SSI_SDATA9 | 1 | 1 | 1 | DRV2_USB30_OVC | 1 | 1 | 1 |
| bit12 | DRV1_MLB_SIG | 1 | 1 | 1 | DRV1_SSI_SCK349 | 1 | 1 | 1 | DRV1_SSI_SDATA5 | 1 | 1 | 1 | DRV1_SSI_SDATA9 | 1 | 1 | 1 | DRV1_USB30_OVC | 1 | 1 | 1 |
| bit11 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit10 | DRV3_MLB_DAT | 1 | 1 | 1 | DRV3_SSI_WS349 | 1 | 1 | 1 | DRV3_SSI_SCK6 | 1 | 1 | 1 | DRV3_AUDIO_CLKA | 1 | 1 | 1 | [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] [RZ/G2H] DRV3_GP6_30* | 1 | 1 | 1 |
| bit9 | DRV2_MLB_DAT | 1 | 1 | 1 | DRV2_SSI_WS349 | 1 | 1 | 1 | DRV2_SSI_SCK6 | 1 | 1 | 1 | DRV2_AUDIO_CLKA | 1 | 1 | 1 | [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] [RZ/G2H] DRV2_GP6_30* | 1 | 1 | 1 |
| bit8 | DRV1_MLB_DAT | 1 | 1 | 1 | DRV1_SSI_WS349 | 1 | 1 | 1 | DRV1_SSI_SCK6 | 1 | 1 | 1 | DRV1_AUDIO_CLKA | 1 | 1 | 1 | [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] [RZ/G2H] DRV1_GP6_30* | 1 | 1 | 1 |
| bit7 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit6 | DRV3_MLB_REF | 1 | 1 | 1 | DRV3_SSI_SDATA3 | 1 | 1 | 1 | DRV3_SSI_WS6 | 1 | 1 | 1 | DRV3_AUDIO_CLKB | 1 | 1 | 1 | [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] [RZ/G2H] DRV3_GP6_31* | 1 | 1 | 1 |

| | | Initial value | | | Initial value | | | | Initial value | | | | Initial value | | | | | | | | |
|-------------|-------------------|---------------|-------------|-------------|---------------|-----------|-------------|-------------|---------------|-----------|-------------|-------------|---------------|-----------|-------------|-------------|--------|--|-------------|-------------|--------|
| DRVCTRL20 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | DRVCTRL21 | | | | DRVCTRL22 | | | | DRVCTRL23 | | | | DRVCTRL24 | | | |
| | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N |
| bit5 | DRV2_MLB_REF | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] [RZ/G2H] DRV2_GP6_31* | 1 | 1 | 1 |
| bit4 | DRV1_MLB_REF | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] [RZ/G2H] DRV1_GP6_31* | 1 | 1 | 1 |
| bit3 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | — | 1 | 1 | 1 |
| bit2 | DRV3_SSI_SCK01239 | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | — | 1 | 1 | 1 |
| bit1 | DRV2_SSI_SCK01239 | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | — | 1 | 1 | 1 |
| bit0 | DRV1_SSI_SCK01239 | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | 1 | 1 | 1 | — | — | 1 | 1 | 1 |

Note: * Different for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N

8.2.5 POC Control Register 0 (POCCTRL0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: Each bit in POCCTRL0 must be set according to IO voltage level that is supplied to the pin.

0: 1.8v 1: 3.3v

| Supply voltage | Setting register | Usage |
|----------------|------------------|---------------------|
| 1.8v | 0 | Possible |
| 1.8v | 1 | Do not use |
| 3.3v | 0 | Do not set (broken) |
| 3.3v | 1 | Possible |

| | | | | | | | | | | | | | | | | |
|------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | POC0 [31] | POC0 [30] | POC0 [29] | POC0 [28] | POC0 [27] | POC0 [26] | POC0 [25] | POC0 [24] | POC0 [23] | POC0 [22] | POC0 [21] | POC0 [20] | POC0 [19] | POC0 [18] | POC0 [17] | POC0 [16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | POC0 [15] | POC0 [14] | POC0 [13] | POC0 [12] | POC0 [11] | POC0 [10] | POC0 [9] | POC0 [8] | POC0 [7] | POC0 [6] | POC0 [5] | POC0 [4] | POC0 [3] | POC0 [2] | POC0 [1] | POC0 [0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: To enable this register to be set, appropriately set the multiplexed pin setting mask register (PMMR) immediately before setting this register.

Note: The power domain is common for SD channels.

When you use an unused port as GPIO, you must be careful about the voltage.

Table 8.12 Configuration of Registers in POCCTRL0

| | POCCTRL0 | Initial value |
|--------------|--------------|---------------|
| bit31 | — | 0 |
| bit30 | — | 0 |
| bit29 | POC_SD3_DS | 1 |
| bit28 | POC_SD3_DAT7 | 1 |
| bit27 | POC_SD3_DAT6 | 1 |
| bit26 | POC_SD3_DAT5 | 1 |
| bit25 | POC_SD3_DAT4 | 1 |
| bit24 | POC_SD3_DAT3 | 1 |
| bit23 | POC_SD3_DAT2 | 1 |
| bit22 | POC_SD3_DAT1 | 1 |
| bit21 | POC_SD3_DAT0 | 1 |
| bit20 | POC_SD3_CMD | 1 |
| bit19 | POC_SD3_CLK | 1 |
| bit18 | POC_SD2_DS | 1 |
| bit17 | POC_SD2_DAT3 | 1 |
| bit16 | POC_SD2_DAT2 | 1 |
| bit15 | POC_SD2_DAT1 | 1 |
| bit14 | POC_SD2_DAT0 | 1 |

| | POCTRL0 | Initial value |
|--------------|----------------|--------------------------|
| bit13 | POC_SD2_CMD | 1 |
| bit12 | POC_SD2_CLK | 1 |
| bit11 | POC_SD1_DAT3 | 1 |
| bit10 | POC_SD1_DAT2 | 1 |
| bit9 | POC_SD1_DAT1 | 1 |
| bit8 | POC_SD1_DAT0 | 1 |
| bit7 | POC_SD1_CMD | 1 |
| bit6 | POC_SD1_CLK | 1 |
| bit5 | POC_SD0_DAT3 | 1 |
| bit4 | POC_SD0_DAT2 | 1 |
| bit3 | POC_SD0_DAT1 | 1 |
| bit2 | POC_SD0_DAT0 | 1 |
| bit1 | POC_SD0_CMD | 1 |
| bit0 | POC_SD0_CLK | 1 |

8.2.6 TDSEL Control Register 0 (TDSELCTRL0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: TDSELCTRL0 controls the driving abilities of pins in use for the SDHI.

This function is delay adjustment of the SDHI clock return path for the LSI inside.

| | | TDOUT | |
|--------|--------|-------------------------|---|
| TDSEL1 | TDSEL0 | Target delay | |
| | | L | L |
| H | L | 20pF (@1.8 V operation) | |
| L | H | 30pF (@1.8 V operation) | |
| H | H | 40pF (@1.8 V operation) | |

The value of these bits must be 00.

| | | | | | | | | | | | | | | | | |
|------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TD SEL0 [31] | TD SEL0 [30] | TD SEL0 [29] | TD SEL0 [28] | TD SEL0 [27] | TD SEL0 [26] | TD SEL0 [25] | TD SEL0 [24] | TD SEL0 [23] | TD SEL0 [22] | TD SEL0 [21] | TD SEL0 [20] | TD SEL0 [19] | TD SEL0 [18] | TD SEL0 [17] | TD SEL0 [16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TD SEL0 [15] | TD SEL0 [14] | TD SEL0 [13] | TD SEL0 [12] | TD SEL0 [11] | TD SEL0 [10] | TD SEL0 [9] | TD SEL0 [8] | TD SEL0 [7] | TD SEL0 [6] | TD SEL0 [5] | TD SEL0 [4] | TD SEL0 [3] | TD SEL0 [2] | TD SEL0 [1] | TD SEL0 [0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: To enable this register to be set, appropriately set the multiplexed pin setting mask register (PMMR) immediately before setting this register.

Table 8.13 Configuration of Registers in TDSELCTRL0

| | TDSELCTRL0 | Initial value |
|-------|------------|---------------|
| bit31 | — | 0 |
| bit30 | — | 0 |
| bit29 | — | 0 |
| bit28 | — | 0 |
| bit27 | — | 0 |
| bit26 | — | 0 |
| bit25 | — | 0 |
| bit24 | — | 0 |
| bit23 | — | 0 |
| bit22 | — | 0 |
| bit21 | — | 0 |
| bit20 | — | 0 |
| bit19 | — | 0 |
| bit18 | — | 0 |
| bit17 | — | 0 |
| bit16 | — | 0 |

| | TDSELCTRL0 | Initial value |
|--------------|-------------------|----------------------|
| bit15 | — | 0 |
| bit14 | — | 0 |
| bit13 | — | 0 |
| bit12 | — | 0 |
| bit11 | — | 0 |
| bit10 | — | 0 |
| bit9 | — | 0 |
| bit8 | — | 0 |
| bit7 | SD3CLK_TDSEL1 | 0 |
| bit6 | SD3CLK_TDSEL0 | 0 |
| bit5 | SD2CLK_TDSEL1 | 0 |
| bit4 | SD2CLK_TDSEL0 | 0 |
| bit3 | SD1CLK_TDSEL1 | 0 |
| bit2 | SD1CLK_TDSEL0 | 0 |
| bit1 | SD0CLK_TDSEL1 | 0 |
| bit0 | SD0CLK_TDSEL0 | 0 |

8.2.7 LSI pin pull-enable register 0-6 (PUEN0-6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: PUEN0-6 performs on/off control of the pull resistors.

0: Pull-up/down function is disabled.

1: Pull-up/down function is enabled.

| | | | | | | | | | | | | | | | | |
|------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | PUE Nn* [31] | PUE Nn* [30] | PUE Nn* [29] | PUE Nn* [28] | PUE Nn* [27] | PUE Nn* [26] | PUE Nn* [25] | PUE Nn* [24] | PUE Nn* [23] | PUE Nn* [22] | PUE Nn* [21] | PUE Nn* [20] | PUE Nn* [19] | PUE Nn* [18] | PUE Nn* [17] | PUE Nn* [16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PUE Nn* [15] | PUE Nn* [14] | PUE Nn* [13] | PUE Nn* [12] | PUE Nn* [11] | PUE Nn* [10] | PUE Nn* [9] | PUE Nn* [8] | PUE Nn* [7] | PUE Nn* [6] | PUE Nn* [5] | PUE Nn* [4] | PUE Nn* [3] | PUE Nn* [2] | PUE Nn* [1] | PUE Nn* [0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * n = 0-6

When the I2C1, I2C2, I2C4 and I2C6 pins are selected, the pull-up/down function of these pins is disabled respectively even if the pull-up/down function is set to enable.

Table 8.14 Configuration of Registers in PUEN0-3

| PUEN0 | Initial value | | | | PUEN1 | Initial value | | | | PUEN2 | Initial value | | | | PUEN3 | Initial value | | | |
|--------------------------|---------------|-------------|-------------|--------|-----------|---------------|-------------|-------------|-------------------|-------|---------------|-------------|---------------|--------|-------|---------------|-------------|-------------|--------|
| | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N |
| bit31 PUEN_AVB_PHY_INT | 1 | 1 | 1 | | PUEN_A19 | 0 | 0 | 0 | PUEN_DU_DOTCLKIN1 | 1 | 1 | 1 | PUEN_SD3_DAT0 | 0 | 0 | 0 | | | |
| bit30 PUEN_AVB_MAGIC | 1 | 1 | 1 | | PUEN_A18 | 0 | 0 | 0 | PUEN_DU_DOTCLKIN0 | 1 | 1 | 1 | PUEN_SD3_CMD | 0 | 0 | 0 | | | |
| bit29 PUEN_AVB_MDC | 0 | 0 | 0 | | PUEN_A17 | 0 | 0 | 0 | PUEN_GP7_03 | 1 | 1 | 1 | PUEN_SD3_CLK | 0 | 0 | 0 | | | |
| bit28 PUEN_AVB_MDIO | 0 | 0 | 0 | | PUEN_A16 | 0 | 0 | 0 | PUEN_GP7_02 | 1 | 1 | 1 | PUEN_SD2_DS | 0 | 0 | 0 | | | |
| bit27 PUEN_AVB_TXCREFLCK | 1 | 1 | 1 | | PUEN_A15 | 0 | 0 | 0 | PUEN_AVS2 | 1 | 1 | 1 | PUEN_SD2_DAT3 | 0 | 0 | 0 | | | |
| bit26 PUEN_AVB_TD3 | 1 | 1 | 1 | | PUEN_A14 | 0 | 0 | 0 | PUEN_AVS1 | 1 | 1 | 1 | PUEN_SD2_DAT2 | 0 | 0 | 0 | | | |
| bit25 PUEN_AVB_TD2 | 1 | 1 | 1 | | PUEN_A13 | 0 | 0 | 0 | PUEN_D15 | 1 | 1 | 1 | PUEN_SD2_DAT1 | 0 | 0 | 0 | | | |
| bit24 PUEN_AVB_TD1 | 1 | 1 | 1 | | PUEN_A12 | 0 | 0 | 0 | PUEN_D14 | 1 | 1 | 1 | PUEN_SD2_DAT0 | 0 | 0 | 0 | | | |
| bit23 PUEN_AVB_TD0 | 1 | 1 | 1 | | PUEN_A11 | 1 | 1 | 1 | PUEN_D13 | 0 | 0 | 0 | PUEN_SD2_CMD | 0 | 0 | 0 | | | |
| bit22 PUEN_AVB_TXC | 1 | 1 | 1 | | PUEN_A10 | 0 | 0 | 0 | PUEN_D12 | 0 | 0 | 0 | PUEN_SD2_CLK | 0 | 0 | 0 | | | |
| bit21 PUEN_AVB_TX_CTL | 1 | 1 | 1 | | PUEN_A9 | 1 | 1 | 1 | PUEN_D11 | 0 | 0 | 0 | PUEN_SD1_DAT3 | 0 | 0 | 0 | | | |
| bit20 PUEN_AVB_RD3 | 1 | 1 | 1 | | PUEN_A8 | 1 | 1 | 1 | PUEN_D10 | 0 | 0 | 0 | PUEN_SD1_DAT2 | 0 | 0 | 0 | | | |
| bit19 PUEN_AVB_RD2 | 1 | 1 | 1 | | PUEN_A7 | 0 | 0 | 0 | PUEN_D9 | 0 | 0 | 0 | PUEN_SD1_DAT1 | 0 | 0 | 0 | | | |
| bit18 PUEN_AVB_RD1 | 1 | 1 | 1 | | PUEN_A6 | 0 | 0 | 0 | PUEN_D8 | 0 | 0 | 0 | PUEN_SD1_DAT0 | 0 | 0 | 0 | | | |
| bit17 PUEN_AVB_RD0 | 1 | 1 | 1 | | PUEN_A5 | 1 | 1 | 1 | PUEN_D7 | 0 | 0 | 0 | PUEN_SD1_CMD | 0 | 0 | 0 | | | |
| bit16 PUEN_AVB_RXC | 1 | 1 | 1 | | PUEN_A4 | 0 | 0 | 0 | PUEN_D6 | 0 | 0 | 0 | PUEN_SD1_CLK | 0 | 0 | 0 | | | |
| bit15 PUEN_AVB_RX_CTL | 1 | 1 | 1 | | PUEN_A3 | 0 | 0 | 0 | PUEN_D5 | 0 | 0 | 0 | PUEN_SD0_DAT3 | 0 | 0 | 0 | | | |
| bit14 PUEN_RPC_RESET# | 0 | 0 | 0 | | PUEN_A2 | 1 | 1 | 1 | PUEN_D4 | 0 | 0 | 0 | PUEN_SD0_DAT2 | 0 | 0 | 0 | | | |
| bit13 PUEN_RPC_WP# | 0 | 0 | 0 | | PUEN_A1 | 0 | 0 | 0 | PUEN_D3 | 0 | 0 | 0 | PUEN_SD0_DAT1 | 0 | 0 | 0 | | | |
| bit12 PUEN_RPC_INT# | 1 | 1 | 1 | | PUEN_A0 | 0 | 0 | 0 | PUEN_D2 | 0 | 0 | 0 | PUEN_SD0_DAT0 | 0 | 0 | 0 | | | |
| bit11 PUEN_QSPI1_SSL | 0 | 0 | 0 | | PUEN_PWM2 | 1 | 1 | 1 | PUEN_D1 | 0 | 0 | 0 | PUEN_SD0_CMD | 0 | 0 | 0 | | | |

| PUEN0 | Initial value | | | PUEN1 | Initial value | | | PUEN2 | Initial value | | | PUEN3 | Initial value | | |
|---------------------------------|---------------|-------------|-------------|-----------------------|---------------|-------------|-------------|-----------------|---------------|-------------|-------------|--|---------------|-------------|-------------|
| | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 |
| bit10 PUEN_QSPI1_IO3 | 0 | 0 | 0 | PUEN_PWM1 | 1 | 1 | 1 | PUEN_D0 | 1 | 1 | 1 | PUEN_SD0_CLK | 0 | 0 | 0 |
| bit9 PUEN_QSPI1_IO2 | 0 | 0 | 0 | PUEN_PWM0 | 1 | 1 | 1 | PUEN_PRESETOUT# | 0 | 0 | 0 | PUEN_ASEBRK | 1 | 1 | 1 |
| bit8 PUEN_QSPI1_MISO/IO1 | 0 | 0 | 0 | PUEN_IRQ5 | 1 | 1 | 1 | PUEN_EX_WAIT0 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit7 PUEN_QSPI1_MOSI/IO0 | 0 | 0 | 0 | PUEN_IRQ4 | 1 | 1 | 1 | PUEN_WE1# | 1 | 1 | 1 | PUEN_TDI | 1 | 1 | 1 |
| bit6 PUEN_QSPI1_SPCLK | 0 | 0 | 0 | PUEN_IRQ3 | 1 | 1 | 1 | PUEN_WE0# | 1 | 1 | 1 | PUEN_TMS | 1 | 1 | 1 |
| bit5 PUEN_QSPI0_SSL | 0 | 0 | 0 | PUEN_IRQ2 | 1 | 1 | 1 | PUEN_RD/WR# | 1 | 1 | 1 | PUEN_TCK | 1 | 1 | 1 |
| bit4 PUEN_QSPI0_IO3 | 0 | 0 | 0 | PUEN_IRQ1 | 1 | 1 | 1 | PUEN_RD# | 0 | 0 | 0 | PUEN_TRST# | 1 | 1 | 1 |
| bit3 PUEN_QSPI0_IO2 | 0 | 0 | 0 | PUEN_IRQ0 | 1 | 1 | 1 | PUEN_BS# | 0 | 0 | 0 | PUEN_EXTALR | 0 | 0 | 0 |
| bit2 PUEN_QSPI0_MISO/IO1 | 0 | 0 | 0 | PUEN_AVB_AVTP_CAPTURE | 1 | 1 | 1 | PUEN_CS1# | 1 | 1 | 1 | — | 1 | 1 | 1 |
| bit1 PUEN_QSPI0_MOSI/IO0 | 0 | 0 | 0 | PUEN_AVB_AVTP_MATCH | 1 | 1 | 1 | PUEN_CS0# | 1 | 1 | 1 | [RZ/G2H] [RZ/G2N] PUEN_DU_DOT CLKIN3 [RZ/G2M V1.3] [RZ/G2M V3.0] — | 1 | 0 | 0 |
| bit0 PUEN_QSPI0_SPCLK | 0 | 0 | 0 | PUEN_AVB_LINK | 1 | 1 | 1 | PUEN_CLKOUT | 0 | 0 | 0 | [RZ/G2M V1.3] [RZ/G2M V3.0] PUEN_DU_DOT CLKIN2 [RZ/G2H] [RZ/G2N] — | 1 | 1 | 1 |

Table 8.15 Configuration of Registers in PUEN4-6

| | PUEN4 | Initial value | | | PUEN5 | Initial value | | | PUEN6 | Initial value | | |
|-------|------------------|---------------|-------------|--------|-------------------|---------------|-------------|--------|---------------|---------------|-------------|--------|
| | | RZ/G2M V1.3 | | | | RZ/G2M V1.3 | | | | RZ/G2M V1.3 | | |
| | | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V3.0 | RZ/G2N |
| bit31 | PUEN_MSIOF0_SS1 | 1 | 1 | 1 | PUEN_USB0_PWEN | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit30 | PUEN_MSIOF0_SYNC | 1 | 1 | 1 | PUEN_AUDIO_CLKB | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit29 | PUEN_MSIOF0_SCK | 1 | 1 | 1 | PUEN_AUDIO_CLKA | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit28 | PUEN_HRTS0# | 1 | 1 | 1 | PUEN_SSI_SDATA9 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit27 | PUEN_HCTS0# | 1 | 1 | 1 | PUEN_SSI_SDATA8 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit26 | PUEN_HTX0 | 1 | 1 | 1 | PUEN_SSI_SDATA7 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit25 | PUEN_HRX0 | 1 | 1 | 1 | PUEN_SSI_WS78 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit24 | PUEN_HSCK0 | 1 | 1 | 1 | PUEN_SSI_SCK78 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit23 | PUEN_RX2 | 1 | 1 | 1 | PUEN_SSI_SDATA6 | 0 | 0 | 0 | — | 0 | 0 | 0 |
| bit22 | PUEN_TX2 | 1 | 1 | 1 | PUEN_SSI_WS6 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit21 | PUEN_SCK2 | 1 | 1 | 1 | PUEN_SSI_SCK6 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit20 | PUEN_RTS1# | 1 | 1 | 1 | PUEN_SSI_SDATA5 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit19 | PUEN_CTS1# | 1 | 1 | 1 | PUEN_SSI_WS5 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit18 | PUEN_TX1 | 1 | 1 | 1 | PUEN_SSI_SCK5 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit17 | PUEN_RX1 | 1 | 1 | 1 | PUEN_SSI_SDATA4 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit16 | PUEN_RTS0# | 1 | 1 | 1 | PUEN_SSI_WS4 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit15 | PUEN_CTS0# | 1 | 1 | 1 | PUEN_SSI_SCK4 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit14 | PUEN_TX0 | 1 | 1 | 1 | PUEN_SSI_SDATA3 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit13 | PUEN_RX0 | 1 | 1 | 1 | PUEN_SSI_WS349 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit12 | PUEN_SCK0 | 1 | 1 | 1 | PUEN_SSI_SCK349 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit11 | PUEN_SD1_WP | 1 | 1 | 1 | PUEN_SSI_SDATA2 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit10 | PUEN_SD1_CD | 1 | 1 | 1 | PUEN_SSI_SDATA1 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit9 | PUEN_SD0_WP | 1 | 1 | 1 | PUEN_SSI_SDATA0 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit8 | PUEN_SD0_CD | 1 | 1 | 1 | PUEN_SSI_WS01239 | 1 | 1 | 1 | — | 0 | 0 | 0 |
| bit7 | PUEN_SD3_DS | 0 | 0 | 0 | PUEN_SSI_SCK01239 | 1 | 1 | 1 | PRESET# *2 *3 | 0 | 1 | 0 |

| | PUEN4 | Initial value | | | PUEN5 | Initial value | | | PUEN6 | Initial value | | |
|-------------|---------------|---------------|-------------|--------|-----------------|---------------|-------------|--------|---|---------------|-------------|--------|
| | | RZ/G2M V1.3 | | | | RZ/G2M V1.3 | | | | RZ/G2M V1.3 | | |
| | | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V3.0 | RZ/G2N |
| bit6 | PUEN_SD3_DAT7 | 0 | 0 | 0 | PUEN_MLB_REF | 0 | 0 | 0 | [RZ/G2H] [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] PUEN_GP6_31 *1 | 1 | 1 | 1 |
| bit5 | PUEN_SD3_DAT6 | 0 | 0 | 0 | PUEN_MLB_DAT | 0 | 0 | 0 | [RZ/G2H] [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] PUEN_GP6_30 *1 | 1 | 1 | 1 |
| bit4 | PUEN_SD3_DAT5 | 0 | 0 | 0 | PUEN_MLB_SIG | 0 | 0 | 0 | PUEN_USB30_OVC | 1 | 1 | 1 |
| bit3 | PUEN_SD3_DAT4 | 0 | 0 | 0 | PUEN_MLB_CLK | 0 | 0 | 0 | PUEN_USB30_PWEN | 1 | 1 | 1 |
| bit2 | PUEN_SD3_DAT3 | 0 | 0 | 0 | PUEN_MSIOF0_RXD | 1 | 1 | 1 | PUEN_USB1_OVC | 1 | 1 | 1 |
| bit1 | PUEN_SD3_DAT2 | 0 | 0 | 0 | PUEN_MSIOF0_SS2 | 1 | 1 | 1 | PUEN_USB1_PWEN | 1 | 1 | 1 |
| bit0 | PUEN_SD3_DAT1 | 0 | 0 | 0 | PUEN_MSIOF0_TXD | 1 | 1 | 1 | PUEN_USB0_OVC | 1 | 1 | 1 |

- Notes: 1. Different for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N
2. RZ/G2M V1.3, RZ/G2M V3.0 Only
3. Since there is no pull-up resistor to PRESET#, to control the pull-down in this bit7.

8.2.8 LSI pin pull-up/down control Register 0-6 (PUD0-6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: PUD0-6 performs pull-up/pull-down control of the pull resistors.

0: Pull-down is enabled.

1: Pull-up is enabled.

| | | | | | | | | | | | | | | | | |
|------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | PUDn* [31] | PUDn* [30] | PUDn* [29] | PUDn* [28] | PUDn* [27] | PUDn* [26] | PUDn* [25] | PUDn* [24] | PUDn* [23] | PUDn* [22] | PUDn* [21] | PUDn* [20] | PUDn* [19] | PUDn* [18] | PUDn* [17] | PUDn* [16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PUDn* [15] | PUDn* [14] | PUDn* [13] | PUDn* [12] | PUDn* [11] | PUDn* [10] | PUDn* [9] | PUDn* [8] | PUDn* [7] | PUDn* [6] | PUDn* [5] | PUDn* [4] | PUDn* [3] | PUDn* [2] | PUDn* [1] | PUDn* [0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * n = 0-6

Table 8.16 Configuration of Registers in PUD0-3

| PUD0 | Initial value | | | | PUD1 | Initial value | | | | PUD2 | Initial value | | | | PUD3 | Initial value | | | |
|--------------------------------|---------------|-------------|-------------|--------|---------|---------------|-------------|-------------|--------|------------------|---------------|-------------|-------------|--------|--------------|---------------|-------------|-------------|--------|
| | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N |
| bit31 PUD_AVB_PHY_INT | 0 | 0 | 0 | | PUD_A19 | 1 | 1 | 1 | | PUD_DU_DOTCLKIN1 | 0 | 0 | 0 | | PUD_SD3_DAT0 | 1 | 1 | 1 | |
| bit30 PUD_AVB_MAGIC | 0 | 0 | 0 | | PUD_A18 | 1 | 1 | 1 | | PUD_DU_DOTCLKIN0 | 0 | 0 | 0 | | PUD_SD3_CMD | 1 | 1 | 1 | |
| bit29 PUD_AVB_MDC | 1 | 1 | 1 | | PUD_A17 | 1 | 1 | 1 | | PUD_GP7_03 | 1 | 1 | 1 | | PUD_SD3_CLK | 1 | 1 | 1 | |
| bit28 PUD_AVB_MDIO | 1 | 1 | 1 | | PUD_A16 | 1 | 1 | 1 | | PUD_GP7_02 | 1 | 1 | 1 | | PUD_SD2_DS | 1 | 1 | 1 | |
| bit27 PUD_AVB_TXCREFLCK | 0 | 0 | 0 | | PUD_A15 | 1 | 1 | 1 | | PUD_AVS2 | 0 | 0 | 0 | | PUD_SD2_DAT3 | 1 | 1 | 1 | |
| bit26 PUD_AVB_TD3 | 0 | 0 | 0 | | PUD_A14 | 1 | 1 | 1 | | PUD_AVS1 | 0 | 0 | 0 | | PUD_SD2_DAT2 | 1 | 1 | 1 | |
| bit25 PUD_AVB_TD2 | 0 | 0 | 0 | | PUD_A13 | 1 | 1 | 1 | | PUD_D15 | 1 | 1 | 1 | | PUD_SD2_DAT1 | 1 | 1 | 1 | |
| bit24 PUD_AVB_TD1 | 0 | 0 | 0 | | PUD_A12 | 1 | 1 | 1 | | PUD_D14 | 1 | 1 | 1 | | PUD_SD2_DAT0 | 1 | 1 | 1 | |
| bit23 PUD_AVB_TD0 | 0 | 0 | 0 | | PUD_A11 | 0 | 0 | 0 | | PUD_D13 | 1 | 1 | 1 | | PUD_SD2_CMD | 1 | 1 | 1 | |
| bit22 PUD_AVB_TXC | 0 | 0 | 0 | | PUD_A10 | 1 | 1 | 1 | | PUD_D12 | 1 | 1 | 1 | | PUD_SD2_CLK | 1 | 1 | 1 | |
| bit21 PUD_AVB_TX_CTL | 0 | 0 | 0 | | PUD_A9 | 1 | 1 | 1 | | PUD_D11 | 1 | 1 | 1 | | PUD_SD1_DAT3 | 1 | 1 | 1 | |
| bit20 PUD_AVB_RD3 | 0 | 0 | 0 | | PUD_A8 | 1 | 1 | 1 | | PUD_D10 | 1 | 1 | 1 | | PUD_SD1_DAT2 | 1 | 1 | 1 | |

| PUD0 | | Initial value | | | | PUD1 | | Initial value | | | | PUD2 | | Initial value | | | | PUD3 | | Initial value | | | |
|-------|--------------------|---------------|-------------|-------------|--------|----------------------|---|---------------|-------------|----------------|--------|------|---|---|-------------|-------------|--------|------|--|---------------|-------------|-------------|--------|
| | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N |
| bit19 | PUD_AVB_RD2 | 0 | 0 | 0 | | PUD_A7 | 1 | 1 | 1 | PUD_D9 | 1 | 1 | 1 | PUD_SD1_DAT1 | 1 | 1 | 1 | | | | | | |
| bit18 | PUD_AVB_RD1 | 0 | 0 | 0 | | PUD_A6 | 1 | 1 | 1 | PUD_D8 | 1 | 1 | 1 | PUD_SD1_DAT0 | 1 | 1 | 1 | | | | | | |
| bit17 | PUD_AVB_RD0 | 0 | 0 | 0 | | PUD_A5 | 1 | 1 | 1 | PUD_D7 | 1 | 1 | 1 | PUD_SD1_CMD | 1 | 1 | 1 | | | | | | |
| bit16 | PUD_AVB_RXC | 0 | 0 | 0 | | PUD_A4 | 1 | 1 | 1 | PUD_D6 | 1 | 1 | 1 | PUD_SD1_CLK | 1 | 1 | 1 | | | | | | |
| bit15 | PUD_AVB_RX_CTL | 0 | 0 | 0 | | PUD_A3 | 1 | 1 | 1 | PUD_D5 | 1 | 1 | 1 | PUD_SD0_DAT3 | 1 | 1 | 1 | | | | | | |
| bit14 | PUD_RPC_RESET# | 1 | 1 | 1 | | PUD_A2 | 0 | 0 | 0 | PUD_D4 | 1 | 1 | 1 | PUD_SD0_DAT2 | 1 | 1 | 1 | | | | | | |
| bit13 | PUD_RPC_WP# | 1 | 1 | 1 | | PUD_A1 | 1 | 1 | 1 | PUD_D3 | 1 | 1 | 1 | PUD_SD0_DAT1 | 1 | 1 | 1 | | | | | | |
| bit12 | PUD_RPC_INT# | 1 | 1 | 1 | | PUD_A0 | 1 | 1 | 1 | PUD_D2 | 1 | 1 | 1 | PUD_SD0_DAT0 | 1 | 1 | 1 | | | | | | |
| bit11 | PUD_QSPI1_SSL | 1 | 1 | 1 | | PUD_PWM2 | 1 | 1 | 1 | PUD_D1 | 1 | 1 | 1 | PUD_SD0_CMD | 1 | 1 | 1 | | | | | | |
| bit10 | PUD_QSPI1_IO3 | 1 | 1 | 1 | | PUD_PWM1 | 1 | 1 | 1 | PUD_D0 | 0 | 0 | 0 | PUD_SD0_CLK | 1 | 1 | 1 | | | | | | |
| bit9 | PUD_QSPI1_IO2 | 1 | 1 | 1 | | PUD_PWM0 | 1 | 1 | 1 | PUD_PRESETOUT# | 1 | 1 | 1 | PUD_ASEBRK | 1 | 1 | 1 | | | | | | |
| bit8 | PUD_QSPI1_MISO/IO1 | 1 | 1 | 1 | | PUD_IRQ5 | 1 | 1 | 1 | PUD_EX_WAIT0 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | | | | |
| bit7 | PUD_QSPI1_MOSI/IO0 | 1 | 1 | 1 | | PUD_IRQ4 | 1 | 1 | 1 | PUD_WE1# | 1 | 1 | 1 | PUD_TDI | 1 | 1 | 1 | | | | | | |
| bit6 | PUD_QSPI1_SPCLK | 1 | 1 | 1 | | PUD_IRQ3 | 1 | 1 | 1 | PUD_WE0# | 1 | 1 | 1 | PUD_TMS | 1 | 1 | 1 | | | | | | |
| bit5 | PUD_QSPI0_SSL | 1 | 1 | 1 | | PUD_IRQ2 | 1 | 1 | 1 | PUD_RD/WR# | 1 | 1 | 1 | PUD_TCK | 1 | 1 | 1 | | | | | | |
| bit4 | PUD_QSPI0_IO3 | 1 | 1 | 1 | | PUD_IRQ1 | 1 | 1 | 1 | PUD_RD# | 1 | 1 | 1 | PUD_TRST# | 0 | 0 | 0 | | | | | | |
| bit3 | PUD_QSPI0_IO2 | 1 | 1 | 1 | | PUD_IRQ0 | 1 | 1 | 1 | PUD_BS# | 1 | 1 | 1 | PUD_EXTALR | 1 | 1 | 1 | | | | | | |
| bit2 | PUD_QSPI0_MISO/IO1 | 1 | 1 | 1 | | PUD_AVB_AVTP_CAPTURE | 1 | 1 | 1 | PUD_CS1# | 1 | 1 | 1 | — | 1 | 1 | 1 | | | | | | |
| bit1 | PUD_QSPI0_MOSI/IO0 | 1 | 1 | 1 | | PUD_AVB_AVTP_MATCH | 1 | 1 | 1 | PUD_CS0# | 1 | 1 | 1 | [RZ/G2H] [RZ/G2N] PUD_DU_DOTC LKIN3 [RZ/G2M V1.3] [RZ/G2M V3.0] — | 0 | 0 | 0 | | | | | | |

| | | Initial value | | | Initial value | | | Initial value | | | Initial value | | | | | | | | | | | | | |
|-------------|-----------------|---------------|-------------|-------------|---------------|--------------|-------------|---------------|-------------|-------------|---------------|-------------|-------------|---|------|---------------|---|---|---|---------------|-------------------|----------|----------|---|
| PUD0 | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | | | | | | | | | | | |
| bit0 | PUD_QSPI0_SPCLK | 1 | 1 | 1 | PUD1 | PUD_AVB_LINK | 0 | 0 | 0 | PUD2 | PUD_CLKOUT | 1 | 1 | 1 | PUD3 | [RZ/G2M V1.3] | 0 | 0 | 0 | [RZ/G2M V3.0] | PUD_DU_DOTC LKIN2 | [RZ/G2H] | [RZ/G2N] | — |

Table 8.17 Configuration of Registers in PUD4-6

| | PUD4 | Initial value | | | | PUD5 | Initial value | | | | PUD6 | Initial value | | | |
|-------|-----------------|---------------|-------------|-------------|------------------|------|---------------|-------------|--|--------|------|---------------|-------------|-------------|--------|
| | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N |
| bit31 | PUD_MSIOF0_SS1 | 1 | 1 | 1 | PUD_USB0_PWEN | 0 | 0 | 0 | — | 0 | 0 | 0 | | | |
| bit30 | PUD_MSIOF0_SYNC | 1 | 1 | 1 | PUD_AUDIO_CLKB | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit29 | PUD_MSIOF0_SCK | 1 | 1 | 1 | PUD_AUDIO_CLKA | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit28 | PUD_HRTS0# | 1 | 1 | 1 | PUD_SSI_SDATA9 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit27 | PUD_HCTS0# | 1 | 1 | 1 | PUD_SSI_SDATA8 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit26 | PUD_HTX0 | 1 | 1 | 1 | PUD_SSI_SDATA7 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit25 | PUD_HRX0 | 1 | 1 | 1 | PUD_SSI_WS78 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit24 | PUD_HSCK0 | 1 | 1 | 1 | PUD_SSI_SCK78 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit23 | PUD_RX2 | 1 | 1 | 1 | PUD_SSI_SDATA6 | 1 | 0 | 1 | — | 0 | 0 | 0 | | | |
| bit22 | PUD_TX2 | 1 | 1 | 1 | PUD_SSI_WS6 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit21 | PUD_SCK2 | 1 | 1 | 1 | PUD_SSI_SCK6 | 0 | 0 | 0 | — | 0 | 0 | 0 | | | |
| bit20 | PUD_RTS1# | 1 | 1 | 1 | PUD_SSI_SDATA5 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit19 | PUD_CTS1# | 1 | 1 | 1 | PUD_SSI_WS5 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit18 | PUD_TX1 | 1 | 1 | 1 | PUD_SSI_SCK5 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit17 | PUD_RX1 | 1 | 1 | 1 | PUD_SSI_SDATA4 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit16 | PUD_RTS0# | 1 | 1 | 1 | PUD_SSI_WS4 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit15 | PUD_CTS0# | 1 | 1 | 1 | PUD_SSI_SCK4 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit14 | PUD_TX0 | 1 | 1 | 1 | PUD_SSI_SDATA3 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit13 | PUD_RX0 | 1 | 1 | 1 | PUD_SSI_WS349 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit12 | PUD_SCK0 | 1 | 1 | 1 | PUD_SSI_SCK349 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit11 | PUD_SD1_WP | 1 | 1 | 1 | PUD_SSI_SDATA2 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit10 | PUD_SD1_CD | 1 | 1 | 1 | PUD_SSI_SDATA1 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit9 | PUD_SD0_WP | 1 | 1 | 1 | PUD_SSI_SDATA0 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit8 | PUD_SD0_CD | 1 | 1 | 1 | PUD_SSI_WS01239 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit7 | PUD_SD3_DS | 1 | 1 | 1 | PUD_SSI_SCK01239 | 1 | 1 | 1 | — | 0 | 0 | 0 | | | |
| bit6 | PUD_SD3_DAT7 | 1 | 1 | 1 | PUD_MLB_REF | 1 | 1 | 1 | [RZ/G2H][RZ/G2M V1.3] [RZ/G2M V3.0][RZ/G2N] PUD_GP6_31 * | 1 | 1 | 1 | | | |
| bit5 | PUD_SD3_DAT6 | 1 | 1 | 1 | PUD_MLB_DAT | 1 | 1 | 1 | [RZ/G2H][RZ/G2M V1.3] [RZ/G2M V3.0][RZ/G2N] PUD_GP6_30 * | 0 | 0 | 0 | | | |
| bit4 | PUD_SD3_DAT5 | 1 | 1 | 1 | PUD_MLB_SIG | 1 | 1 | 1 | PUD_USB30_OVC | 1 | 1 | 1 | | | |
| bit3 | PUD_SD3_DAT4 | 1 | 1 | 1 | PUD_MLB_CLK | 1 | 1 | 1 | PUD_USB30_PWEN | 0 | 0 | 0 | | | |
| bit2 | PUD_SD3_DAT3 | 1 | 1 | 1 | PUD_MSIOF0_RXD | 1 | 1 | 1 | PUD_USB1_OVC | 1 | 1 | 1 | | | |
| bit1 | PUD_SD3_DAT2 | 1 | 1 | 1 | PUD_MSIOF0_SS2 | 1 | 1 | 1 | PUD_USB1_PWEN | 0 | 0 | 0 | | | |
| bit0 | PUD_SD3_DAT1 | 1 | 1 | 1 | PUD_MSIOF0_TXD | 1 | 1 | 1 | PUD_USB0_OVC | 1 | 1 | 1 | | | |

Note: * Different for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N

8.2.9 Module Select Register 0-2 (MOD_SEL0-2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: MOD_SEL0-2 selects the group for multiple LSI pins with multiplexed pin functions.

Each input or input/output signal are assigned to two or more groups of pins. Select one of these groups when using these signals. Do not use the module pins in the unselected group; if a module pin in the unselected group is used, correct operation is not guaranteed. Group is “Set value” of Table 8.19.

For some modules, however, although the output signals are assigned to two or more groups of pins, there is no bit for selecting the group. Select one of these pins for each output signal through the corresponding peripheral function select register. Also note that each pin can only be used in combination with the other input or input/output pins of the same group. Correct operation is not guaranteed when a pin is used in combination with pins from other groups. If this is not the case, correct operation is not guaranteed.

| | | | | | | | | | | | | | | | | |
|------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MOD_SE Ln*[31] | MOD_SE Ln*[30] | MOD_SE Ln*[29] | MOD_SE Ln*[28] | MOD_SE Ln*[27] | MOD_SE Ln*[26] | MOD_SE Ln*[25] | MOD_SE Ln*[24] | MOD_SE Ln*[23] | MOD_SE Ln*[22] | MOD_SE Ln*[21] | MOD_SE Ln*[20] | MOD_SE Ln*[19] | MOD_SE Ln*[18] | MOD_SE Ln*[17] | MOD_SE Ln*[16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MOD_SE Ln*[15] | MOD_SE Ln*[14] | MOD_SE Ln*[13] | MOD_SE Ln*[12] | MOD_SE Ln*[11] | MOD_SE Ln*[10] | MOD_SE Ln*[9] | MOD_SE Ln*[8] | MOD_SE Ln*[7] | MOD_SE Ln*[6] | MOD_SE Ln*[5] | MOD_SE Ln*[4] | MOD_SE Ln*[3] | MOD_SE Ln*[2] | MOD_SE Ln*[1] | MOD_SE Ln*[0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * n = 0-2

Note: To enable this register to be set, appropriately set the multiplexed pin setting mask register (PMMR) immediately before setting this register.

Table 8.18 Configuration of Registers in MOD_SEL0-2

| | MOD_SEL0 | | | Initial value | MOD_SEL1 | | | Initial value | MOD_SEL2 | | | Initial value |
|-------|--------------|----------------------------|--------------|---------------|--------------------|----------------------------|-------------------|---------------|--------------------|----------------------------|--------------------|---------------|
| | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | |
| bit31 | sel_msiof3_2 | sel_msiof3_2 | sel_msiof3_2 | 0 | — | — | — | 0 | i2c_sel_5 | i2c_sel_5 | i2c_sel_5 | 0 |
| bit30 | sel_msiof3_1 | sel_msiof3_1 | sel_msiof3_1 | 0 | — | — | — | 0 | i2c_sel_3 | i2c_sel_3 | i2c_sel_3 | 0 |
| bit29 | sel_msiof3_0 | sel_msiof3_0 | sel_msiof3_0 | 0 | — | — | — | 0 | i2c_sel_0 | i2c_sel_0 | i2c_sel_0 | 0 |
| bit28 | sel_msiof2_1 | sel_msiof2_1 | sel_msiof2_1 | 0 | — | — | — | 0 | — | — | — | 0 |
| bit27 | sel_msiof2_0 | sel_msiof2_0 | sel_msiof2_0 | 0 | — | — | — | 0 | — | — | — | 0 |
| bit26 | sel_msiof1_2 | sel_msiof1_2 | sel_msiof1_2 | 0 | sel_timer_ tmu1 | sel_timer_ tmu | sel_timer_ tmu | 0 | sel_scif5 | sel_scif5 | sel_scif5 | 0 |
| bit25 | sel_msiof1_1 | sel_msiof1_1 | sel_msiof1_1 | 0 | — | — | — | 0 | sel_i2c6_2 | sel_i2c6_2 | sel_i2c6_2 | 0 |
| bit24 | sel_msiof1_0 | sel_msiof1_0 | sel_msiof1_0 | 0 | — | — | — | 0 | sel_i2c6_1 | sel_i2c6_1 | sel_i2c6_1 | 0 |
| bit23 | sel_lbsc | sel_lbsc | sel_lbsc | 0 | — | — | — | 0 | sel_i2c6_0 | sel_i2c6_0 | sel_i2c6_0 | 0 |
| bit22 | — | — | — | 0 | — | — | — | 0 | — | sel_ndf | sel_ndf | 0 |
| bit21 | sel_i2c2 | sel_i2c2 | sel_i2c2 | 0 | — | — | — | 0 | sel_ssi2 | sel_ssi2 | sel_ssi2 | 0 |
| bit20 | sel_i2c1 | sel_i2c1 | sel_i2c1 | 0 | sel_ssi1 | sel_ssi1 | sel_ssi1 | 0 | sel_ssi9 | sel_ssi9 | sel_ssi9 | 0 |
| bit19 | sel_hscif4 | sel_hscif4 | sel_hscif4 | 0 | — | — | — | 0 | sel_timer_ tmu2 | sel_timer_ tmu2 | sel_timer_ tmu2 | 0 |
| bit18 | sel_hscif3_1 | sel_hscif3_1 | sel_hscif3_1 | 0 | — | — | — | 0 | sel_adgb | sel_adgb | sel_adgb | 0 |
| bit17 | sel_hscif3_0 | sel_hscif3_0 | sel_hscif3_0 | 0 | — | — | — | 0 | sel_adgc | sel_adgc | sel_adgc | 0 |
| bit16 | sel_hscif1 | sel_hscif1 | sel_hscif1 | 0 | sel_sdhi2 | sel_sdhi2 | sel_sdhi2 | 0 | — | — | — | 0 |
| bit15 | — | — | — | 0 | sel_scif4_1 | sel_scif4_1 | sel_scif4_1 | 0 | — | — | — | 0 |
| bit14 | sel_hscif2_1 | sel_hscif2_1 | sel_hscif2_1 | 0 | sel_scif4_0 | sel_scif4_0 | sel_scif4_0 | 0 | — | — | — | 0 |
| bit13 | sel_hscif2_0 | sel_hscif2_0 | sel_hscif2_0 | 0 | sel_scif3 | sel_scif3 | sel_scif3 | 0 | — | — | — | 0 |
| bit12 | sel_etheravb | sel_etheravb | sel_etheravb | 0 | sel_scif2 | sel_scif2 | sel_scif2 | 0 | — | — | — | 0 |
| bit11 | — | — | — | 0 | sel_scif1 | sel_scif1 | sel_scif1 | 0 | — | — | — | 0 |
| bit10 | — | — | — | 0 | sel_scif | sel_scif | sel_scif | 0 | — | — | — | 0 |
| bit9 | — | — | — | 0 | — | — | — | 0 | — | — | — | 0 |
| bit8 | — | — | — | 0 | — | — | — | 0 | — | — | — | 0 |
| bit7 | — | — | — | 0 | — | — | — | 0 | — | — | — | 0 |
| bit6 | — | — | — | 0 | sel_rcan0 | sel_rcan0 | sel_rcan0 | 0 | — | — | — | 0 |
| bit5 | sel_canfd0 | sel_canfd0 | sel_canfd0 | 0 | sel_pwm6 | sel_pwm6 | sel_pwm6 | 0 | — | — | — | 0 |
| bit4 | sel_adga_1 | sel_adga_1 | sel_adga_1 | 0 | sel_pwm5 | sel_pwm5 | sel_pwm5 | 0 | — | — | — | 0 |
| bit3 | sel_adga_0 | sel_adga_0 | sel_adga_0 | 0 | sel_pwm4 | sel_pwm4 | sel_pwm4 | 0 | — | — | — | 0 |
| bit2 | — | — | — | 0 | sel_pwm3 | sel_pwm3 | sel_pwm3 | 0 | — | — | — | 0 |
| bit1 | — | — | — | 0 | sel_pwm2 | sel_pwm2 | sel_pwm2 | 0 | — | — | — | 0 |
| bit0 | — | — | — | 0 | sel_pwm1 | sel_pwm1 | sel_pwm1 | 0 | sel_vin4 | sel_vin4 | sel_vin4 | 0 |

Table 8.19 Configuration of Registers in MOD_SEL0-2

| Register | Set Value = H'0 | | Set Value = H'1 | | Set Value = H'2 | | Set Value = H'3 | | Set Value = H'4 | | Set Value = H'5 | | Set Value = H'6 | |
|-----------------------------|-----------------|--------------|-----------------|-----|-----------------|-----|-----------------|-----|-----------------|-----|-----------------|-----|-----------------|-----|
| | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin |
| sel_canfd0 | CANFD0_ RD# | WR# | CANFD0_ IRQ1 | | | | | | | | | | | |
| | RX_A | | RX_B | | | | | | | | | | | |
| | CANFD0_ RD# | | CANFD0_ IRQ0 | | | | | | | | | | | |
| | TX_A | | TX_B | | | | | | | | | | | |
| sel_etheravb | AVB_ AVB_ | | AVB_ A11 | | | | | | | | | | | |
| | AVTP_ AVTP_ | | AVTP_ | | | | | | | | | | | |
| | CAPTUR CAPTURE | | CAPTUR | | | | | | | | | | | |
| | E_A _A | | E_B | | | | | | | | | | | |
| | AVB_ AVB_ | | AVB_ A8 | | | | | | | | | | | |
| | AVTP_ AVTP_ | | AVTP_ | | | | | | | | | | | |
| | MATCH_ MATCH_A | | MATCH_ B | | | | | | | | | | | |
| | A | | | | | | | | | | | | | |
| sel_hscif1 | HCTS1#_ CTS1# | | HCTS1#_ CTS0# | | | | | | | | | | | |
| | A | | B | | | | | | | | | | | |
| | HRTS1#_ RTS1# | | HRTS1#_ RTS0# | | | | | | | | | | | |
| | A | | B | | | | | | | | | | | |
| HRX1_A RX1_A | | HRX1_B RX0 | | | | | | | | | | | | |
| HSCK1_A SSI_ | | HSCK1_B SCK0 | | | | | | | | | | | | |
| | SDATA9_ A | | | | | | | | | | | | | |
| HTX1_A TX1_A | | HTX1_B TX0 | | | | | | | | | | | | |
| Set value[1] = sel_hscif3_1 | HRX3_A RD# | | HRX3_B D10 | | HRX3_C D14 | | HRX3_D PWM1_A | | | | | | | |
| | HTX3_A RD# | WR# | HTX3_B D11 | | HTX3_C D15 | | HTX3_D PWM2_A | | | | | | | |
| Set value[0] = sel_hscif3_0 | | | | | | | | | | | | | | |
| sel_hscif4 | HRX4_A A12 | | HRX4_B A8 | | | | | | | | | | | |
| | HTX4_A A13 | | HTX4_B A11 | | | | | | | | | | | |
| sel_i2c1 | SCL1_A TX2_A | | SCL1_B MLB_CLK | | | | | | | | | | | |
| | SDA1_A RX2_A | | SDA1_B MLB_SIG | | | | | | | | | | | |
| sel_i2c2 | SCL2_A RTS0# | | SCL2_B SD0_CD | | | | | | | | | | | |
| | SDA2_A SCK0 | | SDA2_B SD0_WP | | | | | | | | | | | |
| sel_lbsc | EX_ EX_ | | EX_WAIT CS1# | | | | | | | | | | | |
| | WAIT0_A WAIT0_A | | 0_B | | | | | | | | | | | |

| Register | Set Value = H'0 | | Set Value = H'1 | | Set Value = H'2 | | Set Value = H'3 | | Set Value = H'4 | | Set Value = H'5 | | Set Value = H'6 | |
|---------------------------------------|------------------------------|-------------------------|--|------------------------------|----------------------------|--------------------------------|----------------------------|-----|-----------------|-----|-----------------|-----|-----------------|-----|
| | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin |
| Set value[2] = sel_msiof1 _2 | MSIOF1_ SSI_ RXD_A SDATA4 | MSIOF1_ CTS1# RXD_B | MSIOF1_ SSI_ RXD_C SDATA7 | MSIOF1_ HRX0 RXD_D | MSIOF1_ SD0_ RXD_E DAT0 | MSIOF1_ MLB_DAT RXD_F | MSIOF1_ SD1_ RXD_G DAT0 | | | | | | | |
| Set value[1] = sel_msiof1 _1 | MSIOF1_ SSI_ SS1_A SCK349 | MSIOF1_ RTS0# SS1_B | MSIOF1_ SSI_ SS1_C SDATA9_A | MSIOF1_ HRTS0# SS1_D | MSIOF1_ SD0_ SS1_E DAT2 | MSIOF1_ SSI_WS SS1_F 01239 | MSIOF1_ SD1_ SS1_G DAT2 | | | | | | | |
| Set value[0] = sel_msiof1 _0 | MSIOF1_ SSI_ SS2_A WS349 | MSIOF1_ SCK0 SS2_B | MSIOF1_ USB1_ SS2_C OVC | MSIOF1_ MSIOF0_ SS2_D SS2 | MSIOF1_ SD0_ SS2_E DAT3 | MSIOF1_ SSI_ SS2_F SDATA0 | MSIOF1_ SD1_ SS2_G DAT3 | | | | | | | |
| | MSIOF1_ SSI_WS4 SYNC_A | MSIOF1_ CTS0# SYNC_B | MSIOF1_ SSI_ SYNC_C WS78 | MSIOF1_ HCTS0# SYNC_D | MSIOF1_ SD0_ SYNC_E CMD | MSIOF1_ MLB_SIG SYNC_F | MSIOF1_ SD1_ SYNC_G CMD | | | | | | | |
| | MSIOF1_ SSI_ TXD_A SDATA3 | MSIOF1_ RTS1# TXD_B | MSIOF1_ SSI_ TXD_C SDATA8 | MSIOF1_ HTX0 TXD_D | MSIOF1_ SD0_DAT TXD_E 1 | MSIOF1_ SSI_SCK0 TXD_F 1239 | MSIOF1_ SD1_ TXD_G DAT1 | | | | | | | |
| Set value[1] = sel_msiof2 _1 | MSIOF2_ A10 RXD_A | MSIOF2_ D6 RXD_B | MSIOF2_ AVB_ RXD_C AVTP_ MATCH_A | MSIOF2_ D10 RXD_D | — | — | — | — | — | — | — | — | — | — |
| Set value[0] = sel_msiof2 _0 | MSIOF2_ A9 SCK_A | MSIOF2_ D4 SCK_B | MSIOF2_ AVB_LINK SCK_C | MSIOF2_ D8 SCK_D | — | — | — | — | — | — | — | — | — | — |
| | MSIOF2_ A6 SS1_A | MSIOF2_ D0 SS1_B | MSIOF2_ AVB_ SS1_C MAGIC | MSIOF2_ D12 SS1_D | — | — | — | — | — | — | — | — | — | — |
| | MSIOF2_ A7 SS2_A | MSIOF2_ D1 SS2_B | MSIOF2_ AVB_ SS2_C MDC | MSIOF2_ D13 SS2_D | — | — | — | — | — | — | — | — | — | — |
| | MSIOF2_ A8 SYNC_A | MSIOF2_ D5 SYNC_B | MSIOF2_ AVB_ SYNC_C PHY_INT | MSIOF2_ D9 SYNC_D | — | — | — | — | — | — | — | — | — | — |
| | MSIOF2_ A11 TXD_A | MSIOF2_ D7 TXD_B | MSIOF2_ AVB_ TXD_C AVTP_ CAPTURE _A | MSIOF2_ D11 TXD_D | — | — | — | — | — | — | — | — | — | — |
| sel_pwm1 | PWM1_A PWM1_A | PWM1_B A8 | — | — | — | — | — | — | — | — | — | — | — | — |
| sel_pwm2 | PWM2_A PWM2_A | PWM2_B A11 | — | — | — | — | — | — | — | — | — | — | — | — |
| sel_pwm3 | PWM3_A A0 | PWM3_B IRQ2 | — | — | — | — | — | — | — | — | — | — | — | — |
| sel_pwm4 | PWM4_A A1 | PWM4_B IRQ3 | — | — | — | — | — | — | — | — | — | — | — | — |
| sel_pwm5 | PWM5_A A2 | PWM5_B IRQ4 | — | — | — | — | — | — | — | — | — | — | — | — |
| sel_pwm6 | PWM6_A A3 | PWM6_B IRQ5 | — | — | — | — | — | — | — | — | — | — | — | — |
| sel_rcan0 | CAN0_ RD/WR# RX_A | CAN0_RX IRQ1 _B | — | — | — | — | — | — | — | — | — | — | — | — |
| | CAN0_ RD# TX_A | CAN0_TX IRQ0 _B | — | — | — | — | — | — | — | — | — | — | — | — |
| sel_scif | SCIF_ AUDIO_ CLK_A CLKB_B | SCIF_ SCK2 CLK_B | — | — | — | — | — | — | — | — | — | — | — | — |
| sel_scif1 | RX1_A RX1_A | RX1_B MLB_SIG | — | — | — | — | — | — | — | — | — | — | — | — |
| | TX1_A TX1_A | TX1_B MLB_DAT | — | — | — | — | — | — | — | — | — | — | — | — |
| sel_scif2 | RX2_A RX2_A | RX2_B HCTS0# | — | — | — | — | — | — | — | — | — | — | — | — |
| | TX2_A TX2_A | TX2_B HRTS0# | — | — | — | — | — | — | — | — | — | — | — | — |
| sel_scif3 | RX3_A RD# | RX3_B A8 | — | — | — | — | — | — | — | — | — | — | — | — |
| | TX3_A RD/WR# | TX3_B A11 | — | — | — | — | — | — | — | — | — | — | — | — |

| Register | Set Value = H'0 | | Set Value = H'1 | | Set Value = H'2 | | Set Value = H'3 | | Set Value = H'4 | | Set Value = H'5 | | Set Value = H'6 | |
|----------------------------------|-----------------|--------------------------------|-----------------|-------|-----------------|-----|-----------------|-----|-----------------|-----|-----------------|-----|-----------------|-----|
| | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin |
| Set value[1] = sel_scif4_1 | CTS4#_A | AVB_ AVTP_ MATCH_A | CTS4#_B | A9 | CTS4#_C | D10 | — | — | — | — | — | — | — | — |
| | RTS4#_A | AVB_ AVTP_ CAPTURE _A | RTS4#_B | A10 | RTS4#_C | D11 | — | — | — | — | — | — | — | — |
| Set value[0] = sel_scif4_0 | RX4_A | AVB_ PHY_INT | RX4_B | A6 | RX4_C | D12 | — | — | — | — | — | — | — | — |
| | SCK4_A | AVB_ MAGIC | SCK4_B | A5 | SCK4_C | D8 | — | — | — | — | — | — | — | — |
| sel_sdh2 | TX4_A | AVB_LINK | TX4_B | A7 | TX4_C | D13 | — | — | — | — | — | — | — | — |
| | SD2_CD_ A | SD3_ DAT4 | SD2_CD_ B | TX2_A | — | — | — | — | — | — | — | — | — | — |
| sel_vin4 | SD2_WP_ A | SD3_ DAT5 | SD2_WP_ B | RX2_A | — | — | — | — | — | — | — | — | — | — |
| | VI4_ DATA0_A | D8 | VI4_ DATA0_B | IRQ0 | — | — | — | — | — | — | — | — | — | — |
| | VI4_ DATA1_A | D9 | VI4_ DATA1_B | IRQ1 | — | — | — | — | — | — | — | — | — | — |
| | VI4_ DATA2_A | D10 | VI4_ DATA2_B | IRQ2 | — | — | — | — | — | — | — | — | — | — |
| | VI4_ DATA3_A | D11 | VI4_ DATA3_B | IRQ3 | — | — | — | — | — | — | — | — | — | — |
| | VI4_ DATA4_A | D12 | VI4_ DATA4_B | IRQ4 | — | — | — | — | — | — | — | — | — | — |
| | VI4_ DATA5_A | D13 | VI4_ DATA5_B | IRQ5 | — | — | — | — | — | — | — | — | — | — |
| | VI4_ DATA6_A | D14 | VI4_ DATA6_B | PWM0 | — | — | — | — | — | — | — | — | — | — |
| VI4_ DATA7_A | D15 | VI4_ DATA7_B | PWM1_A | — | — | — | — | — | — | — | — | — | — | |

| Register | Set Value = H'0 | | Set Value = H'1 | | Set Value = H'2 | | Set Value = H'3 | | Set Value = H'4 | | Set Value = H'5 | | Set Value = H'6 | |
|--|----------------------|--------------------------------|----------------------|--------------------------------|------------------|----------------|-----------------|-----|-----------------|-----|-----------------|-----|-----------------|-----|
| | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin |
| [I2C] | | | | | | | | | | | | | | |
| i2c_sel_0 | Other | SD1_CD | SCL0 (*2) | SD1_CD | — | — | — | — | — | — | — | — | — | — |
| | Other | SD1_WP | SDA0 (*2) | SD1_WP | — | — | — | — | — | — | — | — | — | — |
| i2c_sel_3 | Other | PWM1_A | SCL3 (*2) | PWM1_A | — | — | — | — | — | — | — | — | — | — |
| | Other | PWM2_A | SDA3 (*2) | PWM2_A | — | — | — | — | — | — | — | — | — | — |
| i2c_sel_5 | Other | AVB_ AVTP_ MATCH_A | SCL5 (*2) | AVB_ AVTP_ MATCH_A | — | — | — | — | — | — | — | — | — | — |
| | Other | AVB_ AVTP_ CAPTURE _A | SDA5 (*2) | AVB_ AVTP_ CAPTURE _A | — | — | — | — | — | — | — | — | — | — |
| sel_ssi1 | SSI_ SCK1_A | USB1_ PWEN | SSI_ SCK1_B | SSI_ SDATA2_ A | — | — | — | — | — | — | — | — | — | — |
| | SSI_ SDATA1_ A | SSI_ SDATA1_ A | SSI_ SDATA1_ B | HACK0 | — | — | — | — | — | — | — | — | — | — |
| | SSI_ WS1_A | USB1_ OVC | SSI_ WS1_B | SSI_ SDATA9_ A | — | — | — | — | — | — | — | — | — | — |
| sel_ssi2 | SSI_ SDATA2_ A | SSI_ SDATA2_ A | SSI_ SDATA2_ B | HRX0 | — | — | — | — | — | — | — | — | — | — |
| | SSI_SCK 2_A | MSIOF0_S S1 | SSI_ SCK2_B | USB30_ PWEN | — | — | — | — | — | — | — | — | — | — |
| | SSI_WS2 _A | MSIOF0_S S2 | SSI_ WS2_B | USB30_ OVC | — | — | — | — | — | — | — | — | — | — |
| sel_timer_ tmu1 | TCLK1_A | AUDIO_ CLKB_B | TCLK1_B | MSIOF0_ SS1 | — | — | — | — | — | — | — | — | — | |
| sel_timer_ tmu2 | TCLK2_A | SSI_ SDATA7 | TCLK2_B | USB30_ PWEN | — | — | — | — | — | — | — | — | — | |
| Set value[1] = sel_adga_1 Set value[0] = sel_adga_0 | AUDIO_ CLKA_A | AUDIO_ CLKA_A | AUDIO_ CLKA_B | RTS0# | AUDIO_ CLKA_C | MSIOF0_ SS1 | — | — | — | — | — | — | — | — |
| sel_adgb | AUDIO_ CLKB_A | HACK0 | AUDIO_ CLKB_B | AUDIO_ CLKB_B | — | — | — | — | — | — | — | — | — | — |
| sel_adgc | AUDIO_ CLKC_A | MSIOF0_S S2 | AUDIO_ CLKC_B | SCK0 | — | — | — | — | — | — | — | — | — | — |
| sel_scif5 | RX5_A | MSIOF0_S S1 | RX5_B | HACK0 | — | — | — | — | — | — | — | — | — | — |
| | TX5_A | MSIOF0_S S2 | TX5_B | MSIOF0_ SYNC | — | — | — | — | — | — | — | — | — | — |
| | SCK5_A | SSI_ SDATA9_ A | SCK5_B | SCK0 | — | — | — | — | — | — | — | — | — | — |

| Register | Set Value = H'0 | | Set Value = H'1 | | Set Value = H'2 | | Set Value = H'3 | | Set Value = H'4 | | Set Value = H'5 | | Set Value = H'6 | |
|----------------|-----------------|------------|-----------------|--------------|-----------------|------------|-----------------|--------|-----------------|---------|-----------------|-----|-----------------|-----|
| | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin |
| sel_ndf | NFCE#_A | SD1_WP | NFCE#_B | SD1_CMD | — | — | — | — | — | — | — | — | — | — |
| | NFWP#_A | MSIOF0_S1 | NFWP#_B | SD1_DAT0 | — | — | — | — | — | — | — | — | — | — |
| | NFDATA14_A | SD0_CD | NFDATA14_B | SD1_DAT1 | — | — | — | — | — | — | — | — | — | — |
| | NFDATA15_A | SD0_WP | NFDATA15_B | SD1_DAT2 | — | — | — | — | — | — | — | — | — | — |
| | NFRB#_A | SD1_CD | NFRB#_B | SD1_DAT3 | — | — | — | — | — | — | — | — | — | — |
| Set value[2] = | MSIOF3_RXD_A | D2 | MSIOF3_RXD_B | A3 | MSIOF3_RXD_C | A14 | MSIOF3_RXD_D | RD/WR# | MSIOF3_RXD_E | IRQ4(*) | — | — | — | — |
| sel_msiof3_2 | MSIOF3_SCK_A | D0 | MSIOF3_SCK_B | A2 | MSIOF3_SCK_C | A12 | MSIOF3_SCK_D | BS# | MSIOF3_SCK_E | IRQ3(*) | — | — | — | — |
| Set value[1] = | MSIOF3_SS1_A | D14 | MSIOF3_SS1_B | A4 | — | — | MSIOF3_SS1_D | WE1# | MSIOF3_SS1_E | IRQ1(*) | — | — | — | — |
| sel_msiof3_1 | MSIOF3_SS2_A | D15 | MSIOF3_SS2_B | A5 | — | — | — | — | MSIOF3_SS2_E | IRQ0(*) | — | — | — | — |
| Set value[0] = | MSIOF3_SYNC_A | D1 | MSIOF3_SYNC_B | A0 | MSIOF3_SYNC_C | A13 | MSIOF3_SYNC_D | RD# | MSIOF3_SYNC_E | IRQ2(*) | — | — | — | — |
| sel_msiof3_0 | MSIOF3_TXD_A | D3 | MSIOF3_TXD_B | A1 | MSIOF3_TXD_C | A15 | MSIOF3_TXD_D | WE0# | MSIOF3_TXD_E | IRQ5(*) | — | — | — | — |
| Set value[1] = | HCTS2#_A | SSI_WS349 | HCTS2#_B | SSI_SDATA7 | HCTS2#_C | USB1_OVC | — | — | — | — | — | — | — | — |
| sel_hscif2_1 | HRTS2#_A | SSI_SDATA3 | HRTS2#_B | SSI_SDATA8 | HRTS2#_C | USB30_PWEN | — | — | — | — | — | — | — | — |
| Set value[0] = | HRX2_A | SSI_SCK4 | HRX2_B | SSI_SCK78 | HRX2_C | USB0_OVC | — | — | — | — | — | — | — | — |
| sel_hscif2_0 | HSCK2_A | SSI_SDATA4 | HSCK2_B | SSI_SDATA9_A | HSCK2_C | USB0_PWEN | — | — | — | — | — | — | — | — |
| | HTX2_A | SSI_WS4 | HTX2_B | SSI_WS78 | HTX2_C | USB1_PWEN | — | — | — | — | — | — | — | — |
| Set value[2] = | SCL6_A | A11 | SCL6_B | WE0# | SCL6_C | D14 | — | — | — | — | — | — | — | — |
| sel_i2c6_2 | | | | | | | | | | | | | | |
| Set value[1] = | SDA6_A | A8 | SDA6_B | WE1# | SDA6_C | D15 | — | — | — | — | — | — | — | — |
| sel_i2c6_1 | | | | | | | | | | | | | | |
| Set value[0] = | | | | | | | | | | | | | | |
| sel_i2c6_0 | | | | | | | | | | | | | | |

| Register | Set Value = H'0 | | Set Value = H'1 | | Set Value = H'2 | | Set Value = H'3 | | Set Value = H'4 | | Set Value = H'5 | | Set Value = H'6 | |
|----------|-----------------|----------|-----------------|---|-----------------|-----|-----------------|-----|-----------------|-----|-----------------|-----|-----------------|-----|
| | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin |
| sel_ssi9 | SSL_ | HCTS0# | SSL_ | [RZ/G2H] | — | — | — | — | — | — | — | — | — | — |
| | SCK9_A | | SCK9_B | [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] GP6_30 *1 | | | | | | | | | | |
| | SSI_WS9_A | HRTS0# | SSI_WS9_B | [RZ/G2H] [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] GP6_31 *1 | | | | | | | | | | |
| SSL_ | SSL_ | SSL_ | HTX0 | — | — | — | — | — | — | — | — | — | — | — |
| | SDATA9_A | SDATA9_A | SDATA9_B | | | | | | | | | | | |

- Note:
1. Different for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N
 2. When I2C is used on 1.8v, these pins must be Off PULLUP just after power on by software. I2C device should be confirmed voltage rating of connecting I2C device, it is 3.3v of these pins.

8.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

8.3.1 Function Setting for Multiplexed Pins

Setting the LSI multiplexed pin setting mask register (PMMR) is necessary before setting each of the GPIO/peripheral function select registers GPSR0 to GPSR7, peripheral function select registers IPSR0 to IPSR18, DRV control registers DRVCTRL0-24, TDSEL control registers TDSELCTRL0, POC control register POCCTRL0 and Module select register MOD_SEL0-2. Specifically, the inverse of the value to be set in the select register must be written to the LSI multiplexed pin setting mask register. Otherwise, the GPIO/peripheral function select registers 0 to 7 (GPSR0 to GPSR7) and peripheral function select registers 0 to 18 (IPSR0 to IPSR18) cannot be set. IPSR0 to IPSR18, and MOD_SEL0 to MOD_SEL2 registers shall be set before setting GPSR0 to GPSR7 registers in case that they need to be configured. MOD_SEL0 to MOD_SEL2 registers can be set either earlier or later than setting IPSR0 to IPSR18 registers.

(1) Procedure for changing pin function from GPIO to peripheral function

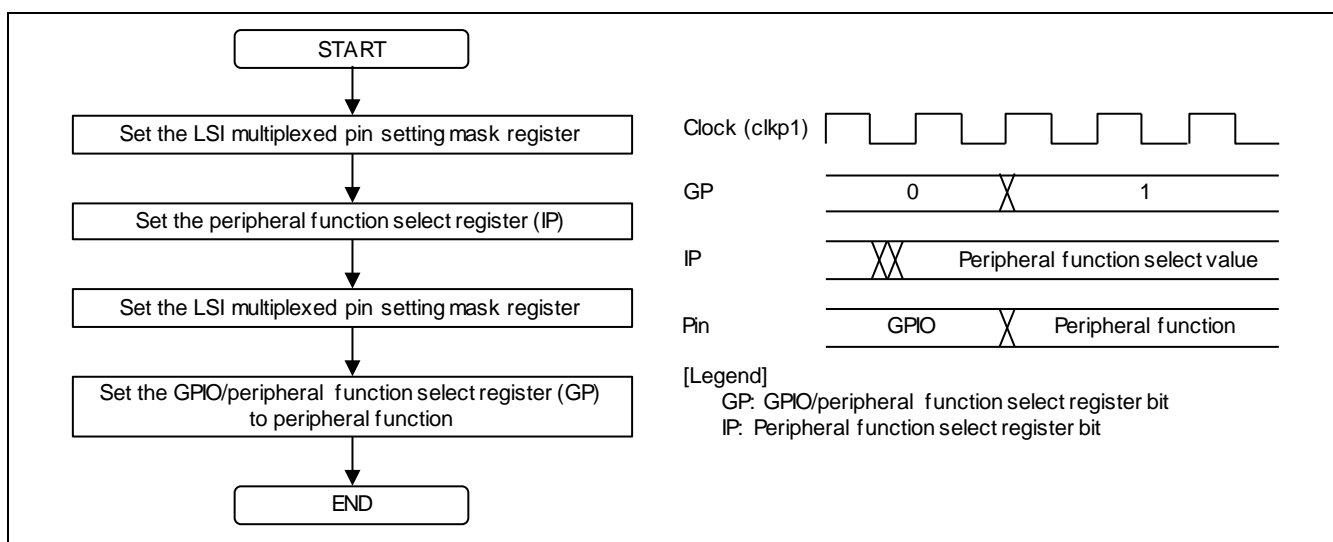


Figure 8.2 Procedure for Changing Pin Function from GPIO to Peripheral Function

(2) Procedure for changing pin function from peripheral function to GPIO

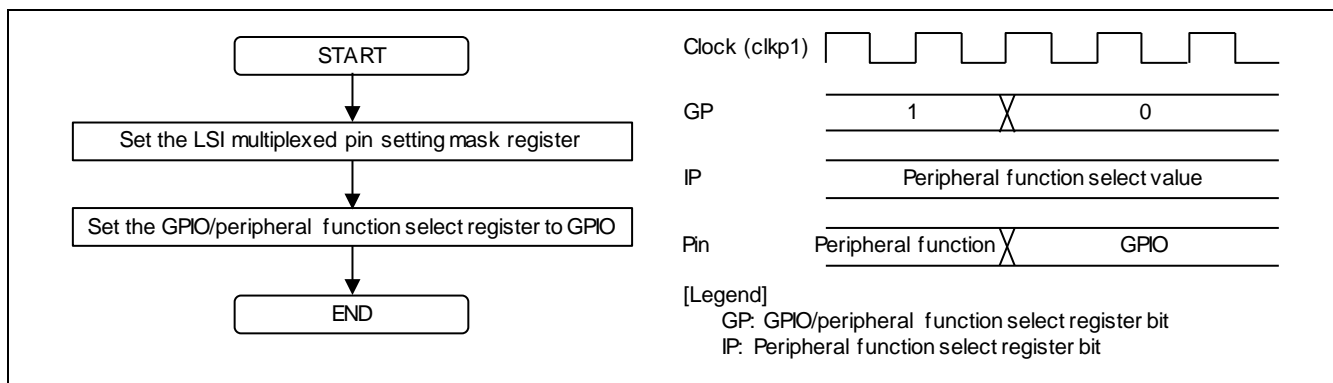


Figure 8.3 Procedure for Changing Pin Function from Peripheral function to GPIO

(3) Procedure 1 for changing pin function from one peripheral function to another peripheral function

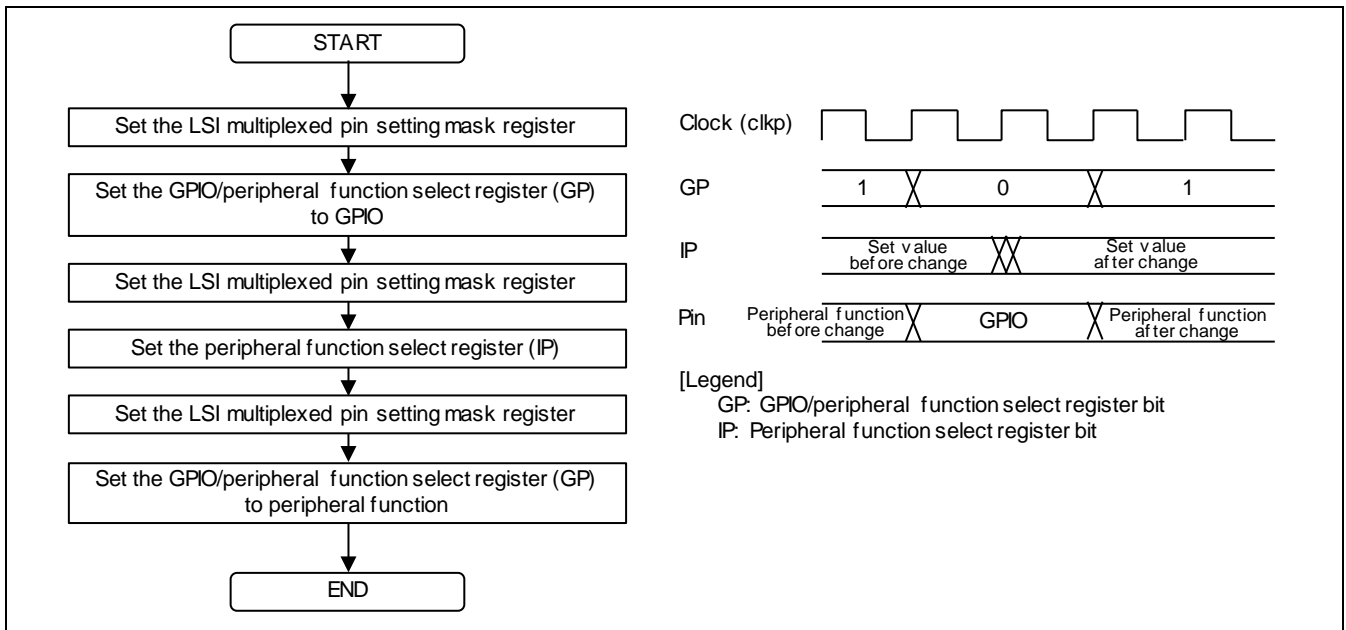


Figure 8.4 Procedure for Changing Pin Function from One Peripheral Function to Another Peripheral Function (with GPIO Setting)

8.3.2 Setting Pull-Up/Down Resistors

The LSI pin pull-on/off control registers 0 to 6 (PUEN0 to PUEN6) and pull-up/down control registers 0 to 6 (PUD0 to PUD6) are used.

8.4 Usage Note

8.4.1 Notes for Using I2C0/3/5

When using the I2C0/3/5, the external connected devices must be 3.3-V compatible devices even though using the I2C0/3/5 system with 1.8-V.

If not using 3.3-V compatible devices, the external devices may be permanently damaged at a power-on reset.

Note.

1. The default functions of I2C0/3/5 pins after a power-on reset are GPIOs and pulled-up by 3.3-V through the VDDQ33 until they are specified for the I2C function or pulled-up off when other than I2C function is selected.
2. During the I2C0/3/5 pins are default state, i.e., GPIOs pins and pulled-up at 3.3-V, the voltage of external devices connected to these pins shall be up to 3.3-V through the VDDQ33.

Note that an internal leakage current of the I2C0/3/5 will be up to 55uA from the 3.3-V line to the 1.8-V line of each I2C block until the pulled-up off.

9. Pin Function Controller (PFC)



9.1 Overview

The pin function controller (PFC) is a module that consists of registers for selecting the function of the multiplexed pins and controlling the pull-up resistor on each LSI pin.

9.1.1 Features

- Setting multiplexed pin functions for LSI pins
Function of the RZ/G2E pin selectable by setting the registers in the PFC module (The function of the LSI pin can be selected by the GPIO/peripheral function select registers 0 to 6 (GPSR0 to GPSR6) and peripheral function select registers 0 to 15 (IPSR0 to IPSR15) in the PFC module. For details, see sections 9.2.2, GPIO/Peripheral Function Select Register 0-6 (GPSR0-6) through 9.2.3, Peripheral Function Select Register 0-15 (IPSR0-15).)
- POC control for each LSI pin
POC control registers must be set according to IO voltage level that is supplied to the pin.
POC control resistors on each LSI pin can be controlled by setting the registers in the PFC module. (Selection is handled by the IO voltage level select registers 0, 2 (POCCTRL0, POCCTRL2). For details, see sections 9.2.5, POC Control Register 0 (POCCTRL0) through 9.2.6, POC Control Register 2 (POCCTRL2).)
- TDSEL control for each LSI pin
TDSEL control registers can control the driving abilities of pins in use for the SDHI.
TDSEL control resistors on each LSI pin can be controlled by setting the registers in the PFC module. (Selection is handled by the return path for clock drive select register 0 (TDSELCTRL0). For details, see section 9.2.7, TDSEL Control Register 0 (TDSELCTRL0).)
- Pull-up/down control for each LSI pin.
PUEN registers can on/off control of the pull resistors.
On/off of the pull resistors on each LSI pin can be controlled by setting the registers in the PFC module. (Selection is handled by the Pull-on/off select registers 0 to 5 (PUEN0 to PUEN 5). For details, see section 9.2.8, LSI pin pull-enable register 0-5 (PUEN0-5).
PUD registers can pull-up/pull-down control of the pull resistors.
Pull-up/Pull-down control resistors on each LSI pin can be controlled by setting the registers in the PFC module. (Selection is handled by the Pull-up/down select registers 0 to 5 (PUD0 to PUD 5). For details, see section 9.2.9, LSI pin pull-up/down control Register 0-5 (PUD0-5).)
- Module selection
Module Select Register can select the group for multiple LSI pins with multiplexed pin functions.
Enable and disable the functions of RZ/G2E LSI pins to which pin functions from multiple pin groups are assigned by setting the registers in the PFC module.
(Selection is handled by the module select register 0 to 1 (MOD_SEL0 to MOD_SEL1). For details, see section 9.2.10, Module Select Register 0-1 (MOD_SEL0-1).)
- Notes on configuring multiplexed pin functions
The multiplexed LSI pins (MOD_SEL0-1, GPSR0-6, IPSR0-15) must be set in the initial sequence (*).
Switching multiplexed LSI pins during operation is not guaranteed.
*: The initial sequence is Appendix B.(3)
- Module clock
Module clock is S3D4φ.

9.1.2 Block Diagram

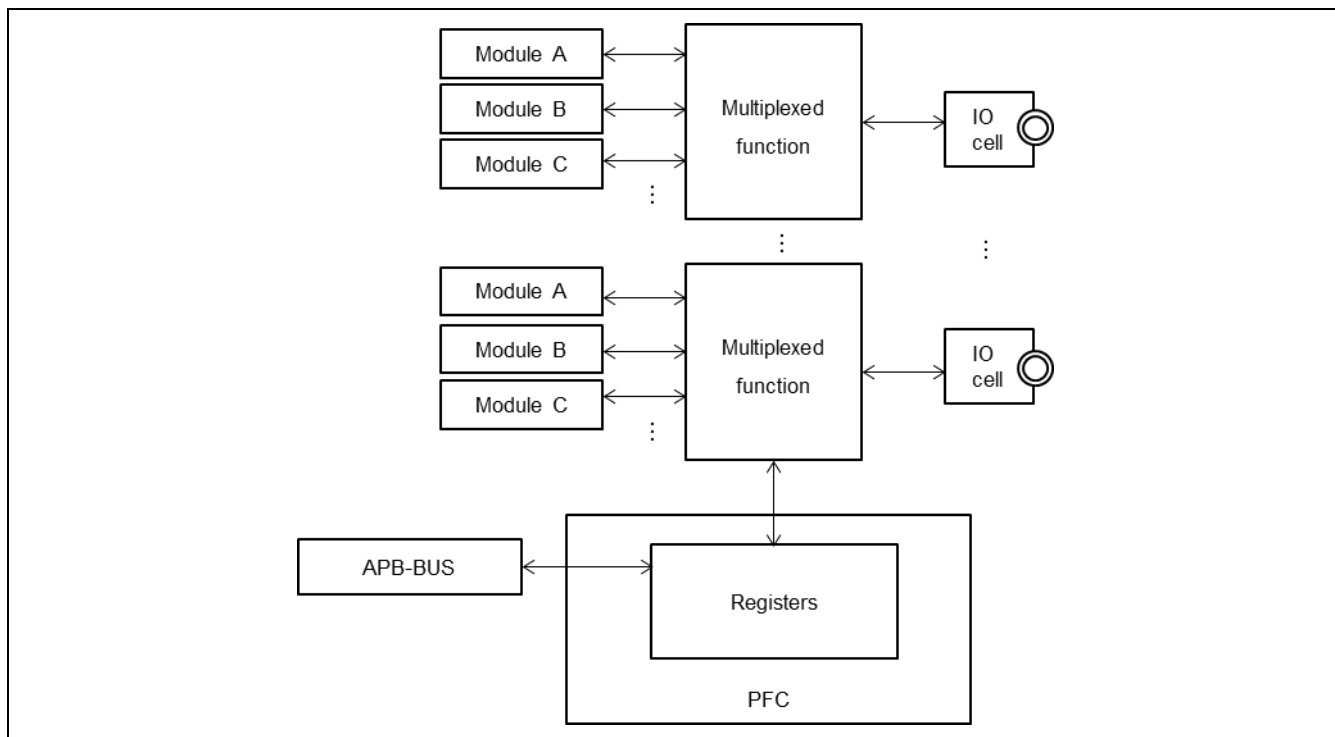


Figure 9.1 PFC Block Configuration

9.1.3 External Pins

PFC does not have External Pins.

9.1.4 Register Configuration

All the registers in the PFC are mapped into the APB bus space. Table 9.1 shows the configuration of the registers provided in the PFC. Details on each register in the PFC are given in sections 9.2.1 to 9.2.10.

Table 9.1 Configuration of Registers in PFC

| Name | Abbr. | R/W | Initial Value | Address | Access Size | Condition |
|--|-------|-----|--|-------------|-------------|-----------|
| LSI Multiplexed Pin Setting Mask Register | PMMR | R/W | H'0000_0000 | H'E606_0000 | 32 | — |
| GPIO/Peripheral Function Select register 0 | GPSR0 | R/W | H'0003_FFFF (when md[4:1] = B'0000), H'0000_0000 (when md[4:1] ≠ B'0000) | H'E606_0100 | 32 | — |
| GPIO/Peripheral Function Select register 1 | GPSR1 | R/W | H'007F_FFFF (when md[4:1] = B'0000), H'1000_0000 (when md[4:1] ≠ B'0000) | H'E606_0104 | 32 | — |
| GPIO/Peripheral Function Select register 2 | GPSR2 | R/W | H'0EFF_FFFF (when md[4:1] = B'0000) H'0C3F_C03F (when md[4:1] = B'0001) H'0C3F_FFFF (when md[4:1] = B'0010) H'0C3F_FFFF (when md[4:1] = B'0011) H'0C3F_C03F (when md[4:1] = B'0100) H'0C3F_C000 (when md[4:1] = B'0101) H'0C3F_C03F (when md[4:1] = B'0110) H'0C3F_FFFF (when md[4:1] = B'0111) H'0C3F_C03F (when md[4:1] = B'1000) H'0C3F_C03F (when md[4:1] = B'1001) H'0C3F_FFFF (when md[4:1] = B'1010) H'0C3F_FFFF (when md[4:1] = B'1011) H'0C3FFFFF (when md[4:1] = B'1100) H'0C3FC000 (when md[4:1] = B'1101) H'0C3F_C000 (when md[4:1] = B'1110) H'0C3F_C000 (when md[4:1] = B'1111) | H'E606_0108 | 32 | — |
| GPIO/Peripheral Function Select register 3 | GPSR3 | R/W | H'0000_0000 | H'E606_010C | 32 | — |
| GPIO/Peripheral Function Select register 4 | GPSR4 | R/W | H'0000_0000 | H'E606_0110 | 32 | — |

| Name | Abbr. | R/W | Initial Value | Address | Access Size | Condition |
|---|----------------|-----|---------------|-------------|-------------|-----------|
| GPIO/Peripheral Function Select register 5 | GPSR5 | R/W | H'0000_0000 | H'E606_0114 | 32 | — |
| GPIO/Peripheral Function Select register 6 | GPSR6 | R/W | H'0002_0200 | H'E606_0118 | 32 | — |
| Peripheral Function Select register 0 | IPSR0 | R/W | H'0000_0000 | H'E606_0200 | 32 | — |
| Peripheral Function Select register 1 | IPSR1 | R/W | H'0000_0000 | H'E606_0204 | 32 | — |
| Peripheral Function Select register 2 | IPSR2 | R/W | H'0000_0000 | H'E606_0208 | 32 | — |
| Peripheral Function Select register 3 | IPSR3 | R/W | H'0000_0000 | H'E606_020C | 32 | — |
| Peripheral Function Select register 4 | IPSR4 | R/W | H'0000_0000 | H'E606_0210 | 32 | — |
| Peripheral Function Select register 5 | IPSR5 | R/W | H'0000_0000 | H'E606_0214 | 32 | — |
| Peripheral Function Select register 6 | IPSR6 | R/W | H'0000_0000 | H'E606_0218 | 32 | — |
| Peripheral Function Select register 7 | IPSR7 | R/W | H'0110_0000 | H'E606_021C | 32 | — |
| Peripheral Function Select register 8 | IPSR8 | R/W | H'0000_0000 | H'E606_0220 | 32 | — |
| Peripheral Function Select register 9 | IPSR9 | R/W | H'0000_0000 | H'E606_0224 | 32 | — |
| Peripheral Function Select register 10 | IPSR10 | R/W | H'0000_0000 | H'E606_0228 | 32 | — |
| Peripheral Function Select register 11 | IPSR11 | R/W | H'0000_0000 | H'E606_022C | 32 | — |
| Peripheral Function Select register 12 | IPSR12 | R/W | H'0000_0000 | H'E606_0230 | 32 | — |
| Peripheral Function Select register 13 | IPSR13 | R/W | H'0000_0000 | H'E606_0234 | 32 | — |
| Peripheral Function Select register 14 | IPSR14 | R/W | H'0000_0000 | H'E606_0238 | 32 | — |
| Peripheral Function Select register 15 | IPSR15 | R/W | H'0000_0000 | H'E606_023C | 32 | — |
| DRV control register 8 | DRVCTR L8 | R/W | H'9244_9249 | H'E606_0320 | 32 | — |
| DRV control register 9 | DRVCTR L9 | R/W | H'2492_4924 | H'E606_0324 | 32 | — |
| DRV control register 10 | DRVCTR L10 | R/W | H'2492_5258 | H'E606_0328 | 32 | — |
| POC control register 0 | POCCTR L0 | R/W | H'3FFF_FFFF | H'E606_0380 | 32 | — |
| POC control register 2 | POCCTR L2 | R/W | H'FFFF_FFFF | H'E606_0388 | 32 | — |
| TDSEL control register 0 | TDSELC TRL0 | R/W | H'0000_0000 | H'E606_03C0 | 32 | — |
| LSI pin pull-enable register 0 | PUEN0 | R/W | H'000F_FFE4 | H'E606_0400 | 32 | — |
| LSI pin pull-enable register 1 | PUEN1 | R/W | H'CE29_8464 | H'E606_0404 | 32 | — |

| Name | Abbr. | R/W | Initial Value | Address | Access Size | Condition |
|---|----------|-----|---------------|-------------|-------------|-----------|
| LSI pin pull-enable register 2 | PUEN2 | R/W | H'A4C3_80F4 | H'E606_0408 | 32 | — |
| LSI pin pull-enable register 3 | PUEN3 | R/W | H'0000_079F | H'E606_040C | 32 | — |
| LSI pin pull-enable register 4 | PUEN4 | R/W | H'FFF0_FFFF | H'E606_0410 | 32 | — |
| LSI pin pull-enable register 5 | PUEN5 | R/W | H'4000_0000 | H'E606_0414 | 32 | — |
| LSI pin pull-up/down control register 0 | PUD0 | R/W | H'0008_0000 | H'E606_0440 | 32 | — |
| LSI pin pull-up/down control register 1 | PUD1 | R/W | H'CE29_8464 | H'E606_0444 | 32 | — |
| LSI pin pull-up/down control register 2 | PUD2 | R/W | H'A4C3_80F0 | H'E606_0448 | 32 | — |
| LSI pin pull-up/down control register 3 | PUD3 | R/W | H'0000_079F | H'E606_044C | 32 | — |
| LSI pin pull-up/down control register 4 | PUD4 | R/W | H'FFF0_FFFF | H'E606_0450 | 32 | — |
| LSI pin pull-up/down control register 5 | PUD5 | R/W | H'4000_0000 | H'E606_0454 | 32 | — |
| Module select register 0 | MOD_SEL0 | R/W | H'0000_0000 | H'E606_0500 | 32 | — |
| Module select register 1 | MOD_SEL1 | R/W | H'0000_0000 | H'E606_0504 | 32 | — |

9.1.5 Connected Module

Table 9.2 Connected module

| Module name | Connected module name | Function of connected module |
|-------------|-----------------------|------------------------------|
| PFC | AP-System Core | Access the Registers |
| | CPG | Output clocks |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |

9.2 Register Description

[Legend]

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R: Read-only. The write value should always be 0.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

W: Write-only. Reading this bit is prohibited. When the bit is reserved, the write value should always be 0.

—/W: Write-only. The read value is undefined.

All the bits are active high unless otherwise specified, and deactivated on reset.

All access to registers is made in longword units.

The write value to a reserved bit should always be 0.

9.2.1 LSI Multiplexed Pin Setting Mask Register (PMMR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

Function: PMMR enables/disables writing to the multiplexed pin setting registers.

| | | | | | | | | | | | | | | | | |
|----------------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MPM[31] | MPM[30] | MPM[29] | MPM[28] | MPM[27] | MPM[26] | MPM[25] | MPM[24] | MPM[23] | MPM[22] | MPM[21] | MPM[20] | MPM[19] | MPM[18] | MPM[17] | MPM[16] |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MPM[15] | MPM[14] | MPM[13] | MPM[12] | MPM[11] | MPM[10] | MPM[9] | MPM[8] | MPM[7] | MPM[6] | MPM[5] | MPM[4] | MPM[3] | MPM[2] | MPM[1] | MPM[0] |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 0 | MPM[31:0] | H'0000_0000 | R/W | Multiplexed Pin Setting Mask Writing a value to any register from among the GPIO/peripheral function select registers GPSR0 to GPSR6, peripheral function select registers IPSR0 to IPSR15, TDSEL control register TDSELCTRL0, Module select register MOD_SEL0 to MOD_SEL1 and POC control register POCCTRL0, POCCTRL2 is enabled by writing the inverse of the value to this register. |

Note: This register must be set before setting each of the GPIO/peripheral function select registers GPSR0 to GPSR6, peripheral function select registers IPSR0 to IPSR15, TDSEL control register TDSELCTRL0, POC control register POCCTRL0, POCCTRL2 and Module select register MOD_SEL0 to MOD_SEL1.

9.2.2 GPIO/Peripheral Function Select Register 0-6 (GPSR0-6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

Function: GPSR0-6 selects the functions of the multiplexed LSI pins.

| | | | | | | | | | | | | | | | | |
|------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | GPn* [31] | GPn* [30] | GPn* [29] | GPn* [28] | GPn* [27] | GPn* [26] | GPn* [25] | GPn* [24] | GPn* [23] | GPn* [22] | GPn* [21] | GPn* [20] | GPn* [19] | GPn* [18] | GPn 1[17] | GPn* [16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | GPn* [15] | GPn* [14] | GPn* [13] | GPn* [12] | GPn* [11] | GPn* [10] | GPn* [9] | GPn* [8] | GPn* [7] | GPn* [6] | GPn* [5] | GPn* [4] | GPn* [3] | GPn* [2] | GPn* [1] | GPn* [0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * n = 0-6

| Bit | Bit Name | R/W | Description |
|---------|-------------|-----|---|
| 31 to 0 | GP0-6[31:0] | R/W | 0: GPIO 1: Peripheral function The functions of the LSI pins are selected according to the table below. |

Note: To enable this register to be set, appropriately set the multiplexed pin setting mask register (PMMR) immediately before setting this register.

Table 9.3 Configuration of Registers in GPSR0-2

| bit | GPSR0 | Initial value | GPSR1 | Initial value | GPSR2 | Initial value | | | |
|-------|-------|---------------|--------|---------------|----------------|------------------|--------------------------|-------------------------------|-------------------|
| | | | | | | MD[4:1]= LBSC | MD[4:1]= series flash | MD[4:1]=eMMC, download mod | MD[4:1]= other |
| bit31 | — | 0 | — | 0 | — | 0 | 0 | 0 | 0 |
| bit30 | — | 0 | — | 0 | — | 0 | 0 | 0 | 0 |
| bit29 | — | 0 | — | 0 | — | 0 | 0 | 0 | 0 |
| bit28 | — | 0 | — | 0 | — | 0 | 0 | 0 | 0 |
| bit27 | — | 0 | — | 0 | — | 1 | 1 | 1 | 1 |
| bit26 | — | 0 | — | 0 | — | 1 | 1 | 1 | 1 |
| bit25 | — | 0 | — | 0 | EX_WAIT0 | 1 | 0 | 0 | 0 |
| bit24 | — | 0 | — | 0 | RD/WR# | 0 | 0 | 0 | 0 |
| bit23 | — | 0 | — | 0 | RD# | 1 | 0 | 0 | 0 |
| bit22 | — | 0 | WE0# | 1 | BS# | 1 | 0 | 0 | 0 |
| bit21 | — | 0 | CS0# | 1 | AVB_PHY_INT | 1 | 1 | 1 | 1 |
| bit20 | — | 0 | CLKOUT | 1 | AVB_TXCREFLCK | 1 | 1 | 1 | 1 |
| bit19 | — | 0 | A19 | 1 | AVB_RD3 | 1 | 1 | 1 | 1 |
| bit18 | — | 0 | A18 | 1 | AVB_RD2 | 1 | 1 | 1 | 1 |
| bit17 | SDA4 | 1 | A17 | 1 | AVB_RD1 | 1 | 1 | 1 | 1 |
| bit16 | SCL4 | 1 | A16 | 1 | AVB_RD0 | 1 | 1 | 1 | 1 |
| bit15 | D15 | 1 | A15 | 1 | AVB_RXC | 1 | 1 | 1 | 1 |
| bit14 | D14 | 1 | A14 | 1 | AVB_RX_CTL | 1 | 1 | 1 | 1 |
| bit13 | D13 | 1 | A13 | 1 | RPC_RESET# | 1 | 0 | 0 | 1 |
| bit12 | D12 | 1 | A12 | 1 | RPC_INT# | 1 | 0 | 0 | 1 |
| bit11 | D11 | 1 | A11 | 1 | QSPI1_SSL | 1 | 0 | 0 | 1 |
| bit10 | D10 | 1 | A10 | 1 | QSPI1_IO3 | 1 | 0 | 0 | 1 |
| bit9 | D9 | 1 | A9 | 1 | QSPI1_IO2 | 1 | 0 | 0 | 1 |
| bit8 | D8 | 1 | A8 | 1 | QSPI1_MISO/IO1 | 1 | 0 | 0 | 1 |
| bit7 | D7 | 1 | A7 | 1 | QSPI1_MOSI/IO0 | 1 | 0 | 0 | 1 |
| bit6 | D6 | 1 | A6 | 1 | QSPI1_SPCLK | 1 | 0 | 0 | 1 |
| bit5 | D5 | 1 | A5 | 1 | QSPI0_SSL | 1 | 1 | 0 | 1 |
| bit4 | D4 | 1 | A4 | 1 | QSPI0_IO3 | 1 | 1 | 0 | 1 |
| bit3 | D3 | 1 | A3 | 1 | QSPI0_IO2 | 1 | 1 | 0 | 1 |
| bit2 | D2 | 1 | A2 | 1 | QSPI0_MISO/IO1 | 1 | 1 | 0 | 1 |
| bit1 | D1 | 1 | A1 | 1 | QSPI0_MOSI/IO0 | 1 | 1 | 0 | 1 |
| bit0 | D0 | 1 | A0 | 1 | QSPI0_SPCLK | 1 | 1 | 0 | 1 |

Table 9.4 Configuration of Registers in GPSR3-6

| | GPSR3 | Initial value | GPSR4 | Initial value | GPSR5 | Initial value | GPSR6 | Initial value |
|-------|----------|---------------|----------|---------------|-------------|---------------|--------------|---------------|
| bit31 | — | 0 | — | 0 | — | 0 | — | 0 |
| bit30 | — | 0 | — | 0 | — | 0 | — | 0 |
| bit29 | — | 0 | — | 0 | — | 0 | — | 0 |
| bit28 | — | 0 | — | 0 | — | 0 | — | 0 |
| bit27 | — | 0 | — | 0 | — | 0 | — | 0 |
| bit26 | — | 0 | — | 0 | — | 0 | — | 0 |
| bit25 | — | 0 | — | 0 | — | 0 | — | 0 |
| bit24 | — | 0 | — | 0 | — | 0 | — | 0 |
| bit23 | — | 0 | — | 0 | — | 0 | — | 0 |
| bit22 | — | 0 | — | 0 | — | 0 | — | 0 |
| bit21 | — | 0 | — | 0 | — | 0 | — | 0 |
| bit20 | — | 0 | — | 0 | — | 0 | — | 0 |
| bit19 | — | 0 | — | 0 | MLB_DAT | 0 | — | 0 |
| bit18 | — | 0 | — | 0 | MLB_SIG | 0 | — | 0 |
| bit17 | — | 0 | — | 0 | MLB_CLK | 0 | USB30_PWEN | 1 |
| bit16 | — | 0 | — | 0 | SSI_SDATA9 | 0 | SSI_SDATA6 | 0 |
| bit15 | SD1_WP | 0 | — | 0 | MSIOF0_SS2 | 0 | SSI_WS6 | 0 |
| bit14 | SD1_CD | 0 | — | 0 | MSIOF0_SS1 | 0 | SSI_SCK6 | 0 |
| bit13 | SD0_WP | 0 | — | 0 | MSIOF0_SYNC | 0 | SSI_SDATA5 | 0 |
| bit12 | SD0_CD | 0 | — | 0 | MSIOF0_TXD | 0 | SSI_WS5 | 0 |
| bit11 | SD1_DAT3 | 0 | — | 0 | MSIOF0_RXD | 0 | SSI_SCK5 | 0 |
| bit10 | SD1_DAT2 | 0 | SD3_DS | 0 | MSIOF0_SCK | 0 | SSI_SDATA4 | 0 |
| bit9 | SD1_DAT1 | 0 | SD3_DAT7 | 0 | RX2_A | 0 | USB30_OVC | 1 |
| bit8 | SD1_DAT0 | 0 | SD3_DAT6 | 0 | TX2_A | 0 | AUDIO_CLKA | 0 |
| bit7 | SD1_CMD | 0 | SD3_DAT5 | 0 | SCK2_A | 0 | SSI_SDATA3 | 0 |
| bit6 | SD1_CLK | 0 | SD3_DAT4 | 0 | TX1 | 0 | SSI_WS349 | 0 |
| bit5 | SD0_DAT3 | 0 | SD3_DAT3 | 0 | RX1 | 0 | SSI_SCK349 | 0 |
| bit4 | SD0_DAT2 | 0 | SD3_DAT2 | 0 | RTS0#_A | 0 | SSI_SDATA2 | 0 |
| bit3 | SD0_DAT1 | 0 | SD3_DAT1 | 0 | CTS0#_A | 0 | SSI_SDATA1 | 0 |
| bit2 | SD0_DAT0 | 0 | SD3_DAT0 | 0 | TX0_A | 0 | SSI_SDATA0 | 0 |
| bit1 | SD0_CMD | 0 | SD3_CMD | 0 | RX0_A | 0 | SSI_WS01239 | 0 |
| bit0 | SD0_CLK | 0 | SD3_CLK | 0 | SCK0_A | 0 | SSI_SCK01239 | 0 |

9.2.3 Peripheral Function Select Register 0-15 (IPSR0-15)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

Function: IPSR0-15 selects the functions of the multiplexed LSI pins.

| | | | | | | | | | | | | | | | | |
|------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | IPn* [31] | IPn* [30] | IPn* [29] | IPn* [28] | IPn* [27] | IPn* [26] | IPn* [25] | IPn* [24] | IPn* [23] | IPn* [22] | IPn* [21] | IPn* [20] | IPn* [19] | IPn* [18] | IPn* [17] | IPn* [16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | IPn*[15] | IPn*[14] | IPn*[13] | IPn*[12] | IPn*[11] | IPn*[10] | IPn*[9] | IPn*[8] | IPn*[7] | IPn*[6] | IPn*[5] | IPn*[4] | IPn*[3] | IPn*[2] | IPn*[1] | IPn*[0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * n = 0-15

| Bit | R/W | Description |
|---------|-----|--|
| 31 to 0 | R/W | The functions of the LSI pins are selected according to the table below. |

Note: To enable this register to be set, appropriately set the multiplexed pin setting mask register (PMMR) immediately before setting this register.

IPSR register must not set "—", "Reserved" and IPSR = H'9, H'A, H'B, H'C, H'D, H'E, H'F.

Table 9.5 Configuration of Registers in IPSR

| Register | IPSR = H'0 | IPSR = H'1 | IPSR = H'2 | IPSR = H'3 | IPSR = H'4 | IPSR = H'5 | IPSR = H'6 | IPSR = H'7 | IPSR = H'8 |
|------------|----------------|------------|------------------|-------------|------------|------------|------------|-------------|------------|
| IP0[3:0] | QSPI0_SPCLK | HCK4_A | | | | | | | |
| IP0[7:4] | QSPI0_MOSI/IO0 | HCTS4#_A | | | | | | | |
| IP0[11:8] | QSPI0_MISO/IO1 | HRTS4#_A | | | | | | | |
| IP0[15:12] | QSPI0_IO2 | HTX4_A | | | | | | | |
| IP0[19:16] | QSPI0_IO3 | HRX4_A | | | | | | | |
| IP0[23:20] | QSPI1_SPCLK | | HCK4_B | VI4_DATA0_A | | | | | |
| IP0[27:24] | QSPI1_MOSI/IO0 | | HTX4_B | VI4_DATA1_A | | | | | |
| IP0[31:28] | QSPI1_MISO/IO1 | | HRX4_B | VI4_DATA2_A | | | | | |
| IP1[3:0] | QSPI1_IO2 | | HTX3_C | VI4_DATA3_A | | | | | |
| IP1[7:4] | QSPI1_IO3 | | HRX3_C | VI4_DATA4_A | | | | | |
| IP1[11:8] | QSPI1_SSL | | HCK3_C | VI4_DATA5_A | | | | | |
| IP1[15:12] | RPC_INT# | | HCTS3#_C | VI4_DATA6_A | | | | | |
| IP1[19:16] | RPC_RESET# | | HRTS3#_C | VI4_DATA7_A | | | | | |
| IP1[23:20] | AVB_RD0 | | | | | | | | |
| IP1[27:24] | AVB_RD1 | | | | | | | | |
| IP1[31:28] | AVB_RD2 | | | | | | | | |
| IP2[3:0] | AVB_TXC REFCLK | | | | | | | | |
| IP2[7:4] | AVB_MDIO | | | | | | | | |
| IP2[11:8] | AVB_MDC | | | | | | | | |
| IP2[15:12] | BS# | PWM0_A | AVB_MAGIC | VI4_CLK | | TX3_C | | VI5_CLK_B | |
| IP2[19:16] | RD# | PWM1_A | AVB_LINK | VI4_FIELD | | RX3_C | | VI5_DATA0_B | |
| IP2[23:20] | RD/WR# | SCL7_A | AVB_AVTP_MATCH | VI4_VSYNC# | TX5_B | SCK3_C | PWM5_A | | |
| IP2[27:24] | EX_WAIT0 | SDA7_A | AVB_AVTP_CAPTURE | VI4_HSYNC# | RX5_B | PWM6_A | | | |

| Register | IPSR = H'0 | IPSR = H'1 | IPSR = H'2 | IPSR = H'3 | IPSR = H'4 | IPSR = H'5 | IPSR = H'6 | IPSR = H'7 | IPSR = H'8 |
|------------|------------|--------------|---------------|---------------|--------------|----------------|-------------|------------|------------|
| IP2[31:28] | A0 | IRQ0 | PWM2_A | MSIOF3_SS1_B | VI5_CLK_A | DU_CDE | HRX3_D | | |
| IP3[3:0] | A1 | IRQ1 | PWM3_A | DU_DOT_CLKIN1 | VI5_DATA0_A | DU_DISP_CDE | SDA6_B | | |
| IP3[7:4] | A2 | IRQ2 | AVB_AVTP_PPS | VI4_CLK_ENB | VI5_DATA1_A | DU_DISP | SCL6_B | | |
| IP3[11:8] | A3 | CTS4#_A | PWM4_A | VI4_DATA12 | | DU_DOT_CLKOUT0 | HTX3_D | | |
| IP3[15:12] | A4 | RTS4#_A | MSIOF3_SYNC_B | VI4_DATA8 | PWM2_B | DU_DG4 | | | |
| IP3[19:16] | A5 | SCK4_A | MSIOF3_SCK_B | VI4_DATA9 | PWM3_B | | | | |
| IP3[23:20] | A6 | RX4_A | MSIOF3_RXD_B | VI4_DATA10 | | | | | |
| IP3[27:24] | A7 | TX4_A | MSIOF3_TXD_B | VI4_DATA11 | | | | | |
| IP3[31:28] | A8 | SDA6_A | RX3_B | HRX4_C | VI5_HSYNC#_A | DU_HSYNC | VI4_DATA0_B | | |
| IP4[3:0] | A9 | TX5_A | IRQ3 | VI4_DATA16 | VI5_VSYNC#_A | DU_DG7 | | | |
| IP4[7:4] | A10 | IRQ4 | MSIOF2_SYNC_B | VI4_DATA13 | VI5_FIEL D_A | DU_DG5 | | | |
| IP4[11:8] | A11 | SCL6_A | TX3_B | HTX4_C | | DU_VSYNC | VI4_DATA1_B | | |
| IP4[15:12] | A12 | RX5_A | MSIOF2_SS2_B | VI4_DATA17 | VI5_DATA3_A | DU_DG6 | | | |
| IP4[19:16] | A13 | SCK5_A | MSIOF2_SCK_B | VI4_DATA14 | HRX4_D | DU_DB2 | | | |
| IP4[23:20] | A14 | MSIOF1_SS1 | MSIOF2_RXD_B | VI4_DATA15 | HTX4_D | DU_DB3 | | | |
| IP4[27:24] | A15 | MSIOF1_SS2 | MSIOF2_TXD_B | VI4_DATA18 | VI5_DATA4_A | DU_DB4 | | | |
| IP4[31:28] | A16 | MSIOF1_SYNC | MSIOF2_SS1_B | VI4_DATA19 | VI5_DATA5_A | DU_DB5 | | | |
| IP5[3:0] | A17 | MSIOF1_RXD | | VI4_DATA20 | VI5_DATA6_A | DU_DB6 | | | |
| IP5[7:4] | A18 | MSIOF1_TXD | | VI4_DATA21 | VI5_DATA7_A | DU_DB0 | | HRX4_E | |
| IP5[11:8] | A19 | MSIOF1_SCK | | VI4_DATA22 | VI5_DATA2_A | DU_DB1 | | HTX4_E | |
| IP5[15:12] | CS0# | SCL5 | | | | DU_DR0 | VI4_DATA2_B | | |
| IP5[19:16] | WE0# | SDA5 | | | | DU_DR1 | VI4_DATA3_B | | |
| IP5[23:20] | D0 | MSIOF3_SCK_A | | | | DU_DR2 | CTS4#_C | | |

| Register | IPSR = H'0 | IPSR = H'1 | IPSR = H'2 | IPSR = H'3 | IPSR = H'4 | IPSR = H'5 | IPSR = H'6 | IPSR = H'7 | IPSR = H'8 |
|------------|------------|---------------|------------|--------------|--------------|---------------|-------------|-------------|------------|
| IP5[27:24] | D1 | MSIOF3_SYNC_A | SCK3_A | VI4_DATA23 | VI5_CLKENB_A | DU_DB7 | RTS4#_C | | |
| IP5[31:28] | D2 | MSIOF3_RXD_A | RX5_C | | VI5_DATA14_A | DU_DR3 | RX4_C | | |
| IP6[3:0] | D3 | MSIOF3_TXD_A | TX5_C | | VI5_DATA15_A | DU_DR4 | TX4_C | | |
| IP6[7:4] | D4 | CANFD1_TX | HCK3_B | CAN1_TX | RTS3#_A | MSIOF3_SS2_A | | VI5_DATA1_B | |
| IP6[11:8] | D5 | RX3_A | HRX3_B | | | DU_DR5 | VI4_DATA4_B | | |
| IP6[15:12] | D6 | TX3_A | HTX3_B | | | DU_DR6 | VI4_DATA5_B | | |
| IP6[19:16] | D7 | CANFD1_RX | IRQ5 | CAN1_RX | CTS3#_A | | | VI5_DATA2_B | |
| IP6[23:20] | D8 | MSIOF2_SCK_A | SCK4_B | | VI5_DATA12_A | DU_DR7 | | HCTS3#_E | |
| IP6[27:24] | D9 | MSIOF2_SYNC_A | | | VI5_DATA10_A | DU_DG0 | | HRX3_E | |
| IP6[31:28] | D10 | MSIOF2_RXD_A | | | VI5_DATA13_A | DU_DG1 | | HTX3_E | |
| IP7[3:0] | D11 | MSIOF2_TXD_A | | | VI5_DATA11_A | DU_DG2 | | HRTS3#_E | |
| IP7[7:4] | D12 | CANFD0_TX | TX4_B | CAN0_TX | VI5_DATA8_A | | | VI5_DATA3_B | |
| IP7[11:8] | D13 | CANFD0_RX | RX4_B | CAN0_RX | VI5_DATA9_A | SCL7_B | | VI5_DATA4_B | |
| IP7[15:12] | D14 | CAN_CLK | HRX3_A | MSIOF2_SS2_A | | SDA7_B | | VI5_DATA5_B | |
| IP7[19:16] | D15 | MSIOF2_SS1_A | HTX3_A | MSIOF3_SS1_A | | DU_DG3 | | | |
| IP7[23:20] | SCL4 | CS1#/A26 | | | | DU_DOT_CLKIN0 | VI4_DATA6_B | VI5_DATA6_B | |
| IP7[27:24] | SDA4 | WE1# | | | | | VI4_DATA7_B | VI5_DATA7_B | |
| IP7[31:28] | SD0_CLK | NFDATA8 | SCL1_C | HCK1_B | SDA2_E | | | | |
| IP8[3:0] | SD0_CMD | NFDATA9 | | HRX1_B | | | | | |
| IP8[7:4] | SD0_DAT0 | NFDATA10 | | HTX1_B | | | | | |
| IP8[11:8] | SD0_DAT1 | NFDATA11 | SDA2_C | HCTS1#_B | | | | | |
| IP8[15:12] | SD0_DAT2 | NFDATA12 | SCL2_C | HRTS1#_B | | | | | |
| IP8[19:16] | SD0_DAT3 | NFDATA13 | SDA1_C | SCL2_E | | | | | |

| Register | IPSR = H'0 | IPSR = H'1 | IPSR = H'2 | IPSR = H'3 | IPSR = H'4 | IPSR = H'5 | IPSR = H'6 | IPSR = H'7 | IPSR = H'8 |
|-------------|---------------|----------------|-------------------------|---------------|----------------|---------------|----------------|---------------|---------------|
| IP8[23:20] | SD1_CLK | NFDATA 14_B | | | | | | | |
| IP8[27:24] | SD1_ CMD | NFDATA 15_B | | | | | | | |
| IP8[31:28] | SD1_ DAT0 | NFWP#_ B | | | | | | | |
| IP9[3:0] | SD1_ DAT1 | NFCE#_B | | | | | | | |
| IP9[7:4] | SD1_ DAT2 | NFALE_B | | | | | | | |
| IP9[11:8] | SD1_ DAT3 | NFRB#_B | | | | | | | |
| IP9[15:12] | SD3_CLK | NFWE# | | | | | | | |
| IP9[19:16] | SD3_ CMD | NFRE# | | | | | | | |
| IP9[23:20] | SD3_ DAT0 | NFDATA0 | | | | | | | |
| IP9[27:24] | SD3_ DAT1 | NFDATA1 | | | | | | | |
| IP9[31:28] | SD3_ DAT2 | NFDATA2 | | | | | | | |
| IP10[3:0] | SD3_ DAT3 | NFDATA3 | | | | | | | |
| IP10[7:4] | SD3_ DAT4 | NFDATA4 | | | | | | | |
| IP10[11:8] | SD3_ DAT5 | NFDATA5 | | | | | | | |
| IP10[15:12] | SD3_ DAT6 | NFDATA6 | | | | | | | |
| IP10[19:16] | SD3_ DAT7 | NFDATA7 | | | | | | | |
| IP10[23:20] | SD3_DS | NFCLE | | | | | | | |
| IP10[27:24] | SD0_CD | NFALE_A | SD3_CD | | SCL2_B | TCLK1_A | SSI_ SCK2_B | | |
| IP10[31:28] | SD0_WP | NFRB#_A | SD3_WP | | SDA2_B | TCLK2_A | SSI_ WS2_B | | |
| IP11[3:0] | SD1_CD | NFCE#_A | SSI_ SCK1 | | | | | | |
| IP11[7:4] | SD1_WP | NFWP#_ A | SSI_WS1 | | | | | | |
| IP11[11:8] | RX0_A | HRX1_A | SSI_ SCK2_A | | | | | | |
| IP11[15:12] | TX0_A | HTX1_A | SSI_WS2 _A | | | | | | |
| IP11[19:16] | CTS0#_A | NFDATA 14_A | AUDIO_ CLKOUT_ A | | SCIF_ CLK_A | | | | |
| IP11[23:20] | RTS0#_A | NFDATA 15_A | AUDIO_ CLKOUT1 _A | | SCL2_A | | | | |

| Register | IPSR = H'0 | IPSR = H'1 | IPSR = H'2 | IPSR = H'3 | IPSR = H'4 | IPSR = H'5 | IPSR = H'6 | IPSR = H'7 | IPSR = H'8 |
|-------------|--------------|-----------------|-----------------|-----------------|------------|------------|------------|------------|------------|
| IP11[27:24] | SCK0_A | HSCK1_A | USB3HS0_ID | RTS1# | SDA2_A | | | | USB0_ID |
| IP11[31:28] | RX1 | HRX2_B | SSI_SCK9_B | AUDIO_CLKOUT1_B | | | | | |
| IP12[3:0] | TX1 | HTX2_B | SSI_WS9_B | AUDIO_CLKOUT3_B | | | | | |
| IP12[7:4] | SCK2_A | HSCK0_A | AUDIO_CLKB_A | CTS1# | | | SCIF_CLK_B | | |
| IP12[11:8] | TX2_A | HRX0_A | AUDIO_CLKOUT2_A | | SCL1_A | | | | |
| IP12[15:12] | RX2_A | HTX0_A | AUDIO_CLKOUT3_A | | SDA1_A | | | | |
| IP12[19:16] | MSIOF0_SCK | | SSI_SCK78 | | | | | | |
| IP12[23:20] | MSIOF0_RXD | | SSI_WS78 | | | TX2_B | | | |
| IP12[27:24] | MSIOF0_TXD | | SSI_SDATA7 | | | RX2_B | | | |
| IP12[31:28] | MSIOF0_SYNC | AUDIO_CLKOUT_B | SSI_SDATA8 | | | | | | |
| IP13[3:0] | MSIOF0_SS1 | HRX2_A | SSI_SCK4 | HCTS0#_A | | | | | |
| IP13[7:4] | MSIOF0_SS2 | HTX2_A | SSI_WS4 | HRTS0#_A | | | | | |
| IP13[11:8] | SSI_SDATA9 | | AUDIO_CLKC_A | SCK1 | | | | | |
| IP13[15:12] | MLB_CLK | RX0_B | | | SCL1_B | TCLK1_B | | | |
| IP13[19:16] | MLB_SIG | SCK0_B | | | SDA1_B | TCLK2_B | | | |
| IP13[23:20] | MLB_DAT | TX0_B | | | | | | | |
| IP13[27:24] | SSI_SCK01239 | | | | | | | | |
| IP13[31:28] | SSI_WS01239 | | | | | | | | |
| IP14[3:0] | SSI_SDATA0 | | | | | | | | |
| IP14[7:4] | SSI_SDATA1 | AUDIO_CLKC_B | | | | | PWM0_B | | |
| IP14[11:8] | SSI_SDATA2 | AUDIO_CLKOUT2_B | SSI_SCK9_A | | | | PWM1_B | | |
| IP14[15:12] | SSI_SCK349 | | | | | | PWM2_C | | |
| IP14[19:16] | SSI_WS349 | | | | | | PWM3_C | | |

| Register | IPSR = H'0 | IPSR = H'1 | IPSR = H'2 | IPSR = H'3 | IPSR = H'4 | IPSR = H'5 | IPSR = H'6 | IPSR = H'7 | IPSR = H'8 |
|-------------|---------------|-------------------|-------------------|---------------|---------------|---------------|---------------|---------------|---------------|
| IP14[23:20] | SSI_ SDATA3 | AUDIO_ CLKOUT1_ C | AUDIO_ CLKB_ B | | | | PWM4_ B | | |
| IP14[27:24] | SSI_ SDATA4 | | SSI_ WS9_ A | | | | PWM5_ B | | |
| IP14[31:28] | SSI_ SCK5 | HRX0_ B | | USB0_ PWEN_ B | SCL2_ D | | PWM6_ B | | |
| IP15[3:0] | SSI_ WS5 | HTX0_ B | | USB0_ OVC_ B | SDA2_ D | | | | |
| IP15[7:4] | SSI_ SDATA5 | HSCK0_ B | AUDIO_ CLKB_ C | TPU0TO0 | | | | | |
| IP15[11:8] | SSI_ SCK6 | HSCK2_ A | AUDIO_ CLKC_ C | TPU0TO1 | | | | | |
| IP15[15:12] | SSI_ WS6 | HCTS2#_ A | AUDIO_ CLKOUT2_ C | TPU0TO2 | SDA1_ D | | | | |
| IP15[19:16] | SSI_ SDATA6 | HRTS2#_ A | AUDIO_ CLKOUT3_ C | TPU0TO3 | SCL1_ D | | | | |
| IP15[23:20] | AUDIO_ CLKA | | | | | | | | |
| IP15[27:24] | USB30_ PWEN | USB0_ PWEN_ A | | | | | | | |
| IP15[31:28] | USB30_ OVC | USB0_ OVC_ A | | | | | | | |

9.2.4 DRV Control Register 0-24 (DRVCTRL8-10)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

Function: DRVCTRL8-10 controls the driving abilities of pins. This setting is related to only the output.

| Output buffer | | |
|---------------|------|------------------|
| DRV1 | DRV2 | Drive capability |
| L | L | 1/4 |
| H | L | 2/4 |
| L | H | 3/4 |
| H | H | Full |

The value of these bits must be B'11 in the case of 2bit (DRV2, DRV1).

| | | | | | | | | | | | | | | | | |
|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* |
| | [31] | [30] | [29] | [28] | [27] | [26] | [25] | [24] | [23] | [22] | [21] | [20] | [19] | [18] | [17] | [16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* | DRVn* |
| | [15] | [14] | [13] | [12] | [11] | [10] | [9] | [8] | [7] | [6] | [5] | [4] | [3] | [2] | [1] | [0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * n = 8-10

Note: To enable this register to be set, appropriately set the multiplexed pin setting mask register (PMMR) immediately before setting this register..

Table 9.6 Configuration of Registers in DRVCTRL8-10

| DRVCTRL8 | Initial value | DRVCTRL9 | Initial value | DRVCTRL10 | Initial value |
|----------|---------------|----------|---------------|-----------|---------------|
| bit31 | — | 1 | — | 0 | — |
| bit30 | — | 0 | SD1_CMD | 0 | SD3_DAT3 |
| bit29 | — | 0 | SD1_CMD | 1 | SD3_DAT3 |
| bit28 | — | 1 | — | 0 | — |
| bit27 | — | 0 | SD1_DAT0 | 0 | SD3_DAT4 |
| bit26 | — | 0 | SD1_DAT0 | 1 | SD3_DAT4 |
| bit25 | — | 1 | — | 0 | — |
| bit24 | — | 0 | SD1_DAT1 | 0 | SD3_DAT5 |
| bit23 | — | 0 | SD1_DAT1 | 1 | SD3_DAT5 |
| bit22 | — | 1 | — | 0 | — |
| bit21 | — | 0 | SD1_DAT2 | 0 | SD3_DAT6 |
| bit20 | — | 0 | SD1_DAT2 | 1 | SD3_DAT6 |
| bit19 | SD0_CLK | 0 | — | 0 | — |
| bit18 | SD0_CLK | 1 | SD1_DAT3 | 0 | SD3_DAT7 |
| bit17 | — | 0 | SD1_DAT3 | 1 | SD3_DAT7 |

| | DRVCTRL8 | Initial value | DRVCTRL9 | Initial value | DRVCTRL10 | Initial value |
|-------|----------|---------------|----------|---------------|-----------|---------------|
| bit16 | SD0_CMD | 0 | — | 0 | — | 0 |
| bit15 | SD0_CMD | 1 | SD3_CLK | 0 | SD3_DS | 0 |
| bit14 | — | 0 | SD3_CLK | 1 | SD3_DS | 1 |
| bit13 | SD0_DAT0 | 0 | — | 0 | — | 0 |
| bit12 | SD0_DAT0 | 1 | SD3_CMD | 0 | — | 1 |
| bit11 | — | 0 | SD3_CMD | 1 | — | 0 |
| bit10 | SD0_DAT1 | 0 | — | 0 | — | 0 |
| bit9 | SD0_DAT1 | 1 | SD3_DAT0 | 0 | — | 1 |
| bit8 | — | 0 | SD3_DAT0 | 1 | — | 0 |
| bit7 | SD0_DAT2 | 0 | — | 0 | — | 0 |
| bit6 | SD0_DAT2 | 1 | SD3_DAT1 | 0 | — | 1 |
| bit5 | — | 0 | SD3_DAT1 | 1 | — | 0 |
| bit4 | SD0_DAT3 | 0 | — | 0 | — | 1 |
| bit3 | SD0_DAT3 | 1 | SD3_DAT2 | 0 | — | 1 |
| bit2 | — | 0 | SD3_DAT2 | 1 | — | 0 |
| bit1 | SD1_CLK | 0 | — | 0 | — | 0 |
| bit0 | SD1_CLK | 1 | — | 0 | — | 0 |

9.2.5 POC Control Register 0 (POCCTRL0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

Function: Each bit in POCCTRL0 must be set according to IO voltage level that is supplied to the pin.

It must be set to match the voltage of VDDQ_MMC.

0: 1.8v 1: 3.3v

| Supply voltage | Setting register | Usage |
|----------------|------------------|---------------------|
| 1.8v | 0 | Possible |
| 1.8v | 1 | Do not use |
| 3.3v | 0 | Do not set (broken) |
| 3.3v | 1 | Possible |

| | | | | | | | | | | | | | | | | |
|------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | POCn* [31] | POCn* [30] | POCn* [29] | POCn* [28] | POCn* [27] | POCn* [26] | POCn* [25] | POCn* [24] | POCn* [23] | POCn* [22] | POCn* [21] | POCn* [20] | POCn* [19] | POCn* [18] | POCn* [17] | POCn* [16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | POCn* [15] | POCn* [14] | POCn* [13] | POCn* [12] | POCn* [11] | POCn* [10] | POCn* [9] | POCn* [8] | POCn* [7] | POCn* [6] | POCn* [5] | POCn* [4] | POCn* [3] | POCn* [2] | POCn* [1] | POCn* [0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * n = 0

Note: To enable this register to be set, appropriately set the multiplexed pin setting mask register (PMMR) immediately before setting this register.

Table 9.7 Configuration of Registers in POCCTRL0

| | POCCTRL0 | Initial value |
|-------|--------------|---------------|
| bit31 | — | 0 |
| bit30 | — | 0 |
| bit29 | POC_SD3_DS | 1 |
| bit28 | POC_SD3_DAT7 | 1 |
| bit27 | POC_SD3_DAT6 | 1 |
| bit26 | POC_SD3_DAT5 | 1 |
| bit25 | POC_SD3_DAT4 | 1 |
| bit24 | POC_SD3_DAT3 | 1 |
| bit23 | POC_SD3_DAT2 | 1 |
| bit22 | POC_SD3_DAT1 | 1 |
| bit21 | POC_SD3_DAT0 | 1 |
| bit20 | POC_SD3_CMD | 1 |
| bit19 | POC_SD3_CLK | 1 |
| bit18 | — | 1 |
| bit17 | — | 1 |
| bit16 | — | 1 |

| | POCTRL0 | Initial value |
|--------------|----------------|----------------------|
| bit15 | — | 1 |
| bit14 | — | 1 |
| bit13 | — | 1 |
| bit12 | — | 1 |
| bit11 | POC_SD1_DAT3 | 1 |
| bit10 | POC_SD1_DAT2 | 1 |
| bit9 | POC_SD1_DAT1 | 1 |
| bit8 | POC_SD1_DAT0 | 1 |
| bit7 | POC_SD1_CMD | 1 |
| bit6 | POC_SD1_CLK | 1 |
| bit5 | POC_SD0_DAT3 | 1 |
| bit4 | POC_SD0_DAT2 | 1 |
| bit3 | POC_SD0_DAT1 | 1 |
| bit2 | POC_SD0_DAT0 | 1 |
| bit1 | POC_SD0_CMD | 1 |
| bit0 | POC_SD0_CLK | 1 |

9.2.6 POC Control Register 2 (POCCTRL2)

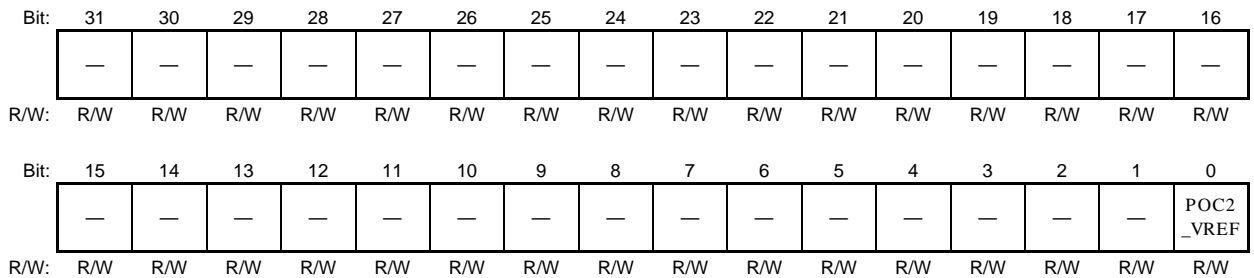
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

Function: Each bit in POCCTRL2 must be set according to IO voltage level that is supplied to the pin.

It must be set to match the voltage of VDDQ25_AVB0.

0: 2.5v 1: 3.3v

| Supply voltage | Setting register | Usage |
|----------------|------------------|---------------------|
| 2.5v | 0 | Possible |
| 2.5v | 1 | Do not use |
| 3.3v | 0 | Do not set (broken) |
| 3.3v | 1 | Possible |



Note: To enable this register to be set, appropriately set the multiplexed pin setting mask register (PMMR) immediately before setting this register.

9.2.7 TDSEL Control Register 0 (TDSELCTRL0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

Function: TDSELCTRL0 controls the driving abilities of pins in use for the SDHI.

This function is delay adjustment of the SDHI clock return path for the LSI inside.

| TDSEL1 | TDSEL0 | TDOUT |
|--------|--------|-------------------------|
| | | Target delay |
| L | L | 10pF (@1.8 V operation) |
| H | L | 20pF (@1.8 V operation) |
| L | H | 30pF (@1.8 V operation) |
| H | H | 40pF (@1.8 V operation) |

The value of these bits must be 00.

| | | | | | | | | | | | | | | | | |
|------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TD SEL0[31] | TD SEL0[30] | TD SEL0[29] | TD SEL0[28] | TD SEL0[27] | TD SEL0[26] | TD SEL0[25] | TD SEL0[24] | TD SEL0[23] | TD SEL0[22] | TD SEL0[21] | TD SEL0[20] | TD SEL0[19] | TD SEL0[18] | TD SEL0[17] | TD SEL0[16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TD SEL0[15] | TD SEL0[14] | TD SEL0[13] | TD SEL0[12] | TD SEL0[11] | TD SEL0[10] | TD SEL0[9] | TD SEL0[8] | TD SEL0[7] | TD SEL0[6] | TD SEL0[5] | TD SEL0[4] | TD SEL0[3] | TD SEL0[2] | TD SEL0[1] | TD SEL0[0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: To enable this register to be set, appropriately set the multiplexed pin setting mask register (PMMR) immediately before setting this register.

Table 9.8 Configuration of Registers in TDSELCTRL0

| | TDSELCTRL0 | Initial value |
|-------|------------|---------------|
| bit31 | — | 0 |
| bit30 | — | 0 |
| bit29 | — | 0 |
| bit28 | — | 0 |
| bit27 | — | 0 |
| bit26 | — | 0 |
| bit25 | — | 0 |
| bit24 | — | 0 |
| bit23 | — | 0 |
| bit22 | — | 0 |
| bit21 | — | 0 |
| bit20 | — | 0 |
| bit19 | — | 0 |
| bit18 | — | 0 |
| bit17 | — | 0 |
| bit16 | — | 0 |

| | TDSELCTRL0 | Initial value |
|--------------|-------------------|----------------------|
| bit15 | — | 0 |
| bit14 | — | 0 |
| bit13 | — | 0 |
| bit12 | — | 0 |
| bit11 | — | 0 |
| bit10 | — | 0 |
| bit9 | — | 0 |
| bit8 | — | 0 |
| bit7 | SD3TDSEL1 | 0 |
| bit6 | SD3TDSEL0 | 0 |
| bit5 | — | 0 |
| bit4 | — | 0 |
| bit3 | SD1TDSEL1 | 0 |
| bit2 | SD1TDSEL0 | 0 |
| bit1 | SD0TDSEL1 | 0 |
| bit0 | SD0TDSEL0 | 0 |

9.2.8 LSI pin pull-enable register 0-5 (PUEN0-5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

Function: PUEN0-5 performs on/off control of the pull resistors.

0: Pull-up/down function is disabled.

1: Pull-up/down function is enabled.

| | | | | | | | | | | | | | | | | |
|------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* |
| | [31] | [30] | [29] | [28] | [27] | [26] | [25] | [24] | [23] | [22] | [21] | [20] | [19] | [18] | [17] | [16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* | PUENn* |
| | [15] | [14] | [13] | [12] | [11] | [10] | [9] | [8] | [7] | [6] | [5] | [4] | [3] | [2] | [1] | [0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * n = 0-5

When the all I2C pins are selected, the pull-up/down function of these pins is disabled respectively even if the pull-up/down function is set to enable.

Table 9.9 Configuration of Registers in PUEN0-5

| | Initial value | Initial value | Initial value | Initial value | Initial value | Initial value | Initial value | Initial value | Initial value | Initial value | |
|---------------------------|---------------|---------------|---------------|-----------------|---------------|---------------|---------------|------------------|---------------|------------------|---|
| PUEN0 | PUEN1 | PUEN2 | PUEN3 | PUEN4 | PUEN5 | | | | | | |
| bit31 PUEN_QSPI0_SPCLK | 0 | PUEN_RD_WR# | 1 | PUEN_D5 | 1 | PUEN_SD0_DAT0 | 0 | PUEN_RX1 | 1 | PUEN_USB30_PWEN | 0 |
| bit30 PUEN_QSPI0_MOSI/IO0 | 0 | PUEN_EX_WAIT0 | 1 | PUEN_D6 | 0 | PUEN_SD0_DAT1 | 0 | PUEN_TX1 | 1 | PUEN_USB30_OVC * | 1 |
| bit29 PUEN_QSPI0_MISO/IO1 | 0 | — | 0 | PUEN_D7 | 1 | PUEN_SD0_DAT2 | 0 | PUEN_SCK2_A | 1 | — | 0 |
| bit28 PUEN_QSPI0_IO2 | 0 | — | 0 | PUEN_D8 | 0 | PUEN_SD0_DAT3 | 0 | PUEN_TX2_A | 1 | — | 0 |
| bit27 PUEN_QSPI0_IO3 | 0 | PUEN_A0 | 1 | PUEN_D9 | 0 | PUEN_SD1_CLK | 0 | PUEN_RX2_A | 1 | — | 0 |
| bit26 PUEN_QSPI0_SSL | 0 | PUEN_A1 | 1 | PUEN_D10 | 1 | PUEN_SD1_CMD | 0 | PUEN_MSIOF0_SCK | 1 | — | 0 |
| bit25 PUEN_QSPI1_SPCLK | 0 | PUEN_A2 | 1 | PUEN_D11 | 0 | PUEN_SD1_DAT0 | 0 | PUEN_MSIOF0_RXD | 1 | — | 0 |
| bit24 PUEN_QSPI1_MOSI/IO0 | 0 | PUEN_A3 | 0 | PUEN_D12 | 0 | PUEN_SD1_DAT1 | 0 | PUEN_MSIOF0_TXD | 1 | — | 0 |
| bit23 PUEN_QSPI1_MISO/IO1 | 0 | PUEN_A4 | 0 | PUEN_D13 | 1 | PUEN_SD1_DAT2 | 0 | PUEN_MSIOF0_SYNC | 1 | — | 0 |
| bit22 PUEN_QSPI1_IO2 | 0 | PUEN_A5 | 0 | PUEN_D14 | 1 | PUEN_SD1_DAT3 | 0 | PUEN_MSIOF0_SS1 | 1 | — | 0 |
| bit21 PUEN_QSPI1_IO3 | 0 | PUEN_A6 | 1 | PUEN_D15 | 0 | PUEN_SD3_CLK | 0 | PUEN_MSIOF0_SS2 | 1 | — | 0 |
| bit20 PUEN_QSPI1_SSL | 0 | PUEN_A7 | 0 | PUEN_PRESETOUT# | 0 | PUEN_SD3_CMD | 0 | PUEN_SSL_SDATA9 | 1 | — | 0 |

| | | Initial value | | Initial value | | Initial value | | Initial value | | Initial value | | Initial value |
|-------|--------------------|---------------|-------------|---------------|---------------|---------------|---------------|---------------|-------------------|---------------|-------|---------------|
| | PUEN0 | | PUEN1 | | PUEN2 | | PUEN3 | | PUEN4 | | PUEN5 | |
| bit19 | PUEN_RPC_INT# | 1 | PUEN_A8 | 1 | — | 0 | PUEN_SD3_DAT0 | 0 | PUEN_MLB_CLK | 0 | — | 0 |
| bit18 | PUEN_RPC_RESET# | 1 | PUEN_A9 | 0 | — | 0 | PUEN_SD3_DAT1 | 0 | PUEN_MLB_SIG | 0 | — | 0 |
| bit17 | PUEN_AVB_RX_CTL | 1 | PUEN_A10 | 0 | PUEN_SCL4 | 1 | PUEN_SD3_DAT2 | 0 | PUEN_MLB_DAT | 0 | — | 0 |
| bit16 | PUEN_AVB_RXC | 1 | PUEN_A11 | 1 | PUEN_SDA4 | 1 | PUEN_SD3_DAT3 | 0 | PUEN_MLB_REF | 0 | — | 0 |
| bit15 | PUEN_AVB_RD0 | 1 | PUEN_A12 | 1 | PUEN_FSCLKST# | 1 | PUEN_SD3_DAT4 | 0 | PUEN_SSI_SCK01239 | 1 | — | 0 |
| bit14 | PUEN_AVB_RD1 | 1 | PUEN_A13 | 0 | — | 0 | PUEN_SD3_DAT5 | 0 | PUEN_SSI_WS01239 | 1 | — | 0 |
| bit13 | PUEN_AVB_RD2 | 1 | PUEN_A14 | 0 | — | 0 | PUEN_SD3_DAT6 | 0 | PUEN_SSI_SDATA0 | 1 | — | 0 |
| bit12 | PUEN_AVB_RD3 | 1 | PUEN_A15 | 0 | — | 0 | PUEN_SD3_DAT7 | 0 | PUEN_SSI_SDATA1 | 1 | — | 0 |
| bit11 | PUEN_AVB_TX_CTL | 1 | PUEN_A16 | 0 | — | 0 | PUEN_SD3_DS | 0 | PUEN_SSI_SDATA2 | 1 | — | 0 |
| bit10 | PUEN_AVB_TXC | 1 | PUEN_A17 | 1 | — | 0 | PUEN_SD0_CD | 1 | PUEN_SSI_SCK349 | 1 | — | 0 |
| bit9 | PUEN_AVB_TD0 | 1 | PUEN_A18 | 0 | — | 0 | PUEN_SD0_WP | 1 | PUEN_SSI_WS349 | 1 | — | 0 |
| bit8 | PUEN_AVB_TD1 | 1 | PUEN_A19 | 0 | — | 0 | PUEN_SD1_CD | 1 | PUEN_SSI_SDATA3 | 1 | — | 0 |
| bit7 | PUEN_AVB_TD2 | 1 | PUEN_CLKOUT | 0 | PUEN_TRST# | 1 | PUEN_SD1_WP | 1 | PUEN_SSI_SDATA4 | 1 | — | 0 |
| bit6 | PUEN_AVB_TD3 | 1 | PUEN_CS0# | 1 | PUEN_TCK | 1 | — | 0 | PUEN_SSI_SCK5 | 1 | — | 0 |
| bit5 | PUEN_AVB_TXCREFLCK | 1 | PUEN_WE0# | 1 | PUEN_TMS | 1 | — | 0 | PUEN_SSI_WS5 | 1 | — | 0 |
| bit4 | PUEN_AVB_MDIO | 0 | PUEN_D0 | 0 | PUEN_TDI | 1 | PUEN_RX0_A | 1 | PUEN_SSI_SDATA5 | 1 | — | 0 |
| bit3 | PUEN_AVB_MDC | 0 | PUEN_D1 | 0 | — | 0 | PUEN_TX0_A | 1 | PUEN_SSI_SCK6 | 1 | — | 0 |
| bit2 | PUEN_AVB_PHY_INT | 1 | PUEN_D2 | 1 | PUEN_ASEBRK | 1 | PUEN_CTS0_N_A | 1 | PUEN_SSI_WS6 | 1 | — | 0 |
| bit1 | PUEN_BS# | 0 | PUEN_D3 | 0 | PUEN_SD0_CLK | 0 | PUEN_RTS0_N_A | 1 | PUEN_SSI_SDATA6 | 1 | — | 0 |
| bit0 | PUEN_RD# | 0 | PUEN_D4 | 0 | PUEN_SD0_CMD | 0 | PUEN_SCK0_A | 1 | PUEN_AUDIO_CLKA | 1 | — | 0 |

Note: * pull-up only

9.2.9 LSI pin pull-up/down control Register 0-5 (PUD0-5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

Function: PUD0-5 performs pull-up/pull-down control of the pull resistors.

0: Pull-down is enabled.

1: Pull-up is enabled.

| | | | | | | | | | | | | | | | | |
|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* |
| | [31] | [30] | [29] | [28] | [27] | [26] | [25] | [24] | [23] | [22] | [21] | [20] | [19] | [18] | [17] | [16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* | PUDn* |
| | [15] | [14] | [13] | [12] | [11] | [10] | [9] | [8] | [7] | [6] | [5] | [4] | [3] | [2] | [1] | [0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * n = 0-5

Table 9.10 Configuration of Registers in PUD0-5

| | PUD0 | Initial value | PUD1 | Initial value | PUD2 | Initial value | PUD3 | Initial value | PUD4 | Initial value | PUD5 | Initial value |
|--------------|---------------------|---------------|--------------|---------------|------------------|---------------|---------------|---------------|------------------|---------------|-----------------|---------------|
| bit31 | PUD_QSPI0_ SPCLK | 0 | PUD_RD_WR # | 1 | PUD_D5 | 1 | PUD_SD0_ DAT0 | 0 | PUD_RX1 | 1 | PUD_USB30_ PWEN | 0 |
| bit30 | PUD_QSPI0_ MOSI/IO0 | 0 | PUD_EX_WAIT0 | 1 | PUD_D6 | 0 | PUD_SD0_ DAT1 | 0 | PUD_TX1 | 1 | — * | 1 |
| bit29 | PUD_QSPI0_ MISO/IO1 | 0 | — | 0 | PUD_D7 | 1 | PUD_SD0_ DAT2 | 0 | PUD_SCK2_ A | 1 | — | 0 |
| bit28 | PUD_QSPI0_ IO2 | 0 | — | 0 | PUD_D8 | 0 | PUD_SD0_ DAT3 | 0 | PUD_TX2_ A | 1 | — | 0 |
| bit27 | PUD_QSPI0_ IO3 | 0 | PUD_A0 | 1 | PUD_D9 | 0 | PUD_SD1_ CLK | 0 | PUD_RX2_ A | 1 | — | 0 |
| bit26 | PUD_QSPI0_ SSL | 0 | PUD_A1 | 1 | PUD_D10 | 1 | PUD_SD1_ CMD | 0 | PUD_MSIOF0_ SCK | 1 | — | 0 |
| bit25 | PUD_QSPI1_ SPCLK | 0 | PUD_A2 | 1 | PUD_D11 | 0 | PUD_SD1_ DAT0 | 0 | PUD_MSIOF0_ RXD | 1 | — | 0 |
| bit24 | PUD_QSPI1_ MOSI/IO0 | 0 | PUD_A3 | 0 | PUD_D12 | 0 | PUD_SD1_ DAT1 | 0 | PUD_MSIOF0_ TXD | 1 | — | 0 |
| bit23 | PUD_QSPI1_ MISO/IO1 | 0 | PUD_A4 | 0 | PUD_D13 | 1 | PUD_SD1_ DAT2 | 0 | PUD_MSIOF0_ SYNC | 1 | — | 0 |
| bit22 | PUD_QSPI1_ IO2 | 0 | PUD_A5 | 0 | PUD_D14 | 1 | PUD_SD1_ DAT3 | 0 | PUD_MSIOF0_ SS1 | 1 | — | 0 |
| bit21 | PUD_QSPI1_ IO3 | 0 | PUD_A6 | 1 | PUD_D15 | 0 | PUD_SD3_ CLK | 0 | PUD_MSIOF0_ SS2 | 1 | — | 0 |
| bit20 | PUD_QSPI1_ SSL | 0 | PUD_A7 | 0 | PUD_ PRESETOUT # | 0 | PUD_SD3_ CMD | 0 | PUD_SSI_ SDATA9 | 1 | — | 0 |
| bit19 | PUD_RPC_ INT# | 1 | PUD_A8 | 1 | — | 0 | PUD_SD3_ DAT0 | 0 | PUD_MLB_ CLK | 0 | — | 0 |

| | PUD0 | Initial value | PUD1 | Initial value | PUD2 | Initial value | PUD3 | Initial value | PUD4 | Initial value | PUD5 | Initial value |
|-------|------------------|---------------|------------|---------------|--------------|---------------|--------------|---------------|------------------|---------------|------|---------------|
| bit18 | PUD_RPC_RESET# | 0 | PUD_A9 | 0 | — | 0 | PUD_SD3_DAT1 | 0 | PUD_MLB_SIG | 0 | — | 0 |
| bit17 | PUD_AVB_RX_CTL | 0 | PUD_A10 | 0 | PUD_SCL4 | 1 | PUD_SD3_DAT2 | 0 | PUD_MLB_DAT | 0 | — | 0 |
| bit16 | PUD_AVB_RXC | 0 | PUD_A11 | 1 | PUD_SDA4 | 1 | PUD_SD3_DAT3 | 0 | PUD_MLB_REF | 0 | — | 0 |
| bit15 | PUD_AVB_RD0 | 0 | PUD_A12 | 1 | PUD_FSCLKST# | 1 | PUD_SD3_DAT4 | 0 | PUD_SSI_SCK01239 | 1 | — | 0 |
| bit14 | PUD_AVB_RD1 | 0 | PUD_A13 | 0 | — | 0 | PUD_SD3_DAT5 | 0 | PUD_SSI_WS01239 | 1 | — | 0 |
| bit13 | PUD_AVB_RD2 | 0 | PUD_A14 | 0 | — | 0 | PUD_SD3_DAT6 | 0 | PUD_SSI_SDATA0 | 1 | — | 0 |
| bit12 | PUD_AVB_RD3 | 0 | PUD_A15 | 0 | — | 0 | PUD_SD3_DAT7 | 0 | PUD_SSI_SDATA1 | 1 | — | 0 |
| bit11 | PUD_AVB_TX_CTL | 0 | PUD_A16 | 0 | — | 0 | PUD_SD3_DS | 0 | PUD_SSI_SDATA2 | 1 | — | 0 |
| bit10 | PUD_AVB_TXC | 0 | PUD_A17 | 1 | — | 0 | PUD_SD0_CD | 1 | PUD_SSI_SCK349 | 1 | — | 0 |
| bit9 | PUD_AVB_TD0 | 0 | PUD_A18 | 0 | — | 0 | PUD_SD0_WP | 1 | PUD_SSI_WS349 | 1 | — | 0 |
| bit8 | PUD_AVB_TD1 | 0 | PUD_A19 | 0 | — | 0 | PUD_SD1_CD | 1 | PUD_SSI_SDATA3 | 1 | — | 0 |
| bit7 | PUD_AVB_TD2 | 0 | PUD_CLKOUT | 0 | — | 1 | PUD_SD1_WP | 1 | PUD_SSI_SDATA4 | 1 | — | 0 |
| bit6 | PUD_AVB_TD3 | 0 | PUD_CS0# | 1 | — | 1 | — | 0 | PUD_SSI_SCK5 | 1 | — | 0 |
| bit5 | PUD_AVB_TXCREFLK | 0 | PUD_WE0# | 1 | — | 1 | — | 0 | PUD_SSI_WS5 | 1 | — | 0 |
| bit4 | PUD_AVB_MDIO | 0 | PUD_D0 | 0 | — | 1 | PUD_RX0_A | 1 | PUD_SSI_SDATA5 | 1 | — | 0 |
| bit3 | PUD_AVB_MDC | 0 | PUD_D1 | 0 | — | 0 | PUD_TX0_A | 1 | PUD_SSI_SCK6 | 1 | — | 0 |
| bit2 | PUD_AVB_PHY_INT | 0 | PUD_D2 | 1 | PUD_ASEBRK | 0 | PUD_CTS0_N_A | 1 | PUD_SSI_WS6 | 1 | — | 0 |
| bit1 | PUD_BS# | 0 | PUD_D3 | 0 | PUD_SD0_CLK | 0 | PUD_RTS0_N_A | 1 | PUD_SSI_SDATA6 | 1 | — | 0 |
| bit0 | PUD_RD# | 0 | PUD_D4 | 0 | PUD_SD0_CMD | 0 | PUD_SCK0_A | 1 | PUD_AUDIO_CLKA | 1 | — | 0 |

Note: * Setting prohibited

9.2.10 Module Select Register 0-1 (MOD_SEL0-1)

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| — | — | — | √ |

Function: MOD_SEL0-1 selects the group for multiple LSI pins with multiplexed pin functions.

Each input or input/output signals are assigned to two or more groups of pins. Select one of these groups when using these signals. Do not use the module pins in the unselected group; if a module pin in the unselected group is used, correct operation is not guaranteed. Group is “Set value” of Table 9.12. and Table 9.13.

For some modules, however, although the output signals are assigned to two or more groups of pins, there is no bit for selecting the group. Select one of these pins for each output signal through the corresponding peripheral function select register. Also note that each pin can only be used in combination with the other input or input/output pins of the same group. Correct operation is not guaranteed when a pin is used in combination with pins from other groups. If this is not the case, correct operation is not guaranteed.

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| | MOD_SE Ln*[31] | MOD_SE Ln*[30] | MOD_SE Ln*[29] | MOD_SE Ln*[28] | MOD_SE Ln*[27] | MOD_SE Ln*[26] | MOD_SE Ln*[25] | MOD_SE Ln*[24] | MOD_SE Ln*[23] | MOD_SE Ln*[22] | MOD_SE Ln*[21] | MOD_SE Ln*[20] | MOD_SE Ln*[19] | MOD_SE Ln*[18] | MOD_SE Ln*[17] | MOD_SE Ln*[16] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| | MOD_SE Ln*[15] | MOD_SE Ln*[14] | MOD_SE Ln*[13] | MOD_SE Ln*[12] | MOD_SE Ln*[11] | MOD_SE Ln*[10] | MOD_SE Ln*[9] | MOD_SE Ln*[8] | MOD_SE Ln*[7] | MOD_SE Ln*[6] | MOD_SE Ln*[5] | MOD_SE Ln*[4] | MOD_SE Ln*[3] | MOD_SE Ln*[2] | MOD_SE Ln*[1] | MOD_SE Ln*[0] |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * n = 0-1

Note: To enable this register to be set, appropriately set the multiplexed pin setting mask register (PMMR) immediately before setting this register.

Table 9.11 Configuration of Registers in MOD_SEL0-1

| | MOD_SEL0 | Initial value | MOD_SEL1 | Initial value |
|-------|------------|---------------|----------------|---------------|
| bit31 | — | 0 | — | 0 |
| bit30 | sel_adgb_0 | 0 | sel_ssi2 | 0 |
| bit29 | sel_adgb_1 | 0 | sel_timer_tmu | 0 |
| bit28 | — | 0 | sel_usb_20_ch0 | 0 |
| bit27 | — | 0 | — | 0 |
| bit26 | — | 0 | — | 0 |
| bit25 | — | 0 | — | 0 |
| bit24 | sel_hscif0 | 0 | sel_hscif3_0 | 0 |
| bit23 | sel_hscif1 | 0 | sel_hscif3_1 | 0 |
| bit22 | sel_hscif2 | 0 | sel_hscif3_2 | 0 |
| bit21 | sel_i2c1_0 | 0 | sel_hscif4_0 | 0 |
| bit20 | sel_i2c1_1 | 0 | sel_hscif4_1 | 0 |
| bit19 | sel_i2c2_0 | 0 | sel_hscif4_2 | 0 |
| bit18 | sel_i2c2_1 | 0 | sel_i2c6 | 0 |
| bit17 | sel_i2c2_2 | 0 | sel_i2c7 | 0 |
| bit16 | sel_ndfc | 0 | sel_msiof2 | 0 |
| bit15 | sel_pwm0 | 0 | sel_msiof3 | 0 |
| bit14 | sel_pwm1 | 0 | sel_scif3_0 | 0 |

| | MOD_SEL0 | Initial value | MOD_SEL1 | Initial value |
|--------------|-----------------|----------------------|-----------------|----------------------|
| bit13 | sel_pwm2_0 | 0 | sel_scif3_1 | 0 |
| bit12 | sel_pwm2_1 | 0 | sel_scif4_0 | 0 |
| bit11 | sel_pwm3_0 | 0 | sel_scif4_1 | 0 |
| bit10 | sel_pwm3_1 | 0 | sel_scif5_0 | 0 |
| bit9 | sel_pwm4 | 0 | sel_scif5_1 | 0 |
| bit8 | sel_pwm5 | 0 | sel_vin4 | 0 |
| bit7 | sel_pwm6 | 0 | sel_vin5 | 0 |
| bit6 | — | 0 | sel_adgc_0 | 0 |
| bit5 | — | 0 | sel_adgc_1 | 0 |
| bit4 | sel_scif | 0 | sel_ssi9 | 0 |
| bit3 | sel_scif0 | 0 | — | 0 |
| bit2 | sel_scif2 | 0 | — | 0 |
| bit1 | — | 0 | — | 0 |
| bit0 | — | 0 | — | 0 |

Table 9.12 Configuration of Registers in MOD_SEL0

| Register | Set Value = H'0 | | Set Value = H'1 | | Set Value = H'2 | | Set Value = H'3 | | Set Value = H'4 | |
|--------------------------|------------------|----------------|------------------|----------------|------------------|----------------|-----------------|-----|-----------------|-----|
| | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin |
| sel_adgb_1 sel_adgb_0 | AUDIO_ CLKB_A | SCK2_A | AUDIO_ CLKB_B | SSI_ SDATA3 | AUDIO_ CLKB_C | SSI_ SDATA5 | — | — | — | — |
| sel_hscif0 | HCK0_A | SCK2_A | HRX0_B | SSI_ SCK5 | — | — | — | — | — | — |
| | HRX0_A | TX2_A | HCK0_B | SSI_ SDATA5 | — | — | — | — | — | — |
| | HCTS0#_A | MSIOF0_ SS1 | — | — | — | — | — | — | — | — |
| | HRTS0#_A | MSIOF0_ SS2 | — | — | — | — | — | — | — | — |
| sel_hscif1 | HRX1_A | RX0_A | HCK1_B | SD0_CLK | — | — | — | — | — | — |
| | HCK1_A | SCK0_A | HRX1_B | SD0_ CMD | — | — | — | — | — | — |
| | — | — | HCTS1#_B | SD0_ DAT1 | — | — | — | — | — | — |
| | — | — | HRTS1#_B | SD0_ DAT2 | — | — | — | — | — | — |
| sel_hscif2 | HRX2_A | MSIOF0_ SS1 | HRX2_B | RX1 | — | — | — | — | — | — |
| | HCK2_A | SSI_SCK 6 | — | — | — | — | — | — | — | — |
| | HCTS2#_A | SSI_WS6 | — | — | — | — | — | — | — | — |
| | HRTS2#_A | SSI_ SDATA6 | — | — | — | — | — | — | — | — |

| Register | Set Value = H'0 | | Set Value = H'1 | | Set Value = H'2 | | Set Value = H'3 | | Set Value = H'4 | |
|------------|-----------------|----------|-----------------|------------|-----------------|------------|-----------------|------------|-----------------|----------|
| | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin |
| sel_i2c1_1 | SCL1_A | TX2_A | SCL1_B | MLB_CLK | SCL1_C | SD0_CLK | SDA1_D | SSI_WS6 | — | — |
| sel_i2c1_0 | SDA1_A | RX2_A | SDA1_B | MLB_SIG | SDA1_C | SD0_DAT3 | SCL1_D | SSI_SDATA6 | — | — |
| sel_i2c2_2 | SCL2_A | RTS0#_A | SCL2_B | SD0_CD | SDA2_C | SD0_DAT1 | SCL2_D | SSI_SCK5 | SCL2_E | SD0_DAT3 |
| sel_i2c2_1 | SDA2_A | SCK0_A | SDA2_B | SD0_WP | SCL2_C | SD0_DAT2 | SDA2_D | SSI_WS5 | SDA2_E | SD0_CLK |
| sel_i2c2_0 | | | | | | | | | | |
| sel_ndfc | NFCE#_A | SD1_CD | NFCE#_B | SD1_DAT1 | — | — | — | — | — | — |
| | NFWP#_A | SD1_WP | NFWP#_B | SD1_DAT0 | — | — | — | — | — | — |
| | NFDATA_14_A | CTS0#_A | NFDATA_14_B | SD1_CLK | — | — | — | — | — | — |
| | NFDATA_15_A | RTS0#_A | NFDATA_15_B | SD1_CMD | — | — | — | — | — | — |
| | NFALE_A | SD0_CD | NFALE_B | SD1_DAT2 | — | — | — | — | — | — |
| | NFRB_N_A | SD0_WP | NFRB#_B | SD1_DAT3 | — | — | — | — | — | — |
| sel_pwm0 | PWM0_A | BS# | PWM0_B | SSI_SDATA1 | — | — | — | — | — | — |
| sel_pwm1 | PWM1_A | RD# | PWM1_B | SSI_SDATA2 | — | — | — | — | — | — |
| sel_pwm2_1 | PWM2_A | A0 | PWM2_B | A4 | PWM2_C | SSI_SCK349 | — | — | — | — |
| sel_pwm2_0 | | | | | | | | | | |
| sel_pwm3_1 | PWM3_A | A1 | PWM3_B | A5 | PWM3_C | SSI_WS349 | — | — | — | — |
| sel_pwm3_0 | | | | | | | | | | |
| sel_pwm4 | PWM4_A | A3 | PWM4_B | SSI_SDATA3 | — | — | — | — | — | — |
| sel_pwm5 | PWM5_A | RD_WR# | PWM5_B | SSI_SDATA4 | — | — | — | — | — | — |
| sel_pwm6 | PWM6_A | EX_WAIT0 | PWM6_B | SSI_SCK5 | — | — | — | — | — | — |
| sel_scif | SCIF_CLK_A | CTS0#_A | SCIF_CLK_B | SCK2_A | — | — | — | — | — | — |

| Register | Set Value = H'0 | | Set Value = H'1 | | Set Value = H'2 | | Set Value = H'3 | | Set Value = H'4 | |
|-----------|-----------------|---------|-----------------|------------|-----------------|-----|-----------------|-----|-----------------|-----|
| | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin |
| sel_scif0 | RX0_A | RX0_A | RX0_B | MLB_CLK | — | — | — | — | — | — |
| | TX0_A | TX0_A | TX0_B | MLB_DATA | — | — | — | — | — | — |
| | CTS0#_A | CTS0#_A | — | — | — | — | — | — | — | — |
| | RTS0#_A | RTS0#_A | — | — | — | — | — | — | — | — |
| | SCK0_A | SCK0_A | SCK0_B | MLB_SIG | — | — | — | — | — | — |
| sel_scif2 | RX2_A | RX2_A | RX2_B | MSIOF0_TXD | — | — | — | — | — | — |
| | TX2_A | TX2_A | TX2_B | MSIOF0_RXD | — | — | — | — | — | — |
| | SCK2_A | SCK2_A | — | — | — | — | — | — | — | — |

Table 9.13 Configuration of Registers in MOD_SEL1

| Register | Set Value = H'0 | | Set Value = H'1 | | Set Value = H'2 | | Set Value = H'3 | | Set Value = H'4 | |
|----------------|-----------------|-----------|-----------------|---------|-----------------|-----|-----------------|-----|-----------------|-----|
| | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin |
| sel_ssi2 | SSI_SCK2_A | RX0_A | SSI_SCK2_B | SD0_CD | — | — | — | — | — | — |
| | SSI_WS2_A | TX0_A | SSI_WS2_B | SD0_WP | — | — | — | — | — | — |
| sel_timer_tmu | TCLK1_A | SD0_CD | TCLK1_B | MLB_CLK | — | — | — | — | — | — |
| | TCLK2_A | SD0_WP | TCLK2_B | MLB_SIG | — | — | — | — | — | — |
| sel_usb_20_ch0 | USB0_OVC_A | USB30_OVC | USB0_OVC_B | SSI_WS5 | — | — | — | — | — | — |

| Register | Set Value = H'0 | | Set Value = H'1 | | Set Value = H'2 | | Set Value = H'3 | | Set Value = H'4 | |
|--------------|-----------------|----------------|-----------------|----------------|-----------------|------------|-----------------|-----|-----------------|-----|
| | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin |
| sel_hscif3_2 | HRX3_A | D14 | HRX3_B | D5 | HRX3_C | QSPI1_IO3 | HRX3_D | A0 | HRX3_E | D9 |
| | — | — | HSC3_B | D4 | HSC3_C | QSPI1_SSL | — | — | — | — |
| sel_hscif3_1 | — | — | — | — | HCTS3#_C | RPC_INT# | — | — | HCTS3#_E | D8 |
| sel_hscif3_0 | — | — | — | — | HRTS3#_C | RPC_RESET# | — | — | HRTS3#_E | D11 |
| sel_hscif4_2 | HSC4_A | QSPI0_SPCLK | HSC4_B | QSPI1_SPCLK | — | — | — | — | — | — |
| sel_hscif4_1 | HCTS4#_A | QSPI0_MOSI/IO0 | — | — | — | — | — | — | — | — |
| sel_hscif4_0 | HRTS4#_A | QSPI0_MISO/IO1 | — | — | — | — | — | — | — | — |
| | HRX4_A | QSPI0_IO3 | HRX4_B | QSPI1_MISO/IO1 | HRX4_C | A8 | HRX4_D | A13 | HRX4_E | A18 |
| sel_i2c6 | SDA6_A | A8 | SDA6_B | A1 | — | — | — | — | — | — |
| | SCL6_A | A11 | SCL6_B | A2 | — | — | — | — | — | — |
| sel_i2c7 | SCL7_A | RD/WR# | SCL7_B | D13 | — | — | — | — | — | — |
| | SDA7_A | EX_WAIT0 | SDA7_B | D14 | — | — | — | — | — | — |
| sel_msiof2 | MSIOF2_SCK_A | D8 | MSIOF2_SCK_B | A13 | — | — | — | — | — | — |
| | MSIOF2_SYNC_A | D9 | MSIOF2_SYNC_B | A10 | — | — | — | — | — | — |
| | MSIOF2_RXD_A | D10 | MSIOF2_RXD_B | A14 | — | — | — | — | — | — |
| sel_msiof3 | MSIOF3_SCK_A | D0 | MSIOF3_SCK_B | A5 | — | — | — | — | — | — |
| | MSIOF3_SYNC_A | D1 | MSIOF3_SYNC_B | A4 | — | — | — | — | — | — |
| | MSIOF3_RXD_A | D2 | MSIOF3_RXD_B | A6 | — | — | — | — | — | — |
| | — | — | MSIOF3_SS1_B | A0 | — | — | — | — | — | — |
| sel_scif3_1 | RX3_A | D5 | RX3_B | A8 | RX3_C | RD# | — | — | — | — |
| | SCK3_A | D1 | — | — | SCK3_C | RD/WR# | — | — | — | — |
| sel_scif3_0 | RTS3#_A | D4 | — | — | — | — | — | — | — | — |
| | CTS3#_A | D7 | — | — | — | — | — | — | — | — |
| sel_scif4_1 | CTS4#_A | A3 | — | — | CTS4#_C | D0 | — | — | — | — |
| | RTS4#_A | A4 | — | — | RTS4#_C | D1 | — | — | — | — |
| sel_scif4_0 | SCK4_A | A5 | SCK4_B | D8 | — | — | — | — | — | — |
| | RX4_A | A6 | RX4_B | D13 | RX4_C | D2 | — | — | — | — |
| sel_scif5_1 | RX5_A | A12 | RX5_B | EX_WAIT0 | RX5_C | D2 | — | — | — | — |
| | SCK5_A | A13 | — | — | — | — | — | — | — | — |
| sel_scif5_0 | — | — | — | — | — | — | — | — | — | |

| Register | Set Value = H'0 | | Set Value = H'1 | | Set Value = H'2 | | Set Value = H'3 | | Set Value = H'4 | |
|--------------|-----------------|----------------|-----------------|------|-----------------|-----|-----------------|-----|-----------------|-----|
| | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin |
| sel_vin4 | VI4_DATA0_A | QSPI1_SPCLK | VI4_DATA0_B | A8 | — | — | — | — | — | — |
| | VI4_DATA1_A | QSPI1_MOSI/IO0 | VI4_DATA1_B | A11 | — | — | — | — | — | — |
| | VI4_DATA2_A | QSPI1_MISO/IO1 | VI4_DATA2_B | CS0# | — | — | — | — | — | — |
| | VI4_DATA3_A | QSPI1_IO2 | VI4_DATA3_B | WE0# | — | — | — | — | — | — |
| | VI4_DATA4_A | QSPI1_IO3 | VI4_DATA4_B | D5 | — | — | — | — | — | — |
| | VI4_DATA5_A | QSPI1_SSL | VI4_DATA5_B | D6 | — | — | — | — | — | — |
| | VI4_DATA6_A | RPC_INT# | VI4_DATA6_B | SCL4 | — | — | — | — | — | — |
| | VI4_DATA7_A | RPC_RESET# | VI4_DATA7_B | SDA4 | — | — | — | — | — | — |
| sel_vin5 | VI5_CLK_A | A0 | VI5_CLK_B | BS# | — | — | — | — | — | — |
| | VI5_DATA0_A | A1 | VI5_DATA0_B | RD# | — | — | — | — | — | — |
| | VI5_DATA1_A | A2 | VI5_DATA1_B | D4 | — | — | — | — | — | — |
| | VI5_HSYN_C#_A | A8 | — | — | — | — | — | — | — | — |
| | VI5_VSYN_C#_A | A9 | — | — | — | — | — | — | — | — |
| | VI5_FIELD_A | A10 | — | — | — | — | — | — | — | — |
| | VI5_DATA3_A | A12 | VI5_DATA3_B | D12 | — | — | — | — | — | — |
| | VI5_DATA4_A | A15 | VI5_DATA4_B | D13 | — | — | — | — | — | — |
| | VI5_DATA5_A | A16 | VI5_DATA5_B | D14 | — | — | — | — | — | — |
| | VI5_DATA6_A | A17 | VI5_DATA6_B | SCL4 | — | — | — | — | — | — |
| | VI5_DATA7_A | A18 | VI5_DATA7_B | SDA4 | — | — | — | — | — | — |
| | VI5_DATA2_A | A19 | VI5_DATA2_B | D7 | — | — | — | — | — | — |
| | VI5_CLKENB_A | D1 | — | — | — | — | — | — | — | — |
| | VI5_DATA14_A | D2 | — | — | — | — | — | — | — | — |
| | VI5_DATA15_A | D3 | — | — | — | — | — | — | — | — |
| | VI5_DATA12_A | D8 | — | — | — | — | — | — | — | — |
| VI5_DATA10_A | D9 | — | — | — | — | — | — | — | — | |

| Register | Set Value = H'0 | | Set Value = H'1 | | Set Value = H'2 | | Set Value = H'3 | | Set Value = H'4 | |
|------------|-----------------|-----------|-----------------|--------|-----------------|----------|-----------------|-----|-----------------|-----|
| | Function | Pin | Function | Pin | Function | Pin | Function | Pin | Function | Pin |
| sel_vin5 | VI5_ | D10 | — | — | — | — | — | — | — | — |
| | DATA13_A | | | | | | | | | |
| | VI5_ | D11 | — | — | — | — | — | — | — | — |
| | DATA11_A | | | | | | | | | |
| sel_adgc_1 | VI5_DATA | D12 | — | — | — | — | — | — | — | — |
| | 8_A | | | | | | | | | |
| sel_adgc_0 | VI5_DATA | D13 | — | — | — | — | — | — | — | — |
| | 9_A | | | | | | | | | |
| sel_ssi9 | AUDIO_ | SSI_SDATA | AUDIO_ | SSI_ | AUDIO_ | SSI_SCK6 | — | — | — | — |
| | CLKC_A | 9 | CLKC_B | SDATA1 | CLKC_C | | | | | |
| sel_ssi9 | SSI_SCK9 | SSI_SDATA | SSI_SCK9_ | RX1 | — | — | — | — | — | — |
| | _A | 2 | B | | | | | | | |
| sel_ssi9 | SSI_WS9 | SSI_SDATA | SSI_WS9_ | TX1 | — | — | — | — | — | — |
| | A | 4 | B | | | | | | | |

9.3 Operation



9.3.1 Function Setting for Multiplexed Pins

Setting the LSI multiplexed pin setting mask register (PMMR) is necessary before setting each of the GPIO/peripheral function select registers GPSR0 to GPSR6, peripheral function select registers IPSR0 to IPSR13, POC control registers POCCTRL0, POCCTRL2, TDSEL control registers TDSELCTRL0 and Module select register MOD_SEL0-1. Specifically, the inverse of the value to be set in the select register must be written to the LSI multiplexed pin setting mask register. Otherwise, the GPIO/peripheral function select registers 0 to 6 (GPSR0 to GPSR6) and peripheral function select registers 0 to 13 (IPSR0 to IPSR13) cannot be set.

(1) Procedure for changing pin function from GPIO to peripheral function

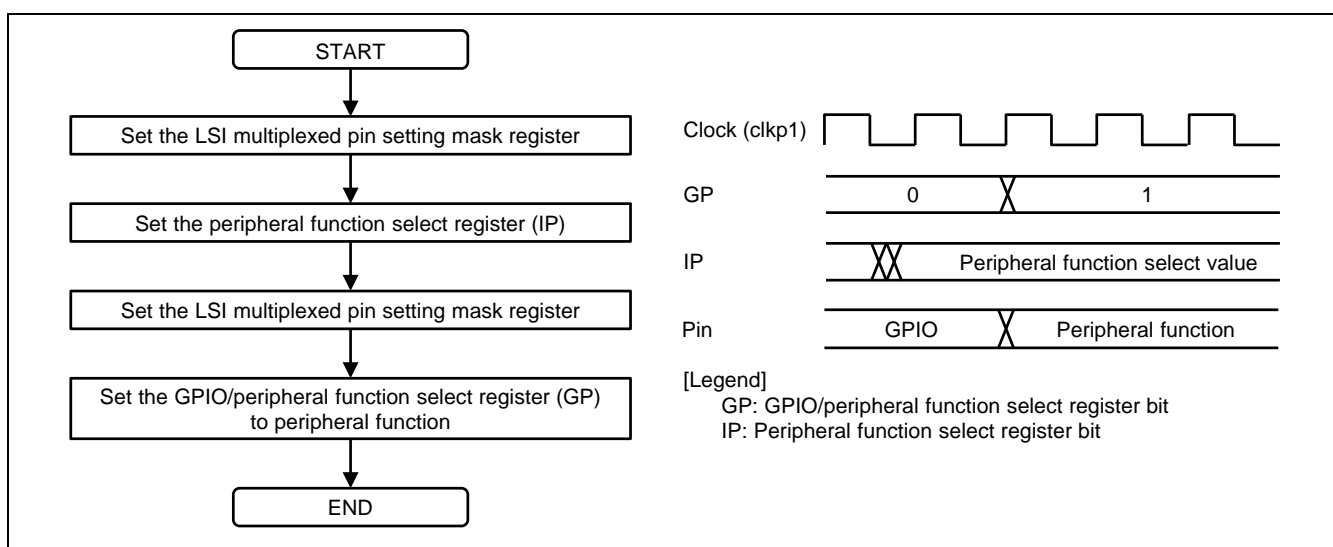


Figure 9.2 Procedure for Changing Pin Function from GPIO to Peripheral Function

(2) Procedure for changing pin function from peripheral function to GPIO

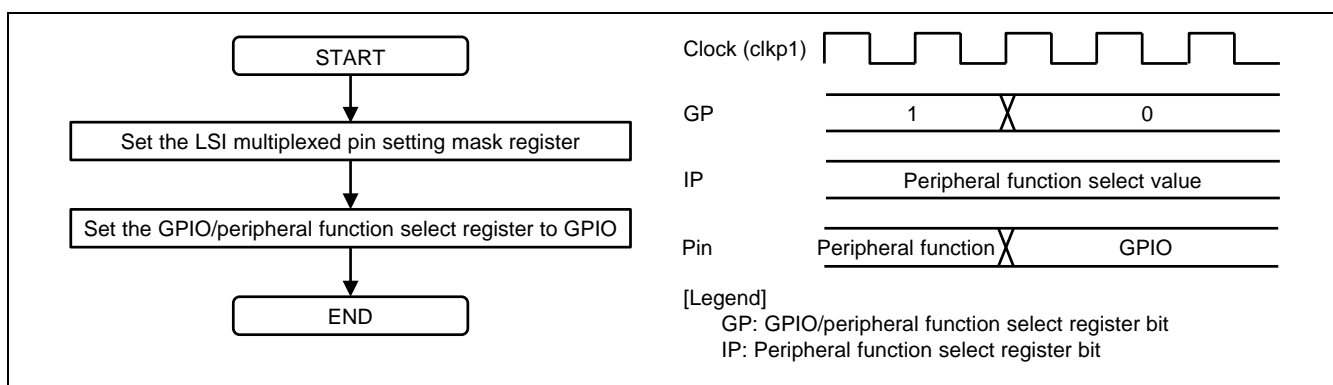


Figure 9.3 Procedure for Changing Pin Function from Peripheral function to GPIO

(3) Procedure 1 for changing pin function from one peripheral function to another peripheral function

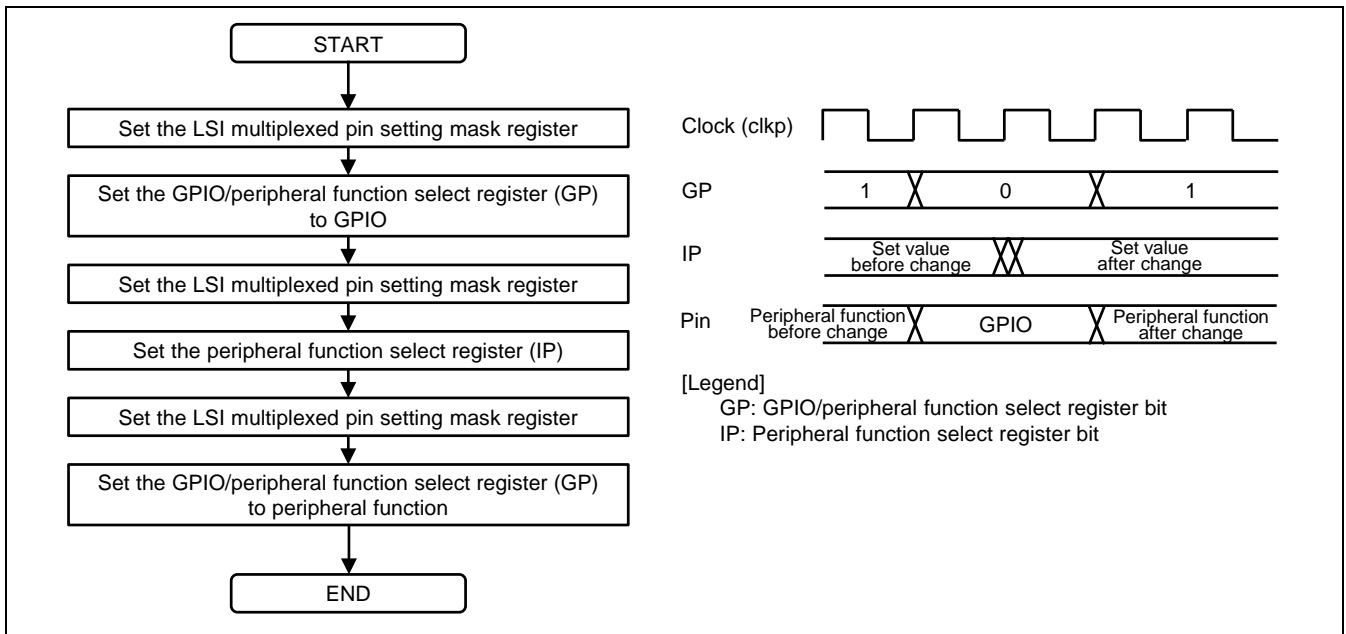


Figure 9.4 Procedure for Changing Pin Function from One Peripheral Function to Another Peripheral Function (with GPIO Setting)

9.3.2 Setting Pull-Up/Down Resistors

The LSI pin pull-on/off control registers 0 to 5 (PUEN0 to PUEN5) and pull-up/down control registers 0 to 5 (PUD0 to PUD5) are used.

10. General-Purpose Input/Output Ports (GPIO)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

10.1 Overview

10.1.1 Features

The second generation RZ/G series products have up to eight GPIO blocks (RZ/G2E series product has up to seven GPIO blocks), each of which is a functional block that supports up to 32 port pins for general input/output and interrupt input. (A maximum of 156 ports pins for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, and RZ/G2N, and 132 ports pins for RZ/G2E in total can be used. Note that the port pins are multiplexed.) When the relevant register is written to, a signal is output via the corresponding general output port pin. When a signal is input via the general input port pin, the corresponding register indicates the value of the input signal; specifically, when an interrupt is input via a general port pin, the relevant register indicates that it is currently receiving an interrupt input, and an interrupt is also requested to the CPU core via the interrupt control block. The functions (modes) can be assigned to each port pin as desired by some setting the corresponding registers. It is also possible to select the signal polarity (positive or negative logic) and the interrupt detection condition (one edge/both edge or level) for each port. In this LSI, general output data mode of GPIO can be set in normal mode (outputting data as normal) or high/low level data output mode. Particularly, a filtering function to prevent external chattering is also available for port pins 0 to 3 in input modes for each GPIO block. Module clock is S3D4 ϕ for all channels.

10.1.2 Block Diagram

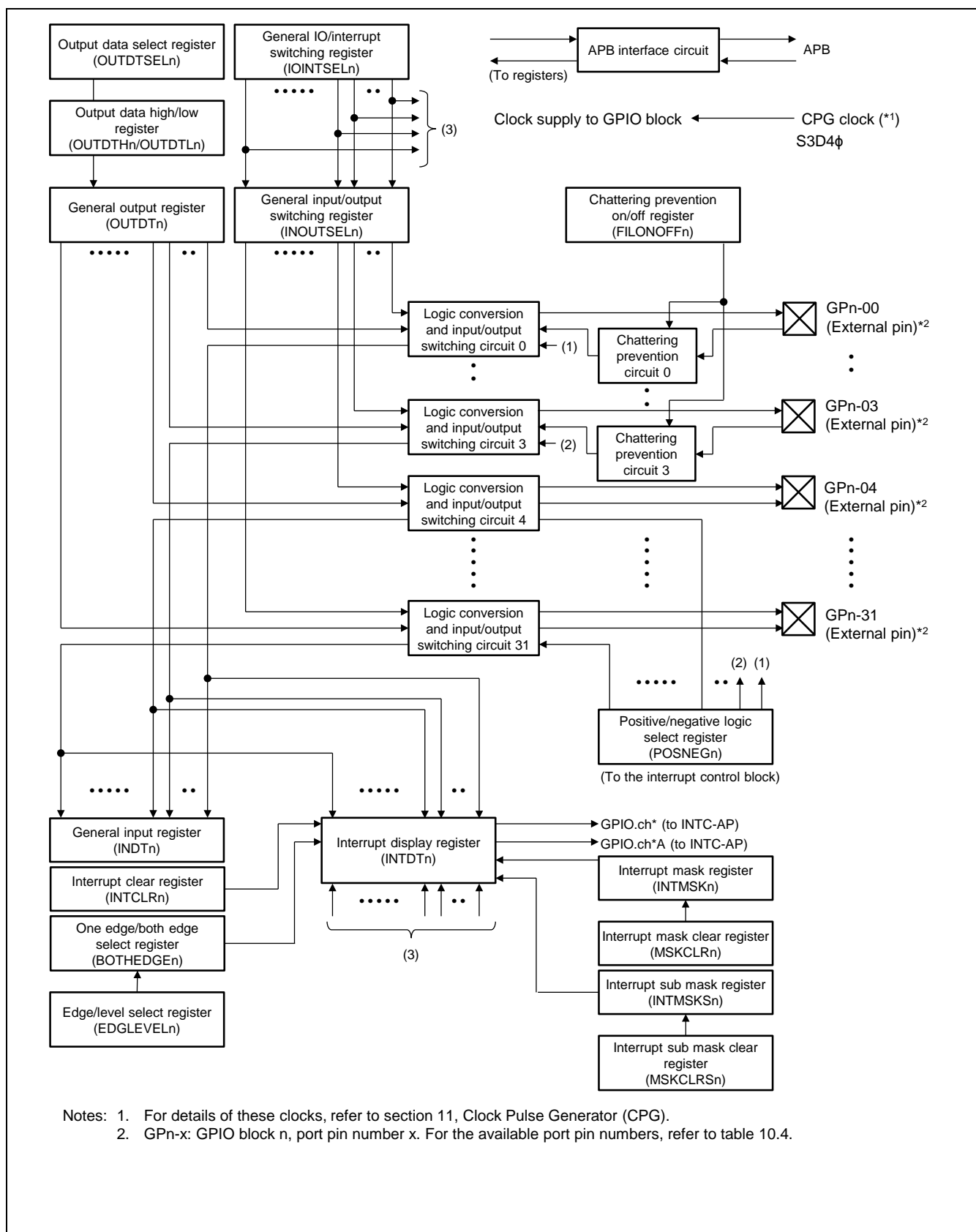


Figure 10.1 GPIO Block Configuration (1)

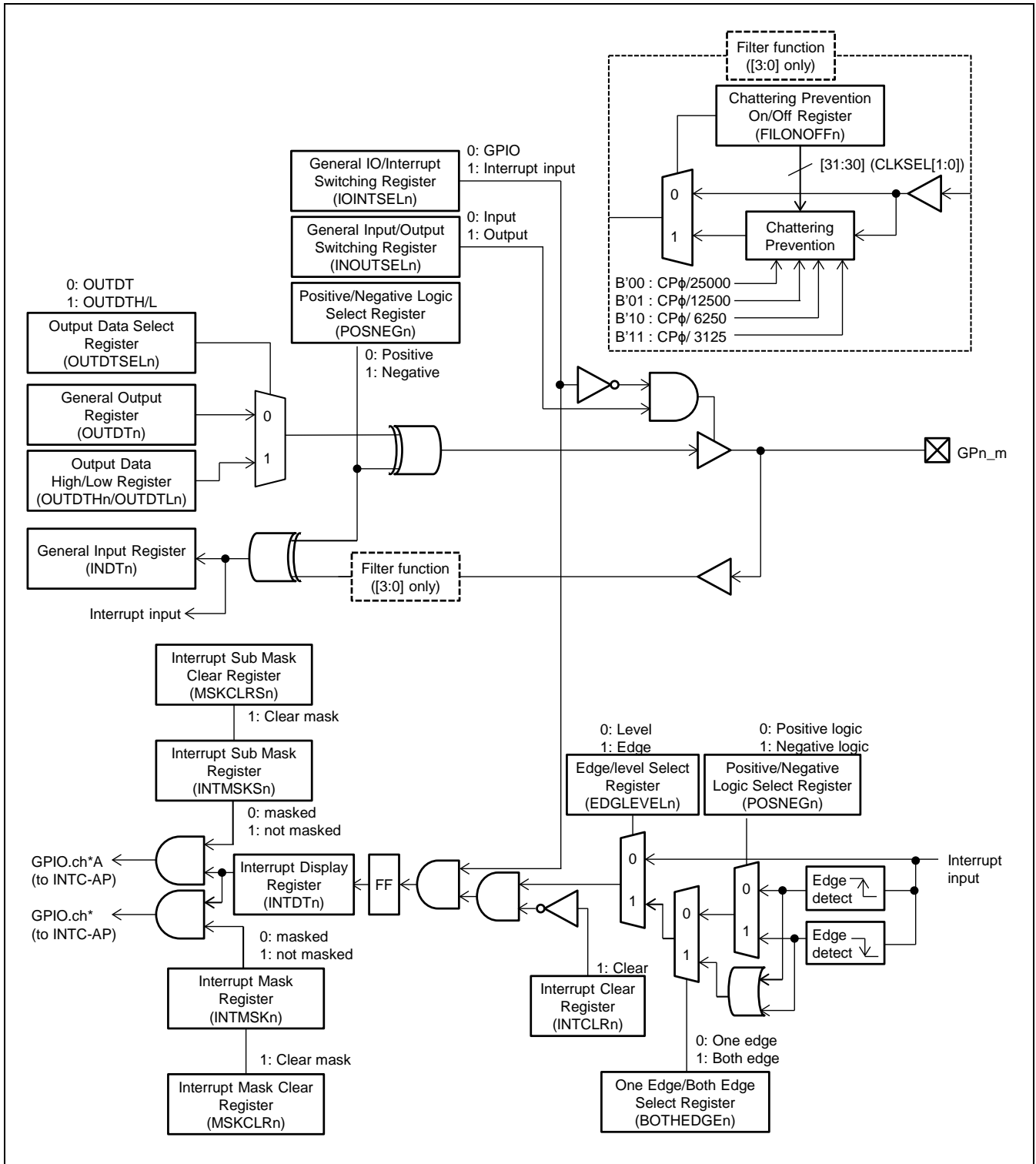


Figure 10.2 GPIO Block Configuration (2)

10.1.3 External Pins

Table 10.1 shows the pin configuration of the GPIO.

Table 10.1 Pin Configuration

| | | | | Second Generation RZ/G Series Products | | | |
|---------------------|-----|-----------------------------|---|---|----------------------------|--------|--------|
| | I/O | Function | Descriptions | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| GP0_00 to GP7_03 | I/O | IO/interrupt input ports | General input/output and interrupt input | √ | √ | √ | — |
| GP0_00 to GP6_13 | I/O | IO/interrupt input ports | General input/output and interrupt input | — | — | — | √ |

10.1.4 Register Configuration

Each GPIO block incorporates seventeen 32-bit registers. These registers can be accessed via the APB interface. Table 10.2 describes all the GPIO block registers.

Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined.

Table 10.2 Register Configuration

| | | | | | | Second Generation RZ/G Series Products | | | |
|--|--------------|-----|--------------------------------|-------------|----------------|---|----------------------------|--------|--------|
| Register Name | Abbreviation | R/W | Value after Power- On Reset | Address | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| General IO/interrupt switching register 0 | IOINTSEL0 | R/W | H'0000_0000 | H'E605_0000 | 32 | √ | √ | √ | √ |
| General input/output switching register 0 | INOUTSEL0 | R/W | H'0000_0000 | H'E605_0004 | 32 | √ | √ | √ | √ |
| General output register 0 | OUTDT0 | R/W | H'0000_0000 | H'E605_0008 | 32 | √ | √ | √ | √ |
| General input register 0 | INDT0 | R | State of the port pins | H'E605_000C | 32 | √ | √ | √ | √ |
| Interrupt display register 0 | INTDT0 | R | H'0000_0000 | H'E605_0010 | 32 | √ | √ | √ | √ |
| Interrupt clear register 0 | INTCLR0 | W | H'0000_0000 | H'E605_0014 | 32 | √ | √ | √ | √ |
| Interrupt mask register 0 | INTMSK0 | R/W | H'0000_0000 | H'E605_0018 | 32 | √ | √ | √ | √ |
| Interrupt mask clear register 0 | MSKCLR0 | W | H'0000_0000 | H'E605_001C | 32 | √ | √ | √ | √ |
| Positive/negative logic select register 0 | POSNEG0 | R/W | H'0000_0000 | H'E605_0020 | 32 | √ | √ | √ | √ |
| Edge/level select register 0 | EDGLEVELO | R/W | H'0000_0000 | H'E605_0024 | 32 | √ | √ | √ | √ |
| Chattering prevention on/off register 0 | FILONOFF0 | R/W | H'0000_0000 | H'E605_0028 | 32 | √ | √ | √ | √ |
| Interrupt sub mask register 0 | INTMSKS0 | R/W | H'0000_0000 | H'E605_0038 | 32 | √ | √ | √ | √ |

| Register Name | Abbreviation | R/W | Value after Power-On Reset | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|--------------|-----|----------------------------|-------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Interrupt sub mask clear register 0 | MSKCLRS0 | W | H'0000_0000 | H'E605_003C | 32 | √ | √ | √ | √ |
| Output data select register 0 | OUTDTSEL0 | R/W | H'0000_0000 | H'E605_0040 | 32 | √ | √ | √ | √ |
| Output data high register 0 | OUTDTH0 | R/W | H'0000_0000 | H'E605_0044 | 32 | √ | √ | √ | √ |
| Output data low register 0 | OUTDTL0 | R/W | H'0000_0000 | H'E605_0048 | 32 | √ | √ | √ | √ |
| One edge/both edge select register 0 | BOTHEDGE0 | R/W | H'0000_0000 | H'E605_004C | 32 | √ | √ | √ | √ |
| General IO/interrupt switching register 1 | IOINTSEL1 | R/W | H'0000_0000 | H'E605_1000 | 32 | √ | √ | √ | √ |
| General input/output switching register 1 | INOUTSEL1 | R/W | H'0000_0000 | H'E605_1004 | 32 | √ | √ | √ | √ |
| General output register 1 | OUTDT1 | R/W | H'0000_0000 | H'E605_1008 | 32 | √ | √ | √ | √ |
| General input register 1 | INDT1 | R | State of the port pins | H'E605_100C | 32 | √ | √ | √ | √ |
| Interrupt display register 1 | INTDT1 | R | H'0000_0000 | H'E605_1010 | 32 | √ | √ | √ | √ |
| Interrupt clear register 1 | INTCLR1 | W | H'0000_0000 | H'E605_1014 | 32 | √ | √ | √ | √ |
| Interrupt mask register 1 | INTMSK1 | R/W | H'0000_0000 | H'E605_1018 | 32 | √ | √ | √ | √ |
| Interrupt mask clear register 1 | MSKCLR1 | W | H'0000_0000 | H'E605_101C | 32 | √ | √ | √ | √ |
| Positive/negative logic select register 1 | POSNEG1 | R/W | H'0000_0000 | H'E605_1020 | 32 | √ | √ | √ | √ |
| Edge/level select register 1 | EDGLEVE1 | R/W | H'0000_0000 | H'E605_1024 | 32 | √ | √ | √ | √ |
| Chattering prevention on/off register 1 | FILONOFF1 | R/W | H'0000_0000 | H'E605_1028 | 32 | √ | √ | √ | √ |
| Interrupt sub mask register 1 | INTMSKS1 | R/W | H'0000_0000 | H'E605_1038 | 32 | √ | √ | √ | √ |
| Interrupt sub mask clear register 1 | MSKCLRS1 | W | H'0000_0000 | H'E605_103C | 32 | √ | √ | √ | √ |
| Output data select register 1 | OUTDTSEL1 | R/W | H'0000_0000 | H'E605_1040 | 32 | √ | √ | √ | √ |
| Output data high register 1 | OUTDTH1 | R/W | H'0000_0000 | H'E605_1044 | 32 | √ | √ | √ | √ |
| Output data low register 1 | OUTDTL1 | R/W | H'0000_0000 | H'E605_1048 | 32 | √ | √ | √ | √ |
| One edge/both edge select register 1 | BOTHEDGE1 | R/W | H'0000_0000 | H'E605_104C | 32 | √ | √ | √ | √ |
| General IO/interrupt switching register 2 | IOINTSEL2 | R/W | H'0000_0000 | H'E605_2000 | 32 | √ | √ | √ | √ |
| General input/output switching register 2 | INOUTSEL2 | R/W | H'0000_0000 | H'E605_2004 | 32 | √ | √ | √ | √ |
| General output register 2 | OUTDT2 | R/W | H'0000_0000 | H'E605_2008 | 32 | √ | √ | √ | √ |
| General input register 2 | INDT2 | R | State of the port pins | H'E605_200C | 32 | √ | √ | √ | √ |
| Interrupt display register 2 | INTDT2 | R | H'0000_0000 | H'E605_2010 | 32 | √ | √ | √ | √ |
| Interrupt clear register 2 | INTCLR2 | W | H'0000_0000 | H'E605_2014 | 32 | √ | √ | √ | √ |
| Interrupt mask register 2 | INTMSK2 | R/W | H'0000_0000 | H'E605_2018 | 32 | √ | √ | √ | √ |
| Interrupt mask clear register 2 | MSKCLR2 | W | H'0000_0000 | H'E605_201C | 32 | √ | √ | √ | √ |

| | | | | | | Second Generation | | | |
|---|--------------|-----|----------------------------|-------------|-------------|----------------------|----------------------------|--------|--------|
| | | | | | | RZ/G Series Products | | | |
| Register Name | Abbreviation | R/W | Value after Power-On Reset | Address | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Positive/negative logic select register 2 | POSNEG2 | R/W | H'0000_0000 | H'E605_2020 | 32 | √ | √ | √ | √ |
| Edge/level select register 2 | EDGLEVEL2 | R/W | H'0000_0000 | H'E605_2024 | 32 | √ | √ | √ | √ |
| Chattering prevention on/off register 2 | FILONOFF2 | R/W | H'0000_0000 | H'E605_2028 | 32 | √ | √ | √ | √ |
| Interrupt sub mask register 2 | INTMSKS2 | R/W | H'0000_0000 | H'E605_2038 | 32 | √ | √ | √ | √ |
| Interrupt sub mask clear register 2 | MSKCLRS2 | W | H'0000_0000 | H'E605_203C | 32 | √ | √ | √ | √ |
| Output data select register 2 | OUTDTSEL2 | R/W | H'0000_0000 | H'E605_2040 | 32 | √ | √ | √ | √ |
| Output data high register 2 | OUTDTH2 | R/W | H'0000_0000 | H'E605_2044 | 32 | √ | √ | √ | √ |
| Output data low register 2 | OUTDTL2 | R/W | H'0000_0000 | H'E605_2048 | 32 | √ | √ | √ | √ |
| One edge/both edge select register 2 | BOTHEDGE2 | R/W | H'0000_0000 | H'E605_204C | 32 | √ | √ | √ | √ |
| General IO/interrupt switching register 3 | IOINTSEL3 | R/W | H'0000_0000 | H'E605_3000 | 32 | √ | √ | √ | √ |
| General input/output switching register 3 | INOUTSEL3 | R/W | H'0000_0000 | H'E605_3004 | 32 | √ | √ | √ | √ |
| General output register 3 | OUTDT3 | R/W | H'0000_0000 | H'E605_3008 | 32 | √ | √ | √ | √ |
| General input register 3 | INDT3 | R | State of the port pins | H'E605_300C | 32 | √ | √ | √ | √ |
| Interrupt display register 3 | INTDT3 | R | H'0000_0000 | H'E605_3010 | 32 | √ | √ | √ | √ |
| Interrupt clear register 3 | INTCLR3 | W | H'0000_0000 | H'E605_3014 | 32 | √ | √ | √ | √ |
| Interrupt mask register 3 | INTMSK3 | R/W | H'0000_0000 | H'E605_3018 | 32 | √ | √ | √ | √ |
| Interrupt mask clear register 3 | MSKCLR3 | W | H'0000_0000 | H'E605_301C | 32 | √ | √ | √ | √ |
| Positive/negative logic select register 3 | POSNEG3 | R/W | H'0000_0000 | H'E605_3020 | 32 | √ | √ | √ | √ |
| Edge/level select register 3 | EDGLEVEL3 | R/W | H'0000_0000 | H'E605_3024 | 32 | √ | √ | √ | √ |
| Chattering prevention on/off register 3 | FILONOFF3 | R/W | H'0000_0000 | H'E605_3028 | 32 | √ | √ | √ | √ |
| Interrupt sub mask register 3 | INTMSKS3 | R/W | H'0000_0000 | H'E605_3038 | 32 | √ | √ | √ | √ |
| Interrupt sub mask clear register 3 | MSKCLRS3 | W | H'0000_0000 | H'E605_303C | 32 | √ | √ | √ | √ |
| Output data select register 3 | OUTDTSEL3 | R/W | H'0000_0000 | H'E605_3040 | 32 | √ | √ | √ | √ |
| Output data high register 3 | OUTDTH3 | R/W | H'0000_0000 | H'E605_3044 | 32 | √ | √ | √ | √ |
| Output data low register 3 | OUTDTL3 | R/W | H'0000_0000 | H'E605_3048 | 32 | √ | √ | √ | √ |
| One edge/both edge select register 3 | BOTHEDGE3 | R/W | H'0000_0000 | H'E605_304C | 32 | √ | √ | √ | √ |
| General IO/interrupt switching register 4 | IOINTSEL4 | R/W | H'0000_0000 | H'E605_4000 | 32 | √ | √ | √ | √ |
| General input/output switching register 4 | INOUTSEL4 | R/W | H'0000_0000 | H'E605_4004 | 32 | √ | √ | √ | √ |
| General output register 4 | OUTDT4 | R/W | H'0000_0000 | H'E605_4008 | 32 | √ | √ | √ | √ |
| General input register 4 | INDT4 | R | State of the port pins | H'E605_400C | 32 | √ | √ | √ | √ |

| | | | | | | Second Generation RZ/G Series Products | | | |
|---|--------------|-----|----------------------------|-------------|-------------|---|----------------------------|--------|--------|
| Register Name | Abbreviation | R/W | Value after Power-On Reset | Address | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Interrupt display register 4 | INTDT4 | R | H'0000_0000 | H'E605_4010 | 32 | √ | √ | √ | √ |
| Interrupt clear register 4 | INTCLR4 | W | H'0000_0000 | H'E605_4014 | 32 | √ | √ | √ | √ |
| Interrupt mask register 4 | INTMSK4 | R/W | H'0000_0000 | H'E605_4018 | 32 | √ | √ | √ | √ |
| Interrupt mask clear register 4 | MSKCLR4 | W | H'0000_0000 | H'E605_401C | 32 | √ | √ | √ | √ |
| Positive/negative logic select register 4 | POSNEG4 | R/W | H'0000_0000 | H'E605_4020 | 32 | √ | √ | √ | √ |
| Edge/level select register 4 | EDGLEVEL4 | R/W | H'0000_0000 | H'E605_4024 | 32 | √ | √ | √ | √ |
| Chattering prevention on/off register 4 | FILONOFF4 | R/W | H'0000_0000 | H'E605_4028 | 32 | √ | √ | √ | √ |
| Interrupt sub mask register 4 | INTMSKS4 | R/W | H'0000_0000 | H'E605_4038 | 32 | √ | √ | √ | √ |
| Interrupt sub mask clear register 4 | MSKCLRS4 | W | H'0000_0000 | H'E605_403C | 32 | √ | √ | √ | √ |
| Output data select register 4 | OUTDTSEL4 | R/W | H'0000_0000 | H'E605_4040 | 32 | √ | √ | √ | √ |
| Output data high register 4 | OUTDTH4 | R/W | H'0000_0000 | H'E605_4044 | 32 | √ | √ | √ | √ |
| Output data low register 4 | OUTDTL4 | R/W | H'0000_0000 | H'E605_4048 | 32 | √ | √ | √ | √ |
| One edge/both edge select register 4 | BOTHEDGE4 | R/W | H'0000_0000 | H'E605_404C | 32 | √ | √ | √ | √ |
| General IO/interrupt switching register 5 | IOINTSEL5 | R/W | H'0000_0000 | H'E605_5000 | 32 | √ | √ | √ | √ |
| General input/output switching register 5 | INOUTSEL5 | R/W | H'0000_0000 | H'E605_5004 | 32 | √ | √ | √ | √ |
| General output register 5 | OUTDT5 | R/W | H'0000_0000 | H'E605_5008 | 32 | √ | √ | √ | √ |
| General input register 5 | INDT5 | R | State of the port pins | H'E605_500C | 32 | √ | √ | √ | √ |
| Interrupt display register 5 | INTDT5 | R | H'0000_0000 | H'E605_5010 | 32 | √ | √ | √ | √ |
| Interrupt clear register 5 | INTCLR5 | W | H'0000_0000 | H'E605_5014 | 32 | √ | √ | √ | √ |
| Interrupt mask register 5 | INTMSK5 | R/W | H'0000_0000 | H'E605_5018 | 32 | √ | √ | √ | √ |
| Interrupt mask clear register 5 | MSKCLR5 | W | H'0000_0000 | H'E605_501C | 32 | √ | √ | √ | √ |
| Positive/negative logic select register 5 | POSNEG5 | R/W | H'0000_0000 | H'E605_5020 | 32 | √ | √ | √ | √ |
| Edge/level select register 5 | EDGLEVEL5 | R/W | H'0000_0000 | H'E605_5024 | 32 | √ | √ | √ | √ |
| Chattering prevention on/off register 5 | FILONOFF5 | R/W | H'0000_0000 | H'E605_5028 | 32 | √ | √ | √ | √ |
| Interrupt sub mask register 5 | INTMSKS5 | R/W | H'0000_0000 | H'E605_5038 | 32 | √ | √ | √ | √ |
| Interrupt sub mask clear register 5 | MSKCLRS5 | W | H'0000_0000 | H'E605_503C | 32 | √ | √ | √ | √ |
| Output data select register 5 | OUTDTSEL5 | R/W | H'0000_0000 | H'E605_5040 | 32 | √ | √ | √ | √ |
| Output data high register 5 | OUTDTH5 | R/W | H'0000_0000 | H'E605_5044 | 32 | √ | √ | √ | √ |
| Output data low register 5 | OUTDTL5 | R/W | H'0000_0000 | H'E605_5048 | 32 | √ | √ | √ | √ |
| One edge/both edge select register 5 | BOTHEDGE5 | R/W | H'0000_0000 | H'E605_504C | 32 | √ | √ | √ | √ |

| | | | | | | Second Generation RZ/G Series Products | | | |
|---|--------------|-----|----------------------------|-------------|-------------|---|----------------------------|--------|--------|
| Register Name | Abbreviation | R/W | Value after Power-On Reset | Address | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| General IO/interrupt switching register 6 | IOINTSEL6 | R/W | H'0000_0000 | H'E605_5400 | 32 | √ | √ | √ | √ |
| General input/output switching register 6 | INOUTSEL6 | R/W | H'0000_0000 | H'E605_5404 | 32 | √ | √ | √ | √ |
| General output register 6 | OUTDT6 | R/W | H'0000_0000 | H'E605_5408 | 32 | √ | √ | √ | √ |
| General input register 6 | INDT6 | R | State of the port pins | H'E605_540C | 32 | √ | √ | √ | √ |
| Interrupt display register 6 | INTDT6 | R | H'0000_0000 | H'E605_5410 | 32 | √ | √ | √ | √ |
| Interrupt clear register 6 | INTCLR6 | W | H'0000_0000 | H'E605_5414 | 32 | √ | √ | √ | √ |
| Interrupt mask register 6 | INTMSK6 | R/W | H'0000_0000 | H'E605_5418 | 32 | √ | √ | √ | √ |
| Interrupt mask clear register 6 | MSKCLR6 | W | H'0000_0000 | H'E605_541C | 32 | √ | √ | √ | √ |
| Positive/negative logic select register 6 | POSNEG6 | R/W | H'0000_0000 | H'E605_5420 | 32 | √ | √ | √ | √ |
| Edge/level select register 6 | EDGLEVE6 | R/W | H'0000_0000 | H'E605_5424 | 32 | √ | √ | √ | √ |
| Chattering prevention on/off register 6 | FILONOFF6 | R/W | H'0000_0000 | H'E605_5428 | 32 | √ | √ | √ | √ |
| Interrupt sub mask register 6 | INTMSKS6 | R/W | H'0000_0000 | H'E605_5438 | 32 | √ | √ | √ | √ |
| Interrupt sub mask clear register 6 | MSKCLRS6 | W | H'0000_0000 | H'E605_543C | 32 | √ | √ | √ | √ |
| Output data select register 6 | OUTDTSEL6 | R/W | H'0000_0000 | H'E605_5440 | 32 | √ | √ | √ | √ |
| Output data high register 6 | OUTDTH6 | R/W | H'0000_0000 | H'E605_5444 | 32 | √ | √ | √ | √ |
| Output data low register 6 | OUTDTL6 | R/W | H'0000_0000 | H'E605_5448 | 32 | √ | √ | √ | √ |
| One edge/both edge select register 6 | BOTHEDE6 | R/W | H'0000_0000 | H'E605_544C | 32 | √ | √ | √ | √ |
| General IO/interrupt switching register 7 | IOINTSEL7 | R/W | H'0000_0000 | H'E605_5800 | 32 | √ | √ | √ | — |
| General input/output switching register 7 | INOUTSEL7 | R/W | H'0000_0000 | H'E605_5804 | 32 | √ | √ | √ | — |
| General output register 7 | OUTDT7 | R/W | H'0000_0000 | H'E605_5808 | 32 | √ | √ | √ | — |
| General input register 7 | INDT7 | R | State of the port pins | H'E605_580C | 32 | √ | √ | √ | — |
| Interrupt display register 7 | INTDT7 | R | H'0000_0000 | H'E605_5810 | 32 | √ | √ | √ | — |
| Interrupt clear register 7 | INTCLR7 | W | H'0000_0000 | H'E605_5814 | 32 | √ | √ | √ | — |
| Interrupt mask register 7 | INTMSK7 | R/W | H'0000_0000 | H'E605_5818 | 32 | √ | √ | √ | — |
| Interrupt mask clear register 7 | MSKCLR7 | W | H'0000_0000 | H'E605_581C | 32 | √ | √ | √ | — |
| Positive/negative logic select register 7 | POSNEG7 | R/W | H'0000_0000 | H'E605_5820 | 32 | √ | √ | √ | — |
| Edge/level select register 7 | EDGLEVE7 | R/W | H'0000_0000 | H'E605_5824 | 32 | √ | √ | √ | — |
| Chattering prevention on/off register 7 | FILONOFF7 | R/W | H'0000_0000 | H'E605_5828 | 32 | √ | √ | √ | — |
| Interrupt sub mask register 7 | INTMSKS7 | R/W | H'0000_0000 | H'E605_5838 | 32 | √ | √ | √ | — |
| Interrupt sub mask clear register 7 | MSKCLRS7 | W | H'0000_0000 | H'E605_583C | 32 | √ | √ | √ | — |

| | | | | | | Second Generation RZ/G Series Products | | | |
|--------------------------------------|--------------|-----|----------------------------|-------------|-------------|---|----------------------------|--------|--------|
| Register Name | Abbreviation | R/W | Value after Power-On Reset | Address | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Output data select register 7 | OUTDTSEL7 | R/W | H'0000_0000 | H'E605_5840 | 32 | √ | √ | √ | — |
| Output data high register 7 | OUTDTH7 | R/W | H'0000_0000 | H'E605_5844 | 32 | √ | √ | √ | — |
| Output data low register 7 | OUTDTL7 | R/W | H'0000_0000 | H'E605_5848 | 32 | √ | √ | √ | — |
| One edge/both edge select register 7 | BOTHEDGE7 | R/W | H'0000_0000 | H'E605_584C | 32 | √ | √ | √ | — |

10.1.5 Connected Module

Table 10.3 Connected module

| Module name | Connected module name | Function of connected module |
|-------------|-----------------------|------------------------------|
| GPIO | AP-System Core | Access the Registers |
| | CPG | Output clocks |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |
| | INTC-AP | Control to interrupt |

10.2 Register Description

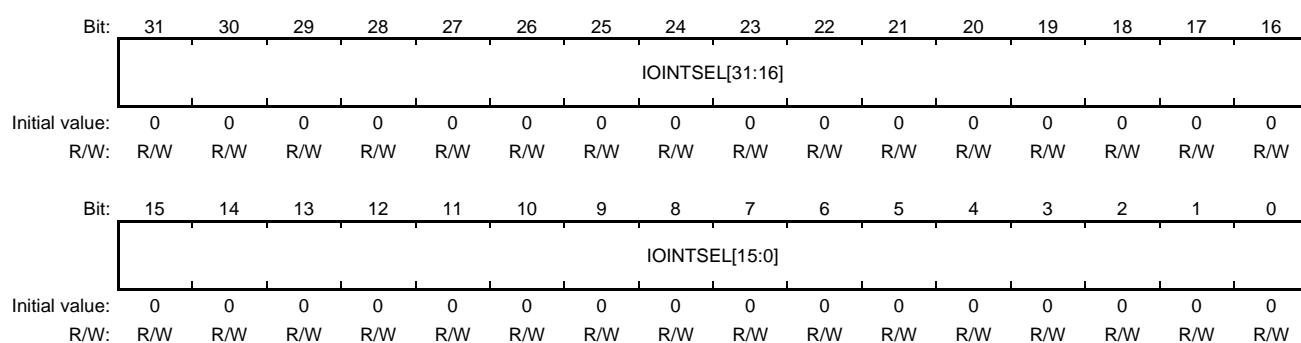
10.2.1 General IO/Interrupt Switching Register n (IOINTSEL0 to IOINTSELn)

| | | | |
|-------------|-------------------------------|-------------|-------------|
| RZ/G2H ↓ | RZ/G2M V1.3, RZ/G2M V3.0 ↓ | RZ/G2N ↓ | RZ/G2E ↓ |
|-------------|-------------------------------|-------------|-------------|

Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

IOINTSEL selects either general input/output mode or interrupt input mode for each of the port pins 0 to 31 of the GPIO block. When general input/output mode is selected for a port, it is also necessary to select either input or output mode for the port using the corresponding bit in the general input/output switching register. When interrupt input mode is selected for a port, the setting of the general input/output switching register for the port is ignored.

[Hardware default value: H'0000_0000 = general input/output mode is selected for all the ports.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 31 to 0 | IOINTSEL [31:0] | H'0000_0000 | R/W | Selects either general input/output mode or interrupt input mode for each port using the bits corresponding to the port numbers. 0: General input/output mode. 1: Interrupt input mode. |

Note: Unused bits should be set to the initial values.

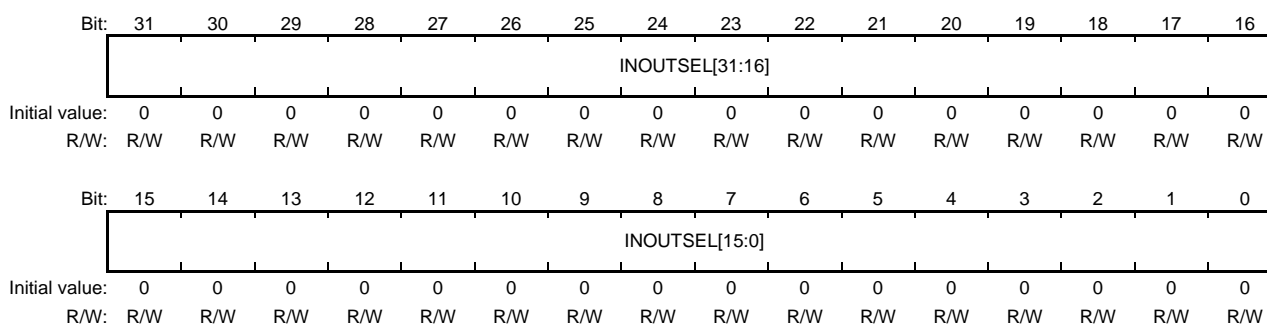
10.2.2 General Input/Output Switching Register n (INOUTSEL0 to INOUTSELn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

INOUTSEL is valid only for the ports for which general input/output mode is selected by the general IO/interrupt switching register. Specifically, INOUTSEL selects either general input or general output mode for a port using the bit corresponding to the port number. The INOUTSEL bits can be written to only when the corresponding bits in the general IO/interrupt switching register are 0. Note that after general input/output mode is changed to interrupt input mode, INOUTSEL retains the setting but is read as 0.

[Hardware default value: H'0000_0000 = general input mode is selected for all the ports.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 31 to 0 | INOUTSEL [31:0] | H'0000_0000 | R/W | Selects either general input mode or general output mode for each port using the bits corresponding to the port numbers. 0: General input mode 1: General output mode |

Note: Unused bits should be set to the initial values.

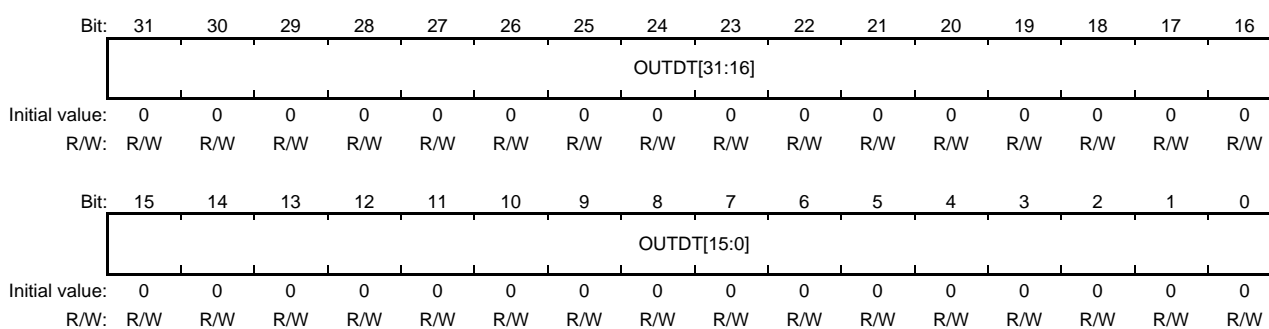
10.2.3 General Output Register n (OUTDT0 to OUTDTn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

OUTDT is valid only for the ports for which general input/output mode is selected by the general IO/interrupt switching register and then general output mode is selected by the general input/output switching register. Specifically, the value of the bit in OUTDT corresponding to the port number is inverted or not inverted depending on the setting of the positive/negative logic select register before being output from the corresponding port pin. Note that the polarity of the output signal should previously be set using the corresponding bit in the positive/negative logic select register. This register must be set after the output data select register is appropriately set to choose level of output data.

[Hardware default value: H'0000_0000 = 0 is output from all the ports.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|--|
| 31 to 0 | OUTDT[31:0] | H'0000_0000 | R/W | Allows the port to output the value set in the bit corresponding to the port number when the port is appropriately set by IOINTSEL0 to IOINTSELn, INOUTSEL0 to INOUTSELn and OUTDTSEL0 to OUTDTSELn. 0: 0 is output. 1: 1 is output. |

Note: The values set in OUTDT are not directly output from the GPIO pins; the above set values are processed according to the settings of the positive/negative logic select register before being output. Unused bits should be set to the initial values.

About write of output data: Output data can be selected exclusively of OUTDT or OUTDTH / OUTDTL, based on OUTDTSEL setting.

About read of output data: Either of OUTDT and OUTDTH / OUTDTL read as the actual output data.

10.2.4 General Input Register n (INDT0 to INDTn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

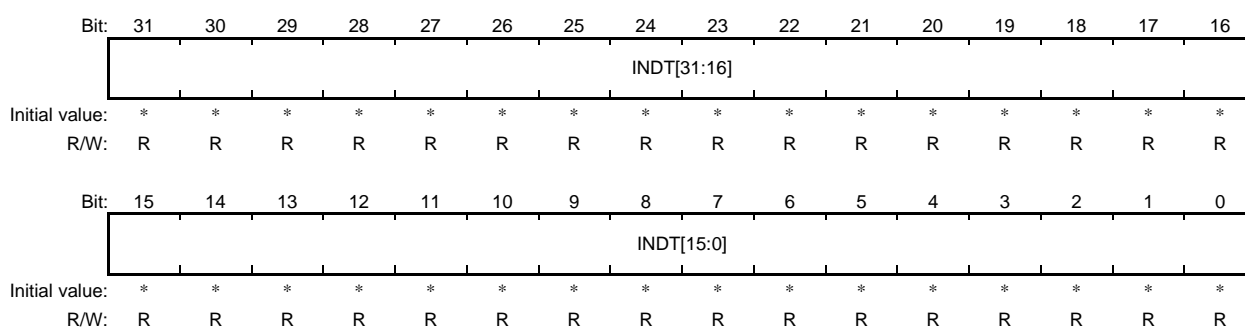
Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

INDT is a register that can read the status of General-Purpose Input / Output Ports.

Each bit reflects the value received through the corresponding port pin.

Note that when a bit in the positive/negative logic select register is 1, the corresponding bit in INDT indicates the inverted value of the input signal.

[Hardware default value: state of the signals input to the port pins.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 0 | INDT[31:0] | * | R | Each bit reflects the value received through the corresponding port pin. 0: Input is 0. (assuming positive logic) 1: Input is 1. (assuming positive logic) |

Note: Unused bits should be set to the initial values.

* State of the signals input to the port pins.

10.2.5 Interrupt Display Register n (INTDT0 to INTDTn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

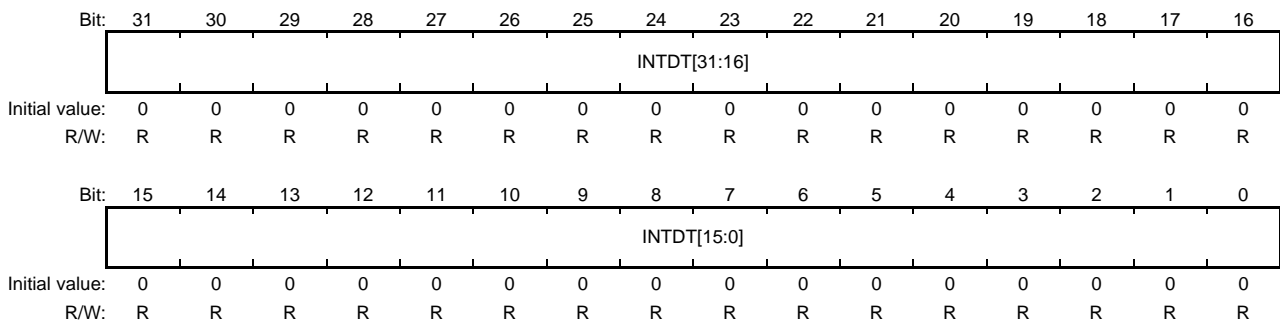
Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

INTDT is valid only when interrupt input mode is selected by the general IO/interrupt switching register. Specifically, when an interrupt is input via a port pin when INTDT is valid, the bit in INTDT corresponding to the port indicates whether the port has received an interrupt input or not. In interrupt input mode, the polarity and detection conditions (one edge/both edge or level) of the external input signal can be set for each port pin. Before using a port pin for interrupt input, the corresponding bits in the positive/negative logic select register and edge/level select register (one edge/both edge register should be appropriately configured if edge detection mode is selected) should be set, respectively.

If a port is set for edge detection using the corresponding bit in the edge/level select register, even when an external pulse interrupt signal is input, the corresponding bit in INTDT holds the input using the FF and allows the level interrupt signal to be output to the interrupt control block.

To stop all the interrupt signal outputs, all the bits in the interrupt clear register corresponding to the bits in INTDT currently indicating the reception of the corresponding interrupt signals should be cleared to 0. Note that if a port is set for level detection using the corresponding bit in the edge/level select register and an external level interrupt signal is input, the corresponding bit in INTDT does not use the FF to hold the input. Therefore, when an external input signal is stopped, the corresponding bit in INTDT is cleared automatically. When all the bits in INTDT are turned off (= 0), the GPIO stops outputting all the interrupt signals.

[Hardware default value: H'0000_0000 = no interrupt signals are input from ports.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|---|
| 31 to 0 | INTDT[31:0] | H'0000_0000 | R | Each bit indicates the input of an interrupt signal on the corresponding port pin. 0: No interrupt signal has been input. 1: Interrupt signal has been input. |

Conditions of Indicating Interrupt Input:

- For level-sensitive interrupt input (EDGLEVEL = 0)
 - External input signals are constantly monitored and indicated. (When the negative logic is selected, the inverted value of the external input signal is indicated.)
- For edge-sensitive interrupt input (EDGLEVEL = 1)
 - Clearing condition: When the interrupt clear register is cleared, indication is cleared regardless of the positive/negative logic select register.

- Setting condition: With the positive logic (POSNEG = 0), when the rising edge of an external interrupt signal is detected, the interrupt input is indicated. With the negative logic (POSNEG = 1), when the falling edge is detected, the interrupt input is indicated. With both edge mode (BOTHEDGE = 1), when either the rising or falling edge, the interrupt input is indicated.

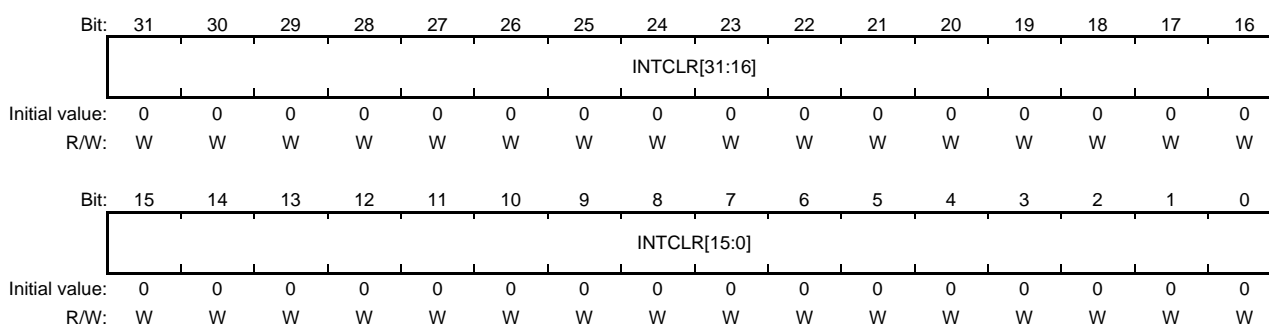
10.2.6 Interrupt Clear Register n (INTCLR0 to INTCLRn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

When the interrupt display register is currently indicates the reception of the interrupt input on the port for which the edge detection is selected by the edge/level select register (with configuring for one edge/both edge select register) in interrupt input mode, INTCLR clears the indication. Specifically, writing 1 to the bits in INTCLR corresponding to port numbers can clear the corresponding bits in the interrupt display register. However, when the interrupt display register is currently indicates the reception of the interrupt input on the port for which the level detection is selected by the edge/level select register, writing 1 to the corresponding bits in INTCLR cannot clear the corresponding bits in the interrupt display register. Only writing 1 to INTCLR is effective; INTCLR is always read as 0.

[Hardware default value: H'0000_0000 = interrupt indication is cleared for no ports.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 0 | INTCLR [31:0] | H'0000_0000 | W | Writing 1 to bits corresponding to port numbers clears the corresponding bits in the interrupt display register. 0: No effect 1: Interrupt display register bit is cleared. |

Note: Unused bits should be set to the initial values.

10.2.7 Interrupt Mask Register n (INTMSK0 to INTMSKn)

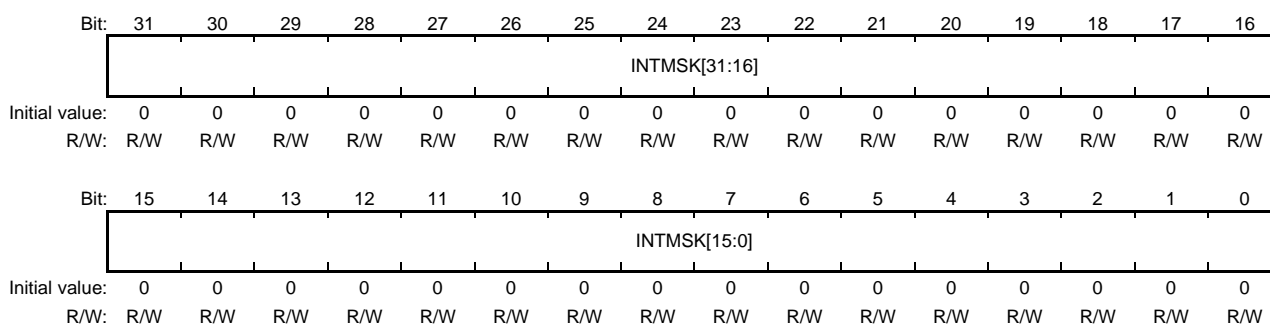
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

* 0 to 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], 0 to 6 [RZ/G2E]

INTMSK masks the interrupt requests indicated by the interrupt display register of GPIO.ch*. Interrupts can be separately masked using the corresponding bits in INTMSK. When all the bits currently indicating the reception of the interrupt signals are masked, no interrupt signals are output to the interrupt control block. Masks can be canceled by writing 1 to the corresponding bits in the interrupt mask clear register. Only writing 0 to this register is effective.

[Hardware default value: H'0000_0000 = all the ports are masked.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 31 to 0 | INTMSK [31:0] | H'0000_0000 | R/W | Setting a mask to the bit disables the corresponding interrupt signal to be output to the interrupt control block. 0: Interrupt is masked. 1: Interrupt is not masked. |

Note: Unused bits should be set to the initial values.

10.2.8 Interrupt Mask Clear Register n (MSKCLR0 to MSKCLRn)

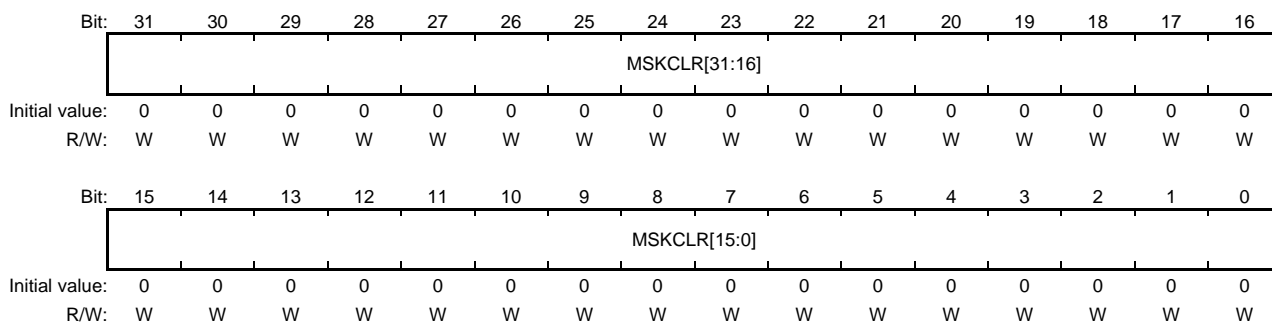
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

* 0 to 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], 0 to 6 [RZ/G2E]

MSKCLR cancels masks that are set by the interrupt mask register of GPIO.ch*. Each mask can be canceled (cleared) by writing 1 to the corresponding bit in MSKCLR. Only writing 1 to MSKCLR is effective; MSKCLR is always read as 0.

[Hardware default value: H'0000_0000 = interrupt masks are cleared for no ports.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 0 | MSKCLR [31:0] | H'0000_0000 | W | Setting a mask to the bit disables the corresponding interrupt signal to be output to the interrupt control block. 0: No effect 1: Interrupt is not masked. |

Note: Unused bits should be set to the initial values. (When GPIO is not selected by the pin multiplex settings, do not cancel the interrupt mask.)

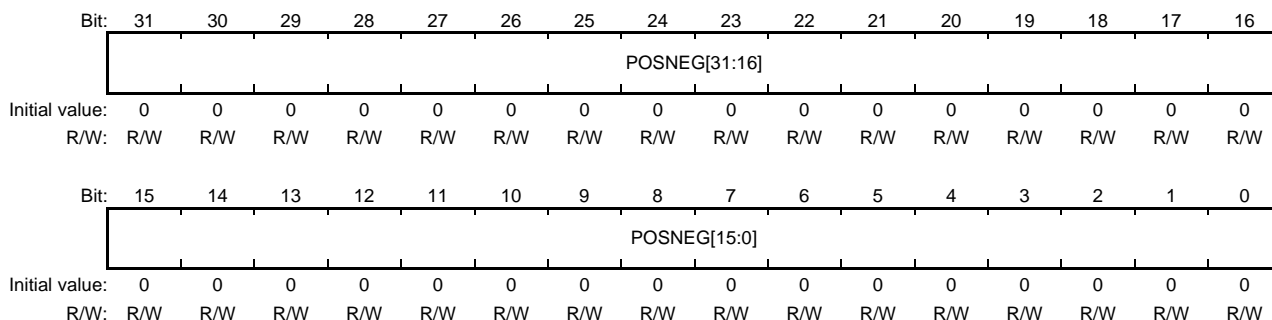
10.2.9 Positive/Negative Logic Select Register n (POSNEG0 to POSNEGn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

POSNEG selects the polarity (positive or negative logic) of each port pin in general input mode, general output mode, or interrupt input mode. POSNEG should be set before mode selection.

[Hardware default value: H'0000_0000 = positive logic is selected for all the ports.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 0 | POSNEG [31:0] | H'0000_0000 | R/W | Selects the polarity (positive or negative logic) of each port pin. 0: Positive logic 1: Negative logic |

Note: Unused bits should be set to the initial values.

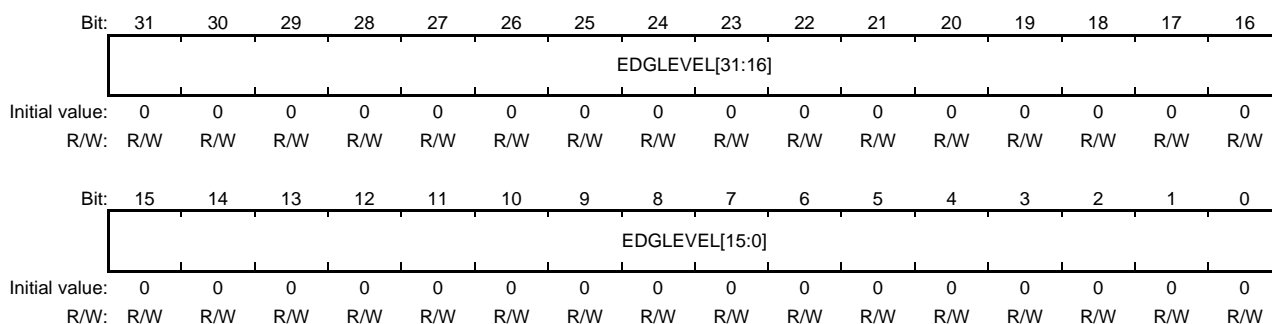
10.2.10 Edge/level Select Register n (EDGLEVEL0 to EDGLEVELn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

EDGLEVEL is valid only for the ports for which interrupt input mode is selected by the general IO/interrupt switching register. Specifically, EDGLEVEL selects the detection conditions (edge or level) of the interrupt input signal on each port pin for which interrupt input mode is selected. EDGLEVEL should be set before selection of interrupt input mode.

[Hardware default value: H'0000_0000 = level detection is selected for all the ports.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 31 to 0 | EDGLEVEL [31:0] | H'0000_0000 | R/W | Selects the level or edge as detection conditions of the interrupt input signal on each port pin for which interrupt input mode is selected. 0: Level 1: Edge |

Note: Unused bits should be set to the initial values.

10.2.11 Chattering Prevention On/Off Register n (FILONOFF0 to FILONOFFn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Notes: 1. n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]
 2. Some restrictions apply.

FILONOFF prevents chattering input to the port pins 0 to 3 of each GPIO block and controls frequency of filter clock (generated from peripheral clock S3D4φ) for chattering prevention function. For details, refer to section 10.3.3, Handling of Input Signals on Port Pins.

[Hardware default value: H'0000_0000 = chattering prevention function is turned off for all the ports.]

| | | | | | | | | | | | | | | | | |
|----------------|-------------|-----|----|----|----|----|----|----|----|----|----|----|---------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CLKSEL[1:0] | | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | FILONOFF[3:0] | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|---|
| 31, 30 | CLKSEL[1:0] | B'00 | R/W | Filter clock frequency setting B'00: S3D4φ /66000. B'01: S3D4φ /33000. B'10: S3D4φ /16500. B'11: S3D4φ /8250. |
| 29 to 4 | — | — | R | Reserved |
| 3 to 0 | FILONOFF [3:0] | B'0000 | R/W | Enables or disables the chattering prevention function. 0: Chattering prevention function is disabled. 1: Chattering prevention function is enabled. The bits FILONOFF[n] (n = 0 to 3) are used to control the port pin n. |

Note: Unused bits should be set to the initial values.

10.2.12 Interrupt Sub Mask Register n (INTMSKS0 to INTMSKSn)

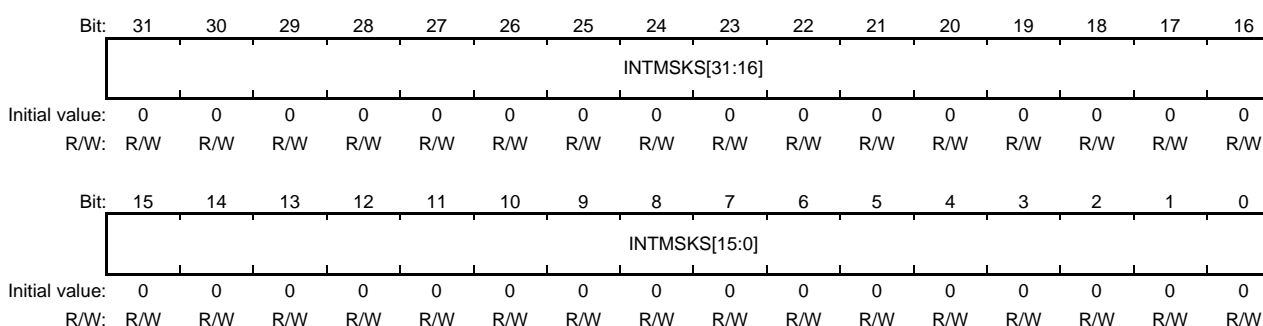
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

* 0 to 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], 0 to 6 [RZ/G2E]

INTMSKS masks the alternative interrupt requests indicated by the interrupt display register of GPIO.ch*A. Interrupts can be separately masked using the corresponding bits in INTMSKS. When all the bits currently indicating the reception of the interrupt signals are masked, no alternative interrupt signals are output to the interrupt control block. Masks can be canceled by writing 1 to the corresponding bits in the interrupt sub mask clear register. Only writing 0 to this register is effective.

[Hardware default value: H'0000_0000 = all the ports are masked.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|--|
| 31 to 0 | INTMSKS [31:0] | H'0000_0000 | R/W | Setting a mask to the bit disables the corresponding alternative interrupt signal to be output to the interrupt control block. 0: Interrupt is masked. 1: Interrupt is not masked. |

Note: Unused bits should be set to the initial values.

10.2.13 Interrupt Sub Mask Clear Register n (MSKCLRS0 to MSKCLRSn)

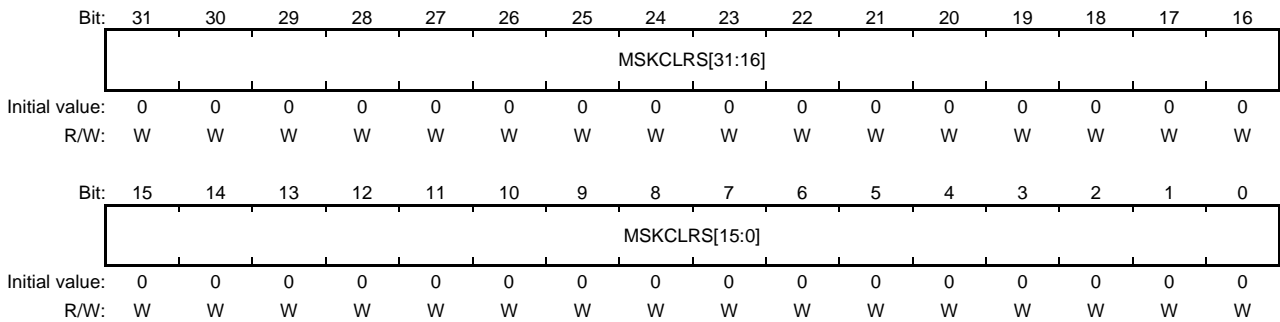
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

* 0 to 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], 0 to 6 [RZ/G2E]

MSKCLRS cancels masks that are set by the interrupt sub mask register of GPIO.ch*A. Each mask can be canceled (cleared) by writing 1 to the corresponding bit in MSKCLRS. Only writing 1 to MSKCLRS is effective; MSKCLRS is always read as 0.

[Hardware default value: H'0000_0000 = alternative interrupt masks are cleared for no ports.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|---|
| 31 to 0 | MSKCLRS [31:0] | H'0000_0000 | W | Setting a mask to the bit disables the corresponding alternative interrupt signal to be output to the interrupt control block. 0: No effect 1: Interrupt is not masked. |

Note: Unused bits should be set to the initial values. (When GPIO is not selected by the pin multiplex settings, do not cancel the alternative interrupt mask.)

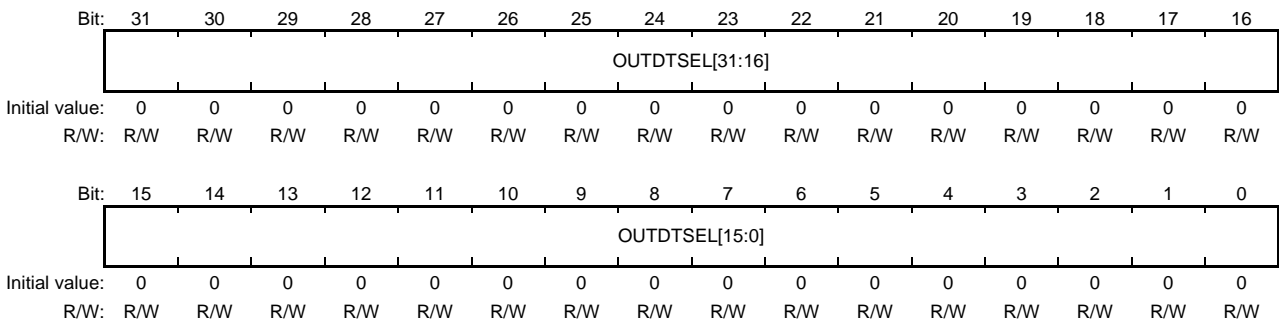
10.2.14 Output Data Select Register n (OUTDTSEL0 to OUTDTSELn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

OUTDTSEL is valid only for the ports for which general input/output mode is selected by the general IO/interrupt switching register and then general output mode is selected by the general input/output switching register. OUTDTSEL selects if OUTDT or OUTDTH/OUTDTL will be the output data of GPIO. When choosing OUTDT, configuration is performed as described in section 10.2.3, General Output Register n (OUTDT0 to OUTDTn). When choosing OUTDTH/OUTDTL, output data will be output by writing the appropriate data to the corresponding bits in OUTDTH or OUTDTL. Note that the polarity of the output signal should be previously set using the corresponding bit in the positive/negative logic select register. Furthermore, this register should be set before writing data to OUTDTH/OUTDTL registers.

[Hardware default value: H'0000_0000 = Out data register is used to output data.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 31 to 0 | OUTDTSEL [31:0] | H'0000_0000 | R/W | Choosing whether output data is output by general output register OUTDT or output data high register OUTDTH/output data low register OUTDTL. 0: General output register is used to output the data. 1: Output data high register and output data low register is used to output the data. |

Note: Unused bits should be set to the initial values.

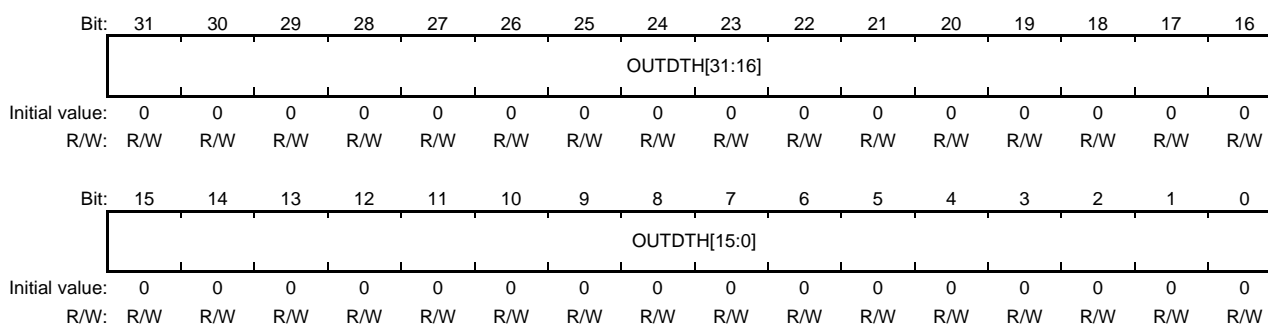
10.2.15 Output Data High Register n (OUTDTH0 to OUTDTHn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

OUTDTH is valid only for the ports for which general input/output mode is selected by the general IO/interrupt switching register and then general output mode is selected by the general input/output switching register, and the output data select register OUTDTSEL is configured to choose OUTDTH/OUTDTL register to output the data of GPIO. Only writing 1 to OUTDTH is effective. Otherwise, setting makes no changes. Reading OUTDTH returns the values of the latest data set to OUTDTH or OUTDTL right before that. Note that the polarity of the output signal should be previously set using the corresponding bit in the positive/negative logic select register. Furthermore, this register should be written and read after output data select register OUTDTSEL is set. Reading OUTDTH without appropriately configuring OUTDTSEL can return value of OUTDT register.

[Hardware default value: H'0000_0000 = 0 is output from all the ports with setting OUTDTSEL.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 0 | OUTDTH [31:0] | H'0000_0000 | R/W | Outputting high value data. 0: Invalid data. 1: Valid data. |

Note: Unused bits should be set to the initial values.

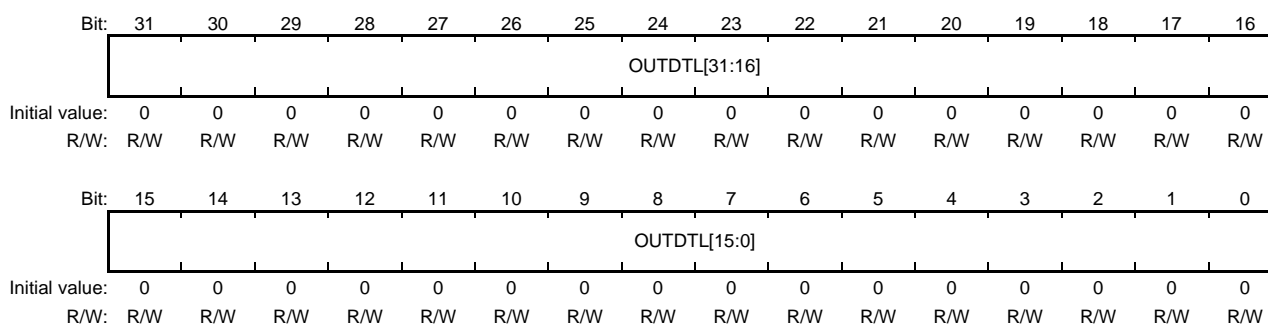
10.2.16 Output Data Low Register n (OUTDTL0 to OUTDTLn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

OUTDTL is valid only for the ports for which general input/output mode is selected by the general IO/interrupt switching register and then general output mode is selected by the general input/output switching register, and the output data select register OUTDTSEL is configured to choose OUTDTH/OUTDTL register to output the data of GPIO. Only writing 0 to OUTDTL is effective. Otherwise, setting makes no changes. Reading OUTDTL returns the values of the latest data set to OUTDTL or OUTDTH right before that. Note that the polarity of the output signal should be previously set using the corresponding bit in the positive/negative logic select register. Furthermore, this register should be written or read after output data select registers OUTDTSEL is set. Reading OUTDTH without appropriately configuring OUTDTSEL can return value of OUTDT register.

[Hardware default value: H'0000_0000 = 0 is output from all the ports with setting OUTDTSEL.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 31 to 0 | OUTDTL [31:0] | H'0000_0000 | R/W | Outputting low value data. 0: Valid data. 1: Invalid data. |

Note: Unused bits should be set to the initial values.

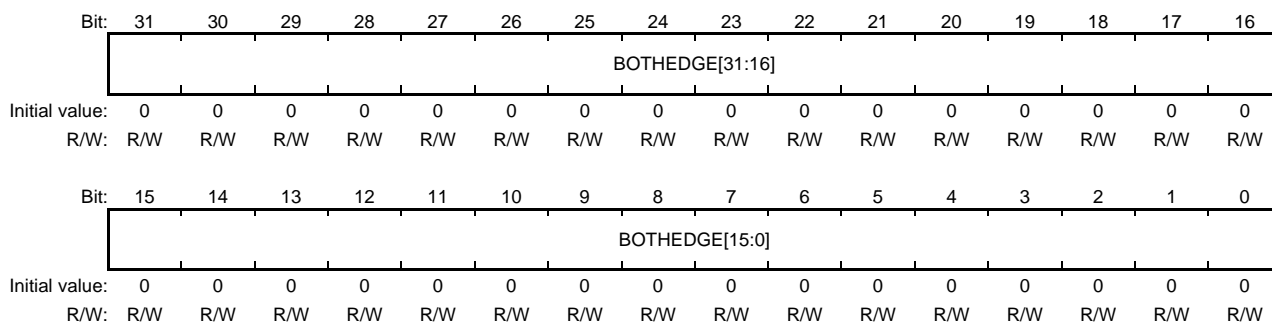
10.2.17 One Edge/Both Edge Select Register n (BOTHEDGE0 to BOTHEDGEN)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: n = 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 6 [RZ/G2E]

BOTHEDGE is valid only when the edge detection mode is selected by the edge/level select registers. Specially, BOTHEDGE selects the detection condition (one edge or both edges) of the interrupt input signal on each port pin for which interrupt input mode (selected by the general IO/interrupt switching registers) and edge detection mode are selected. BOTHEDGE should be set before selection of interrupt input mode.

[Hardware default value: H'0000_0000 = both edge detection mode is disabled for all the ports.]



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 31 to 0 | BOTHEDGE [31:0] | H'0000_0000 | R/W | Selecting one edge or both edge detection condition of the interrupt input signal on each port pin for which interrupt input mode and edge detection mode are selected. 0: One edge. 1: Both edges. |

Note: Unused bits should be set to the initial values.

10.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

10.3.1 Port Pin Specifications

Each GPIO block is provided with up to 32 port pins for general input/output and external interrupt input ports. Table 10.4 specifies these pins.

Table 10.4 Port Pin Specifications

| Block | Number | Abbreviation | Name | Applicable | | | | Descriptions |
|--|--------|--------------|-----------------------------|------------|----------------------------|--------|--------|---|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| GPIO-0 Applicable registers: IOINTSEL0 INOUTSEL0 OUTDT0 INDT0 INTDT0 INTCLR0 INTMSK0 MSKCLR0 POSNEG0 EDGLEVELO FILONOFF0 INTMSKS0 MSKCLRS0 OUTDTSEL0 OUTDTH0 OUTDTL0 BOTHEDGE0 | 1 | GP0_00 | IO/interrupt input port A0 | √ | √ | √ | √ | <ul style="list-style-type: none"> • Either general input/output mode or interrupt input mode can be set for each port. • In general input mode, the polarity of input signals can be set for each port. • In general output mode, the polarity of output signals can be set for each port. • In interrupt input mode, the polarity of interrupt signal can be set for each port. • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 2 | GP0_01 | IO/interrupt input port A1 | √ | √ | √ | √ | |
| | 3 | GP0_02 | IO/interrupt input port A2 | √ | √ | √ | √ | |
| | 4 | GP0_03 | IO/interrupt input port A3 | √ | √ | √ | √ | |
| | 5 | GP0_04 | IO/interrupt input port A4 | √ | √ | √ | √ | |
| | 6 | GP0_05 | IO/interrupt input port A5 | √ | √ | √ | √ | |
| | 7 | GP0_06 | IO/interrupt input port A6 | √ | √ | √ | √ | |
| | 8 | GP0_07 | IO/interrupt input port A7 | √ | √ | √ | √ | |
| | 9 | GP0_08 | IO/interrupt input port A8 | √ | √ | √ | √ | |
| | 10 | GP0_09 | IO/interrupt input port A9 | √ | √ | √ | √ | |
| | 11 | GP0_10 | IO/interrupt input port A10 | √ | √ | √ | √ | |
| | 12 | GP0_11 | IO/interrupt input port A11 | √ | √ | √ | √ | |
| | 13 | GP0_12 | IO/interrupt input port A12 | √ | √ | √ | √ | |
| | 14 | GP0_13 | IO/interrupt input port A13 | √ | √ | √ | √ | |
| | 15 | GP0_14 | IO/interrupt input port A14 | √ | √ | √ | √ | |
| | 16 | GP0_15 | IO/interrupt input port A15 | √ | √ | √ | √ | |
| | 17 | GP0_16 | IO/interrupt input port A16 | — | — | — | √ | |
| | 18 | GP0_17 | IO/interrupt input port A17 | — | — | — | √ | |
| | 19 | GP0_18 | IO/interrupt input port A18 | — | — | — | — | |
| | 20 | GP0_19 | IO/interrupt input port A19 | — | — | — | — | |
| | 21 | GP0_20 | IO/interrupt input port A20 | — | — | — | — | |
| | 22 | GP0_21 | IO/interrupt input port A21 | — | — | — | — | |

| Block | Number | Abbreviation | Name | Applicable | | | | Descriptions |
|-----------------------|--------|--------------|-----------------------------|------------|----------------------------|--------|--------|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| GPIO-1 | 23 | GP1_00 | IO/interrupt input port B0 | √ | √ | √ | √ | • Either general input/output mode or interrupt input mode can be set for each port. |
| | 24 | GP1_01 | IO/interrupt input port B1 | √ | √ | √ | √ | |
| Applicable registers: | 25 | GP1_02 | IO/interrupt input port B2 | √ | √ | √ | √ | • In general input mode, the polarity of input signals can be set for each port. |
| IOINTSEL1 | 26 | GP1_03 | IO/interrupt input port B3 | √ | √ | √ | √ | |
| INOUTSEL1 | 27 | GP1_04 | IO/interrupt input port B4 | √ | √ | √ | √ | • In general output mode, the polarity of output signals can be set for each port. |
| OUTDT1 | 28 | GP1_05 | IO/interrupt input port B5 | √ | √ | √ | √ | |
| INDT1 | 29 | GP1_06 | IO/interrupt input port B6 | √ | √ | √ | √ | • In interrupt input mode, the polarity of interrupt signal can be set for each port. |
| INTDT1 | 30 | GP1_07 | IO/interrupt input port B7 | √ | √ | √ | √ | |
| INTCLR1 | 31 | GP1_08 | IO/interrupt input port B8 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| INTMSK1 | 32 | GP1_09 | IO/interrupt input port B9 | √ | √ | √ | √ | |
| MSKCLR1 | 33 | GP1_10 | IO/interrupt input port B10 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| POSNEG1 | 34 | GP1_11 | IO/interrupt input port B11 | √ | √ | √ | √ | |
| EDGLEVEL1 | 35 | GP1_12 | IO/interrupt input port B12 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| FILONOFF1 | 36 | GP1_13 | IO/interrupt input port B13 | √ | √ | √ | √ | |
| INTMSKS1 | 37 | GP1_14 | IO/interrupt input port B14 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| MSKCLRS1 | 38 | GP1_15 | IO/interrupt input port B15 | √ | √ | √ | √ | |
| OUTDTSEL1 | 39 | GP1_16 | IO/interrupt input port B16 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| OUTDTH1 | 40 | GP1_17 | IO/interrupt input port B17 | √ | √ | √ | √ | |
| BOTHEDGE1 | 41 | GP1_18 | IO/interrupt input port B18 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 42 | GP1_19 | IO/interrupt input port B19 | √ | √ | √ | √ | |
| | 43 | GP1_20 | IO/interrupt input port B20 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 44 | GP1_21 | IO/interrupt input port B21 | √ | √ | √ | √ | |
| | 45 | GP1_22 | IO/interrupt input port B22 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 46 | GP1_23 | IO/interrupt input port B23 | √ | √ | √ | — | |
| | 47 | GP1_24 | IO/interrupt input port B24 | √ | √ | √ | — | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 48 | GP1_25 | IO/interrupt input port B25 | √ | √ | √ | — | |
| | 49 | GP1_26 | IO/interrupt input port B26 | √ | √ | √ | — | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 50 | GP1_27 | IO/interrupt input port B27 | √ | √ | √ | — | |
| | 51 | GP1_28 | IO/interrupt input port B28 | √ | √ | √ | — | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 52 | GP1_29 | IO/interrupt input port B29 | — | — | — | — | |
| | 53 | GP1_30 | IO/interrupt input port B30 | — | — | — | — | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 54 | GP1_31 | IO/interrupt input port B31 | — | — | — | — | |

| Block | Number | Abbreviation | Name | Applicable | | | | Descriptions |
|-----------------------|--------|--------------|-----------------------------|------------|----------------------------|--------|--------|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| GPIO-2 | 55 | GP2_00 | IO/interrupt input port C0 | √ | √ | √ | √ | • Either general input/output mode or interrupt input mode can be set for each port. |
| | 56 | GP2_01 | IO/interrupt input port C1 | √ | √ | √ | √ | |
| Applicable registers: | 57 | GP2_02 | IO/interrupt input port C2 | √ | √ | √ | √ | • In general input mode, the polarity of input signals can be set for each port. |
| | 58 | GP2_03 | IO/interrupt input port C3 | √ | √ | √ | √ | |
| IOINTSEL2 | 59 | GP2_04 | IO/interrupt input port C4 | √ | √ | √ | √ | • In general output mode, the polarity of output signals can be set for each port. |
| INOUTSEL2 | 60 | GP2_05 | IO/interrupt input port C5 | √ | √ | √ | √ | |
| OUTDT2 | 61 | GP2_06 | IO/interrupt input port C6 | √ | √ | √ | √ | • In interrupt input mode, the polarity of interrupt signal can be set for each port. |
| INDT2 | 62 | GP2_07 | IO/interrupt input port C7 | √ | √ | √ | √ | |
| INTDT2 | 63 | GP2_08 | IO/interrupt input port C8 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| INTCLR2 | 64 | GP2_09 | IO/interrupt input port C9 | √ | √ | √ | √ | |
| INTMSK2 | 65 | GP2_10 | IO/interrupt input port C10 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| MSKCLR2 | 66 | GP2_11 | IO/interrupt input port C11 | √ | √ | √ | √ | |
| POSNEG2 | 67 | GP2_12 | IO/interrupt input port C12 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| EDGLEVEL2 | 68 | GP2_13 | IO/interrupt input port C13 | √ | √ | √ | √ | |
| FILONOFF2 | 69 | GP2_14 | IO/interrupt input port C14 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| INTMSKS2 | 70 | GP2_15 | IO/interrupt input port C15 | — | — | — | √ | |
| MSKCLRS2 | 71 | GP2_16 | IO/interrupt input port C16 | — | — | — | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| OUTDTSEL2 | 72 | GP2_17 | IO/interrupt input port C17 | — | — | — | √ | |
| OUTDTH2 | 73 | GP2_18 | IO/interrupt input port C18 | — | — | — | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| OUTDTL2 | 74 | GP2_19 | IO/interrupt input port C19 | — | — | — | √ | |
| BOTHEDGE2 | 75 | GP2_20 | IO/interrupt input port C20 | — | — | — | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 76 | GP2_21 | IO/interrupt input port C21 | — | — | — | √ | |
| | 77 | GP2_22 | IO/interrupt input port C22 | — | — | — | √ | |
| | 78 | GP2_23 | IO/interrupt input port C23 | — | — | — | √ | |
| | 79 | GP2_24 | IO/interrupt input port C24 | — | — | — | √ | |
| | 80 | GP2_25 | IO/interrupt input port C25 | — | — | — | √ | |
| | 81 | GP2_26 | IO/interrupt input port C26 | — | — | — | — | |
| | 82 | GP2_27 | IO/interrupt input port C27 | — | — | — | — | |
| | 83 | GP2_28 | IO/interrupt input port C28 | — | — | — | — | |
| | 84 | GP2_29 | IO/interrupt input port C29 | — | — | — | — | |
| | 85 | GP2_30 | IO/interrupt input port C30 | — | — | — | — | |
| | 86 | GP2_31 | IO/interrupt input port C31 | — | — | — | — | |

| Block | Number | Abbreviation | Name | Applicable | | | | Descriptions |
|-----------------------|--------|--------------|-----------------------------|------------|----------------------------|--------|--------|---|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| GPIO-3 | 87 | GP3_00 | IO/interrupt input port D0 | √ | √ | √ | √ | <ul style="list-style-type: none"> • Either general input/output mode or interrupt input mode can be set for each port. • In general input mode, the polarity of input signals can be set for each port. • In general output mode, the polarity of output signals can be set for each port. • In interrupt input mode, the polarity of interrupt signal can be set for each port. • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 88 | GP3_01 | IO/interrupt input port D1 | √ | √ | √ | √ | |
| Applicable registers: | 89 | GP3_02 | IO/interrupt input port D2 | √ | √ | √ | √ | |
| IOINTSEL3 | 90 | GP3_03 | IO/interrupt input port D3 | √ | √ | √ | √ | |
| INOUTSEL3 | 91 | GP3_04 | IO/interrupt input port D4 | √ | √ | √ | √ | |
| OUTDT3 | 92 | GP3_05 | IO/interrupt input port D5 | √ | √ | √ | √ | |
| INDT3 | 93 | GP3_06 | IO/interrupt input port D6 | √ | √ | √ | √ | |
| INTDT3 | 94 | GP3_07 | IO/interrupt input port D7 | √ | √ | √ | √ | |
| INTCLR3 | 95 | GP3_08 | IO/interrupt input port D8 | √ | √ | √ | √ | |
| INTMSK3 | 96 | GP3_09 | IO/interrupt input port D9 | √ | √ | √ | √ | |
| MSKCLR3 | 97 | GP3_10 | IO/interrupt input port D10 | √ | √ | √ | √ | |
| POSNEG3 | 98 | GP3_11 | IO/interrupt input port D11 | √ | √ | √ | √ | |
| EDGLEVEL3 | 99 | GP3_12 | IO/interrupt input port D12 | √ | √ | √ | √ | |
| FILONOFF3 | 100 | GP3_13 | IO/interrupt input port D13 | √ | √ | √ | √ | |
| INTMSKS3 | 101 | GP3_14 | IO/interrupt input port D14 | √ | √ | √ | √ | |
| MSKCLRS3 | 102 | GP3_15 | IO/interrupt input port D15 | √ | √ | √ | √ | |
| OUTDTSEL3 | 103 | GP3_16 | IO/interrupt input port D16 | — | — | — | — | |
| OUTDTH3 | | | | | | | | |
| OUTDTL3 | | | | | | | | |
| BOTHEDGE3 | | | | | | | | |

| Block | Number | Abbreviation | Name | Applicable | | | | Descriptions |
|-----------------------|--------|--------------|-----------------------------|------------|----------------------------|--------|--------|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| GPIO-4 | 104 | GP4_00 | IO/interrupt input port E0 | √ | √ | √ | √ | • Either general input/output mode or interrupt input mode can be set for each port. |
| | 105 | GP4_01 | IO/interrupt input port E1 | √ | √ | √ | √ | |
| Applicable registers: | 106 | GP4_02 | IO/interrupt input port E2 | √ | √ | √ | √ | • In general input mode, the polarity of input signals can be set for each port. |
| | 107 | GP4_03 | IO/interrupt input port E3 | √ | √ | √ | √ | |
| IOINTSEL4 | 108 | GP4_04 | IO/interrupt input port E4 | √ | √ | √ | √ | • In general output mode, the polarity of output signals can be set for each port. |
| INOUTSEL4 | 109 | GP4_05 | IO/interrupt input port E5 | √ | √ | √ | √ | |
| OUTDT4 | 110 | GP4_06 | IO/interrupt input port E6 | √ | √ | √ | √ | • In interrupt input mode, the polarity of interrupt signal can be set for each port. |
| INDT4 | 111 | GP4_07 | IO/interrupt input port E7 | √ | √ | √ | √ | |
| INTDT4 | 112 | GP4_08 | IO/interrupt input port E8 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| INTCLR4 | 113 | GP4_09 | IO/interrupt input port E9 | √ | √ | √ | √ | |
| INTMSK4 | 114 | GP4_10 | IO/interrupt input port E10 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| MSKCLR4 | 115 | GP4_11 | IO/interrupt input port E11 | √ | √ | √ | — | |
| POSNEG4 | 116 | GP4_12 | IO/interrupt input port E12 | √ | √ | √ | — | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| EDGLEVEL4 | 117 | GP4_13 | IO/interrupt input port E13 | √ | √ | √ | — | |
| FILONOFF4 | 118 | GP4_14 | IO/interrupt input port E14 | √ | √ | √ | — | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| INTMSKS4 | 119 | GP4_15 | IO/interrupt input port E15 | √ | √ | √ | — | |
| MSKCLRS4 | 120 | GP4_16 | IO/interrupt input port E16 | √ | √ | √ | — | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| OUTDTSEL4 | 121 | GP4_17 | IO/interrupt input port E17 | √ | √ | √ | — | |
| OUTDTH4 | 122 | GP4_18 | IO/interrupt input port E18 | — | — | — | — | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| OUTDTL4 | 123 | GP4_19 | IO/interrupt input port E19 | — | — | — | — | |
| BOTHEDGE4 | 124 | GP4_20 | IO/interrupt input port E20 | — | — | — | — | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 125 | GP4_21 | IO/interrupt input port E21 | — | — | — | — | |
| | 126 | GP4_22 | IO/interrupt input port E22 | — | — | — | — | |
| | 127 | GP4_23 | IO/interrupt input port E23 | — | — | — | — | |
| | 128 | GP4_24 | IO/interrupt input port E24 | — | — | — | — | |
| | 129 | GP4_25 | IO/interrupt input port E25 | — | — | — | — | |
| | 130 | GP4_26 | IO/interrupt input port E26 | — | — | — | — | |
| | 131 | GP4_27 | IO/interrupt input port E27 | — | — | — | — | |
| | 132 | GP4_28 | IO/interrupt input port E28 | — | — | — | — | |
| | 133 | GP4_29 | IO/interrupt input port E29 | — | — | — | — | |
| | 134 | GP4_30 | IO/interrupt input port E30 | — | — | — | — | |
| | 135 | GP4_31 | IO/interrupt input port E31 | — | — | — | — | |

| Block | Number | Abbreviation | Name | Applicable | | | | Descriptions |
|-----------------------|--------|--------------|-----------------------------|------------|----------------------------|--------|--------|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| GPIO-5 | 136 | GP5_00 | IO/interrupt input port F0 | √ | √ | √ | √ | • Either general input/output mode or interrupt input mode can be set for each port. |
| | 137 | GP5_01 | IO/interrupt input port F1 | √ | √ | √ | √ | |
| Applicable registers: | 138 | GP5_02 | IO/interrupt input port F2 | √ | √ | √ | √ | • In general input mode, the polarity of input signals can be set for each port. |
| IOINTSEL5 | 139 | GP5_03 | IO/interrupt input port F3 | √ | √ | √ | √ | |
| INOUTSEL5 | 140 | GP5_04 | IO/interrupt input port F4 | √ | √ | √ | √ | • In general output mode, the polarity of output signals can be set for each port. |
| OUTDT5 | 141 | GP5_05 | IO/interrupt input port F5 | √ | √ | √ | √ | |
| INDT5 | 142 | GP5_06 | IO/interrupt input port F6 | √ | √ | √ | √ | • In interrupt input mode, the polarity of interrupt signal can be set for each port. |
| INTDT5 | 143 | GP5_07 | IO/interrupt input port F7 | √ | √ | √ | √ | |
| INTCLR5 | 144 | GP5_08 | IO/interrupt input port F8 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| INTMSK5 | 145 | GP5_09 | IO/interrupt input port F9 | √ | √ | √ | √ | |
| MSKCLR5 | 146 | GP5_10 | IO/interrupt input port F10 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| POSNEG5 | 147 | GP5_11 | IO/interrupt input port F11 | √ | √ | √ | √ | |
| EDGLEVEL5 | 148 | GP5_12 | IO/interrupt input port F12 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| FILONOFF5 | 149 | GP5_13 | IO/interrupt input port F13 | √ | √ | √ | √ | |
| INTMSKS5 | 150 | GP5_14 | IO/interrupt input port F14 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| MSKCLR5 | 151 | GP5_15 | IO/interrupt input port F15 | √ | √ | √ | √ | |
| OUTDTSEL5 | 152 | GP5_16 | IO/interrupt input port F16 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| OUTDTH5 | 153 | GP5_17 | IO/interrupt input port F17 | √ | √ | √ | √ | |
| OUTDTL5 | 154 | GP5_18 | IO/interrupt input port F18 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| BOTHEDGE5 | 155 | GP5_19 | IO/interrupt input port F19 | √ | √ | √ | √ | |
| | 156 | GP5_20 | IO/interrupt input port F20 | √ | √ | √ | — | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 157 | GP5_21 | IO/interrupt input port F21 | √ | √ | √ | — | |
| | 158 | GP5_22 | IO/interrupt input port F22 | √ | √ | √ | — | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 159 | GP5_23 | IO/interrupt input port F23 | √ | √ | √ | — | |
| | 160 | GP5_24 | IO/interrupt input port F24 | √ | √ | √ | — | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 161 | GP5_25 | IO/interrupt input port F25 | √ | √ | √ | — | |

| Block | Number | Abbreviation | Name | Applicable | | | | Descriptions |
|-----------------------|--------|-----------------------------|-----------------------------|------------|----------------------------|--------|--------|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| GPIO-6 | 162 | GP6_00 | IO/interrupt input port G0 | √ | √ | √ | √ | • Either general input/output mode or interrupt input mode can be set for each port. |
| | 163 | GP6_01 | IO/interrupt input port G1 | √ | √ | √ | √ | |
| Applicable registers: | 164 | GP6_02 | IO/interrupt input port G2 | √ | √ | √ | √ | • In general input mode, the polarity of input signals can be set for each port. |
| | 165 | GP6_03 | IO/interrupt input port G3 | √ | √ | √ | √ | |
| IOINTSEL6 | 166 | GP6_04 | IO/interrupt input port G4 | √ | √ | √ | √ | • In general output mode, the polarity of output signals can be set for each port. |
| INOUTSEL6 | 167 | GP6_05 | IO/interrupt input port G5 | √ | √ | √ | √ | |
| OUTDT6 | 168 | GP6_06 | IO/interrupt input port G6 | √ | √ | √ | √ | • In interrupt input mode, the polarity of interrupt signal can be set for each port. |
| INDT6 | 169 | GP6_07 | IO/interrupt input port G7 | √ | √ | √ | √ | |
| INTDT6 | 170 | GP6_08 | IO/interrupt input port G8 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| INTCLR6 | 171 | GP6_09 | IO/interrupt input port G9 | √ | √ | √ | √ | |
| INTMSK6 | 172 | GP6_10 | IO/interrupt input port G10 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| MSKCLR6 | 173 | GP6_11 | IO/interrupt input port G11 | √ | √ | √ | √ | |
| POSNEG6 | 174 | GP6_12 | IO/interrupt input port G12 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| EDGLEVEL6 | 175 | GP6_13 | IO/interrupt input port G13 | √ | √ | √ | √ | |
| FILONOFF6 | 176 | GP6_14 | IO/interrupt input port G14 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| INTMSKS6 | 177 | GP6_15 | IO/interrupt input port G15 | √ | √ | √ | √ | |
| MSKCLRS6 | 178 | GP6_16 | IO/interrupt input port G16 | √ | √ | √ | √ | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| OUTDTSEL6 | 179 | GP6_17 | IO/interrupt input port G17 | √ | √ | √ | √ | |
| OUTDTH6 | 180 | GP6_18 | IO/interrupt input port G18 | √ | √ | √ | — | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| OUTDTL6 | 181 | GP6_19 | IO/interrupt input port G19 | √ | √ | √ | — | |
| BOTHEDGE6 | 182 | GP6_20 | IO/interrupt input port G20 | √ | √ | √ | — | • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| | 183 | GP6_21 | IO/interrupt input port G21 | √ | √ | √ | — | |
| | 184 | GP6_22 | IO/interrupt input port G22 | √ | √ | √ | — | |
| | 185 | GP6_23 | IO/interrupt input port G23 | √ | √ | √ | — | |
| | 186 | GP6_24 | IO/interrupt input port G24 | √ | √ | √ | — | |
| | 187 | GP6_25 | IO/interrupt input port G25 | √ | √ | √ | — | |
| | 188 | GP6_26 | IO/interrupt input port G26 | √ | √ | √ | — | |
| | 189 | GP6_27 | IO/interrupt input port G27 | √ | √ | √ | — | |
| | 190 | GP6_28 | IO/interrupt input port G28 | √ | √ | √ | — | |
| | 191 | GP6_29 | IO/interrupt input port G29 | √ | √ | √ | — | |
| | 192 | GP6_30 | IO/interrupt input port G30 | √ | √ | √ | — | |
| 193 | GP6_31 | IO/interrupt input port G31 | √ | √ | √ | — | | |

| Block | Number | Abbreviation | Name | Applicable | | | | Descriptions |
|-----------------------|--------|--------------|----------------------------|------------|----------------------------|--------|--------|---|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| GPIO-7 | 194 | GP7_00 | IO/interrupt input port H0 | √ | √ | √ | — | <ul style="list-style-type: none"> • Either general input/output mode or interrupt input mode can be set for each port. |
| | 195 | GP7_01 | IO/interrupt input port H1 | √ | √ | √ | — | |
| | 196 | GP7_02 | IO/interrupt input port H2 | √ | √ | √ | — | |
| | 197 | GP7_03 | IO/interrupt input port H3 | √ | √ | √ | — | |
| Applicable registers: | | | | | | | | <ul style="list-style-type: none"> • In general input mode, the polarity of input signals can be set for each port. • In general output mode, the polarity of output signals can be set for each port. • In interrupt input mode, the polarity of interrupt signal can be set for each port. • In interrupt input mode, the detection conditions of interrupt signals can be set for each port. |
| IOINTSEL7 | | | | | | | | |
| INOUTSEL7 | | | | | | | | |
| OUTDT7 | | | | | | | | |
| INDT7 | | | | | | | | |
| INTDT7 | | | | | | | | |
| INTCLR7 | | | | | | | | |
| INTMSK7 | | | | | | | | |
| MSKCLR7 | | | | | | | | |
| POSNEG7 | | | | | | | | |
| EDGLEVEL7 | | | | | | | | |
| FILONOFF7 | | | | | | | | |
| INTMSKS7 | | | | | | | | |
| MSKCLRS7 | | | | | | | | |
| OUTDTSEL7 | | | | | | | | |
| OUTDTH7 | | | | | | | | |
| OUTDTL7 | | | | | | | | |
| BOTHEDGE7 | | | | | | | | |

Note: "—" indicates the port pin is not in use.

10.3.2 Operations in Each Mode

10.3.2.1 Mode Switching

Two registers are used to switch modes of the general IO/interrupt input pins of the GPIO blocks. Each register is provided with up to 32 bits each controlling one of the GPIO_n* port pins. The general IO/interrupt switching register is first used to select either general input/output mode or interrupt input mode for each port pin. When general input/output mode is selected, the setting of the relevant bit in the second register, i.e., the general input/output switching register, is used. Specifically, when a bit in the general input/output switching register is set for general output mode, the corresponding port pin is turned to the output direction and the route is formed so that the set value in the corresponding bit in the general output register should be output via the pin. Likewise, when set for the general input mode, the corresponding port pin is turned to the input direction and the route is formed so that the value received via the pin should be indicated by the corresponding bit in the general input register. When interrupt input mode is selected, the corresponding port pin is turned to the input direction and the route is formed so that the reception of the signal input via the pin should be indicated by the interrupt display register. Here, the setting of the second register, i.e., the general input/output switching register, is invalid.

Note: * n = 0 to 7 [RZ/G2H], [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2N], n = 0 to 6 [RZ/G2E]

10.3.2.2 General Input/Output Mode

When a port is set for general input/output mode using the corresponding bit in the general IO/interrupt switching register, the corresponding port serves as a general input/output pin. In general input/output mode, either mode can be selected using the corresponding bit in the general input/output switching register. When a port is set for general output mode, the port outputs the value set in the corresponding bit in the general output register or output data high/output data low register with appropriate configuring in output data select register. Here, the polarity of the actual output signal is determined by the setting of the corresponding bit in the positive/negative logic select register. When a port is set for general input mode, the polarity of the input signal is also determined by the setting of the corresponding bit in the positive/negative logic select register. The general input register indicates the value accordingly. Note that the general input register does not hold the input signal using the FF.

10.3.2.3 Interrupt Input Mode

When a port is set for interrupt input mode using the corresponding bit in the general IO/interrupt switching register, the corresponding port serves as an interrupt input pin. In interrupt input mode, when the port receives an external interrupt, the corresponding bit in the interrupt display register indicates the input of an interrupt signal on the corresponding port pin, and an interrupt signal is output to the interrupt control block. In this mode, the polarity and detection conditions (edge or level) of the external input signal can be set for each port. The corresponding bits in the positive/negative logic select register and edge/level select register, one edge/both edge select register should be used to set the polarity and detection conditions, respectively.

If a port is set for edge detection using the corresponding bit in the edge/level select register, even when an external pulse interrupt signal is input, the corresponding bit in the interrupt display register holds the input using the FF and allows the level interrupt signal to be output to the interrupt control block. To stop all the interrupt signal outputs, all the bits in the interrupt clear register corresponding to the bits in the interrupt display register currently indicating the reception of the corresponding interrupt signals should be cleared to 0. Note that if a port is set for level detection using the corresponding bit in the edge/level select register and an external level interrupt signal is input, the corresponding bit in the interrupt display register does not use the FF to hold the input.

Interrupts indicated by the interrupt display register can be separately masked using the corresponding bits in the interrupt mask register and the interrupt sub mask register. When all the bits currently indicating the reception of the interrupt signals are masked, no interrupt signals are output to the interrupt control block. Masks can be canceled by writing 1 to the corresponding bits in the interrupt mask clear register or the interrupt sub mask clear register depending on the interrupt mask register or the interrupt sub mask register is used.

10.3.3 Handling of Input Signals on Port Pins

10.3.3.1 Chattering

In general input mode and interrupt input modes, a filtering function can be used for the port pins 0 to 3 of each GPIO block to prevent external chattering input. Specifically, when a bit in the chattering prevention on/off register is set to use the function, the external input to the corresponding port pin is sampled four consecutive times based on the filter clock signal, which is internally generated by the GPIO. The external input is canceled except when the active input is detected four consecutive times. Therefore, when a filtering function is used, input to the port pins 0 to 3 of each GPIO block need to be at least four sampling clock cycles long (The sampling clock is generated from peripheral clock/k, where k is determined by FILONOFFn.CLKSEL.).

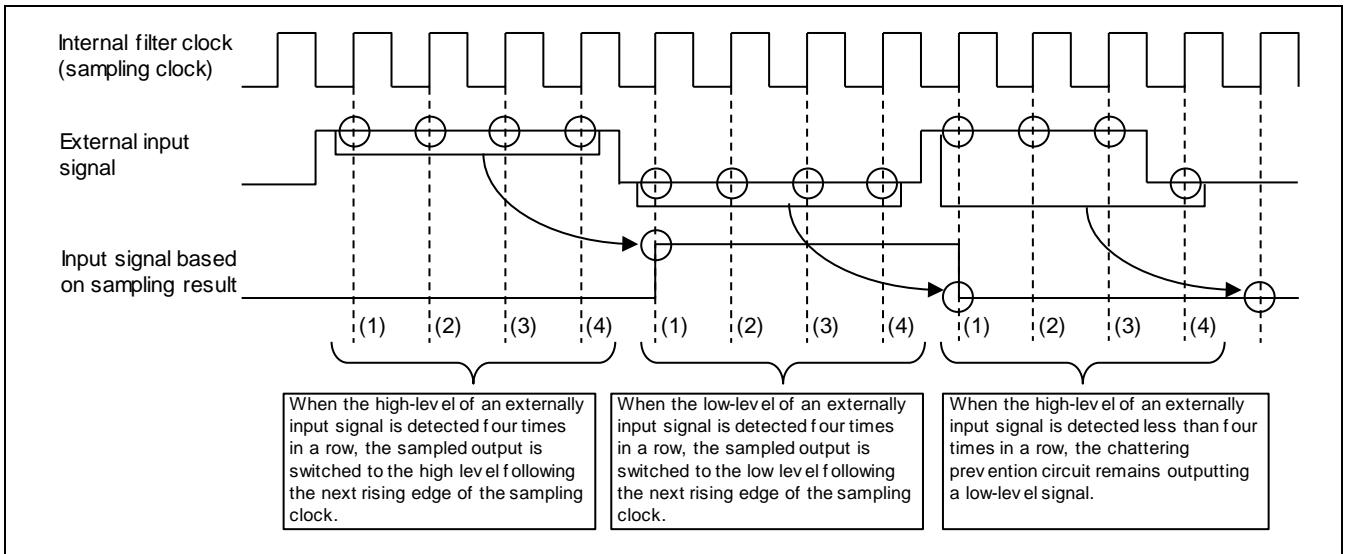


Figure 10.3 Sampling Timing Chart

10.3.3.2 Input Signal Synchronization

In general input mode and interrupt input mode, external input signals on all port pins are synchronized with the GPIO clock (S3D4φ).

10.3.4 Interrupt Display Timing Charts

Figure 10.4 shows the interrupt display timing and Figure 10.5 shows the note on the timing. In both figures, the positive logic and edge-sensitive input are assumed.

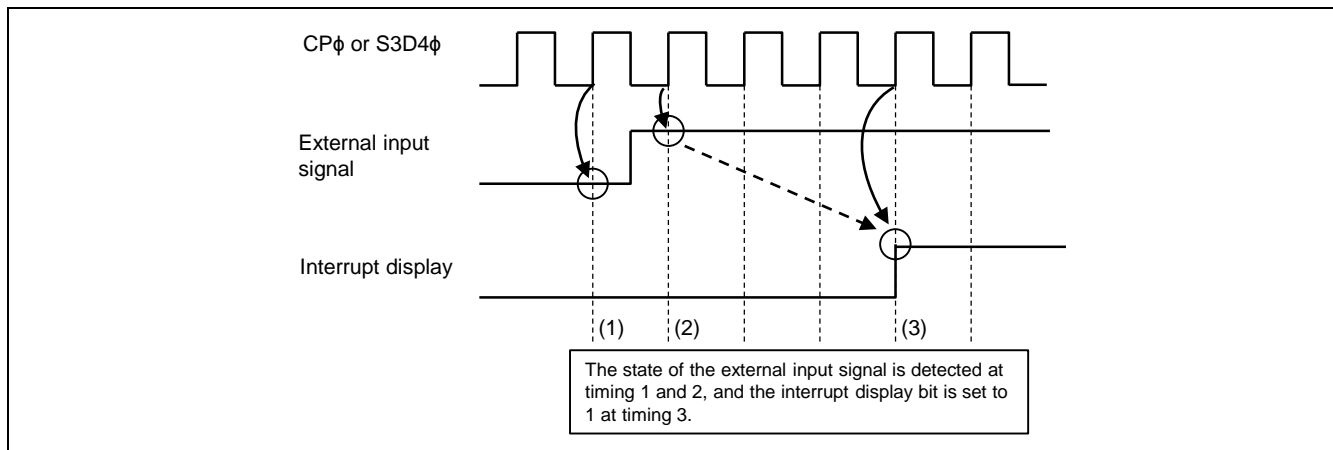


Figure 10.4 Interrupt Display Timing

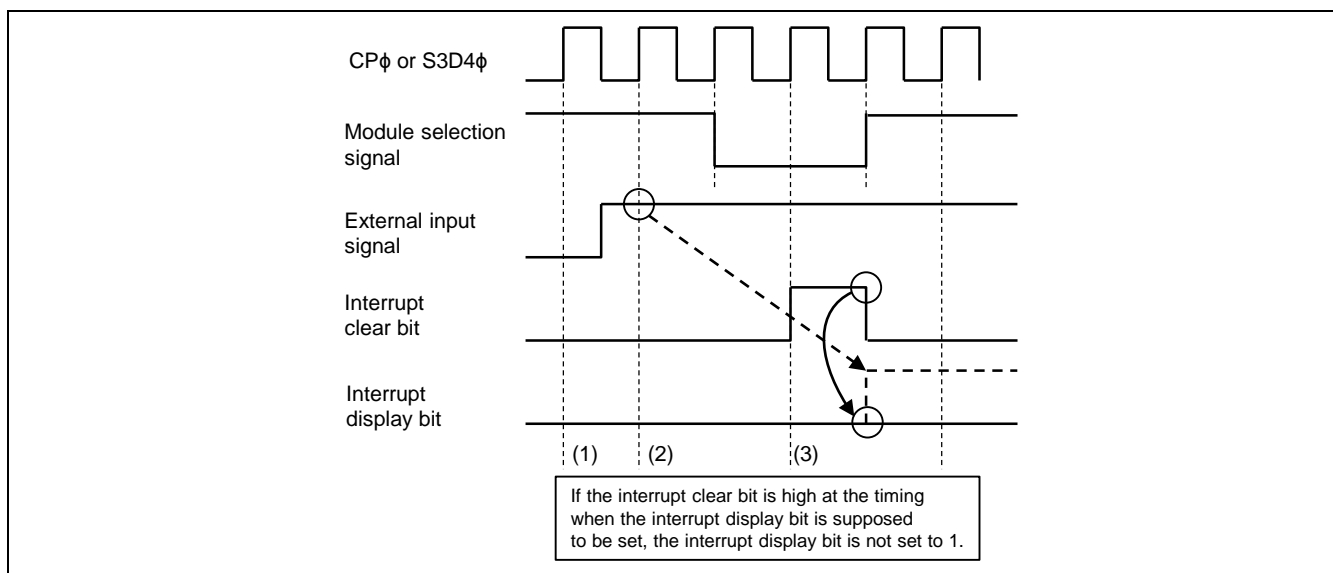


Figure 10.5 Note on Interrupt Display Timing

10.3.5 Using GPIO

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The following sections describe how to use the GPIO. If the GPIO is not used according to the procedures shown here, operations are not guaranteed.

10.3.5.1 Setting Edge-Sensitive Interrupt Input Mode

For setting edge-sensitive interrupt input mode, refer to the procedure shown in Figure 10.6.

Note that an unexpected interrupt might be generated in the module if setting (1), (2), (3) or (4) in the flowchart is changed. When changing the setting, (5) and (6) should be done.

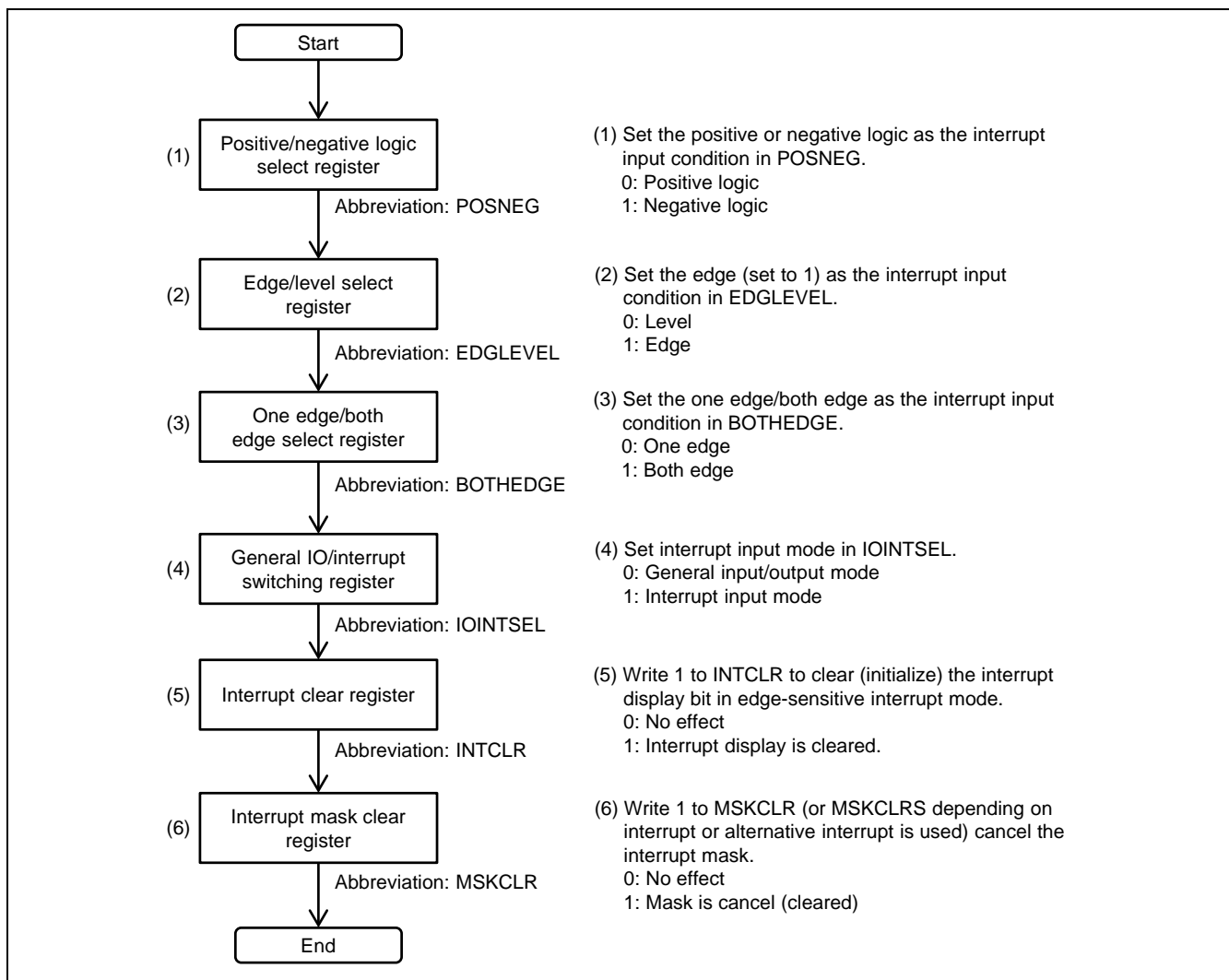


Figure 10.6 Flowchart of Setting the GPIO to Edge-Sensitive Interrupt Input Mode

10.3.5.2 Setting Level-Sensitive Interrupt Input Mode

For setting level-sensitive interrupt input mode, refer to the procedure shown in Figure 10.7.

Note that when an external level-sensitive interrupt input signal is stopped, the corresponding interrupt is canceled automatically. In level-sensitive interrupt input mode, the interrupt clear register is invalid.

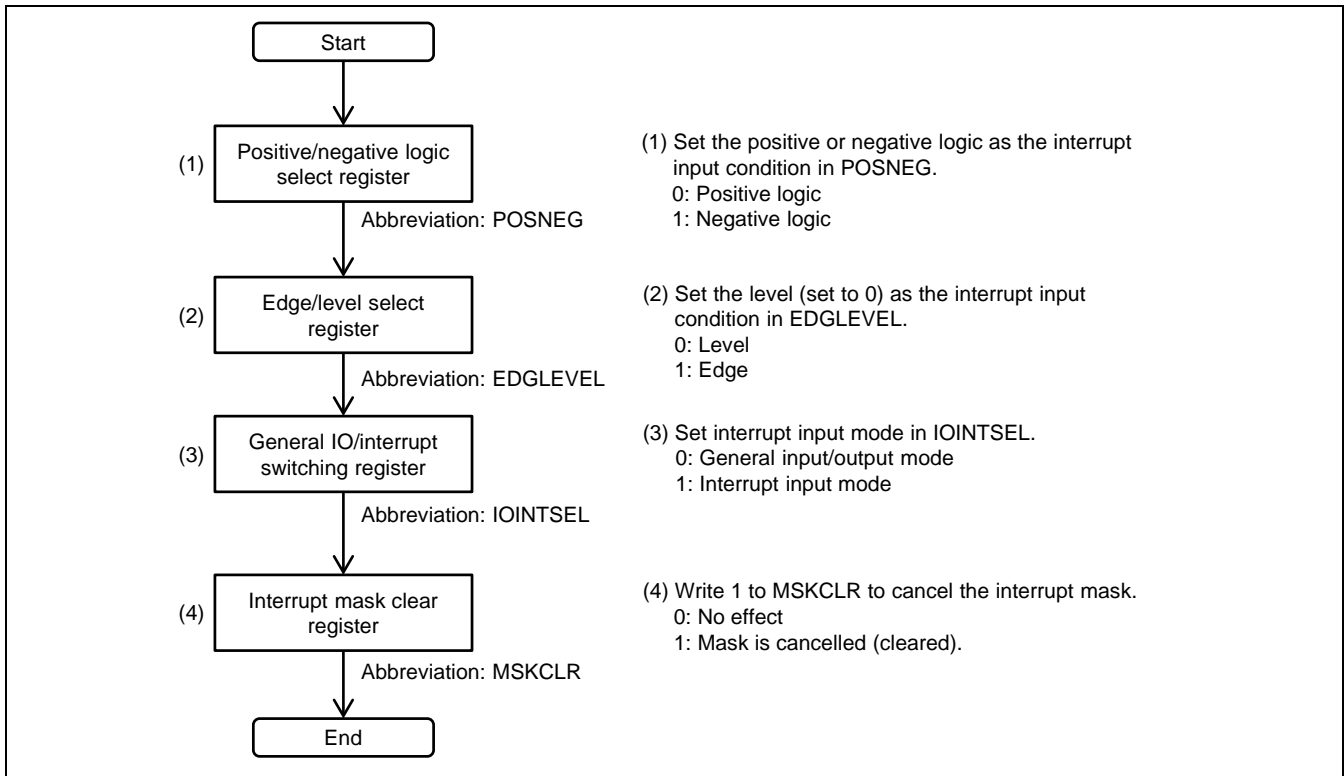


Figure 10.7 Flowchart of Setting the GPIO to Level-Sensitive Interrupt Input Mode

10.3.5.3 Setting General Output Mode

For setting general output mode, refer to the procedure shown in Figure 10.8.

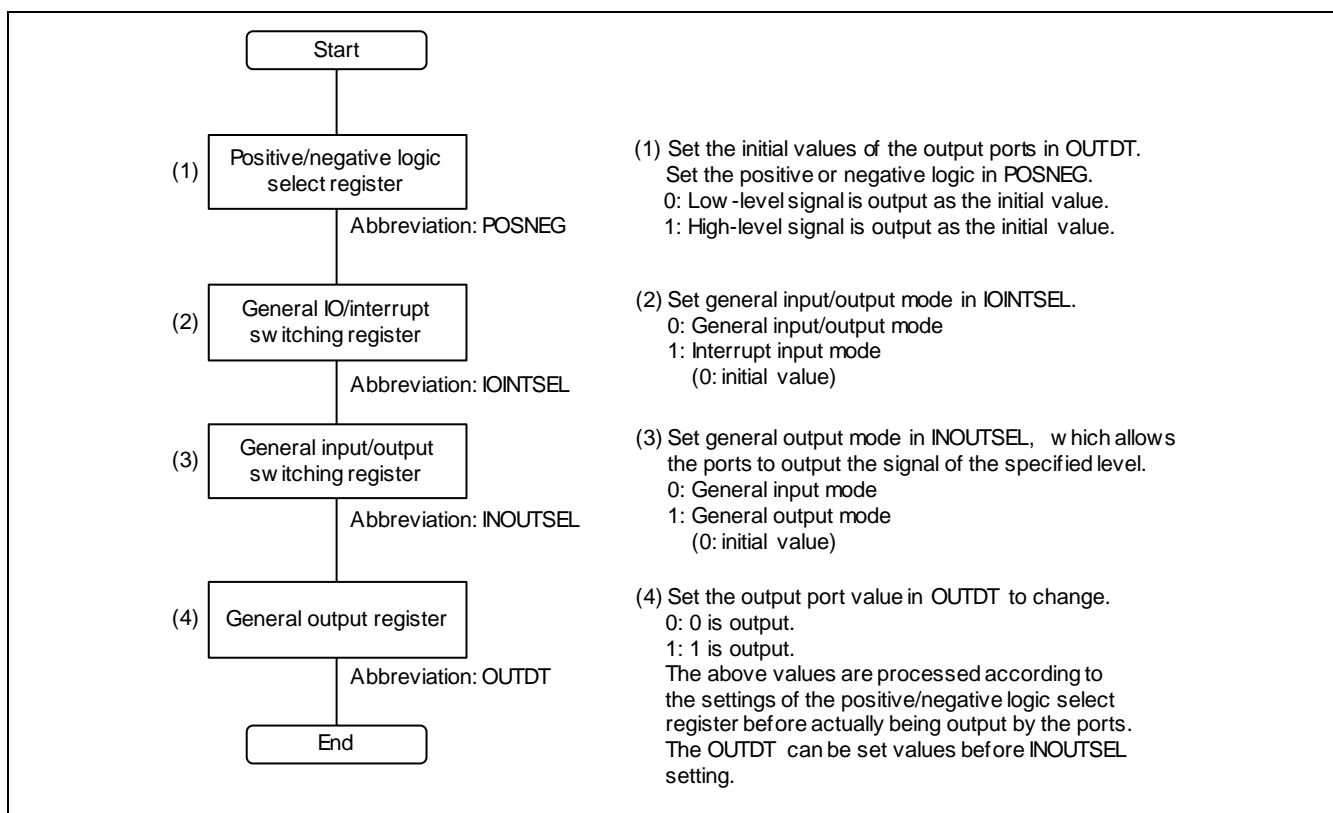


Figure 10.8 Flowchart of Setting the GPIO to General Output Mode

10.3.5.4 Setting Output data high/Output data low Mode

For setting output data high/output data low mode, refer to the procedure shown in Figure 10.9.

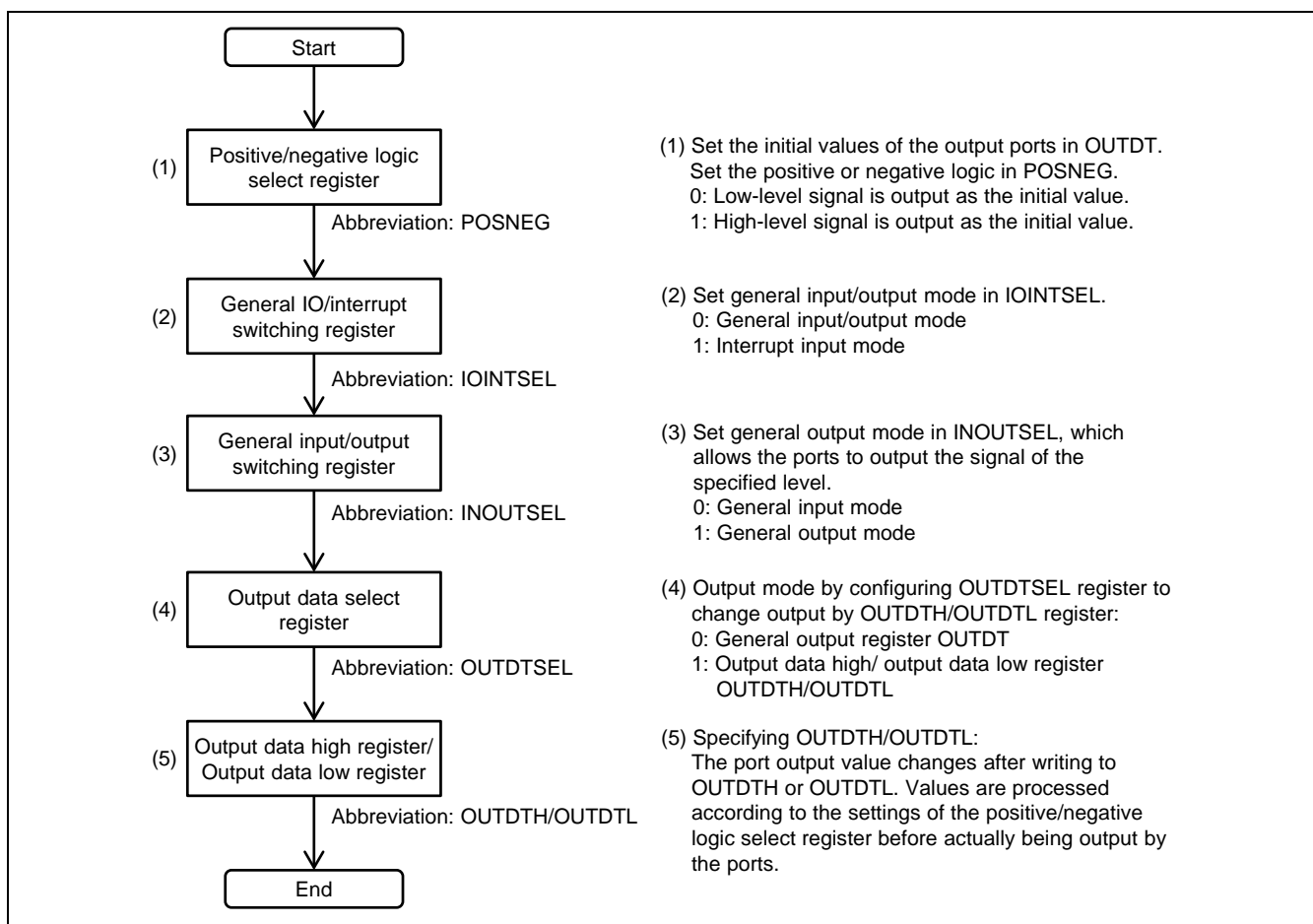


Figure 10.9 Flowchart of Setting the GPIO to Output data high/ Output data low Mode

10.3.5.5 Setting General Input Mode

For setting general input mode, refer to the procedure shown in Figure 10.10.

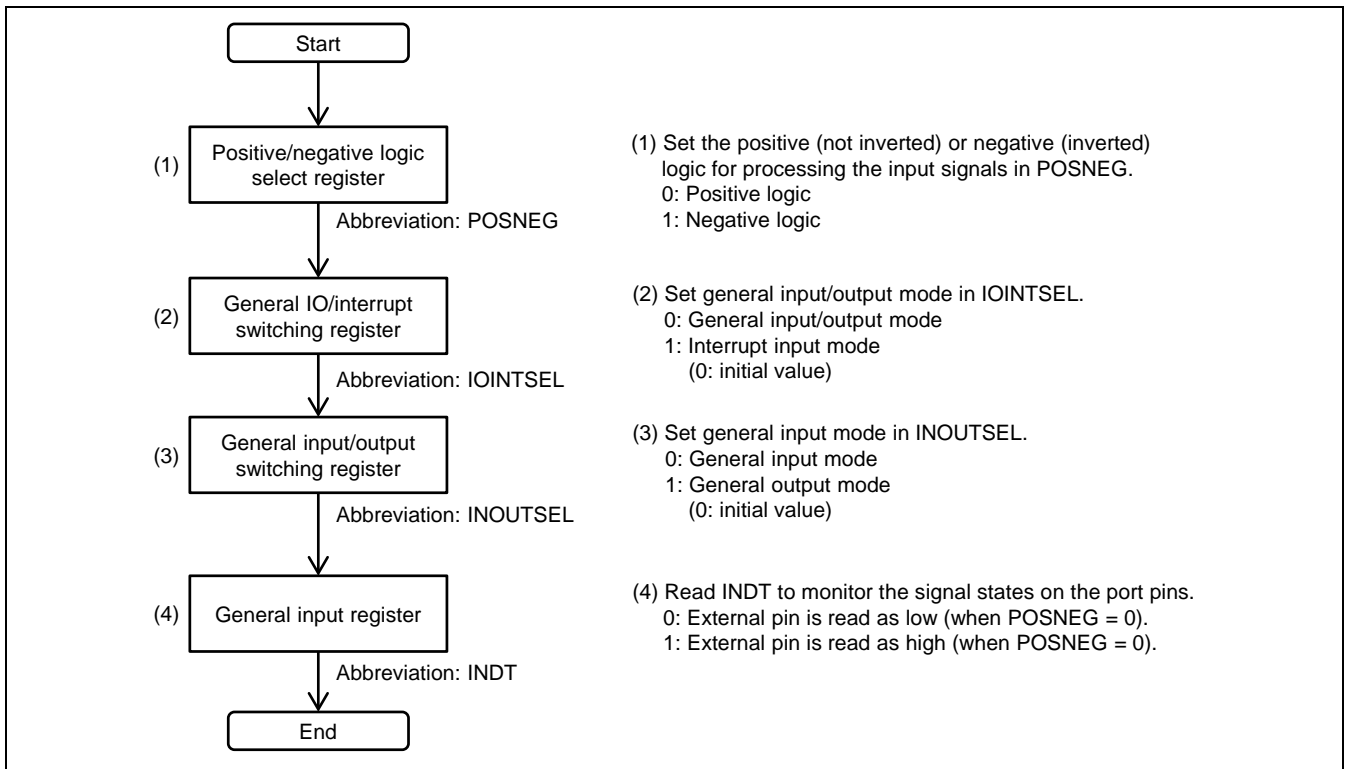


Figure 10.10 Flowchart of Setting the GPIO to General Input Mode

11. Clock Pulse Generator (CPG)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

11.1 Overview

This LSI has a clock pulse generator (CPG). The CPG consists of oscillators, PLL circuit n (n = 0 to 4), clock dividers, and the control circuit. The CPG generates various clocks used by this LSI.

11.1.1 Features

- Generates various clocks for LSI internal operation.
 - 5 PLLs for common/application part modules [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]
 - 4 PLLs for common/application part modules [RZ/G2N]
 - 2 PLLs and 1 SSCG PLL for common/application part modules [RZ/G2E]
 - SYS-CPU divider, common divider and DDR divider for system basic operation.
 - Dedicated dividers for special clock
- Controls clock supply to modules according to the module status (power supply status, etc.)
- There are registers related to secure and safety.

11.1.2 Block Diagram

A block diagram of the CPG is shown in Figure 11.1 to 11.4.

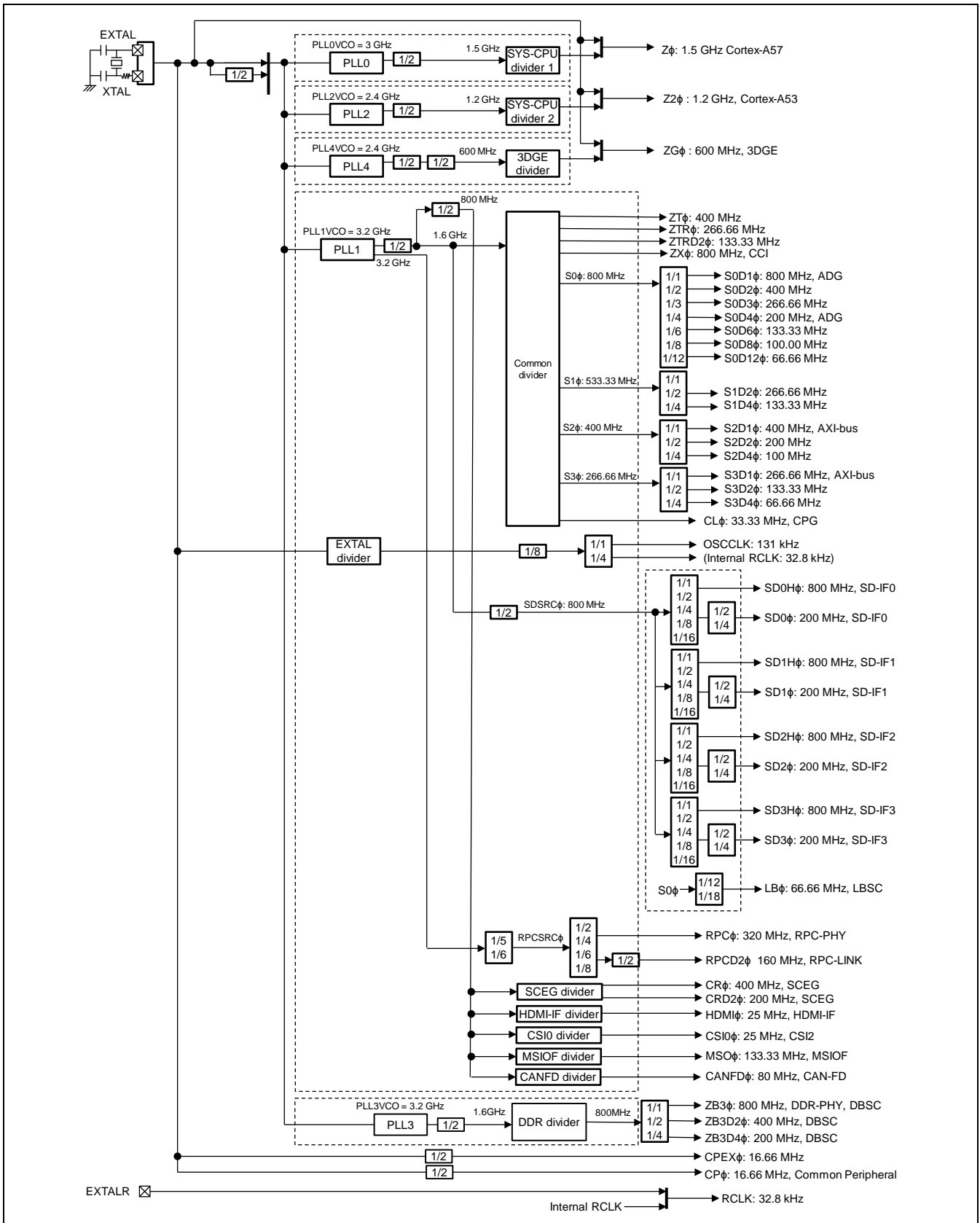


Figure 11.1 Block Diagram of CPG (RZ/G2H)

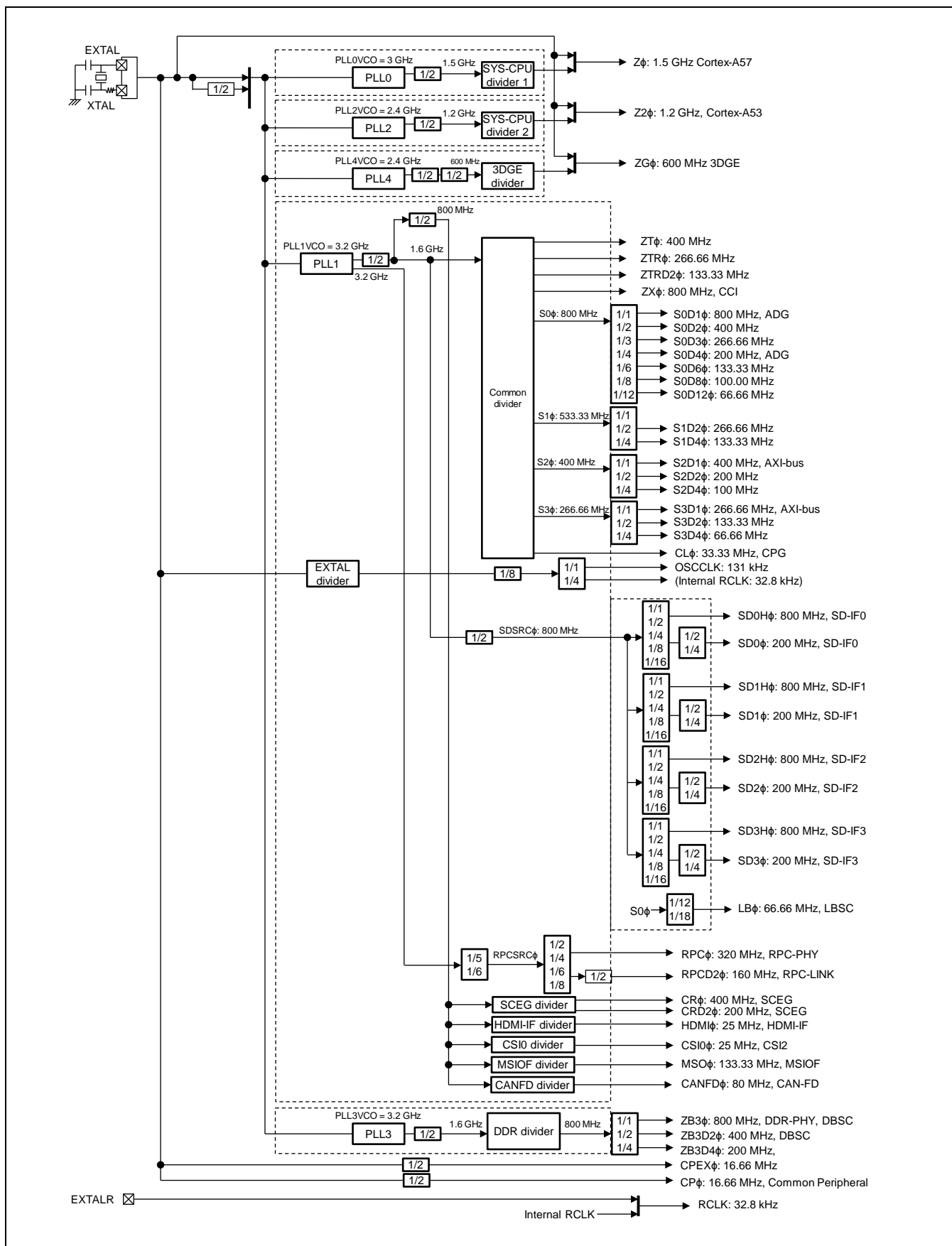


Figure 11.2 Block Diagram of CPG (RZ/G2M V1.3, RZ/G2M V3.0)

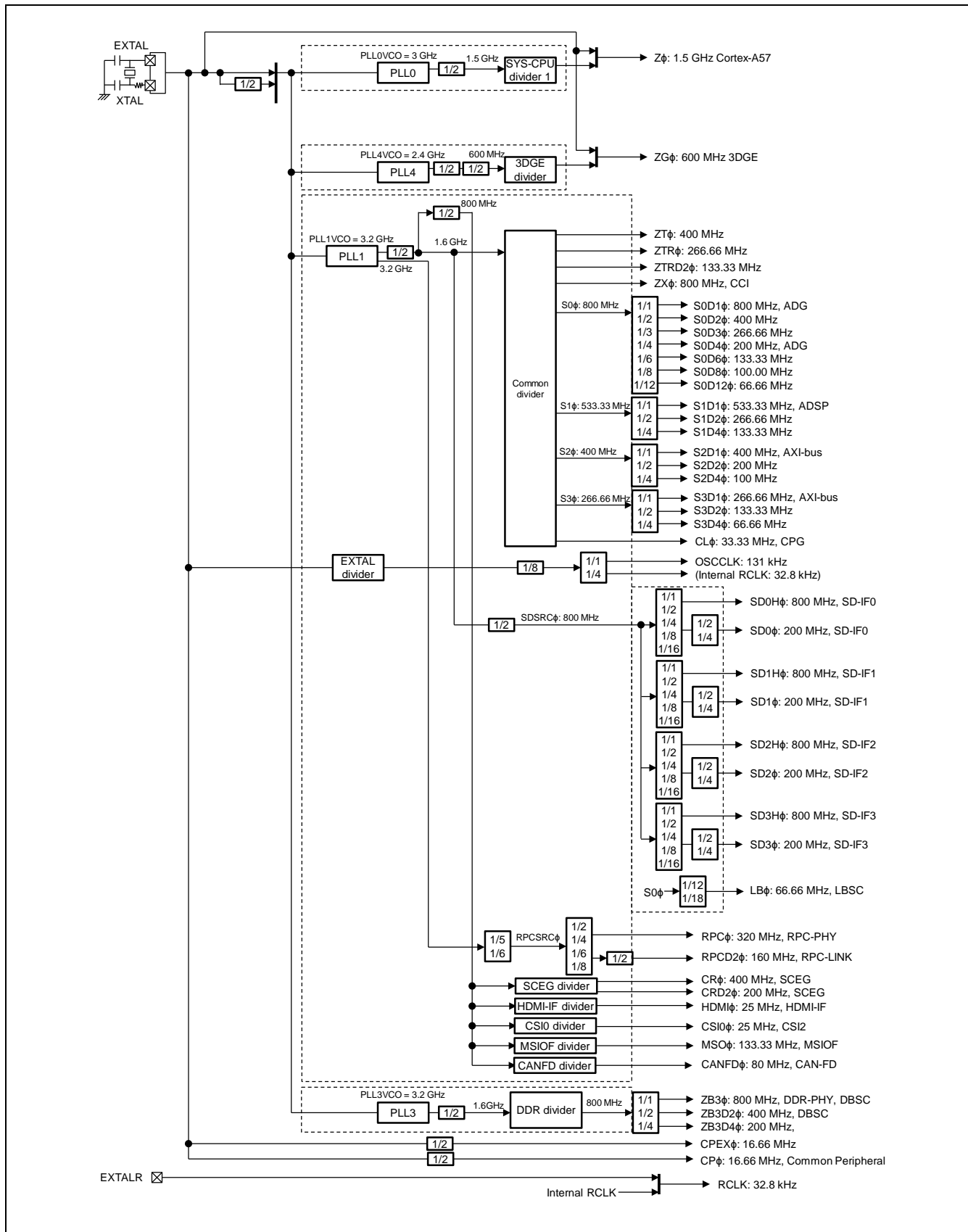


Figure 11.3 Block Diagram of CPG (RZ/G2N)

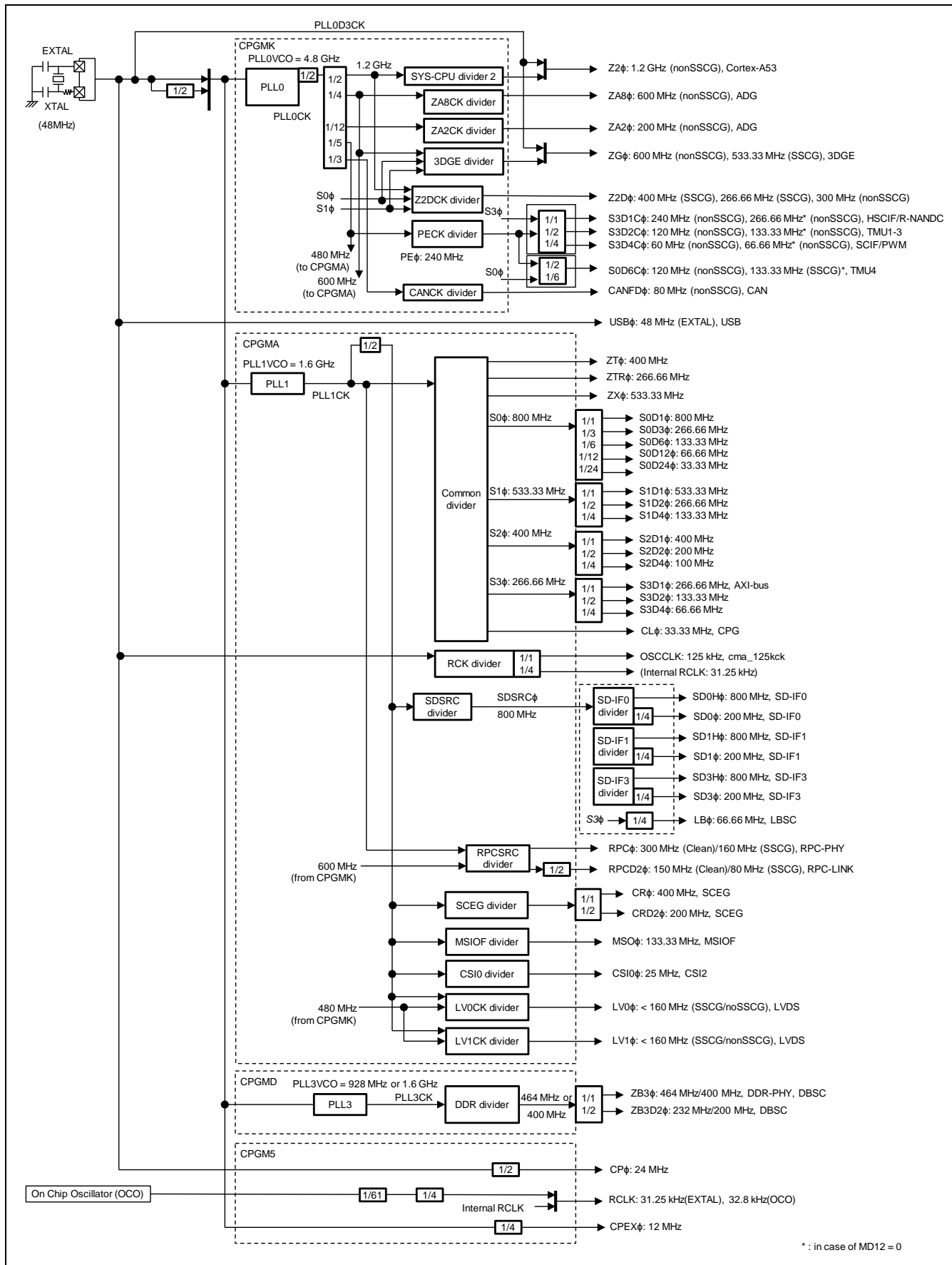


Figure 11.4 Block Diagram of CPG (RZ/G2E)

Followings are the functions of each block of CPG.

(1) PLL circuit 0 (PLL0)

PLL circuit 0 multiplies EXTAL or EXTAL/2 clock. The multiplication ratio is set by the PLL0CR.

(2) PLL circuit 1 (PLL1)

PLL circuit 1 multiplies the input clock from EXTAL (EXTAL or EXTAL/2) by the multiplication ratio shown in Table 11.2 to 11.5.

[RZ/G2E] SSCG (Spread Spectrum Clock Generator) support.

(3) PLL Circuit 2 (PLL2) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]

PLL circuit 2 multiplies the input clock from EXTAL (EXTAL or EXTAL/2). The multiplication ratio is set by the PLL2CR.

(4) PLL Circuit 3 (PLL3)

PLL circuit 3 multiplies the input clock from EXTAL (EXTAL or EXTAL/2) by the multiplication ratio shown in Table 11.2 to 11.5.

(5) PLL Circuit 4 (PLL4) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

PLL circuit 4 multiplies the input clock from EXTAL (EXTAL or EXTAL/2) by the multiplication ratio shown in Table 11.2 to 11.5.

(6) SYS-CPU clock divider 1

The SYS-CPU clock divider 1 divides PLL0 output clock. This divider generates the AP-System core clocks ($Z\phi$). The division ratio is set by the frequency control register C (FRQCRC).

(7) SYS-CPU clock divider 2

The SYS-CPU clock divider 2 divides PLLn output clock. This divider generates the AP-System core clocks ($Z2\phi$).

$n = 2$ [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0], $n = 0$ [RZ/G2E].

(8) 3DGE clock divider

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

The 3DGE clock divider divides PLL4 output clock by 2. This divider generates the system 3DGE clock ($ZG\phi$).

[RZ/G2E]

The 3DGE clock divider divides PLL0 output clock by 2. This divider generates the system 3DGE clock ($ZG\phi$). The division ratio is set by the frequency control register B (FRQCRB).

(9) Common clock divider

The common clock divider divides PLL1 output clock. This divider generates various system clocks. The division ratio is set by the frequency control register B (FRQCRB).

(10) DDR clock divider

The DDR clock divider divides PLL3 output clock, and generates DDR-PHY clock and DBSC clock.

(11) Dedicated dividers

The each dedicated divider generates special clock for the related modules.

(12) On Chip Oscillator (OCO) [RZ/G2E]On Chip Oscillator (OCO) [RZ/G2E]

By controlling RCKCR register, the clock generated from On Chip Oscillator (OCO) is output as RCLK.

11.1.3 External Pins

Table 11.1 lists the CPG pin configuration.

Table 11.1 Pin Configuration and Functions of CPG

| Name | I/O | Function | Second Generation RZ/G Series Products | | | | |
|--------|--------|---|---|----------------------------|--------|--------|--|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| XTAL | Output | Clock input pin Outputs amplified negative feedback of EXTAL | √ | √ | √ | √ | |
| EXTAL | Input | Clock input pin Used as an external clock input pin | √ | √ | √ | √ | |
| EXTALR | Input | Clock input pin Used as an external clock input pin The clock source of RCLK [RZ/G2M V1.3] Regardless of whether it is used as RCLK, it is necessary to input the clock of the frequency of 10kHz to less than EXTAL. | √ | √ | √ | — | |
| CLKOUT | Output | External Bus Clock output pin Used as an external clock output pin | √ | √ | √ | √ | |
| MD0 | Input | Mode 0 Fix to L | √ | √ | √ | √ | |
| MD1 | Input | Mode 1 Boot device select | √ | √ | √ | √ | |
| MD2 | Input | Mode 2 Boot device select | √ | √ | √ | √ | |
| MD3 | Input | Mode 3 Boot device select | √ | √ | √ | √ | |
| MD4 | Input | Mode 4 Boot device select | √ | √ | √ | √ | |
| MD6 | Input | Mode 6 Master boot processor select | √ | √ | √ | √ | |
| MD7 | Input | Mode 7 Master boot processor select | √ | √ | √ | √ | |
| MD9 | Input | Mode 9 EXTAL/XTAL pin setting | √ | √ | √ | √ | |

| Name | I/O | Function | Second Generation RZ/G Series Products | | | |
|------|-------|--|---|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| MD12 | Input | Mode 12 [RZ/G2E] PLL1 SSCG/nonSSCG select [RZ/G2M V1.3, RZ/G2M V3.0] Fix to L. | — | √ | — | √ |
| MD13 | Input | Mode 13 PLL multiplication ratio setting | √ | √ | √ | — |
| MD14 | Input | Mode 14 PLL multiplication ratio setting | √ | √ | √ | — |
| MD19 | Input | Mode 19 DDR clock frequency setting | √ | √ | √ | √ |
| MD17 | Input | Mode 17 DDR clock frequency setting | √ | √ | √ | √ |
| MD18 | Input | Mode 18 External bus clock frequency setting | √ | √ | √ | — |
| MD28 | Input | Mode 28 Select initial RCLK source | √ | √ | √ | — |

11.1.4 List of Clock Outputs

Table 11.2 to 11.5 lists the clock output of CPG.

Table 11.2 List of Clocks [RZ/G2H]

| Name | Clock Source | Frequency | | | Clock Domain, Function |
|--------------|-------------------|----------------|------------|--------------------------|--|
| | | Dividing ratio | Maximum | Initial | |
| Z ϕ | SYS-CPU divider 1 | Variable | 1.5 GHz | 1.5 GHz | System CPU (Cortex-A57) clock |
| Z2 ϕ | SYS-CPU divider 2 | Variable | 1200 MHz | 1200 MHz | System CPU (Cortex-A53) clock |
| ZG ϕ | 3DGE divider | Variable | 600 MHz | 600 MHz | 3DGE clock |
| ZTR ϕ | Common divider | Variable | 266.66 MHz | 266.66 MHz ^{*5} | Trace interface clock |
| ZTRD2 ϕ | Common divider | Variable | 133.33 MHz | 133.33 MHz ^{*5} | System trace interface clock |
| ZT ϕ | Common divider | Variable | 400 MHz | 400 MHz ^{*5} | Internal trace clock |
| ZX ϕ | Common divider | Fixed | 800 MHz | 800 MHz | Cache coherent interconnect clock |
| S0 ϕ | Common divider | Fixed | 800 MHz | 800 MHz | Clock source of S0D1 ϕ , S0D2 ϕ , S0D3 ϕ , S0D4 ϕ , S0D6 ϕ , S0D8 ϕ and S0D12 ϕ |
| S1 ϕ | Common divider | Fixed | 533.33 MHz | 533.33 MHz | Clock source of S1D2 ϕ , and S1D4 ϕ |
| S2 ϕ | Common divider | Fixed | 400 MHz | 400 MHz | Clock source of S2D1 ϕ , S2D2 ϕ , and S2D4 ϕ |
| S3 ϕ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | Clock source of S3D1 ϕ , S3D2 ϕ , and S3D4 ϕ |
| S0D1 ϕ | Common divider | Fixed | 800 MHz | 800 MHz | ADG |
| S0D2 ϕ | Common divider | Fixed | 400 MHz | 400 MHz | |
| S0D3 ϕ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | |
| S0D4 ϕ | Common divider | Fixed | 200 MHz | 200 MHz | ADG |
| S0D6 ϕ | Common divider | Fixed | 133.33 MHz | 133.33 MHz | |
| S0D8 ϕ | Common divider | Fixed | 100 MHz | 100 MHz | |
| S0D12 ϕ | Common divider | Fixed | 66.66 MHz | 66.66 MHz | |
| S1D2 ϕ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | |
| S1D4 ϕ | Common divider | Fixed | 133.33 MHz | 133.33 MHz | |
| S2D1 ϕ | Common divider | Fixed | 400.00 MHz | 400.00 MHz | AXI-bus |
| S2D2 ϕ | Common divider | Fixed | 200.00 MHz | 200.00 MHz | |
| S2D4 ϕ | Common divider | Fixed | 100.00 MHz | 100.00 MHz | |
| S3D1 ϕ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | AXI-bus |
| S3D2 ϕ | Common divider | Fixed | 133.33 MHz | 133.33 MHz | |
| S3D4 ϕ | Common divider | Fixed | 66.66 MHz | 66.66 MHz | |
| LB ϕ | Common divider | Fixed | 66.66 MHz | ^{*1} | LBSC CLKOUT |
| CL ϕ | Common divider | Fixed | 33.33 MHz | 33.33 MHz | CPG |
| ZB3 ϕ | DDR divider | Variable | 800 MHz | 800 MHz ^{*4} | DBSC clock |
| ZB3D2 ϕ | DDR divider | Variable | 400 MHz | 400 MHz ^{*4} | DBSC clock |
| ZB3D4 ϕ | DDR divider | Variable | 200 MHz | 200 MHz ^{*4} | DBSC clock |

| Name | Clock Source | Frequency | | | Clock Domain, Function |
|--------------|----------------------------|----------------|------------|----------|---|
| | | Dividing ratio | Maximum | Initial | |
| CR ϕ | SCEG divider | Fixed | 400 MHz | 400 MHz | SCEG clock |
| CRD2 ϕ | SCEG divider | Fixed | 200 MHz | 200 MHz | SCEG clock |
| SDSRC ϕ | SDSRC divider | Fixed | 800 MHz | 800 MHz | Source clock of SDnH ϕ and SDn ϕ (n = 0 to 3) |
| SD0H ϕ | SD-IF0 divider | Variable | 800 MHz | Stopped | SD-IF0 clock |
| SD0 ϕ | SD-IF0 divider | Variable | 200 MHz | 50 MHz | SD-IF0 clock |
| SD1H ϕ | SD-IF1 divider | Variable | 800 MHz | Stopped | SD-IF1 clock |
| SD1 ϕ | SD-IF1 divider | Variable | 200 MHz | 50 MHz | SD-IF1 clock |
| SD2H ϕ | SD-IF2 divider | Variable | 800 MHz | Stopped | SD-IF2 clock |
| SD2 ϕ | SD-IF2 divider | Variable | 200 MHz | 50 MHz | SD-IF2 clock |
| SD3H ϕ | SD-IF3 divider | Variable | 800 MHz | Stopped | SD-IF3 clock |
| SD3 ϕ | SD-IF3 divider | Variable | 200 MHz | 50 MHz | SD-IF3 clock |
| RPC ϕ | RPC divider | Variable | 320 MHz | *2 | RPC-IF clock |
| RPCD2 ϕ | RPC divider | Variable | 160 MHz | *2 | RPC-IF clock |
| MSO ϕ | MSO divider | Variable | 133.33 MHz | 12.5 MHz | MSIOF clock |
| CANFD ϕ | CANFD divider | Variable | 80 MHz | 12.5 MHz | CANFD clock |
| HDMI ϕ | HDMI-IF divider | Variable | 25 MHz | Stopped | HDMI-IF clock |
| CSI0 ϕ | CSI0 divider | Variable | 25 MHz | Stopped | CSI2 clock |
| CP ϕ | EXTAL | Fixed | 16.66 MHz | *3 | Common peripheral clock, EXTAL \times 1/2 |
| CPEX ϕ | EXTAL | Fixed | 16.66 MHz | *3 | EXTAL \times 1/2 |
| RCLK | Common divider / EXTALR | Fixed | 32.89 kHz | *6 | |
| OSCCLK | Common divider | Fixed | 131.57 kHz | *7 | |

- Notes:
- The frequency of LB ϕ depends on the value of MD18.
 - The frequency of RPC ϕ depends on the value of MD4, MD3, MD2 and MD1.
When MD4 = H or L, MD3 = L, MD2 = H, MD1 = L, RPC ϕ = 320 MHz, RPCD2 ϕ = 160MHz.
When MD4 = H or L, MD3 = L, MD2 = H, MD1 = H, RPC ϕ = 160 MHz, RPCD2 ϕ = 80MHz.
In the other cases of MD4, MD3, MD2 and MD1 setting, RPC ϕ = 80 MHz, RPCD2 ϕ = 40MHz.
 - The frequency of CP ϕ is equal to EXTAL \times 1/2. For example, when the frequency of EXTAL is equal to 20 MHz, the frequency of CPCLK is equal to 10 MHz.
 - The initial frequency of ZB3 ϕ and ZB3D2 ϕ depends on the setting of MD19 and MD17. The values shown in the above table corresponds to the case of MD19 = L and MD17 = L.
 - These clocks are oscillated in the debug mode. Otherwise, they are stopped.
 - The initial value depends on MD14, MD13 and MD28.
When MD28 = H, RCLK = EXTALR.
When MD28 = L, MD14 = L and MD13 = H, RCLK = 32.89 (kHz)
In the other setting, RCLK = 32.55 (kHz)
 - The initial value depends on MD14, MD13.
When MD14 = L and MD13 = H, OSCCLK = 131.57 (kHz).
In the other setting, OSCCLK = 130.20 (kHz)
 - Divider setting for listed clock in Table 11.2 must not be over than maximum clock frequency of each clocks.

| MD14 | MD13 | EXTAL (MHz) | EXTAL divider | | Internal RCLK | | OSCCLK | |
|------|------|----------------|----------------------------|--------------------|----------------------------|--------------------|----------------------------|--------------------|
| | | | Fixed division ratio | Frequency (MHz) | Fixed division ratio | Frequency (kHz) | Fixed division ratio | Frequency (kHz) |
| 0 | 0 | 16.66 | × 1/16 | 1.04 | × 1/32 | 32.55 | × 1/8 | 130.20 |
| 0 | 1 | 20.00 | × 1/19 | 1.05 | × 1/32 | 32.89 | × 1/8 | 131.57 |
| 1 | 0 | 25.00 | × 1/24 | 1.04 | × 1/32 | 32.55 | × 1/8 | 130.20 |
| 1 | 1 | 33.33 | × 1/32 | 1.04 | × 1/32 | 32.55 | × 1/8 | 130.20 |

Table 11.3 List of Clocks [RZ/G2M V1.3, RZ/G2M V3.0]

| Name | Clock Source | Frequency | | | Clock Domain, Function |
|--------------|-------------------|----------------|------------|------------------|--|
| | | Dividing ratio | Maximum | Initial | |
| Z ϕ | SYS-CPU divider 1 | Variable | 1.5 GHz | 1.5 GHz | System CPU (Cortex-A57) clock |
| Z2 ϕ | SYS-CPU divider 2 | Variable | 1200 MHz | 1200 MHz | System CPU (Cortex-A53) clock |
| ZG ϕ | 3DGE divider | Variable | 600 MHz | 600 MHz | 3DGE clock |
| ZTR ϕ | Common divider | Variable | 266.66 MHz | 266.66 MHz *5 | Trace interface clock |
| ZTRD2 ϕ | Common divider | Variable | 133.33 MHz | 133.33 MHz *5 | System trace interface clock |
| ZT ϕ | Common divider | Variable | 400 MHz | 400 MHz *5 | Internal trace clock |
| ZX ϕ | Common divider | Fixed | 800 MHz | 800 MHz | Cache coherent interconnect clock |
| S0 ϕ | Common divider | Fixed | 800 MHz | 800 MHz | Clock source of S0D1 ϕ , S0D2 ϕ , S0D3 ϕ , S0D4 ϕ , S0D6 ϕ , S0D8 ϕ , and S0D12 ϕ |
| S1 ϕ | Common divider | Fixed | 533.33 MHz | 533.33 MHz | Clock source of S1D2 ϕ , and S1D4 ϕ |
| S2 ϕ | Common divider | Fixed | 400 MHz | 400 MHz | Clock source of S2D1 ϕ , S2D2 ϕ , and S2D4 ϕ |
| S3 ϕ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | Clock source of S3D1 ϕ , S3D2 ϕ , and S3D4 ϕ |
| S0D1 ϕ | Common divider | Fixed | 800 MHz | 800 MHz | ADG |
| S0D2 ϕ | Common divider | Fixed | 400 MHz | 400 MHz | |
| S0D3 ϕ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | |
| S0D4 ϕ | Common divider | Fixed | 200 MHz | 200 MHz | ADG |
| S0D6 ϕ | Common divider | Fixed | 133.33 MHz | 133.33 MHz | |
| S0D8 ϕ | Common divider | Fixed | 100 MHz | 100 MHz | |
| S0D12 ϕ | Common divider | Fixed | 66.66 MHz | 66.66 MHz | |
| S1D2 ϕ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | |
| S1D4 ϕ | Common divider | Fixed | 133.33 MHz | 133.33 MHz | |
| S2D1 ϕ | Common divider | Fixed | 400.00 MHz | 400.00 MHz | AXI-bus |
| S2D2 ϕ | Common divider | Fixed | 200.00 MHz | 200.00 MHz | |
| S2D4 ϕ | Common divider | Fixed | 100.00 MHz | 100.00 MHz | |
| S3D1 ϕ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | AXI-bus |
| S3D2 ϕ | Common divider | Fixed | 133.33 MHz | 133.33 MHz | |
| S3D4 ϕ | Common divider | Fixed | 66.66 MHz | 66.66 MHz | |
| LB ϕ | Common divider | Fixed | 66.66 MHz | *1 | LBSC CLKOUT |
| CL ϕ | Common divider | Fixed | 33.33 MHz | 33.33 MHz | CPG |
| ZB3 ϕ | DDR divider | Variable | 800 MHz | 800 MHz *4 | DBSC clock |
| ZB3D2 ϕ | DDR divider | Variable | 400 MHz | 400 MHz *4 | DBSC clock |
| ZB3D4 ϕ | DDR divider | Variable | 200 MHz | 200 MHz *4 | DBSC clock |
| CR ϕ | SCEG divider | Fixed | 400 MHz | 400 MHz | SCEG clock |
| CRD2 ϕ | SCEG divider | Fixed | 200 MHz | 200 MHz | SCEG clock |

| Name | Clock Source | Frequency | | | Clock Domain, Function |
|--------------|----------------------------|----------------|------------|----------|---|
| | | Dividing ratio | Maximum | Initial | |
| SDSRC ϕ | SDSRC divider | Fixed | 800 MHz | 800 MHz | Source clock of SDnH ϕ and SDn ϕ (n = 0 to 3) |
| SD0H ϕ | SD-IF0 divider | Variable | 800 MHz | Stopped | SD-IF0 clock |
| SD0 ϕ | SD-IF0 divider | Variable | 200 MHz | 50 MHz | SD-IF0 clock |
| SD1H ϕ | SD-IF1 divider | Variable | 800 MHz | Stopped | SD-IF1 clock |
| SD1 ϕ | SD-IF1 divider | Variable | 200 MHz | 50 MHz | SD-IF1 clock |
| SD2H ϕ | SD-IF2 divider | Variable | 800 MHz | Stopped | SD-IF2 clock |
| SD2 ϕ | SD-IF2 divider | Variable | 200 MHz | 50 MHz | SD-IF2 clock |
| SD3H ϕ | SD-IF3 divider | Variable | 800 MHz | Stopped | SD-IF3 clock |
| SD3 ϕ | SD-IF3 divider | Variable | 200 MHz | 50 MHz | SD-IF3 clock |
| RPC ϕ | RPC divider | Variable | 320 MHz | *2 | RPC-IF clock |
| RPCD2 ϕ | RPC divider | Variable | 160 MHz | *2 | RPC-IF clock |
| MSO ϕ | MSO divider | Variable | 133.33 MHz | 12.5 MHz | MSIOF clock |
| CANFD ϕ | CANFD divider | Variable | 80 MHz | 12.5 MHz | CANFD clock |
| HDMI ϕ | HDMI-IF divider | Variable | 25 MHz | Stopped | HDMI-IF clock |
| CSI0 ϕ | CSI0 divider | Variable | 25 MHz | Stopped | CSI2 clock |
| CP ϕ | EXTAL | Fixed | 16.66 MHz | *3 | Common peripheral clock, EXTAL \times 1/2 |
| CPEX ϕ | EXTAL | Fixed | 16.66 MHz | *3 | EXTAL \times 1/2 |
| RCLK | Common divider / EXTALR | Fixed | 32.89 kHz | *6 | |
| OSCCLK | Common divider | Fixed | 131.57 kHz | *7 | |

Notes: 1. The frequency of LB ϕ depends on the value of MD18.

2. The frequency of RPC ϕ depends on the value of MD4, MD3, MD2 and MD1.

When MD4 = L, MD3 = L, MD2 = L, MD1 = H, RPC ϕ = 80 MHz, RPCD2 ϕ = 40MHz.

When MD4 = H or L, MD3 = L, MD2 = H, MD1 = L, RPC ϕ = 320 MHz, RPCD2 ϕ = 160MHz.

When MD4 = H or L, MD3 = L, MD2 = H, MD1 = H, RPC ϕ = 160 MHz, RPCD2 ϕ = 80MHz.

When MD4 = L, MD3 = H, MD2 = L, MD1 = L, RPC ϕ = 80 MHz, RPCD2 ϕ = 40MHz.

In the other cases of MD4, MD3, MD2 and MD1 setting, RPC ϕ and RPCD2 ϕ are stopped.

3. The frequency of CP ϕ is equal to EXTAL \times 1/2. For example, when the frequency of EXTAL is equal to 20 MHz, the frequency of CPCLK is equal to 10 MHz.
4. The initial frequency of ZB3 ϕ and ZB3D2 ϕ depends on the setting of MD19 and MD17. The values shown in the above table corresponds to the case of MD19 = L and MD17 = L.
5. These clocks are oscillated in the debug mode. Otherwise, they are stopped.
6. The initial value depends on MD14, MD13 and MD28.
When MD28 = H, RCLK = EXTALR.
When MD28 = L, MD14 = L and MD13 = H, RCLK = 32.89 (kHz)
In the other setting, RCLK = 32.55 (kHz)
7. The initial value depends on MD14, MD13.
When MD14 = L and MD13 = H, OSCCLK = 131.57 (kHz).
In the other setting, OSCCLK = 130.20 (kHz)
8. Divider setting for listed clock in Table 11.3 must not be over than maximum clock frequency of each clocks.

| MD14 | MD13 | EXTAL (MHz) | EXTAL divider | | Internal RCLK | | OSCCLK | |
|------|------|----------------|----------------------------|--------------------|----------------------------|--------------------|----------------------------|--------------------|
| | | | Fixed division ratio | Frequency (MHz) | Fixed division ratio | Frequency (kHz) | Fixed division ratio | Frequency (kHz) |
| 0 | 0 | 16.66 | \times 1/16 | 1.04 | \times 1/32 | 32.55 | \times 1/8 | 130.20 |
| 0 | 1 | 20.00 | \times 1/19 | 1.05 | \times 1/32 | 32.89 | \times 1/8 | 131.57 |
| 1 | 0 | 25.00 | \times 1/24 | 1.04 | \times 1/32 | 32.55 | \times 1/8 | 130.20 |
| 1 | 1 | 33.33 | \times 1/32 | 1.04 | \times 1/32 | 32.55 | \times 1/8 | 130.20 |

Table 11.4 List of Clocks [RZ/G2N]

| Name | Clock Source | Frequency | | | Clock Domain, Function |
|--------------|-------------------|----------------|------------|---------------------------|--|
| | | Dividing ratio | Maximum | Initial | |
| Z ϕ | SYS-CPU divider 1 | Variable | 1.5 GHz | 1.5 GHz | System CPU (Cortex-A57) clock |
| ZG ϕ | 3DGE divider | Variable | 600 MHz | 600 MHz | 3DGE clock |
| ZTR ϕ | Common divider | Variable | 266.66 MHz | 266.66 MHz * ⁵ | Trace interface clock |
| ZTRD2 ϕ | Common divider | Variable | 133.33 MHz | 133.33 MHz * ⁵ | System trace interface clock |
| ZT ϕ | Common divider | Variable | 400 MHz | 400 MHz * ⁵ | Internal trace clock |
| ZX ϕ | Common divider | Fixed | 800 MHz | 800 MHz | Cache coherent interconnect clock |
| S0 ϕ | Common divider | Fixed | 800 MHz | 800 MHz | Clock source of S0D1 ϕ , S0D2 ϕ , S0D3 ϕ , S0D4 ϕ , S0D6 ϕ , S0D8 ϕ and S0D12 ϕ |
| S1 ϕ | Common divider | Fixed | 533.33 MHz | 533.33 MHz | Clock source of S1D2 ϕ , and S1D4 ϕ |
| S2 ϕ | Common divider | Fixed | 400 MHz | 400 MHz | Clock source of S2D1 ϕ , S2D2 ϕ , and S2D4 ϕ |
| S3 ϕ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | Clock source of S3D1 ϕ , S3D2 ϕ , and S3D4 ϕ |
| S0D1 ϕ | Common divider | Fixed | 800 MHz | 800 MHz | ADG |
| S0D2 ϕ | Common divider | Fixed | 400 MHz | 400 MHz | |
| S0D3 ϕ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | |
| S0D4 ϕ | Common divider | Fixed | 200 MHz | 200 MHz | ADG |
| S0D6 ϕ | Common divider | Fixed | 133.33 MHz | 133.33 MHz | |
| S0D8 ϕ | Common divider | Fixed | 100 MHz | 100 MHz | |
| S0D12 ϕ | Common divider | Fixed | 66.66 MHz | 66.66 MHz | |
| S1D2 ϕ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | |
| S1D4 ϕ | Common divider | Fixed | 133.33 MHz | 133.33 MHz | |
| S2D1 ϕ | Common divider | Fixed | 400.00 MHz | 400.00 MHz | AXI-bus |
| S2D2 ϕ | Common divider | Fixed | 200.00 MHz | 200.00 MHz | |
| S2D4 ϕ | Common divider | Fixed | 100.00 MHz | 100.00 MHz | |
| S3D1 ϕ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | AXI-bus |
| S3D2 ϕ | Common divider | Fixed | 133.33 MHz | 133.33 MHz | |
| S3D4 ϕ | Common divider | Fixed | 66.66 MHz | 66.66 MHz | |
| LB ϕ | Common divider | Fixed | 66.66 MHz | * ¹ | LBSC CLKOUT |
| CL ϕ | Common divider | Fixed | 33.33 MHz | 33.33 MHz | CPG |
| ZB3 ϕ | DDR divider | Variable | 800 MHz | 800 MHz * ⁴ | DBSC clock |
| ZB3D2 ϕ | DDR divider | Variable | 400 MHz | 400 MHz * ⁴ | DBSC clock |
| ZB3D4 ϕ | DDR divider | Variable | 200 MHz | 200 MHz * ⁴ | DBSC clock |
| CR ϕ | SCEG divider | Fixed | 400 MHz | 400 MHz | SCEG clock |
| CRD2 ϕ | SCEG divider | Fixed | 200 MHz | 200 MHz | SCEG clock |
| SDSRC ϕ | SDSRC divider | Fixed | 800 MHz | 800 MHz | Source clock of SDnH ϕ and SDn ϕ (n = 0 to 3) |
| SD0H ϕ | SD-IF0 divider | Variable | 800 MHz | Stopped | SD-IF0 clock |
| SD0 ϕ | SD-IF0 divider | Variable | 200 MHz | 50 MHz | SD-IF0 clock |

| Name | Clock Source | Frequency | | | Clock Domain, Function |
|--------------|----------------------------|----------------|------------|----------|--|
| | | Dividing ratio | Maximum | Initial | |
| SD1H ϕ | SD-IF1 divider | Variable | 800 MHz | Stopped | SD-IF1 clock |
| SD1 ϕ | SD-IF1 divider | Variable | 200 MHz | 50 MHz | SD-IF1 clock |
| SD2H ϕ | SD-IF2 divider | Variable | 800 MHz | Stopped | SD-IF2 clock |
| SD2 ϕ | SD-IF2 divider | Variable | 200 MHz | 50 MHz | SD-IF2 clock |
| SD3H ϕ | SD-IF3 divider | Variable | 800 MHz | Stopped | SD-IF3 clock |
| SD3 ϕ | SD-IF3 divider | Variable | 200 MHz | 50 MHz | SD-IF3 clock |
| RPC ϕ | RPC divider | Variable | 320 MHz | *2 | RPC-IF clock |
| RPCD2 ϕ | RPC divider | Variable | 160 MHz | *2 | RPC-IF clock |
| MSO ϕ | MSO divider | Variable | 133.33 MHz | 12.5 MHz | MSIOF clock |
| CANFD ϕ | CANFD divider | Variable | 80 MHz | 12.5 MHz | CANFD clock |
| HDMI ϕ | HDMI-IF divider | Variable | 25 MHz | Stopped | HDMI-IF clock |
| CSI0 ϕ | CSI0 divider | Variable | 25 MHz | Stopped | CSI2 clock |
| CP ϕ | EXTAL | Fixed | 16.66 MHz | *3 | Common peripheral clock, EXTAL \times 1/2 |
| CPEX ϕ | EXTAL | Fixed | 16.66 MHz | *3 | EXTAL \times 1/2 |
| RCLK | Common divider / EXTALR | Fixed | 32.89 kHz | *6 | |
| OSCCLK | Common divider | Fixed | 131.57 kHz | *7 | |

- Notes:
- The frequency of LB ϕ depends on the value of MD18.
 - The frequency of RPC ϕ depends on the value of MD4, MD3, MD2 and MD1.
When MD4 = H or L, MD3 = L, MD2 = H, MD1 = L, RPC ϕ = 320 MHz, RPCD2 ϕ = 160MHz.
When MD4 = H or L, MD3 = L, MD2 = H, MD1 = H, RPC ϕ = 160 MHz, RPCD2 ϕ = 80MHz.
In the other cases of MD4, MD3, MD2 and MD1 setting, RPC ϕ = 80 MHz, RPCD2 ϕ = 40MHz.
 - The frequency of CP ϕ is equal to EXTAL \times 1/2. For example, when the frequency of EXTAL is equal to 20 MHz, the frequency of CPCLK is equal to 10 MHz.
 - The initial frequency of ZB3 ϕ and ZB3D2 ϕ depends on the setting of MD19 and MD17. The values shown in the above table corresponds to the case of MD19 = L and MD17 = L.
 - These clocks are oscillated in the debug mode. Otherwise, they are stopped.
 - The initial value depends on MD14, MD13 and MD28.
When MD28 = H, RCLK = EXTALR.
When MD28 = L, MD14 = L and MD13 = H, RCLK = 32.89 (kHz)In the other setting, RCLK = 32.55 (kHz)
 - The initial value depends on MD14, MD13.
When MD14 = L and MD13 = H, OSCCLK = 131.57 (kHz).
In the other setting, OSCCLK = 130.20 (kHz)
 - Divider setting for listed clock in Table 11.4 must not be over than maximum clock frequency of each clocks.

| MD14 | MD13 | EXTAL (MHz) | EXTAL divider | | Internal RCLK | | OSCCLK | |
|------|------|-------------|----------------------|-----------------|----------------------|-----------------|----------------------|-----------------|
| | | | Fixed division ratio | Frequency (MHz) | Fixed division ratio | Frequency (kHz) | Fixed division ratio | Frequency (kHz) |
| 0 | 0 | 16.66 | \times 1/16 | 1.04 | \times 1/32 | 32.55 | \times 1/8 | 130.20 |
| 0 | 1 | 20.00 | \times 1/19 | 1.05 | \times 1/32 | 32.89 | \times 1/8 | 131.57 |
| 1 | 0 | 25.00 | \times 1/24 | 1.04 | \times 1/32 | 32.55 | \times 1/8 | 130.20 |
| 1 | 1 | 33.33 | \times 1/32 | 1.04 | \times 1/32 | 32.55 | \times 1/8 | 130.20 |

Table 11.5 List of Clocks [RZ/G2E]

| Name | Clock Source | Frequency | | | nonSSCG/ SSCG | Clock Domain, Function |
|--------|---------------------------------|----------------|----------------------|------------------|------------------|--|
| | | Dividing ratio | Frequency | Initial | | |
| Z2φ | SYS-CPU divider 2 | Variable | 1200 MHz | 1200 MHz | nonSSCG | System CPU (Cortex-A53) clock |
| ZGφ | 3DGE divider | Variable | 600 MHz | 600 MHz | nonSSCG/ SSCG | 3DGE clock |
| ZTRφ | Common divider | Variable | 266.66 MHz | 266.66 MHz *1 | SSCG | Trace interface clock |
| ZTφ | Common divider | Variable | 400 MHz | 400 MHz *1 | SSCG | Internal trace clock |
| ZXφ | Common divider | Fixed | 533.33 MHz | 533.33 MHz | SSCG | Cache coherent interconnect clock |
| S0φ | Common divider | Fixed | 800 MHz | 800 MHz | SSCG | |
| S1φ | Common divider | Fixed | 533.33 MHz | 533.33 MHz | SSCG | Clock source of S1D1φ, S1D2φ, S1D4φ |
| S2φ | Common divider | Fixed | 400 MHz | 400 MHz | SSCG | Clock source of S2D1φ, S2D2φ, S2D4φ |
| S3φ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | SSCG | Clock source of S3D1φ, S3D2φ, S3D4φ |
| S0D1φ | Common divider | Fixed | 800 MHz | 800 MHz | SSCG | |
| S0D3φ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | SSCG | |
| S0D6φ | Common divider | Fixed | 133.33 MHz | 133.33 MHz | SSCG | |
| S0D12φ | Common divider | Fixed | 66.66 MHz | 66.66 MHz | SSCG | |
| S0D24φ | Common divider | Fixed | 33.33 MHz | 33.33 MHz | SSCG | |
| S1D1φ | Common divider | Fixed | 533.33 MHz | 533.33 MHz | SSCG | |
| S1D2φ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | SSCG | |
| S1D4φ | Common divider | Fixed | 133.33 MHz | 133.33 MHz | SSCG | |
| S2D1φ | Common divider | Fixed | 400.00 MHz | 400.00 MHz | SSCG | |
| S2D2φ | Common divider | Fixed | 200.00 MHz | 200.00 MHz | SSCG | |
| S2D4φ | Common divider | Fixed | 100.00 MHz | 100.00 MHz | SSCG | |
| S3D1φ | Common divider | Fixed | 266.66 MHz | 266.66 MHz | SSCG | AXI-bus |
| S3D2φ | Common divider | Fixed | 133.33 MHz | 133.33 MHz | SSCG | |
| S3D4φ | Common divider | Fixed | 66.66 MHz | 66.66 MHz | SSCG | |
| S0D6Cφ | Common divider/ PECK divider | Variable | 133.33MHz/ 120MHz | 120MHz*2 | nonSSCG | |
| S3D1Cφ | Common divider/ PECK divider | Variable | 266.66MHz/ 240MHz | 240MHz*2 | nonSSCG | |
| S3D2Cφ | Common divider/ PECK divider | Variable | 133.33MHz/ 120MHz | 120MHz*2 | nonSSCG | |
| S3D4Cφ | Common divider/ PECK divider | Variable | 66.66MHz/ 60MHz | 60MHz*2 | nonSSCG | |
| LBφ | Common divider | Fixed | 66.66 MHz | 66.66 MHz*1 | SSCG | LBSC, CLKOUT |
| CLφ | Common divider | Fixed | 33.33 MHz | 33.33 MHz | SSCG | CPG |
| ZB3φ | DDR divider | Variable | 464MHz/ 400MHz | 400 MHz *3 | nonSSCG | DBSC clock |
| ZB3D2φ | DDR divider | Variable | 232MHz/ 200MHz | 200 MHz *3 | nonSSCG | DBSC clock |

| Name | Clock Source | Frequency | | | nonSSCG/ SSCG | Clock Domain, Function |
|--------------|----------------------|----------------|----------------------|----------|------------------|--|
| | | Dividing ratio | Frequency | Initial | | |
| CR ϕ | SCEG divider | Fixed | 400 MHz | 400 MHz | SSCG | SCEG clock |
| CRD2 ϕ | SCEG divider | Fixed | 200 MHz | 200 MHz | SSCG | SCEG clock |
| SDSRC ϕ | SDSRC divider | Fixed | 800 MHz | 800 MHz | SSCG | Source clock of SDnH ϕ and SDn ϕ (n = 0, 1, 3) |
| SD0H ϕ | SD-IF0 divider | Variable | 800 MHz | Stopped | SSCG | SD-IF0 |
| SD0 ϕ | SD-IF0 divider | Variable | 200 MHz | 50 MHz | SSCG | SD-IF0 |
| SD1H ϕ | SD-IF1 divider | Variable | 800 MHz | Stopped | SSCG | SD-IF1 |
| SD1 ϕ | SD-IF1 divider | Variable | 200 MHz | 50 MHz | SSCG | SD-IF1 |
| SD3H ϕ | SD-IF3 divider | Variable | 800 MHz | Stopped | SSCG | SD-IF3 |
| SD3 ϕ | SD-IF3 divider | Variable | 200 MHz | 50 MHz | SSCG | SD-IF3 |
| RPC ϕ | RPC divider | Variable | 300 MHz/ 160MHz | *4 | nonSSCG/ SSCG | RPC-IF clock |
| RPCD2 ϕ | RPC divider | Variable | 150 MHz/ 80 MHz | *4 | nonSSCG/ SSCG | RPC-IF clock |
| ZA2 ϕ | ZA2CK divider | Fixed | 200MHz | 200MHz | nonSSCG | ADG clock |
| ZA8 ϕ | ZA8CK divider | Fixed | 600MHz | 600MHz | nonSSCG | ADG clock |
| Z2D ϕ | Z2DCK divider | Variable | 400MHz- 266.66MHz | 400MHz | nonSSCG/ SSCG | |
| CANFD ϕ | CANCK divider | Variable | 80MHz | 12.5MHz | nonSSCG | CANFD clock |
| MSO ϕ | MSIOF divider | Variable | 133.33MHz | 12.5MHz | SSCG | MSIOF clock |
| CSI0 ϕ | CSI0 divider | Variable | 25 MHz | Stopped | SSCG | CSI2 clock |
| RCLK ϕ | RCK divider / OCO | Variable | 32.8kHz/ 31.25kHz | 31.25kHz | — | |
| OSC ϕ | OSC divider | Fixed | 125kHz | 125kHz | — | |
| LV0 ϕ | LV0CK divider | Variable | 160MHz- 0.2MHz | 0.39MHz | nonSSCG/ SSCG | LVDS clock |
| LV1 ϕ | LV1CK divider | Variable | 160MHz- 0.2MHz | 0.39MHz | nonSSCG/ SSCG | LVDS clock |
| CP ϕ | EXTAL | Fixed | 24MHz | 24MHz | — | |
| CPEX ϕ | EXTAL | Fixed | 12MHz | 12MHz | — | EXTAL \times 1/4 |

- Notes:
1. These clocks are oscillated in the debug mode. Otherwise, they are stopped.
 2. When MD12 = L, source clock of S3DnC ϕ is S3 ϕ , and source clock of S1D4C ϕ is S1 ϕ . When MD12 = H, source clock of S3DnC ϕ , S1D4C ϕ is PE ϕ .
 3. The initial frequency of ZB3 ϕ and ZB3D2 ϕ depends on the setting of MD19. The values shown in the above table corresponds to the case of MD19 = L.
 4. The frequency of RPC ϕ depends on the value of MD4, MD3 MD2 and MD1.
 When MD4 = L, MD3 = L, MD2 = L, MD1 = H, RPC ϕ = 80 MHz, RPCD2 ϕ = 40MHz.
 When MD4 = H or L, MD3 = L, MD2 = H, MD1 = L, RPC ϕ = 300 MHz, RPCD2 ϕ = 150MHz.
 When MD4 = H or L, MD3 = L, MD2 = H, MD1 = H, RPC ϕ = 160 MHz, RPCD2 ϕ = 80MHz.
 When MD4 = L, MD3 = H, MD2 = L, MD1 = L, RPC ϕ = 80 MHz, RPCD2 ϕ = 40MHz.
 In the other cases of MD4, MD3 MD2 and MD1 setting RPC ϕ = 75 MHz, RPCD2 ϕ = 37.5MHz.
 5. Divider setting for listed clock in Table 11.5 must not be over than maximum clock frequency of each clocks.

11.1.5 Clock Operating Modes

Table 11.6 MD9 Settings

| MD9 | EXTAL/XTAL Pin Settings |
|-----|--|
| 0 | Inputs an external clock to the EXTAL pin. |
| 1 | Connects the crystal resonator to the EXTAL and XTAL pins. |

Table 11.7 PLL Initial Multiplication Ratio [RZ/G2H]

| MD14 | MD13 | MD19 | MD17 | EXTAL Input Frequency (MHz) | Divider Setting of EXTAL Input | PLLn reference clock (n = 0 to 4) (MHz) | PLL Multiplication Ratio | | | | |
|------|------|------|------|--------------------------------------|--|---|--------------------------|--------|--------|--------|--------|
| | | | | | | | PLL0*1 | PLL1*2 | PLL2*3 | PLL3*4 | PLL4*5 |
| 0 | 0 | 0 | 0 | 16.66 | × 1 | 16.66 | × 180 | × 192 | × 144 | × 192 | × 144 |
| 0 | 0 | 0 | 1 | 16.66 | × 1 | 16.66 | × 180 | × 192 | × 144 | × 128 | × 144 |
| 0 | 0 | 1 | 0 | Prohibited setting | | | | | | | |
| 0 | 0 | 1 | 1 | 16.66 | × 1 | 16.66 | × 180 | × 192 | × 144 | × 192 | × 144 |
| 0 | 1 | 0 | 0 | 20 | × 1 | 20 | × 150 | × 160 | × 120 | × 160 | × 120 |
| 0 | 1 | 0 | 1 | 20 | × 1 | 20 | × 150 | × 160 | × 120 | × 106 | × 120 |
| 0 | 1 | 1 | 0 | Prohibited setting | | | | | | | |
| 0 | 1 | 1 | 1 | 20 | × 1 | 20 | × 150 | × 160 | × 120 | × 160 | × 120 |
| 1 | 0 | 0 | 0 | 25 | × 1 | 25 | × 120 | × 128 | × 96 | × 128 | × 96 |
| 1 | 0 | 0 | 1 | 25 | × 1 | 25 | × 120 | × 128 | × 96 | × 84 | × 96 |
| 1 | 0 | 1 | 0 | Prohibited setting | | | | | | | |
| 1 | 0 | 1 | 1 | 25 | × 1 | 25 | × 120 | × 128 | × 96 | × 128 | × 96 |
| 1 | 1 | 0 | 0 | 33.33 | × 1/2 | 16.66 | × 180 | × 192 | × 144 | × 192 | × 144 |
| 1 | 1 | 0 | 1 | 33.33 | × 1/2 | 16.66 | × 180 | × 192 | × 144 | × 128 | × 144 |
| 1 | 1 | 1 | 0 | Prohibited setting | | | | | | | |
| 1 | 1 | 1 | 1 | 33.33 | × 1/2 | 16.66 | × 180 | × 192 | × 144 | × 192 | × 144 |

Notes: 1. PLL to create Z ϕ for Cortex-A57

Example of frequency calculation (in the case MD14 = H, MD13 = H, MD19 = L, and MD17 = L)

Maximum frequency of Z ϕ = 33.33 MHz × 1/2 (fixed divider) × 180 (PLL multiplication ratio) × 1/2 (fixed divider) × 32/32 (SYS-CPU clock divider 1 setting) = 1500 MHz

2. PLL to create clocks related to AXI and so on.
3. PLL to create Z2 ϕ for Cortex-A53
4. PLL to create ZB3 ϕ and ZB3D2 ϕ
5. PLL to create ZG ϕ

Table 11.8 PLL Initial Multiplication Ratio [RZ/G2M V1.3, RZ/G2M V3.0]

| MD14 | MD13 | MD12 | MD19 | MD17 | EXTAL Input Frequency (MHz) | Divider Setting of EXTAL Input | PLLn reference clock (n = 0 to 4) (MHz) | PLL Multiplication Ratio | | | | |
|------|------|------|------|------|--------------------------------------|---|---|--------------------------|--------|--------|--------|--------|
| | | | | | | | | PLL0*1 | PLL1*2 | PLL2*3 | PLL3*4 | PLL4*5 |
| 0 | 0 | 0 | 0 | 0 | 16.66 | × 1 | 16.66 | × 180 | × 192 | × 144 | × 192 | × 144 |
| 0 | 0 | 0 | 0 | 1 | 16.66 | × 1 | 16.66 | × 180 | × 192 | × 144 | × 128 | × 144 |
| 0 | 0 | 0 | 1 | 0 | Prohibited setting | | | | | | | |
| 0 | 0 | 0 | 1 | 1 | 16.66 | × 1 | 16.66 | × 180 | × 192 | × 144 | × 192 | × 144 |
| 0 | 1 | 0 | 0 | 0 | 20 | × 1 | 20 | × 150 | × 160 | × 120 | × 160 | × 120 |
| 0 | 1 | 0 | 0 | 1 | 20 | × 1 | 20 | × 150 | × 160 | × 120 | × 106 | × 120 |
| 0 | 1 | 0 | 1 | 0 | Prohibited setting | | | | | | | |
| 0 | 1 | 0 | 1 | 1 | 20 | × 1 | 20 | × 150 | × 160 | × 120 | × 160 | × 120 |
| 1 | 0 | 0 | 0 | 0 | 25 | × 1 | 25 | × 120 | × 128 | × 96 | × 128 | × 96 |
| 1 | 0 | 0 | 0 | 1 | 25 | × 1 | 25 | × 120 | × 128 | × 96 | × 84 | × 96 |
| 1 | 0 | 0 | 1 | 0 | Prohibited setting | | | | | | | |
| 1 | 0 | 0 | 1 | 1 | 25 | × 1 | 25 | × 120 | × 128 | × 96 | × 128 | × 96 |
| 1 | 1 | 0 | 0 | 0 | 33.33 | × 1/2 | 16.66 | × 180 | × 192 | × 144 | × 192 | × 144 |
| 1 | 1 | 0 | 0 | 1 | 33.33 | × 1/2 | 16.66 | × 180 | × 192 | × 144 | × 128 | × 144 |
| 1 | 1 | 0 | 1 | 0 | Prohibited setting | | | | | | | |
| 1 | 1 | 0 | 1 | 1 | 33.33 | × 1/2 | 16.66 | × 180 | × 192 | × 144 | × 192 | × 144 |

Notes: 1. PLL to create Z ϕ for Cortex-A57

Example of frequency calculation (in the case MD14 = H, MD13 = H, MD19 = L, and MD17 = L)

Maximum frequency of Z ϕ = 33.33 MHz × 1/2 (fixed divider) × 180 (PLL multiplication ratio) × 1/2 (fixed divider) × 32/32 (SYS-CPU clock divider 1 setting) = 1500 MHz

2. PLL to create clocks related to AXI and so on.
3. PLL to create Z2 ϕ for Cortex-A53
4. PLL to create ZB3 ϕ and ZB3D2 ϕ
5. PLL to create ZG ϕ

Table 11.9 PLL Initial Multiplication Ratio [RZ/G2N]

| MD14 | MD13 | MD19 | MD17 | EXTAL Input Frequency (MHz) | Divider Setting of EXTAL Input | PLLn reference clock (n = 0, 1, 3, 4) (MHz) | PLL Multiplication Ratio | | | |
|------|------|------|------|--------------------------------------|---|---|--------------------------|--------|--------|--------|
| | | | | | | | PLL0*1 | PLL1*2 | PLL3*3 | PLL4*4 |
| 0 | 0 | 0 | 0 | 16.66 | × 1 | 16.66 | × 180 | × 192 | × 192 | × 144 |
| 0 | 0 | 0 | 1 | 16.66 | × 1 | 16.66 | × 180 | × 192 | × 128 | × 144 |
| 0 | 0 | 1 | 0 | Prohibited setting | | | | | | |
| 0 | 0 | 1 | 1 | 16.66 | × 1 | 16.66 | × 180 | × 192 | × 192 | × 144 |
| 0 | 1 | 0 | 0 | 20 | × 1 | 20 | × 150 | × 160 | × 160 | × 120 |
| 0 | 1 | 0 | 1 | 20 | × 1 | 20 | × 150 | × 160 | × 106 | × 120 |
| 0 | 1 | 1 | 0 | Prohibited setting | | | | | | |
| 0 | 1 | 1 | 1 | 20 | × 1 | 20 | × 150 | × 160 | × 160 | × 120 |
| 1 | 0 | 0 | 0 | 25 | × 1 | 25 | × 120 | × 128 | × 128 | × 96 |
| 1 | 0 | 0 | 1 | 25 | × 1 | 25 | × 120 | × 128 | × 84 | × 96 |
| 1 | 0 | 1 | 0 | Prohibited setting | | | | | | |
| 1 | 0 | 1 | 1 | 25 | × 1 | 25 | × 120 | × 128 | × 128 | × 96 |
| 1 | 1 | 0 | 0 | 33.33 | × 1/2 | 16.66 | × 180 | × 192 | × 192 | × 144 |
| 1 | 1 | 0 | 1 | 33.33 | × 1/2 | 16.66 | × 180 | × 192 | × 128 | × 144 |
| 1 | 1 | 1 | 0 | Prohibited setting | | | | | | |
| 1 | 1 | 1 | 1 | 33.33 | × 1/2 | 16.66 | × 180 | × 192 | × 192 | × 144 |

Notes: 1. PLL to create Z ϕ for Cortex-A57

Example of frequency calculation (in the case MD14 = H, MD13 = H, MD19 = L, and MD17 = L)

Maximum frequency of Z ϕ = 33.33 MHz × 1/2 (fixed divider) × 180 (PLL multiplication ratio) × 1/2(fixed divider) × 32/32 (SYS-CPU clock divider 1 setting) = 1500 MHz

2. PLL to create clocks related to AXI and so on.
3. PLL to create ZB3 ϕ and ZB3D2 ϕ
4. PLL to create ZG ϕ

Table 11.10 PLL Initial Multiplication Ratio [RZ/G2E]

| MD19 | EXTAL Input Frequency (MHz) | Divider Setting of EXTAL Input | PLL _n reference clock (n = 0, 1, 3) (MHz) | PLL Multiplication Ratio | | |
|------|-----------------------------|--------------------------------|--|--------------------------|--------------------|--------------------|
| | | | | PLL0* ¹ | PLL1* ² | PLL3* ³ |
| 0 | 48 | x 1 | 48 | 200/2 | 100/3 | 100/3 |
| 1 | 48 | x 1 | 48 | 200/2 | 100/3 | 58/3 |

Notes: 1. PLL to create Z2 ϕ for Cortex-A53 and ZG ϕ at nonSSCG clock

2. PLL to create SYSTEM

3. PLL to create ZB3 ϕ and ZB3D2 ϕ

MD19 = 0 PLL3 = 1600MHz ZB3 ϕ = PLL3 / 4 = 400MHz

MD19 = 1 PLL3 = 928MHz ZB3 ϕ = PLL3 / 2 = 464MHz

Table 11.11 MD18 Settings

| MD18 | LB ϕ (CLKOUT) Frequency Setting |
|------|--------------------------------------|
| 0 | Division ratio = 1/12 of S0 ϕ |
| 1 | Division ratio = 1/18 of S0 ϕ |

Table 11.12 MD28 Settings [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| MD28 | MD28 Settings |
|------|-----------------------|
| 0 | Internal RCLK select. |
| 1 | EXTALR select. |

Table 11.13 MD12 Settings [RZ/G2E]

| MD12 | MD12 Settings |
|------|---------------------------|
| 0 | PLL1 nonSSCG Clock select |
| 1 | PLL1 SSCG Clock select |

11.1.6 Register Configuration

Table 11.14 shows the CPG register configuration. Table 11.15 shows the register states in each operating mode.

32bit-width access is only available.

Table 11.14 Register Configurations

| Register Name | Abbreviation | R/W | Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | |
|--|--------------|-----|-------------|---------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| CPG Write Protect Control Register | CPGWPCR | R/W | H'E615_0904 | H'0000_0001 | 32 | √ | √ | √ | √ |
| CPG Write Protect Register | CPGWPR | R/W | H'E615_0900 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Frequency control register B | FRQCRB | R/W | H'E615_0004 | H'0032_0235* | 32 | √ | √ | √ | √ |
| Frequency control register C | FRQCRC | R/W | H'E615_00E0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PLL Enable Control Register | PLLECR | R/W | H'E615_00D0 | H'0000_1F1F | 32 | √ | √ | √ | √ |
| PLL0 control register | PLL0CR | R/W | H'E615_00D8 | H'---0_-000 | 32 | √ | √ | √ | √ |
| PLL2 control register | PLL2CR | R/W | H'E615_002C | H'--00_00-0 | 32 | √ | √ | — | — |
| PLL3 control register | PLL3CR | R/W | H'E615_00DC | H'---0_00-0 | 32 | √ | √ | √ | √ |
| PLL4 control register | PLL4CR | R/W | H'E615_01F4 | H'--00_00-0 | 32 | √ | √ | √ | — |
| SD-IF0 clock frequency control register | SD0CKCR | R/W | H'E615_0074 | H'0000_0209* | 32 | √ | √ | √ | √ |
| SD-IF1 clock frequency control register | SD1CKCR | R/W | H'E615_0078 | H'0000_0209 | 32 | √ | √ | √ | √ |
| SD-IF2 clock frequency control register | SD2CKCR | R/W | H'E615_0268 | H'0000_0209 | 32 | √ | √ | √ | — |
| SD-IF3 clock frequency control register | SD3 CKCR | R/W | H'E615_026C | H'0000_0209 | 32 | √ | √ | √ | √ |
| RPC-IF clock frequency control register | RPCCKCR | R/W | H'E615_0238 | H'0000_0--- | 32 | √ | √ | √ | √ |
| CAN-FD clock frequency control register | CANFDCKCR | R/W | H'E615_0244 | H'0000_003F | 32 | √ | √ | √ | √ |
| MSIOF clock frequency control register | MSOCKCR | R/W | H'E615_0014 | H'0000_003F | 32 | √ | √ | √ | √ |
| HDMI-IF clock frequency control register | HDMICKCR | R/W | H'E615_0250 | H'0000_011F | 32 | √ | √ | √ | — |
| CSI0 clock frequency control register | CSI0CKCR | R/W | H'E615_000C | H'0000_011F | 32 | √ | √ | √ | √ |
| RCLK frequency control register | RCKCR | R/W | H'E615_0240 | H'0000_00-- | 32 | — | — | — | √ |
| LV0 Clock Control Register | LV0CKCR | R/W | H'E615_04CC | H'3F3F_0040 | 32 | — | — | — | √ |

| Register Name | Abbreviation | R/W | Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | |
|------------------------------|--------------|-----|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| LV1 Clock Control Register | LV1CKCR | R/W | H'E615_04D0 | H'3F3F_0040 | 32 | — | — | — | √ |
| ZA2 Clock Control Register | ZA2CKCR | R/W | H'E615_04DC | H'0000_0000 | 32 | — | — | — | √ |
| ZA8 Clock Control Register | ZA8CKCR | R/W | H'E615_04E0 | H'0000_0000 | 32 | — | — | — | √ |
| Z2D Clock Control Register | Z2DCKCR | R/W | H'E615_04E8 | H'0000_0009 | 32 | — | — | — | √ |
| Frequency control register D | FRQCRD | R/W | H'E615_00E4 | H'000-_0-0- | 32 | √ | √ | √ | — |

Note: Do not write to any addresses other than listed above. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed above are undefined.

Note: *Initial value is different for each product. For detail, refer to Register Description.

Table 11.15 Register States in Each Operating Mode

| Register | Power-On Reset |
|---------------|----------------|
| All Registers | Initialized |

11.2 Register Description

Legend for Register Description

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

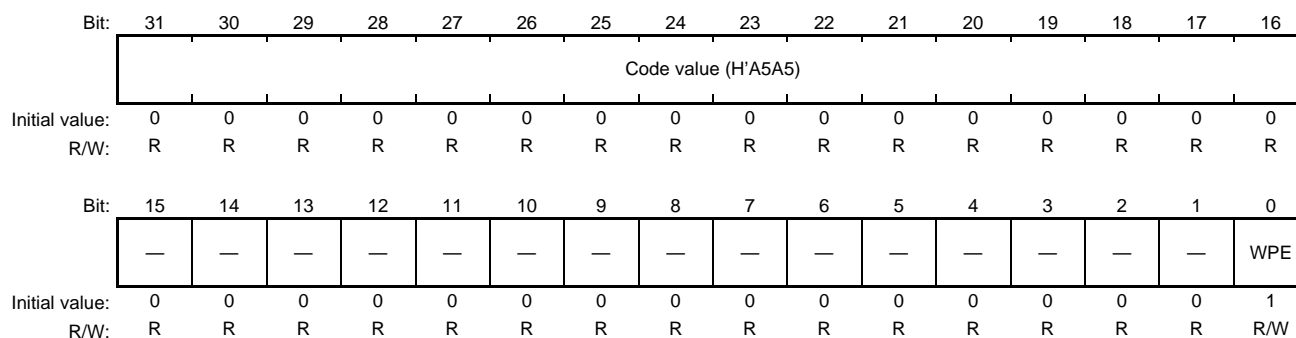
R: Read-only.

W: Write-only. The read value is undefined.

11.2.1 CPG Write Protect Control Register (CPGWPCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CPGWPCR is a 32-bit readable/writable register.



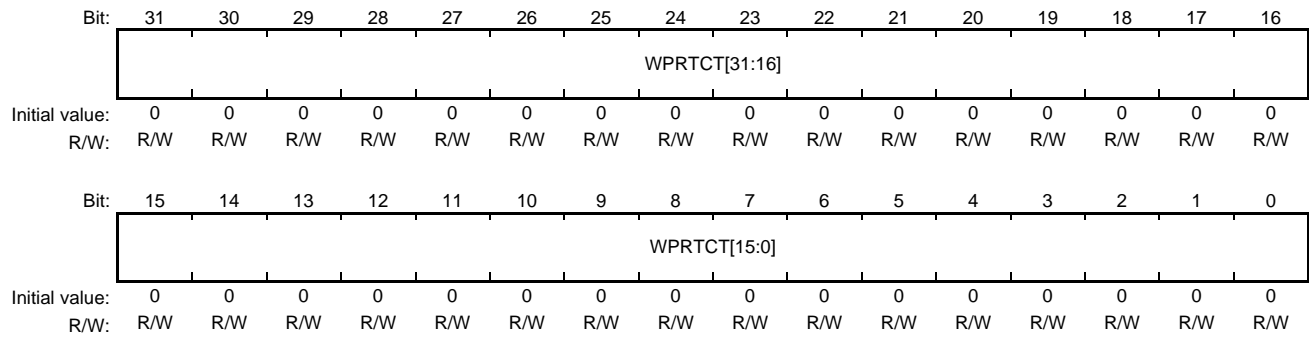
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Code value (H'A5A5) When read, returns 0. |
| 15 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | WPE | B'1 | R/W | Write protect enable 0: Disable the write protect 1: Enable the write protect |

Refer to section 11.4.1 Write protection function for CPG registers.

11.2.2 CPG Write Protect Register (CPGWPR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CPGWPR is a 32-bit readable/writable register. CPGWPR enables/disables writing to all registers in CPG except for CPGWPCR.



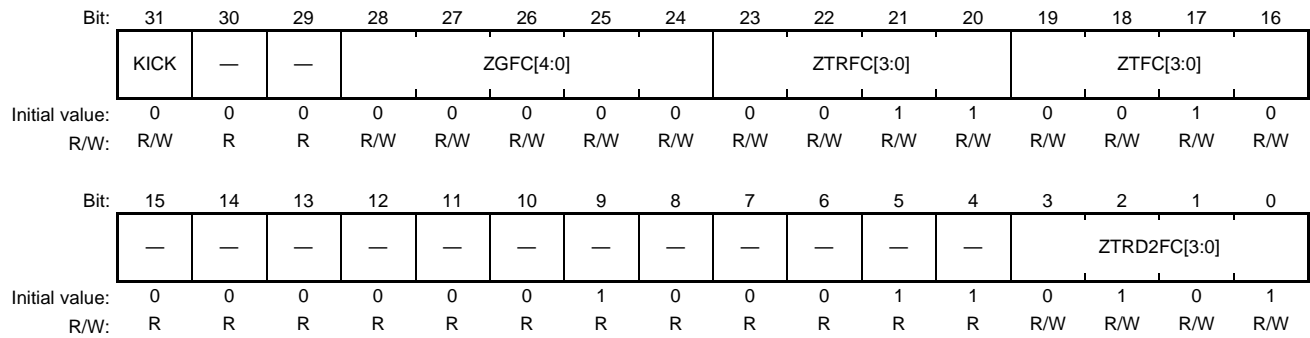
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 0 | WPRTCT [31:0] | All 0 | R/W | Writing a value to CPG registers is enabled by writing the inverse of the value to this register. |

11.2.3 Frequency Control Register B (FRQCRB)

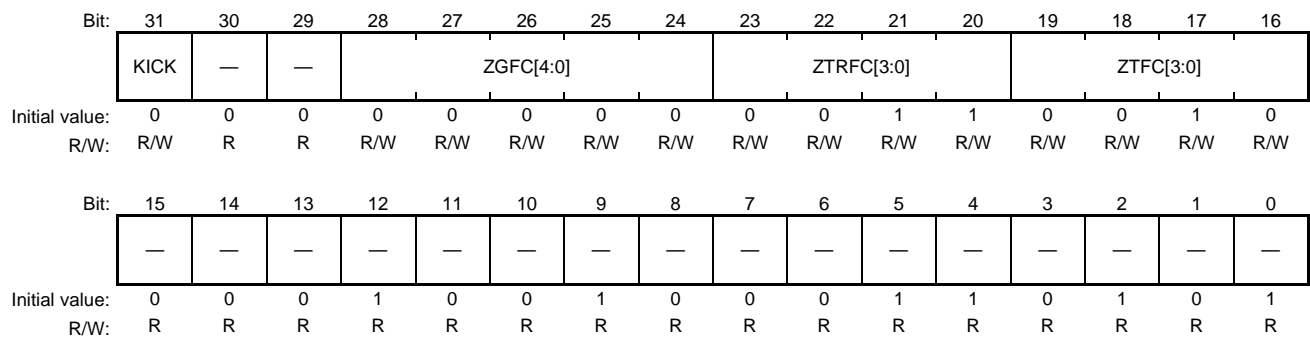
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

FRQCRB is a 32-bit readable/writable register. This register specifies the frequency division ratios of GPU Clock(ZGφ), Debug Trace port clock (ZTRφ), Debug Trace bus clock (ZTφ), and Debug clock (ZTRD2φ).

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]



[RZ/G2E]



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 | KICK | B'0 | R/W | KICK bit Setting 1 to this register activates the FRQCRB and FRQCRC setting. 0: Does not activate the FRQCRB and FRQCRC settings. 1: Activates the FRQCRB, FRQCRC settings. This bit is automatically cleared to '0' when the frequency division setting is completed after '1' is written to this bit. |
| 30 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 29 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 28 to 24 | ZGFC[4:0] | B'0_0000 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] GPU Clock (ZG ϕ) Frequency Division Ratio B'0_0000: $\times 1/1$ B'0_1011: $\times 21/32$ B'1_0000: $\times 1/2$ B'1_0101: $\times 11/32$ Other values: Setting prohibited. |
| | ZGFC[4:0] | B'0_0000 | R/W | [RZ/G2E] GPU Clock (ZG ϕ) Frequency Division Ratio B'0_00xx: 600 MHz (nonSSCG) B'0_01xx: 300 MHz (nonSSCG) B'0_1001: 400 MHz (SSCG) B'0_1010: 200 MHz (SSCG) B'0_1100: 533.33 MHz (SSCG) B'0_1101: 266.66 MHz (SSCG) B'0_1110: 133.33 MHz (SSCG) Other values: Setting prohibited |
| 23 to 20 | ZTRFC[3:0] | B'0011 | R/W | Debug Trace port Clock (ZTR ϕ) Frequency Division Ratio B'0011: $\times 1/6$ B'0100: $\times 1/8$ B'0101: $\times 1/12$ B'0110: $\times 1/16$ B'0111: $\times 1/18$ B'1000: $\times 1/24$ Other values: Setting prohibited Note: ZTR ϕ is supplied only in debugging mode. |
| 19 to 16 | ZTFC[3:0] | B'0010 | R/W | Debug Trace bus Clock (ZT ϕ) Frequency Division Ratio B'0010: $\times 1/4$ B'0011: $\times 1/6$ B'0100: $\times 1/8$ B'0101: $\times 1/12$ B'0110: $\times 1/16$ B'0111: $\times 1/18$ B'1000: $\times 1/24$ Other values: Setting prohibited Note: ZT ϕ is supplied only in debugging mode. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 15 to 12 | — | All 0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Reserved These bits are always read as 0. The write value should always be 0. |
| | — | B'0001 | R | [RZ/G2E] Reserved These bits are always read as 0001. The write value should always be 0001. |
| 11 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 10 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 9 | — | B'1 | R | Reserved This bit is always read as 1. The write value should always be 1. |
| 8 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 5 | — | B'1 | R | Reserved This bit is always read as 1. The write value should always be 1. |
| 4 | — | B'1 | R | Reserved This bit is always read as 1. The write value should always be 1. |
| 3 to 0 | ZTRD2FC[3:0] | B'0101 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Debug Clock (ZTRD2 ϕ) Frequency Division Ratio B'0101: $\times 1/12$ B'0110: $\times 1/16$ B'0111: $\times 1/18$ B'1000: $\times 1/24$ Other values: Setting prohibited Note: ZTRD2 ϕ is supplied only in debugging mode. |
| | — | B'0101 | R | [RZ/G2E] Reserved These bits are always read as B'0101. The write value should always be B'0101. |

11.2.4 Frequency Control Register C (FRQCRC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

When the values of ZTR ϕ , ZT ϕ or ZTRD2 ϕ are not changed, FRQCRB.KICK bit needs setting to 1 again after FRQCRB.ZTRFC, FRQCRB.ZTFC, or FRQCRB.ZTRD2FC are stored.

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]

FRQCRC is a 32-bit readable/writable register. This register specifies the frequency division ratios of the System CPU (Cortex-A57) clock (Z ϕ), System CPU (Cortex-A53) clock (Z2 ϕ).

[RZ/G2N]

FRQCRC is a 32-bit readable/writable register. This register specifies the frequency division ratios of the System CPU (Cortex-A57) clock (Z ϕ).

[RZ/G2E]

FRQCRC is a 32-bit readable/writable register. This register specifies the frequency division ratios of the System CPU (Cortex-A53) clock (Z2 ϕ).

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----------|-----|-----|-----|-----|----|----|-----------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | ZFC[4:0] | | | | — | — | — | Z2FC[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

[RZ/G2E]

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----------|-----|-----|-----|-----|----|----|----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | Z2FC[4:0] | | | | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 to 16 | — | B'0_000 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | — | 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 14, 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 8 | ZFC[4:0] | B'0_000 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] AP-System Core (Cortex-A57) Clock (Z ϕ) Frequency Division Ratio B'0_0000: $\times 32/32$ B'0_0001: $\times 31/32$: B'1_1110: $\times 2/32$ B'1_1111: $\times 1/32$ Other values: Setting prohibited |
| | Z2FC[4:0] | B'0_000 | R/W | [RZ/G2E] AP-System Core (Cortex-A53) Clock (Z2 ϕ) Frequency Division Ratio B'0_0000: $\times 32/32$ B'0_0001: $\times 31/32$: B'1_1110: $\times 2/32$ B'1_1111: $\times 1/32$ Other values: Setting prohibited |
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | Z2FC[4:0] | B'0_000 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] AP-System Core (Cortex-A53) Clock (Z2 ϕ) Frequency Division Ratio B'0_0000: $\times 32/32$ B'0_0001: $\times 31/32$: B'1_1110: $\times 2/32$ B'1_1111: $\times 1/32$ Other values: Setting prohibited |
| | — | All 0 | R/W | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |

11.2.5 PLL Enable Control Register (PLLECR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

PLLECR is a 32-bit readable/writable register.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|------------|------------|------------|------------|------------|----|----|----|-------|-------|-------|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | PLL4S T | PLL3S T | PLL2S T | PLL1S T | PLL0S T | — | — | — | PLL4E | PLL3E | PLL2E | — | PLL0E |
| Initial value: | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | PLL4ST | B'1 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] PLL circuit 4 status Displays PLL circuit 4 status (on or off). 0: PLL circuit 4 is turned off. The main clock is supplied. 1: PLL circuit 4 is turned on. The output from PLL circuit 4 is supplied. |
| | — | B'1 | R | [RZ/G2E] Reserved This bit is always read as 1. The write value should always be 1. |
| 11 | PLL3ST | B'1 | R | PLL circuit 3 status Displays PLL circuit 3 status (on or off). 0: PLL circuit 3 is turned off. The main clock is supplied. 1: PLL circuit 3 is turned on. The output from PLL circuit 3 is supplied. |
| 10 | PLL2ST | B'1 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] PLL circuit 2 status Displays PLL circuit 2 status (on or off). 0: PLL circuit 2 is turned off. The main clock is supplied. 1: PLL circuit 2 is turned on. The output from PLL circuit 2 is supplied. |
| | — | B'1 | R | [RZ/G2N, RZ/G2E] Reserved This bit is always read as 1. The write value should always be 1. |
| 9 | PLL1ST | B'1 | R | PLL circuit 1 status Displays PLL circuit q status (on or off). 0: PLL circuit 1 is turned off. The main clock is supplied. 1: PLL circuit 1 is turned on. The output from PLL circuit 1 is supplied. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 8 | PLL0ST | B'1 | R | PLL circuit 0 status Displays PLL circuit 0 status (on or off). 0: PLL circuit 0 is turned off. The main clock is supplied. 1: PLL circuit 0 is turned on. The output from PLL circuit 0 is supplied. |
| 7 to 5 | — | B'000 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | PLL4E | B'1 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] PLL circuit 4 enable Turns PLL circuit 4 on or off. Actual status of PLL4 is shown in PLL4ST bit. 0: Turns off PLL circuit 4 1: Turns on PLL circuit 4 |
| | — | B'1 | R | [RZ/G2E] Reserved This bit is always read as 1. The write value should always be 1. |
| 3 | PLL3E | B'1 | R/W | PLL circuit 3 enable Turns PLL circuit 3 on or off. Actual status of PLL3 is shown in PLL3ST bit. 0: Turns off PLL circuit 3 1: Turns on PLL circuit 3 |
| 2 | PLL2E | B'1 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] PLL circuit 2 enable Turns PLL circuit 2 on or off. Actual status of PLL2 is shown in PLL2ST bit. 0: Turns off PLL circuit 2 1: Turns on PLL circuit 2 |
| | — | B'1 | R | [RZ/G2N, RZ/G2E] Reserved This bit is always read as 1. The write value should always be 1. |
| 1 | — | B'1 | R | Reserved This bit is always read as 1. The write value should always be 1. |
| 0 | PLL0E | B'1 | R/W | PLL circuit 0 enable Turns PLL circuit 0 on or off. Actual status of PLL0 is shown in PLL0ST bit. 0: Turns off PLL circuit 0 1: Turns on PLL circuit 0 |

11.2.6 PLL0 Control Register (PLL0CR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

PLL0CR is a 32-bit readable/writable register. This register specifies the multiplication ratio of PLL circuit 0. When writing to this register, change only bits STC[6:0] (Read-modify-write).

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| | | | | | | | | | | | | | | | | | |
|----------------|----|----------|-----|---------|-----|-----|-----|-----|----|----|----|----|----|----|----|----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | STC[6:0] | | | | | | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | —*1 | —*1 | —*1 | —*1 | —*1 | —*1 | —*1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | REF DIV | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | —*2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R/W | R | R | R | R | R | R | R | R | R | R | R | R | |

[RZ/G2E]

| | | | | | | | | | | | | | | | | |
|----------------|----|-----|-------|---------|-----|-----|-----|-----|----|----|-----|-----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | CKSEL | REF DIV | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | —*3 | —*2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 30 to 24 | STC[6:0] | *1 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] PLL Circuit 0 Multiplication Ratio PLL circuit 0 performs multiplication with a ratio of (setting + 1) × 2 The frequency has the following relationship. (PLL0VCO) = (PLL0 reference clock) × (setting+1) × 2 (SYS-CPU divider 1 input) = PLL0VCO × 1/2 (fixed divider) B'011_1011: x (60 × 2) = x120 B'011_1111: x (64 × 2) = x128 B'100_0000: x (65 × 2) = x130 ... B'100_0101: x (70 × 2) = x140 B'100_1010: x (75 × 2) = x150 B'101_1001: x (90 × 2) = x180 For example, Z ϕ = 1500MHz with EXTAL = 20MHz, STC[6:0] = B'100_1010. PLL0VCO = 20 MHz × (75 × 2) = 3000MHz. (SYS-CPU divider 1 input) = 3000MHz × 1/2 = 1500MHz. When SYS-CPU divider 1 setting is x 32/32, Z ϕ = 1500MHz × 32/32 = 1500MHz. [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Set PLL0VCO between 2000 MHz and 3000 MHz |
| — | — | H'63 | R/W | [RZ/G2E] Reserved These bits are always read as H'63. The write value should always be H'63. |
| 23 to 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 to 20 | — | B'00 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Reserved These bits are always read as 0. The write value should always be 0. |
| — | — | B'01 | R/W | [RZ/G2E] Reserved These bits are always read as 01. The write value should always be 01. |
| 19 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 | — | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| | CKSEL | *3 | R/W | [RZ/G2E] Peripheral clock source select S3D1CCLK, S3D2CCLK, S3D4CCLK, S0D6CCLK 0: PLL1 1: PLL0 |
| 12 | REFDIV | *2 | R/W | PLL0 reference clock division ratio 0: x 1/1 1: x 1/2 Don't change from initial value. |
| 11 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

- Notes:
1. Initial value of STC[6:0] is determined by MD14 and MD13 as shown in the Table 11.7 to 11.10.
 2. When MD14 = 1 and MD13 = 1, the initial value is 1. When the other mode setting, the initial value is 0. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
The initial value is 0. [RZ/G2E]
 3. Initial value of CKSEL is MD12

11.2.7 PLL2 Control Register (PLL2CR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | — |

PLL2CR is a 32-bit readable/writable register. This register specifies the multiplication ratio of PLL circuit 2. When writing to this register, change only bits STC[6:0] (Read-modify-write).

| | | | | | | | | | | | | | | | | | |
|----------------|----|----------|-----|-----|-----|-----|-----|-----|----|----|---------|----|----|----|----|----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | STC[6:0] | | | | | | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | —*1 | —*1 | —*1 | —*1 | —*1 | —*1 | —*1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | — | — | — | — | — | REF DIV | — | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | —*2 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R | R | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 31 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|--|-----|---|
| 30 to 24 | STC[6:0] | *1 | R/W | <p>PLL Circuit 2 Multiplication Ratio</p> <p>PLL circuit 2 performs multiplication with a ratio of (setting + 1) × 2</p> <p>The frequency has the following relationship.</p> $(PLL2VCO) = (PLL2 \text{ reference clock}) \times (\text{setting} + 1) \times 2$ $(\text{SYS-CPU divider 2 input}) = PLL2VCO \times 1/2 \text{ (fixed divider)}$ <p>B'010_0111: x (40 × 2) = x80 ... B'010_1111: x (48 × 2) = x96 ... B'011_1011: x (60 × 2) = x120 ... B'011_1110: x (63 × 2) = x126 B'011_1111: x (64 × 2) = x128 B'100_0000: x (65 × 2) = x130 ... B'100_0111: x (72 × 2) = x144 ... B'100_1101: x (78 × 2) = x156</p> <p>For example, Z2φ = 1200 MHz with EXTAL = 16.66 MHz, STC[6:0] = B'100_0111.</p> $PLL2VCO = 16.66 \text{ MHz} \times (72 \times 2) = 2400 \text{ MHz.}$ $(\text{SYS-CPU divider 2 input}) = 2400 \text{ MHz} \times 1/2 = 1200 \text{ MHz.}$ <p>When SYS-CPU divider 2 setting is x 32/32, Z2φ = 1200 MHz × 32/32 = 1200 MHz.</p> <p>Set PLL2VCO between 2000 MHz and 2400 MHz.</p> |
| 23 to 6 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 5 | REFDIV | *2 | R/W | <p>PLL2 reference clock division ratio</p> <p>0: x 1/1 1: x 1/2</p> <p>Don't change from initial value.</p> |
| 4 to 0 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| Notes: | 1. | Initial value of STC[6:0] is determined by MD14 and MD13 as shown in the Table 11.7 to 11.10. | | |
| | 2. | When MD14 = 1 and MD13 = 1, the initial value is 1. When the other mode setting, the initial value is 0. | | |

11.2.8 PLL3 Control Register (PLL3CR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

PLL3CR is a 32-bit readable/writable register. This register specifies the multiplication ratio of PLL circuit 3. When writing to this register, please change only bits STC[6:0] (Read-modify-write).

| | | | | | | | | | | | | | | | | | |
|----------------|---|----------|-----|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | STC[6:0] | | | | | | — | — | — | — | — | — | — | — | — | — |
| Initial value: | Values of bit 31, 21 are different among products | | | | | | | | | | | | | | | | |
| RZ/G2H | | | | | | | | | | | | | | | | | |
| RZ/G2M V1.3 | 0 | —*1 | —*1 | —*1 | —*1 | —*1 | —*1 | —*1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| RZ/G2M V3.0 | | | | | | | | | | | | | | | | | |
| RZ/G2N | | | | | | | | | | | | | | | | | |
| RZ/G2E | 1 | —*1 | —*1 | —*1 | —*1 | —*1 | —*1 | —*1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | Value of bit 7 is different among products | | | | | | | | | | | | | | | | |
| RZ/G2H | | | | | | | | | | | | | | | | | |
| RZ/G2M V1.3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | —*2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| RZ/G2M V3.0 | | | | | | | | | | | | | | | | | |
| RZ/G2E | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 31 | — | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Reserved These bits are always read as 0. The write value should always be 0. |
| | | B'1 | R | [RZ/G2E] Reserved These bits are always read as 1. The write value should always be 1. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 30 to 24 | STC[6:0] | —*1 | R/W | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>PLL Circuit 3 Multiplication Ratio</p> <p>The PLL3VCO frequency is calculated as following.</p> $(PLL3VCO) = (PLL3 \text{ reference clock}) \times (\text{setting}+1) \times 2$ <p>B'110_1111: x112 B'100_1111: x80 B'011_1111: x64 B'010_1111: x48 B'110_1110: x111 B'100_1110: x79 B'011_1110: x63 B'010_1110: x47 B'110_1101: x110 B'100_1101: x78 B'011_1101: x62 B'010_1101: x46 B'110_1011: x108 B'100_1001: x74 B'011_1011: x60 B'010_1001: x42 B'101_1111: x96 B'100_1000: x73 B'011_1010: x59 B'010_1000: x41 B'101_1110: x95 B'100_0111: x72 B'011_1001: x58 B'010_0111: x40 B'101_1101: x94 B'100_0110: x71 B'011_0111: x56 B'101_1100: x93 B'100_0101: x70 B'011_0110: x55 B'101_1011: x92 B'100_0100: x69 B'011_0101: x54 B'101_1010: x91 B'100_0011: x68 B'011_0100: x53 B'101_0011: x84 B'011_0011: x52 B'101_0010: x83 B'011_0010: x51 B'101_0001: x82</p> <p>Others: setting prohibited</p> <hr/> <p>[RZ/G2E]</p> <p>PLL Circuit 3 Multiplication Ratio</p> <p>The PLL3VCO frequency is calculated as following.</p> $(PLL3VCO) = (PLL3 \text{ reference clock}) / 3 \times (\text{setting}+1)$ <p>In case of MD19 = 0 B'110_0010: = x99 B'110_0011: = x100 others: setting prohibited</p> <p>In case of MD19 = 1 B'011_1001: = x58 others: setting prohibited</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 23 to 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 | — | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Reserved This bit is always read as 0. The write value should always be 0. |
| | | B'1 | R | [RZ/G2E] Reserved This bit is always read as 1. The write value should always be 1. |
| 20 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 19 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | — | —*2 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Reserved This bit is determined by MD14 and MD13. The write value should always be the same value with the read value before writing. |
| | | B'0 | R | [RZ/G2E] Reserved This bit is always read as 0. The write value should always be 0. |
| 6 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

Notes: 1. Initial value is determined by MD19, MD17, MD14 and MD13 as shown in the Table

Table 11.7 [RZ/G2H]

Table 11.8 [RZ/G2M V1.3, RZ/G2M V3.0]

Table 11.9 [RZ/G2N]

Initial value is determined by MD19, MD14 and MD13 as shown in the Table.

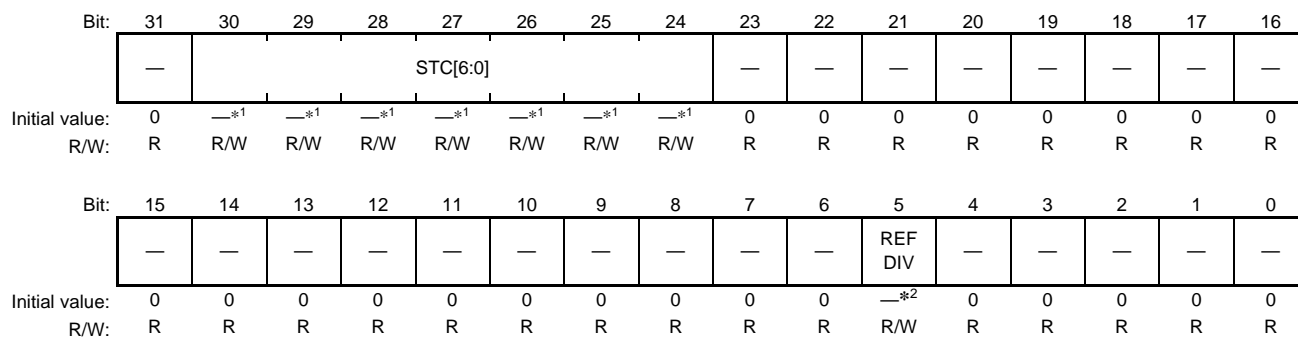
Initial value is determined by MD19 as shown in the Table 11.10. [RZ/G2E]

2. When MD14 = 1 and MD13 = 1, the initial value is 1. When the other mode setting, the initial value is 0.

11.2.9 PLL4 Control Register (PLL4CR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

PLL4CR is a 32-bit readable/writable register. This register specifies the multiplication ratio of PLL circuit 4. When writing to this register, change only bits STC[6:0] (Read-modify-write).



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 30 to 24 | STC[6:0] | *1 | R/W | PLL Circuit 4 Multiplication Ratio*3 PLL circuit 4 performs multiplication with a ratio of (setting + 1) × 2 The frequency has the following relationship. (PLL4VCO) = (PLL2 reference clock) × (setting+1) × 2 (3DGE divider input) = PLL4VCO × 1/2 (fixed divider) ZGφ = (3DGE divider input) × 1/2 (fixed divider) B'010_1111: × (48 × 2) = ×96 (Ex: 25 MHz × 96 = 2400 MHz) ... B'011_1011: × (60 × 2) = ×120 (Ex: 20 MHz × 120 = 2400 MHz) ... B'100_0111: × (72 × 2) = ×144 (Ex: 16.66 MHz × 144 = 2400 MHz) For example, ZGφ = 600 MHz with EXTAL=16.66 MHz, STC[6:0] = B'100_0111. PLL4VCO = 16.66 MHz × (72 × 2) = 2400 MHz. (3DGE divider input) = 2400 MHz × 1/2 = 1200 MHz. ZGφ = 1200 MHz × 1/2 = 600 MHz. Set PLL4VCO at 2400 MHz. |
| 23 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 5 | REFDIV | *2 | R/W | PLL4 reference clock division ratio 0: x 1/1 1: x 1/2 Don't change from initial value. |
| 4 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

- Notes:
1. Initial value of STC[6:0] is determined by MD14 and MD13 as shown in the Table 11.7 to 11.10 .
 2. When MD14 = 1 and MD13 = 1, the initial value is 1. When the other mode setting, the initial value is 0.
 3. Actual multiplication ratio of PLL4 VCO to the reference clock of PLL4 is (setting + 1) × 2. The factor “2” is derived from the fixed divider in the PLL feedback loop. The input clock to the 3DGE divider is generated by dividing PLL4 VCO by 2. That is why the input clock frequency to the 3DGE divider is expressed as (STC[6:0] + 1) × (reference clock of PLL4).

11.2.10 SD-IFn Clock Frequency Control Register (SDnCKCR, n = 0 to 3 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N], n = 0, 1, 3 [RZ/G2E])

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SDnCKCR is a 32-bit readable/writable register. This register controls SDnH clock (SDnHφ) and SDHIn clock (SDnφ) frequency. This register should be set before SD-IFn module is operated. Do not access SD-IFn module during changing the clock frequency of SDnHφ and SDnφ or stopping these clocks.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-------------|------------|----|----|----|---------------|-----|-----|------------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | STPn HCK | STPn CK | — | — | — | SDnSRCFC[2:0] | | | SDnFC[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | STPnHCK | B'1 | R/W | Clock Stop 0: Supplies SDnH clock 1: Stops SDnH clock |
| 8 | STPnCK | B'0 | R/W | Clock Stop 0: Supplies SDn clock 1: Stops SDn clock |
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------|---------------|-----|---|
| 4 to 2 | SDnSRCFC [2:0] | B'010 | R/W | <p>SDnHϕ clock Frequency Division Ratio</p> <p>B'000: SDnHϕ = SDSRCϕ \times 1/1 SDnϕ = SD0SRCϕ \times 1/4 (when SDnFC[1:0] = 01) (e.g. SDnHϕ = 800 MHz, SDnϕ = 200 MHz)</p> <p>B'001: SDnHϕ = SDSRCϕ \times 1/2, SDnϕ = SDSRCϕ \times 1/8 (when SDnFC[1:0] = 01) (e.g. SDnHϕ = 400, SDnϕ = 100MHz) SDnϕ = SDSRCϕ \times 1/4 (when SDnFC[1:0] = 00) (e.g. SDnHϕ = 400, SDnϕ = 200 MHz)</p> <p>B'010: SDnHϕ = SDSRCϕ \times 1/4 (should be stopped by STPnHCK) SDnϕ = SDSRCϕ \times 1/16 (when SDnFC[1:0] = 01) (e.g. SDnHϕ = should be stopped, SDnϕ = 50 MHz)</p> <p>B'011: SDnHϕ = should be stopped (should be stopped by STPnHCK) SDnϕ = SDSRCϕ \times 1/32 (when SDnFC[1:0] = 01) (e.g. SDnHϕ = should be stopped, SDnϕ = 25 MHz)</p> <p>B'100: SDnHϕ = should be stopped (should be stopped by STPnHCK) SDnϕ = SDSRCϕ \times 1/64 (when SDnFC[1:0] = 01) (e.g. SDnHϕ = should be stopped, SDnϕ = 12.5 MHz)</p> <p>B'101: Setting prohibited B'110: Setting prohibited B'111: Setting prohibited</p> |
| 1 to 0 | SDnFC[1:0] | B'01 | R/W | <p>Select SDnϕ division Ratio</p> <p>B'00: SDnϕ = SDnHϕ \times 1/2 B'01: SDnϕ = SDnHϕ \times 1/4 B'10: Reserved B'11: Reserved</p> <p>Example 1 SDnSRCFC[2:0] = 000, SDnFC[1:0] = 01, SDnHϕ = 800 MHz, SDnϕ = 200 MHz</p> <p>Example 2 SDnSRCFC[2:0] = 001, SDnFC[1:0] = 00, SDnHϕ = 400 MHz, SDnϕ = 200 MHz</p> <p>Refer to SDnSRCFC[2:0] explanation.</p> |

11.2.11 RPC-IF Clock Frequency Control Register (RPCCKCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

RPCCKCR is a 32-bit readable/writable register. This register controls the RPC-IF clock (RPC ϕ). This register should be set before RPC-IF module is operated. Do not access RPC-IF module during changing the clock frequency of RPC ϕ .

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|--------|-------|----|----|----|----------|-----|-----|-----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | CKSTP2 | CKSTP | — | — | — | DIV[4:0] | | | | |
| Initial value: | | | | | | | | | | | | | | | | |
| RZ/G2H | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | *2 | *2 | 1 |
| RZ/G2N | | | | | | | | | | | | | | | | |
| RZ/G2M V1.3 | 0 | 0 | 0 | 0 | 0 | 0 | *1 | *1 | 0 | 0 | 0 | 1 | 0 | *2 | *2 | 1 |
| RZ/G2M V3.0 | | | | | | | | | | | | | | | | |
| RZ/G2E | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | *3 | *3 | *3 | *3 | 1 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R |

Notes: 1. Depend on MD4, MD3, MD2, and MD1 values.

When (MD4, MD3, MD2, MD1) = 0001/0010/0011/0100/1010/1011, the initial value of CKSTP&CKSTP2 = 0.

When the other setting of (MD4, MD3, MD2, MD1), the initial value of CKSTP&CKSTP2 = 1.

2. Depend on MD4, MD3, MD2, and MD1 values

When (MD4, MD3, MD2, MD1) = (Don't care, 0, 1, 0), DIV[2:0] = 001, DIV[4:3] = 10.

When (MD4, MD3, MD2, MD1) = (Don't care, 0, 1, 1), DIV[2:0] = 011, DIV[4:3] = 10.

In the other setting of (MD4, MD3, MD2, MD1), DIV[2:0] = 111, DIV[4:3] = 10.

3. Depend on MD4, MD3, MD2, and MD1 values

When (MD4, MD3, MD2, MD1) = (0, 0, 1, 0) or (1, 0, 1, 0): DIV[2:0] = 001, DIV[4:3] = 10 (300 MHz PLL0)

When (MD4, MD3, MD2, MD1) = (0, 0, 1, 1) or (1, 0, 1, 1): DIV[2:0] = 001, DIV[4:3] = 00 (160 MHz PLL1) [RZ/G2E]

When (MD4, MD3, MD2, MD1) = (0, 0, 0, 1) or (0, 1, 0, 0): DIV[2:0] = 011, DIV[4:3] = 00 (300 MHz PLL0)

In the other setting of (MD4, MD3, MD2, MD1), DIV[2:0] = 111, DIV[4:3] = 10 (75 MHz PLL0).

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | CKSTP2 | *1 | R/W | [RZ/G2M V1.3, RZ/G2M V3.0] RPCD2 ϕ Clock Stop 0: Supplies clock to RPC-IF 1: Stops clock to RPC-IF |
| | | B'0 | R/W | [RZ/G2H, RZ/G2N, RZ/G2E] RPCD2 ϕ Clock Stop 0: Supplies clock to RPC-IF 1: Stops clock to RPC-IF |
| 8 | CKSTP | *1 | R/W | [RZ/G2M V1.3, RZ/G2M V3.0] RPC ϕ Clock Stop 0: Supplies clock to RPC-IF 1: Stops clock to RPC-IF |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| | | B'0 | R/W | [RZ/G2H, RZ/G2N, RZ/G2E] RPC ϕ Clock Stop 0: Supplies clock to RPC-IF 1: Stops clock to RPC-IF |
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | DIV[4:0] | *2 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] RPC-IF clock (RPC ϕ , RPCD2 ϕ) Frequency Division Ratio DIV[4:3] B'00: Prohibited setting B'01: Prohibited setting B'10: $RPCSRC\phi = (\text{Clock source} = PLL1VCO) \times 1/5$ B'11: $RPCSRC\phi = (\text{Clock source} = PLL1VCO) \times 1/6$ DIV[2:0] $RPC\phi = RPCSRC\phi / (DIV[2:0] + 1)$ $RPCD2\phi = RPCSRC\phi / \{ (DIV[2:0] + 1) \times 2 \}$ The setting DIV[2:0] = 000 is prohibited. The setting DIV[0] = 0 is also prohibited. Example: Setting: Frequency of RPC ϕ , Frequency of RPCD2 ϕ B'1_0000: Prohibited B'1_0001: $PLL1VCO (3200 \text{ MHz}) \times 1/5 \times 1/2 = 320 \text{ MHz}, 160 \text{ MHz}$ (RPC-IF max.) B'1_0010: Prohibited B'1_0011: $PLL1VCO (3200 \text{ MHz}) \times 1/5 \times 1/4 = 160 \text{ MHz}, 80 \text{ MHz}$ (QSPI max.) B'1_0100: Prohibited B'1_0101: $PLL1VCO (3200 \text{ MHz}) \times 1/5 \times 1/6 = 106 \text{ MHz}, 53 \text{ MHz}$ B'1_0110: Prohibited B'1_0111: $PLL1VCO (3200 \text{ MHz}) \times 1/5 \times 1/8 = 80 \text{ MHz}, 40 \text{ MHz}$ B'1_1000: Prohibited B'1_1001: $PLL1VCO (3200 \text{ MHz}) \times 1/6 \times 1/2 = 266 \text{ MHz}, 133 \text{ MHz}$ B'1_1010: Prohibited B'1_1011: $PLL1VCO (3200 \text{ MHz}) \times 1/6 \times 1/4 = 133 \text{ MHz}, 66 \text{ MHz}$ B'1_1100: Prohibited B'1_1101: $PLL1VCO (3200 \text{ MHz}) \times 1/6 \times 1/6 = 88 \text{ MHz}, 44 \text{ MHz}$ B'1_1110: Prohibited B'1_1111: $PLL1VCO (3200 \text{ MHz}) \times 1/6 \times 1/8 = 66 \text{ MHz}, 33 \text{ MHz}$ |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 4 to 0 | DIV[4:0] | *3 | R/W | <p>[RZ/G2E]</p> <p>DIV[4:3]</p> <p>B'00: $\text{RPCSRC}\phi = (\text{Clock source} = \text{PLL1VCO}) \times 1/5$ (320 MHz)</p> <p>B'01: $\text{RPCSRC}\phi = (\text{Clock source} = \text{PLL1VCO}) \times 1/3$ (533.33 MHz)</p> <p>B'10: $\text{RPCSRC}\phi = (\text{Clock source} = \text{PLL0VCO}) \times 1/n$ (600 MHz) ($n = 5$ $n = 8$ [RZ/G2E])</p> <p>B'11: $\text{RPCSRC}\phi = (\text{Clock source} = \text{PLL1VCO}) \times 1/2$ (800 MHz)</p> <p>DIV[2:0]</p> <p>$\text{RPC}\phi = \text{RPCSRC}\phi / (\text{DIV}[2:0] + 1)$</p> <p>$\text{PRCD}2\phi = \text{RPCSRC}\phi / \{ (\text{DIV}[2:0] + 1) \times 2 \}$</p> <p>The setting DIV[2:0] = 000 is prohibited.</p> <p>The setting DIV[0] = 0 is also prohibited.</p> <p>RPCϕ must be less than 300 MHz.</p> |

11.2.12 CAN-FD Clock Frequency Control Register (CANFDCKCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CANFDCKCR is a 32-bit readable/writable register. This register controls the CANFD clock (CANFD ϕ). This register should be set before CAN-FD module is operated. Do not access CAN-FD module during changing the clock frequency of CANFD ϕ .

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----------|----|----|----------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | CKST P | — | — | DIV[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | CKSTP | B'0 | R/W | Clock Stop 0: Supplies CANFD clock 1: Stops CANFD clock |
| 7, 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | DIV[5:0] | B'11_1 111 | R/W | Division Ratio These bits set the frequency division ratio of CANFD clock. CANFD clock source is divided by the division ratio of 1 / (setting + 1). |

11.2.13 MSIOF Clock Frequency Control Register (MSOCKCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

MSOCKCR is a 32-bit readable/writable register. This register controls the MSIOF clock (MSO ϕ). This register should be set before MSIOF module is operated. Do not access MSIOF module during changing the clock frequency of MSO ϕ .

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----------|----|----|----------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | CKST P | — | — | DIV[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | CKSTP | B'0 | R/W | Clock Stop 0: Supplies MSIOF clock 1: Stops MSIOF clock |
| 7, 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | DIV[5:0] | B'11_1 111 | R/W | Division Ratio These bits set the frequency division ratio of MSIOF clock. MSIOF clock source is divided by the division ratio of 1 / (setting + 1). |

11.2.14 HDMI-IF Clock Frequency Control Register (HDMICKCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

HDMICKCR is a 32-bit readable/writable register. This register controls the HDMI clock (HDMI ϕ). This register should be set before HDMI-IF module is operated. Do not access HDMI-IF module during changing the clock frequency of HDMI ϕ .

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----------|----|----|----------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | CKST P | — | — | DIV[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | CKSTP | B'1 | R/W | Clock Stop 0: Supplies HDMI clock 1: Stops HDMI clock |
| 7, 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | DIV[5:0] | B'01_1 111 | R/W | Division Ratio These bits set the frequency division ratio of HDMI clock. HDMI clock source is divided by the division ratio of 1 / (setting + 1). |

11.2.15 CSI0 Clock Frequency Control Register (CSI0CKCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CSI0CKCR is a 32-bit readable/writable register. This register controls the CSI0 clock (CSI0 ϕ). This register should be set before CSI2 module is operated. Do not access CSI2 module during changing the clock frequency of CSI0 ϕ .

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----------|----|----|----------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | CKST P | — | — | DIV[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | CKSTP | B'1 | R/W | Clock Stop 0: Supplies CSI0 clock 1: Stops CSI0 clock |
| 7, 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | DIV[5:0] | B'01_1 111 | R/W | Division Ratio These bits set the frequency division ratio of CSI0 clock. CSI0 clock source is divided by the division ratio of 1 / (setting + 1). |

11.2.16 RCLK Frequency Control Register (RCKCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

RCKCR is a 32-bit readable/writable register. This register controls the clock source of RCLK. When write to this register, update by read-modify-write.

This register is only initialized by Power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|-------|----|----|------|----|----|----|--------|----|----|----------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CKSEL | — | — | LOCK | — | — | — | CKST P | — | — | DIV[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | —* | —* | —* | —* | —* | —* |
| R/W: | R/W | R | R | R/W | R | R | R | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * H'2F when MD13 = 0, H'12 when MD13 = 1

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | CKSEL | B'0 | R/W | Select clock source of RCLK 0: internal RCLK 1: OCO |
| 14 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | LOCK | B'0 | R/W | CKSEL lock enable. This bit can update only once. 0: Can write to CKSEL, CKSTP and DIV[5:0] and can write 1 to LOCK bit. 1: Can not write to this register and can not write 0 to LOCK bit. |
| 11 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | CKSTP | B'0 | R/W | Clock Stop of RCLK. 0: Supplies RCLK 1: Stops RCLK |
| 7 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 5 to 0 | DIV[5:0] | —* | R/W | Division Ratio These bits set the frequency division ratio of RCLK. Source clock is divided by the division ratio of 1 / (setting + 1) B'10_1111: 1 / 48 Others: setting prohibited |

Note: * When MD13 = 0, initial value is H'2F. When MD13 = 1, initial value is H'12.
 When MD13 = 1, write H'2F immediately to DIV[5:0] after boot. The following is procedure of writing to DIV[5:0].

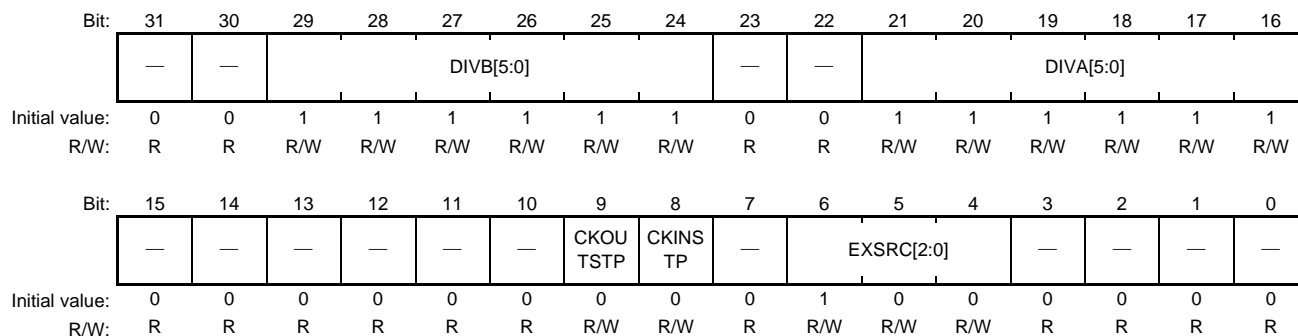
(1) Procedure of writing to DIV[5:0]

1. Stop RCLK by setting CKSTP to 1.
2. Wait 3 cycles by RCLK source.
3. Write H'2F to DIV[5:0] and 0 to CKSTP.

11.2.17 LV0 Clock Control Register (LV0CKCR)

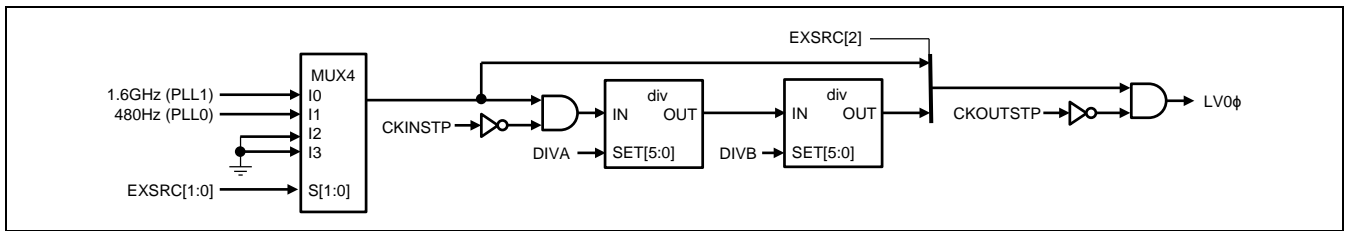
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

LV0CKCR is a 32-bit readable/writable register. Prohibit clock frequency setting over 160MHz.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 24 | DIVB[5:0] | B'11_1 111 | R/W | LV0CLK 2nd divide setting Div = DIVB + 1 |
| 23 to 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 to 16 | DIVA[5:0] | B'11_1 111 | R/W | LV0CLK 1st divide setting Div = DIVA + 1 |
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | CKOUTSTP | B'0 | R/W | LV0CLK OUTClock Stop 0: Supplies LV0CLK output lock 1: Stops LV0CLK output clock |
| 8 | CKINSTP | B'0 | R/W | LV0CLK INClock Stop 0: Supplies LV0CLK source clock 1: Stops LV0CLK source clock |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 to 4 | EXSRC[2:0] | B'100 | R/W | LV0 clock source select B'0xx:prohibit B'100: divide 1.6 GHz base clock (PLL1) B'101: divide 480 MHz base clock (PLL0) B'110: reserved B'111: reserved Prohibit over 160 MHz setting |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

LV0CKCR Logic and setting bit.



11.2.18 LV1 Clock Control Register (LV1CKCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

LV1CKCR is a 32-bit readable/writable register. Prohibit clock frequency setting over 160MHz.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|-----------|-----|-----|-----|--------------|-------------|----|------------|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | DIVB[5:0] | | | | | — | — | DIVA[5:0] | | | | | | |
| Initial value: | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | CKOU TSTP | CKINS TP | — | EXSRC[2:0] | | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R/W | R/W | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 24 | DIVB[5:0] | B'11_1 111 | R/W | LV1CLK 2nd divide setting Div = DIVB + 1 |
| 23 to 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 to 16 | DIVA[5:0] | B'11_1 111 | R/W | LV1CLK 1st divide setting Div = DIVA + 1 |
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | CKOUTSTP | B'0 | R/W | LV1CLK OUTClock Stop 0: Supplies LV1CLK output clock 1: Stops LV1CLK output clock |
| 8 | CKINSTP | B'0 | R/W | LV1CLK INClock Stop 0: Supplies LV1CLK source clock 1: Stops LV1CLK source clock |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 to 4 | EXSRC[2:0] | B'100 | R/W | LV1 clock source select. Prohibit over 160 MHz setting. B'0xx: prohibit B'100: divide 1.6 GHz base clock (PLL1) B'101: divide 480 MHz base clock (PLL0) B'110: reserved B'111: reserved |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

11.2.19 ZA2 Clock Control Register (ZA2CKCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

ZA2CKCR is a 32-bit readable/writable register.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----------|----|----|----|----|----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | CKST P | — | — | — | — | DIV[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | CKSTP | B'0 | R/W | ZA2CLK Clock Stop 0: Supplies ZA2CLK clock 1: Stops ZA2CLK clock |
| 7 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | DIV[3:0] | B'0000 | R/W | ZA2CLK clock divide select B'00xx: 200 MHz (nonSSCG/Clock source is PLL0) Others: Setting prohibited |

Note: *The clock source is PLL1, As ZA2CLK is used for ADG that should be used nonSSCG clock, when select the clock, PLL1 should be used by nonSSCG mode (MD12 = 0).

11.2.20 ZA8 Clock Control Register (ZA8CKCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

ZA8CKCR is a 32-bit readable/writable register.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----------|----|----|----|----|-----|-----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | CKST P | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R/W | R/W | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | CKSTP | B'0 | R/W | ZA8CLK clock Stop 0: Supplies ZA8CLK clock 1: Stops ZA8CLK clock |
| 7 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 2 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

11.2.21 Z2D Clock Control Register (Z2DCKCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

Z2DCKCR is a 32-bit readable/writable register.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----------|----|----|----|----|----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | CKST P | — | — | — | — | DIV[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | CKSTP | B'0 | R/W | Z2D Clock Stop 0: Supplies Z2DCLK clock 1: Stops Z2DCLK clock |
| 7 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | DIV[3:0] | B'1001 | R/W | Z2DCLK clock divide select B'0101: 300 MHz (nonSSCG) B'1001: 400 MHz (SSCG) B'1101: 266.66 MHz (SSCG) others: Reserved |

11.2.22 Frequency Control Register D (FRQCRD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

FRQCRD is a 32-bit readable/writable register. This register specifies DBSC clock frequency (ZB3 ϕ).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | *1 | *1 | *1 | *1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | ZB3FC[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | *1 | *1 | *1 | *1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 to 16 | — | *1 | R/W | Reserved Read value is depended on MD19*1. The write value should always be the same as read value. |
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | ZB3FC[3:0] | *1 | R/W | DBSC clock (ZB3 ϕ) Frequency Division Ratio B'0000: $\times 1/2$ B'0010: $\times 1/4$ B'0011: $\times 1/6$ Other values: Setting prohibited |

Note: 1 When MD19 = 0, initial value is B'0000. When MD19 = 1, initial value is B'0010.

11.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

11.3.1 Changing Frequency

The clock frequencies controlled by the frequency control registers can be changed either by changing the multiplication ratio of PLL circuits n ($n = 0, 2, 3, 4$) * or by changing the division ratio of the dividers. They are controlled by software through the frequency control registers. The methods are described below.

(*RZ/G2N does not support PLL circuit 2. RZ/G2E do not support PLL circuit 2 and 4.)

(1) Changing Multiplication Ratio of PLL circuit 0, 2, 3, 4 *

Changing the multiplication ratio of PLL circuit n ($n = 0, 2, 3, 4$) * can be done by modifying the STC [6:0] bits in PLLnCR ($n = 0, 2, 3, 4$)*. The PLL oscillation settling time is internally detected automatically. If the oscillation of PLL circuits is settled, 1 can be read through PLLnST bit of PLLECR.

(* RZ/G2N do not support PLL circuit 2. RZ/G2E do not support PLL circuit 2 and 4.)

(2) Changing Division Ratio

Changing the frequency of $Z\phi^*$, $Z2\phi^*$, $ZG\phi^*$, $ZT\phi$, $ZTR\phi$, and $ZTRD2\phi^*$ can be done by modifying each set of bits for setting the division ratio in FRQCRB and FRQCRC. After setting new ratio, write 1 to the KICK bit in FRQCRB to start the division ratio change. Before changing the division ratio, the KICK bit must be checked for 0. After 1 is written to the KICK bit in FRQCRB, 0 can be read when the change of new division ratio is completed. While the setting is being changed with the KICK bit being 1, do not modify FRQCRB and FRQCRC. When the KICK bit is read as 1, do not write 0 into the KICK bit.

(*RZ/G2N does not support $Z2\phi$. RZ/G2E does not support $Z\phi$ and $ZTRD2\phi$.)

11.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

11.4.1 Write protect function for CPG registers

(1) Protected registers

All registers in CPG (registers described in the section 11 CPG, 12 CPG_MSTPRST, and 13 CPG_APMU) except for CPGWPR can be protected from unexpected write operation when CPGWPCR.WPE = 1. When CPGWPCR.WPE = 1, CPG registers can't be written without using CPGWPR before the write access to the target register.

(2) How to write to CPG registers when CPGWPCR.WPE = 1

When CPGWPCR.WPE = 1, writing to CPRWPR is required before writing to the target register. The sample procedure is shown below (the case in which you want to write H'AAAA_5555 to the target register).

1. Write H'5555_AAAA (= inverted value of H'AAAA_5555) to CPGWPR
2. Write H'AAAA_5555 to the target register

Then, the target register is written by H'AAAA_5555.

When you skip the first step above, you can't write H'AAAA_5555 to the target register.

(3) How to clear CPGWPCR.WPE

When CPGWPCR.WPE = 1, the bit can be cleared by the following procedure.

0. CPGWPCR.WPE is 1
1. Write H'5A5A_FFFF to CPGWPR
2. Write H'A5A5_0000 to CPGWPCR
3. CPGWPCR.WPE becomes 0

When CPGWPCR.WPE = 0, the write protect function is disabled. Registers can be written directly without using CPGWPR.

11.4.2 Notes on Board Designing

RZ/G2H

RZ/G2M V1.3

RZ/G2M V3.0

RZ/G2N

RZ/G2E

(1) Bypass Capacitors

Insert laminated ceramic capacitors as bypass capacitors for each V_{SS}/V_{CC} pair. Mount the bypass capacitor near the power supply pins of the LSI. Use components with a frequency characteristic suitable for the operating frequency of the LSI, as well as a suitable capacitance value.

(2) Notes on Using a PLL Oscillation Circuit

Keep the wiring from the PLL V_{DD} and V_{SS} connection pattern to the power supply pins short, and make the pattern width large, to minimize the inductance component.

The analog power supply system of the PLL circuits is sensitive to noise. Therefore system malfunction may occur by the intervention with another power supply. Do not supply the analog power supply with the same resource as the digital power supply of V_{DD} and V_{CCQ} .

(3) When Using an External Crystal Resonator

Place the crystal resonator, capacitors CL1 and CL2, and damping resistor R as close to the XTAL, and EXTAL pins as possible. To minimize induction and thus obtain oscillation at the correct frequency, the capacitors to be attached to the resonator must be grounded to the same ground. Do not bring wiring patterns close to these components.

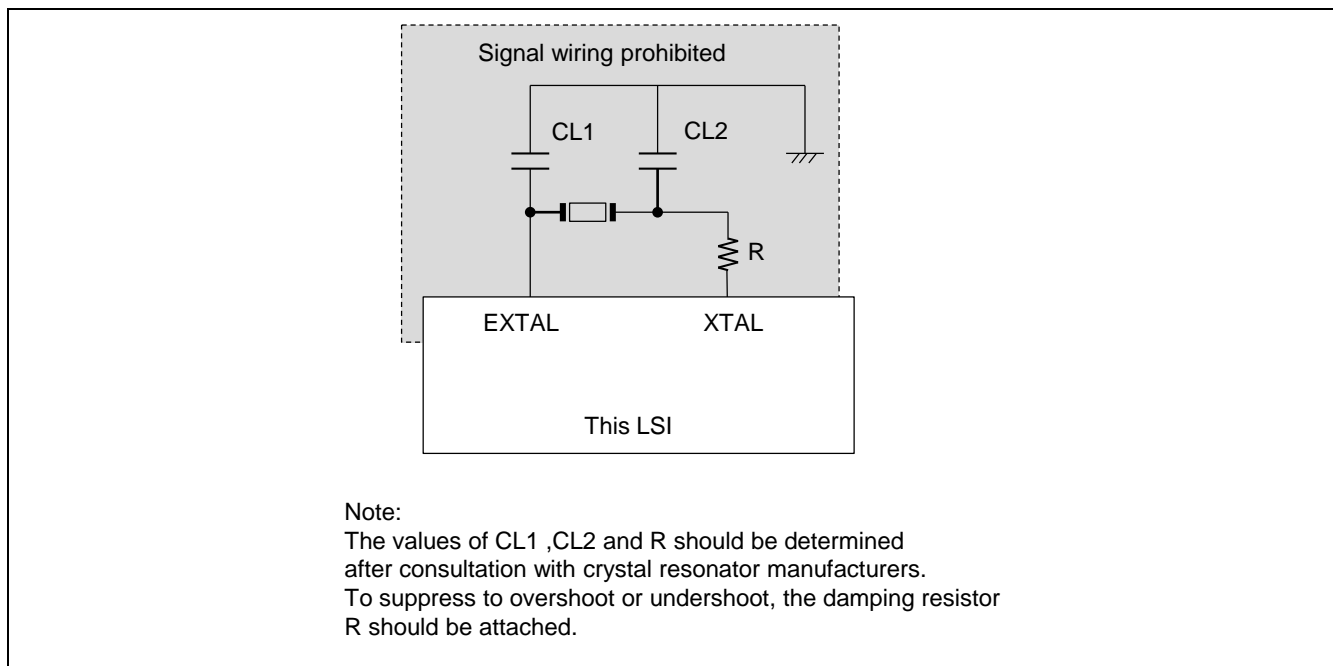


Figure 11.5 Note on Using a Crystal Resonator

(4) When Supplying External Clock from EXTAL Pin

Leave the XTAL pin open.

12. Module Standby, Software Reset

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

12.1 Overview

The CPG functions related to module control are explained in this section. Under software control, the CPG is capable of turning the supply of clock signals to individual modules on or off and of resetting individual modules.

12.1.1 Features

- Module standby
Clock supply to specified modules is stopped by setting the module stop control register bits.
Four registers (MSTPSR, RMSTPCR, SMSTPCR and SCMSTPCR) for one module, and supply of the clock signal to a module is stopped when all these register bits are set to 'stop'.
- Software Reset
Initialize the specified module by setting the software reset register bit.
Secure Module Stop Control function.
Secure Software Reset Access Enable Control function.
Secure Access protection function.

12.1.2 External Pins

There is no input/output pin related with this function.

12.1.3 Register Configuration

Table 12.1 lists of the CPG registers for module standby and software reset. Table 12.2 lists the register states in response to a reset.

Table 12.1 Register Configurations

| | | | | | | Second Generation RZ/G Series Products | | | |
|---|--------------|-----|-------------|---|-------------|---|----------------------------|--------|--------|
| Register Name | Abbreviation | R/W | Address | Initial value | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Module stop status register 1 | MSTPSR1 | R | H'E615_0038 | H'FFFF_FFF | 32 | √ | √ | √ | √ |
| Module stop status register 2 | MSTPSR2 | R | H'E615_0040 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Module stop status register 3 | MSTPSR3 | R | H'E615_0048 | H'FFFF_FFDF | 32 | √ | √ | √ | √ |
| Module stop status register 4 | MSTPSR4 | R | H'E615_004C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Module stop status register 5 | MSTPSR5 | R | H'E615_003C | H'83BF_FFFF | 32 | √ | √ | √ | √ |
| Module stop status register 6 | MSTPSR6 | R | H'E615_01C0 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| Module stop status register 7 | MSTPSR7 | R | H'E615_01C4 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| Module stop status register 8 | MSTPSR8 | R | H'E615_09A0 | H'00F1_FFF7 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E] H'01F1_FFF7 [RZ/G2N] | 32 | √ | √ | √ | √ |
| Module stop status register 9 | MSTPSR9 | R | H'E615_09A4 | H'03F_E017 | 32 | √ | √ | √ | √ |
| Module stop status register 10 | MSTPSR10 | R | H'E615_09A8 | H'FFFF_FFE0 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] H'FFFF_FFE0 [RZ/G2N, RZ/G2E] | 32 | √ | √ | √ | √ |
| Realtime module stop control register 1 | RMSTPCR1 | R/W | H'E615_0114 | H'FFFF_FFF | 32 | √ | √ | √ | √ |
| Realtime module stop control register 2 | RMSTPCR2 | R/W | H'E615_0118 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Realtime module stop control register 3 | RMSTPCR3 | R/W | H'E615_011C | H'FFFF_FFDF | 32 | √ | √ | √ | √ |
| Realtime module stop control register 4 | RMSTPCR4 | R/W | H'E615_0120 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Realtime module stop control register 5 | RMSTPCR5 | R/W | H'E615_0124 | H'83BF_FFFF | 32 | √ | √ | √ | √ |
| Realtime module stop control register 6 | RMSTPCR6 | R/W | H'E615_0128 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| Realtime module stop control register 7 | RMSTPCR7 | R/W | H'E615_012C | H'FFFF_FFFF | 32 | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Register Name | Abbreviation | R/W | Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | |
|--|--------------|-----|-------------|---|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Realtime module stop control register 8 | RMSTPCR8 | R/W | H'E615_0980 | H'00F1_FFF7 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E] H'01F1_FFF7 [RZ/G2N] | 32 | √ | √ | √ | √ |
| Realtime module stop control register 9 | RMSTPCR9 | R/W | H'E615_0984 | H'03F-_E017 | 32 | √ | √ | √ | √ |
| Realtime module stop control register 10 | RMSTPCR10 | R/W | H'E615_0988 | H'FFFE_FFE0 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] H'FFFF_FFE0 [RZ/G2N, RZ/G2E] | 32 | √ | √ | √ | √ |
| System module stop control register 1 | SMSTPCR1 | R/W | H'E615_0134 | H'FFFF_-FFF | 32 | √ | √ | √ | √ |
| System module stop control register 2 | SMSTPCR2 | R/W | H'E615_0138 | H'0000_0000 | 32 | √ | √ | √ | √ |
| System module stop control register 3 | SMSTPCR3 | R/W | H'E615_013C | H'FFFF_FFDF | 32 | √ | √ | √ | √ |
| System module stop control register 4 | SMSTPCR4 | R/W | H'E615_0140 | H'0000_0000 | 32 | √ | √ | √ | √ |
| System module stop control register 5 | SMSTPCR5 | R/W | H'E615_0144 | H'83BF_FFFF | 32 | √ | √ | √ | √ |
| System module stop control register 6 | SMSTPCR6 | R/W | H'E615_0148 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| System module stop control register 7 | SMSTPCR7 | R/W | H'E615_014C | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| System module stop control register 8 | SMSTPCR8 | R/W | H'E615_0990 | H'00F1_FFF7 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2] H'01F1_FFF7 [RZ/G2N] | 32 | √ | √ | √ | √ |
| System module stop control register 9 | SMSTPCR9 | R/W | H'E615_0994 | H'03F-_E017 | 32 | √ | √ | √ | √ |
| System module stop control register 10 | SMSTPCR10 | R/W | H'E615_0998 | H'FFFE_FFE0 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] H'FFFF_FFE0 [RZ/G2N, RZ/G2E] | 32 | √ | √ | √ | √ |
| Software reset register 1 | SRCR1 | R/W | H'E615_00A8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Software reset register 2 | SRCR2 | R/W | H'E615_00B0 | H'00-0_0000 | 32 | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Register Name | Abbreviation | R/W | Address | Initial value | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--|--------------|-----|-------------|---------------|-------------|--------|----------------------------|--------|--------|
| Software reset register 3 | SRCR3 | R/W | H'E615_00B8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Software reset register 4 | SRCR4 | R/W | H'E615_00BC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Software reset register 5 | SRCR5 | R/W | H'E615_00C4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Software reset register 6 | SRCR6 | R/W | H'E615_01C8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Software reset register 7 | SRCR7 | R/W | H'E615_01CC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Software reset register 8 | SRCR8 | R/W | H'E615_0920 | H'-000_0000 | 32 | √ | √ | √ | √ |
| Software reset register 9 | SRCR9 | R/W | H'E615_0924 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Software reset register 10 | SRCR10 | R/W | H'E615_0928 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Software reset clearing register 0 | SRSTCLR0 | W | H'E615_0940 | — | 32 | √ | √ | √ | √ |
| Software reset clearing register 1 | SRSTCLR1 | W | H'E615_0944 | — | 32 | √ | √ | √ | √ |
| Software reset clearing register 2 | SRSTCLR2 | W | H'E615_0948 | — | 32 | √ | √ | √ | √ |
| Software reset clearing register 3 | SRSTCLR3 | W | H'E615_094C | — | 32 | √ | √ | √ | √ |
| Software reset clearing register 4 | SRSTCLR4 | W | H'E615_0950 | — | 32 | √ | √ | √ | √ |
| Software reset clearing register 5 | SRSTCLR5 | W | H'E615_0954 | — | 32 | √ | √ | √ | √ |
| Software reset clearing register 6 | SRSTCLR6 | W | H'E615_0958 | — | 32 | √ | √ | √ | √ |
| Software reset clearing register 7 | SRSTCLR7 | W | H'E615_095C | — | 32 | √ | √ | √ | √ |
| Software reset clearing register 8 | SRSTCLR8 | W | H'E615_0960 | — | 32 | √ | √ | √ | √ |
| Software reset clearing register 9 | SRSTCLR9 | W | H'E615_0964 | — | 32 | √ | √ | √ | √ |
| Software reset clearing register 10 | SRSTCLR10 | W | H'E615_0968 | — | 32 | √ | √ | √ | √ |
| Software reset clearing register 11 | SRSTCLR11 | W | H'E615_096C | — | 32 | √ | √ | √ | √ |
| Secure Module Stop Control Register 0 | SCMSTPCR0 | R/W | H'E615_0B20 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| Secure Module Stop Control Register 1 | SCMSTPCR1 | R/W | H'E615_0B24 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| Secure Module Stop Control Register 2 | SCMSTPCR2 | R/W | H'E615_0B28 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| Secure Module Stop Control Register 3 | SCMSTPCR3 | R/W | H'E615_0B2C | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| Secure Module Stop Control Register 4 | SCMSTPCR4 | R/W | H'E615_0B30 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| Secure Module Stop Control Register 5 | SCMSTPCR5 | R/W | H'E615_0B34 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| Secure Module Stop Control Register 6 | SCMSTPCR6 | R/W | H'E615_0B38 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| Secure Module Stop Control Register 7 | SCMSTPCR7 | R/W | H'E615_0B3C | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| Secure Module Stop Control Register 8 | SCMSTPCR8 | R/W | H'E615_0B40 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| Secure Module Stop Control Register 9 | SCMSTPCR9 | R/W | H'E615_0B44 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| Secure Module Stop Control Register 10 | SCMSTPCR10 | R/W | H'E615_0B48 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |

| Register Name | Abbreviation | R/W | Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Secure Module Stop Control Register 11 | SCMSTPCR1 1 | R/W | H'E615_0B4C | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| Secure Software Reset Access Enable Control Register 0 | SCSRSTECR 0 | R/W | H'E615_0B80 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Secure Software Reset Access Enable Control Register 1 | SCSRSTECR 1 | R/W | H'E615_0B84 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Secure Software Reset Access Enable Control Register 2 | SCSRSTECR 2 | R/W | H'E615_0B88 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Secure Software Reset Access Enable Control Register 3 | SCSRSTECR 3 | R/W | H'E615_0B8C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Secure Software Reset Access Enable Control Register 4 | SCSRSTECR 4 | R/W | H'E615_0B90 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Secure Software Reset Access Enable Control Register 5 | SCSRSTECR 5 | R/W | H'E615_0B94 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Secure Software Reset Access Enable Control Register 6 | SCSRSTECR 6 | R/W | H'E615_0B98 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Secure Software Reset Access Enable Control Register 7 | SCSRSTECR 7 | R/W | H'E615_0B9C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Secure Software Reset Access Enable Control Register 8 | SCSRSTECR 8 | R/W | H'E615_0BA0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Secure Software Reset Access Enable Control Register 9 | SCSRSTECR 9 | R/W | H'E615_0BA4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Secure Software Reset Access Enable Control Register 10 | SCSRSTECR 10 | R/W | H'E615_0BA8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Secure Software Reset Access Enable Control Register 11 | SCSRSTECR 11 | R/W | H'E615_0BAC | H'0000_0000 | 32 | √ | √ | √ | √ |

Note: Do not write to any addresses other than listed above. Operations cannot be guaranteed. Values read from addresses other than those listed above are undefined.

Table 12.2 Register States in Response to a Reset

| Register | Power-On Reset/WDT Reset |
|---------------|--------------------------|
| All Registers | Initialized |

12.2 Register Description

Legend for Register Description

Initial value : Register value after a reset

— : Undefined value

R/W : Readable/writable. The written value can be read.

R : Read-only.

W : Write-only. The read value is undefined.

12.2.1 Module Stop Status Register (MSTPSRn (n = 1 to 10))

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

MSTPSRn is a 32-bit readable register that indicates whether the on-chip modules are in the module standby state.

Setting a bit in this register to 1 stops supply of the clock signal to the corresponding module and the setting a bit to 0 enables clock supply to the corresponding module.

Bit assignment and initial values are shown. The tables from 12.3 to 12.12 show the assignment of modules to bits.

12.2.1.1 Module Stop Status Register 1 (MSTPSR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------------|---------------|----|---------------|----|----|---------------|---------------|---------------|---------------|---------------|----|---------------|---------------|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MSTP ST131 | MSTP ST130 | — | MSTP ST128 | — | — | MSTP ST125 | MSTP ST124 | MSTP ST123 | MSTP ST122 | MSTP ST121 | — | MSTP ST119 | MSTP ST118 | — | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | MSTP ST112 | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 1 | 1 | 1 | —* | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

Note: * When debug mode, initial value is 0, in other case is 1.

0: Supply of the clock signal to the corresponding module is enabled.

1: Supply of the clock signal to the corresponding module is stopped.

Table 12.3 Assignment of Modules to Bits in MSTPSR1

| Bit | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----|----------------------------|-----------------|----------------------------|----------------------------|----------------------------|
| 31 | VCP4 (VDPB) | VCP4 (VDPB) | VCP4 (VDPB) | VCP4 (VDPB) | VCP4 (VDPB) |
| 30 | VCP4 (VCPLF), iVDP1C | VCP4 (VCPLF) | VCP4 (VCPLF), iVDP1C | VCP4 (VCPLF), iVDP1C | VCP4 (VCPLF), iVDP1C |
| 29 | — | — | — | — | — |
| 28 | — | iVDP1C | — | — | — |
| 27 | — | — | — | — | — |
| 26 | — | — | — | — | — |
| 25 | TMU0 | TMU0 | TMU0 | TMU0 | TMU0 |
| 24 | TMU1 | TMU1 | TMU1 | TMU1 | TMU1 |
| 23 | TMU2 | TMU2 | TMU2 | TMU2 | TMU2 |
| 22 | TMU3 | TMU3 | TMU3 | TMU3 | TMU3 |
| 21 | TMU4 | TMU4 | TMU4 | TMU4 | TMU4 |
| 20 | — | — | — | — | — |
| 19 | FDP1-0 | FDP1-0 | FDP1-0 | FDP1-0 | FDP1-0 |
| 18 | FDP1-1 | — | — | — | — |
| 17 | — | — | — | — | — |
| 16 | — | — | — | — | — |
| 15 | — | — | — | — | — |
| 14 | — | — | — | — | — |
| 13 | — | — | — | — | — |
| 12 | 3DGE | 3DGE | 3DGE | 3DGE | 3DGE |
| 11 | — | — | — | — | — |
| 10 | — | — | — | — | — |
| 9 | — | — | — | — | — |
| 8 | — | — | — | — | — |

| Bit | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----|--------|-------------|-------------|--------|--------|
| 7 | — | — | — | — | — |
| 6 | — | — | — | — | — |
| 5 | — | — | — | — | — |
| 4 | — | — | — | — | — |
| 3 | — | — | — | — | — |
| 2 | — | — | — | — | — |
| 1 | — | — | — | — | — |
| 0 | — | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.1.2 Module Stop Status Register 2 (MSTPSR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|------------|------------|------------|------------|------------|------------|----|------------|------------|------------|------------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | MSTP ST219 | MSTP ST218 | MSTP ST217 | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | MSTP ST211 | MSTP ST210 | MSTP ST209 | MSTP ST208 | MSTP ST207 | MSTP ST206 | — | MSTP ST204 | MSTP ST203 | MSTP ST202 | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

Note: * The initial value of bit 23 varies depending on the products.
 0: Supply of the clock signal to the corresponding module is enabled.
 1: Supply of the clock signal to the corresponding module is stopped.

Table 12.4 Assignment of Modules to Bits in MSTPSR2

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|-----------|-----------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | — | — | — | — |
| 30 | — | — | — | — |
| 29 | — | — | — | — |
| 28 | — | — | — | — |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | — | — | — | — |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | SYS-DMAC0 | SYS-DMAC0 | SYS-DMAC0 | SYS-DMAC0 |
| 18 | SYS-DMAC1 | SYS-DMAC1 | SYS-DMAC1 | SYS-DMAC1 |
| 17 | SYS-DMAC2 | SYS-DMAC2 | SYS-DMAC2 | SYS-DMAC2 |
| 16 | — | — | — | — |
| 15 | — | — | — | — |
| 14 | — | — | — | — |
| 13 | — | — | — | — |
| 12 | — | — | — | — |
| 11 | MSIOF0 | MSIOF0 | MSIOF0 | MSIOF0 |
| 10 | MSIOF1 | MSIOF1 | MSIOF1 | MSIOF1 |
| 9 | MSIOF2 | MSIOF2 | MSIOF2 | MSIOF2 |
| 8 | MSIOF3 | MSIOF3 | MSIOF3 | MSIOF3 |
| 7 | SCIF0 | SCIF0 | SCIF0 | SCIF0 |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|--------|--------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 6 | SCIF1 | SCIF1 | SCIF1 | SCIF1 |
| 5 | — | — | — | — |
| 4 | SCIF3 | SCIF3 | SCIF3 | SCIF3 |
| 3 | SCIF4 | SCIF4 | SCIF4 | SCIF4 |
| 2 | SCIF5 | SCIF5 | SCIF5 | SCIF5 |
| 1 | — | — | — | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.1.3 Module Stop Status Register 3 (MSTPSR3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|----|----|----|----|----|------------|------------|------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MSTP ST331 | MSTP ST330 | — | MSTP ST328 | — | — | — | — | — | — | — | — | MSTP ST319 | MSTP ST318 | — | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | MSTP ST314 | MSTP ST313 | MSTP ST312 | MSTP ST311 | MSTP ST310 | — | — | — | — | — | MSTP ST304 | MSTP ST303 | MSTP ST302 | MSTP ST301 | MSTP ST300 |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

0: Supply of the clock signal to the corresponding module is enabled.

1: Supply of the clock signal to the corresponding module is stopped.

Table 12.5 Assignment of Modules to Bits in MSTPSR3

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|-------------|-------------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | USB-DMAC0-1 | USB-DMAC1 | USB-DMAC0-1 | USB-DMAC0-1 |
| 30 | USB-DMAC0-0 | USB-DMAC0 | USB-DMAC0-0 | USB-DMAC0-0 |
| 29 | — | — | — | — |
| 28 | USB3.0-IF0 | USB3.0-IF0 | USB3.0-IF0 | USB3.0-IF0 |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | — | — | — | — |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | PCIEC0 | PCIEC0 | PCIEC0 | PCIEC0 |
| 18 | PCIEC1 | PCIEC1 | PCIEC1 | — |
| 17 | — | — | — | — |
| 16 | — | — | — | — |
| 15 | — | — | — | — |
| 14 | SD-IF0 | SD-IF0 | SD-IF0 | SD-IF0 |
| 13 | SD-IF1 | SD-IF1 | SD-IF1 | SD-IF1 |
| 12 | SD-IF2 | SD-IF2 | SD-IF2 | — |
| 11 | SD-IF3 | SD-IF3 | SD-IF3 | SD-IF3 |
| 10 | SCIF2 | SCIF2 | SCIF2 | SCIF2 |
| 9 | — | — | — | — |
| 8 | — | — | — | — |
| 7 | — | — | — | — |
| 6 | — | — | — | — |
| 5 | — | — | — | — |

| Bit | RZ/G2H | RZ/G2M V1.3 | | |
|-----|--------|-------------|--------|--------|
| | | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 4 | TPU0 | TPU0 | TPU0 | TPU0 |
| 3 | CMT0 | CMT0 | CMT0 | CMT0 |
| 2 | CMT1 | CMT1 | CMT1 | CMT1 |
| 1 | CMT2 | CMT2 | CMT2 | CMT2 |
| 0 | CMT3 | CMT3 | CMT3 | CMT3 |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.1.4 Module Stop Status Register 4 (MSTPSR4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|----|----|----|----|----|----|----|----|----|----|----|----|------------|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MSTP ST431 | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | MSTP ST402 | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

0: Supply of the clock signal to the corresponding module is enabled.
 1: Supply of the clock signal to the corresponding module is stopped.

Table 12.6 Assignment of Modules to Bits in MSTPSR4

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|--------|--------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | SUCMT | SUCMT | SUCMT | SUCMT |
| 30 | — | — | — | — |
| 29 | — | — | — | — |
| 28 | — | — | — | — |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | — | — | — | — |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | — | — | — | — |
| 18 | — | — | — | — |
| 17 | — | — | — | — |
| 16 | — | — | — | — |
| 15 | — | — | — | — |
| 14 | — | — | — | — |
| 13 | — | — | — | — |
| 12 | — | — | — | — |
| 11 | — | — | — | — |
| 10 | — | — | — | — |
| 9 | — | — | — | — |
| 8 | — | — | — | — |
| 7 | — | — | — | — |
| 6 | — | — | — | — |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|--------|--------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 5 | — | — | — | — |
| 4 | — | — | — | — |
| 3 | — | — | — | — |
| 2 | RWDT | RWDT | RWDT | RWDT |
| 1 | — | — | — | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.1.5 Module Stop Status Register 5 (MSTPSR5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---|----|----|----|----|----|----|----|------------|------------|----|------------|------------|------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | MSTP ST523 | MSTP ST522 | — | MSTP ST520 | MSTP ST519 | MSTP ST518 | MSTP ST517 | MSTP ST516 |
| Initial value: | Values of bit 31, 25, 24, 21 are different among products | | | | | | | | | | | | | | | |
| | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | MSTP ST502 | MSTP ST501 |
| Initial value: | Value of bit 0 is different among products | | | | | | | | | | | | | | | |
| | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

0: Supply of the clock signal to the corresponding module is enabled.
 1: Supply of the clock signal to the corresponding module is stopped.

Table 12.7 Assignment of Modules to Bits in MSTPSR5

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|---------|---------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | — | — | — | — |
| 30 | — | — | — | — |
| 29 | — | — | — | — |
| 28 | — | — | — | — |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | PWM | PWM | PWM | PWM |
| 22 | THS/TSC | THS/TSC | THS/TSC | THS/TSC |
| 21 | — | — | — | — |
| 20 | HSCIF0 | HSCIF0 | HSCIF0 | HSCIF0 |
| 19 | HSCIF1 | HSCIF1 | HSCIF1 | HSCIF1 |
| 18 | HSCIF2 | HSCIF2 | HSCIF2 | HSCIF2 |
| 17 | HSCIF3 | HSCIF3 | HSCIF3 | HSCIF3 |
| 16 | HSCIF4 | HSCIF4 | HSCIF4 | HSCIF4 |
| 15 | — | — | — | — |
| 14 | — | — | — | — |
| 13 | — | — | — | — |
| 12 | — | — | — | — |
| 11 | — | — | — | — |
| 10 | — | — | — | — |
| 9 | — | — | — | — |
| 8 | — | — | — | — |
| 7 | — | — | — | — |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|-------------|-------------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 6 | — | — | — | — |
| 5 | — | — | — | — |
| 4 | — | — | — | — |
| 3 | — | — | — | — |
| 2 | Audio-DMAC0 | Audio-DMAC0 | Audio-DMAC0 | Audio-DMAC0 |
| 1 | Audio-DMAC1 | Audio-DMAC1 | Audio-DMAC1 | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.1.6 Module Stop Status Register 6 (MSTPSR6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------------|---------------|----|----|---------------|---------------|----|---------------|---------------|---------------|---------------|----|---------------|---------------|---------------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MSTP ST631 | MSTP ST630 | — | — | — | MSTP ST626 | — | MSTP ST624 | MSTP ST623 | MSTP ST622 | MSTP ST621 | — | MSTP ST619 | — | MSTP ST617 | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MSTP ST615 | MSTP ST614 | — | — | MSTP ST611 | MSTP ST610 | — | — | MSTP ST607 | MSTP ST606 | — | — | MSTP ST603 | MSTP ST602 | MSTP ST601 | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

0: Supply of the clock signal to the corresponding module is enabled.

1: Supply of the clock signal to the corresponding module is stopped.

Table 12.8 Assignment of Modules to Bits in MSTPSR6

| Bit | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----|-------------|-------------|-------------|-------------|-------------|
| 31 | VSP (VSPI0) | VSP (VSPI0) | VSP (VSPI0) | VSP (VSPI0) | VSP (VSPI0) |
| 30 | VSP (VSPI1) | — | — | — | — |
| 29 | — | — | — | — | — |
| 28 | — | — | — | — | — |
| 27 | — | — | — | — | — |
| 26 | VSP (VSPBD) | VSP (VSPB) | VSP (VSPB) | VSP (VSPB) | VSP (VSPB) |
| 25 | — | — | — | — | — |
| 24 | VSP (VSPBC) | — | — | — | — |
| 23 | VSP (VSPD0) | VSP (VSPD0) | VSP (VSPD0) | VSP (VSPD0) | VSP (VSPD0) |
| 22 | VSP (VSPD1) | VSP (VSPD1) | VSP (VSPD1) | VSP (VSPD1) | VSP (VSPD1) |
| 21 | — | VSP (VSPD2) | VSP (VSPD2) | — | — |
| 20 | — | — | — | — | — |
| 19 | FCPCS | FCPCS | FCPCS | FCPCS | FCPCS |
| 18 | — | — | — | — | — |
| 17 | — | FCPCI | — | — | — |
| 16 | — | — | — | — | — |
| 15 | FCPF0 | FCPF0 | FCPF0 | FCPF0 | FCPF0 |
| 14 | FCPF1 | — | — | — | — |
| 13 | — | — | — | — | — |
| 12 | — | — | — | — | — |
| 11 | FCPVI0 | FCPVI0 | FCPVI0 | FCPVI0 | FCPVI0 |
| 10 | FCPVI1 | — | — | — | — |
| 9 | — | — | — | — | — |
| 8 | — | — | — | — | — |
| 7 | FCPVB0 | FCPVB0 | FCPVB0 | FCPVB0 | FCPVB0 |
| 6 | FCPVB1 | — | — | — | — |

| Bit | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----|--------|-------------|-------------|--------|--------|
| 5 | — | — | — | — | — |
| 4 | — | — | — | — | — |
| 3 | FCPVD0 | FCPVD0 | FCPVD0 | FCPVD0 | FCPVD0 |
| 2 | FCPVD1 | FCPVD1 | FCPVD1 | FCPVD1 | FCPVD1 |
| 1 | — | FCPVD2 | FCPVD2 | — | — |
| 0 | — | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.1.7 Module Stop Status Register 7 (MSTPSR7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|------------|------------|----|------------|----|----|------------|------------|------------|------------|------------|------------|------------|----|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | MSTP ST729 | — | MSTP ST727 | — | — | MSTP ST724 | MSTP ST723 | MSTP ST722 | MSTP ST721 | — | — | — | — | MSTP ST716 |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | MSTP ST714 | — | — | — | — | — | — | — | — | — | MSTP ST704 | MSTP ST703 | MSTP ST702 | — | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

0: Supply of the clock signal to the corresponding module is enabled.
 1: Supply of the clock signal to the corresponding module is stopped.

Table 12.9 Assignment of Modules to Bits in MSTPSR7

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|------------|------------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | — | — | — | — |
| 30 | — | — | — | — |
| 29 | HDMI-IF0 | HDMI-IF0 | HDMI-IF0 | — |
| 28 | — | — | — | — |
| 27 | LVDS-IF | LVDS-IF | LVDS-IF | LVDS-IF |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | DU0 | DU0 | DU0 | DU0 |
| 23 | DU1 | DU1 | DU1 | DU1 |
| 22 | — | DU2 | — | — |
| 21 | DU3 | — | DU3 | — |
| 20 | — | — | — | — |
| 19 | — | — | — | — |
| 18 | — | — | — | — |
| 17 | — | — | — | — |
| 16 | CSI40 | CSI40 | CSI40 | CSI40 |
| 15 | — | — | — | — |
| 14 | CSI20 | CSI20 | CSI20 | — |
| 13 | — | — | — | — |
| 12 | — | — | — | — |
| 11 | — | — | — | — |
| 10 | — | — | — | — |
| 9 | — | — | — | — |
| 8 | — | — | — | — |
| 7 | — | — | — | — |
| 6 | — | — | — | — |
| 5 | — | — | — | — |
| 4 | HS-USB-IF0 | HS-USB-IF0 | HS-USB-IF0 | HS-USB-IF0 |

| Bit | RZ/G2H | RZ/G2M V1.3 | | |
|-----|------------|-------------|------------|------------|
| | | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 3 | EHCI/OHCI0 | EHCI/OHCI0 | EHCI/OHCI0 | EHCI/OHCI0 |
| 2 | EHCI/OHCI1 | EHCI/OHCI1 | EHCI/OHCI1 | — |
| 1 | — | — | — | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.1.8 Module Stop Status Register 8 (MSTPSR8)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |

Initial value: Values of bit 24 is different among products

| | | | | | | | | | | | | | | | | |
|-------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RZ/G2H | | | | | | | | | | | | | | | | |
| RZ/G2M V1.3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |
| RZ/G2M V3.0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |
| RZ/G2E | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |
| RZ/G2N | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| | | | | | | | | | | | | | | | | |
|------|------------|----|----|------------|------------|------------|------------|------------|------------|------------|------------|------------|---|---|---|------------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MSTP ST815 | — | — | MSTP ST812 | MSTP ST811 | MSTP ST810 | MSTP ST809 | MSTP ST808 | MSTP ST807 | MSTP ST806 | MSTP ST805 | MSTP ST804 | — | — | — | MSTP ST800 |

| | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Initial value: | | | | | | | | | | | | | | | | |
| RZ/G2H | | | | | | | | | | | | | | | | |
| RZ/G2M V1.3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| RZ/G2M V3.0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| RZ/G2E | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| RZ/G2N | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

Note:

- 0: Supply of the clock signal to the corresponding module is enabled.
- 1: Supply of the clock signal to the corresponding module is stopped.

Table 12.10 Assignment of Modules to Bits in MSTPSR8

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|---------|--------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | — | — | — | — |
| 30 | — | — | — | — |
| 29 | — | — | — | — |
| 28 | — | — | — | — |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | — | — | — | — |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | — | — | — | — |
| 18 | — | — | — | — |
| 17 | — | — | — | — |
| 16 | — | — | — | — |
| 15 | SATA-IF | — | SATA-IF | — |
| 14 | — | — | — | — |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|---------|---------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 13 | — | — | — | — |
| 12 | EAVB-IF | EAVB-IF | EAVB-IF | EAVB-IF |
| 11 | VIN0 | VIN0 | VIN0 | — |
| 10 | VIN1 | VIN1 | VIN1 | — |
| 9 | VIN2 | VIN2 | VIN2 | — |
| 8 | VIN3 | VIN3 | VIN3 | — |
| 7 | VIN4 | VIN4 | VIN4 | VIN4 |
| 6 | VIN5 | VIN5 | VIN5 | VIN5 |
| 5 | VIN6 | VIN6 | VIN6 | — |
| 4 | VIN7 | VIN7 | VIN7 | — |
| 3 | — | — | — | — |
| 2 | — | — | — | — |
| 1 | — | — | — | — |
| 0 | — | R-NANDC | R-NANDC | R-NANDC |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.1.9 Module Stop Status Register 9 (MSTPSR9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|----|------------|------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MSTP ST931 | MSTP ST930 | MSTP ST929 | MSTP ST928 | MSTP ST927 | MSTPS T926 | — | — | — | MSTP ST922 | — | — | MSTP ST919 | MSTP ST918 | MSTP ST917 | MSTP ST916 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | * | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MSTP ST915 | MSTP ST914 | — | MSTP ST912 | MSTP ST911 | MSTP ST910 | MSTP ST909 | MSTP ST908 | MSTP ST907 | MSTP ST906 | MSTP ST905 | — | — | — | — | — |
| Initial value: | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

Note: * [RZ/G2H, RZ/G2N, RZ/G2E]
 Initial value is 0.
 [RZ/G2M V1.3, RZ/G2M V3.0]
 The initial value of bit 17 depends on the mode pins MD4, MD3, MD2 and MD1.
 When (MD4, MD3, MD2, MD1) = 0001/0010/0011/0100/1010/1011, initial value is 0
 , other cases, initial value is 1.

0: Supply of the clock signal to the corresponding module is enabled.
 1: Supply of the clock signal to the corresponding module is stopped.

Table 12.11 Assignment of Modules to Bits in MSTPSR9

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|----------|----------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | I2C-IF0 | I2C-IF0 | I2C-IF0 | I2C-IF0 |
| 30 | I2C-IF1 | I2C-IF1 | I2C-IF1 | I2C-IF1 |
| 29 | I2C-IF2 | I2C-IF2 | I2C-IF2 | I2C-IF2 |
| 28 | I2C-IF3 | I2C-IF3 | I2C-IF3 | I2C-IF3 |
| 27 | I2C-IF4 | I2C-IF4 | I2C-IF4 | I2C-IF4 |
| 26 | IIC-PMIC | IIC-PMIC | IIC-PMIC | IIC-PMIC |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | ADG | ADG | ADG | ADG |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | I2C-IF5 | I2C-IF5 | I2C-IF5 | I2C-IF5 |
| 18 | I2C-IF6 | I2C-IF6 | I2C-IF6 | I2C-IF6 |
| 17 | RPC-IF | RPC-IF | RPC-IF | RPC-IF |
| 16 | CAN-IF0 | CAN-IF0 | CAN-IF0 | CAN-IF0 |
| 15 | CAN-IF1 | CAN-IF1 | CAN-IF1 | CAN-IF1 |
| 14 | CAN-FD | CAN-FD | CAN-FD | CAN-FD |
| 13 | — | — | — | — |
| 12 | GPIO0 | GPIO0 | GPIO0 | GPIO0 |
| 11 | GPIO1 | GPIO1 | GPIO1 | GPIO1 |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|--------|--------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 10 | GPIO2 | GPIO2 | GPIO2 | GPIO2 |
| 9 | GPIO3 | GPIO3 | GPIO3 | GPIO3 |
| 8 | GPIO4 | GPIO4 | GPIO4 | GPIO4 |
| 7 | GPIO5 | GPIO5 | GPIO5 | GPIO5 |
| 6 | GPIO6 | GPIO6 | GPIO6 | GPIO6 |
| 5 | GPIO7 | GPIO7 | GPIO7 | — |
| 4 | — | — | — | — |
| 3 | — | — | — | — |
| 2 | — | — | — | — |
| 1 | — | — | — | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.1.10 Module Stop Status Register 10 (MSTPSR10)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MSTP ST1031 | MSTP ST1030 | MSTP ST1029 | MSTP ST1028 | MSTP ST1027 | MSTP ST1026 | MSTP ST1025 | MSTP ST1024 | MSTP ST1023 | MSTP ST1022 | MSTP ST1021 | MSTP ST1020 | MSTP ST1019 | MSTP ST1018 | MSTP ST1017 | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | *1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MSTP ST1015 | MSTP ST1014 | MSTP ST1013 | MSTP ST1012 | MSTP ST1011 | MSTP ST1010 | MSTP ST1009 | MSTP ST1008 | MSTP ST1007 | MSTP ST1006 | MSTP ST1005 | — | MSTPS T1003 | — | — | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | *2 | *2 | *2 | *2 | *2 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

- Notes: 1. The initial value of bit 16 varies depending on the products.
[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]: The initial value is 0.
[RZ/G2N, RZ/G2E]: The initial value is 1.
2. The initial values of bits 4 to 0 vary depending on the products.
The initial value is 0.
0: Supply of the clock signal to the corresponding module is enabled.
1: Supply of the clock signal to the corresponding module is stopped.

Table 12.12 Assignment of Modules to Bits in MSTPSR10

| Bit | RZ/G2M V1.3 | | | |
|-----|---|---|---|---|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | SCU (SRC0) | SCU (SRC0) | SCU (SRC0) | SCU (SRC0) |
| 30 | SCU (SRC1) | SCU (SRC1) | SCU (SRC1) | SCU (SRC1) |
| 29 | SCU (SRC2) | SCU (SRC2) | SCU (SRC2) | SCU (SRC2) |
| 28 | SCU (SRC3) | SCU (SRC3) | SCU (SRC3) | SCU (SRC3) |
| 27 | SCU (SRC4) | SCU (SRC4) | SCU (SRC4) | SCU (SRC4) |
| 26 | SCU (SRC5) | SCU (SRC5) | SCU (SRC5) | SCU (SRC5) |
| 25 | SCU (SRC6) | SCU (SRC6) | SCU (SRC6) | SCU (SRC6) |
| 24 | SCU (SRC7) | SCU (SRC7) | SCU (SRC7) | SCU (SRC7) |
| 23 | SCU (SRC8) | SCU (SRC8) | SCU (SRC8) | SCU (SRC8) |
| 22 | SCU (SRC9) | SCU (SRC9) | SCU (SRC9) | SCU (SRC9) |
| 21 | SCU (CTU00, CTU01, CTU02, CTU03, MIX0) | SCU (CTU00, CTU01, CTU02, CTU03, MIX0) | SCU (CTU00, CTU01, CTU02, CTU03, MIX0) | SCU (CTU00, CTU01, CTU02, CTU03, MIX0) |
| 20 | SCU (CTU10, CTU11, CTU12, CTU13, MIX1) | SCU (CTU10, CTU11, CTU12, CTU13, MIX1) | SCU (CTU10, CTU11, CTU12, CTU13, MIX1) | SCU (CTU10, CTU11, CTU12, CTU13, MIX1) |
| 19 | SCU (DVC0) | SCU (DVC0) | SCU (DVC0) | SCU (DVC0) |
| 18 | SCU (DVC1) | SCU (DVC1) | SCU (DVC1) | SCU (DVC1) |
| 17 | SCU (all)*1 | SCU (all)*1 | SCU (all)*1 | SCU (all)*1 |
| 16 | — | — | — | — |
| 15 | SSI0 | SSI0 | SSI0 | SSI0 |
| 14 | SSI1 | SSI1 | SSI1 | SSI1 |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|-------------|-------------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 13 | SSI2 | SSI2 | SSI2 | SSI2 |
| 12 | SSI3 | SSI3 | SSI3 | SSI3 |
| 11 | SSI4 | SSI4 | SSI4 | SSI4 |
| 10 | SSI5 | SSI5 | SSI5 | SSI5 |
| 9 | SSI6 | SSI6 | SSI6 | SSI6 |
| 8 | SSI7 | SSI7 | SSI7 | SSI7 |
| 7 | SSI8 | SSI8 | SSI8 | SSI8 |
| 6 | SSI9 | SSI9 | SSI9 | SSI9 |
| 5 | SSI (all)*2 | SSI (all)*2 | SSI (all)*2 | SSI (all)*2 |
| 4 | — | — | — | — |
| 3 | — | — | — | I2C-IF7 |
| 2 | — | — | — | — |
| 1 | — | — | — | — |
| 0 | — | — | — | — |

Notes: "—" indicates the bit is reserved. Set the read value.

1. When the MSTPST1017 bit is set to 1, supply of the clock signal to the circuits assigned to bits MSTPST1031 to MSTPST1018 is stopped simultaneously without writing 1 to these bits. When supply of the clock signal to any of the circuits assigned to bits MSTPST1031 to MSTPST1018 is to be enabled, clear the MSTPST1017 bit.
2. When the MSTPST1005 bit is set to 1, supply of the clock signal to the SSI0 to SSI9 modules is stopped simultaneously without setting bits MSTPST1015 to MSTPST1006 to 1. When supply of the clock signal to any of the SSI0 to SSI9 modules is to be enabled, clear the MSTPST1005 bit.

12.2.2 Module Stop Control Register (RMSTPCRn/SMSTPCRn (n = 1 to 10))

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

RMSTPCRn and SMSTPCRn are 32-bit readable/writable registers which control supply of the clock signal to the modules assigned to the corresponding bits. RMSTPCRn and SMSTPCRn registers are for the AP realtime CPU core and the AP system CPU core.

When all of the control bits in RMSTPCRn, SMSTPCRn and SCMSTPCRn are set to 1 (halt), input of the clock signal to the module is halted (the module is on standby). When a corresponding control bit of RMSTPCRn, SMSTPCRn or SCMSTPCRn is 0, the clock signal is supplied.

Whether the module is stopped or not can be checked through a corresponding bit in MSTPSRn register.

Setting a bit in this register to 1 request that the module be placed on standby and setting a bit to 0 request that the module operate.

The reserved bits should be set to the values read from the bits. Positions, names and initial values of each bit are shown below. The tables from 12.13 to 12.22 show the assignment of modules to bits.

12.2.2.1 Realtime Module Stop Control Register 1 (RMSTPCR1)/System Module Stop Control Register 1 (SMSTPCR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------|-------------|------|-------------|------|------|-------------|-------------|-------------|-------------|-------------|----|-------------|-------------|------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MSTP 131 | MSTP 130 | — | MSTP 128 | — | — | MSTP 125 | MSTP 124 | MSTP 123 | MSTP 122 | MSTP 121 | — | MSTP 119 | MSTP 118 | — | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W* | R/W* | R | R/W* | R/W* | R | R/W | R/W | R/W | R/W | R/W | R | R/W* | R/W* | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | MSTP 112 | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 1 | 1 | 1 | —*1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R | R/W | R | R | R/W* | R/W* | R |

Note: *1. When debug mode, initial value is 0, in other case is 1.

0: Enables supply of the clock signal to the corresponding module.

1: Stops supply of the clock signal to the corresponding module.

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.13 Assignment of Modules to Bits in RMSTPCR1 and SMSTPCR1

| Bit | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----|----------------------------|-----------------|----------------------------|----------------------------|----------------------------|
| 31 | VCP4 (VDPB) | VCP4 (VDPB) | VCP4 (VDPB) | VCP4 (VDPB) | VCP4 (VDPB) |
| 30 | VCP4 (VCPLF), iVDP1C | VCP4 (VCPLF) | VCP4 (VCPLF), iVDP1C | VCP4 (VCPLF), iVDP1C | VCP4 (VCPLF), iVDP1C |
| 29 | — | — | — | — | — |
| 28 | — | iVDP1C | — | — | — |
| 27 | — | — | — | — | — |
| 26 | — | — | — | — | — |

| Bit | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----|--------|-------------|-------------|--------|--------|
| 25 | TMU0 | TMU0 | TMU0 | TMU0 | TMU0 |
| 24 | TMU1 | TMU1 | TMU1 | TMU1 | TMU1 |
| 23 | TMU2 | TMU2 | TMU2 | TMU2 | TMU2 |
| 22 | TMU3 | TMU3 | TMU3 | TMU3 | TMU3 |
| 21 | TMU4 | TMU4 | TMU4 | TMU4 | TMU4 |
| 20 | — | — | — | — | — |
| 19 | FDP1-0 | FDP1-0 | FDP1-0 | FDP1-0 | FDP1-0 |
| 18 | FDP1-1 | — | — | — | — |
| 17 | — | — | — | — | — |
| 16 | — | — | — | — | — |
| 15 | — | — | — | — | — |
| 14 | — | — | — | — | — |
| 13 | — | — | — | — | — |
| 12 | 3DGE | 3DGE | 3DGE | 3DGE | 3DGE |
| 11 | — | — | — | — | — |
| 10 | — | — | — | — | — |
| 9 | — | — | — | — | — |
| 8 | — | — | — | — | — |
| 7 | — | — | — | — | — |
| 6 | — | — | — | — | — |
| 5 | — | — | — | — | — |
| 4 | — | — | — | — | — |
| 3 | — | — | — | — | — |
| 2 | — | — | — | — | — |
| 1 | — | — | — | — | — |
| 0 | — | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.2.2 Realtime Module Stop Control Register 2 (RMSTPCR2)/System Module Stop Control Register 2 (SMSTPCR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------|----|------|-----|----------|----------|----------|----------|----------|----------|----|----------|----------|----------|----------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | MSTP 219 | MSTP 218 | MSTP 217 | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W* | R/W | R | R/W | R | R | R | R | R | R | R/W* | R/W | R/W | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | MSTP 211 | MSTP 210 | MSTP 209 | MSTP 208 | MSTP 207 | MSTP 206 | — | MSTP 204 | MSTP 203 | MSTP 202 | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W* | R | R |

0: Enables supply of the clock signal to the corresponding module.

1: Stops supply of the clock signal to the corresponding module.

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.14 Assignment of Modules to Bits in RMSTPCR2 and SMSTPCR2

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|-----------|-----------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | — | — | — | — |
| 30 | — | — | — | — |
| 29 | — | — | — | — |
| 28 | — | — | — | — |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | — | — | — | — |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | SYS-DMAC0 | SYS-DMAC0 | SYS-DMAC0 | SYS-DMAC0 |
| 18 | SYS-DMAC1 | SYS-DMAC1 | SYS-DMAC1 | SYS-DMAC1 |
| 17 | SYS-DMAC2 | SYS-DMAC2 | SYS-DMAC2 | SYS-DMAC2 |
| 16 | — | — | — | — |
| 15 | — | — | — | — |
| 14 | — | — | — | — |
| 13 | — | — | — | — |
| 12 | — | — | — | — |
| 11 | MSIOF0 | MSIOF0 | MSIOF0 | MSIOF0 |
| 10 | MSIOF1 | MSIOF1 | MSIOF1 | MSIOF1 |
| 9 | MSIOF2 | MSIOF2 | MSIOF2 | MSIOF2 |
| 8 | MSIOF3 | MSIOF3 | MSIOF3 | MSIOF3 |

| Bit | RZ/G2H | RZ/G2M V1.3 | | |
|-----|--------|-------------|--------|--------|
| | | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 7 | SCIF0 | SCIF0 | SCIF0 | SCIF0 |
| 6 | SCIF1 | SCIF1 | SCIF1 | SCIF1 |
| 5 | — | — | — | — |
| 4 | SCIF3 | SCIF3 | SCIF3 | SCIF3 |
| 3 | SCIF4 | SCIF4 | SCIF4 | SCIF4 |
| 2 | SCIF5 | SCIF5 | SCIF5 | SCIF5 |
| 1 | — | — | — | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.2.3 Realtime Module Stop Control Register 3 (RMSTPCR3)/System Module Stop Control Register 3 (SMSTPCR3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----------|----------|----------|----------|----------|----------|----|----|------|------|----|----------|----------|----------|----------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MSTP 331 | MSTP 330 | — | MSTP 328 | — | — | — | — | — | — | — | — | MSTP 319 | MSTP 318 | — | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W* | R/W* | R/W* | R/W* | R | R/W* | R | R | R/W* | R/W* | R | R | R/W* | R/W* | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | MSTP 314 | MSTP 313 | MSTP 312 | MSTP 311 | MSTP 310 | — | — | — | — | — | MSTP 304 | MSTP 303 | MSTP 302 | MSTP 301 | MSTP 300 |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W* | R/W | R/W* | R/W* | R/W* | R/W* | R | R | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W |

0: Enables supply of the clock signal to the corresponding module.

1: Stops supply of the clock signal to the corresponding module.

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.15 Assignment of Modules to Bits in RMSTPCR3 and SMSTPCR3

| Bit | RZ/G2H | RZ/G2M V1.3 | | RZ/G2E |
|-----|-------------|-------------|-------------|-------------|
| | | RZ/G2M V3.0 | RZ/G2N | |
| 31 | USB-DMAC0-1 | USB-DMAC1 | USB-DMAC0-1 | USB-DMAC0-1 |
| 30 | USB-DMAC0-0 | USB-DMAC0 | USB-DMAC0-0 | USB-DMAC0-0 |
| 29 | — | — | — | — |
| 28 | USB3.0-IF0 | USB3.0-IF0 | USB3.0-IF0 | USB3.0-IF0 |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | — | — | — | — |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | PCIEC0 | PCIEC0 | PCIEC0 | PCIEC0 |
| 18 | PCIEC1 | PCIEC1 | PCIEC1 | — |
| 17 | — | — | — | — |
| 16 | — | — | — | — |
| 15 | — | — | — | — |
| 14 | SD-IF0 | SD-IF0 | SD-IF0 | SD-IF0 |
| 13 | SD-IF1 | SD-IF1 | SD-IF1 | SD-IF1 |
| 12 | SD-IF2 | SD-IF2 | SD-IF2 | — |
| 11 | SD-IF3 | SD-IF3 | SD-IF3 | SD-IF3 |
| 10 | SCIF2 | SCIF2 | SCIF2 | SCIF2 |
| 9 | — | — | — | — |

| Bit | RZ/G2H | RZ/G2M V1.3 | | |
|-----|--------|-------------|--------|--------|
| | | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 8 | — | — | — | — |
| 7 | — | — | — | — |
| 6 | — | — | — | — |
| 5 | — | — | — | — |
| 4 | TPU0 | TPU0 | TPU0 | TPU0 |
| 3 | CMT0 | CMT0 | CMT0 | CMT0 |
| 2 | CMT1 | CMT1 | CMT1 | CMT1 |
| 1 | CMT2 | CMT2 | CMT2 | CMT2 |
| 0 | CMT3 | CMT3 | CMT3 | CMT3 |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.2.4 Realtime Module Stop Control Register 4 (RMSTPCR4)/System Module Stop Control Register 4 (SMSTPCR4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------|----|----|----|----|----|----|----|----|----|----|----|----|-------------|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MSTP 431 | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | MSTP 402 | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R |

Note: * Writing to a bit is only possible if the product has the corresponding module.

0: Enables supply of the clock signal to the corresponding module.

1: Stops supply of the clock signal to the corresponding module.

Table 12.16 Assignment of Modules to Bits in RMSTPCR4 and SMSTPCR4

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|--------|--------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | SUCMT | SUCMT | SUCMT | SUCMT |
| 30 | — | — | — | — |
| 29 | — | — | — | — |
| 28 | — | — | — | — |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | — | — | — | — |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | — | — | — | — |
| 18 | — | — | — | — |
| 17 | — | — | — | — |
| 16 | — | — | — | — |
| 15 | — | — | — | — |
| 14 | — | — | — | — |
| 13 | — | — | — | — |
| 12 | — | — | — | — |
| 11 | — | — | — | — |
| 10 | — | — | — | — |
| 9 | — | — | — | — |
| 8 | — | — | — | — |
| 7 | — | — | — | — |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|--------|--------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 6 | — | — | — | — |
| 5 | — | — | — | — |
| 4 | — | — | — | — |
| 3 | — | — | — | — |
| 2 | RWDT | RWDT | RWDT | RWDT |
| 1 | — | — | — | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.2.5 Realtime Module Stop Control Register 5 (RMSTPCR5)/System Module Stop Control Register 5 (SMSTPCR5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|----------|----------|------|----------|----------|----------|----------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | MSTP 523 | MSTP 522 | — | MSTP 520 | MSTP 519 | MSTP 518 | MSTP 517 | MSTP 516 |
| Initial value: | | | | | | | | | | | | | | | | |
| | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W | R/W | R/W* | R/W | R/W | R/W | R/W | R/W* |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | MSTP 502 | MSTP 501 | — |
| Initial value: | | | | | | | | | | | | | | | | |
| | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R | R/W* | R/W* | R | R/W* | R/W* | R/W* | R/W* |

0: Enables supply of the clock signal to the corresponding module.

1: Stops supply of the clock signal to the corresponding module.

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.17 Assignment of Modules to Bits in RMSTPCR5 and SMSTPCR5

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|---------|---------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | — | — | — | — |
| 30 | — | — | — | — |
| 29 | — | — | — | — |
| 28 | — | — | — | — |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | PWM | PWM | PWM | PWM |
| 22 | THS/TSC | THS/TSC | THS/TSC | THS/TSC |
| 21 | — | — | — | — |
| 20 | HSCIF0 | HSCIF0 | HSCIF0 | HSCIF0 |
| 19 | HSCIF1 | HSCIF1 | HSCIF1 | HSCIF1 |
| 18 | HSCIF2 | HSCIF2 | HSCIF2 | HSCIF2 |
| 17 | HSCIF3 | HSCIF3 | HSCIF3 | HSCIF3 |
| 16 | HSCIF4 | HSCIF4 | HSCIF4 | HSCIF4 |
| 15 | — | — | — | — |
| 14 | — | — | — | — |
| 13 | — | — | — | — |
| 12 | — | — | — | — |
| 11 | — | — | — | — |
| 10 | — | — | — | — |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|-------------|-------------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 9 | — | — | — | — |
| 8 | — | — | — | — |
| 7 | — | — | — | — |
| 6 | — | — | — | — |
| 5 | — | — | — | — |
| 4 | — | — | — | — |
| 3 | — | — | — | — |
| 2 | Audio-DMAC0 | Audio-DMAC0 | Audio-DMAC0 | Audio-DMAC0 |
| 1 | Audio-DMAC1 | Audio-DMAC1 | Audio-DMAC1 | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.2.6 Realtime Module Stop Control Register 6 (RMSTPCR6)/System Module Stop Control Register 6 (SMSTPCR6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------|-------------|----|------|-------------|-------------|------|-------------|-------------|-------------|-------------|------|-------------|-------------|-------------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MSTP 631 | MSTP 630 | — | — | — | MSTP 626 | — | MSTP 624 | MSTP 623 | MSTP 622 | MSTP 621 | — | MSTP 619 | — | MSTP 617 | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W* | R/W* | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W | R/W* | R/W* | R | R/W* | R/W* | R/W* | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MSTP 615 | MSTP 614 | — | — | MSTP 611 | MSTP 610 | — | — | MSTP 607 | MSTP 606 | — | — | MSTP 603 | MSTP 602 | MSTP 601 | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W* | R/W* | R | R/W* | R/W* | R/W* | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W | R/W* | R/W* | R |

0: Enables supply of the clock signal to the corresponding module.

1: Stops supply of the clock signal to the corresponding module.

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.18 Assignment of Modules to Bits in RMSTPCR6 and SMSTPCR6

| Bit | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----|-------------|-------------|-------------|-------------|-------------|
| 31 | VSP (VSPI0) | VSP (VSPI0) | VSP (VSPI0) | VSP (VSPI0) | VSP (VSPI0) |
| 30 | VSP (VSPI1) | — | — | — | — |
| 29 | — | — | — | — | — |
| 28 | — | — | — | — | — |
| 27 | — | — | — | — | — |
| 26 | VSP (VSPBD) | VSP (VSPB) | VSP (VSPB) | VSP (VSPB) | VSP (VSPB) |
| 25 | — | — | — | — | — |
| 24 | VSP (VSPBC) | — | — | — | — |
| 23 | VSP (VSPD0) | VSP (VSPD0) | VSP (VSPD0) | VSP (VSPD0) | VSP (VSPD0) |
| 22 | VSP (VSPD1) | VSP (VSPD1) | VSP (VSPD1) | VSP (VSPD1) | VSP (VSPD1) |
| 21 | — | VSP (VSPD2) | VSP (VSPD2) | — | — |
| 20 | — | — | — | — | — |
| 19 | FCPCS | FCPCS | FCPCS | FCPCS | FCPCS |
| 18 | — | — | — | — | — |
| 17 | — | FCPCI0 | — | — | — |
| 16 | — | — | — | — | — |
| 15 | FCPF0 | FCPF0 | FCPF0 | FCPF0 | FCPF0 |
| 14 | FCPF1 | — | — | — | — |
| 13 | — | — | — | — | — |
| 12 | — | — | — | — | — |
| 11 | FCPVI0 | FCPVI0 | FCPVI0 | FCPVI0 | FCPVI0 |
| 10 | FCPVI1 | — | — | — | — |
| 9 | — | — | — | — | — |

| Bit | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----|--------|-------------|-------------|--------|--------|
| 8 | — | — | — | — | — |
| 7 | FCPVB0 | FCPVB0 | FCPVB0 | FCPVB0 | FCPVB0 |
| 6 | FCPVB1 | — | — | — | — |
| 5 | — | — | — | — | — |
| 4 | — | — | — | — | — |
| 3 | FCPVD0 | FCPVD0 | FCPVD0 | FCPVD0 | FCPVD0 |
| 2 | FCPVD1 | FCPVD1 | FCPVD1 | FCPVD1 | FCPVD1 |
| 1 | — | FCPVD2 | FCPVD2 | — | — |
| 0 | — | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.2.7 Realtime Module Stop Control Register 7 (RMSTPCR7)/System Module Stop Control Register 7 (SMSTPCR7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------|----------|----------|------|----------|------|------|----------|----------|----------|----------|----------|----------|----------|------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | MSTP 729 | — | MSTP 727 | — | — | MSTP 724 | MSTP 723 | MSTP 722 | MSTP 721 | — | — | — | — | MSTP 716 |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R/W* | R/W* | R/W* | R | R | R/W | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | MSTP 714 | — | — | — | — | — | — | — | — | — | MSTP 704 | MSTP 703 | MSTP 702 | — | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W* | R/W* | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* |

0: Enables supply of the clock signal to the corresponding module.

1: Stops supply of the clock signal to the corresponding module.

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.19 Assignment of Modules to Bits in RMSTPCR7 and SMSTPCR7

| Bit | RZ/G2H | RZ/G2M V1.3 | | RZ/G2E |
|-----|----------|-------------|----------|---------|
| | | RZ/G2M V3.0 | RZ/G2N | |
| 31 | — | — | — | — |
| 30 | — | — | — | — |
| 29 | HDMI-IF0 | HDMI-IF0 | HDMI-IF0 | — |
| 28 | — | — | — | — |
| 27 | LVDS-IF | LVDS-IF | LVDS-IF | LVDS-IF |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | DU0 | DU0 | DU0 | DU0 |
| 23 | DU1 | DU1 | DU1 | DU1 |
| 22 | — | DU2 | — | — |
| 21 | DU3 | — | DU3 | — |
| 20 | — | — | — | — |
| 19 | — | — | — | — |
| 18 | — | — | — | — |
| 17 | — | — | — | — |
| 16 | CSI40 | CSI40 | CSI40 | CSI40 |
| 15 | — | — | — | — |
| 14 | CSI20 | CSI20 | CSI20 | — |
| 13 | — | — | — | — |
| 12 | — | — | — | — |
| 11 | — | — | — | — |
| 10 | — | — | — | — |
| 9 | — | — | — | — |
| 8 | — | — | — | — |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|------------|------------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 7 | — | — | — | — |
| 6 | — | — | — | — |
| 5 | — | — | — | — |
| 4 | HS-USB-IF0 | HS-USB-IF0 | HS-USB-IF0 | HS-USB-IF0 |
| 3 | EHCI/OHCI0 | EHCI/OHCI0 | EHCI/OHCI0 | EHCI/OHCI0 |
| 2 | EHCI/OHCI1 | EHCI/OHCI1 | EHCI/OHCI1 | — |
| 1 | — | — | — | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.2.8 Realtime Module Stop Control Register 8 (RMSTPCR8)/System Module Stop Control Register 8 (SMSTPCR8)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |

Initial value: Values of bit 24 is different among products.

| | | | | | | | | | | | | | | | | | |
|-------------|------|------|------|------|------|------|------|-----|-----|-----|-----|------|------|---|---|------|---|
| RZ/G2H | | | | | | | | | | | | | | | | | |
| RZ/G2M V1.3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | |
| RZ/G2M V3.0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | |
| RZ/G2E | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | |
| RZ/G2N | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | |
| R/W: | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W | R/W | R/W | R/W | R/W* | R/W* | R | R | R/W* | R |

| | | | | | | | | | | | | | | | | |
|------|-------------|----|----|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|---|---|---|-------------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MSTP 815 | — | — | MSTP 812 | MSTP 811 | MSTP 810 | MSTP 809 | MSTP 808 | MSTP 807 | MSTP 806 | MSTP 805 | MSTP 804 | — | — | — | MSTP 800 |

| | | | | | | | | | | | | | | | | | |
|----------------|------|------|------|-----|-----|-----|-----|-----|-----|------|------|------|------|---|------|------|------|
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| R/W: | R/W* | R/W* | R/W* | R/W | R/W | R/W | R/W | R/W | R/W | R/W* | R/W* | R/W* | R/W* | R | R/W* | R/W* | R/W* |

0: Enables supply of the clock signal to the corresponding module.

1: Stops supply of the clock signal to the corresponding module.

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.20 Assignment of Modules to Bits in RMSTPCR8 and SMSTPCR8

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|---------|---------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | — | — | — | — |
| 30 | — | — | — | — |
| 29 | — | — | — | — |
| 28 | — | — | — | — |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | — | — | — | — |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | — | — | — | — |
| 18 | — | — | — | — |
| 17 | — | — | — | — |
| 16 | — | — | — | — |
| 15 | SATA-IF | — | SATA-IF | — |
| 14 | — | — | — | — |
| 13 | — | — | — | — |
| 12 | EAVB-IF | EAVB-IF | EAVB-IF | EAVB-IF |
| 11 | VIN0 | VIN0 | VIN0 | — |

| Bit | RZ/G2H | RZ/G2M V1.3 | | |
|-----|--------|-------------|---------|---------|
| | | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 10 | VIN1 | VIN1 | VIN1 | — |
| 9 | VIN2 | VIN2 | VIN2 | — |
| 8 | VIN3 | VIN3 | VIN3 | — |
| 7 | VIN4 | VIN4 | VIN4 | VIN4 |
| 6 | VIN5 | VIN5 | VIN5 | VIN5 |
| 5 | VIN6 | VIN6 | VIN6 | — |
| 4 | VIN7 | VIN7 | VIN7 | — |
| 3 | — | — | — | — |
| 2 | — | — | — | — |
| 1 | — | — | — | — |
| 0 | — | R-NANDC | R-NANDC | R-NANDC |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.2.9 Realtime Module Stop Control Register 9 (RMSTPCR9)/System Module Stop Control Register 9 (SMSTPCR9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|-------|----------|----------|----------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MSTP 931 | MSTP 930 | MSTP 929 | MSTP 928 | MSTP 927 | MSTP 926 | — | — | — | MSTP 922 | — | — | MSTP 919 | MSTP 918 | MSTP 917 | MSTP 916 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | *1 | 1 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W*2 | R | R/W*2 | R/W*2 | R/W*2 | R | R/W*2 | R/W*2 | R/W*2 | R/W | R/W*2 |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MSTP 915 | MSTP 914 | — | MSTP 912 | MSTP 911 | MSTP 910 | MSTP 909 | MSTP 908 | MSTP 907 | MSTP 906 | MSTP 905 | — | — | — | — | — |
| Initial value: | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| R/W: | R/W*2 | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W*2 | R/W*2 | R/W*2 | R/W*2 | R/W*2 | R/W*2 | R |

0: Enables supply of the clock signal to the corresponding module.

1: Stops supply of the clock signal to the corresponding module.

Notes: 1. [RZ/G2H, RZ/G2N, RZ/G2E]

Initial value is 0.

[RZ/G2M V1.3, RZ/G2M V3.0]

The initial value of bit 17 depends on the mode pins MD4, MD3, MD2 and MD1.

When (MD4, MD3, MD2, MD1) = 0001/0010/0011/0100/1010/1011, initial value is 0, other cases, initial value is 1.

2. Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.21 Assignment of Modules to Bits in RMSTPCR9 and SMSTPCR9

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|----------|----------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | I2C-IF0 | I2C-IF0 | I2C-IF0 | I2C-IF0 |
| 30 | I2C-IF1 | I2C-IF1 | I2C-IF1 | I2C-IF1 |
| 29 | I2C-IF2 | I2C-IF2 | I2C-IF2 | I2C-IF2 |
| 28 | I2C-IF3 | I2C-IF3 | I2C-IF3 | I2C-IF3 |
| 27 | I2C-IF4 | I2C-IF4 | I2C-IF4 | I2C-IF4 |
| 26 | IIC-PMIC | IIC-PMIC | IIC-PMIC | IIC-PMIC |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | ADG | ADG | ADG | ADG |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | I2C-IF5 | I2C-IF5 | I2C-IF5 | I2C-IF5 |
| 18 | I2C-IF6 | I2C-IF6 | I2C-IF6 | I2C-IF6 |
| 17 | RPC-IF | RPC-IF | RPC-IF | RPC-IF |
| 16 | CAN-IF0 | CAN-IF0 | CAN-IF0 | CAN-IF0 |
| 15 | CAN-IF1 | CAN-IF1 | CAN-IF1 | CAN-IF1 |
| 14 | CAN-FD | CAN-FD | CAN-FD | CAN-FD |
| 13 | — | — | — | — |
| 12 | GPIO0 | GPIO0 | GPIO0 | GPIO0 |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|--------|--------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 11 | GPIO1 | GPIO1 | GPIO1 | GPIO1 |
| 10 | GPIO2 | GPIO2 | GPIO2 | GPIO2 |
| 9 | GPIO3 | GPIO3 | GPIO3 | GPIO3 |
| 8 | GPIO4 | GPIO4 | GPIO4 | GPIO4 |
| 7 | GPIO5 | GPIO5 | GPIO5 | GPIO5 |
| 6 | GPIO6 | GPIO6 | GPIO6 | GPIO6 |
| 5 | GPIO7 | GPIO7 | GPIO7 | — |
| 4 | — | — | — | — |
| 3 | — | — | — | — |
| 2 | — | — | — | — |
| 1 | — | — | — | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.2.10 Realtime Module Stop Control Register 10 (RMSTPCR10)/System Module Stop Control Register 10 (SMSTPCR10)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MSTP 1031 | MSTP 1030 | MSTP 1029 | MSTP 1028 | MSTP 1027 | MSTP 1026 | MSTP 1025 | MSTP 1024 | MSTP 1023 | MSTP 1022 | MSTP 1021 | MSTP 1020 | MSTP 1019 | MSTP 1018 | MSTP 1017 | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | *1 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MSTP 1015 | MSTP 1014 | MSTP 1013 | MSTP 1012 | MSTP 1011 | MSTP 1010 | MSTP 1009 | MSTP 1008 | MSTP 1007 | MSTP 1006 | MSTP 1005 | — | MSTP 1003 | — | — | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | *2 | *2 | *2 | *2 | *2 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

- Notes:
- The initial value of bit 16 varies depending on the products.
 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]: The initial value is 0.
 [RZ/G2N, RZ/G2E]: The initial value is 1.
 - The initial values of bits 4 to 0 vary depending on the products.
 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]: The initial value is 0.
 0: Enables supply of the clock signal to the corresponding module.
 1: Stops supply of the clock signal to the corresponding module.

Table 12.22 Assignment of Modules to Bits in RMSTPCR10 and SMSTPCR10

| Bit | RZ/G2M V1.3 | | | |
|-----|--|--|--|--|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | SCU (SRC0) | SCU (SRC0) | SCU (SRC0) | SCU (SRC0) |
| 30 | SCU (SRC1) | SCU (SRC1) | SCU (SRC1) | SCU (SRC1) |
| 29 | SCU (SRC2) | SCU (SRC2) | SCU (SRC2) | SCU (SRC2) |
| 28 | SCU (SRC3) | SCU (SRC3) | SCU (SRC3) | SCU (SRC3) |
| 27 | SCU (SRC4) | SCU (SRC4) | SCU (SRC4) | SCU (SRC4) |
| 26 | SCU (SRC5) | SCU (SRC5) | SCU (SRC5) | SCU (SRC5) |
| 25 | SCU (SRC6) | SCU (SRC6) | SCU (SRC6) | SCU (SRC6) |
| 24 | SCU (SRC7) | SCU (SRC7) | SCU (SRC7) | SCU (SRC7) |
| 23 | SCU (SRC8) | SCU (SRC8) | SCU (SRC8) | SCU (SRC8) |
| 22 | SCU (SRC9) | SCU (SRC9) | SCU (SRC9) | SCU (SRC9) |
| 21 | SCU (CTU00, CTU01, CTU02, CTU03, MIX0) | SCU (CTU00, CTU01, CTU02, CTU03, MIX0) | SCU (CTU00, CTU01, CTU02, CTU03, MIX0) | SCU (CTU00, CTU01, CTU02, CTU03, MIX0) |
| 20 | SCU (CTU10, CTU11, CTU12, CTU13, MIX1) | SCU (CTU10, CTU11, CTU12, CTU13, MIX1) | SCU (CTU10, CTU11, CTU12, CTU13, MIX1) | SCU (CTU10, CTU11, CTU12, CTU13, MIX1) |
| 19 | SCU (DVC0) | SCU (DVC0) | SCU (DVC0) | SCU (DVC0) |
| 18 | SCU (DVC1) | SCU (DVC1) | SCU (DVC1) | SCU (DVC1) |
| 17 | SCU (all)*1 | SCU (all)*1 | SCU (all)*1 | SCU (all)*1 |
| 16 | — | — | — | — |
| 15 | SSI0 | SSI0 | SSI0 | SSI0 |

| Bit | RZ/G2H | RZ/G2M V1.3 | | |
|-----|-------------|-------------|-------------|-------------|
| | | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 14 | SSI1 | SSI1 | SSI1 | SSI1 |
| 13 | SSI2 | SSI2 | SSI2 | SSI2 |
| 12 | SSI3 | SSI3 | SSI3 | SSI3 |
| 11 | SSI4 | SSI4 | SSI4 | SSI4 |
| 10 | SSI5 | SSI5 | SSI5 | SSI5 |
| 9 | SSI6 | SSI6 | SSI6 | SSI6 |
| 8 | SSI7 | SSI7 | SSI7 | SSI7 |
| 7 | SSI8 | SSI8 | SSI8 | SSI8 |
| 6 | SSI9 | SSI9 | SSI9 | SSI9 |
| 5 | SSI (all)*2 | SSI (all)*2 | SSI (all)*2 | SSI (all)*2 |
| 4 | — | — | — | — |
| 3 | — | — | — | I2C-IF7 |
| 2 | — | — | — | — |
| 1 | — | — | — | — |
| 0 | — | — | — | — |

Notes: "—" indicates the bit is reserved. Set the read value.

1. When the MSTP1017 bit is set to 1, supply of the clock signal to the circuits assigned to bits MSTP1031 to MSTP1018 is stopped simultaneously without writing 1 to these bits. When supply of the clock signal to any of the circuits assigned to bits MSTP1031 to MSTP1018 is to be enabled, clear the MSTP1017 bit.
2. When the MSTP1005 bit is set to 1, supply of the clock signal to the SSI0 to SSI9 modules is stopped simultaneously without setting bits MSTP1015 to MSTP1006 to 1. When supply of the clock signal to any of the SSI0 to SSI9 modules is to be enabled, clear the MSTP1005 bit.

12.2.3 Software Reset Register (SRCRn (n = 1 to 10))

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

Each SRCRn is a 32-bit readable/writable register. Each bit is assigned to a module in the chip. Writing 1 to the bit resets the corresponding module. Writing 1 to a given the bit of the software reset clearing register (SRSTCLRn (n = 0 to 11)) clears the corresponding bit of SRCRn and releases the reset. SRCRn and SRSTCLRn does not need read-modify-write sequence. Write 1 to only target module bit of SRCRn or SRSTCLRn. It takes maximum 1 cycle by RCLK to complete software reset after setting 1 to target bit of SRCRn. When release software reset, it is necessary to wait more than 1 cycle by RCLK before setting 1 to target bit of SRSTCLRn. If applying a software reset to a module connected to the AXI, wait the completion of any data transfer.

Bit assignment and initial values are shown in the tables from 12.28 to 12.32.

12.2.3.1 Software Reset Register 1 (SRCR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------|------------|------|------|------------|------------|------------|------------|------------|----|------------|------------|------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SRT 131 | SRT 130 | — | SRT 128 | — | — | SRT 125 | SRT 124 | SRT 123 | SRT 122 | SRT 121 | — | SRT 119 | SRT 118 | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R/W* | R | R/W* | R/W* | R | R/W | R/W | R/W | R/W | R/W | R | R/W* | R/W* | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W* | R/W* | R | R/W* | R/W* | R/W* | R/W* | R/W* | R | R/W | R | R | R/W* | R/W* | R |

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.23 Assignment of Modules to Bits in SRCR1

| Bit | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----|----------------------|--------------|----------------------|----------------------|----------------------|
| 31 | VCP4 (VDPB) | VCP4 (VDPB) | VCP4 (VDPB) | VCP4 (VDPB) | VCP4 (VDPB) |
| 30 | VCP4 (VCPLF), iVDP1C | VCP4 (VCPLF) | VCP4 (VCPLF), iVDP1C | VCP4 (VCPLF), iVDP1C | VCP4 (VCPLF), iVDP1C |
| 29 | — | — | — | — | — |
| 28 | — | iVDP1C | — | — | — |
| 27 | — | — | — | — | — |
| 26 | — | — | — | — | — |
| 25 | TMU0 | TMU0 | TMU0 | TMU0 | TMU0 |
| 24 | TMU1 | TMU1 | TMU1 | TMU1 | TMU1 |
| 23 | TMU2 | TMU2 | TMU2 | TMU2 | TMU2 |
| 22 | TMU3 | TMU3 | TMU3 | TMU3 | TMU3 |
| 21 | TMU4 | TMU4 | TMU4 | TMU4 | TMU4 |
| 20 | — | — | — | — | — |
| 19 | FDP1-0 | FDP1-0 | FDP1-0 | FDP1-0 | FDP1-0 |
| 18 | FDP1-1 | — | — | — | — |
| 17 | — | — | — | — | — |
| 16 | — | — | — | — | — |
| 15 | — | — | — | — | — |
| 14 | — | — | — | — | — |
| 13 | — | — | — | — | — |
| 12 | — | — | — | — | — |
| 11 | — | — | — | — | — |
| 10 | — | — | — | — | — |
| 9 | — | — | — | — | — |
| 8 | — | — | — | — | — |
| 7 | — | — | — | — | — |
| 6 | — | — | — | — | — |

| Bit | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----|--------|-------------|-------------|--------|--------|
| 5 | — | — | — | — | — |
| 4 | — | — | — | — | — |
| 3 | — | — | — | — | — |
| 2 | — | — | — | — | — |
| 1 | — | — | — | — | — |
| 0 | — | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.3.2 Software Reset Register 2 (SRCR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------|----|------|-----|---------|---------|---------|---------|---------|---------|----|---------|---------|---------|---------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | SRT 219 | SRT 218 | SRT 217 | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | * | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W* | R/W | R | R/W | R | R | R | R/W* | R | R | R/W* | R/W | R/W | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | SRT 211 | SRT 210 | SRT 209 | SRT 208 | SRT 207 | SRT 206 | — | SRT 204 | SRT 203 | SRT 202 | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W* | R | R |

Notes: * Writing to a bit is only possible if the product actually has the corresponding module.

1. When MD7 = 1 and MD6 = 1, initial value is 0, other cases initial value is 1.

Table 12.24 Assignment of Modules to Bits in SRCR2

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|-----------|-----------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | — | — | — | — |
| 30 | — | — | — | — |
| 29 | — | — | — | — |
| 28 | — | — | — | — |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | — | — | — | — |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | SYS-DMAC0 | SYS-DMAC0 | SYS-DMAC0 | SYS-DMAC0 |
| 18 | SYS-DMAC1 | SYS-DMAC1 | SYS-DMAC1 | SYS-DMAC1 |
| 17 | SYS-DMAC2 | SYS-DMAC2 | SYS-DMAC2 | SYS-DMAC2 |
| 16 | — | — | — | — |
| 15 | — | — | — | — |
| 14 | — | — | — | — |
| 13 | — | — | — | — |
| 12 | — | — | — | — |
| 11 | MSIOF0 | MSIOF0 | MSIOF0 | MSIOF0 |
| 10 | MSIOF1 | MSIOF1 | MSIOF1 | MSIOF1 |
| 9 | MSIOF2 | MSIOF2 | MSIOF2 | MSIOF2 |
| 8 | MSIOF3 | MSIOF3 | MSIOF3 | MSIOF3 |
| 7 | SCIF0 | SCIF0 | SCIF0 | SCIF0 |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|--------|--------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 6 | SCIF1 | SCIF1 | SCIF1 | SCIF1 |
| 5 | — | — | — | — |
| 4 | SCIF3 | SCIF3 | SCIF3 | SCIF3 |
| 3 | SCIF4 | SCIF4 | SCIF4 | SCIF4 |
| 2 | SCIF5 | SCIF5 | SCIF5 | SCIF5 |
| 1 | — | — | — | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.3.3 Software Reset Register 3 (SRCR3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|------|------|------|------|------|------------|------------|------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SRT 331 | SRT 330 | — | SRT 328 | — | — | — | — | — | — | — | — | SRT 319 | SRT 318 | SRT 317 | SRT 316 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R/W* | R/W* | R/W* | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R | R/W* | R/W* | R/W* | R/W* |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | SRT 314 | SRT 313 | SRT 312 | SRT 311 | SRT 310 | — | — | — | — | — | SRT 304 | SRT 303 | SRT 302 | SRT 301 | SRT 300 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R/W | R/W* | R/W* | R/W* | R/W* | R/W* | R | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W |

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.25 Assignment of Modules to Bits in SRCR3

| Bit | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----|-------------|-------------|-------------|-------------|-------------|
| | 31 | USB-DMAC0-1 | USB-DMAC1 | USB-DMAC0-1 | USB-DMAC0-1 |
| 30 | USB-DMAC0-0 | USB-DMAC0 | USB-DMAC0-0 | USB-DMAC0-0 | USB-DMAC0-0 |
| 29 | — | — | — | — | — |
| 28 | USB3.0-IF0 | USB3.0-IF0 | USB3.0-IF0 | USB3.0-IF0 | USB3.0-IF0 |
| 27 | — | — | — | — | — |
| 26 | — | — | — | — | — |
| 25 | — | — | — | — | — |
| 24 | — | — | — | — | — |
| 23 | — | — | — | — | — |
| 22 | — | — | — | — | — |
| 21 | — | — | — | — | — |
| 20 | — | — | — | — | — |
| 19 | PCIEC0 | PCIEC0 | PCIEC0 | PCIEC0 | PCIEC0 |
| 18 | PCIEC1 | PCIEC1 | PCIEC1 | PCIEC1 | — |
| 17 | PCIE0_PIPE | PCIE0_PIPE | PCIE0_PIPE | PCIE0_PIPE | PCIE0_PIPE |
| 16 | PCIE1_PIPE | PCIE1_PIPE | PCIE1_PIPE | PCIE1_PIPE | — |
| 15 | — | — | — | — | — |
| 14 | SD-IF0 | SD-IF0 | SD-IF0 | SD-IF0 | SD-IF0 |
| 13 | SD-IF1 | SD-IF1 | SD-IF1 | SD-IF1 | SD-IF1 |
| 12 | SD-IF2 | SD-IF2 | SD-IF2 | SD-IF2 | — |
| 11 | SD-IF3 | SD-IF3 | SD-IF3 | SD-IF3 | SD-IF3 |
| 10 | SCIF2 | SCIF2 | SCIF2 | SCIF2 | SCIF2 |
| 9 | — | — | — | — | — |
| 8 | — | — | — | — | — |
| 7 | — | — | — | — | — |

| Bit | RZ/G2H | RZ/G2M V1.3 | | |
|-----|--------|-------------|--------|--------|
| | | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 6 | — | — | — | — |
| 5 | — | — | — | — |
| 4 | TPU0 | TPU0 | TPU0 | TPU0 |
| 3 | CMT0 | CMT0 | CMT0 | CMT0 |
| 2 | CMT1 | CMT1 | CMT1 | CMT1 |
| 1 | CMT2 | CMT2 | CMT2 | CMT2 |
| 0 | CMT3 | CMT3 | CMT3 | CMT3 |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.3.4 Software Reset Register 4 (SRCR4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|----|---------|----|----|----|----|----|----|----|----|----|------|---------|---------|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SRT 431 | — | SRT 429 | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R/W* | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | SRT 402 | SRT 401 | SRT 400 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W* | R/W | R/W* | R/W* |

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.26 Assignment of Modules to Bits in SRCR4

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|--------|--------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | SUCMT | SUCMT | SUCMT | SUCMT |
| 30 | — | — | — | — |
| 29 | — | — | — | DDR |
| 28 | — | — | — | — |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | — | — | — | — |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | — | — | — | — |
| 18 | — | — | — | — |
| 17 | — | — | — | — |
| 16 | — | — | — | — |
| 15 | — | — | — | — |
| 14 | — | — | — | — |
| 13 | — | — | — | — |
| 12 | — | — | — | — |
| 11 | — | — | — | — |
| 10 | — | — | — | — |
| 9 | — | — | — | — |
| 8 | — | — | — | — |
| 7 | — | — | — | — |
| 6 | — | — | — | — |
| 5 | — | — | — | — |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|--------|--------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 4 | — | — | — | — |
| 3 | — | — | — | — |
| 2 | RWDT | RWDT | RWDT | RWDT |
| 1 | SWDT | SWDT | SWDT | SWDT |
| 0 | SCMT | SCMT | SCMT | SCMT |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.3.5 Software Reset Register 5 (SRCR5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|---------|---------|------|---------|---------|---------|---------|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | SRT 523 | SRT 522 | — | SRT 520 | SRT 519 | SRT 518 | SRT 517 | SRT 516 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W | R/W | R/W* | R/W | R/W | R/W | R/W | R/W* |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | SRT 502 | SRT 501 | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* |

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.27 Assignment of Modules to Bits in SRCR5

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|---------|---------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | — | — | — | — |
| 30 | — | — | — | — |
| 29 | — | — | — | — |
| 28 | — | — | — | — |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | PWM | PWM | PWM | PWM |
| 22 | THS/TSC | THS/TSC | THS/TSC | THS/TSC |
| 21 | — | — | — | — |
| 20 | HSCIF0 | HSCIF0 | HSCIF0 | HSCIF0 |
| 19 | HSCIF1 | HSCIF1 | HSCIF1 | HSCIF1 |
| 18 | HSCIF2 | HSCIF2 | HSCIF2 | HSCIF2 |
| 17 | HSCIF3 | HSCIF3 | HSCIF3 | HSCIF3 |
| 16 | HSCIF4 | HSCIF4 | HSCIF4 | HSCIF4 |
| 15 | — | — | — | — |
| 14 | — | — | — | — |
| 13 | — | — | — | — |
| 12 | — | — | — | — |
| 11 | — | — | — | — |
| 10 | — | — | — | — |
| 9 | — | — | — | — |
| 8 | — | — | — | — |
| 7 | — | — | — | — |
| 6 | — | — | — | — |
| 5 | — | — | — | — |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|-------------|-------------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 4 | — | — | — | — |
| 3 | — | — | — | — |
| 2 | Audio-DMAC0 | Audio-DMAC0 | Audio-DMAC0 | Audio-DMAC0 |
| 1 | Audio-DMAC1 | Audio-DMAC1 | Audio-DMAC1 | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.3.6 Software Reset Register 6 (SRCR6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|---------|----|------|---------|---------|------|---------|---------|---------|---------|------|---------|---------|---------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SRT 631 | SRT 630 | — | — | — | SRT 626 | — | SRT 624 | SRT 623 | SRT 622 | SRT 621 | — | SRT 619 | — | SRT 617 | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R/W* | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W | R/W* | R/W* | R | R/W* | R/W* | R/W* | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SRT 615 | SRT 614 | — | — | SRT 611 | SRT 610 | — | — | SRT 607 | SRT 606 | — | — | SRT 603 | SRT 602 | SRT 601 | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R/W* | R | R/W* | R/W* | R/W* | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W | R/W* | R/W* | R |

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.28 Assignment of Modules to Bits in SRCR6

| Bit | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----|-------------|-------------|-------------|-------------|-------------|
| 31 | VSP (VSPI0) | VSP (VSPI0) | VSP (VSPI0) | VSP (VSPI0) | VSP (VSPI0) |
| 30 | VSP (VSPI1) | — | — | — | — |
| 29 | — | — | — | — | — |
| 28 | — | — | — | — | — |
| 27 | — | — | — | — | — |
| 26 | VSP (VSPBD) | VSP (VSPB) | VSP (VSPB) | VSP (VSPB) | VSP (VSPB) |
| 25 | — | — | — | — | — |
| 24 | VSP (VSPBC) | — | — | — | — |
| 23 | VSP (VSPD0) | VSP (VSPD0) | VSP (VSPD0) | VSP (VSPD0) | VSP (VSPD0) |
| 22 | VSP (VSPD1) | VSP (VSPD1) | VSP (VSPD1) | VSP (VSPD1) | VSP (VSPD1) |
| 21 | — | VSP (VSPD2) | VSP (VSPD2) | — | — |
| 20 | — | — | — | — | — |
| 19 | FCPCS | FCPCS | FCPCS | FCPCS | FCPCS |
| 18 | — | — | — | — | — |
| 17 | — | FCPCI | — | — | — |
| 16 | — | — | — | — | — |
| 15 | FCPF0 | FCPF0 | FCPF0 | FCPF0 | FCPF0 |
| 14 | FCPF1 | — | — | — | — |
| 13 | — | — | — | — | — |
| 12 | — | — | — | — | — |
| 11 | FCPVI0 | FCPVI0 | FCPVI0 | FCPVI0 | FCPVI0 |
| 10 | FCPVI1 | — | — | — | — |
| 9 | — | — | — | — | — |
| 8 | — | — | — | — | — |
| 7 | FCPVB0 | FCPVB0 | FCPVB0 | FCPVB0 | FCPVB0 |
| 6 | FCPVB1 | — | — | — | — |
| 5 | — | — | — | — | — |

| Bit | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----|--------|-------------|-------------|--------|--------|
| 4 | — | — | — | — | — |
| 3 | FCPVD0 | FCPVD0 | FCPVD0 | FCPVD0 | FCPVD0 |
| 2 | FCPVD1 | FCPVD1 | FCPVD1 | FCPVD1 | FCPVD1 |
| 1 | — | FCPVD2 | FCPVD2 | — | — |
| 0 | — | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.3.7 Software Reset Register 7 (SRCR7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|---------|---------|------|---------|---------|---------|---------|------|---------|------|---------|---------|---------|------|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | SRT 729 | — | SRT 727 | SRT 726 | SRT 725 | SRT 724 | — | SRT 722 | — | — | — | — | — | SRT 716 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W | R | R/W* | R | R/W* | R/W* | R/W* | R/W* | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SRT 715 | SRT 714 | — | — | — | — | — | — | — | — | — | SRT 704 | SRT 703 | SRT 702 | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R/W* | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* |

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.29 Assignment of Modules to Bits in SRCR7

| Bit | RZ/G2H | RZ/G2M V1.3 | | RZ/G2E |
|-----|----------|-------------|----------|-------------|
| | | RZ/G2M V3.0 | RZ/G2N | |
| 31 | — | — | — | — |
| 30 | — | — | — | — |
| 29 | HDMI-IF0 | HDMI-IF0 | HDMI-IF0 | — |
| 28 | — | — | — | — |
| 27 | LVDS-IF | LVDS-IF | LVDS-IF | LVDS-IF0 |
| 26 | — | — | — | LVDS-IF1 |
| 25 | — | — | — | LVDS-PLL0_1 |
| 24 | DU0, DU1 | DU0, DU1 | DU0, DU1 | DU0, DU1 |
| 23 | — | — | — | — |
| 22 | DU3 | DU2 | DU3 | — |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | — | — | — | — |
| 18 | — | — | — | — |
| 17 | — | — | — | — |
| 16 | CSI40 | CSI40 | CSI40 | CSI40 |
| 15 | CSI41 | — | — | — |
| 14 | CSI20 | CSI20 | CSI20 | — |
| 13 | — | — | — | — |
| 12 | — | — | — | — |
| 11 | — | — | — | — |
| 10 | — | — | — | — |
| 9 | — | — | — | — |
| 8 | — | — | — | — |
| 7 | — | — | — | — |
| 6 | — | — | — | — |
| 5 | — | — | — | — |

| Bit | RZ/G2H | RZ/G2M V1.3 | | |
|-----|------------|-------------|------------|------------|
| | | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 4 | HS-USB-IF0 | HS-USB-IF0 | HS-USB-IF0 | HS-USB-IF0 |
| 3 | EHCI/OHCI0 | EHCI/OHCI0 | EHCI/OHCI0 | EHCI/OHCI0 |
| 2 | EHCI/OHCI1 | EHCI/OHCI1 | EHCI/OHCI1 | — |
| 1 | — | — | — | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.3.8 Software Reset Register 8 (SRCR8)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|------|------|---------|---------|---------|---------|---------|---------|---------|---------|---------|------|------|------|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | *1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W | R/W | R/W | R/W* | R/W* | R/W* | R/W* | R/W* | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SRT 815 | — | — | SRT 812 | SRT 811 | SRT 810 | SRT 809 | SRT 808 | SRT 807 | SRT 806 | SRT 805 | SRT 804 | — | — | — | SRT 800 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R/W* | R/W* | R/W | R/W | R/W | R/W | R/W | R/W* | R/W* | R/W* | R/W* | R | R/W* | R/W* | R/W* |

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

*1 In non-secure mode, initial value is 0. In secure mode, initial value is 1

Table 12.30 Assignment of Modules to Bits in SRCR8

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|---------|---------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | — | — | — | — |
| 30 | — | — | — | — |
| 29 | — | — | — | — |
| 28 | — | — | — | — |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | — | — | — | — |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | — | — | — | — |
| 18 | — | — | — | — |
| 17 | — | — | — | — |
| 16 | — | — | — | — |
| 15 | SATA-IF | — | SATA-IF | — |
| 14 | — | — | — | — |
| 13 | — | — | — | — |
| 12 | EAVB-IF | EAVB-IF | EAVB-IF | EAVB-IF |
| 11 | VIN0 | VIN0 | VIN0 | — |
| 10 | VIN1 | VIN1 | VIN1 | — |
| 9 | VIN2 | VIN2 | VIN2 | — |
| 8 | VIN3 | VIN3 | VIN3 | — |
| 7 | VIN4 | VIN4 | VIN4 | VIN4 |
| 6 | VIN5 | VIN5 | VIN5 | VIN5 |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|---------|---------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 5 | VIN6 | VIN6 | VIN6 | — |
| 4 | VIN7 | VIN7 | VIN7 | — |
| 3 | — | — | — | — |
| 2 | — | — | — | — |
| 1 | — | — | — | — |
| 0 | — | R-NANDC | R-NANDC | R-NANDC |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.3.9 Software Reset Register 9 (SRCR9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|------|---------|---------|---------|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SRT 931 | SRT 930 | SRT 929 | SRT 928 | SRT 927 | SRT 926 | — | — | — | SRT 922 | — | — | SRT 919 | SRT 918 | SRT 917 | SRT 916 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R/W | R/W | R/W | R/W | R/W* | R | R/W* | R/W* | R/W* | R | R/W* | R/W* | R/W* | R/W | R/W* |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SRT 915 | SRT 914 | — | SRT 912 | SRT 911 | SRT 910 | SRT 909 | SRT 908 | SRT 907 | SRT 906 | SRT 905 | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R |

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.31 Assignment of Modules to Bits in SRCR9

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|----------|----------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | I2C-IF0 | I2C-IF0 | I2C-IF0 | I2C-IF0 |
| 30 | I2C-IF1 | I2C-IF1 | I2C-IF1 | I2C-IF1 |
| 29 | I2C-IF2 | I2C-IF2 | I2C-IF2 | I2C-IF2 |
| 28 | I2C-IF3 | I2C-IF3 | I2C-IF3 | I2C-IF3 |
| 27 | I2C-IF4 | I2C-IF4 | I2C-IF4 | I2C-IF4 |
| 26 | IIC-PMIC | IIC-PMIC | IIC-PMIC | IIC-PMIC |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | ADG | ADG | ADG | ADG |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | I2C-IF5 | I2C-IF5 | I2C-IF5 | I2C-IF5 |
| 18 | I2C-IF6 | I2C-IF6 | I2C-IF6 | I2C-IF6 |
| 17 | RPC-IF | RPC-IF | RPC-IF | RPC-IF |
| 16 | CAN-IF0 | CAN-IF0 | CAN-IF0 | CAN-IF0 |
| 15 | CAN-IF1 | CAN-IF1 | CAN-IF1 | CAN-IF1 |
| 14 | CAN-FD | CAN-FD | CAN-FD | CAN-FD |
| 13 | — | — | — | — |
| 12 | GPIO0 | GPIO0 | GPIO0 | GPIO0 |
| 11 | GPIO1 | GPIO1 | GPIO1 | GPIO1 |
| 10 | GPIO2 | GPIO2 | GPIO2 | GPIO2 |
| 9 | GPIO3 | GPIO3 | GPIO3 | GPIO3 |
| 8 | GPIO4 | GPIO4 | GPIO4 | GPIO4 |
| 7 | GPIO5 | GPIO5 | GPIO5 | GPIO5 |
| 6 | GPIO6 | GPIO6 | GPIO6 | GPIO6 |
| 5 | GPIO7 | GPIO7 | GPIO7 | — |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|--------|--------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 4 | — | — | — | — |
| 3 | — | — | — | — |
| 2 | — | — | — | — |
| 1 | — | — | — | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

12.2.3.10 Software Reset Register 10 (SRCR10)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------|-------------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SRT 1015 | SRT 1014 | SRT 1013 | SRT 1012 | SRT 1011 | SRT 1010 | SRT 1009 | SRT 1008 | SRT 1007 | SRT 1006 | SRT 1005 | — | SRT 1003 | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* |

Note: * Writing to a bit is only possible if the product actually has the corresponding module.

Table 12.32 Assignment of Modules to Bits in SRCR10

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|------------|------------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 31 | — | — | — | — |
| 30 | — | — | — | — |
| 29 | — | — | — | — |
| 28 | — | — | — | — |
| 27 | — | — | — | — |
| 26 | — | — | — | — |
| 25 | — | — | — | — |
| 24 | — | — | — | — |
| 23 | — | — | — | — |
| 22 | — | — | — | — |
| 21 | — | — | — | — |
| 20 | — | — | — | — |
| 19 | — | — | — | — |
| 18 | — | — | — | — |
| 17 | — | — | — | — |
| 16 | — | — | — | — |
| 15 | SSI0 | SSI0 | SSI0 | SSI0 |
| 14 | SSI1 | SSI1 | SSI1 | SSI1 |
| 13 | SSI2 | SSI2 | SSI2 | SSI2 |
| 12 | SSI3 | SSI3 | SSI3 | SSI3 |
| 11 | SSI4 | SSI4 | SSI4 | SSI4 |
| 10 | SSI5 | SSI5 | SSI5 | SSI5 |
| 9 | SSI6 | SSI6 | SSI6 | SSI6 |
| 8 | SSI7 | SSI7 | SSI7 | SSI7 |
| 7 | SSI8 | SSI8 | SSI8 | SSI8 |
| 6 | SSI9 | SSI9 | SSI9 | SSI9 |
| 5 | SSI (all)* | SSI (all)* | SSI (all)* | SSI (all)* |

| Bit | RZ/G2M V1.3 | | | |
|-----|-------------|-------------|--------|---------|
| | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 4 | — | — | — | — |
| 3 | — | — | — | I2C-IF7 |
| 2 | — | — | — | — |
| 1 | — | — | — | — |
| 0 | — | — | — | — |

Note: "—" indicates the bit is reserved. Set the read value.

Note: * When the SRT1005 bit is set to 1, the bit of the SRT1015 to SRT1006 is simultaneously set without setting SRT1015 to SRT1006. When any of the SRT1015 to SRT1006 is cleared, SRT1005 bit is cleared.

12.2.4 Software Reset Clearing Register n (SRSTCLRn (n = 0 to 11))

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Each SRSTCLRn is a 32-bit writable register. Each bit is assigned to a module in this chip. Writing 1 to the bit clears the corresponding bit of SRCRn and releases the reset. Write 0 to the reserved bits.

Bit assignment of SRSTCLRn is same with one of SRCRn.

| | | | | | | | | | | | | | | | | |
|----------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SRTCL Rn31 | SRTCL Rn30 | SRTCL Rn29 | SRTCL Rn28 | SRTCL Rn27 | SRTCL Rn26 | SRTCL Rn25 | SRTCL Rn24 | SRTCL Rn23 | SRTCL Rn22 | SRTCL Rn21 | SRTCL Rn20 | SRTCL Rn19 | SRTCL Rn18 | SRTCL Rn17 | SRTCL Rn16 |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | W | W | W | W | W | W | W | W | W | W | W | W | W | W | W | W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SRTCL Rn15 | SRTCL Rn14 | SRTCL Rn13 | SRTCL Rn12 | SRTCL Rn11 | SRTCL Rn10 | SRTCL Rn09 | SRTCL Rn08 | SRTCL Rn07 | SRTCL Rn06 | SRTCL Rn05 | SRTCL Rn04 | SRTCL Rn03 | SRTCL Rn02 | SRTCL Rn01 | SRTCL Rn00 |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | W | W | W | W | W | W | W | W | W | W | W | W | W | W | W | W |

12.2.5 Secure Module Stop Control Register n (SCMSTPCRn, n = 0 to 11)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SCMSTPCRn (n = 0 to 11) are 32-bit readable/writable registers. This register can be accessed only in the secure mode. When write accessed in the non-secure mode, the write operation becomes invalid. When read accessed in the non-secure mode, the returned value becomes H'0000_0000.

| | | | | | | | | | | | | | | | | |
|----------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SCMST Pn31 | SCMST Pn30 | SCMST Pn29 | SCMST Pn28 | SCMST Pn27 | SCMST Pn26 | SCMST Pn25 | SCMST Pn24 | SCMST Pn23 | SCMST Pn22 | SCMST Pn21 | SCMST Pn20 | SCMST Pn19 | SCMST Pn18 | SCMST Pn17 | SCMST Pn16 |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SCMST Pn15 | SCMST Pn14 | SCMST Pn13 | SCMST Pn12 | SCMST Pn11 | SCMST Pn10 | SCMST Pn09 | SCMST Pn08 | SCMST Pn07 | SCMST Pn06 | SCMST Pn05 | SCMST Pn04 | SCMST Pn03 | SCMST Pn02 | SCMST Pn01 | SCMST Pn00 |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---|---------------|-----|---|
| 31 to 0 | SCMSTPnm (n = 0 to 11) (m = 31 to 00) | All 1 | R/W | [Valid access] These bits can be written only in the secure mode. These bits can be read only in the secure mode. [Invalid access] When accessing these registers in the non-secure mode, these bits cannot be written and cannot be read. When all corresponding bits in RMSTPCR, SMSTPCR and SCMSTPCR registers are set to 1, the clock signal for the module is halted. |

Assignment of modules to bits m in SCMSTPCRn is same with one of RMSTPCRn and SMSTPCRn.

12.2.6 Secure Software Reset Access Enable Control Register n (SCSRSTECRn, n = 0 to 11)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SCSRSTECRn (n = 0 to 11) are 32-bit readable/writable registers. This register can be accessed only in the secure mode. When write accessed in the non-secure mode, the write operation becomes invalid. When read accessed in the non-secure mode, the returned value becomes H'0000_0000.

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SCE n31 | SCE n30 | SCE n29 | SCE n28 | SCE n27 | SCE n26 | SCE n25 | SCE n24 | SCE n23 | SCE n22 | SCE n21 | SCE n20 | SCE n19 | SCE n18 | SCE n17 | SCE n16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SCE n15 | SCE n14 | SCE n13 | SCE n12 | SCE n11 | SCE n10 | SCE n09 | SCE n08 | SCE n07 | SCE n06 | SCE n05 | SCE n04 | SCE n03 | SCE n02 | SCE n01 | SCE n00 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--|---------------|-----|---|
| 31 to 0 | SCEnm (n = 0 to 11) (m = 31 to 00) | All 0 | R/W | <p>Access protection to SRCRn and SRSTCLRn (n = 0 to 11) in the viewpoint of the secure mode</p> <p>0: Not check access to SRCRn bit m and SRSTCLRn bit m (m = 31 to 0) in the viewpoint of secure mode. All write access to these bits are valid.</p> <p>1: Check access to SRCRn bit m and SRSTCLRn bit m (m = 31 to 0) are accessed in the secure mode or not.</p> <p>Write access to SRCRnm and SRSTCLRnm is valid only in the secure mode. Write access to SRCRnm and SRSTCLRnm is invalid in the non-secure mode.</p> |

Assignment of modules to bits m in SCSRSTECRn is same with one of SRCRn and SRSTCLRn.

The validity of write access to SRCRn[m] and SRSTCLRn[m] is judged by the flowchart shown in Figure 12.1.

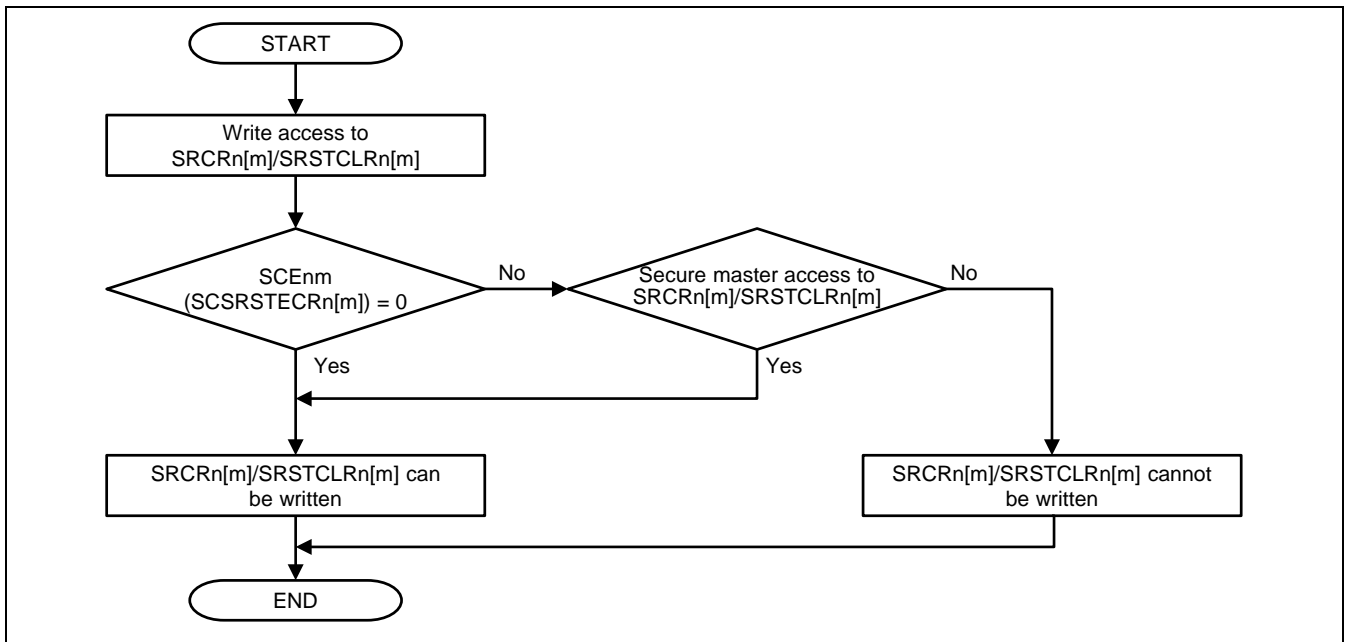


Figure 12.1 The validity check of write access to SRCRn and SRSTCLRn

13. Advanced Power Management Unit for AP-System Core (APMU)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

13.1 Overview

The AP-system core power management unit (APMU) is a module to control power supply and clock supply to the AP-system core.

13.1.1 Features

This module includes the following function

- Controls power supply to each CPU core and L2 cache area.

The APMU includes the following submodule:

- Power supply control unit (PCCU) which controls the power supply to each CPU core and L2 cache area.

13.1.2 External Pins

There are no external pins related with this function.

13.1.3 Register Configuration

Table 13.1 lists the registers of the APMU. All registers listed in this table are cleared by the reset.

Table 13.1 Register Configurations

| | | | | | | Second Generation RZ/G Series Products | | | | |
|--|------------------|-----|-----------------------------|---------------|-------------|---|----------------------------|--------|--------|--|
| Register Name | Abbreviation | R/W | Address | Initial value | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| PCCU registers | | | | | | | | | | |
| Cortex-A53 CPU Wake Up Control Register | CA53WUPCR | R/W | H'E615_1010 / H'E615_4010*1 | H'0000_0000 | 32 | √ | √ | — | √ | |
| Cortex-A53 Power Status Register | CA53PSTR | R | H'E615_1040 / H'E615_4040*1 | H'----_333- | 32 | √ | √ | — | √ | |
| Cortex-A53 CPU0 Power Status Control Register | CA53CPU0CR | R/W | H'E615_1100 / H'E615_4100*1 | H'0000_0000 | 32 | √ | √ | — | √ | |
| Cortex-A53 CPU1 Power Status Control Register | CA53CPU1CR | R/W | H'E615_1110 / H'E615_4110*1 | H'0000_0000 | 32 | √ | √ | — | √ | |
| Cortex-A53 CPU2 Power Status Control Register | CA53CPU2CR | R/W | H'E615_1120 / H'E615_4120*1 | H'0000_0000 | 32 | √ | √ | — | — | |
| Cortex-A53 CPU3 Power Status Control Register | CA53CPU3CR | R/W | H'E615_1130 / H'E615_4130*1 | H'0000_0000 | 32 | √ | √ | — | — | |
| Cortex-A53 Common Power Control Register | CA53CPU0CR CR | R/W | H'E615_1184 / H'E615_4184*1 | H'0000_0000 | 32 | √ | √ | — | √ | |
| Cortex-A57 CPU Wake Up Control Register | CA57WUPCR | R/W | H'E615_2010 / H'E615_4010*2 | H'0000_0000 | 32 | √ | √ | √ | — | |
| Cortex-A57 Power Status Register | CA57PSTR | R | H'E615_2040 / H'E615_4040*2 | H'----_333- | 32 | √ | √ | √ | — | |
| Cortex-A57 CPU0 Power Status Control Register | CA57CPU0CR | R/W | H'E615_2100 / H'E615_4100*2 | H'0000_0000 | 32 | √ | √ | √ | — | |
| Cortex-A57 CPU1 Power Status Control Register | CA57CPU1CR | R/W | H'E615_2110 / H'E615_4110*2 | H'0000_0000 | 32 | √ | √ | √ | — | |
| Cortex-A57 CPU2 Power Status Control Register | CA57CPU2CR | R/W | H'E615_2120 / H'E615_4120*2 | H'0000_0000 | 32 | √ | — | — | — | |
| Cortex-A57 CPU3 Power Status Control Register | CA57CPU3CR | R/W | H'E615_2130 / H'E615_4130*2 | H'0000_0000 | 32 | √ | — | — | — | |
| Cortex-A57 Common Power Control Register | CA57CPU0CR CR | R/W | H'E615_2184 / H'E615_4184*2 | H'0000_0000 | 32 | √ | √ | √ | — | |
| Cortex-A53 Debug Resource Reset Control Register | CA53DBGRC | R/W | H'E615_1180 | H'01F-_----- | 32 | √ | √ | — | √ | |
| Cortex-A57 Debug Resource Reset Control Register | CA57DBGRC | R/W | H'E615_2180 | H'01F-_----- | 32 | √ | √ | √ | — | |

Notes: The register address map "H'E615_4xxx" is to share the same address between registers for Cortex-A57 and registers for Cortex-A53.

1. When the address "H'E615_4xxx" is accessed by Cortex-A53, registers for Cortex-A53 (CA53XXX) are accessed.
2. When the address "H'E615_4xxx" is accessed by Cortex-A57, registers for Cortex-A57 (CA57XXX) are accessed.

13.1.4 Connected Module

Table 13.2 Connected Modules

| Module name | Connected module name | Function of connected module |
|--------------------|------------------------------|---|
| CPG | AP-System Core | Cortex-A53/Cortex-A57 power status control [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] Cortex-A53 power status control [RZ/G2E] Cortex-A57 power status control [RZ/G2N] |
| | SYSC | Cooperate with SYSC for power status control of AP-System Core |

13.2 Register Description

13.2.1 Cortex-A53/Cortex-A57 CPU_n Power Status Control Register (CA53CPU_nCR, CA57CPU_nCR (n = 0 to 3))

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Each RZ/G Series 2nd Generation product provides these registers as shown in Table 13.1.

Cortex-A53/Cortex-A57 CPU_n power status control registers are 32-bit readable/writable registers of PCCU. These registers define the operation of related CPU when the CPU_n core issues WFI instruction. Write these registers while the power of related CPU is ON.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CPUPWR[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|--|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1, 0 | CPUPWR[1:0] | B'00 | R/W | CPU _n PWR Bit Defines the power supply of corresponding CPU _n when the CPU _n issues WFI instruction. B'00: Sleep mode B'11: CoreStandby mode (power shut off of CPU core, L1\$) Other than above: Setting prohibited These bits are automatically cleared to B'00 when the PCCU completes CoreStandby instruction after B'11 is written to these bits. |

13.2.2 Cortex-A53/Cortex-A57 Common Power Control Register (CA53CPUCMCR, CA57CPUCMCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Each RZ/G Series 2nd Generation product provides these registers as shown in Table 13.1.

Cortex-A53/Cortex-A57 common power control registers are 32-bit readable/writable registers. These registers define power supply control of CPU peripheral (SCU and L2 cache controller) of AP-system core (Cortex-A53) and CPU peripheral (SCU and L2 cache controller) of AP-system core (Cortex-A57). The CMPWR bits of this register define the operation when all CPU cores are in the CoreStandby state.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CMPWR[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 3, 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1, 0 | CMPWR[1:0] | B'00 | R/W | CPU Common Power Control Bit Defines the power supply when all of the CPUs in the cluster are in CoreStandby mode (power shut off). B'00: Normal mode, CPU peripheral (SCU and L2 cache controller) ON B'01: Setting prohibited B'10: L2 shutdown mode, CPU peripheral (SCU and L2 cache controller) OFF B'11: Setting prohibited |

[RZ/G2M V1.3] (not RZ/G2M V3.0)
Don't write anything other than B'00.

13.2.3 Cortex-A53/Cortex-A57 CPU Wake Up Control Register (CA53WUPCR, CA57WUPCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Each RZ/G Series 2nd Generation product provides these registers as shown in Table 13.1.

Cortex-A53/Cortex-A57 CPU wake up control registers are 32-bit readable/writable registers of PCCU. These registers request power on of each CPU core. Writing 1 to the CPU_nWUP bit wakes up the corresponding CPU_n, and the CPU_n starts operation from the reset vector. Each bit is automatically cleared to 0 after the wake up procedure is finished.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|-------------|-------------|-------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CPU3 WUP | CPU2 WUP | CPU1 WUP | CPU0 WUP |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

[CA53WUPCR]

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 | CPU3WUP | B'0 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]: CPU3 Wake Up Bit Write access: 0: Ignores. 1: Requests wake up of CPU3 (CPU3 power on). Read access: 0: CPU3 wakeup sequence is completed. 1: CPU3 wakeup sequence is in progress. |
| — | — | B'0 | R | [RZ/G2N, RZ/G2E]: Reserved This bit is always read as 0. The write value should always be 0. |
| 2 | CPU2WUP | B'0 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]: CPU2 Wake Up Bit Write access: 0: Ignores. 1: Requests wake up of CPU2 (CPU2 power on). Read access: 0: CPU2 wakeup sequence is completed. 1: CPU2 wakeup sequence is in progress. |
| — | — | B'0 | R | [RZ/G2N, RZ/G2E]: Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | CPU1WUP | B'0 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]: CPU1 Wake Up Bit Write access: 0: Ignores. 1: Requests wake up of CPU1 (CPU1 power on). Read access: 0: CPU1 wakeup sequence is completed. 1: CPU1 wakeup sequence is in progress. |
| — | | B'0 | R | [RZ/G2N]: Reserved This bit is always read as 0. The write value should always be 0. |
| 0 | CPU0WUP | B'0 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]: CPU0 Wake Up Bit Write access: 0: Ignores. 1: Requests wake up of CPU0 (CPU0 power on). Read access: 0: CPU0 wakeup sequence is completed. 1: CPU0 wakeup sequence is in progress. |
| — | | B'0 | R | [RZ/G2N]: Reserved This bit is always read as 0. The write value should always be 0. |

[CA57WUPCR]

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 | CPU3WUP | B'0 | R/W | [RZ/G2H]: CPU3 Wake Up Bit Write access: 0: Ignores. 1: Requests wake up of CPU3 (CPU3 power on). Read access: 0: CPU3 wakeup sequence is completed. 1: CPU3 wakeup sequence is in progress. |
| — | | B'0 | R | [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]: Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | CPU2WUP | B'0 | R/W | [RZ/G2H]: CPU2 Wake Up Bit Write access: 0: Ignores. 1: Requests wake up of CPU2 (CPU2 power on). Read access: 0: CPU2 wakeup sequence is completed. 1: CPU2 wakeup sequence is in progress. |
| — | | B'0 | R | [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]: Reserved This bit is always read as 0. The write value should always be 0. |
| 1 | CPU1WUP | B'0 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]: CPU1 Wake Up Bit Write access: 0: Ignores. 1: Requests wake up of CPU1 (CPU1 power on). Read access: 0: CPU1 wakeup sequence is completed. 1: CPU1 wakeup sequence is in progress. |
| — | | B'0 | R | [RZ/G2E]: Reserved This bit is always read as 0. The write value should always be 0. |
| 0 | CPU0WUP | B'0 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]: CPU0 Wake Up Bit Write access: 0: Ignores. 1: Requests wake up of CPU0 (CPU0 power on). Read access: 0: CPU0 wakeup sequence is completed. 1: CPU0 wakeup sequence is in progress. |
| — | | B'0 | R | [RZ/G2E]: Reserved This bit is always read as 0. The write value should always be 0. |

13.2.4 Cortex-A53/Cortex-A57 Power Status Register (CA53PSTR, CA57PSTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Each RZ/G Series 2nd Generation product provides these registers as shown in Table 13.1.

Cortex-A53/Cortex-A57 power status registers are 32-bit readable registers of PCCU. These registers show power status of AP-system CPU cores.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|-------------|----|----|----|-------------|----|----|----|-------------|----|----|----|-------------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | — | 0 | 0 | 0 | — | 0 | 0 | 0 | — | 0 | 0 | 0 | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | CPU3ST[1:0] | | — | — | CPU2ST[1:0] | | — | — | CPU1ST[1:0] | | — | — | CPU0ST[1:0] | |
| Initial value: | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | * | * |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

Note: “—” stands for unknown value.

- * CA53PSTR: When (MD7, MD6) = (L, H), the initial value is 0. In the other setting, the initial value is 1. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]
- CA57PSTR: When (MD7, MD6) = (L, L), the initial value is 0. In the other setting, the initial value is 1. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

[CA53PSTR]

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. |
| 28 | — | Undefined | R | Reserved This bit is always read as unknown value. |
| 27 to 25 | — | All 0 | R | Reserved These bits are always read as 0. |
| 24 | — | Undefined | R | Reserved This bit is always read as unknown value. |
| 23 to 21 | — | All 0 | R | Reserved These bits are always read as 0. |
| 20 | — | Undefined | R | Reserved This bit is always read as unknown value. |
| 19 to 17 | — | All 0 | R | Reserved These bits are always read as 0. |
| 16 | — | Undefined | R | Reserved This bit is always read as unknown value. |
| 15, 14 | — | All 0 | R | Reserved These bits are always read as 0. |
| 13, 12 | CPU3ST [1:0] | B'11 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]: CPU3 Status Bit Displays CPU3 status. B'00: Run mode B'11: CoreStandby mode Other than above: Invalid [RZ/G2N, RZ/G2E]: Reserved These bits are always read as B'11. |
| 11, 10 | — | All 0 | R | Reserved These bits are always read as 0. |
| 9, 8 | CPU2ST [1:0] | B'11 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]: CPU2 Status Bit Displays CPU2 status. B'00: Run mode B'11: CoreStandby mode Other than above: Invalid [RZ/G2N, RZ/G2E]: Reserved These bits are always read as B'11. |
| 7, 6 | — | All 0 | R | Reserved These bits are always read as 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|-----------------|---------------|-----|--|
| 5, 4 | CPU1ST [1:0] | B'11 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]: CPU1 Status Bit Displays CPU1 status. B'00: Run mode B'11: CoreStandby mode Other than above: Invalid <hr/> [RZ/G2N]: Reserved These bits are always read as B'11. |
| 3, 2 | — | All 0 | R | Reserved These bits are always read as 0. |
| 1, 0 | CPU0ST [1:0] | * | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]: CPU0 Status Bit Displays CPU0 status. B'00: Run mode B'11: CoreStandby mode Other than above: Invalid <hr/> B'11 |
| | | B'11 | R | [RZ/G2N]: Reserved These bits are always read as B'11. |

[CA57PSTR]

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. |
| 28 | — | Undefined | R | Reserved This bit is always read as unknown value. |
| 27 to 25 | — | All 0 | R | Reserved These bits are always read as 0. |
| 24 | — | Undefined | R | Reserved This bit is always read as unknown value. |
| 23 to 21 | — | All 0 | R | Reserved These bits are always read as 0. |
| 20 | — | Undefined | R | Reserved This bit is always read as unknown value. |
| 19 to 17 | — | All 0 | R | Reserved These bits are always read as 0. |
| 16 | — | Undefined | R | Reserved This bit is always read as unknown value. |
| 15, 14 | — | All 0 | R | Reserved These bits are always read as 0. |
| 13, 12 | CPU3ST [1:0] | B'11 | R | [RZ/G2H]: CPU3 Status Bit Displays CPU3 status. B'00: Run mode B'11: CoreStandby mode Other than above: Invalid [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]: Reserved These bits are always read as B'11. |
| 11, 10 | — | All 0 | R | Reserved These bits are always read as 0. |
| 9, 8 | CPU2ST [1:0] | B'11 | R | [RZ/G2H]: CPU2 Status Bit Displays CPU2 status. B'00: Run mode B'11: CoreStandby mode Other than above: Invalid [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]: Reserved These bits are always read as B'11. |
| 7, 6 | — | All 0 | R | Reserved These bits are always read as 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|-----------------|---------------|-----|---|
| 5, 4 | CPU1ST [1:0] | B'11 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]: CPU1 Status Bit Displays CPU1 status. B'00: Run mode B'11: CoreStandby mode Other than above: Invalid <hr/> [RZ/G2E]: Reserved These bits are always read as B'11. |
| 3, 2 | — | All 0 | R | Reserved These bits are always read as 0. |
| 1, 0 | CPU0ST [1:0] | * | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]: CPU0 Status Bit Displays CPU0 status. B'00: Run mode B'11: CoreStandby mode Other than above: Invalid <hr/> B'11 R [RZ/G2E]: Reserved These bits are always read as B'11. |

Note: “—” stands for unknown value.

- * CA53PSTR: When (MD7, MD6) = (L, H), the initial value is B'00. In the other setting, the initial value is B'11.
[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]
- CA57PSTR: When (MD7, MD6) = (L, L), the initial value is B'00. In the other setting, the initial value is B'11.
[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

13.2.5 Cortex-A53/Cortex-A57 Debug Resource Reset Control Register (CA53DBGRCR, CA57DBGRCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Each RZ/G Series 2nd Generation product provides these registers as shown in Table 13.1.

Cortex-A53/Cortex-A57 debug resource reset control registers are 32-bit readable/writable registers of PCCU. These registers enable the reset requests derived from power shut off to the AP-system CPU cores in the debug mode. When you write to these registers, modify only the target bits you want to change (read-modify-write). In the debug mode, bit 19 must be set to 1 before the AP-system cores first resume from power shut off. In the normal mode (i.e. not debug mode), writing all 1 to bit 19 doesn't disturb normal operation. Therefore, the same code to write these registers can be applied for both debug mode and normal mode. About CA53DBGRCR and CA57DBGRCR, refer to Appendix B.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----------------|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | DBGCP UPREN | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | — | — | 0 | 0 | — | — | 0 | 0 | — | — | 0 | 0 | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | — | B'1 | R | Reserved This bit is always read as 1. The write value should always be 1. |
| 23 | — | B'1 | R | Reserved This bit is always read as 1. The write value should always be 1. |
| 22 | — | B'1 | R | Reserved This bit is always read as 1. The write value should always be 1. |
| 21 | — | B'1 | R | Reserved This bit is always read as 1. The write value should always be 1. |
| 20 | — | B'1 | R | Reserved This bit is always read as 1. The write value should always be 1. |
| 19 | DBGCPUPRE N | B'0 | R/W | Enable the reset request derived from power shut off to CPU Peripheral (SCU and L2 cache controller) in the debug mode. 0: Disables the reset request derived from power shut off to CPU Peripheral (SCU and L2 cache controller) in the debug mode. 1: Enables the reset request derived from power shut off to CPU Peripheral (SCU and L2 cache controller) in the debug mode. |
| 18 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 17, 16 | — | Undefined | R | Reserved These bits are always read as unknown values. The write value should always be the same as read values (read-modify-write). |
| 15, 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13, 12 | — | Undefined | R | Reserved These bits are always read as unknown values. The write value should always be the same as read values (read-modify-write). |
| 11, 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9, 8 | — | Undefined | R | Reserved These bits are always read as unknown values. The write value should always be the same as read values (read-modify-write). |
| 7, 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5, 4 | — | Undefined | R | Reserved These bits are always read as unknown values. The write value should always be the same as read values (read-modify-write). |
| 3, 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1, 0 | — | Undefined | R | Reserved These bits are always read as unknown values. The write value should always be the same as read values (read-modify-write). |

13.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

13.3.1 Power Control

The PCCU manages the CPU power supply when the CPU issues WFI instruction, and recovers from the low power state. Table 13.3 shows power status of CPU.

Table 13.3 CPU Power Mode

| Power Mode | CPU#n Clock | CPU#n Power | CPU Peripheral (SCU and L2 Cache Controller) Power |
|------------------|----------------|----------------|---|
| Run mode | ON | ON | ON |
| Sleep mode | OFF | ON | ON |
| CoreStandby mode | OFF | OFF | ON |
| L2 shutdown mode | OFF | OFF | OFF |

The following is method to enter each mode. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]

Sleep mode

Issue WFI instruction when the CPUPWR bits of CA53CPU_nCR/CA57CPU_nCR are B'00.

CoreStandby mode

Issue WFI instruction when the CPUPWR bits of CA53CPU_nCR/CA57CPU_nCR are B'11.

The CPUPWR bits of CA53CPU_nCR/CA57CPU_nCR are automatically cleared to B'00 when the PCCU completes CoreStandby operation. So the interrupt to CPU_n is masked while CPUPWR bits of CA53CPU_nCR/CA57CPU_nCR are B'11. In the case of cancel, set B'00 to the CPUPWR bits of CA53CPU_nCR/CA57CPU_nCR.

L2 shutdown mode

Issue WFI instruction when all of the conditions below are met:

Other CPU cores in the same CPU cluster are in the CoreStandby mode.

The CMPWR bits of CA53CPUCMCR/CA57CPUCMCR are B'10.

The CPUPWR bits of CA53CPU_nCR/CA57CPU_nCR are B'11.

Run mode

Interrupt and reset of CPU recovers each CPU to the RUN mode.

The following is method to enter each mode. [RZ/G2E]

Sleep mode

Issue WFI instruction when the CPUPWR bits of CA53CPU_nCR are B'00.

CoreStandby mode

Issue WFI instruction when the CPUPWR bits of CA53CPU_nCR are B'11.

The CPUPWR bits of CA53CPU_nCR are automatically cleared to B'00 when the PCCU completes CoreStandby operation. So the interrupt to CPU_n is masked while CPUPWR bits of CA53CPU_nCR are B'11. In the case of cancel, set B'00 to the CPUPWR bits of CA53CPU_nCR.

L2 shutdown mode

Issue WFI instruction when all of the conditions below are met:

Other CPU cores in the same CPU cluster are in the CoreStandby mode.

The CMPWR bits of CA53CPUCMCR are B'10.

The CPUPWR bits of CA53CPUnCR are B'11.

Run mode

Interrupt and reset of CPU recovers each CPU to the RUN mode.

The following is method to enter each mode. [RZ/G2N]

Sleep mode

Issue WFI instruction when the CPUPWR bits of CA57CPUnCR are B'00.

CoreStandby mode

Issue WFI instruction when the CPUPWR bits of CA57CPUnCR are B'11.

The CPUPWR bits of CA57CPUnCR are automatically cleared to B'00 when the PCCU completes CoreStandby operation. So the interrupt to CPUT is masked while CPUPWR bits of CA57CPUnCR are B'11. In the case of cancel, set B'00 to the CPUPWR bits of CA57CPUnCR.

L2 shutdown mode

Issue WFI instruction when all of the conditions below are met:

Other CPU cores in the same CPU cluster are in the CoreStandby mode.

The CMPWR bits of CA57CPUCMCR are B'10.

The CPUPWR bits of CA57CPUnCR are B'11.

Run mode

Interrupt and reset of CPU recovers each CPU to the RUN mode.

14. System Controller (SYSC)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

14.1 Overview

The SYSC module controls the power supply to CPUs (Cortex-A57, Cortex-A53), 3D-Graphics Engine, Video Processing domain (A3VP), VC domain (Video Codec domain, including A3VC, A2VC0, and A2VC1). When those modules are not in use, shutting off power of the module lowers leakage power than just stopping clock supply.

Refer to 14.3.1 for modules in the various power domains.

14.1.1 Features

- Control power-shutoff and resume sequence of power domains.
- Support shutoff and resumption of power supply, for power domains. Refer to Table 14.4 Power Domain.
- During power-shutoff status of each power domain, clock supply for the domain is stopped, reset signal is asserted, and output signals of the domain is fixed by the isolation cells.
- Shutting off sequence of power domain for Cortex-A57 CPUs (CPU0 to CPU3), Cortex-A53 CPUs (CPU0 to CPU3) is initiated by WFI instructions.
- Resuming sequence of the power domain for Cortex-A57 CPUs (CPU0 to CPU3), Cortex-A53 CPUs (CPU0 to CPU3) is controlled by interrupts to each domain, or register access to APMU module (in case of Cortex-A57 and Cortex-A53).
- Shutting off and resuming sequence of other power domains are initiated by register access in this module.

14.1.2 Block Diagram

Figure 14.1 to 14.3 shows block diagrams of the SYSC.

Connected to the internal peripheral bus (the APB bus), the SYSC operates in coordination with the CPG, the RESET module, the CPU, and the debugger; it contains a module that controls the objects of power shutoff.

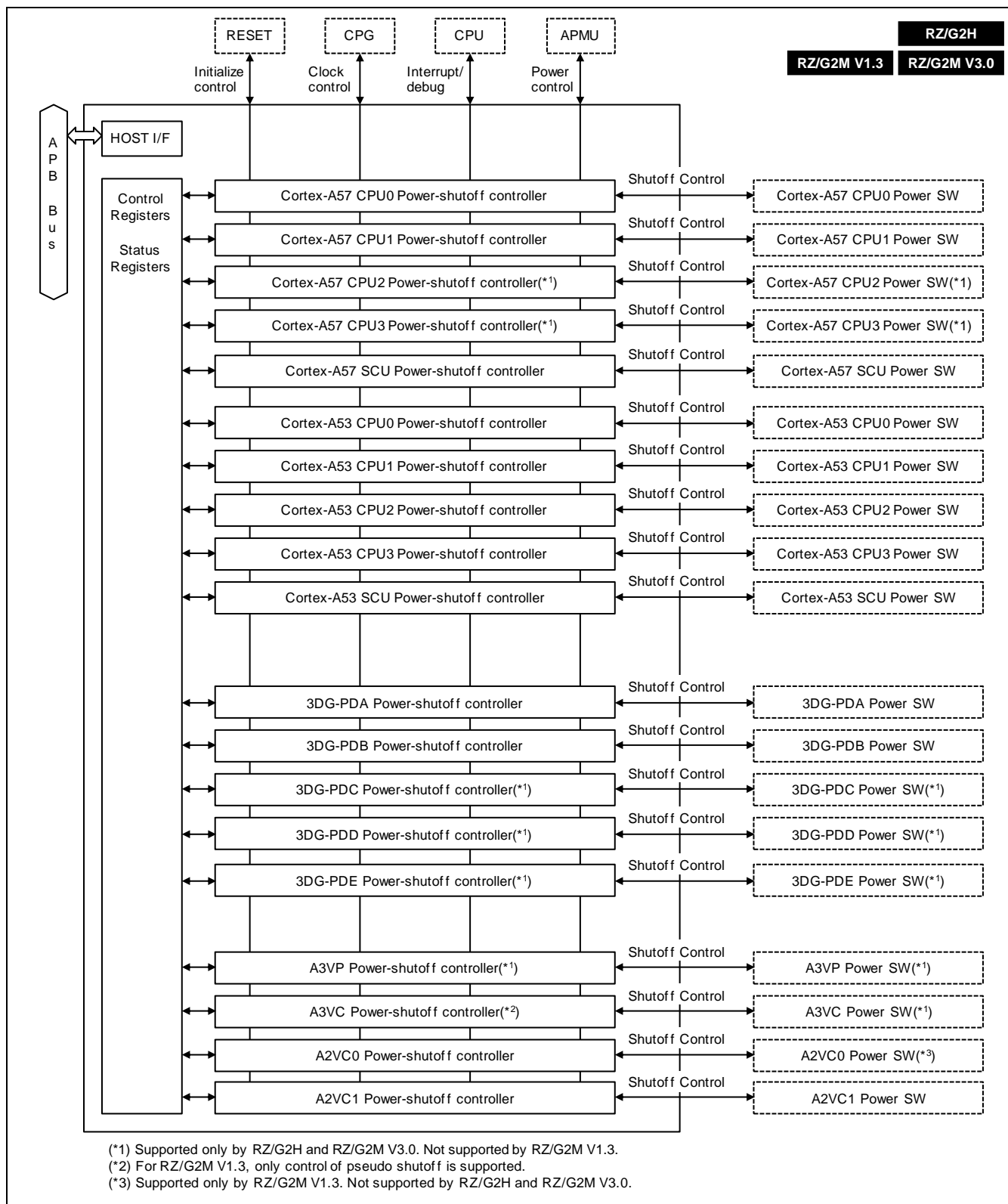


Figure 14.1 Block Diagram [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]

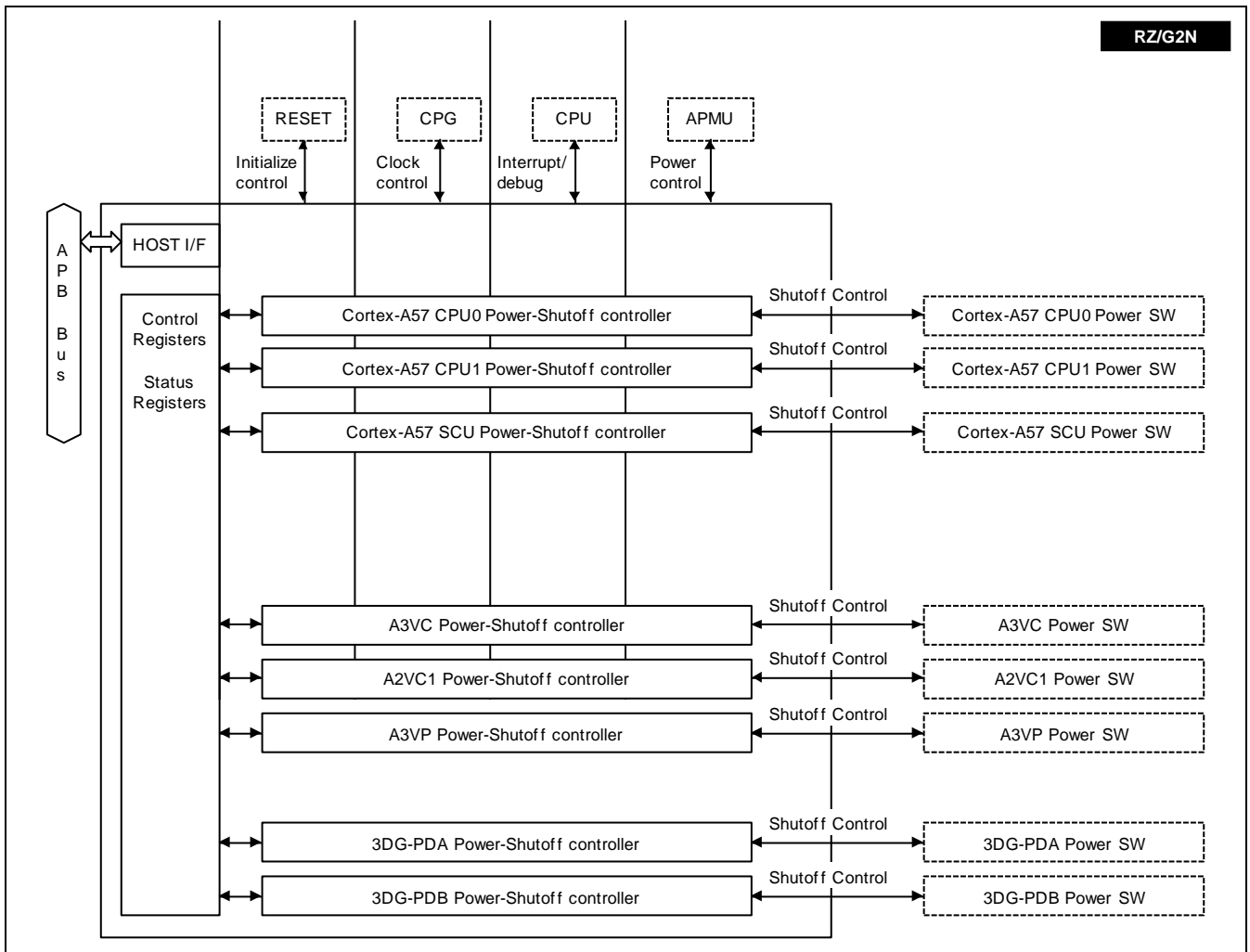


Figure 14.2 Block Diagram [RZ/G2N]

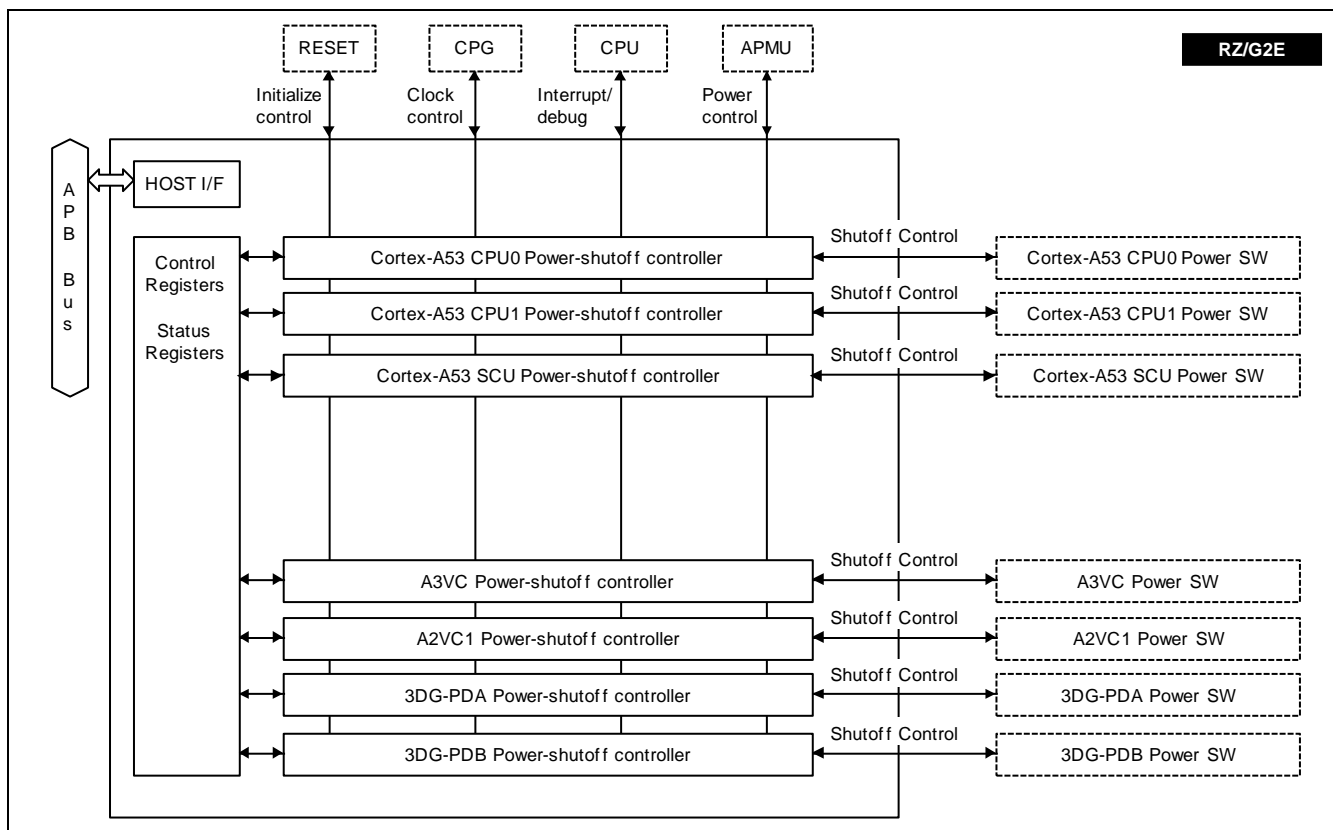


Figure 14.3 Block Diagram [RZ/G2E]

14.1.3 External Pins

The SYSC does not have external pins.

14.1.4 Register Configuration

Table 14.1 to 14.2 list the SYSC registers.

Base-address of SYSC is H'E618_0000.

Do not write to addresses other than those listed below, otherwise normal operation cannot be guaranteed. Values read from other addresses are undefined.

[Legend]

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R: Read-only. The write value should always be the initial value.

—/WB: Write-only. The read value is undefined.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

Table 14.1 List of Registers (Common Registers)

| Register Name | Abbreviation | R/W | Offset Address | Initial Value | Access size | Second Generation RZ/G Series Products | | | | |
|--|--------------|------|----------------|-----------------|-------------|---|-------------|-------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SYSC Status Register | SYSCSR | R | H'0000 | H'0000_0003 | 32 | √ | √ | √ | √ | √ |
| Interrupt Status Register | SYSCISR | R | H'0004 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Interrupt Status Clear Register | SYSCISCR | —/WB | H'0008 | — | 32 | √ | √ | √ | √ | √ |
| Interrupt Enable Register | SYSCIER | R/W | H'000C | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Interrupt Mask Register | SYSCIMR | R/W | H'0010 | H'0131_11EF (*) | 32 | √ | √ | √ | √ | √ |
| Cortex-A57 Wake Up Mask Register | WUPMSKCA57 | R/W | H'0014 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Cortex-A53 Wake Up Mask Register | WUPMSKCA53 | R/W | H'0018 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| External Event Request Status Register | SYSCEERSR | R | H'0020 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| External Event Request Status Clear Register | SYSCEERSCR | —/WB | H'0024 | — | 32 | √ | √ | √ | √ | — |

**Second Generation
RZ/G Series Products**

| Register Name | Abbreviation | R/W | Offset Address | Initial Value | Access size | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---|--------------|------|----------------|---------------|-------------|--------|-------------|-------------|--------|--------|
| External Event Request Status Enable Register | SYSCEERSER | R/W | H'0028 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| External Event Request Status Register 2 | SYSCEERSR2 | R | H'002C | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| External Event Request Status Clear register 2 | SYSCEERSCR2 | —/WB | H'0030 | — | 32 | √ | √ | √ | √ | √ |
| External Event Request Status Enable Register 2 | SYSCEERSER2 | R/W | H'0034 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| External Event Request Status Register 3 | SYSCEERSR3 | R | H'0040 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| External Event Request Status Clear register 3 | SYSCEERSCR3 | —/WB | H'0044 | — | 32 | √ | √ | √ | √ | √ |
| External Event Request Status Enable Register 3 | SYSCEERSER3 | R/W | H'0048 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| External Request Mask Register | SYSCEXTMASK | R/W | H'02F8 | H'0000_0000 | 32 | √ | — | √ | √ | √ |

Note: * Initial value of SYSCIMR will differ for each product. For detail, refer to 14.2.1.5 Interrupt Mask Register (SYSCIMR).

Table 14.2 List of Registers (Power Control Registers for each domain)

| | | | | | | Second Generation RZ/G Series Products | | | | |
|---|--------------|------|----------------|---------------|-------------|---|-------------|-------------|--------|--------|
| Register Name | Abbreviation | R/W | Offset Address | Initial Value | Access size | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Power status register 0 (Cortex-A57) | PWRSR0 | R | H'0080 | H'0000_001E* | 32 | √ | √ | √ | √ | — |
| Power shutoff status register 0 (Cortex-A57) | PWROFFSR0 | R | H'0088 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power resume status register 0 (Cortex-A57) | PWRONSR0 | R | H'0090 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power shutoff/resume error register 0 (Cortex-A57) | PWRER0 | R | H'0094 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power pseudo shutoff register 0 (Cortex-A57) | PWRPSEU0 | R/W | H'00B8 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power status register 2 (3DG) | PWRSR2 | R | H'0100 | H'0000_001F | 32 | √ | √ | √ | √ | √ |
| Power shutoff control register 2 (3DG) | PWROFFCR2 | —/WB | H'0104 | — | 32 | √ | √ | √ | √ | √ |
| Power shutoff status register 2 (3DG) | PWROFFSR2 | R | H'0108 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Power resume control register 2 (3DG) | PWRONCR2 | —/WB | H'010C | — | 32 | √ | √ | √ | √ | √ |
| Power resume status register 2 (3DG) | PWRONSR2 | R | H'0110 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Power shutoff/resume error register 2 (3DG) | PWRER2 | R | H'0114 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Power pseudo shutoff register 2 (3DG) | PWRPSEU2 | R/W | H'0138 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Power status register 3 (Cortex-A53 SCU) | PWRSR3 | R | H'0140 | H'0000_0010* | 32 | √ | √ | √ | — | √ |
| Power shutoff control register 3 (Cortex-A53 SCU) | PWROFFCR3 | —/WB | H'0144 | — | 32 | √ | √ | √ | — | √ |
| Power shutoff status register 3 (Cortex-A53 SCU) | PWROFFSR3 | R | H'0148 | H'0000_0000 | 32 | √ | √ | √ | — | √ |
| Power resume control register 3 (Cortex-A53 SCU) | PWRONCR3 | —/WB | H'014C | — | 32 | √ | √ | √ | — | √ |
| Power resume status register 3 (Cortex-A53 SCU) | PWRONSR3 | R | H'0150 | H'0000_0000 | 32 | √ | √ | √ | — | √ |
| Power shutoff/resume error register 3 (Cortex-A53 SCU) | PWRER3 | R | H'0154 | H'0000_0000 | 32 | √ | √ | √ | — | √ |
| Power pseudo shutoff register 3 (Cortex-A53 SCU) | PWRPSEU3 | R/W | H'0178 | H'0000_0000 | 32 | √ | √ | √ | — | √ |
| Power isolation error detection register 3 (Cortex-A53 SCU) | PWRISOER3 | R/W | H'017C | H'0000_0000 | 32 | √ | √ | √ | — | √ |

| | | | | | | Second Generation RZ/G Series Products | | | | |
|--|--------------|------|----------------|---------------|-------------|---|-------------|-------------|--------|--------|
| Register Name | Abbreviation | R/W | Offset Address | Initial Value | Access size | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Power status register 4 (A3IR) | PWRSR4 | R | H'0180 | H'0000_0001* | 32 | √ | √ | √ | √ | — |
| Power shutoff control register 4 (A3IR) | PWROFFCR4 | —/WB | H'0184 | — | 32 | √ | √ | √ | √ | — |
| Power shutoff status register 4 (A3IR) | PWROFFSR4 | R | H'0188 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power resume control register 4 (A3IR) | PWRONCR4 | —/WB | H'018C | — | 32 | √ | √ | √ | √ | — |
| Power resume status register 4 (A3IR) | PWRONSR4 | R | H'0190 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power shutoff/resume error register 4 (A3IR) | PWRER4 | R | H'0194 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power pseudo shutoff register 4 (A3IR) | PWRPSEU4 | R/W | H'01B8 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power isolation error detection register 4 (A3IR) | PWRISOER4 | R/W | H'01BC | H'0000_0000 | 32 | √ | √ | √ | — | — |
| Power status register 5 (Cortex-A57 SCU) | PWRSR5 | R | H'01C0 | H'0000_0010* | 32 | √ | √ | √ | √ | — |
| Power shutoff control register 5 (Cortex-A57 SCU) | PWROFFCR5 | —/WB | H'01C4 | — | 32 | √ | √ | √ | √ | — |
| Power shutoff status register 5 (Cortex-A57 SCU) | PWROFFSR5 | R | H'01C8 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power resume control register 5 (Cortex-A57 SCU) | PWRONCR5 | —/WB | H'01CC | — | 32 | √ | √ | √ | √ | — |
| Power resume status register 5 (Cortex-A57 SCU) | PWRONSR5 | R | H'01D0 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power shutoff/resume error register 5 (Cortex-A57 SCU) | PWRER5 | R | H'01D4 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power pseudo shutoff register 5 (Cortex-A57 SCU) | PWRPSEU5 | R/W | H'01F8 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power status register 6 (Cortex-A53 CPU) | PWRSR6 | R | H'0200 | H'0000_001E* | 32 | √ | √ | √ | — | √ |
| Power shutoff status register 6 (Cortex-A53 CPU) | PWROFFSR6 | R | H'0208 | H'0000_0000 | 32 | √ | √ | √ | — | √ |
| Power resume status register 6 (Cortex-A53 CPU) | PWRONSR6 | R | H'0210 | H'0000_0000 | 32 | √ | √ | √ | — | √ |
| Power shutoff/resume error register 6 (Cortex-A53 CPU) | PWRER6 | R | H'0214 | H'0000_0000 | 32 | √ | √ | √ | — | √ |

| | | | | | | Second Generation RZ/G Series Products | | | | |
|--|--------------|------|----------------|---------------|-------------|---|-------------|-------------|--------|--------|
| Register Name | Abbreviation | R/W | Offset Address | Initial Value | Access size | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Power pseudo shutoff register 6 (Cortex-A53 CPU) | PWRPSEU6 | R/W | H'0238 | H'0000_0000 | 32 | √ | √ | √ | — | √ |
| Power status register 8 (A3VP) | PWRSR8 | R | H'0340 | H'0000_0001* | 32 | √ | √ | √ | √ | — |
| Power shutoff control register 8 (A3VP) | PWROFFCR8 | —/WB | H'0344 | — | 32 | √ | √ | √ | √ | — |
| Power shutoff status register 8 (A3VP) | PWROFFSR8 | R | H'0348 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power resume control register 8 (A3VP) | PWRONCR8 | —/WB | H'034C | — | 32 | √ | √ | √ | √ | — |
| Power resume status register 8 (A3VP) | PWRONSR8 | R | H'0350 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power shutoff/resume error register 8 (A3VP) | PWRER8 | R | H'0354 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power pseudo shutoff register 8 (A3VP) | PWRPSEU8 | R/W | H'0378 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Power status register 9 (A3VC) | PWRSR9 | R | H'0380 | H'0000_0001* | 32 | √ | √ | √ | √ | √ |
| Power shutoff control register 9 (A3VC) | PWROFFCR9 | —/WB | H'0384 | — | 32 | √ | √ | √ | √ | √ |
| Power shutoff status register 9 (A3VC) | PWROFFSR9 | R | H'0388 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Power resume control register 9 (A3VC) | PWRONCR9 | —/WB | H'038C | — | 32 | √ | √ | √ | √ | √ |
| Power resume status register 9 (A3VC) | PWRONSR9 | R | H'0390 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Power shutoff/resume error register 9 (A3VC) | PWRER9 | R | H'0394 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Power pseudo shutoff register 9 (A3VC) | PWRPSEU9 | R/W | H'03B8 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Power status register 10 (A2VC) | PWRSR10 | R | H'03C0 | H'0000_0003 | 32 | √ | √ | √ | √ | √ |
| Power shutoff control register 10 (A2VC) | PWROFFCR10 | —/WB | H'03C4 | — | 32 | √ | √ | √ | √ | √ |
| Power shutoff status register 10 (A2VC) | PWROFFSR10 | R | H'03C8 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Power resume control register 10 (A2VC) | PWRONCR10 | —/WB | H'03CC | — | 32 | √ | √ | √ | √ | √ |
| Power resume status register 10 (A2VC) | PWRONSR10 | R | H'03D0 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Power shutoff/resume error register 10 (A2VC) | PWRER10 | R | H'03D4 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Power pseudo shutoff register 10 (A2VC) | PWRPSEU10 | R/W | H'03F8 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |

Note: * Initial value of this register depends on mode pin setting. Refer to 14.3.2. Initial state of power domains for detail.

14.1.5 Connected module

Table 14.3 Connected modules

| Module name | Connected module name | Function of connected module |
|-------------|--|--|
| SYSC | Clock Pulse Generator (CPG) | Provide clock to SYSC and PSO (Power Shut Off) target modules. Based on SYSC's request, clocks for PSO target module will be stopped when they are in PSO state. |
| | Module Standby, Software Reset | Assert reset to SYSC and PSO target module. Based on SYSC's request, reset for PSO target module will be asserted when they are in PSO state. |
| | Advanced Power Management Unit for AP-System Core (APMU) | Activate power resume sequence for Arm CPUs. |
| | AP-System Core (Cortex-A57, Cortex-A53), 3D-Graphics Engine, Video Signal Processor (VSP2), Fine Display Processor (FDP1), FCPV, FCPF, FCPCS, FCPCI, IPMMU (VC), Video Codec Processor (VCP4, H.264 version), Video Decoding Processor for inter-device video transfer (iVDP1C), and Video Codec Processor (VCP4, Full Version). | Target of PSO. |
| | Interrupt Controller (INTC-SYS) | Report interrupt signal. |

14.2 Register Description

Explanation of abbreviation of register

Initial value: Value of the register after power-on reset

—: Undefined value

R/W: The bit or field is readable and writable.

R/WC0: The bit or field is readable and writable. Writing 0 to the bit initializes the bit. Writing 1 to the bit is ignored.

R/WC1: The bit or field is readable and writable. Writing 1 to the bit initializes the bit. Writing 0 to the bit is ignored.

R: The bit or field is readable only. When writing to the register, write 0 to it.

—/WB: The bit or field is writable. Note that values read from write-only bits are not guaranteed.

14.2.1 Common Registers and Arm CPU related registers

Common registers are used to for specifying SYSC status, and interrupts generation control. Arm CPU related registers are used for controlling shutting off/resuming of Arm CPUs by interrupts or external events.

Each RZ/G product supports their specific power-domains. Refer to “14.3.1 Power domain structure”, for supported power-domains by each product. In following registers, bits for power-domains are defined as reserved bits, if corresponding power-domain is not supported by the product, if special notice is not written. In this case, those bits are defined as reserved bits, and read value for those bits are unknown, and write value for those bits should be 0.

14.2.1.1 SYSC Status Register (SYSCSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | PONE NB | POFF ENB |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. |
| 1 | PONENB | B'1 | R | This bit indicates whether the SYSC is ready to accept power resume requests. Any power resume request issued when the SYSC is disabled from accepting power resume requests will be ignored. 1: Power resume request is acceptable. 0: Power resume request is not accepted. |
| 0 | POFFENB | B'1 | R | This bit indicates whether the SYSC is ready to accept power shutoff requests. Any power shutoff request issued when the SYSC is disabled from accepting power shutoff requests will be ignored. 1: Power shutoff request is acceptable. 0: Power shutoff request is not acceptable. |

14.2.1.2 Interrupt Status Register (SYSCISR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|------|----|----------|----|-------|-------|---------------|----|-------|----------|---------------|-------|-------|-------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | A2VC1 | A2VC0 | — | — | 3DG-E | CA53-SCU | 3DG-D | 3DG-C | 3DG-B | 3DG-A | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | A3VC | — | CA57-SCU | — | — | A3VP | CA53-CPU[3:0] | | | — | CA57-CPU[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------|-------------------------|---------------------|-----|---|
| 31 to 27 | — | All 0 | R | Reserved These bits are always read as 0. |
| 26 to 25 | A2VC1, A2VC0 | B'0, B'0 | R | Indicates A2VC1/0 power-processing status. This bit is set to 1 when the related bit in the interrupt enable register is 1. If the related bit in the interrupt enable register is 0, this bit is not set to 1 even when the power shutoff or power resume processing is completed. An interrupt request is issued when the related bit in the interrupt mask register is 0 and this bit is 1. 0: Power shutoff or power resume processing of A2VC1/0 is not completed. 1: Power shutoff or power resume processing of A2VC1/0 is completed. [RZ/G2H, RZ/G2M V3.0, RZ/G2N and RZ/G2E] A2VC0: Reserved This bit is always read as 0. |
| 24 to 23 | — | All 0 | R | Reserved |
| 22, 20 to 19 | 3DG-E 3DG-D 3DG-C | B'0, B'0, B'0 | R | Indicates 3DG-C/D/E power-processing status. [3DG-C/D/E are supported by RZ/G2H.] This bit is set to 1 when the related bit in the interrupt enable register is 1. If the related bit in the interrupt enable register is 0, this bit is not set to 1 even when the power shutoff or power resume processing is completed. An interrupt request is issued when the related bit in the interrupt mask register is 0 and this bit is 1. 0: Power shutoff or power resume processing of 3DG-C/D/E is not completed. 1: Power shutoff or power resume processing of 3DG-C/D/E is completed. 3DG-C/D/E [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E] Reserved These bits are always read as 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 18 to 17 | 3DG-B 3DG-A | All 0 | R | <p>Indicates 3DG-A/B power-processing status.</p> <p>This bit is set to 1 when the related bit in the interrupt enable register is 1. If the related bit in the interrupt enable register is 0, this bit is not set to 1 even when the power shutoff or power resume processing is completed. An interrupt request is issued when the related bit in the interrupt mask register is 0 and this bit is 1.</p> <p>0: Power shutoff or power resume processing of 3DG-A/B is not completed. 1: Power shutoff or power resume processing of 3DG-A/B is completed.</p> |
| 21 | CA53-SCU | B'0 | R | <p>Indicates Cortex-A53 power-processing status. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2E]</p> <p>This bit is set to 1 when the related bit in the interrupt enable register is 1. If the related bit in the interrupt enable register is 0, this bit is not set to 1 even when the power shutoff or power resume processing is completed. An interrupt request is issued when the related bit in the interrupt mask register is 0 and this bit is 1.</p> <p>0: Power shutoff or power resume processing of Cortex-A53 SCU is not completed. 1: Power shutoff or power resume processing of Cortex-A53 SCU is completed.</p> <p>[RZ/G2N] Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 16 to 15 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0.</p> |
| 14 | A3VC | B'0 | R | <p>Indicates A3VC power-processing status.</p> <p>This bit is set to 1 when the related bit in the interrupt enable register is 1. If the related bit in the interrupt enable register is 0, this bit is not set to 1 even when the power shutoff or power resume processing is completed. An interrupt request is issued when the related bit in the interrupt mask register is 0 and this bit is 1.</p> <p>0: Power shutoff or power resume processing of A3VC is not completed. 1: Power shutoff or power resume processing of A3VC is completed.</p> |
| 13 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0.</p> |
| 12 | CA57-SCU | B'0 | R | <p>Indicates Cortex-A57 SCU power-processing status. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N]</p> <p>This bit is set to 1 when the CA57-SCU bit in the interrupt enable register is 1 upon completion of the Cortex-A57 SCU power shutoff or power resume processing. If the CA57-SCU bit in the interrupt enable register is 0, this bit is not set to 1 even when the Cortex-A57 SCU power shutoff or power resume processing is completed. An interrupt request is issued when the CA57-SCU bit in the interrupt mask register is 0 and this bit is 1.</p> <p>0: Power shutoff or power resume processing of Cortex-A57 SCU is not completed. 1: Power shutoff or power resume processing of Cortex-A57 SCU is completed.</p> |
| 11 to 10 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------|---------------|-----|---|
| 9 | A3VP | B'0 | R | <p>Indicates A3VP power-processing status.</p> <p>[RZ/G2H and RZ/G2N]</p> <p>This bit is set to 1 when the A3VP bit in the interrupt enable register is 1 upon completion of the A3VP power shutoff or power resume processing. If the A3VP bit in the interrupt enable register is 0, this bit is not set to 1 even when the A3VP power shutoff or power resume processing is completed. An interrupt request is issued when the A3VP bit in the interrupt mask register is 0 and this bit is 1.</p> <p>0: Power shutoff or power resume processing of A3VP is not completed. 1: Power shutoff or power resume processing of A3VP is completed.</p> <p>[RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2E]</p> <p>Reserved</p> <p>This bit is always read as 0.</p> |
| 8 to 5 | CA53-CPU[3:0] | H'0 | R | <p>Indicates Cortex-A53 CPU3-0 power-processing status.</p> <p>[RZ/G2H, RZ/G2M V1.3 and RZ/G2M V3.0]</p> <p>[RZ/G2E only for Cortex-A53 CPU[0] and Cortex-A53 CPU[1]]</p> <p>This bit is set to 1 when the CA53-CPU [3:0] bit in the interrupt enable register is 1 upon completion of the Cortex-A53 core CPU3-0 power shutoff or power resume processing. If the CA53-CPU[3:0] bit in the interrupt enable register is 0, this bit is not set to 1 even when power shutoff or power resume processing for the Cortex-A53 core CPU3-0 is completed. An interrupt request is issued when the CA53-CPU[3:0] bit in the interrupt mask register is 0 and this bit is 1.</p> <p>0: Power shutoff or power resume processing of Cortex-A53 CPU3-0 is not completed. 1: Power shutoff or power resume processing of Cortex-A53-CPU3 0 is completed.</p> <p>[RZ/G2N]</p> <p>Reserved</p> <p>These bits are always read as 0.</p> <p>[RZ/G2E]</p> <p>CPU[3:2] reserved</p> <p>These bits are always read as 0.</p> |
| 4 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------|---------------|-----|--|
| 3 to 0 | CA57-CPU[3:0] | H'0 | R | <p>Indicates Cortex-A57 CPU3-0 power-processing status.</p> <p>[RZ/G2H]</p> <p>[RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N, only for Cortex-A57 CPU[0] and CA57-CPU[1]]</p> <p>This bit is set to 1 when the CA57-CPU[3:0] bit in the interrupt enable register is 1 upon completion of the Cortex-A57 core CPU3-0 power shutoff or power resume processing. If the CA57-CPU[3:0] bit in the interrupt enable register is 0, this bit is not set to 1 even when power shutoff or power resume processing for the Cortex-A57 core CPU3-0 is completed. An interrupt request is issued when the CA57-CPU[3:0] bit in the interrupt mask register is 0 and this bit is 1.</p> <p>0: Power shutoff or power resume processing of Cortex-A57 CPU3-0 is not complete.</p> <p>1: Power shutoff or power resume processing of Cortex-A57 CPU3-0 is complete.</p> <p>Note: Bits for CPU3 and CPU2 are valid for RZ/G2H. For other product, those bits are reserved. Read value for those bits are unknown.</p> <p>[RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0.</p> <p>[RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N]</p> <p>CPU[3:2] reserved</p> <p>These bits are always read as 0.</p> |

14.2.1.3 Interrupt Status Clear Register (SYSCISCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|---------|------|------|------|----------|------|-------|-------|---------------|------|-------|----------|---------------|-------|-------|-------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | A2VC1 | A2VC0 | — | — | 3DG-E | CA53-SCU | 3DG-D | 3DG-C | 3DG-B | 3DG-A | — |
| Initial | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | A3VC | — | CA57-SCU | — | — | A3VP | CA53-CPU[3:0] | | | — | CA57-CPU[3:0] | | | | |
| Initial | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|------|---|
| 31 to 27 | — | — | —/WB | Reserved These bits are always read as 0. |
| 26 to 25 | A2VC1 A2VC0 | — | —/WB | Specify clear of A2VC1/0 interrupt status. Writing 1 to this bit clears the same bit in the interrupt status register (SYSCISR) to 0. Writing 0 to this bit is ignored. [RZ/G2H, RZ/G2M V3.0, RZ/G2N and RZ/G2E] A2VC0: Reserved This bit is always read as 0. |
| 24 to 23 | — | — | —/WB | [RZ/G2E] Reserved This bit is always read as 0. |
| 21 | CA53-SCU | — | —/WB | Specify clear of Cortex-A53 SCU interrupt status. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2E] Writing 1 to this bit clears the same bit in the interrupt status register (SYSCISR) to 0. Writing 0 to this bit is ignored. [RZ/G2N] Reserved This bit is always read as 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------|---|---------------|------|--|
| 22, 20 to 17 | 3DG-E 3DG-D 3DG-C 3DG-B 3DG-A | — | —/WB | Specify clear of 3DG-E/D/C/B/A interrupt status. [3DG-A/B/C/D/E are supported by RZ/G2H.] [3DG-A/B are supported by RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E and RZ/G2N.] Writing 1 to this bit clears the same bit in the interrupt status register (SYSCISR) to 0. Writing 0 to this bit is ignored. [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E] 3DG-E/D/C Reserved These bits are always read as 0. [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E] 3DG-E: Reserved The write value should always be 0. |
| 16 to 15 | — | — | —/WB | Reserved The write value should always be 0. |
| 14 | A3VC | — | —/WB | Specify clear of A3VC interrupt status. Writing 1 to this bit clears the same bit in the interrupt status register (SYSCISR) to 0. Writing 0 to this bit is ignored. |
| 13 | — | — | —/WB | Reserved This bit is always read as 0. |
| 12 | CA57-SCU | — | —/WB | Specify clear of Cortex-A57-SCU interrupt status. [RZ/G2H, RZ/G2N, RZ/G2M V1.3 and RZ/G2M V3.0] Writing 1 to this bit clears the same bit in the interrupt status register (SYSCISR) to 0. Writing 0 to this bit is ignored. [RZ/G2E] Reserved This bit is always read as 0. |
| 11 to 10 | — | — | —/WB | Reserved The write value should always be 0. |
| 9 | A3VP | — | —/WB | Specify clear of A3VP interrupt status. [RZ/G2H and RZ/G2N] Writing 1 to this bit clears the same bit in the interrupt status register (SYSCISR) to 0. Writing 0 to this bit is ignored. [RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2E] Reserved This bit is always read as 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|--|---------------|------|---|
| 8 to 5 | CA53-CPU3 CA53-CPU2 CA53-CPU1 CA53-CPU0 | — | —/WB | Specify clear of CA53-CPU3/2/1/0 interrupt status. [RZ/G2H, RZ/G2M V1.3, and RZ/G2M V3.0] [RZ/G2E only for CA53-CPU[0] and CA53-CPU[1]] Writing 1 to this bit clears the same bit in the interrupt status register (SYSCISR) to 0. Writing 0 to this bit is ignored. [RZ/G2N] Reserved These bits are always read as 0. [RZ/G2E] CA53-CPU3/2 reserved The write value should always be 0. |
| 4 | — | — | —/WB | Reserved The write value should always be 0. |
| 3 to 0 | CA57-CPU3 CA57-CPU2 CA57-CPU1 CA57-CPU0 | — | —/WB | Specify clear of CA57-CPU3/2/1/0 interrupt status. [RZ/G2H] [RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N only for CA57-CPU[0] and CA57-CPU[1]] Writing 1 to this bit clears the same bit in the interrupt status register (SYSCISR) to 0. Writing 0 to this bit is ignored. Note: Bits for CPU3 and CPU2 are valid for RZ/G2H. For other product, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0. [RZ/G2E] Reserved These bits are always read as 0. |

14.2.1.4 Interrupt Enable Register (SYSCIER)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|------|-----|----------|-----|-------|-------|---------------|-----|-------|----------|---------------|-------|-------|-------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | A2VC1 | A2VC0 | — | — | 3DG-E | CA53-SCU | 3DG-D | 3DG-C | 3DG-B | 3DG-A | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | A3VC | — | CA57-SCU | — | — | A3VP | CA53-CPU[3:0] | | | — | CA57-CPU[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 27 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 26 to 25 | A2VC1 A2VC0 | B'0, B'0 | R/W | Specifies that status update of interrupt enable register by A2VC1/0. By setting those bits, same bit in the interrupt status register (SYSCISR) is enabled to be updated. 0: SYSCISR is not updated when power shutoff or power resume processing is complete. 1: SYSCISR is updated when power shutoff or power resume processing is complete. [RZ/G2H, RZ/G2M V3.0, RZ/G2N and RZ/G2E] A2VC0: Reserved This bit is always read as 0. The write value should always be 0. |
| 24 to 23 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 | CA53-SCU | B'0 | R/W | Specifies that status update of interrupt enable register by Cortex-A53 SCU. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2E] By setting this bit, interrupt in same bit in the interrupt status register (SYSCISR) is enabled. 0: SYSCISR is not updated when power shutoff or power resume processing is complete. 1: SYSCISR is updated when power shutoff or power resume processing is complete. [RZ/G2N] Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----------------|---|-------------------------------------|-----|---|
| 22, 20 to 17 | 3DG-E 3DG-D 3DG-C 3DG-B 3DG-A | B'0, B'0, B'0, B'0, B'0 | R/W | <p>Specifies that status update of interrupt enable register by 3DG-E/D/C/B/A.</p> <p>[3DG-A/B/C/D/E are supported by RZ/G2H.]</p> <p>[3DG-A/B are supported by RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E and RZ/G2N.]</p> <p>By setting this bit, interrupt in same bit in the interrupt status register (SYSCISR) is enabled.</p> <p>0: SYSCISR is not updated when power shutoff or power resume processing is complete.</p> <p>1: SYSCISR is updated when power shutoff or power resume processing is complete.</p> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E]</p> <p>3DG-E/D/C: Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E]</p> <p>3DG-E: Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 16 to 15 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 14 | A3VC | B'0 | R/W | <p>Specifies that status update of interrupt enable register by A3VC.</p> <p>By setting this bit, interrupt in same bit in the interrupt status register (SYSCISR) is enabled.</p> <p>0: SYSCISR is not updated when power shutoff or power resume processing is complete.</p> <p>1: SYSCISR is updated when power shutoff or power resume processing is complete.</p> |
| 13 | — | B'0 | R/W | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 12 | CA57-SCU | B'0 | R/W | <p>Specifies that status update of interrupt enable register by Cortex-A57 SCU.</p> <p>[RZ/G2H, RZ/G2N, RZ/G2M V1.3 and RZ/G2M V3.0]</p> <p>By setting this bit, interrupt in same bit in the interrupt status register (SYSCISR) is enabled.</p> <p>0: SYSCISR is not updated when power shutoff or power resume processing is complete.</p> <p>1: SYSCISR is updated when power shutoff or power resume processing is complete.</p> <p>[RZ/G2E]</p> <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 11 to 10 | — | All 0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|--|-----------------------------|-----|---|
| 9 | A3VP | B'0 | R/W | <p>Specifies that status update of interrupt enable register by A3VP. [RZ/G2H and RZ/G2N]</p> <p>By setting this bit, interrupt in same bit in the interrupt status register (SYSCISR) is enabled.</p> <p>0: SYSCISR is not updated when power shutoff or power resume processing is complete. 1: SYSCISR is updated when power shutoff or power resume processing is complete.</p> <p>[RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2E] Reserved This bit is always read as 0. The write value should always be 0.</p> |
| 8 to 5 | CA53-CPU3 CA53-CPU2 CA53-CPU1 CA53-CPU0 | B'0, B'0, B'0, B'0 | R/W | <p>Specifies that status update of interrupt enable register by Cortex-A53 CPU3/2/1/0. [RZ/G2H, RZ/G2M V1.3 and RZ/G2M V3.0] [RZ/G2E only for Cortex-A53-CPU[0] and Cortex-A53-CPU[1]]</p> <p>By setting this bit, interrupt in same bit in the interrupt status register (SYSCISR) is enabled.</p> <p>0: SYSCISR is not updated when power shutoff or power resume processing is complete. 1: SYSCISR is updated when power shutoff or power resume processing is complete.</p> <p>[RZ/G2N] Reserved These bits are always read as 0. The write value should always be 0. [RZ/G2E] CA53-CPU3/2 reserved These bits are always read as 0. The write value should always be 0.</p> |
| 4 | — | B'0 | R | <p>Reserved This bit is always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|--|-----------------------------|-----|--|
| 3 to 0 | CA57-CPU3 CA57-CPU2 CA57-CPU1 CA57-CPU0 | B'0, B'0, B'0, B'0 | R/W | <p>Specifies that status update of interrupt enable register by Cortex-A57 CPU3/2/1/0.</p> <p>[RZ/G2H]</p> <p>[RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N only for Cortex-A57 CPU[0] and Cortex-A57 CPU[1]]</p> <p>By setting this bit, interrupt in same bit in the interrupt status register (SYSCISR) is enabled.</p> <p>0: SYSCISR is not updated when power shutoff or power resume processing is complete.</p> <p>1: SYSCISR is updated when power shutoff or power resume processing is complete.</p> <p>Note: Bits for CPU3 and CPU2 are valid for RZ/G2H. For other product, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0.</p> <p>[RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N]</p> <p>CA53-CPU3/2: Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> <p>[RZ/G2E]</p> <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |

14.2.1.5 Interrupt Mask Register (SYSCIMR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|------|-----|----------|-----|-------|-------|---------------|-----|-------|----------|---------------|-------|-------|-------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | A2VC1 | A2VC0 | — | — | 3DG-E | CA53-SCU | 3DG-D | 3DG-C | 3DG-B | 3DG-A | — |
| Initial value: | 1/0 | 1/0 | 1/0 | 1/0 | 1/0 | 1/0 | 1/0 | 1 | 1/0 | 1/0 | 1 | 1/0 | 1/0 | 1/0 | 1/0 | 1/0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | A3VC | — | CA57-SCU | — | — | A3VP | CA53-CPU[3:0] | | | — | CA57-CPU[3:0] | | | | |
| Initial value: | 0 | 1/0 | 1 | 1/0 | 1/0 | 1/0 | 1/0 | 1 | 1 | 1 | 1 | 1/0 | 1/0 | 1/0 | 1/0 | 1/0 |
| R/W: | R | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W |

Note: Initial value of each bits in this register depends on the product. Refer to description for each bits.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 27 | — | All 1 (*) | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 26 to 25 | A2VC1 A2VC0 | B'1, B'0 | R/W | Mask the interrupt output by A2VC1/0. [RZ/G2M V1.3] A2VC0 and A2VC1 are Supported 0: Does not mask the interrupt by this interrupt factor. 1: Masks the interrupt by this interrupt factor. [RZ/G2H, RZ/G2N, RZ/G2E and RZ/G2M V3.0] A2VC1 is Supported 0: Does not mask the interrupt by this interrupt factor. 1: Masks the interrupt by this interrupt factor. A2VC0 is Reserved This bit is always read as 1. The write value should always be 1. |
| 24 | — | B'1 | R/W | Reserved This bit is always read as 1. The write value should always be 1. |
| 23 | — | B'1(*) | R/W | Reserved |
| 22 | — | B'1(*) | R/W | Mask the interrupt output by 3DG-E. [RZ/G2H] 0: Does not mask the interrupt by this interrupt factor. 1: Masks the interrupt by this interrupt factor. [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E] Reserved This bit is always read as 1. The write value should always be 1. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------------------|---|-----|---|
| 21 | CA53-SCU | B'1 | R/W | Mask the interrupt output by Cortex-A53 SCU. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2E] 0: Does not mask the interrupt by this interrupt factor. 1: Masks the interrupt by this interrupt factor. [RZ/G2N] Reserved This bit is always read as 1. The write value should always be 1. |
| 20 to 17 | 3DG-D 3DG-C 3DG-B 3DG-A | B'1(*), B'1(*), B'1(*), B'1(*) | R/W | Mask the interrupt output by 3DG-D/C/B/A. [3DG-D/C/B/A are supported by RZ/G2H.] [3DG-B/A are supported by RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E and RZ/G2N.] 0: Does not mask the interrupt by this interrupt factor. 1: Masks the interrupt by this interrupt factor. [3DG- D/C are supported by RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E] Reserved These bits are always read as 1. The write value should always be 1. |
| 16 | — | B'1/ B'0 | R | Reserved This bit is always read as 1. The write value should always be 1. |
| 15 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 14 | A3VC | B'1(*) | R/W | Mask the interrupt output by A3VC. 0: Does not mask the interrupt by this interrupt factor. 1: Masks the interrupt by this interrupt factor. |
| 13 | — | B'1 | R/W | Reserved This bit is always read as 1. The write value should always be 1. |
| 12 | CA57-SCU | B'1(*) | R/W | Mask the interrupt output by Cortex-A57 SCU. [RZ/G2H, RZ/G2N, RZ/G2M V1.3 and RZ/G2M V3.0] 0: Does not mask the interrupt by this interrupt factor. 1: Masks the interrupt by this interrupt factor. [RZ/G2E] Reserved This bit is always read as 1. The write value should always be 1. |
| 11 to 10 | — | B'1/ B'0 | R | Reserved These bits are always read as 1. The write value should always be 1. |
| 9 | A3VP | B'1(*) | R/W | Mask the interrupt output by A3VP. [RZ/G2H and RZ/G2N] 0: Does not mask the interrupt by this interrupt factor. 1: Masks the interrupt by this interrupt factor. [RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2E] Reserved These bits are always read as 1. The write value should always be 1. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------|---------------|-----|--|
| 8 to 5 | CA53-CPU[3:0] | H'F | R/W | <p>Mask the interrupt output by Cortex-A53 CPU3/2/1/0. [RZ/G2H, RZ/G2M V1.3 and RZ/G2M V3.0] [Cortex-A53 CPU1/0 are supported by RZ/G2E.] 0: Does not mask the interrupt by this interrupt factor. 1: Masks the interrupt by this interrupt factor.</p> <p>[RZ/G2N] [CA53-CPU3/2 are supported by RZ/G2E] Reserved These bits are always read as 1. The write value should always be 1.</p> |
| 4 | — | B'1/ B'0 | R | <p>Reserved This bit is always read as 1. The write value should always be 1.</p> |
| 3 to 0 | CA57-CPU[3:0] | H'F(*) | R/W | <p>Mask the interrupt output by Cortex-A57 CPU3/2/1/0. [RZ/G2H] [Cortex-A57 CPU1/0 are supported by RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N.] 0: Does not mask the interrupt by this interrupt factor. 1: Masks the interrupt by this interrupt factor. Note: Bits for CPU3 and CPU2 are valid for RZ/G2H. For other product, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0.</p> <p>[RZ/G2E] [CA57-CPU3/2 are supported by RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N] Reserved These bits are always read as 1. The write value should always be 1.</p> |

Note: (*) for bits, which are not supported by each product, initial value will be 0.

14.2.1.6 Cortex-A57 Wake Up Mask Register (WUPMSKCA57)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----------|-----|-----|-----|----|----|----|----|----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CSD[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | FIQ[3:0] | | | | — | — | — | — | IRQ[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 to 16 | CSD[3:0] | H'0 | R/W | Enable/Disable wake up Cortex-A57 CPU0 to 3 + SCU area when receiving CSD[3:0] factor [RZ/G2H] [RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N only for CSD[0] and CSD[1].] 0: Enable 1: Disable This setting is available only when Cortex-A57 SCU area is in OFF state. Note: If does not exist Arm CPUs, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0. If does exist Arm CPUs, refer to Appendix A of Arm CPUs configuration. |
| 15 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 8 | FIQ[3:0] | H'0 | R/W | Enable/Disable wake up Cortex-A57 CPU0 to 3 + SCU area when receiving FIQ[3:0] factor [RZ/G2H] [RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N only for FIQ[0] and FIQ[1]] 0: Enable 1: Disable This setting is available only when Cortex-A57-SCU area is in OFF state. Note: If does not exist Arm CPUs, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0. If does exist Arm CPUs, refer to Appendix A of Arm CPUs configuration. |
| 7 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 3 to 0 | IRQ[3:0] | H'0 | R/W | <p>Enable/Disable wake up Cortex-A57 CPU0 to 3 + SCU area when receiving IRQ[3:0] factor</p> <p>[RZ/G2H]</p> <p>[RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N only for IRQ[0] and IRQ[1]]</p> <p>0: Enable</p> <p>1: Disable</p> <p>This setting is available only when CA57-SCU area is in OFF state.</p> <p>Note: If does not exist Arm CPUs, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0. If does exist Arm CPUs, refer to Appendix A of Arm CPUs configuration.</p> |

14.2.1.7 Cortex-A53 Wake Up Mask Register (WUPMSKCA53)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----------|-----|-----|-----|----|----|----|----------|----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CSD[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | FIQ[3:0] | | | — | — | — | — | IRQ[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 to 16 | CSD[3:0] | H'0 | R/W | Enable/Disable wake up Cortex-A53 + SCU area when receiving CSD[3:0] factor [RZ/G2H, RZ/G2M V1.3 and RZ/G2M V3.0] [RZ/G2E only for CSD[0] and CSD[1]] 0: Enable 1: Disable This setting is available only when Cortex-A53 SCU area is in OFF state [RZ/G2E] CSD[3:2]: Reserved Read value for these bits are unknown. The write value should always be 0. |
| 15 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 8 | FIQ[3:0] | H'0 | R/W | Enable/Disable wake up Cortex-A53 CPU0 to 3 + SCU area when receiving FIQ[3:0] factor [RZ/G2H, RZ/G2M V1.3 and RZ/G2M V3.0] [RZ/G2E only for FIQ[0] and FIQ[1]] 0: Enable 1: Disable This setting is available only when Cortex-A53 SCU area is in OFF state. [RZ/G2E] FIQ[3:2]: Reserved Read value for these bits are unknown. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | IRQ[3:0] | H'0 | R/W | Enable/Disable wake up Cortex-A53 CPU0 to 3 + SCU area when receiving IRQ[3:0] factor [RZ/G2H, RZ/G2M V1.3 and RZ/G2M V3.0] [RZ/G2E only for IRQ[0] and IRQ[1]] 0: Enable 1: Disable This setting is available only when Cortex-A53 SCU area is in OFF state. [RZ/G2N] Reserved Read value for these bits are unknown. The write value should always be 0. [RZ/G2E] IRQ[3:2]: Reserved Read value for these bits are unknown. The write value should always be 0. |

14.2.1.8 External Event Request Status Register (SYSCEERSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|---------------|----|----|----|---------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | FIQ_CA57[3:0] | | | | IRQ_CA57[3:0] | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. |
| 11 to 8 | FIQ_CA57[3:0] | H'0 | R | <p>Cortex-A57 Power Resume by FIQ is accepted [RZ/G2H] [RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N only for FIQ_CA57[0] and FIQ_CA57[1]]</p> <p>If the FIQ_CA57[3:0] bit in the external event request status enable register is 1, this bit is set to 1 when a power resume request due to an FIQ interrupt is accepted by the Cortex-A57 core CPU3-0. If the FIQ_CA57[3:0] bit in the external event request status enable register is 0, this bit is not set to 1 even when a power resume request due to an FIQ interrupt is accepted.</p> <p>0: The power resume request due to an FIQ interrupt has not been accepted by the Cortex-A57 core CPU3-0. 1: The power resume request due to an FIQ interrupt has been accepted by the Cortex-A57 core CPU3-0.</p> <p>This bit is set, only when condition above is satisfied in Cortex-A57/SCU is power-shutoff state.</p> <p>Note: If does not exist Arm CPUs, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0. If does exist Arm CPUs, refer to Appendix A of Arm CPUs configuration.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------|---------------|-----|---|
| 7 to 4 | IRQ_CA57[3:0] | H'0 | R | <p>Cortex-A57 Power Resume by IRQ is accepted [RZ/G2H]</p> <p>[RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N only for IRQ_CA57[0] and IRQ_CA57[1]]</p> <p>If the IRQ_CA57[3:0] bit in the external event request status enable register is 1, this bit is set to 1 when a power resume request due to an IRQ interrupt is accepted by the Cortex-A57 core CPU3-0. If the IRQ_CA57[3:0] bit in the external event request status enable register is 0, this bit is not set to 1 even when a power resume request due to an IRQ interrupt is accepted.</p> <p>0: The power resume request due to an IRQ interrupt has not been accepted by the Cortex-A57 core CPU3-0.</p> <p>1: The power resume request due to an IRQ interrupt has been accepted by the Cortex-A57 core CPU3-0.</p> <p>This bit is set, only when condition above is satisfied in Cortex-A57/SCU is power-shutoff state.</p> <p>Note: If does not exist Arm CPUs, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0. If does exist Arm CPUs, refer to Appendix A of Arm CPUs configuration.</p> |
| 3 to 0 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0.</p> |

14.2.1.9 External Event Request Status Clear Register (SYSCEERSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|---------------|------|------|------|---------------|------|------|------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | FIQ_CA57[3:0] | | | | IRQ_CA57[3:0] | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|------|--|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 8 | FIQ_CA57[3:0] | — | —/WB | Clear FIQ_CA57[3:0] in SYSCEERSR. [RZ/G2H] [RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N only for FIQ_CA57[0] and FIQ_CA57[1]] Writing 1 to this bit clears the FIQ_CA57[3:0] bit in the external event request status register (SYSCEERSR) to 0. Writing 0 to this bit, or writing 1 to this bit when the related bit in the external event request status register is 0, will be ignored. Note: If does not exist Arm CPUs, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0. If does exist Arm CPUs, refer to Appendix A of Arm CPUs configuration. |
| 7 to 4 | IRQ_CA57[3:0] | — | —/WB | Clear IRQ_CA57[3:0] in SYSCEERSR. [RZ/G2H] [RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N only for IRQ_CA57[0] and IRQ_CA57[1]] Writing 1 to this bit clears the IRQ_CA57[3:0] bit in the external event request status register (SYSCEERSR) to 0. Writing 0 to this bit, or writing 1 to this bit when the related bit in the external event request status register is 0, will be ignored. Note: If does not exist Arm CPUs, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0. If does exist Arm CPUs, refer to Appendix A of Arm CPUs configuration. |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

14.2.1.10 External Event Request Status Enable Register (SYSCEERSER)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|---------------|-----|-----|-----|---------------|-----|-----|-----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | FIQ_CA57[3:0] | | | | IRQ_CA57[3:0] | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 8 | FIQ_CA57[3:0] | H'0 | R/W | Update control of FIQ_CA57[3:0] in SYSCEERSR. [RZ/G2H] [RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N only for FIQ_CA57[0] and FIQ_CA57[1]] Specifies that the status of FIQ_CA57[3:0] bit in the external event request status register (SYSCEERSR) is updated or not, when a power resume request by a FIQ interrupt is accepted by the CA57 core CPU3-0. 0: Status is not updated when a power resume request is accepted. 1: Status is updated when a power resume request is accepted. Note: If does not exist Arm CPUs, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0. If does exist Arm CPUs, refer to Appendix A of Arm CPUs configuration. |
| 7 to 4 | IRQ_CA57[3:0] | H'0 | R/W | Update control of IRQ_CA57[3:0] in SYSCEERSR. [RZ/G2H] [RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N only for IRQ_CA57[0] and IRQ_CA57[1]] Specifies that the status of IRQ_CA57[3:0] bit in the external event request status register (SYSCEERSR) is updated or not, when a power resume request by an IRQ interrupt is accepted by the CA57 core CPU3-0. 0: Status is not updated when a power resume request is accepted. 1: Status is updated when a power resume request is accepted. Note: If does not exist Arm CPUs, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0. If does exist Arm CPUs, refer to Appendix A of Arm CPUs configuration. |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

14.2.1.11 External Event Request Status Register2 (SYSCEERSR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|---------------|----|----|----|---------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | FIQ_CA53[3:0] | | | | IRQ_CA53[3:0] | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. |
| 11 to 8 | FIQ_CA53[3:0] | H'0 | R | <p>Cortex-A53 Power Resume by FIQ is accepted [RZ/G2H, RZ/G2M V1.3, and RZ/G2M V3.0] [RZ/G2E only for FIQ_CA53[0] and FIQ_CA53[1]]</p> <p>If the FIQ_CA53[3:0] bit in the external event request status enable register is 1, this bit is set to 1 when a power resume request due to an FIQ interrupt is accepted by the Cortex-A53 core CPU3-0. If the FIQ_CA53[3:0] bit in the external event request status enable register is 0, this bit is not set to 1 even when a power resume request due to an FIQ interrupt is accepted.</p> <p>0: The power resume request due to an FIQ interrupt has not been accepted by the Cortex-A53 core CPU3-0. 1: The power resume request due to an FIQ interrupt has been accepted by the Cortex-A53 core CPU3-0.</p> <p>This bit is set, only when condition above is satisfied in Cortex-A53 SCU is power-shutoff state.</p> <p>Note: If does not exist Arm CPUs, those bits are reserved. Read value for those bits are unknown. If does exist Arm CPUs, refer to Appendix A of Arm CPUs configuration.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------|---------------|-----|---|
| 7 to 4 | IRQ_CA53[3:0] | H'0 | R | <p>Cortex-A53 Power Resume by IRQ is accepted [RZ/G2H, RZ/G2M V1.3, and RZ/G2M V3.0] [RZ/G2E only for IRQ_CA53[0] and IRQ_CA53[1]]</p> <p>If the IRQ_CA53[3:0] bit in the external event request status enable register is 1, this bit is set to 1 when a power resume request due to an IRQ interrupt is accepted by the Arm core CPU3-0. If the IRQ_CA53[3:0] bit in the external event request status enable register is 0, this bit is not set to 1 even when a power resume request due to an IRQ interrupt is accepted.</p> <p>0: The power resume request due to an IRQ interrupt has not been accepted by the Cortex-A53 core CPU3-0. 1: The power resume request due to an IRQ interrupt has been accepted by the Cortex-A53 core CPU3-0.</p> <p>This bit is set, only when condition above is satisfied in Cortex-A53 SCU is power-shutoff state.</p> <p>Note: If does not exist Arm CPUs, those bits are reserved. Read value for those bits are unknown. If does exist Arm CPUs, refer to Appendix A of Arm CPUs configuration.</p> |
| 3 to 0 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0.</p> |

14.2.1.12 External Event Request Status Clear Register2 (SYSCEERSR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|---------------|------|------|------|---------------|------|------|------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | FIQ_CA53[3:0] | | | | IRQ_CA53[3:0] | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | —/WB | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|------|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 8 | FIQ_CA53[3:0] | — | —/WB | Clear FIQ_CA53[3:0] in SYSCEERSR2. [RZ/G2H, RZ/G2M V1.3, and RZ/G2M V3.0] [RZ/G2E only for FIQ_CA53[0] and FIQ_CA53[1]] Writing 1 to this bit clears the FIQ_CA53[3:0] bit in the external event request status register 2 (SYSCEERSR2) to 0. Writing 0 to this bit, or writing 1 to this bit when the related bit in the external event request status register2 is 0, will be ignored. Note: If there is no corresponding Arm CPU, these bits are reserved. Read value for these bits are unknown, and write value for these bits should be 0. Refer to Appendix A for configuration of the Arm CPUs. |
| 7 to 4 | IRQ_CA53[3:0] | — | —/WB | Clear IRQ_CA53[3:0] in SYSCEERSR2. [RZ/G2H, RZ/G2M V1.3, and RZ/G2M V3.0] [RZ/G2E only for IRQ_CA53[0] and IRQ_CA53[1]] Writing 1 to this bit clears the IRQ_CA53[3:0] bit in the external event request status register2 (SYSCEERSR2) to 0. Writing 0 to this bit, or writing 1 to this bit when the related bit in the external event request status register2 is 0, will be ignored. Note: If there is no corresponding Arm CPU, these bits are reserved. Read value for these bits are unknown, and write value for these bits should be 0. Refer to Appendix A for configuration of the Arm CPUs. |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

14.2.1.13 External Event Request Status Enable Register2 (SYSCEERSER2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|---------------|-----|-----|---------------|-----|-----|-----|-----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | FIQ_CA53[3:0] | | | IRQ_CA53[3:0] | | | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 8 | FIQ_CA53[3:0] | H'0 | R/W | Update control of FIQ_CA53[3:0] in SYSCEERSR2. Specifies that the status of FIQ_CA53[3:0] bit in the external event request status register2 (SYSCEERSR2) is update or not, when a power resume request due to a FIQ interrupt is accepted by the Cortex-A53 core CPU3-0. [RZ/G2H, RZ/G2M V1.3, and RZ/G2M V3.0] [RZ/G2E only for FIQ_CA53[0] and FIQ_CA53[1]] 0: Status is not updated when a power resume request is accepted. 1: Status is updated when a power resume request is accepted. Note: If there is no corresponding Arm CPU, these bits are reserved. Read value for these bits are unknown, and write value for these bits should be 0. Refer to Appendix A for configuration of the Arm CPUs. |
| 7 to 4 | IRQ_CA53[3:0] | H'0 | R/W | Update control of IRQ_CA53[3:0] in SYSCEERSR2. Specifies that the status of IRQ_CA53[3:0] bit in the external event request status register2 (SYSCEERSR2) is updated or not, when a power resume request due to an IRQ interrupt is accepted by the Cortex-A53 core CPU3-0. [RZ/G2H, RZ/G2M V1.3, and RZ/G2M V3.0] [RZ/G2E only for IRQ_CA53[0] and IRQ_CA53[1]] 0: Status is not updated when a power resume request is accepted. 1: Status is updated when a power resume request is accepted. Note: If there is no corresponding Arm CPU, these bits are reserved. Read value for these bits are unknown, and write value for these bits should be 0. Refer to Appendix A for configuration of the Arm CPUs. |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

14.2.1.14 External Request Mask Register (SYSCEXTMASK)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | √ | √ |

Below is description of RZ/G2H, RZ/G2M V3.0, RZ/G2N and RZ/G2E.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | EXTMSK0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | EXTMSK0 | B'1(*1) | R/W | [RZ/G2H, RZ/G2N, RZ/G2E and RZ/G2M V3.0] This bit is used for masking external power ON/OFF Cortex-A57, Cortex-A53 or 3DG requests from APMU (Refer to section 13) or 3DG in case of one other power ON/OFF is performing. When this bit is set, the external request cannot be handled until clear this bit. Please refer to 14.3.4 Power Control of non-ARM CPUs for updating routine related to this register. 0: External request is not masked. 1: External request is masked |

Note: External request ON/OFF means all power ON/OFF requests which are not register setting of SYSC.

Notes: 1. supported by , RZ/G2H and RZ/G2M V3.0.

Power Control Registers for Arm CPUs

These registers control power status of Cortex-A57 and Cortex-A53 CPUs.

The “n” in the register name in this section means 0 or 6, and those numbers correspond to modules as follows:

Cortex-A57: n = 0 Supported by RZ/G2H. Only CPU1/0 are supported by RZ/G2M V1.3 and RZ/G2M V3.0.

Cortex-A53: n = 6 Supported by , RZ/G2H, RZ/G2M V1.3 and RZ/G2M V3.0. Only CPU1/0 are supported by RZ/G2E.

Each RZ/G product supports their specific power-domains. Refer to “14.3.1 Power domain structure”, for supported power-domains by each product. In following registers, bits for power-domains are defined as reserved bits, if corresponding power-domain is not supported by the product, if special notice is not written. In this case, those bits are defined as reserved bits, and read value for those bits are unknown, and write value for those bits should be 0.

14.2.1.15 Power Status Register n (PWRSRn) (n = 0/6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----------------|----|----|-----------------|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | PWRUP_CPU[3:0] | | | PWRDWN_CPU[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0/1* | 1 | 1 | 1 | 0/1* |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. |
| 7 to 4 | PWRUP_CPU[3:0] | H'0 for CPU1/2/3 H'F for CPU0 (*) | R | Indicates the non-power-shutoff state of CPU3-0. 0: Not in power on state 1: In non-power-shutoff state |
| 3 to 0 | PWRDWN_CPU[3:0] | H'0 for CPU0 H'F for CPU1/2/3 (*) | R | Indicates the power-shutoff state of CPU3-0. 0: Not in power-shutoff state 1: In power-shutoff state |

Note: (*) Initial value of this register depends on mode pin setting. Refer to 14.3.2 Initial state of power domains for detail.

14.2.1.16 Power Shutoff Status Register n (PWROFFSRn) (n = 0/6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----------|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CPU[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. |
| 3 to 0 | CPU[3:0] | H'0 | R | Indicates the power shutoff sequence execution status of CPU3-0 0: The power shutoff sequence not being executed 1: The power shutoff sequence being executed Note: If there is no corresponding Arm CPU, these bits are reserved. Read value for these bits are unknown, and write value for these bits should be 0. Refer to Appendix A for configuration of the Arm CPUs. |

14.2.1.17 Power Resume Status Register n (PWRONSRn) (n = 0/6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----------|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CPU[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. |
| 3 to 0 | CPU[3:0] | H'0 | R | Indicates the power resume sequence execution status of CPU3-0. 0: The power resume sequence not being executed 1: The power resume sequence being executed Note: If there is no corresponding Arm CPU, these bits are reserved. Read value for these bits are unknown, and write value for these bits should be 0. Refer to Appendix A for configuration of the Arm CPUs. |

14.2.1.18 Power Shutoff/Resume Error Register n (PWREn) (n = 0/6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----------|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CPU[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. |
| 3 to 0 | CPU[3:0] | H'0 | R | Indicates whether a power shutoff or power resume request by a preceding register write to CPU3-0 was accepted or not. 0: Either a preceding power shutoff or power resume request to CPU3-0 was accepted. 1: Either a preceding power shutoff or power resume request to CPU3-0 was not accepted. Note: If there is no corresponding Arm CPU, these bits are reserved. Read value for these bits are unknown, and write value for these bits should be 0. Refer to Appendix A for configuration of the Arm CPUs. |

14.2.1.19 Power Pseudo Shutoff Register n (PWRPSEUn) (n = 0/6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | CPU[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | CPU[3:0] | H'0 | R/W | By setting those bits to perform pseudo power shutoff, power is not turn off in actual, for CPU3-0. 0: Pseudo power shut off is not performed 1: Pseudo power shut off is performed. Note: If there is no corresponding Arm CPU, these bits are reserved. Read value for these bits are unknown, and write value for these bits should be 0. Refer to Appendix A for configuration of the Arm CPUs. |

14.2.2 Power Control Registers for SCU of Cortex-A53, SCU of Cortex-A57, Video Processing, and Video Codec

These registers control power shutoff/resume of following modules. The “n” in the register name in this section means 3, 4, 5, 8 or, 9 and those numbers correspond to modules as follows:

| | | |
|-----------------|-------|--|
| Cortex-A53 SCU: | n = 3 | Supported by RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2E |
| A3IR: | n = 4 | Supported by RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 |
| Cortex-A57 SCU: | n = 5 | Supported by RZ/G2H, RZ/G2N, RZ/G2M V1.3 and RZ/G2M V3.0 |
| A3VP: | n = 8 | Supported by RZ/G2H and RZ/G2N |
| A3VC: | n = 9 | Supported by RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E |

Each RZ/G product supports their specific power-domains. Refer to “14.3.1 Power domain structure”, for supported power-domains by each product. In following registers, bits for power-domains are defined as reserved bits, if corresponding power-domain is not supported by the product and special notice is not written. In this case, those bits are defined as reserved bits, and read value for those bits are unknown, and write value for those bits should be 0.

14.2.2.1 Power Status Register n (PWRSRn) (n = 3/4/5/8/9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|-----------|----|----|----|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | PWRU P | — | — | — | PWRD WN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0/1* | 0 | 0 | 0 | 0/1* |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. |
| 4 | PWRUP | B'0/ B'1 (*) | R | Indicates the power-ON status of a given module. 0: The module is not in the power-ON state. 1: The module is in the power-ON state. |
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. |
| 0 | PWRDWN | B'0/ B'1 (*) | R | Indicates the power shutoff status of a given module. 0: The module is not in the power shutoff state. 1: The module is in the power shutoff state. |

Note: (*) Initial value of this register depends on mode pin setting. Refer to 14.3.2 Initial state of power domains for detail.

14.2.2.2 Power Shutoff Control Register n (PWROFFCRn) (n = 3/4/5/8/9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | PWRDWN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | —/WB |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|-------------|
| 31 to 1 | — | All 0 | R | Reserved |

| | | | | |
|---|-----------|---|------|--|
| 0 | PWRDWN(*) | — | —/WB | <p>Specify the start of power-shutoff sequence.</p> <p>Writing 1 to this bit when the POFFENB bit in the SYSC status register is 1 starts the power shutoff sequence for the module. In this case, the ERR bit in the power shutoff/resume error register n is set to 0.</p> <p>Writing 1 to this bit when the POFFENB bit in the SYSC status register is 0 does not start the power shutoff sequence. In this case, the ERR bit in the power shutoff/resume error register n is set to 1.</p> |
|---|-----------|---|------|--|

- 0: Does not shut off power for the module.
- 1: Starts the power shutoff sequence for the module.

When the power shutoff sequence is started, during the execution of the power shutoff sequence the shutoff processing status is indicated in the DWNSTATE bit in the power shutoff status register n. The PWRUP bit in the power status register n is set to 0. Upon completion of the power shutoff sequence, the PWRDWN bit in the power status register n is set to 1. The pertinent bit in the interrupt status register is set to 1 if it is enabled by the setting of the interrupt status enable register.

If 1 is written to this bit when the module is in the power shutoff state, such a request is ignored. In this case, too, if 1 is written when the POFFENB bit is 1, the pertinent bit in the power shutoff/resume error register is set to 0, and if 1 is written when the POFFENB bit is 0, the pertinent bit in the power shutoff/resume error register is set to 1.

14.2.2.3 Power Shutoff Status Register n (PWROFFSRn) (n = 3/4/5/8/9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | DWNS TATE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. |
| 0 | DWNSTATE | B'0 | R | Indicates the power shutoff sequence execution status for the module. 0: The power shutoff sequence not being executed 1: The power shutoff sequence being executed |

14.2.2.4 Power Resume Control Register n (PWRONCRn) (n = 3/4/5/8/9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | PWRUP |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | —/WB |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| | | | | |
|---|-------|---|------|--|
| 0 | PWRUP | — | —/WB | Specify the start of power-resume sequence. Writing 1 to this bit when the PONENB bit in the SYSC status register is 1 starts the power resume sequence for the module. In this case, the ERR bit in the power shutoff/resume error register n is set to 0. Writing 1 to this bit when the PONENB bit in the SYSC status register is 0 does not start the power resume sequence. In this case, the ERR bit in the power shutoff/resume error register n is set to 1. |
|---|-------|---|------|--|

- 0: Does not resume power for the module.
- 1: Starts the power resume sequence for the module.

When the power resume sequence is started, during the execution of the power resume sequence the resume processing status is indicated in the UPSTATE bit in the power resume status register n. The PWRDWN bit in the power status register n is set to 0. Upon completion of the power resume sequence, the PWRUP bit in the power status register n is set to 1. The pertinent bit in the interrupt status register is set to 1 if it is enabled by the setting of the interrupt status enable register.

If 1 is written to this bit when the module is in the power-ON state, such a request is ignored. In this case, too, if 1 is written when the PONENB bit is 1, the pertinent bit in the power shutoff/resume error register is set to 0, and if 1 is written when the PONENB bit is 0, the pertinent bit in the power shutoff/resume error register is set to 1.

14.2.2.5 Power Resume Status Register n (PWRONSRn) (n = 3/4/5/8/9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | UPSTATE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. |
| 0 | UPSTATE | B'0 | R | Indicates the power resume sequence execution status for the module. 0: The power resume sequence not being executed 1: The power resume sequence being executed |

14.2.2.6 Power Shutoff/Resume Error Register n (PWRERn) (n = 3/4/5/8/9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ERR |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. |
| 0 | ERR | B'0 | R | Indicates that either a power shutoff or power resume request by a preceding register write to the module was issued when such requests were not acceptable. 0: Either a preceding power shutoff or power resume request to the module was accepted. 1: Either a preceding power shutoff or power resume request to the module was not accepted. |

14.2.2.7 Power Pseudo Shutoff Register n (PWRPSEUn) (n = 3/4/5/8/9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | PSE UDO |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | PSEUDO | B'0 | R/W | By setting this bit to perform pseudo power shutoff, power is not turn off in actual, for this domain. 0: Pseudo power shut off is not performed 1: Pseudo power shut off is performed. |

14.2.3 Power Control Registers for 3D Graphics Engine

These registers control power shutoff/resume of 3D Graphics Engine. The “n” in the register name in this section means 2, and those numbers correspond to modules as follows:

3D Graphics Engine: n = 2

3D Graphics Engine have 5 power domains inside modules (*), and they are called as 3DG-A, 3DG-B, 3DG-C, 3DG-D, and 3DG-E, respectively. Each of register for 3D Graphic Engine have bits, whose name have “-A”, “-B”, “-C”, “-D”, and “-E”, and each of them corresponds to power-domains above.

3D Graphics Engine module also have power-management function inside it. When power-management inside 3D Graphics Engine is used, do not use power-control function in this module. Power control function for 3D Graphics Engine in this module can be used only when, all power domain of 3D Graphics Engine will be shut off.

Note: * The number of power-domains inside 3D Graphic engine is as follows.

RZ/G2H: 5

RZ/G2M V1.3 and RZ/G2M V3.0: 2

RZ/G2N: 2 (power-domains A and B)

RZ/G2E: 2 (power-domains A and B)

The register bits which are not supported by each product are reserved. Read value for those bits are unknown, and write value for those bits should be 0.

14.2.3.1 Power Status Register n (PWRSRn) (n = 2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-------------|-------------|-------------|-------------|-------------|--------------|--------------|--------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | PWRU P-E | PWRU P-D | PWRU P-C | PWRU P-B | PWRU P-A | PWRD WN-E | PWRD WN-D | PWRD WN-C | PWRD WN-B | PWRD WN-A |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--|-------------------------------------|-----|--|
| 31 to 10 | — | All 0 | R | Reserved These bits are always read as 0. |
| 9 to 5 | PWRUP-E PWRUP-D PWRUP-C PWRUP-B PWRUP-A | B'0, B'0, B'0, B'0, B'0 | R | Indicates the power-ON status of a given module. 0: The module is not in the power-ON state. 1: The module is in the power-ON state. Note: Bits for 3DG-C, D, and E are valid for RZ/G2H. For other product, those bits are reserved. Read value for those bits are unknown. |
| 4 to 0 | PWRDWN-E PWRDWN-D PWRDWN-C PWRDWN-B PWRDWN-A | B'1, B'1, B'1, B'1, B'1 | R | Indicates the power shutoff status of a given module. 0: The module is not in the power shutoff state. 1: The module is in the power shutoff state. Note: Bits for 3DG-C, D, and E are valid for RZ/G2H. For other product, those bits are reserved. Read value for those bits are unknown. |

14.2.3.2 Power Shutoff Control Register n (PWROFFCRn) (n = 2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|--------------|--------------|--------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | PWRD WN-E | PWRD WN-D | PWRD WN-C | PWRD WN-B | PWRD WN-A |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | —/WB | —/WB | —/WB | —/WB | —/WB |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--|---------------|------|---|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | PWRDWN-E PWRDWN-D PWRDWN-C PWRDWN-B PWRDWN-A | — | —/WB | Specify the start of power-shutoff sequence. Writing 1 to this bit when the POFFENB bit in the SYSC status register is 1 starts the power shutoff sequence for the module. In this case, the ERR bit in the power shutoff/resume error register n is set to 0. Writing 1 to this bit when the POFFENB bit in the SYSC status register is 0 does not start the power shutoff sequence. In this case, the ERR bit in the power shutoff/resume error register n is set to 1. |

- 0: Does not shut off power for the module.
- 1: Starts the power shutoff sequence for the module.

When the power shutoff sequence is started, during the execution of the power shutoff sequence the shutoff processing status is indicated in the DWNSTATE bit in the power shutoff status register n. The PWRUP bit in the power status register n is set to 0. Upon completion of the power shutoff sequence, the PWRDWN bit in the power status register n is set to 1. The pertinent bit in the interrupt status register is set to 1 if it is enabled by the setting of the interrupt status enable register.

If 1 is written to this bit when the module is in the power shutoff state, such a request is ignored. In this case, too, if 1 is written when the POFFENB bit is 1, the pertinent bit in the power shutoff/resume error register is set to 0, and if 1 is written when the POFFENB bit is 0, the pertinent bit in the power shutoff/resume error register is set to 1.

Note: Bits for 3DG-C, D, and E are valid for RZ/G2H. For other product, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0.

14.2.3.3 Power Shutoff Status Register n (PWROFFSRn) (n = 2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----------------|----------------|----------------|----------------|----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | DWNS TATE-E | DWNST ATE-D | DWNS TATE-C | DWNS TATE-B | DWNS TATE-A |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--|-------------------------------------|-----|--|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. |
| 4 to 0 | DWNSTATE-E DWNSTATE-D DWNSTATE-C DWNSTATE-B DWNSTATE-A | B'0, B'0, B'0, B'0, B'0 | R | Indicates the power shutoff sequence execution status for the module. 0: The power shutoff sequence not being executed 1: The power shutoff sequence being executed Note: Bits for 3DG-C, D, and E are valid for RZ/G2H. For other product, those bits are reserved. Read value for those bits are unknown. |

14.2.3.4 Power Resume Control Register n (PWRONCRn) (n = 2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|-------------|-------------|-------------|-------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | PWRU P-E | PWRU P-D | PWRU P-C | PWRU P-B | PWRU P-A |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | —/WB | —/WB | —/WB | —/WB | —/WB |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---|---------------|------|--|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | PWRUP-E PWRUP-D PWRUP-C PWRUP-B PWRUP-A | — | —/WB | Specify the start of power-resume sequence. Writing 1 to this bit when the PONENB bit in the SYSC status register is 1 starts the power resume sequence for the module. In this case, the ERR bit in the power shutoff/resume error register n is set to 0. Writing 1 to this bit when the PONENB bit in the SYSC status register is 0 does not start the power resume sequence. In this case, the ERR bit in the power shutoff/resume error register n is set to 1. |

0: Does not resume power for the module.

1: Starts the power resume sequence for the module.

When the power resume sequence is started, during the execution of the power resume sequence the resume processing status is indicated in the UPSTATE bit in the power resume status register n. The PWRDWN bit in the power status register n is set to 0. Upon completion of the power resume sequence, the PWRUP bit in the power status register n is set to 1. The pertinent bit in the interrupt status register is set to 1 if it is enabled by the setting of the interrupt status enable register.

If 1 is written to this bit when the module is in the power-ON state, such a request is ignored. In this case, too, if 1 is written when the PONENB bit is 1, the pertinent bit in the power shutoff/resume error register is set to 0, and if 1 is written when the PONENB bit is 0, the pertinent bit in the power shutoff/resume error register is set to 1.

Note: Bits for 3DG-C, D, and E are valid for RZ/G2H. For other product, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0.

14.2.3.5 Power Resume Status Register n (PWRONSRn) (n = 2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|---------------|---------------|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | UPSTA TE-E | UPSTA TE-D | UPSTA TE-C | UPSTA TE-B | UPSTA TE-A |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---|-------------------------------------|-----|--|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. |
| 4 to 0 | UPSTATE-E UPSTATE-D UPSTATE-C UPSTATE-B UPSTATE-A | B'0, B'0, B'0, B'0, B'0 | R | Indicates the power resume sequence execution status for the module. 0: The power resume sequence not being executed 1: The power resume sequence being executed |

Note: Bits for 3DG-C, D, and E are valid for RZ/G2H. For other product, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0.

14.2.3.6 Power Shutoff/Resume Error Register n (PWREn) (n = 2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | ERR-E | ERR-D | ERR-C | ERR-B | ERR-A |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---|-------------------------------------|-----|--|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. |
| 4 to 0 | ERR-E ERR-D ERR-C ERR-B ERR-A | B'0, B'0, B'0, B'0, B'0 | R | Indicates that either a power shutoff or power resume request by a preceding register write to the module was issued when such requests were not acceptable. 0: Either a preceding power shutoff or power resume request to the module was accepted. 1: Either a preceding power shutoff or power resume request to the module was not accepted. |

Note: Bits for 3DG-C, D, and E are valid for RZ/G2H. For other product, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0.

14.2.3.7 Power Pseudo Shutoff Register n (PWRPSEUn) (n = 2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|--------|--------|--------|--------|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | PSEUDE | PSEUDD | PSEUDC | PSEUDB | PSEUDA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--|-------------------------------------|-----|--|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | PSEUDE PSEUDD PSEUDC PSEUDB PSEUDA | B'0, B'0, B'0, B'0, B'0 | R/W | By setting those bits to perform pseudo power shutoff, power is not turn off in actual, for this domain. 0: Pseudo power shut off is not performed 1: Pseudo power shut off is performed. Note: Bits for 3DG-C, D, and E are valid for RZ/G2H. For other product, those bits are reserved. Read value for those bits are unknown, and write value for those bits should be 0. |

14.2.4 Power Control Registers for Video Codec

These registers control power shutoff/resume of Video Codec. The “n” in the register name in this section means 10, and those numbers correspond to modules as follows:

A2VC0 and A2VC1: n = 10 Supported by RZ/G2M V1.3

A2VC1: n = 10 Supported by RZ/G2H, RZ/G2M V3.0, RZ/G2N and RZ/G2E

14.2.4.1 Power Status Register n (PWRSRn) (n = 10)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|------------|------------|-------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | PWRU P1 | PWRU P0 | PWRD WN1 | PWRD WN0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. |
| 3 | PWRUP1 | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N] Indicates the power-ON status of a given A2VC1. 0: The A2VC1 is not in the power-ON state. 1: The A2VC1 is in the power-ON state. |
| 2 | PWRUP0 | B'0 | R | [RZ/G2M V1.3] Indicates the power on status of a given A2VC0. 0: The A2VC0 is not in the power on state. 1: The A2VC0 is in the power on state. [RZ/G2H, RZ/G2M V3.0 and RZ/G2N] Reserved These bits are always read as 0. |
| 1 | PWRDWN1 | B'1 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N] Indicates the power shutoff status of a given A2VC1. 0: The A2VC1 is not in the power shutoff state. 1: The A2VC1 is in the power shutoff state. |
| 0 | PWRDWN0 | B'1 | R | [RZ/G2M V1.3] Indicates the power shutoff status of a given A2VC0. 0: The A2VC0 is not in the power shutoff state. 1: The A2VC0 is in the power shutoff state. [RZ/G2H, RZ/G2M V3.0 and RZ/G2N] Reserved These bits are always read as 0. |

14.2.4.2 Power Shutoff Control Register n (PWROFFCRn) (n = 10)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | PWRDWN1 | PWRDWN0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | —/WB | —/WB |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|------|---|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | PWRDWN1 | — | —/WB | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N] Specify the start of power-shutoff sequence.</p> <p>Writing 1 to this bit when the POFFENB bit in the SYSC status register is 1 starts the power shutoff sequence for the A2VC1. In this case, the ERR bit in the power shutoff/resume error register n is set to 0.</p> <p>Writing 1 to this bit when the POFFENB bit in the SYSC status register is 0 does not start the power shutoff sequence. In this case, the ERR bit in the power shutoff/resume error register n is set to 1.</p> <p>0: Does not shut off power for the A2VC1. 1: Starts the power shutoff sequence for the A2VC1.</p> <p>When the power shutoff sequence is started, during the execution of the power shutoff sequence the shutoff processing status is indicated in the DWNSTATE bit in the power shutoff status register n. The PWRUP bit in the power status register n is set to 0. Upon completion of the power shutoff sequence, the PWRDWN bit in the power status register n is set to 1. The pertinent bit in the interrupt status register is set to 1 if it is enabled by the setting of the interrupt status enable register.</p> <p>If 1 is written to this bit when the module is in the power shutoff state, such a request is ignored. In this case, too, if 1 is written when the POFFENB bit is 1, the pertinent bit in the power shutoff/resume error register is set to 0, and if 1 is written when the POFFENB bit is 0, the pertinent bit in the power shutoff/resume error register is set to 1.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|---|
| 0 | PWRDWN0 | — | —/WB | <p>[RZ/G2M V1.3]</p> <p>Specify the start of power-shutoff sequence.</p> <p>Writing 1 to this bit when the POFFENB bit in the SYSC status register is 1 starts the power shutoff sequence for the A2VC0. In this case, the ERR bit in the power shutoff/resume error register n is set to 0.</p> <p>Writing 1 to this bit when the POFFENB bit in the SYSC status register is 0 does not start the power shutoff sequence. In this case, the ERR bit in the power shutoff/resume error register n is set to 1.</p> <p>0: Does not shut off power for the A2VC0. 1: Starts the power shutoff sequence for the A2VC0.</p> <p>[RZ/G2H, RZ/G2M V3.0 and RZ/G2N]</p> <p>Reserved</p> <p>The write value should always be 0.</p> <p>When the power shutoff sequence is started, during the execution of the power shutoff sequence the shutoff processing status is indicated in the DWNSTATE bit in the power shutoff status register n. The PWRUP bit in the power status register n is set to 0. Upon completion of the power shutoff sequence, the PWRDWN bit in the power status register n is set to 1. The pertinent bit in the interrupt status register is set to 1 if it is enabled by the setting of the interrupt status enable register.</p> <p>If 1 is written to this bit when the module is in the power shutoff state, such a request is ignored. In this case, too, if 1 is written when the POFFENB bit is 1, the pertinent bit in the power shutoff/resume error register is set to 0, and if 1 is written when the POFFENB bit is 0, the pertinent bit in the power shutoff/resume error register is set to 1.</p> |

14.2.4.3 Power Shutoff Status Register n (PWROFFSRn) (n = 10)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | DWNS TATE1 | DWNS TATE0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. |
| 1 | DWNSTATE1 | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N] Indicates the power shutoff sequence execution status for the A2VC1. 0: The power shutoff sequence not being executed 1: The power shutoff sequence being executed |
| 0 | DWNSTATE0 | B'0 | R | [RZ/G2M V1.3] Indicates the power shutoff sequence execution status for the A2VC0. 0: The power shutoff sequence not being executed 1: The power shutoff sequence being executed [RZ/G2H, RZ/G2M V3.0, RZ/G2N and RZ/G2E] Reserved These bits are always read as 0. |

14.2.4.4 Power Resume Control Register n (PWRONCRn) (n = 10)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | PWRU P1 | PWRU P0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | —/WB | —/WB |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|------|---|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | PWRUP1 | — | —/WB | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N] Specify the start of power-resume sequence. Writing 1 to this bit when the PONENB bit in the SYSC status register is 1 starts the power resume sequence for the A2VC1. In this case, the ERR bit in the power shutoff/resume error register n is set to 0. Writing 1 to this bit when the PONENB bit in the SYSC status register is 0 does not start the power resume sequence. In this case, the ERR bit in the power shutoff/resume error register n is set to 1. 0: Does not resume power for the A2VC1. 1: Starts the power resume sequence for the A2VC1. When the power resume sequence is started, during the execution of the power resume sequence the resume processing status is indicated in the UPSTATE bit in the power resume status register n. The PWRDWN bit in the power status register n is set to 0. Upon completion of the power resume sequence, the PWRUP bit in the power status register n is set to 1. The pertinent bit in the interrupt status register is set to 1 if it is enabled by the setting of the interrupt status enable register. If 1 is written to this bit when the module is in the power-ON state, such a request is ignored. In this case, too, if 1 is written when the PONENB bit is 1, the pertinent bit in the power shutoff/resume error register is set to 0, and if 1 is written when the PONENB bit is 0, the pertinent bit in the power shutoff/resume error register is set to 1. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 0 | PWRUP0 | — | —/WB | <p>[RZ/G2M V1.3]</p> <p>Specify the start of power-resume sequence.</p> <p>Writing 1 to this bit when the PONENB bit in the SYSC status register is 1 starts the power resume sequence for the A2VC0. In this case, the ERR bit in the power shutoff/resume error register n is set to 0.</p> <p>Writing 1 to this bit when the PONENB bit in the SYSC status register is 0 does not start the power resume sequence. In this case, the ERR bit in the power shutoff/resume error register n is set to 1.</p> <p>0: Does not shut off power for the A2VC0. 1: Starts the power shutoff sequence for the A2VC0.</p> <p>[RZ/G2H, RZ/G2M V3.0 and RZ/G2N]</p> <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> <p>When the power resume sequence is started, during the execution of the power resume sequence the resume processing status is indicated in the UPSTATE bit in the power resume status register n. The PWRDWN bit in the power status register n is set to 0. Upon completion of the power resume sequence, the PWRUP bit in the power status register n is set to 1. The pertinent bit in the interrupt status register is set to 1 if it is enabled by the setting of the interrupt status enable register.</p> <p>If 1 is written to this bit when the module is in the power on state, such a request is ignored. In this case, too, if 1 is written when the PONENB bit is 1, the pertinent bit in the power shutoff/resume error register is set to 0, and if 1 is written when the PONENB bit is 0, the pertinent bit in the power shutoff/resume error register is set to 1.</p> |

14.2.4.5 Power Resume Status Register n (PWRONSRn) (n = 10)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | UPSTA TE1 | UPSTA TE0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. |
| 1 | UPSTATE1 | B'0 | R | Indicates the power resume sequence execution status for the A2VC1. 0: The power resume sequence not being executed 1: The power resume sequence being executed |
| 0 | UPSTATE0 | B'0 | R | [RZ/G2M V1.3] Indicates the power resume sequence execution status for the A2VC0. 0: The power resume sequence not being executed 1: The power resume sequence being executed [RZ/G2H, RZ/G2M V3.0, RZ/G2N and RZ/G2E] Reserved These bits are always read as 0. |

14.2.4.6 Power Shutoff/Resume Error Register n (PWREn) (n = 10)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ERR1 | ERR0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. |
| 1 | ERR1 | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N] Indicates that either a power shutoff or power resume request by a preceding register write to the A2VC1 was issued when such requests were not acceptable. 0: Either a preceding power shutoff or power resume request to the A2VC1 was accepted. 1: Either a preceding power shutoff or power resume request to the A2VC1 was not accepted. |
| 0 | ERR0 | B'0 | R | [RZ/G2M V1.3] Indicates that either a power shutoff or power resume request by a preceding register write to the A2VC0 was issued when such requests were not acceptable. 0: Either a preceding power shutoff or power resume request to the A2VC0 was accepted. 1: Either a preceding power shutoff or power resume request to the A2VC0 was not accepted. [RZ/G2H, RZ/G2M V3.0 and RZ/G2N] Reserved These bits are always read as 0. |

14.2.4.7 Power Pseudo Shutoff Register n (PWRPSEUn) (n = 10)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | PSE UD1 | PSE UD0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | PSEUD1 | B'0 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N] By setting those bits to perform pseudo power shutoff, power is not turn off in actual, for this domain. 0: Pseudo power shut off is not performed 1: Pseudo power shut off is performed. |
| 0 | PSEUD0 | B'0 | R/W | [RZ/G2M V1.3 and RZ/G2M V3.0] By setting those bits to perform pseudo power shutoff, power is not turn off in actual, for this domain. 0: Pseudo power shut off is not performed 1: Pseudo power shut off is performed. [RZ/G2H and RZ/G2N] Reserved These bits are always read as 0. The write value should always be 0. |

14.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

14.3.1 Power domain structure

Figure 14.4 to 14.6 shows power domain structure, and Table 14.4 shows detail of each domain. When one of those power domain is shutoff state, all power domains which are written above the domain in Figure 14.4 to 14.6 must be also shutoff state. For example, when Cortex-A57 SCU is shutoff state, all of Cortex-A57 CPU0 to CPU3 must be shutoff state also.

There are some products which have pseudo power shutoff domain. (RZ/G2M V1.3 A3VC, RZ/G2E A3VC and A2VC1.)

The modules included in each power domain are listed in Table 14.4.

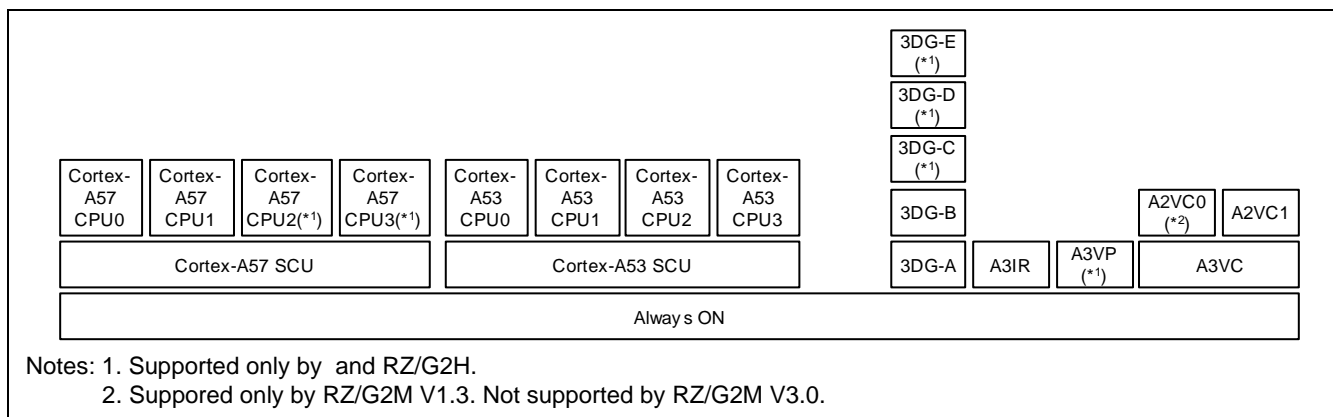


Figure 14.4 Power domain structure (RZ/G2H/RZ/G2M V1.3, RZ/G2M V3.0)

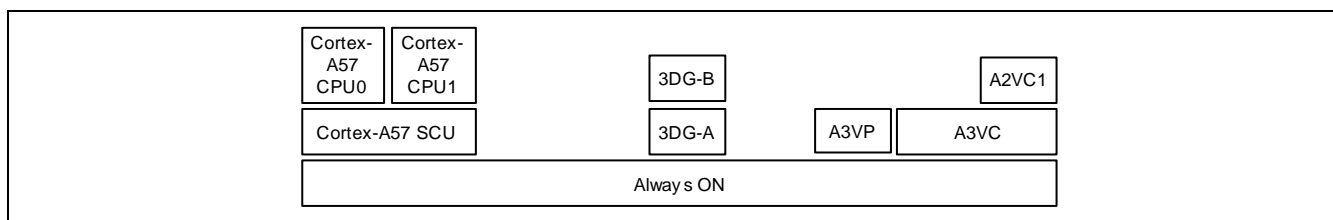


Figure 14.5 Power domain structure (RZ/G2N)

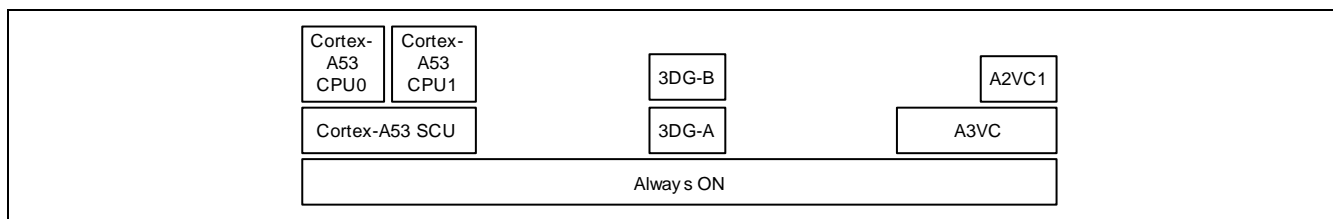


Figure 14.6 Power domain structure (RZ/G2E)

Table 14.4 Power domains.

| Domain name | Modules included in domain. | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E | Remark |
|-----------------|---|--------|-------------|-------------|--------|--------|---|
| Cortex-A57 CPU0 | AP-System Core (Cortex-A57) CPU0, including the L1 cache | √ | √ | √ | √ | — | |
| Cortex-A57 CPU1 | AP-System Core (Cortex-A57) CPU1, including the L1 cache | √ | √ | √ | √ | — | |
| Cortex-A57 CPU2 | AP-System Core (Cortex-A57) CPU2, including the L1 cache | √ | — | — | — | — | |
| Cortex-A57 CPU3 | AP-System Core (Cortex-A57) CPU3, including the L1 cache | √ | — | — | — | — | |
| Cortex-A57 SCU | AP-System Core (Cortex-A57) SCU, including the L2 cache | √ | √ | √ | √ | — | When one of CPU0 to CPU3 of Cortex-A57 is power-ON state, SCU of Cortex-A57 must be power-ON state. |
| Cortex-A53 CPU0 | AP-System Core (Cortex-A53) CPU0, including the L1 cache | √ | √ | √ | — | √ | Pseudo Power shutoff domain. |
| Cortex-A53 CPU1 | AP-System Core (Cortex-A53) CPU1, including the L1 cache | √ | √ | √ | — | √ | |
| Cortex-A53 CPU2 | AP-System Core (Cortex-A53) CPU2, including the L1 cache | √ | √ | √ | — | — | |
| Cortex-A53 CPU3 | AP-System Core (Cortex-A53) CPU3, including the L1 cache | √ | √ | √ | — | — | |
| Cortex-A53 SCU | AP-System Core (Cortex-A53) SCU, including the L2 cache | √ | √ | √ | — | √ | When one of CPU0 to CPU3 of Cortex-A53 is power-ON state, SCU of Cortex-A53 must be power-ON state. |
| 3DG-A | 3D Graphics Engine A | √ | √ | √ | √ | √ | 3DG power-domains can be controlled by SYSC for fully stopping their behavior, or automatically controlled by 3D Graphics Engine hardware. For detail of those domains, refer to chapter of 3D Graphics Engine. If 3D Graphics engine power domains are controlled by SYSC, either all 3DG power domain are ON or all 3DG power domain are OFF is allowed. |
| 3DG-B | 3D Graphics Engine B | √ | √ | √ | √ | √ | |
| 3DG-C | 3D Graphics Engine C | √ | — | — | — | — | |
| 3DG-D | 3D Graphics Engine D | √ | — | — | — | — | |
| 3DG-E | 3D-Graphics Engine E | √ | — | — | — | — | |
| A3IR | IPMMU(IR) module and bus modules. | √ | √ | √ | — | — | When one of the A3IR is power-ON state, A3IR domain must be power-ON state. |
| A3VP | IPMMU(VP0/VP1), VSP1, VSPBD, VSPBC, FDP1, FCPV, FCPF modules and bus modules | √ | — | — | — | — | |
| | IPMMU(VP0), VSP1, FDP1, FCPV, FCPF, VSPB(VSPBC+VSPBD) modules and bus modules | — | — | — | √ | — | |

| Domain name | Modules included in domain. | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E | Remark |
|-------------|---|--------|-------------|-------------|--------|--------|--|
| A3VC | FCPCS, IPMMU (VC), EDC checker (VC) modules and bus modules | — | — | — | — | √ | When one of A2VC0 or A2VC1 is power-ON state, A3VC must be power-ON state. (*Pseudo Power shutoff domain. |
| | FCPCS, IPMMU(VC0/VC1), EDC(VC) modules and bus modules | √ | — | — | — | — | When one of A2VC0 or A2VC1 is power-ON state, A3VC must be power-ON state |
| | IPMMU(VC0), EDC(VC) modules and bus modules | — | — | — | √ | — | When one of A2VC0 or A2VC1 is power-ON state, A3VC must be power-ON state |
| | FCPCS, FCPCI, IPMMU (VC0), VSPI, FDP1, FCPV, FCPF modules and bus modules | — | √ | — | — | — | When one of A2VC0 or A2VC1 is power-ON state, A3VC must be power-ON state. (*Pseudo Power shutoff domain. |
| | IPMMU (VC) modules and VSPI, FDP1, FCPV, FCPF and bus modules | — | — | √ | — | — | When A2VC1 is power-ON state, A3VC must be power-ON state. (*Pseudo Power shutoff domain. |
| A2VC0 | VCP4(VDPB) and iVDP1C modules. | — | √ | — | — | — | |
| A2VC1 | VCP4(VDPB) and VCP4(VCP4) modules | √ | — | — | — | √ | (*Pseudo Power shutoff domain. * |
| | FCPCS, VCP4(VDPB), VCP4(VCP4) modules | — | — | — | √ | — | |
| | VCP4(VCP4) module | — | √ | — | — | — | |
| | VDPB, VCP4(VCP4) and FCPCS modules | — | — | √ | — | — | |

14.3.2 Initial state of power domains

Table 14.5 and 14.6 shows initial states of each power domain. Initial power state of CPUs depends on the selection of boot CPU, which was specified by mode pins.

Table 14.5 Initial state of power domains (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0)

| Domain name | Cortex-A57 Boot | Cortex-A53 Boot | Remark |
|----------------------------|-----------------|-----------------|--------|
| Cortex-A57 CPU0 | ON | OFF | — |
| Cortex-A57 CPU1/2/3 | OFF | OFF | — |
| Cortex-A57 SCU | ON | OFF | — |
| Cortex-A53 CPU0 | OFF | ON | — |
| Cortex-A53 CPU1/2/3 | OFF | OFF | — |
| Cortex-A53 SCU | OFF | ON | — |
| 3DG-A/B/C/D/E | OFF | OFF | — |
| A3IR/A3VP/A3VC/A2VC0/A2VC1 | OFF | OFF | — |

Table 14.6 Initial state of power domains (RZ/G2E)

| Domain name | Cortex-A53 Boot | Boot | Remark |
|-----------------|-----------------|------|--------|
| Cortex-A53 CPU0 | ON | OFF | — |
| Cortex-A53 CPU1 | OFF | OFF | — |
| Cortex-A53 SCU | ON | OFF | — |
| A3VC/A2VC1 | OFF | OFF | — |
| 3DG-A/B | OFF | OFF | — |

14.3.3 Power Control of Arm CPUs

The AP system cores (Cortex-A53 and Cortex-A57) can be powered off by executing the WFI instruction. Similarly, power of Arm CPUs can be resumed either by controlling the APMU (Advanced Power Management Unit) registers or by an IRQ or FIQ interrupt.

For detail of power control by APMU registers, see section 13, Advanced Power Management Unit for AP-System Core (APMU).

(1) Power Control by WFI Instruction and Interrupts

The power shutdown by the WFI instruction is performed only when Cortex-A57/ Cortex-A53 CPU_n power status control register (CA57CPU_nCR, CA53CPU_nCR) are in the core standby mode. In case of other setting, power shutdown sequence is not activated by the WFI instruction. See section 13, Advanced Power Management Unit for AP-System Core (APMU).

Power can be resumed by an IRQ or FIQ interrupt on the Arm core. After resuming power of Cortex-A57, Cortex-A53, CPUs by IRQ/FIQ, read the value of SYSCEERSR, SYSCEERSR 2, and SYSCEERSR3 registers. Without reading those registers after power-resuming, the shutdown sequence of related CPUs will not start.

14.3.4 Power Control of non Arm CPU modules

Power control on modules other than the Arm CPUs can be performed by SYSC registers. Following figure describes power control reference flowchart.

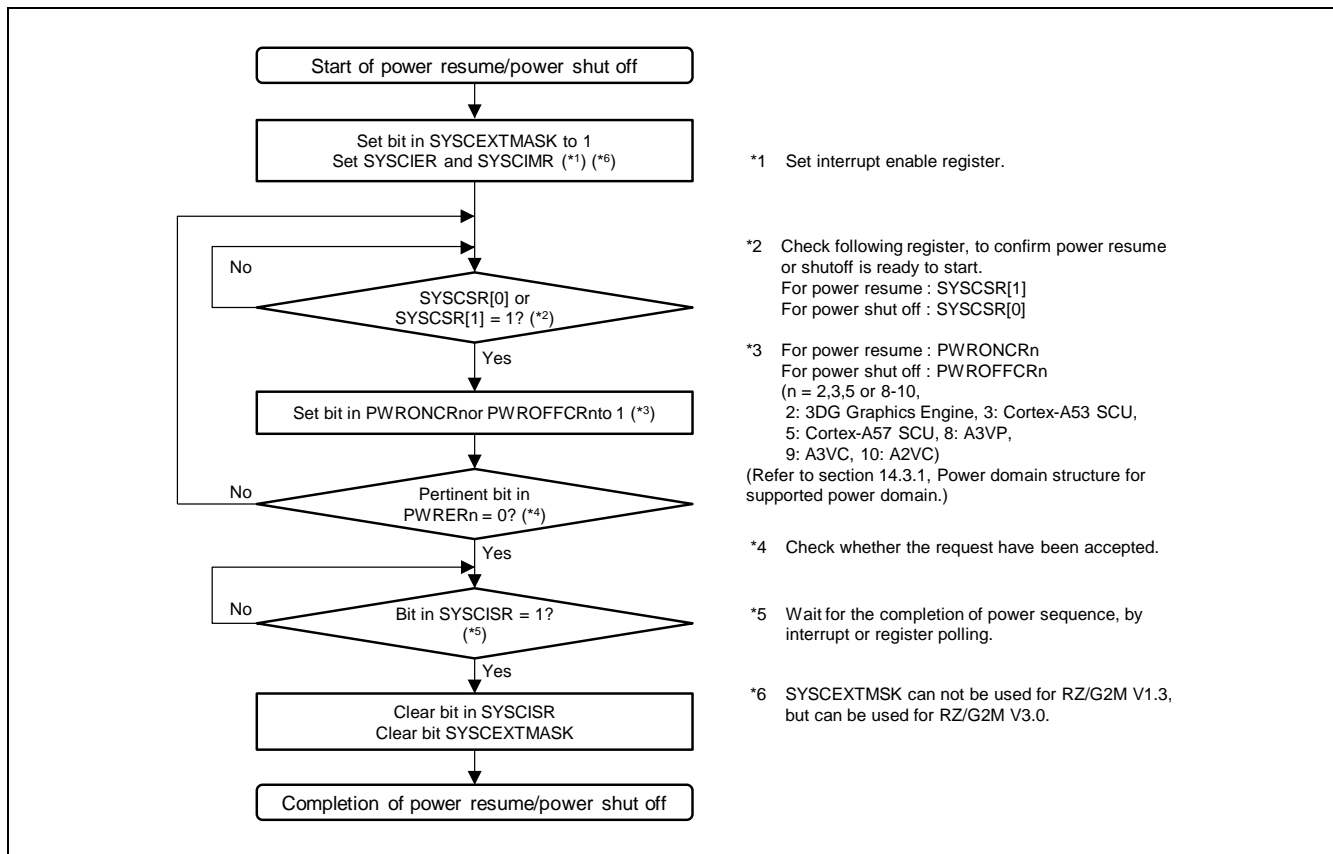


Figure 14.7 Power Resume/Shutoff Flowchart for non Arm CPU domain by SYSC Registers

14.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

1. Only one power domain is allowed to set for power shut off or power resume. Setting for more than 1 power domain at the same time is prohibited. Therefore, setting for power shut off or power resume of 3D Graphics Engine, A2VC0/1, multiple bits cannot be set but single bit must be set in turn, one by one. The order of setting those bits must follow Power domain structure (refer to 14.3.1 Power domain structure)
2. For a module for which power is shut off, register settings and memory contents are not retained. Required information should be saved before power is shut off.
3. It takes several hundreds of microseconds to shutting off and resuming power domain. Because actual time required for shutting off and resuming depends on the status of on-board power line, shutoff/resume time is not guaranteed by electrical specification.
4. Before shutting off one (or more) of the power domain of A3IR, A3VC and, A3VP, set all modules to the Module Standby state for which belong to the power domain. Refer to Figure 14.8.

Note: Power domains for each product;

[RZ/G2H] A3IR, A3VC, A3VP

[RZ/G2M V1.3, RZ/G2M V3.0] A3IR, A3VC,

[RZ/G2N] A3VC, A3VP

[RZ/G2E] A3VC

For the module standby capability of each module, refer to section 12, Module Standby, Software Reset.

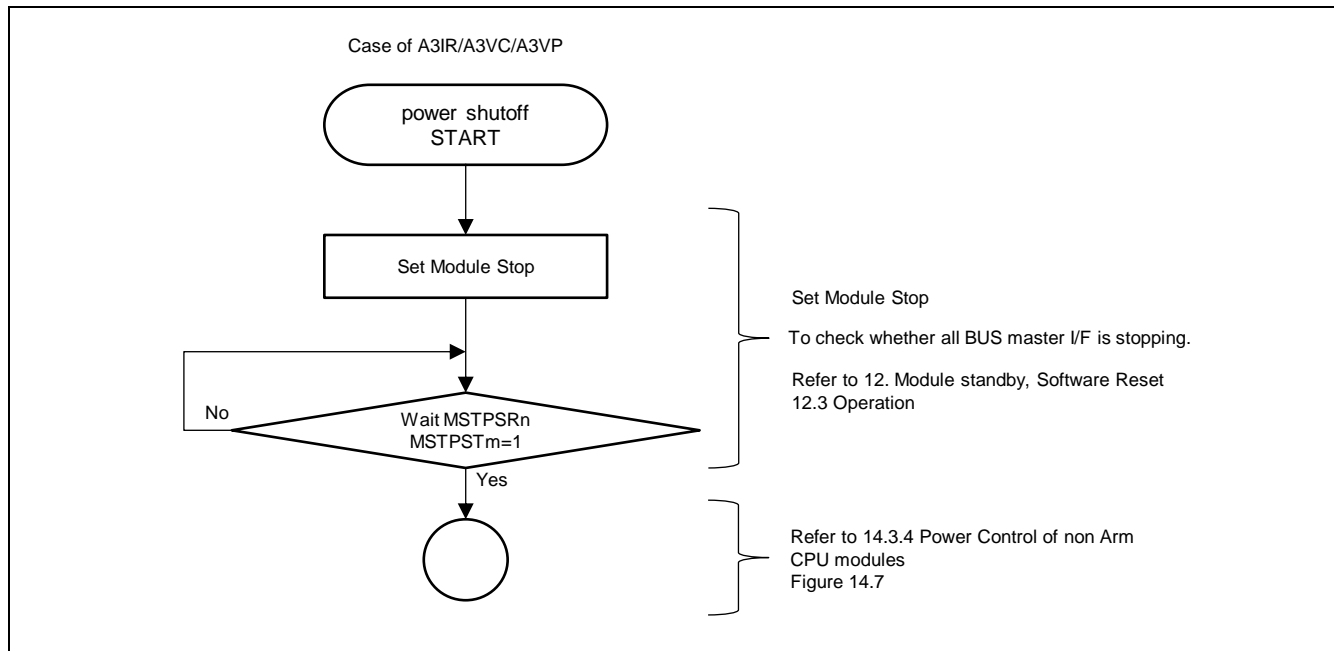


Figure 14.8 Set Module Standby for A3IR/A3VC/A3VP

6. [RZ/G2M V1.3] (not RZ/G2M V3.0) During power domains shut off or power resume sequence by SYSC register, do not use L2 shutdown mode by CMPWR on CA53CPUCMCR or CA57CPUCMCR in APMU.

15. Thermal Sensor/Chip Internal Voltage Monitor (THS/CIVM)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

15.1 Overview

This LSI provides a thermal sensor module that measures the temperature (T_j) inside the LSI. The thermal sensor module also includes a chip internal voltage monitoring module that measures the supply voltage inside the LSI.

15.1.1 Features

- (1) Each of the thermal sensors measures temperature T_j with an accuracy of $\pm 5^\circ\text{C}$ over the range of temperatures from -40°C to 115°C .
- (2) RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N provide three thermal sensor modules (THS1/CIVM1, THS2/CIVM2 and THS3/CIVM3). RZ/G2M V1.3 and RZ/G2M V3.0 provide two thermal sensor modules (THS1/CIVM1 and THS2/CIVM2).
- (3) This module can generate interrupts when the detected temperature T_j within the LSI rises above or falls below a specified temperature.
- (4) Each of the chip internal voltage monitors in the thermal sensor module measures supply voltage inside the LSI with an accuracy of $\pm 60\text{mV}$.
- (5) RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N can monitor VDD_DVFS supply voltage. RZ/G2N can monitor VDD supply voltage.

15.1.2 Block Diagram

A block diagram of the thermal sensor modules is shown in Figure 15.1 (for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N). The thermal sensor module consists of the thermal sensors (THS), chip internal voltage monitors (CIVM) and the thermal sensor controller (TSC). THS measures T_j temperature, and CIVM measures a voltage. TSC controls THS and CIVM.

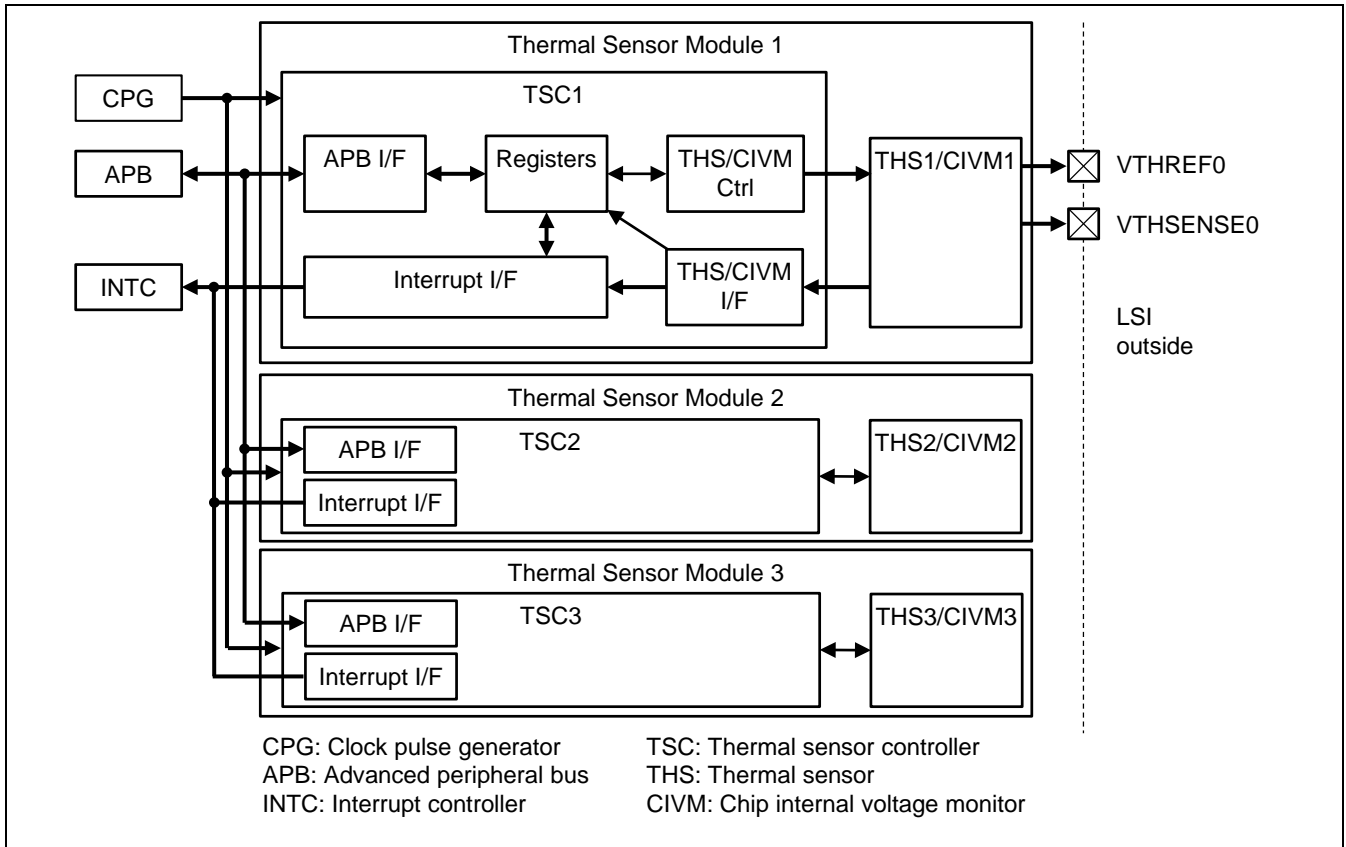


Figure 15.1 Block Diagram of Thermal Sensor Modules (for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N)

15.1.3 External Pins

Table 15.1 is a list of external pins. For the processing of these pins, refer to 15.4 Usage Note.

Table 15.1 Pin Configuration

| Pin Name | I/O | Function | Second Generation RZ/G Series Products | | | |
|-----------|---------------|-----------------------------------|---|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| VTHSENSE0 | Analog I/O | Outputs a voltage that 0 to 0.9V. | √ | √ | √ | — |
| VTHREF0 | Analog I/O | Outputs a voltage that 0 to 1.3V. | √ | √ | √ | — |

15.1.4 Register Configuration

Table 15.2 is a list of registers. The base addresses for the registers of each of the thermal sensor controller are as follows; H'E619_8000 for TSC1, H'E61A_0000 for TSC2, and H'E61A_8000 for TSC3. Any address other than those listed in the table must not be written to. Otherwise, correct operation is not guaranteed.

Table 15.2 List of Registers of the THS/CIVM

| Register Name | Abbreviation | R/W | Address | | | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|----------------------------------|--------------|---------------|-----------------|-----------------|-----------------|--------------------|-------------|---|----------------------------|--------|--------|
| | | | TSC1 | TSC2 | TSC3 | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Interrupt Status Register | IRQSTR | R/(WC0)*1 | H'E619_800 4 | H'E61A_00 04 | H'E61A_80 04 | H'0000_0000 | 32 | √ | √ | √ | — |
| Interrupt Mask Register | IRQMSK | R/(W)*1 | H'E619_800 8 | H'E61A_00 08 | H'E61A_80 08 | H'0000_0000 | 32 | √ | √ | √ | — |
| Threshold Edge/Level Register | IRQCTL | R/(W)*1 | H'E619_800 C | H'E61A_00 0C | H'E61A_80 0C | H'0000_0000 | 32 | √ | √ | √ | — |
| Interrupt Enable Register | IRQEN | R/(W)*1 | H'E619_801 0 | H'E61A_00 10 | H'E61A_80 10 | H'0000_0000 | 32 | √ | √ | √ | — |
| Interrupt Temperature 1 Register | IRQTEMP1 | R/(W)*1 | H'E619_801 4 | H'E61A_00 14 | H'E61A_80 14 | H'0000_0000 | 32 | √ | √ | √ | — |
| Interrupt Temperature 2 Register | IRQTEMP2 | R/(W)*1 | H'E619_801 8 | H'E61A_00 18 | H'E61A_80 18 | H'0000_0000 | 32 | √ | √ | √ | — |
| Interrupt Temperature 3 Register | IRQTEMP3 | R/(W)*1 | H'E619_801 C | H'E61A_00 1C | H'E61A_80 1C | H'0000_0000 | 32 | √ | √ | √ | — |
| Control Register | THCTR | R/(W)*1 *2 | H'E619_802 0 | H'E61A_00 20 | H'E61A_80 20 | Undefined value | 32 | √ | √ | √ | — |
| Status Register | THSTR | R/(WC0)*1 | H'E619_802 4 | H'E61A_00 24 | H'E61A_80 24 | H'0000_0000 | 32 | √ | √ | √ | — |
| Temperature Register | TEMP | R | H'E619_802 8 | H'E61A_00 28 | H'E61A_80 28 | Undefined value | 32 | √ | √ | √ | — |

| Register Name | Abbreviation | R/W | Address | | | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|--|--------------|-----|-----------------|-----------------|-----------------|-----------------|-------------|--|----------------------------|--------|--------|
| | | | TSC1 | TSC2 | TSC3 | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Voltage Register | VOLT | R | H'E619_802 C | H'E61A_00 2C | H'E61A_80 2C | H'0000_0000 | 32 | √ | √ | √ | — |
| THCODE Parameter1 Register | THCODE1 | R | H'E619_805 0 | H'E61A_00 50 | H'E61A_80 50 | Undefined value | 32 | √ | √ | √ | — |
| THCODE Parameter2 Register | THCODE2 | R | H'E619_805 4 | H'E61A_00 54 | H'E61A_80 54 | Undefined value | 32 | √ | √ | √ | — |
| THCODE Parameter3 Register | THCODE3 | R | H'E619_805 8 | H'E61A_00 58 | H'E61A_80 58 | Undefined value | 32 | √ | √ | √ | — |
| PTAT Parameter1 Register | PTAT1 | R | H'E619_805 C | — | — | Undefined value | 32 | √ | √ | √ | — |
| PTAT Parameter2 Register | PTAT2 | R | H'E619_806 0 | — | — | Undefined value | 32 | √ | √ | √ | — |
| PTAT Parameter3 Register | PTAT3 | R | H'E619_806 4 | — | — | Undefined value | 32 | √ | √ | √ | — |
| Software Correction Parameter Register | THSCP | R | H'E619_806 8 | — | — | Undefined value | 32 | √ | √ | √ | — |

Notes: 1. several bits are read-only bits.
2. several bits can be written 0 only.

15.1.5 Connected Module

Table 15.3 is a list of the modules that are connected to this module.

Table 15.3 Connected Modules

| Module Name | Connected Module Name | Connected Module Function |
|-------------|-----------------------|----------------------------|
| THS/CIVM | APB | Control to register access |
| | CPG | Output clock |
| | INTC | Control to interrupts |
| | Module standby | Control to stop clock |
| | Software reset | Execute software reset |

15.2 Register Description

[Legend for register description]

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R: Read-only. The write value should always be 0.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

W: Write-only. Reading this bit is prohibited. When the bit is reserved, the write value should always be 0.

__W: Write-only. The read value is undefined.

15.2.1 Interrupt Status Register (IRQSTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|------------|------------|------------|-----------|-----------|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | TEMPD3_STR | TEMPD2_STR | TEMPD1_STR | TEMP3_STR | TEMP2_STR | TEMP1_STR |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/WC0 | R/WC0 | R/WC0 | R/WC0 | R/WC0 | R/WC0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-------|--|
| 31 to 6 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 5 | TEMPD3_STR | B'0 | R/WC0 | TEMPD3 Detection Status 0: Not detected 1: TEMP_CODE [11:0] bits detect that it falls below the temperature set in IRQTEMP3. |
| 4 | TEMPD2_STR | B'0 | R/WC0 | TEMPD2 Detection Status 0: Not detected 1: TEMP_CODE [11:0] bits detect that it falls below the temperature set in IRQTEMP2. |
| 3 | TEMPD1_STR | B'0 | R/WC0 | TEMPD1 Detection Status 0: Not detected 1: TEMP_CODE [11:0] bits detect that it falls below the temperature set in IRQTEMP1. |
| 2 | TEMP3_STR | B'0 | R/WC0 | TEMP3 Detection Status 0: Not detected 1: TEMP_CODE [11:0] bits detect that it exceeds the temperature set in IRQTEMP3. |
| 1 | TEMP2_STR | B'0 | R/WC0 | TEMP2 Detection Status 0: Not detected 1: TEMP_CODE [11:0] bits detect that it exceeds the temperature set in IRQTEMP2. |
| 0 | TEMP1_STR | B'0 | R/WC0 | TEMP1 Detection Status 0: Not detected 1: TEMP_CODE [11:0] bits detect that it exceeds the temperature set in IRQTEMP1. |

15.2.2 Interrupt Mask Register (IRQMSK)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|------------|------------|------------|-----------|-----------|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | TEMPD3_MSK | TEMPD2_MSK | TEMPD1_MSK | TEMP3_MSK | TEMP2_MSK | TEMP1_MSK |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 to 6 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 5 | TEMPD3_MSK | B'0 | R/W | This bit selects masking or non-masking of TEMPD3 interrupt requests. 0: Interrupts are masked. 1: The mask is cleared. |
| 4 | TEMPD2_MSK | B'0 | R/W | This bit selects masking or non-masking of TEMPD2 interrupt requests. 0: Interrupts are masked. 1: The mask is cleared. |
| 3 | TEMPD1_MSK | B'0 | R/W | This bit selects masking or non-masking of TEMPD1 interrupt requests. 0: Interrupts are masked. 1: The mask is cleared. |
| 2 | TEMP3_MSK | B'0 | R/W | This bit selects masking or non-masking of TEMP3 interrupt requests. 0: Interrupts are masked. 1: The mask is cleared. |
| 1 | TEMP2_MSK | B'0 | R/W | This bit selects masking or non-masking of TEMP2 interrupt requests. 0: Interrupts are masked. 1: The mask is cleared. |
| 0 | TEMP1_MSK | B'0 | R/W | This bit selects masking or non-masking of TEMP1 interrupt requests. 0: Interrupts are masked. 1: The mask is cleared. |

15.2.3 Threshold Edge/Level Register (IRQCTL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-----------|-----------|-----------|----------|----------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | TEMPD3_EL | TEMPD2_EL | TEMPD1_EL | TEMP3_EL | TEMP2_EL | TEMP1_EL |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 6 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 5 | TEMPD3_EL | B'0 | R/W | Specifies the method for detection of the TEMPD3 interrupt signal input. 0: Edge detection 1: Level detection (setting prohibited) |
| 4 | TEMPD2_EL | B'0 | R/W | Specifies the method for detection of the TEMPD2 interrupt signal input. 0: Edge detection 1: Level detection (setting prohibited) |
| 3 | TEMPD1_EL | B'0 | R/W | Specifies the method for detection of the TEMPD1 interrupt signal input. 0: Edge detection 1: Level detection (setting prohibited) |
| 2 | TEMP3_EL | B'0 | R/W | Specifies the method for detection of the TEMP3 interrupt signal input. 0: Edge detection 1: Level detection (setting prohibited) |
| 1 | TEMP2_EL | B'0 | R/W | Specifies the method for detection of the TEMP2 interrupt signal input. 0: Edge detection 1: Level detection (setting prohibited) |
| 0 | TEMP1_EL | B'0 | R/W | Specifies the method for detection of the TEMP1 interrupt signal input. 0: Edge detection 1: Level detection (setting prohibited) |

15.2.4 Interrupt Enable Register (IRQEN)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-----------|-----------|-----------|----------|----------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | TEMPD3_EN | TEMPD2_EN | TEMPD1_EN | TEMP3_EN | TEMP2_EN | TEMP1_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 6 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 5 | TEMPD3_EN | B'0 | R/W | This bit enables or disables TEMPD3 interrupts. 0: TEMPD3 interrupts are disabled. 1: TEMPD3 interrupts are enabled. |
| 4 | TEMPD2_EN | B'0 | R/W | This bit enables or disables TEMPD2 interrupts. 0: TEMPD2 interrupts are disabled. 1: TEMPD2 interrupts are enabled. |
| 3 | TEMPD1_EN | B'0 | R/W | This bit enables or disables TEMPD1 interrupts. 0: TEMPD1 interrupts are disabled. 1: TEMPD1 interrupts are enabled. |
| 2 | TEMP3_EN | B'0 | R/W | This bit enables or disables TEMP3 interrupts. 0: TEMP3 interrupts are disabled. 1: TEMP3 interrupts are enabled. |
| 1 | TEMP2_EN | B'0 | R/W | This bit enables or disables TEMP2 interrupts. 0: TEMP2 interrupts are disabled. 1: TEMP2 interrupts are enabled. |
| 0 | TEMP1_EN | B'0 | R/W | This bit enables or disables TEMP1 interrupts. 0: TEMP1 interrupts are disabled. 1: TEMP1 interrupts are enabled. |

15.2.5 Interrupt Temperature 1 Register (IRQTEMP1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | IRQTEMP1[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 to 12 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 11 to 0 | IRQTEMP1 [11:0] | H'000 | R/W | These bits set the first threshold value of the temperature for TEMP1 and TEMPD1 interrupts. Refer to 15.3.1.1 section about the setting value of IRQTEMP1[11:0]. |

15.2.6 Interrupt Temperature 2 Register (IRQTEMP2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | IRQTEMP2[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 11 to 0 | IRQTEMP2 [11:0] | H'000 | R/W | These bits set the second threshold value of the temperature for TEMP2 and TEMPD2 interrupts. Refer to 15.3.1.1 section about the setting value of IRQTEMP2[11:0]. |

15.2.7 Interrupt Temperature 3 Register (IRQTEMP3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | IRQTEMP3[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 to 12 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 11 to 0 | IRQTEMP3 [11:0] | H'000 | R/W | These bits set the third threshold value of the temperature for TEMP3 and TEMPD3 interrupts. Refer to 15.3.1.1 section about the setting value of IRQTEMP3[11:0]. |

15.2.8 Control Register (THCTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|------|---------------|-----|------|------|-----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | AOUT | SENSSEL [1:0] | — | PONM | THEN | — | — | — | — | — | THSST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R | R/W | R/W | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 to 11 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 10 | AOUT | B'0 | R/W | Enabling/disabling of external output of thermal sensor 0: Enabled 1: Disabled This bit is only valid in THS1 |
| 9, 8 | SENSSEL [1:0] | B'00 | R/W | Select the use sensor B'00: THS, CIVM both ON B'01: THS ON only B'10: CIVM ON only B'11: THS, CIVM both ON (this setting is the same as 00) |
| 7 | — | B'0 | R | Reserved This bit is fixed to 0. |
| 6 | PONM | — | R/W | Select mode signal 0: Normal mode 1: Power-on mode When using Thermal Sensor or Chip Internal Voltage monitor, Be sure to write 0. |
| 5 | THEN | B'0 | R/W | Enabling/disabling of thermal sensor 0: Enabled 1: Disabled |
| 4 to 1 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 0 | THSST | B'0 | R/W | Enabling/disabling of the A/D converter for the thermal sensor 0: Disabled 1: Enabled |

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|--------------|-----|-----|----|------|--------------|-----|----|------|------|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | VOLSELB[2:0] | | | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | VOLSELA[2:0] | | | — | AOUT | SENSEL [1:0] | | — | PONM | THEN | — | — | — | — | THSST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 | — | B'0 | R | Reserved This bit is fixed to 0. |
| 30 to 28 | VOLSELB [2:0] | All 0 | R/W | Same value with VOLSELA[2:0] should be set in these bits to select monitoring voltage. B'000: VDD B'001: VDD_DVFS Other: prohibited |
| 27 to 15 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 14 to 12 | VOLSELA [2:0] | All 0 | R/W | Same value with VOLSELB[2:0] should be set in these bits to select monitoring voltage. B'000: VDD B'001: VDD_DVFS Other: prohibited |
| 11 | — | B'0 | R | Reserved This bit is fixed to 0. |
| 10 | AOUT | B'0 | R/W | Enabling/disabling of external output of thermal sensor 0: Enabled 1: Disabled This bit is only valid in THS1 |
| 9, 8 | SENSEL [1:0] | B'00 | R/W | Select the use sensor B'00: THS, CIVM both ON B'01: THS ON only B'10: CIVM ON only B'11: THS, CIVM both ON (this setting is the same as 00) |
| 7 | — | B'0 | R | Reserved This bit is fixed to 0. |
| 6 | PONM | — | R/W | Select mode signal 0: Normal mode 1: Power-on mode When using Thermal Sensor or Chip Internal Voltage monitor, Be sure to write 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 5 | THEN | B'0 | R/W | Enabling/disabling of thermal sensor 0: Enabled 1: Disabled |
| 4 to 1 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 0 | THSST | B'0 | R/W | Enabling/disabling of the A/D converter for the thermal sensor 0: Disabled 1: Enabled |

15.2.9 Status Register (THSTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-------------|-------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | THCNT OV | THFAIL 1 | THFAIL 0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/WC0 | R/WC0 | R/WC0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-------|---|
| 31 to 3 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 2 | THCNTOV | B'0 | R/WC0 | This bit is detected the overflow of counter. 0: Not detected 1: Counter has overflowed. |
| 1 | THFAIL1 | B'0 | R/WC0 | This bit is set to 1 when all of the TEMP_CODE [11:0] bits are 1 0: Normal operation 1: All of the TEMP_CODE[11:0] bits are all 1 |
| 0 | THFAIL0 | B'0 | R/WC0 | This bit is set to 1 when all of the TEMP_CODE [11:0] bits are 0 0: Normal operation 1: All of the TEMP_CODE[11:0] bits are all 0 |

15.2.10 Temperature Register (TEMP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-----------------|----|----|----|----|----|----|----|----|----|----|----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | TEMP_CODE[11:0] | | | | | | | | | | | | — |
| Initial value: | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — | — | — | — | — | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 11 to 0 | TEMP_CODE [11:0] | — | R | These bits indicate the digital value for the temperature detected by the thermal sensor. Refer to 15.3.1.2 (1) section for the conversion method from TEMP_CODE[11:0] to temperature. |

15.2.11 Voltage Register (VOLT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----------------|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | VOLT_CODE[9:0] | | | | | | | | | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|---|
| 31 to 10 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 9 to 0 | VOLT_CODE [9:0] | H'000 | R | These bits indicate the digital value for the voltage detected by the chip internal voltage monitor. When calculating the voltage, use its average value which read this value 5 times at a period of 500 us. Refer to 15.3.1.2 (2) section for the conversion method from VOLT_CODE[9:0] to voltage. |

15.2.12 THCODE Parameter1 Register (THCODE1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|---------------|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | THCODE1[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 11 to 0 | THCODE1 [11:0] | — | R | Parameter to be used in adjusting the characteristics |

15.2.13 THCODE Parameter2 Register (THCODE2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|---------------|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | THCODE2[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 11 to 0 | THCODE2 [11:0] | — | R | Parameter to be used in adjusting the characteristics |

15.2.14 THCODE Parameter3 Register (THCODE3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|---------------|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | THCODE3[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 11 to 0 | THCODE3 [11:0] | — | R | Parameter to be used in adjusting the characteristics |

15.2.15 PTAT Parameter1 Register (PTAT1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-------------|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | PTAT1[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 11 to 0 | PTAT1[11:0] | — | R | Parameter to be used in adjusting the characteristics |

15.2.16 PTAT Parameter2 Register (PTAT2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-------------|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | PTAT2[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 11 to 0 | PTAT2[11:0] | — | R | Parameter to be used in adjusting the characteristics |

15.2.17 PTAT Parameter3 Register (PTAT3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-------------|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | PTAT3[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 11 to 0 | PTAT3[11:0] | — | R | Parameter to be used in adjusting the characteristics |

15.2.18 Software Correction Parameter Register (THSCP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|------------------|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | COR_PAR_A_VLD[1] | COR_PAR_A_VLD[0] | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are fixed to 0. |
| 15, 14 | COR_PAR_A_VLD[1:0] | — | R | Check the valid value of adjusting parameter. B'00, B'01, B'10: Fixed parameters B'11: Hardware parameters (Read THCODE1 to 3 and PTAT1 to 3 registers) |
| 13 to 0 | — | All 0 | R | Reserved These bits are fixed to 0. |

15.3 Operation

RZ/G2H

RZ/G2M V1.3

RZ/G2M V3.0

RZ/G2N

RZ/G2E

Note: For the RZ/G2M V3.0, following calculation example of the RZ/G2M V1.3 is reference only and values in each equation item will be changed significantly.

15.3.1 The Initial Sequence of Thermal Sensor/Chip Internal Voltage Monitor

Figure 15.2 shows the operation flow for thermal sensor and chip internal voltage monitor.

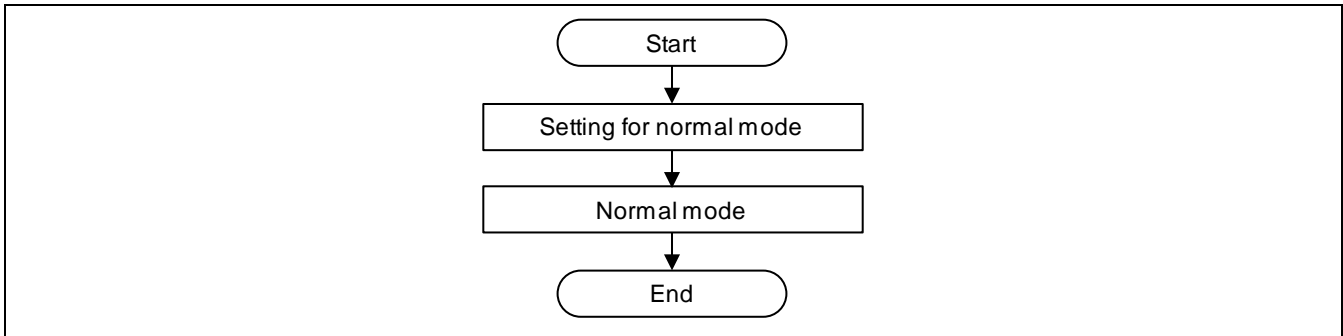


Figure 15.2 The Operation Flow for Thermal Sensor and Chip Internal Voltage Monitor

15.3.1.1 Setting of Normal Mode

Figure 15.3 shows initial sequence for normal mode of thermal sensor and chip internal voltage monitor. IRQTEMP1 to 3[11:0] calculate by using below.

First, calculate T_{j_T} . It is

- RZ/G2M V1.3

$$T_{j_T} = \frac{(PTAT2[11:0] - PTAT3[11:0]) \times 157}{(PTAT1[11:0] - PTAT3[11:0])} - 41 \approx 20 \text{ to } 40 \text{ [}^\circ\text{C]}$$

- Except for RZ/G2M V1.3

$$T_{j_T} = \frac{(PTAT2[11:0] - PTAT3[11:0]) \times 167}{(PTAT1[11:0] - PTAT3[11:0])} - 41 \approx 20 \text{ to } 40 \text{ [}^\circ\text{C]}$$

Next, calculate IRQTEMP1 to 3[11:0]. If setting temperature, STEMP, is less than T_{j_T} ,

- RZ/G2M V1.3

$$IRQTEMP1 \text{ to } 3[11:0] = THCODE3[11:0] - \frac{(STEMP + 41)(PTAT3[11:0] - PTAT1[11:0])(THCODE3[11:0] - THCODE2[11:0])}{157 \times (PTAT3[11:0] - PTAT2[11:0])}$$

- Except for RZ/G2M V1.3

$$IRQTEMP1 \text{ to } 3[11:0] = THCODE3[11:0] - \frac{(STEMP + 41)(PTAT3[11:0] - PTAT1[11:0])(THCODE3[11:0] - THCODE2[11:0])}{167 \times (PTAT3[11:0] - PTAT2[11:0])}$$

If setting temperature, STEMP, is more than T_{j_T} ,

- RZ/G2M V1.3

$$IRQTEMP1 \text{ to } 3[11:0] = THCODE1[11:0] - \frac{(STEMP - 116)(PTAT1[11:0] - PTAT3[11:0])(THCODE2[11:0] - THCODE1[11:0])}{157 \times (PTAT1[11:0] - PTAT2[11:0])}$$

- Except for RZ/G2M V1.3

$$IRQTEMP1 \text{ to } 3[11:0] = THCODE1[11:0] - \frac{(STEMP - 126)(PTAT1[11:0] - PTAT3[11:0])(THCODE2[11:0] - THCODE1[11:0])}{167 \times (PTAT1[11:0] - PTAT2[11:0])}$$

The following is an example of RZ/G2M V1.3.

When THCODE1[11:0] = 3397, THCODE2[11:0] = 2800, THCODE3[11:0] = 2221, PTAT1[11:0] = 2631, PTAT2[11:0] = 1509, PTAT3[11:0] = 435 and STEMP = 110[°C], calculate IRQTEMP1[11:0],

T_{j_T} is

$$T_{j_T} = \frac{(1509-435) \times 157}{(2631-435)} - 41 \approx 35.8 \text{ [}^\circ\text{C]}.$$

IRQTEMP1[11:0] are

$$\text{IRQTEMP1}[11:0] = 3397 - \frac{(110-116)(2631-435)(2800-3397)}{157 \times (2631-1509)} \approx 3352,$$

because the setting temperature, STEMP (110[°C]), is more than T_{j_T}.

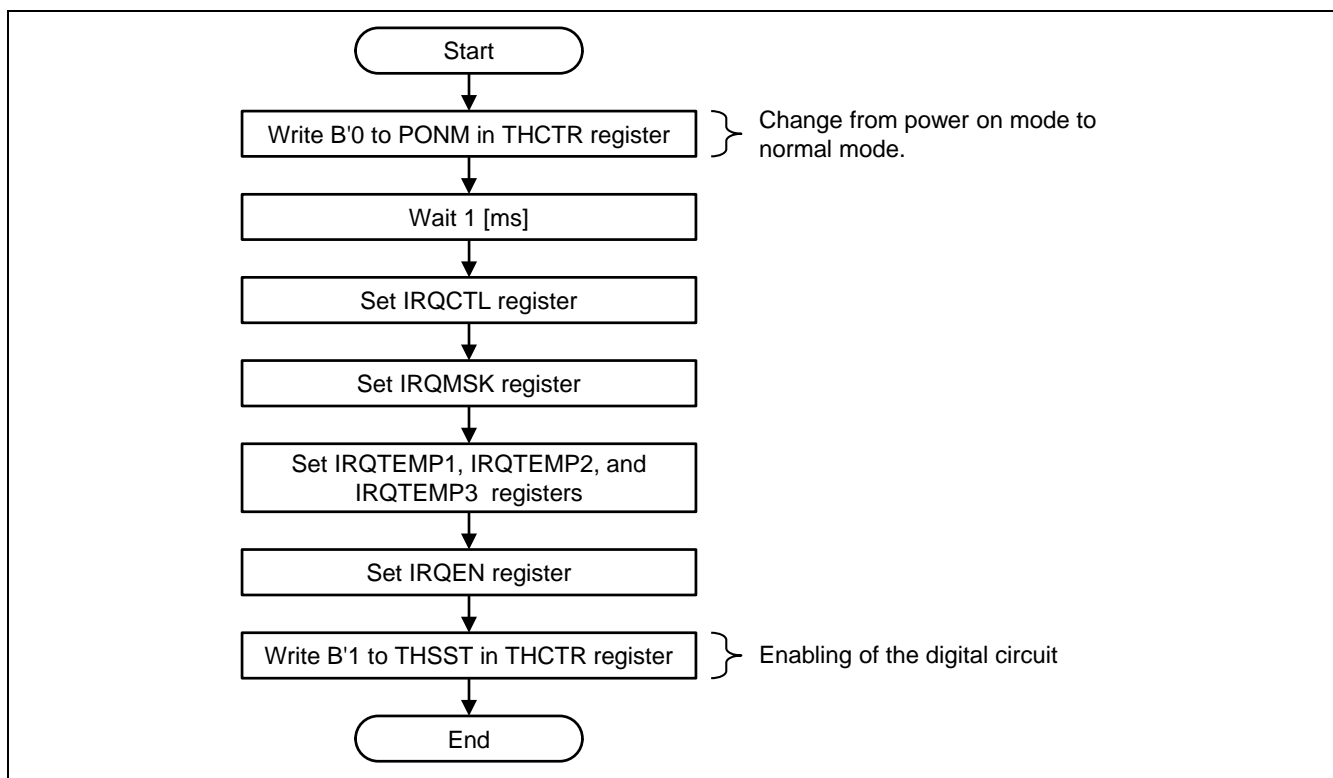


Figure 15.3 Initial Sequence for Normal Mode of Thermal Sensor and Chip Voltage Monitor

15.3.1.2 Normal Mode

Normal mode of thermal sensor module outputs the digital codes of temperature and voltage which are measured by thermal sensor module at a constant period. The conversion method from these codes to temperature and voltage are shown below.

(1) The conversion method from TEMP_CODE[11:0] to temperature

When TEMP_CODE[11:0] < THCODE2[11:0]

- RZ/G2M V1.3

$$\text{Temperature}[\text{°C}] = \frac{157 \times (\text{PTAT3}[11:0] - \text{PTAT2}[11:0]) (\text{THCODE3}[11:0] - \text{TEMP_CODE}[11:0])}{(\text{PTAT3}[11:0] - \text{PTAT1}[11:0]) (\text{THCODE3}[11:0] - \text{THCODE2}[11:0])} - 41$$

- Except for RZ/G2M V1.3

$$\text{Temperature}[\text{°C}] = \frac{167 \times (\text{PTAT3}[11:0] - \text{PTAT2}[11:0]) (\text{THCODE3}[11:0] - \text{TEMP_CODE}[11:0])}{(\text{PTAT3}[11:0] - \text{PTAT1}[11:0]) (\text{THCODE3}[11:0] - \text{THCODE2}[11:0])} - 41$$

When TEMP_CODE[11:0] ≥ THCODE2[11:0]

- RZ/G2M V1.3

$$\text{Temperature}[\text{°C}] = \frac{157 \times (\text{PTAT1}[11:0] - \text{PTAT2}[11:0]) (\text{THCODE1}[11:0] - \text{TEMP_CODE}[11:0])}{(\text{PTAT1}[11:0] - \text{PTAT3}[11:0]) (\text{THCODE2}[11:0] - \text{THCODE1}[11:0])} + 116$$

- Except for RZ/G2M V1.3

$$\text{Temperature}[\text{°C}] = \frac{167 \times (\text{PTAT1}[11:0] - \text{PTAT2}[11:0]) (\text{THCODE1}[11:0] - \text{TEMP_CODE}[11:0])}{(\text{PTAT1}[11:0] - \text{PTAT3}[11:0]) (\text{THCODE2}[11:0] - \text{THCODE1}[11:0])} + 126$$

The followings are examples of RZ/G2M V1.3

First example, when THCODE1[11:0] = 3397, THCODE2[11:0] = 2800, THCODE3[11:0] = 2221, PTAT1[11:0] = 2631, PTAT2[11:0] = 1509, PTAT3[11:0] = 435 and TEMP_CODE[11:0] = 2680, the temperature, Temperature[°C], is

$$\text{Temperature}[\text{°C}] = \frac{157 \times (435 - 1509) (2221 - 2680)}{(435 - 2631) (2221 - 2800)} - 41 \approx 19.9[\text{°C}]$$

because TEMP_CODE[11:0] = 2680 < THCODE2[11:0] = 2800.

Second example, THCODE1 to 3[11:0] and PTAT1 to 3[11:0] are same with the first example. when TEMP_CODE[11:0] = 3350, the temperature, Temperature[°C], is

$$\text{Temperature}[\text{°C}] = \frac{157 \times (2631 - 1509) (3397 - 3350)}{(2631 - 435) (2800 - 3397)} + 116 \approx 109.7[\text{°C}]$$

because TEMP_CODE[11:0] = 3350 ≥ THCODE2[11:0] = 2800.

(2) The conversion method from VOLT_CODE[9:0] to voltage

VOLT_CODE[9:0] can convert to voltage[mV] by the following formula.

- RZ/G2M V1.3 (VDD_DVFS)

$$\text{Voltage}[\text{mV}] = \frac{(360) \times (\text{THCODE3}[11:0] - 4 \times \text{VOLT_CODE}[9:0])}{(\text{THCODE3}[11:0] - \text{THCODE1}[11:0])} + 630 + \text{Temperature}[\text{°C}] \times 0.25$$

- RZ/G2H (VDD_DVFS), RZ/G2N (VDD_DVFS, VDD)

$$\text{Voltage}[\text{mV}] = \frac{(390) \times (\text{THCODE3}[11:0] - 4 \times \text{VOLT_CODE}[9:0])}{(\text{THCODE3}[11:0] - \text{THCODE1}[11:0])} + 630 + \text{Temperature}[\text{°C}] \times 0.1$$

The following is an example of RZ/G2M V1.3

When THCODE1[11:0] = 3397, THCODE3[11:0] = 2221, VOLT_CODE[9:0] = 867 and Temperature[°C] = 19.9[°C], Voltage[mV] is

$$\text{Voltage}[\text{mV}] = \frac{(360) \times (2221 - 4 \times 867)}{(2221 - 3397)} + 630 + 19.9 \times 0.25 \approx 1017[\text{mV}]$$

15.3.2 Interrupt

The block diagrams of interrupt are shown in Figure 15.4(for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N). The interrupts for thermal sensor module have three (Thermal Sensor ch0, Thermal Sensor ch1 and Thermal Sensor ch2). These interrupts receive from TSC1, TSC2 and TSC3. The interrupts for TSC1, TSC2 and TSC3 can detect to exceed (TEMP1, TEMP2 and TEMP3) or to fall below (TEMPD1, TEMPD2 and TEMPD3), and the interrupt signal of its module is set to 1 by is detected each of TEMP or TEMPD.

For instance, if TEMP2 of TSC2 is detected, THS2_IRQ2 is set to 1 and Thermal sensor.ch1 is set to 1.

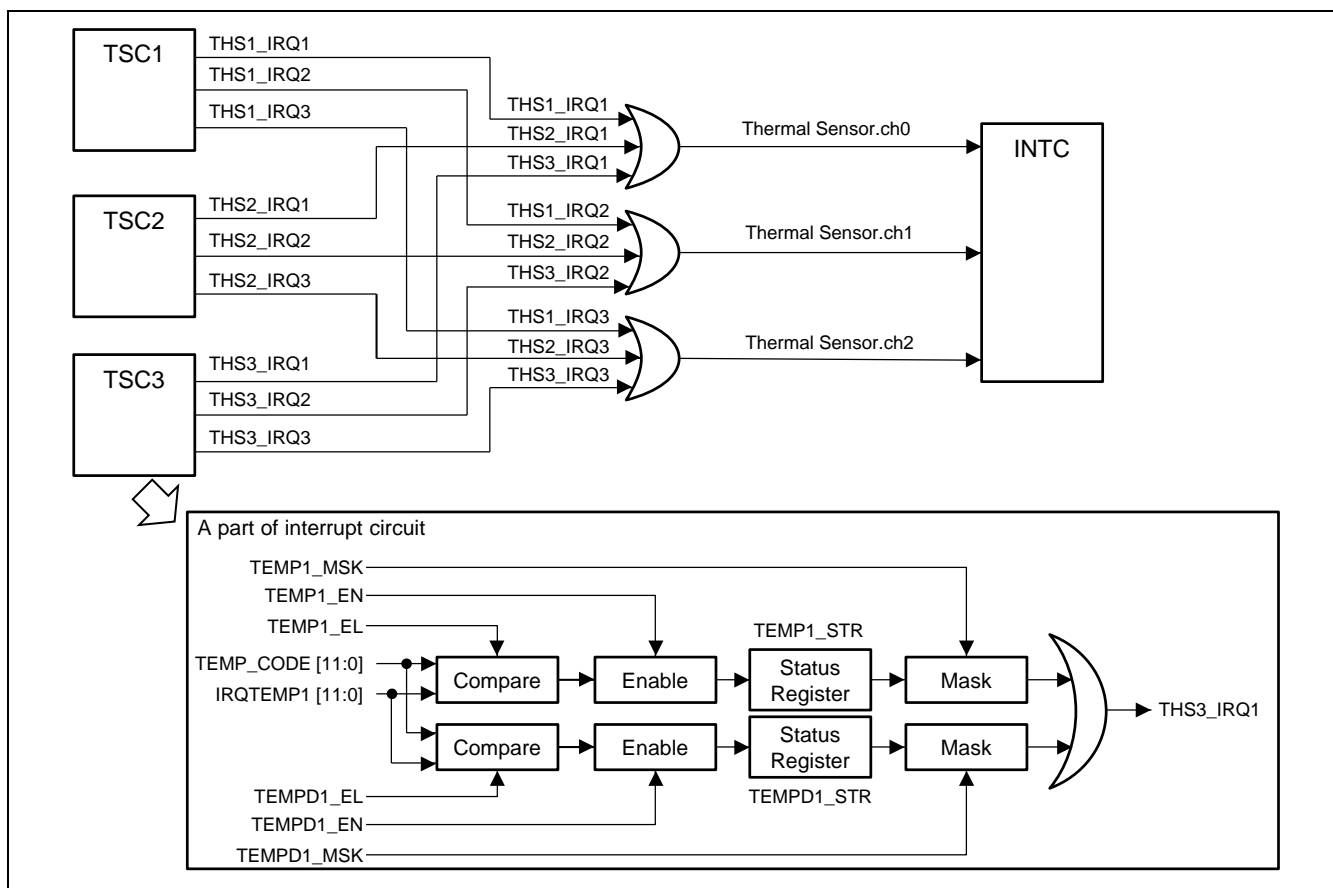


Figure 15.4 Block Diagram of Interrupt (for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N)

15.3.3 Standby Mode

This module supports module standby mode. Standby mode of TSC1, TSC2 and TSC3 are controlled together. Figure 15.5 shows module standby sequence. In the restart method, clear Module Standby register, execute a software reset and initialize in Figure 15.3. Refer to 12. Module Standby, Software Reset for the detail of setting method.

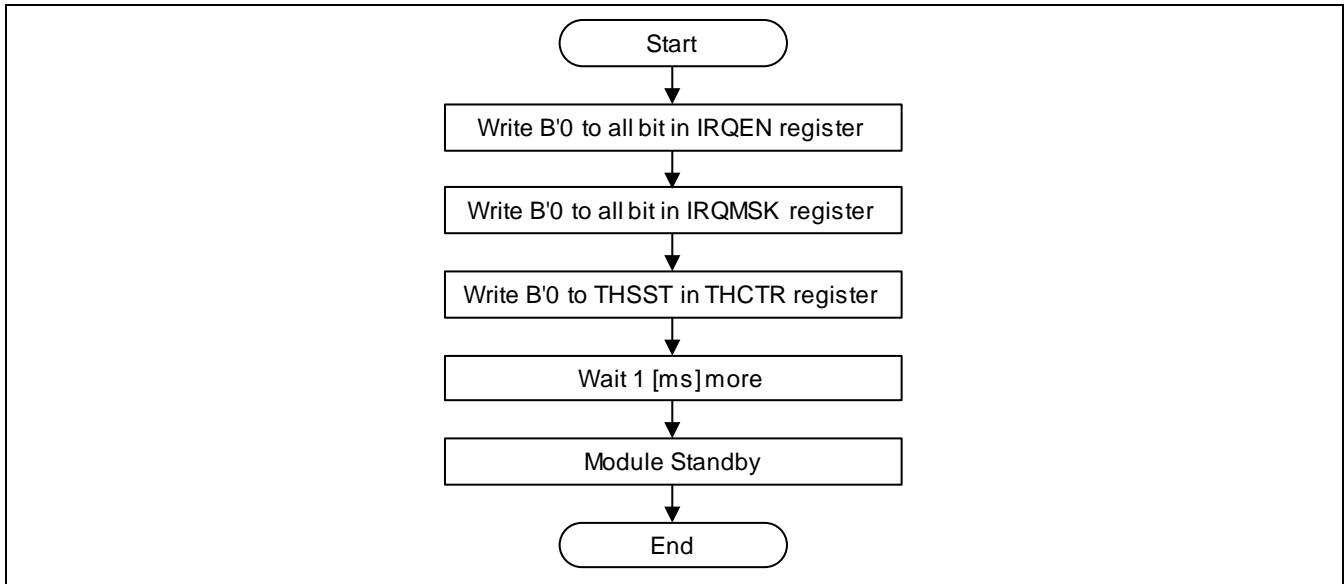


Figure 15.5 Module standby Sequence for Thermal Sensor and Chip Internal Voltage Monitor

15.3.4 Software Reset

This module supports software reset. Software reset of TSC1, TSC2 and TSC3 are controlled together. Refer to 12. Module Standby, Software Reset for the detail of setting method. Figure 15.6 shows software reset sequence. Several registers are not cleared by software reset. Table 15.4 shows behavior of register by software reset.

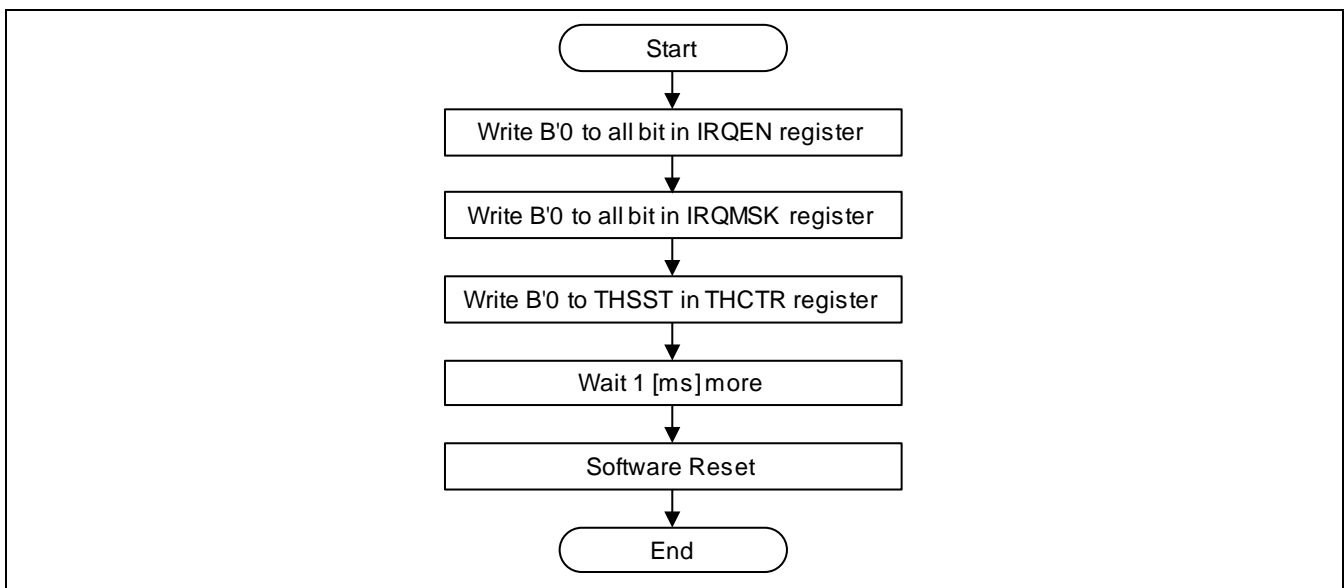


Figure 15.6 Software reset Sequence for Thermal Sensor and Chip Internal Voltage Monitor

Table 15.4 Behavior of Register by Software Reset

| Register Name | Abbreviation | Register values | | | Second Generation RZ/G Series Products | | | |
|--|--------------|-----------------|-------|-------|--|----------------------------|--------|--------|
| | | TSC1 | TSC2 | TSC3 | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Interrupt Status Register | IRQSTR | Clear | Clear | Clear | √ | √ | √ | — |
| Interrupt Mask Register | IRQMSK | Clear | Clear | Clear | √ | √ | √ | — |
| Threshold Edge/Level Register | IRQCTL | Clear | Clear | Clear | √ | √ | √ | — |
| Interrupt Enable Register | IRQEN | Clear | Clear | Clear | √ | √ | √ | — |
| Interrupt Temperature1 Register | IRQTEMP1 | Clear | Clear | Clear | √ | √ | √ | — |
| Interrupt Temperature2 Register | IRQTEMP2 | Clear | Clear | Clear | √ | √ | √ | — |
| Interrupt Temperature3 Register | IRQTEMP3 | Clear | Clear | Clear | √ | √ | √ | — |
| Control Register | THCTR | Clear | Clear | Clear | √ | √ | √ | — |
| Status Register | THSTR | Hold | Hold | Hold | √ | √ | √ | — |
| Temperature Register | TEMP | Clear | Clear | Clear | √ | √ | √ | — |
| Voltage Register | VOLT | Clear | Clear | Clear | √ | √ | √ | — |
| THCODE Parameter1 Register | THCODE1 | Hold | Hold | Hold | √ | √ | √ | — |
| THCODE Parameter2 Register | THCODE2 | Hold | Hold | Hold | √ | √ | √ | — |
| THCODE Parameter3 Register | THCODE3 | Hold | Hold | Hold | √ | √ | √ | — |
| PTAT Parameter1 Register | PTAT1 | Hold | — | — | √ | √ | √ | — |
| PTAT Parameter2 Register | PTAT2 | Hold | — | — | √ | √ | √ | — |
| PTAT Parameter3 Register | PTAT3 | Hold | — | — | √ | √ | √ | — |
| Software Correction Parameter Register | THSCP | Hold | — | — | √ | √ | √ | — |

15.3.5 Power Voltage

Table 15.5 shows the power voltage using the thermal sensor module.

Table 15.5 Power Supply Table

| | Power Supply Name | Function |
|-------------|-------------------|-------------------------------------|
| 3.3V series | — | — |
| 1.8V series | VDDQ18 | Analog circuit power supply voltage |
| 1.5V series | — | — |
| 0.8V series | VDD | Logic circuit power supply voltage |

15.3.6 Input/output Clock

Table 15.6 shows the clock using the thermal sensor module.

Table 15.6 Input/output Clock Table

| | Clock from CPG | Clock from external pins |
|--------|----------------|--------------------------|
| INPUT | CP ϕ | — |
| OUTPUT | — | — |

15.4 Usage Note

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

- (1) When THEN of THCTR register set from 1 to 0, control register after 100 us.
- (2) Leave external LSI pins (VTHSENSE0 and VTHREF0) open.
- (3) Use digital code (TEMP register and VOLT register) when measuring Tj and chip internal voltage.
- (4) When Tj exceeds 115°C, correct operation and device characteristics of this module are not guaranteed.

16. Thermal Sensor (THS)



16.1 Overview

This LSI provides a thermal sensor module that measures the temperature (T_j) inside the LSI. The thermal sensor module also includes a chip internal voltage monitoring module that measures the supply voltage (V_{DD}) inside the LSI.

16.1.1 Features

- (1) The analog voltage of thermal sensor for RZ/G2E measures temperature T_j with an accuracy of $\pm 5^\circ\text{C}$ over the range from -40°C to 115°C . The digital value of thermal sensor for RZ/G2E measures temperature T_j with an accuracy of $\pm 7.75^\circ\text{C}$ over the range from -40°C to 90°C and of $\pm 7.5^\circ\text{C}$ over the range from 90°C to 115°C .
- (2) Provides reference to temperature T_j as measured from outside the chip with the use of external LSI pins (V_{THREF} and $V_{THSENSE}$) for RZ/G2E.
- (3) This module can generate interrupts when the detected temperature T_j within the LSI rises above or falls below several specified temperatures.

16.1.2 Block Diagram

A block diagram of the thermal sensor module for each product is shown in Figure 16.1 (for RZ/G2E). The thermal sensor module consists of the thermal sensors (THS), which are analog circuits for measuring temperature and voltage, respectively, and the thermal sensor controller (TSC), which control the analog circuits.

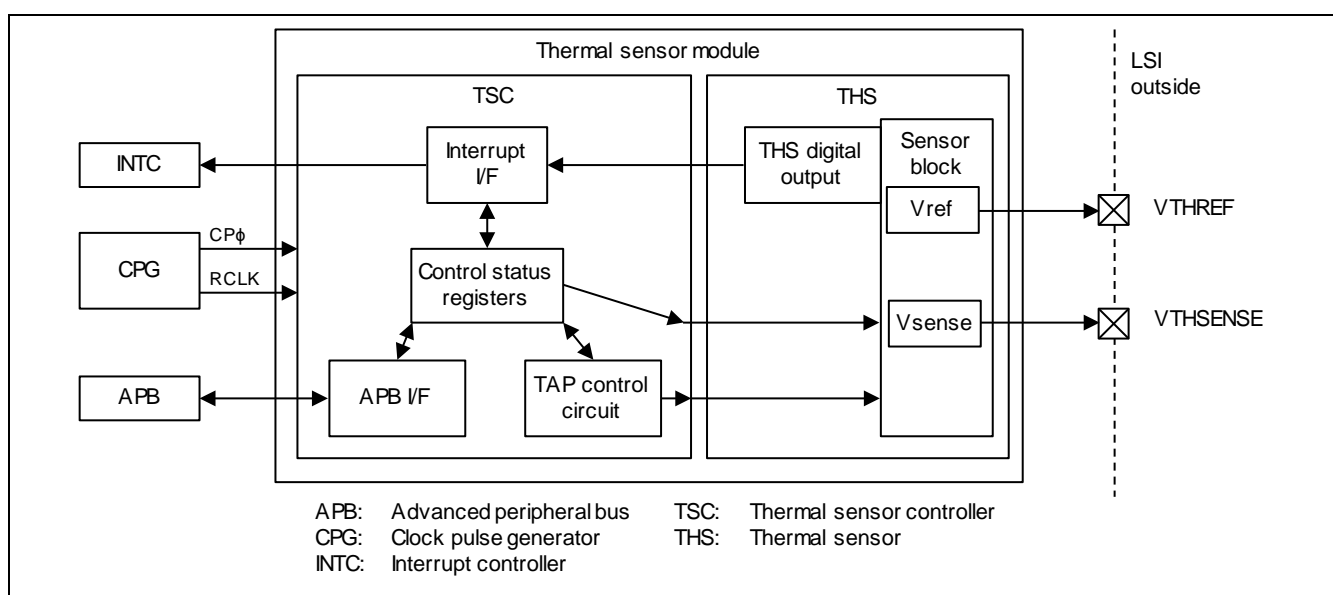


Figure 16.1 Block Diagram of Thermal Sensor Module (RZ/G2E)

The TSC module outputs two kinds of signal to the THS module, that is, signals from control registers and from the TAP control circuit. The signals from control registers are used to control the analog circuits.

The TAP control circuit automatically adjusts the measurement range required in measuring the temperature and voltage, and outputs the results to the THS module.

The THS module outputs two kinds of signal to the TSC module, that is, digital signals from the THS module, which indicate the digital values representing the temperature and voltage, respectively. These values read from the corresponding registers can be used to generate interrupts due to comparison with the specified temperatures or voltages as well as to detect the temperature and voltage within the LSI.

16.1.3 External Pins

Table 16.1 is a list of external pins.

Table 16.1 Pin Configuration

| Name | Pin Name | I/O | Function | Second Generation RZ/G Series Products | | | | |
|---|----------|------------|---|---|----------------------------|--------|--------|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Thermal sensor output voltage | VTHSENSE | Analog I/O | Outputs a voltage that depends on the temperature Tj in the LSI. | — | — | — | √ | |
| Thermal sensor output reference voltage | VTHREF | Analog I/O | Outputs a constant voltage that does not depend on the temperature Tj in the LSI. | — | — | — | √ | |

16.1.4 Register Configuration

Table 16.2 is a list of registers. The base address is H'E619_0000. Any address other than those listed in the table must not be written to. Otherwise, correct operation is not guaranteed.

Table 16.2 List of Registers of the THS

| Register Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | | |
|---|--------------|--------------|-------------|-----------------|-------------|---|----------------------------|--------|--------|--|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Interrupt Status Register | STR | R/(WC0) * | H'E619_0000 | H'0000_0000 | 32 | — | — | — | √ | |
| Interrupt Enable Register | ENR | R/(W)* | H'E619_0004 | H'0000_0000 | 32 | — | — | — | √ | |
| Interrupt Mask Register | INT_MASK | R/W | H'E619_000C | H'000F_0F0F | 32 | — | — | — | √ | |
| Positive/Negative Logic Select Register | POSNEG | R/(W)* | H'E619_0120 | H'0000_0001 | 32 | — | — | — | √ | |
| THS Control Register | THSCR | R/(W)* | H'E619_012C | H'0000_0003 | 32 | — | — | — | √ | |
| THS Status Register | THSSR | R | H'E619_0130 | Undefined value | 32 | — | — | — | √ | |
| Interrupt Control Register | INTCTRL | R/(W)* | H'E619_0134 | H'2500_0000 | 32 | — | — | — | √ | |

Note: * Several bits are read-only bits.

16.1.5 Connected Module

Table 16.3 is a list of the modules that are connected to this module.

Table 16.3 Connected Modules

| Module Name | Connected Module Name | Connected Module Function |
|--------------------|------------------------------|----------------------------------|
| THS | APB | Control to register access |
| | CPG | Output clock |
| | INTC | Control to interrupts |
| | Module standby | Control to stop clock |
| | Software reset | Execute software reset |

16.2 Register Description

[Legend for register description]

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R: Read-only. The write value should always be 0.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

W: Write-only. Reading this bit is prohibited. When the bit is reserved, the write value should always be 0.

_/W: Write-only. The read value is undefined.

16.2.1 Interrupt Status Register (STR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|--------|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | Tj02ST | Tj01ST | Tj00ST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/WC0 | R/WC0 | R/WC0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-------|--|
| 31 to 3 | — | All 0 | R | These bits are fixed to 0. |
| 2 | Tj02ST | B'0 | R/WC0 | Tj02 Interrupt Detection Status 0: Not detected. 1: The Tj02 threshold temperature specified by CTEMP2[5:0] bits in the INTCTLR register is detected. This bit is cleared by writing 0. |
| 1 | Tj01ST | B'0 | R/WC0 | Tj01 Interrupt Detection Status 0: Not detected. 1: The Tj01 threshold temperature specified by CTEMP1[5:0] bits in the INTCTLR register is detected. This bit is cleared by writing 0. |
| 0 | Tj00ST | B'0 | R/WC0 | Tj00 Interrupt Detection Status 0: Not detected. 1: The Tj00 threshold temperature specified by CTEMP0[5:0] bits in the INTCTLR register is detected. This bit is cleared by writing 0. |

16.2.2 Interrupt Enable Register (ENR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|-----|---------|---------|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | Tj02_EN | Tj01_EN | Tj00_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 18 | — | All 0 | R | These bits are fixed to 0. |
| 17, 16 | — | All 0 | R/W | The write value should always be 0. |
| 15 to 4 | — | All 0 | R | These bits are fixed to 0. |
| 3 | — | B'0 | R/W | The write value should always be 0. |
| 2 | Tj02_EN | B'0 | R/W | Enables comparing the CTEMP2[5:0] bits in the INTCTLR register with the CTEMP[5:0] bits in the THSSR register. 0: Disable 1: Enable |
| 1 | Tj01_EN | B'0 | R/W | Enables comparing the CTEMP1[5:0] bits in the INTCTLR register with the CTEMP[5:0] bits in the THSSR register. 0: Disable 1: Enable |
| 0 | Tj00_EN | B'0 | R/W | Enables comparing the CTEMP0[5:0] bits in the INTCTLR register with the CTEMP[5:0] bits in the THSSR register. 0: Disable 1: Enable |

16.2.3 Interrupt Mask Register (INT_MASK)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-------------|-------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | Tj02INT_MSK | Tj01INT_MSK | Tj00INT_MSK |
| Initial value: | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 20 | — | All 0 | R/W | The write value should always be 0. |
| 19 to 16 | — | All 1 | R/W | The write value should always be 1. |
| 15 to 12 | — | All 0 | R/W | The write value should always be 0. |
| 11 to 8 | — | All 1 | R/W | The write value should always be 1. |
| 7 to 4 | — | All 0 | R/W | The write value should always be 0. |
| 3 | — | B'1 | R/W | The write value should always be 1. |
| 2 | Tj02INT_MSK | B'1 | R/W | Selects whether or not the Tj02 interrupt request is masked when Tj02ST bit in the STR register is 1. 0: Mask is cleared. 1: Masked. |
| 1 | Tj01INT_MSK | B'1 | R/W | Selects whether or not the Tj01 interrupt request is masked when Tj01ST bit in the STR register is 1. 0: Mask is cleared. 1: Masked. |
| 0 | Tj00INT_MSK | B'1 | R/W | Selects whether or not the Tj00 interrupt request is masked when Tj00ST bit in the STR register is 1. 0: Mask is cleared. 1: Masked. |

16.2.4 Positive/Negative Logic Select Register (POSNEG)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|-----|-----------|-----------|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | POSNEG[2] | POSNEG[1] | POSNEG[0] |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 4 | — | All 0 | R | These bits are fixed to 0. |
| 3 | — | B'0 | R/W | The write value should always be 0. |
| 2 | POSNEG[2] | B'0 | R/W | Selects the edge polarity of the Tj02 interrupt input signal. 0: Detects a rising edge. (An interrupt is generated when Tj rises above a specified temperature, CTEMP2[5:0] bits in the INTCTLR register.) 1: Detects a falling edge. (An interrupt is generated when Tj falls below a specified temperature, CTEMP2[5:0] bits in the INTCTLR register.) |
| 1 | POSNEG[1] | B'0 | R/W | Selects the edge polarity of the Tj01 interrupt input signal. 0: Detects a rising edge. (An interrupt is generated when Tj rises above a specified temperature, CTEMP1[5:0] bits in the INTCTLR register.) 1: Detects a falling edge. (An interrupt is generated when Tj falls below a specified temperature, CTEMP1[5:0] bits in the INTCTLR register.) |
| 0 | POSNEG[0] | B'1 | R/W | Selects the edge polarity of the Tj00 interrupt input signal. 0: Detects a rising edge. (An interrupt is generated when Tj rises above a specified temperature, CTEMP0[5:0] bits in the INTCTLR register.) 1: Detects a falling edge. (An interrupt is generated when Tj falls below a specified temperature, CTEMP0[5:0] bits in the INTCTLR register.) |

16.2.5 THS Control Register (THSCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|----|-------|----|----|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | CPCTL | — | — | THIDLE[1:0] | | — | | | — | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| R/W: | R | R | R | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31, 30 | — | All 0 | R/W | The write value should always be 0. |
| 29 to 17 | — | All 0 | R | These bits are fixed to 0. |
| 16 | — | B'0 | R/W | The write value should always be 0. |
| 15 to 13 | — | All 0 | R | These bits are fixed to 0. |
| 12 | CPCTL | B'0 | R/W | TAP value select method 0: Setting prohibited 1: TAP value is determined automatically by hardware. When 1 is set, an offset value appropriate for measuring the current chip temperature is selected by hardware. Note: Set this bit to 1 when using the thermal sensor. |
| 11, 10 | — | All 0 | R | These bits are fixed to 0. |
| 9, 8 | THIDLE[1:0] | B'00 | R/W | Selects either the normal operating state or the idle state of the THS. B'00: Normal operating state B'01: Setting prohibited B'10: Normal operating state (output from VTHSENSE or VTHREF is stopped) B'11: Idle state Set this bit to B'11 (idle state) when the thermal sensor is not used. |
| 7 to 4 | — | H'0 | R/W | The write value should be 0. |
| 3 to 0 | — | H'3 | R/W | The write value should be the initial value. |

16.2.6 THS Status Register (THSSR)

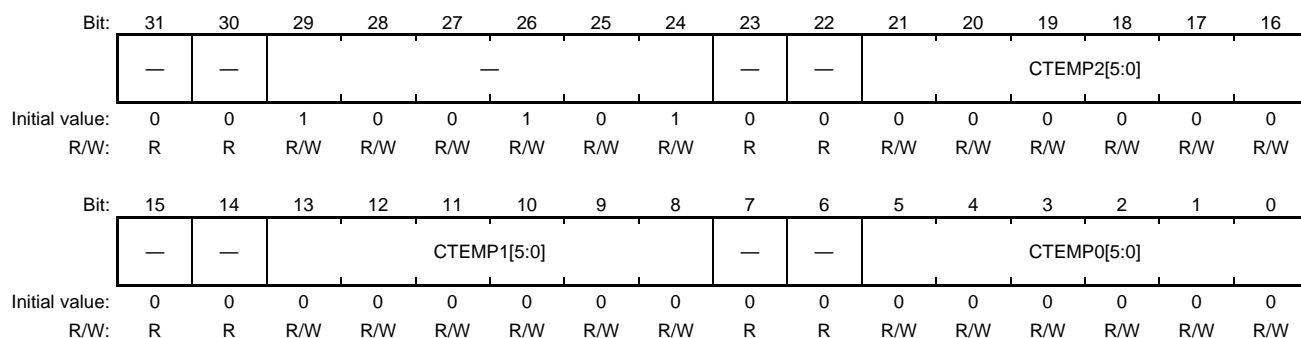
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|------------|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — | 0 | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | CTEMP[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 28 | — | All 0 | R | These bits are fixed to 0. |
| 27 to 24 | — | — | R | These bits are indefinite. |
| 23 to 20 | — | — | R | These bits are indefinite. |
| 19 | — | B'0 | R | This bit is fixed to 0. |
| 18 to 16 | — | — | R | These bits are indefinite. |
| 15 to 6 | — | All 0 | R | These bits are fixed to 0. |
| 5 to 0 | CTEMP[5:0] | — | R | Indicates the current temperature. Convert the value of the bits to actual temperature (°C) by the following formula. When CTEMP[5:0] is less than 24, $T = CTEMP[5:0] \times 5.5 - 72.$ When CTEMP[5:0] is equal to or greater than 24, $T = CTEMP[5:0] \times 5 - 60.$ |

16.2.7 Interrupt Control Register (INTCTLR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31, 30 | — | All 0 | R | These bits are fixed to 0. |
| 29 to 24 | — | H'25 | R/W | The write value should always be H'25. |
| 23, 22 | — | All 0 | R | These bits are fixed to 0. |
| 21 to 16 | CTEMP2[5:0] | All 0 | R/W | Indicates the temperature that causes an INTDT2 interrupt. When interrupt temperature is lower than 60°C, Specify the value calculated by “((interrupt temperature) + 72) / 5.5.” When interrupt temperature is equal to or higher than 60°C, Specify the value calculated by “((interrupt temperature) + 60) / 5.” The INTDT2 interrupt is generated when the CPCTL bit in THSCR is 1 and the following condition is satisfied: CTEMP2[5:0] ≥ CTEMP[5:0] (when POSNEG[2] is 1) or CTEMP2[5:0] < CTEMP[5:0] (when POSNEG[2] is 0). Here CTEMP[5:0] is the current chip temperature indicated with bits 5 to 0 in THSSR. |
| 15, 14 | — | All 0 | R | These bits are fixed to 0. |
| 13 to 8 | CTEMP1[5:0] | All 0 | R/W | Indicates the temperature that causes an INTDT1 interrupt. When interrupt temperature is lower than 60°C, Specify the value calculated by “((interrupt temperature) + 72) / 5.5.” When interrupt temperature is equal to or higher than 60°C, Specify the value calculated by “((interrupt temperature) + 60) / 5.” The INTDT1 interrupt is generated when the CPCTL bit in THSCR is 1 and the following condition is satisfied: CTEMP1[5:0] ≥ CTEMP[5:0] (when POSNEG[1] is 1) or CTEMP1[5:0] < CTEMP[5:0] (when POSNEG[1] is 0). Here CTEMP[5:0] is the current chip temperature indicated with bits 5 to 0 in THSSR. |
| 7, 6 | — | All 0 | R | These bits are fixed to 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-------------|---------------|-----|---|
| 5 to 0 | CTEMP0[5:0] | All 0 | R/W | <p>Indicates the temperature that causes an INTDT0 interrupt.</p> <p>When interrupt temperature is lower than 60°C, Specify the value calculated by “((interrupt temperature) + 72) / 5.5.”</p> <p>When interrupt temperature is equal to or higher than 60°C, Specify the value calculated by “((interrupt temperature) + 60) / 5.”</p> <p>The INTDT0 interrupt is generated when the CPCTL bit in THSCR is 1 and the following condition is satisfied:</p> <p>CTEMP0[5:0] ≥ CTEMP[5:0] (when POSNEG[0] is 1) or CTEMP0[5:0] < CTEMP[5:0] (when POSNEG[0] is 0).</p> <p>Here CTEMP[5:0] is the current chip temperature indicated with bits 5 to 0 in THSSR.</p> |

16.3 Operation



16.3.1 Initial Sequence of Thermal Sensor

Figure 16.2 shows initial sequence of thermal sensor.

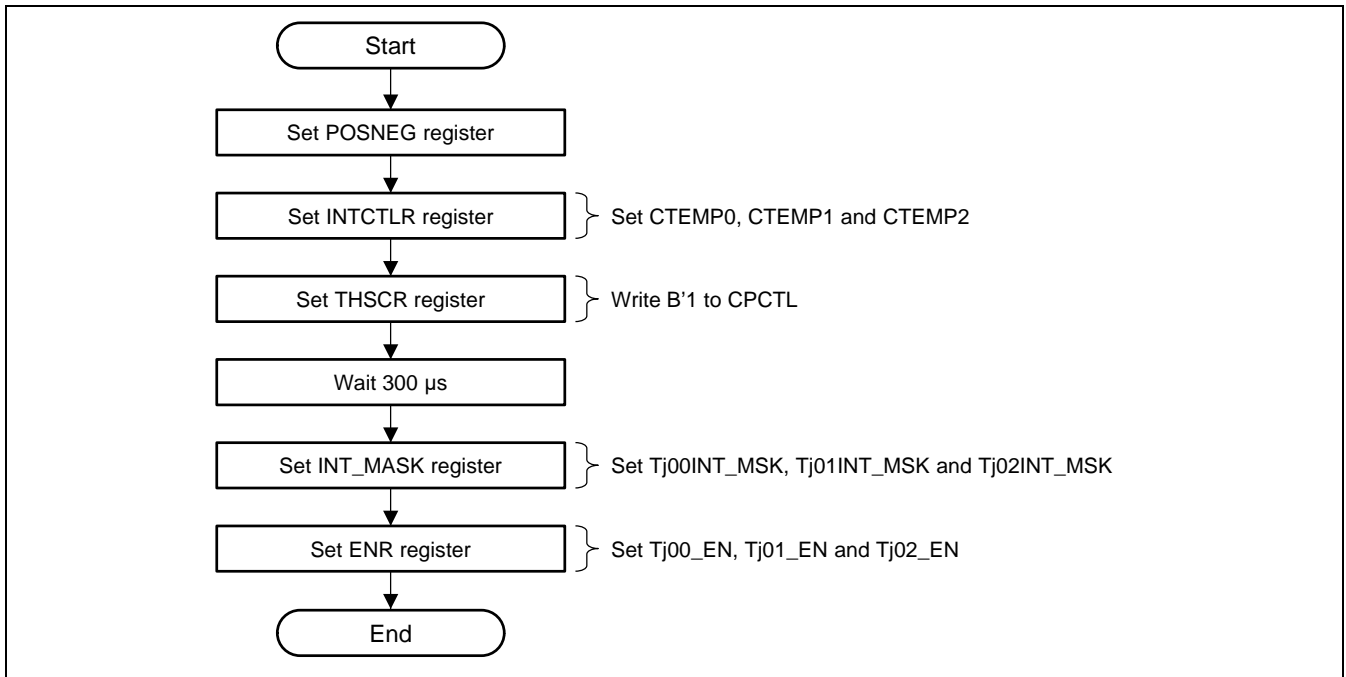


Figure 16.2 Initial Sequence of Thermal Sensor

16.3.2 Temperature Measurement Using Register (When CPCTL = 0)

Setting is prohibited.

16.3.3 Temperature Measurement Using Register (When CPCTL = 1)

When reading the temperature measurement result from a register, read the value of the CTEMP[5:0] bits in THSSR.

Do not use the value of the CTEMP[5:0] bits in THSSR immediately after a change to the offset value for the TAP control circuit because the new offset setting will not yet have been reflected in the value. When CPCTL is 1, hardware automatically updates the offset value but since software is not informed of the timing of updates, consecutively read the value of the CTEMP[5:0] bits in THSSR at an interval over 300 μ s and only accept the result when the same value is read twice (since the temperature changes more gradually than the 300 μ s interval, as long as a change to the offset value is not made during the interval between reading of the CTEMP[5:0] bits in THSSR, the same value will be read out.)

By setting 0 or 1 in the positive/negative logic select register (POSNEG) and specifying the temperatures that cause interrupts in the INTCTLR register, the INTDT2 to INTDT0 interrupts can be generated when the chip temperature rises above or falls below the temperatures specified in the INTCTLR register.

However, immediately after a change to the offset value for the TAP control circuit, an interrupt may be generated before completion of the temperature measurement at the changed offset value. If an interrupt is generated at such a time, read out the value of the CTEMP[5:0] bits in THSSR a second time after 300 μ s, check whether or not the interrupt was valid, and if it was, proceed with interrupt processing.

16.3.4 Interrupt

Figure 16.3 shows the block diagram of interrupt. The interrupts for thermal sensor module have three (Thermal Sensor.ch0, Thermal Sensor.ch1, and Thermal Sensor.ch2).

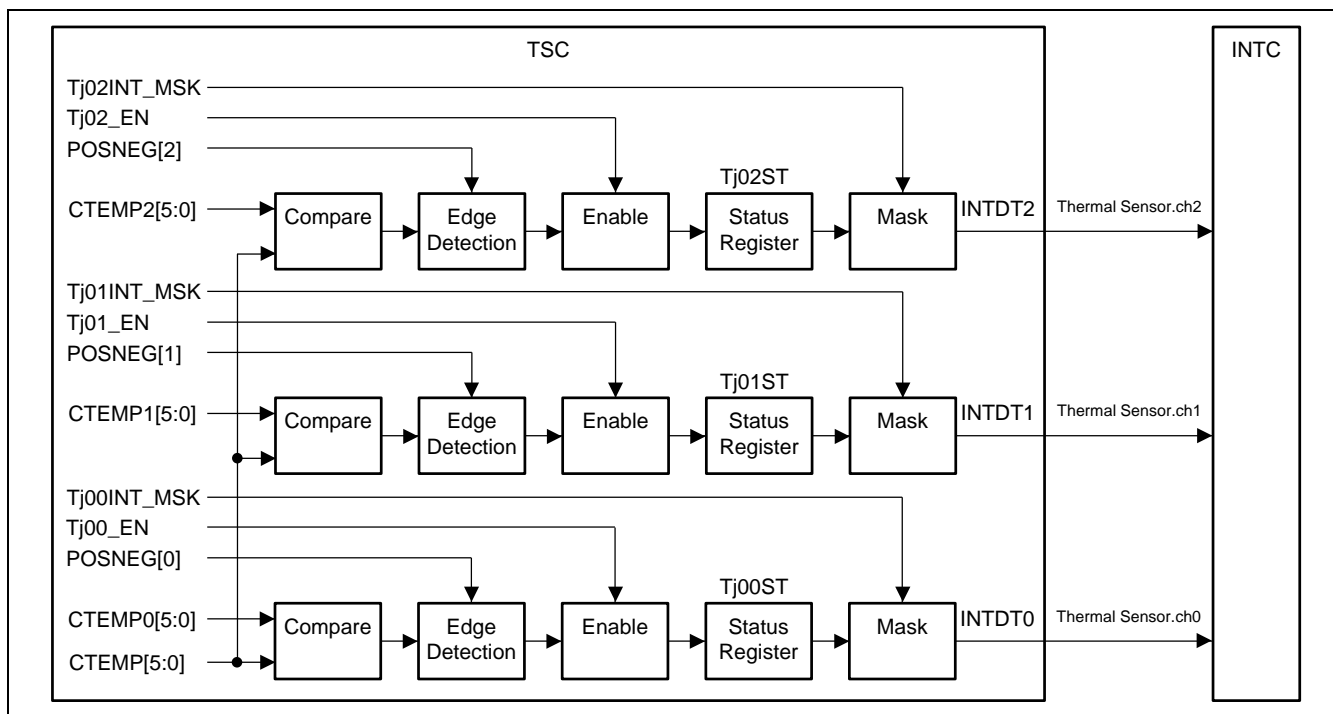


Figure 16.3 Block Diagram of Interrupt

16.3.5 Standby Mode

This module supports module standby mode. Figure 16.4 shows module standby sequence. In the restart method, clear Module Standby register, execute a software reset and initialize in Figure 16.2. Refer to 12. Module Standby, Software Reset for the detail of setting method.

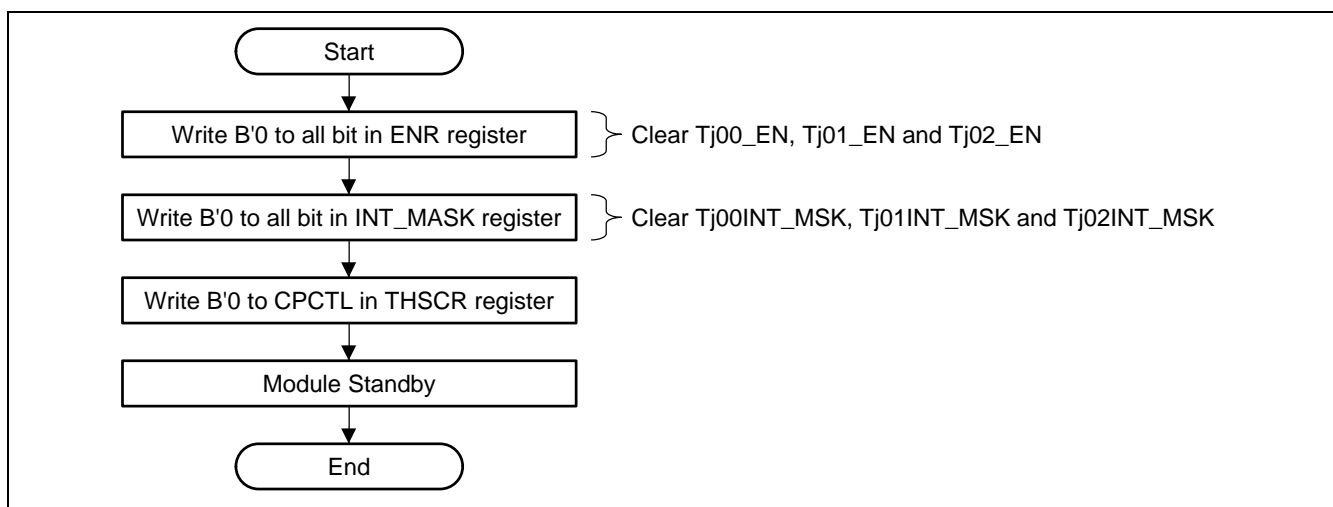


Figure 16.4 Module Standby Sequence of Thermal Sensor

16.3.6 Software Reset

This module supports software reset. Refer to 12. Module Standby, Software Reset for the detail of setting method. Figure 16.5 shows software reset sequence. Table 16.4 shows behavior of register by software reset.

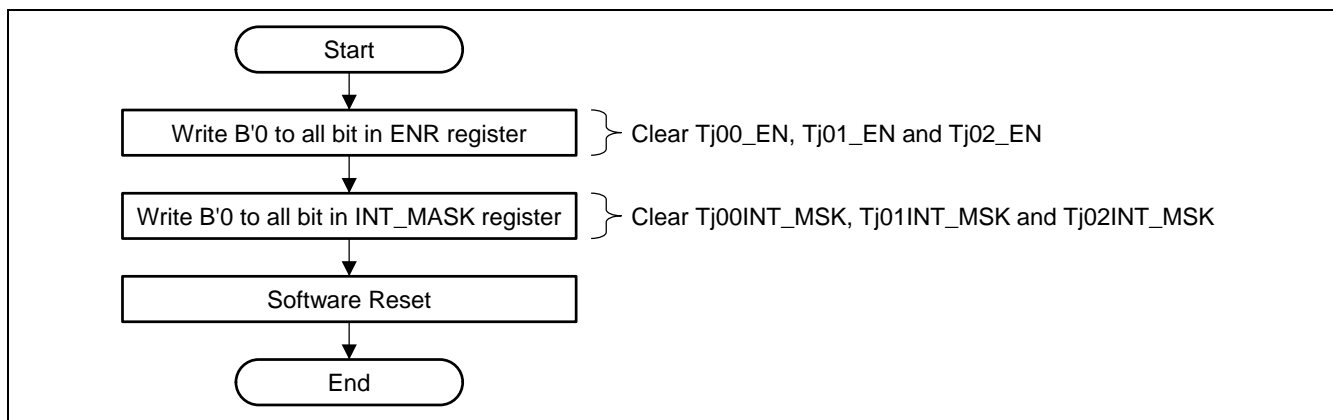


Figure 16.5 Software reset Sequence of Thermal Sensor

Table 16.4 Behavior of Register by Software Reset

| Register Name | Abbreviation | Register values | Second Generation RZ/G Series Products | | | |
|---|--------------|-----------------|---|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Interrupt Status Register | STR | Clear | — | — | — | √ |
| Interrupt Enable Register | ENR | Clear | — | — | — | √ |
| Interrupt Mask Register | INT_MASK | Clear | — | — | — | √ |
| Positive/Negative Logic Select Register | POSNEG | Clear | — | — | — | √ |
| THS Control Register | THSCR | Clear | — | — | — | √ |
| THS Status Register | THSSR | Clear | — | — | — | √ |
| Interrupt Control Register | INTCTRL | Clear | — | — | — | √ |

16.3.7 Power Voltage

Table 16.5 shows the power voltage using the thermal sensor module.

Table 16.5 Power Supply Table

| | Power Supply Name | Function |
|-------------|-------------------|-------------------------------------|
| 3.3V series | — | — |
| 1.8V series | VDDQ18 | Analog circuit power supply voltage |
| 1.5V series | — | — |
| 1.0V series | VDD | Logic circuit power supply voltage |

16.3.8 Input/output Clock

Table 16.6 shows the clock using the thermal sensor module.

Table 16.6 Input/output Clock Table

| | Clock from CPG | Clock from external pins |
|--------|----------------|--------------------------|
| INPUT | CP ϕ | — |
| INPUT | RCLK | — |
| OUTPUT | — | — |

16.3.9 The Temperature Measurement Using External Pins (RZ/G2E)

For temperature measurement with the external pins, connect the external circuit to the VTHSENSE and VTHREF pins and calculate the temperature based on the output voltages on these pins.

(1) External Pins Used for Temperature Measurement

In temperature measurement with the external pins, the pins shown in Table 16.7 are used.

Table 16.7 External Pins Used for Temperature Measurement

| Pin Name | Output Voltage Range [V] | Maximum Allowed Input/Output Current [μ A] | Tj Accuracy |
|----------|--------------------------------|---|--|
| VTHSENSE | 360mV (min.) to 750mV (max.) | ± 3 | Tj accuracy of RZ/G2E is $\pm 5^{\circ}\text{C}$ (-40°C to 115°C). {(VTHREF-VTHSENSE) specification} (This does not include error such as device accuracy of the referenced external connection circuit). The temperature range for detection is -40°C to 115°C . When Tj is outside the range of -40°C to 115°C , the operation and accuracy cannot be guaranteed. |
| VTHREF | 1.18 V (min.) to 1.32 V (max.) | ± 3 | |

(2) Referenced External Connection Circuit

Figure 16.6 shows a referenced external connection circuit for the THS used for temperature measurement with the external pins.

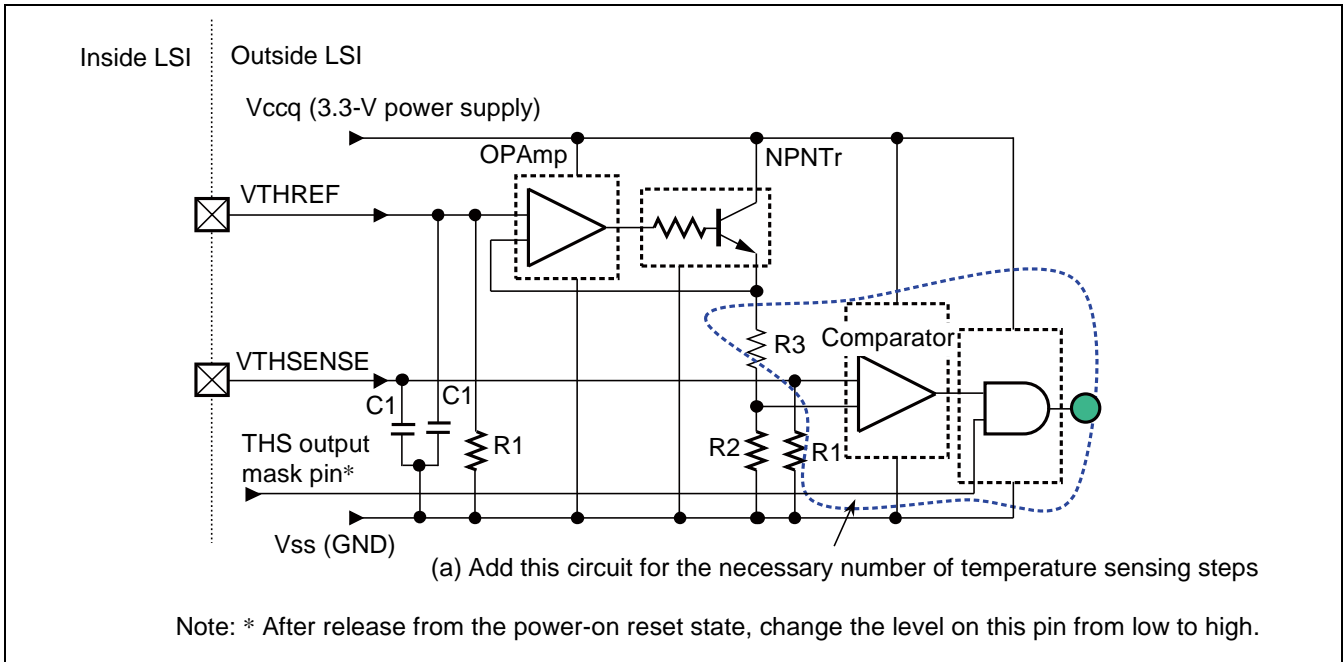


Figure 16.6 Referenced External Connection Circuit for THS

The referenced external connection circuit outputs a high level at a desired temperature. By using VTHREF, which is affected by the noise equal to the external noise that affects VTHSENSE, the external noise is canceled out, and thus the accuracy of Tj is improved.

In addition, the number of steps for digitizing a measured temperature Tj (the number of temperatures with which an actual temperature is compared) can be increased by adding the circuit part indicated with (a) in the figure.

The followings should be considered for the wiring of VTHSENSE and VTHREF.

1. For jumper wiring use a twisted pair.
2. For substrate wiring, arrange differential lines as close to each other as possible on the board.

To obtain low-to-high output at Tj = 115 °C, the constants of the circuit components are as follows:

$$R1 = 560 \text{ k}\Omega, C1 = 0.1 \text{ }\mu\text{F}$$

reference value : R2 = 340Ω, R3 = 280Ω (E96) Tj@115.0°C

For R2 and R3, E96 series resistor with a tolerance of ± 0.1% is recommended. The error of this resistor will be the same value as the error produced when Tj is digitalized. A material which is little affected by the temperature is recommended for the resistor. Note that, R2 and R3 can be configured from two or more resistors.

The accuracy of R1 is not a problem since R1 is used as a pull-down resistor.

C1 is a bypass capacitor. Be sure to mount this even when the referenced external connection circuit is not to be used.

(3) Calculation Formula

Temperature T_j can be calculated from the voltages on VTHSENSE and VTHREF as follows.

$$T_j = (VTHREF - VTHSENSE + \text{temperature coefficient 2}) / \text{temperature coefficient 1} \text{ [}^\circ\text{C]}$$

VTHSENSE, VTHREF: measured voltages [V]

Two temperature coefficients use two types.

When $VTHREF - VTHSENSE$ value is less than 680.13mV,

Temperature coefficient 1: -1.58 [mV/ $^\circ\text{C}$]

Temperature coefficient 2: -763.08 [mV].

When $VTHREF - VTHSENSE$ value is equal to or greater than 680.13mV,

Temperature coefficient 1: -1.56 [mV/ $^\circ\text{C}$]

Temperature coefficient 2: -762.03 [mV].

16.4 Usage Note



- (1) When the THS is not used, THIDLE[1:0] bits in THSCR register should be set to B'11 (leave external LSI pins (VTHSENSE and VTHREF) open in RZ/G2E).
- (2) The guaranteed accuracy of the outputs on the VTHSENSE and VTHREF pins is over the temperature range from -40°C to 115°C. Take measurement errors into account in designing the user system. (RZ/G2E)

Note that, when the internal registers of the TSC are used for measurement of the chip temperature, the operation of the sensor is not guaranteed outside the range of -40°C to 115°C within which operation of the logic circuit is guaranteed.

- (3) When T_j exceeds 115°C, correct operation and device characteristics are not guaranteed.

Although this module provides a bit for detecting $T_j = 135^\circ\text{C}$, this bit should only be used for fail-safe operation where the LSI is stopped on the verge of a thermal runaway condition (but has not yet entered that state) and the power consumption is increasing rapidly.

- (4) External LSI pins (VTHSENSE and VTHREF) are intended to fail-safe. For instance, this circuit can be used as an external fail-safe circuit to stop the power supply to the LSI when thermal runaway occurs and the LSI becomes uncontrollable because of high temperature. (RZ/G2E)

17. Reset (RST)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

17.1 Overview

The reset (RST) consists of the bus interface block and registers related to reset control. Timing of the bus interface block is based on $CP\phi$.

17.1.1 Features

The following functions are implemented by RST.

- Register-based reset control for the Cortex-A57, Cortex-A53 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]
Register-based reset control for the Cortex-A53 [RZ/G2E]
Register-based reset control for the Cortex-A57 [RZ/G2N]
- Latching of the levels on mode pins when PRESET# is negated
- Mode monitoring register
- Boot address registers for the Cortex-A57, Cortex-A53, [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]
Boot address registers for the Cortex-A53, [RZ/G2E]
Boot address registers for the Cortex-A57, [RZ/G2N]
- WDT reset request control function.
- PRESETOUT control function.

17.1.2 Block Diagram

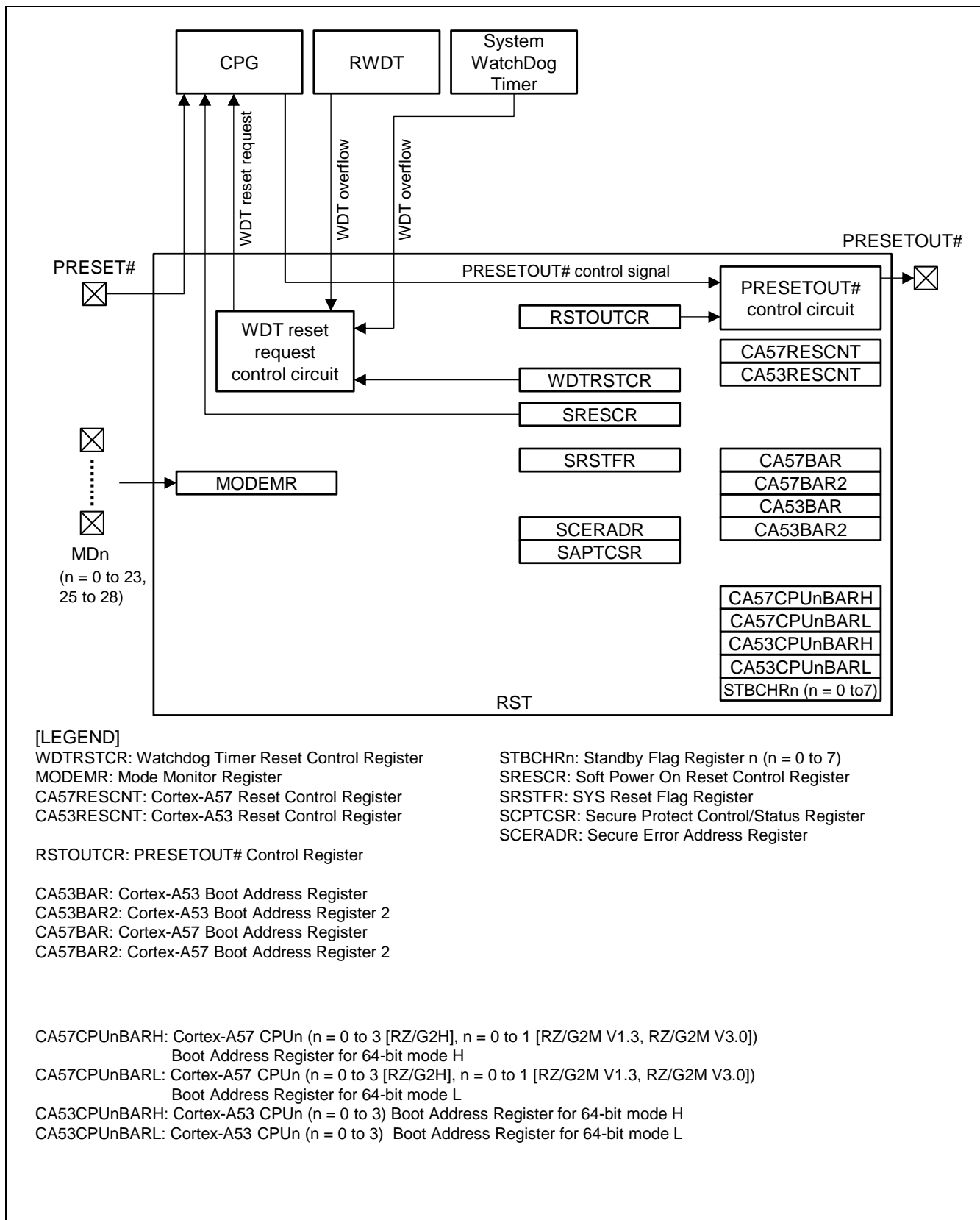


Figure 17.1 Block Diagram of RST [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]

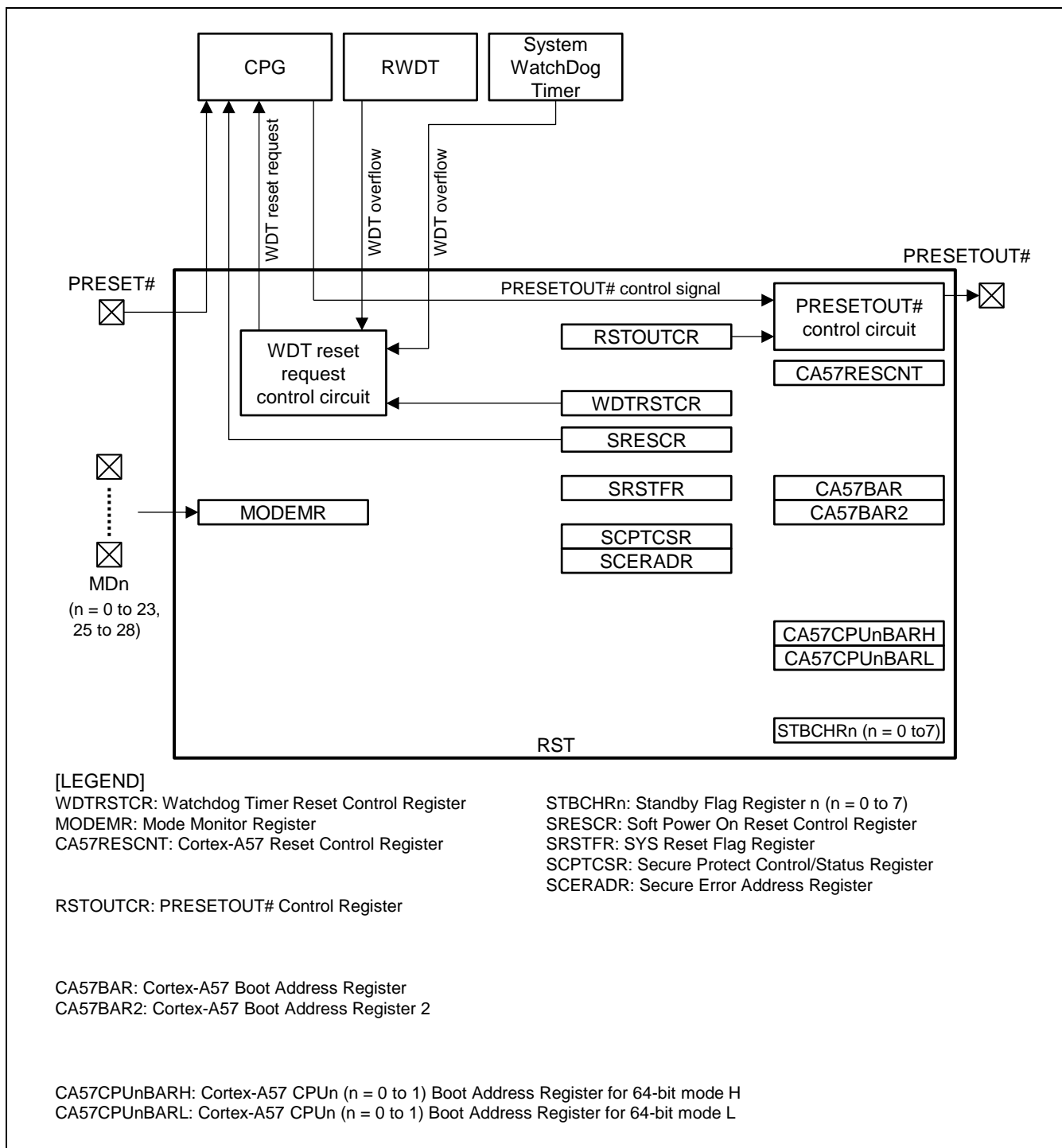


Figure 17.2 Block Diagram of RST [RZ/G2N]

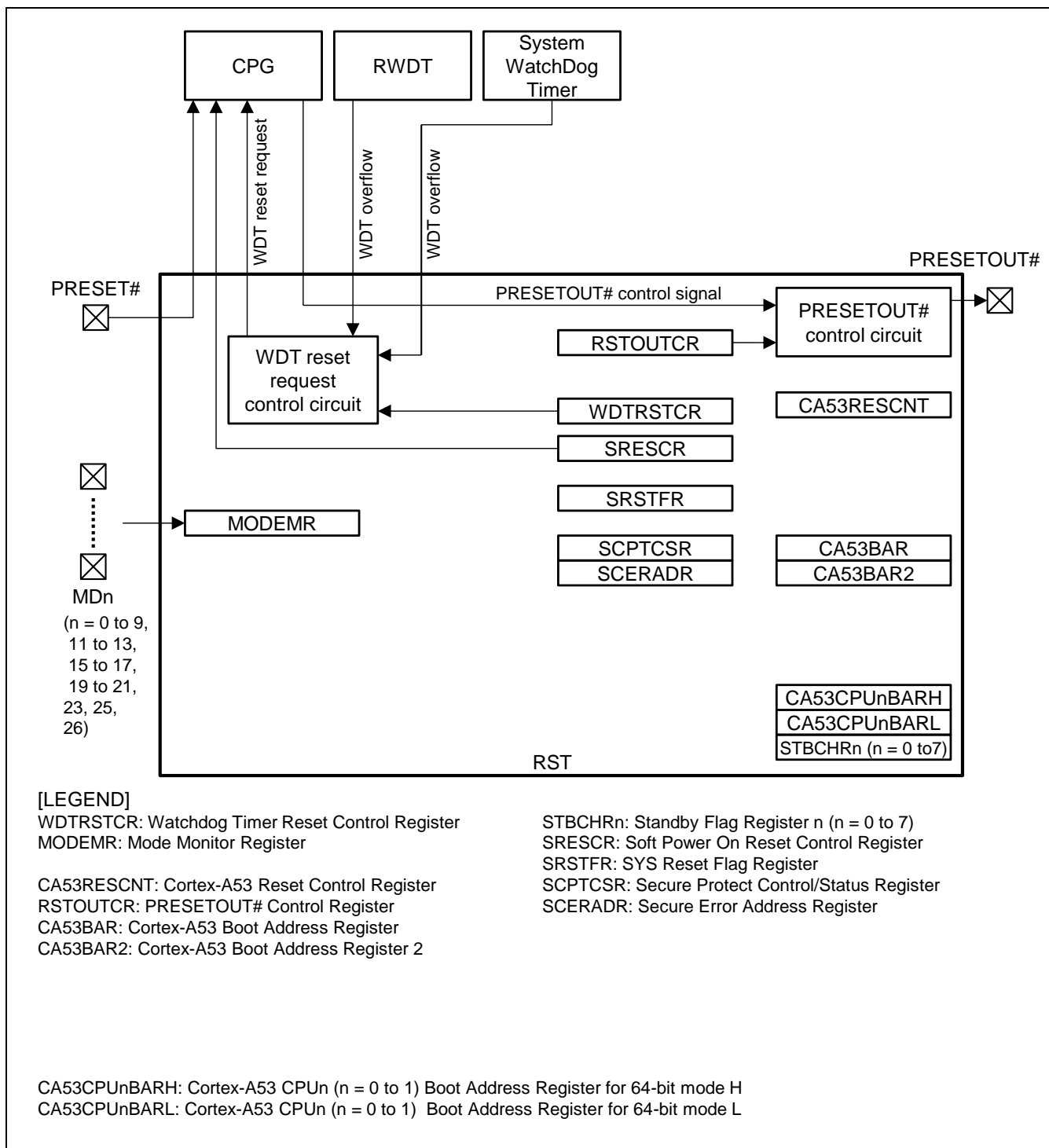


Figure 17.3 Block Diagram of RST [RZ/G2E]

17.1.3 External Pins

The following table lists the pins connected to the RST.

Table 17.1 External Pins for RST

| Pin Name | Function | Direction | Remarks | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------------|-----------------------------|-----------|---|--------|----------------------------|--------|--------|
| MD0 to MD5 | Mode pins Fix MD0 to L. | IN | Mode pins | √ | √ | √ | √ |
| MD6 | Mode pin | IN | Mode pin | √ | √ | √ | √ |
| MD7 to MD9 | Mode pins | IN | Mode pins | √ | √ | √ | √ |
| MD10 | Mode pin | IN | Mode pin | √ | √ | √ | — |
| MD11 | Mode pin | IN | Mode pin | √ | √ | √ | √ |
| MD12 | Mode pin | IN | Mode pin | √ | √ | √ | √ |
| MD13 | Mode pin | IN | Mode pin | √ | √ | √ | √ |
| MD14 | Mode pin | IN | Mode pin | √ | √ | √ | — |
| MD15 to MD16 | Mode pins Fix MD16 to H. | IN | Mode pins | √ | √ | √ | √ |
| MD17 | Mode pin | IN | Mode pin | √ | √ | √ | √ |
| MD18 | Mode pin | IN | Mode pin | √ | √ | √ | — |
| MD19 | Mode pin | IN | Mode pin | √ | √ | √ | √ |
| MD20 | Mode pin | IN | Mode pin | √ | √ | √ | √ |
| MD21 | Mode pin | IN | Mode pin | √ | √ | √ | √ |
| MD22 | Mode pin | IN | Mode pin | √ | √ | √ | — |
| MD23 | Mode pin Fix MD23 to L. | IN | Mode pin | √ | √ | √ | √ |
| MD24 | Mode pin | IN | Mode pin | — | — | — | — |
| MD25 to MD26 | Mode pins Fix MD26 to L. | IN | Mode pins | √ | √ | √ | √ |
| MD27 to MD28 | Mode pins | IN | Mode pins | √ | √ | √ | — |
| MD29 to MD32 | Mode pins | IN | Mode pins | — | — | — | — |
| MDT0, MDT1 | Mode pins | IN | Mode pins | √ | √ | √ | √ |
| PRESET# | Power-on-reset | IN | Power-on-reset The low-level input on this pin places the LSI in the power-on-reset state. | √ | √ | √ | √ |
| PRESETOUT# | Indicates internal reset | OUT | Reset output | √ | √ | √ | √ |

17.1.4 Register Configuration

Table 17.2 RST Register Details

| Register Name | Mnemonic | R/W | Address | Initial value | Size | Access permission | Second Generation RZ/G Series Products | | | | |
|---|------------------|-----|-------------|---------------|------|-------------------|--|----------------------------|--------|--------|--|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Mode Monitor Register | MODEMR | R | H'E616_0060 | H'----_---- | 32 | ALL | √ | √ | √ | √ | |
| Cortex-A57 Reset Control Register | CA57RESCNT | R/W | H'E616_0040 | H'0000_000- | 32 | ALL | √ | √ | √ | — | |
| Cortex-A53 Reset Control Register | CA53RESCNT | R/W | H'E616_0044 | H'0000_000- | 32 | ALL | √ | √ | — | √ | |
| Watchdog Timer Reset Control Register | WDTRSTCR | R/W | H'E616_0054 | H'0000_C003 | 32 | ALL *1 | √ | √ | √ | √ | |
| PRESETOUT# Control Register | RSTOUTCR | R/W | H'E616_0058 | H'0000_0001 | 32 | ALL | √ | √ | √ | √ | |
| Cortex-A53 Boot Address Register | CA53BAR | R/W | H'E616_0030 | H'0000_00-- | 32 | ALL | √ | √ | — | √ | |
| Cortex-A53 Boot Address Register2 | CA53BAR2 | R/W | H'E616_0034 | H'0000_0000 | 32 | ALL *1 | √ | √ | — | √ | |
| Cortex-A57 Boot Address Register | CA57BAR | R/W | H'E616_0020 | H'0000_00-- | 32 | ALL | √ | √ | √ | — | |
| Cortex-A57 Boot Address Register2 | CA57BAR2 | R/W | H'E616_0024 | H'0000_0000 | 32 | ALL *1 | √ | √ | √ | — | |
| Cortex-A57 CPU0 Boot Address Register for 64-bit mode H | CA57CPU0BA RH | R/W | H'E616_00C0 | H'-000_0000 | 32 | ALL *1 | √ | √ | √ | — | |
| Cortex-A57 CPU0 Boot Address Register for 64-bit mode L | CA57CPU0BA RL | R/W | H'E616_00C4 | H'----_---- | 32 | ALL *1 | √ | √ | √ | — | |
| Cortex-A57 CPU1 Boot Address Register for 64-bit mode H | CA57CPU1BA RH | R/W | H'E616_00D0 | H'-000_0000 | 32 | ALL *1 | √ | √ | √ | — | |
| Cortex-A57 CPU1 Boot Address Register for 64-bit mode L | CA57CPU1BA RL | R/W | H'E616_00D4 | H'----_---- | 32 | ALL *1 | √ | √ | √ | — | |
| Cortex-A57 CPU2 Boot Address Register for 64-bit mode H | CA57CPU2BA RH | R/W | H'E616_00E0 | H'-000_0000 | 32 | ALL *1 | √ | — | — | — | |
| Cortex-A57 CPU2 Boot Address Register for 64-bit mode L | CA57CPU2BA RL | R/W | H'E616_00E4 | H'----_---- | 32 | ALL *1 | √ | — | — | — | |
| Cortex-A57 CPU3 Boot Address Register for 64-bit mode H | CA57CPU3BA RH | R/W | H'E616_00F0 | H'-000_0000 | 32 | ALL *1 | √ | — | — | — | |
| Cortex-A57 CPU3 Boot Address Register for 64-bit mode L | CA57CPU3BA RL | R/W | H'E616_00F4 | H'----_---- | 32 | ALL *1 | √ | — | — | — | |
| Cortex-A53 CPU0 Boot Address Register for 64-bit mode H | CA53CPU0BA RH | R/W | H'E616_0080 | H'-000_0000 | 32 | ALL *1 | √ | √ | — | √ | |

| | | | | | | | Second Generation RZ/G Series Products | | | | |
|---|------------------|-----|-------------|---------------|------|-------------------|---|----------------------------|--------|--------|--|
| Register Name | Mnemonic | R/W | Address | Initial value | Size | Access permission | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Cortex-A53 CPU0 Boot Address Register for 64-bit mode L | CA53CPU0BA RL | R/W | H'E616_0084 | H'----_---- | 32 | ALL *1 | √ | √ | — | √ | |
| Cortex-A53 CPU1 Boot Address Register for 64-bit mode H | CA53CPU1BA RH | R/W | H'E616_0090 | H'-000_0000 | 32 | ALL *1 | √ | √ | — | √ | |
| Cortex-A53 CPU1 Boot Address Register for 64-bit mode L | CA53CPU1BA RL | R/W | H'E616_0094 | H'----_---- | 32 | ALL *1 | √ | √ | — | √ | |
| Cortex-A53 CPU2 Boot Address Register for 64-bit mode H | CA53CPU2BA RH | R/W | H'E616_00A0 | H'-000_0000 | 32 | ALL *1 | √ | √ | — | — | |
| Cortex-A53 CPU2 Boot Address Register for 64-bit mode L | CA53CPU2BA RL | R/W | H'E616_00A4 | H'----_---- | 32 | ALL *1 | √ | √ | — | — | |
| Cortex-A53 CPU3 Boot Address Register for 64-bit mode H | CA53CPU3BA RH | R/W | H'E616_00B0 | H'-000_0000 | 32 | ALL *1 | √ | √ | — | — | |
| Cortex-A53 CPU3 Boot Address Register for 64-bit mode L | CA53CPU3BA RL | R/W | H'E616_00B4 | H'----_---- | 32 | ALL *1 | √ | √ | — | — | |
| Standby Flag Register 0 | STBCHR0 | R/W | H'E616_0100 | H'0000_0000 | 32 | ALL *1 | √ | √ | √ | √ | |
| Standby Flag Register 1 | STBCHR1 | R/W | H'E616_0104 | H'0000_0000 | 32 | ALL *1 | √ | √ | √ | √ | |
| Standby Flag Register 2 | STBCHR2 | R/W | H'E616_0108 | H'0000_0000 | 32 | ALL *1 | √ | √ | √ | √ | |
| Standby Flag Register 3 | STBCHR3 | R/W | H'E616_010C | H'0000_0000 | 32 | ALL *1 | √ | √ | √ | √ | |
| Standby Flag Register 4 | STBCHR4 | R/W | H'E616_0120 | H'0000_0000 | 32 | ALL | √ | √ | √ | √ | |
| Standby Flag Register 5 | STBCHR5 | R/W | H'E616_0124 | H'0000_0000 | 32 | ALL | √ | √ | √ | √ | |
| Standby Flag Register 6 | STBCHR6 | R/W | H'E616_0128 | H'0000_0000 | 32 | ALL | √ | √ | √ | √ | |
| Standby Flag Register 7 | STBCHR7 | R/W | H'E616_012C | H'0000_0000 | 32 | ALL | √ | √ | √ | √ | |
| Soft Power On Reset Control Register | SRESR | R/W | H'E616_0110 | H'0000_0000 | 32 | ALL *1 | √ | √ | √ | √ | |
| SYS Reset Flag Register | SRSTFR | R/W | H'E616_0118 | H'0000_0000 | 32 | ALL *1 | √ | √ | √ | √ | |

Notes: 1. Write access is permitted only for secure CPU.

Table 17.3 Initialization condition of RST Register

| Register Name | Mnemonic | Initialization condition | Second Generation RZ/G Series Products | | | | |
|---|--------------|--|---|----------------------------|--------|--------|--|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Mode Monitor Register | MODEMR | PRESET# only | √ | √ | √ | √ | |
| Cortex-A57 Reset Control Register | CA57RESCNT | PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | √ | — | |
| Cortex-A53 Reset Control Register | CA53RESCNT | PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | — | √ | |
| Watchdog Timer Reset Control Register | WDTRSTCR | PRESET# only | √ | √ | √ | √ | |
| PRESETOUT# Control Register | RSTOUTCR | PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | √ | √ | |
| Cortex-A53 Boot Address Register | CA53BAR | PRESET#, WDT reset, Soft Power On Reset (SRESCR.SPRES = 1) | √ | √ | — | √ | |
| Cortex-A53 Boot Address Register2 | CA53BAR2 | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | — | √ | |
| Cortex-A57 Boot Address Register | CA57BAR | PRESET#, WDT reset, Soft Power On Reset (SRESCR.SPRES = 1) | √ | √ | √ | — | |
| Cortex-A57 Boot Address Register2 | CA57BAR2 | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | √ | — | |
| Cortex-A57 CPU0 Boot Address Register for 64-bit mode H | CA57CPU0BARH | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | √ | — | |
| Cortex-A57 CPU0 Boot Address Register for 64-bit mode L | CA57CPU0BARL | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | √ | — | |
| Cortex-A57 CPU1 Boot Address Register for 64-bit mode H | CA57CPU1BARH | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | √ | — | |
| Cortex-A57 CPU1 Boot Address Register for 64-bit mode L | CA57CPU1BARL | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | √ | — | |
| Cortex-A57 CPU2 Boot Address Register for 64-bit mode H | CA57CPU2BARH | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | — | — | — | |
| Cortex-A57 CPU2 Boot Address Register for 64-bit mode L | CA57CPU2BARL | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | — | — | — | |

| Register Name | Mnemonic | Initialization condition | Second Generation RZ/G Series Products | | | |
|---|---------------|--|---|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Cortex-A57 CPU3 Boot Address Register for 64-bit mode H | CA57CPU3BA RH | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | — | — | — |
| Cortex-A57 CPU3 Boot Address Register for 64-bit mode L | CA57CPU3BA RL | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | — | — | — |
| Cortex-A53 CPU0 Boot Address Register for 64-bit mode H | CA53CPU0BA RH | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | — | √ |
| Cortex-A53 CPU0 Boot Address Register for 64-bit mode L | CA53CPU0BA RL | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | — | √ |
| Cortex-A53 CPU1 Boot Address Register for 64-bit mode H | CA53CPU1BA RH | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | — | √ |
| Cortex-A53 CPU1 Boot Address Register for 64-bit mode L | CA53CPU1BA RL | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | — | √ |
| Cortex-A53 CPU2 Boot Address Register for 64-bit mode H | CA53CPU2BA RH | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | — | — |
| Cortex-A53 CPU2 Boot Address Register for 64-bit mode L | CA53CPU2BA RL | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | — | — |
| Cortex-A53 CPU3 Boot Address Register for 64-bit mode H | CA53CPU3BA RH | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | — | — |
| Cortex-A53 CPU3 Boot Address Register for 64-bit mode L | CA53CPU3BA RL | Depend on WDTRSTCR.RESBAR2S PRESET# only or PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | — | — |
| Standby Flag Register 0 | STBCHR0 | PRESET# only | √ | √ | √ | √ |
| Standby Flag Register 1 | STBCHR1 | PRESET# only | √ | √ | √ | √ |
| Standby Flag Register 2 | STBCHR2 | PRESET# only | √ | √ | √ | √ |
| Standby Flag Register 3 | STBCHR3 | PRESET# only | √ | √ | √ | √ |
| Standby Flag Register 4 | STBCHR4 | PRESET# only | √ | √ | √ | √ |
| Standby Flag Register 5 | STBCHR5 | PRESET# only | √ | √ | √ | √ |

| | | | Second Generation RZ/G Series Products | | | |
|---|----------|---|---|----------------------------|--------|--------|
| Register Name | Mnemonic | Initialization condition | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Standby Flag Register 6 | STBCHR6 | PRESET# only | √ | √ | √ | √ |
| Standby Flag Register 7 | STBCHR7 | PRESET# only | √ | √ | √ | √ |
| Soft Power On Reset Control Register | SRESCR | PRESET# only | √ | √ | √ | √ |
| SYS Reset Flag Register | SRSTFR | PRESET# only | √ | √ | √ | √ |
| Secure Protect Control/Status Register | SCPTCSR | PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | √ | √ |
| Secure Error Address Register | SCERADR | PRESET#, Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) | √ | √ | √ | √ |

17.1.5 Connected module

Table 17.4 Connected modules

| Module name | Connected module name | Function of connected module |
|-------------|-----------------------|---|
| RST | AP-System Core | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] Cortex-A53/ Cortex-A57 Boot address registers Cortex-A53/ Cortex-A57 Reset Control registers |
| | | [RZ/G2E] Cortex-A53 Boot address registers Cortex-A53 Reset Control registers |
| | | [RZ/G2N] Cortex-A57 Boot address registers Cortex-A57 Reset Control registers |
| | | CPG Reset control |
| | RWDT | Mask RWDT reset request |
| | System WatchDog Timer | Mask System WatchDog Timer reset request |

17.2 Register Description

Legend for Register Description

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R: Read-only.

W: Write-only. The read value is undefined.

17.2.1 Mode Monitor Register (MODEMR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

MODEMR is a 32-bit read-only register, which can be accessed only in longwords.

This register is initialized only by PRESET#.

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|-----|------|------|------|------|------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | MDT1 | MDT0 | MD28 | MD27 | MD26 | MD25 | — | MD23 | MD22 | MD21 | MD20 | MD19 | MD18 | MD17 | MD16 |
| Initial value: | 0 | — | — | — | — | — | — | 0 | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MD15 | MD14 | MD13 | MD12 | MD11 | MD10 | MD9 | MD8 | MD7 | MD6 | MD5 | MD4 | MD3 | MD2 | MD1 | MD0 |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

[RZ/G2E]

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|-----|------|-----|------|------|------|-----|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | MDT1 | MDT0 | — | — | MD26 | MD25 | — | MD23 | — | MD21 | MD20 | MD19 | — | MD17 | MD16 |
| Initial value: | 0 | — | — | 0 | 0 | — | — | 0 | — | 0 | — | — | — | 0 | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MD15 | — | MD13 | MD12 | MD11 | — | MD9 | MD8 | MD7 | MD6 | MD5 | MD4 | MD3 | MD2 | MD1 | MD0 |
| Initial value: | — | 0 | — | — | — | 0 | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|------------------------|-----|---|
| 31 | — | B'0 | R | Reserved When read, returns 0. |
| 30 to 29 | MDT[1:0] | Depend on mode setting | R | The value of MDT[1:0] |
| 28 | MD28 | Depend on mode setting | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] The value of MD28 |
| | — | B'0 | R | [RZ/G2E] Reserved When read, returns 0. |
| 27 | MD27 | Depend on mode setting | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] The value of MD27 |
| | — | B'0 | R | [RZ/G2E] Reserved When read, returns 0. |
| 26 to 25 | MD[26:25] | Depend on mode setting | R | The value of MD[26:25] |
| 24 | — | B'0 | R | Reserved When read, returns 0. |
| 23 | MD23 | Depend on mode setting | R | The value of MD23 |
| 22 | MD22 | Depend on mode setting | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] The value of MD22 |
| | — | B'0 | R | [RZ/G2E] Reserved When read, returns 0. |
| 21 | MD21 | Depend on mode setting | R | The value of MD21 |
| 20 | MD20 | Depend on mode setting | R | The value of MD20 |
| 19 | MD19 | Depend on mode setting | R | The value of MD19 |
| 18 | MD18 | Depend on mode setting | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] The value of MD18 |
| | — | B'0 | R | [RZ/G2E] Reserved When read, returns 0. |
| 17 | MD17 | Depend on mode setting | R | The value of MD17 |
| 16, 15 | MD[16:15] | Depend on mode setting | R | The value of MD[16:15] |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|------------------------|-----|---|
| 14 | MD14 | Depend on mode setting | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] The value of MD14 |
| | — | B'0 | R | [RZ/G2E] Reserved When read, returns 0. |
| 13 | MD13 | Depend on mode setting | R | The value of MD13 |
| 12 | MD12 | Depend on mode setting | R | The value of MD12 |
| 11 | MD11 | Depend on mode setting | R | The value of MD11 |
| 10 | MD10 | Depend on mode setting | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] The value of MD10 |
| | — | B'0 | R | [RZ/G2E] Reserved When read, returns 0. |
| 9 to 7 | MD[9:7] | Depend on mode setting | R | The value of MD[9:7] |
| 6 | MD6 | Depend on mode setting | R | The value of MD6 |
| 5 to 0 | MD[5:0] | Depend on mode setting | R | The value of MD[5:0] |

17.2.2 Cortex-A57 Reset Control Register (CA57RESCNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

CA57RESCNT is a 32-bit readable/writable register, which can be accessed only in longwords.

This register specifies whether the reset is issued to Cortex-A57 CPU n ($n = 0$ to 3) or not. The upper word of the write value should always be the code value H'A5A5. If the code value is read, it is always read as 0.

This register is initialized by PRESET# and Soft Power On Reset (WDT reset or SRESCR.SPRES=1).

[RZ/G2H]

| | | | | | | | | | | | | | | | | |
|----------------|---------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------------|------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Code value (H'A5A5) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CA57 CPU0R | CA57 CPU1R | CA57 CPU2R | CA57 CPU3R |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | * | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

Note: * When MD7 = 0 and MD6 = 0, the initial value of CA57CPU0R is 0. In the other setting, the initial value of CA57CPU0R is 1.

[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| | | | | | | | | | | | | | | | | |
|----------------|---------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------------|------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Code value (H'A5A5) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CA57 CPU0R | CA57 CPU1R | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | * | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R | R |

Note: * When MD7 = 0 and MD6 = 0, the initial value of CA57CPU0R is 0. In the other setting, the initial value of CA57CPU0R is 1.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 16 | — | H'0000 | R/W | Code value (H'A5A5) When read, returns 0. |
| 15 to 4 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 3 | CA57CPU0R | * | R/W | Issue reset to Cortex-A57 CPU0 0: Dessert the reset to Cortex-A57 CPU0 1: Assert the reset to Cortex-A57 CPU0 |
| 2 | CA57CPU1R | B'1 | R/W | Issue reset to Cortex-A57 CPU1 0: Dessert the reset to Cortex-A57 CPU1 1: Assert the reset to Cortex-A57 CPU1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------|---------------|-----|--|
| 1 | CA57CPU2R | B'1 | R/W | Issue reset to Cortex-A57 CPU2 [RZ/G2H] 0: Dessert the reset to Cortex-A57 CPU2 1: Assert the reset to Cortex-A57 CPU2 |
| | — | B'1 | R | Reserved [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] When read, returns 1. The write value should always be 1. |
| 0 | CA57CPU3R | B'1 | R/W | Issue reset to Cortex-A57 CPU3 [RZ/G2H] 0: Dessert the reset to Cortex-A57 CPU3 1: Assert the reset to Cortex-A57 CPU3 |
| | — | B'1 | R | Reserved [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] When read, returns 1. The write value should always be 1. |

Note: * When MD7 = 0 and MD6 = 0, the initial value of CA57CPU0R is 0. In the other setting, the initial value of CA57CPU0R is 1.

17.2.3 Cortex-A53 Reset Control Register (CA53RESCNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | √ |

CA53RESCNT is a 32-bit readable/writable register, which can be accessed only in longwords.

This register specifies whether the reset is issued to Cortex-A53 CPU_n (n = 0 to 3) or not. The upper word of the write value should always be the code value H'5A5A. If the code value is read, it is always read as 0.

This register is initialized by PRESET# and Soft Power On Reset (WDT reset or SRESCR.SPRES = 1).

| | | | | | | | | | | | | | | | | |
|----------------|---------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------------|------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Code value (H'5A5A) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CA53 CPU0R | CA53 CPU1R | CA53 CPU2R | CA53 CPU3R |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | *1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W*3 | R/W*2 | R/W*2 |

Notes: 1. When MD7 = 0 and MD6 = 1, the initial value of CA53CPU0R is 0. In the other setting, the initial value of CA53CPU0R is 1.

2. R/W of Bit 1 and Bit 0 depends on product model.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 : R/W

RZ/G2E: R only

3. R/W of Bit 2 depends on product model.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E: R/W

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 16 | — | H'0000 | R/W | Code value (H'5A5A) When read, returns 0. |
| 15 to 4 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 3 | CA53CPU0R | * | R/W | Issue reset to Cortex-A53 CPU0 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E] 0: Dessert the reset to Cortex-A53 CPU0 1: Assert the reset to Cortex-A53 CPU0 |
| 2 | CA53CPU1R | B'1 | R/W | Issue reset to Cortex-A53 CPU1 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E] 0: Dessert the reset to Cortex-A53 CPU1 1: Assert the reset to Cortex-A53 CPU1 |
| 1 | CA53CPU2R | B'1 | R/W | Issue reset to Cortex-A53 CPU2 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] 0: Dessert the reset to Cortex-A53 CPU2 1: Assert the reset to Cortex-A53 CPU2 |
| | — | B'1 | R | Reserved [RZ/G2E] When read, returns 1. The write value should always be 1. |
| 0 | CA53CPU3R | B'1 | R/W | Issue reset to Cortex-A53 CPU3 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] 0: Dessert the reset to Cortex-A53 CPU3 1: Assert the reset to Cortex-A53 CPU3 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| | — | B'1 | R | Reserved [RZ/G2E] When read, returns 1. The write value should always be 1. |

Note: * When MD7 = 0 and MD6 = 1, the initial value of CA53CPU0R is 0. In the other setting, the initial value of CA53CPU0R is 1.

17.2.4 Watchdog Timer Reset Control Register (WDTRSTCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

WDTRSTCR is a 32-bit readable/writable register, which can be accessed only in longwords. This register can be changed only in the secure mode. This register specifies whether the watchdog timer overflow should be masked or not as a reset trigger. The upper word of the write value should always be the code value H'A55A. If the code value is read, it is always read as 0.

This register is initialized only by PRESET#.

| | | | | | | | | | | | | | | | | |
|----------------|---------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----------------|----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Code value (H'A55A) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RESBA R2S | — | — | — | — | — | — | — | — | — | — | — | — | — | SWDT_R STMSK | RWDTR STMSK |
| Initial value: | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| R/W: | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | H'0000 | R/W | Code value (H'A55A) When read, returns 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|--|
| 15 | RESBAR2S | B'1 | R/W | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>Select BAR2 registers reset condition</p> <p>0: CA57BAR2, CA53BAR2, CR7BAR2, CA57CPUUnBARH, CA57CPUUnBARL, CA53CPUUnBARH, and CA53CPUUnBARL are initialized only by PRESET#.</p> <p>1: CA57BAR2, CA53BAR2, CR7BAR2, CA57CPUUnBARH, CA57CPUUnBARL, CA53CPUUnBARH, and CA53CPUUnBARL are initialized by PRESET#, WDT reset, and Soft Power On Reset (SRESCR.SPRES = 1).</p> <hr/> <p>[RZ/G2E]</p> <p>Select BAR2 registers reset condition</p> <p>0: CA53BAR2, CR7BAR2, CA53CPUUnBARH, and CA53CPUUnBARL are initialized only by PRESET#.</p> <p>1: CA53BAR2, CR7BAR2, CA53CPUUnBARH, and CA53CPUUnBARL are initialized by PRESET#, WDT reset, and Soft Power On Reset (SRESCR.SPRES = 1).</p> <hr/> <p>[RZ/G2N]</p> <p>Select BAR2 registers reset condition</p> <p>0: CA57BAR2, CR7BAR2, CA57CPUUnBARH, and CA57CPUUnBARL are initialized only by PRESET#.</p> <p>1: CA57BAR2, CR7BAR2, CA57CPUUnBARH, and CA57CPUUnBARL are initialized by PRESET#, WDT reset, and Soft Power On Reset (SRESCR.SPRES = 1).</p> |
| 14 | — | B'1 | R/W | <p>Reserved</p> <p>When read, returns 1. The write value should always be 1.</p> |
| 13 to 4 | — | All 0 | R | <p>Reserved</p> <p>When read, returns 0. The write value should always be 0.</p> |
| 3 | — | B'0 | R | <p>Reserved</p> <p>When read, returns 0. The write value should always be 0.</p> |
| 2 | — | B'0 | R | <p>Reserved</p> <p>When read, returns 0. The write value should always be 0.</p> |
| 1 | SWDT_ RSTMSK | B'1 | R/W | <p>System WatchDog Timer Reset Mask</p> <p>This bit is used to mask the detection of System WatchDog Timer overflow.</p> <p>0: Reset request</p> <p>1: Not reset request</p> |
| 0 | RWDT_ RSTMSK | B'1 | R/W | <p>RWDT Reset Mask</p> <p>This bit is used to mask the detection of RWDT overflow.</p> <p>0: Reset request</p> <p>1: Not reset request</p> |

17.2.5 PRESETOUT# Control Register (RSTOUTCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

RSTOUTCR is a 32-bit readable/writable register, which can be accessed only in longwords.

This register controls the output level of external LSI pin RESETOUT# by software.

This register is initialized by PRESET# and Soft Power On Reset (WDT reset or SRESCR.SPRES = 1).

When this register is initialized, PRESET# is asserted.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | RES OUT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 4 | — | H'000_0000 | R | Reserved When read, returns 0. The write value should always be 0. |
| 3 | — | B'0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 2 | — | B'0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 1 | — | B'0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 0 | RESOUT | B'1 | R/W | PRESETOUT# control by software This bit is used to specify the output level of PRESETOUT#. 0: PRESETOUT# is asserted 1: PRESETOUT# is negated |

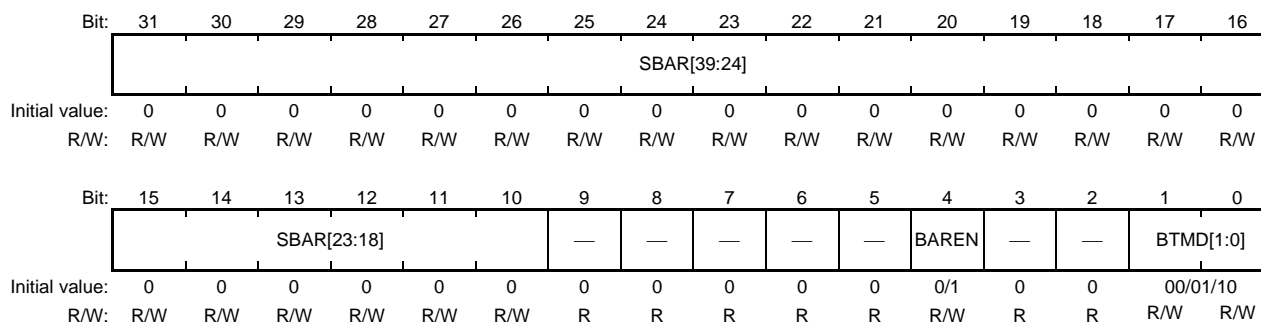
17.2.6 Cortex-A53 Boot Address Register (CA53BAR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | √ |

CA53BAR is a 32-bit readable/writable register, which can be accessed only in longwords.

This register specifies the boot space of the System CPU (Cortex-A53).

This register is initialized by PRESET# and Soft Power On Reset (WDT reset or SRESCR.SPRES=1).



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|----------------------|-----|--|
| 31 to 10 | SBAR[39:18] | {H'0_0000, B'00} | R/W | System CPU (Cortex-A53) Boot Address When System CPU (Cortex-A53) accesses to the range of physical address from H'00_0000_0000 to H'00_0003_FFFF, the address bit in [39:18] is replaced by SBAR[39:18]. |
| 9 to 5 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 4 | BAREN*2 | B'0/ B'1*1 | R/W | BAREN bit 0: SBAR is not valid. 1: SBAR is valid. |
| 3, 2 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 1, 0 | BTMD[1:0] | B'00/B'01/ B'10*1 | R/W | Specifies the Boot area of System CPU (Cortex-A53) During Hardware Power On Reset the initial value of these bits depends upon MD4, MD3, MD2 and MD1 pin setting. 00: SBAR[39:18] is assigned to Boot address, set SBAR[17:0] to 0. B'01: Prohibited. B'10: Boots from BootROM. B'11: Prohibited. |

- Notes: 1. Initial value depends upon MD4, MD3, MD2 and MD1 pin setting. Refer to the Table 17.5.
2. Do not rewrite BAREN = 1 simultaneously with SBAR and BTMD. After completing the setting of SBAR and BTMD with BAREN = 0, set only BAREN = 1.
- For example: Change SBAR[39:18] from A to B, and BTMD[1:0] = B'00
- Step 1: Write SBAR [39:18] = B, BAREN = 0, BTMD[1:0] = B'00
- Step 2: Write SBAR [39:18] = B, BAREN = 1, BTMD[1:0] = B'00

Table 17.5 Initial values of BAREN, BTMD[1:0], and boot address

| Mode pin setting | | | | | Initial values | | | Note |
|------------------|-----|-----|-----|-----|----------------|-----------|----------------|----------------------------|
| MD15 | MD4 | MD3 | MD2 | MD1 | BAREN | BTMD[1:0] | Boot address | |
| 0 | 0 | 0 | 0 | 0 | 0 | B'00 | H'00_0000_0000 | |
| 0 | 0 | 0 | 0 | 1 | 1 | B'01 | H'00_0800_0000 | |
| 0 | 0 | 0 | 1 | 0 | 1 | B'10 | H'00_EB10_0000 | |
| 0 | 0 | 0 | 1 | 1 | 1 | B'10 | H'00_EB10_0000 | |
| 0 | 0 | 1 | 0 | 0 | 1 | B'10 | H'00_EB10_0000 | |
| 0 | 0 | 1 | 0 | 1 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 0 | 1 | 1 | 0 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 0 | 1 | 1 | 1 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 1 | 0 | 0 | 0 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 1 | 0 | 0 | 1 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 1 | 0 | 1 | 0 | 1 | B'01 | H'00_0800_0000 | |
| 0 | 1 | 0 | 1 | 1 | 1 | B'01 | H'00_0800_0000 | |
| 0 | 1 | 1 | 0 | 0 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 1 | 1 | 0 | 1 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 1 | 1 | 1 | 0 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 1 | 1 | 1 | 1 | 1 | B'10 | H'00_EB10_0000 | |
| 1 | 0 | 0 | 0 | 0 | 0 | B'00 | H'00_0000_0000 | |
| 1 | 0 | 0 | 0 | 1 | 0 | B'01 | H'00_0000_0000 | |
| 1 | 0 | 0 | 1 | 0 | 0 | B'10 | H'00_0000_0000 | |
| 1 | 0 | 0 | 1 | 1 | 0 | B'10 | H'00_0000_0000 | |
| 1 | 0 | 1 | 0 | 0 | 0 | B'10 | H'00_0000_0000 | |
| 1 | 0 | 1 | 0 | 1 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 0 | 1 | 1 | 0 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 0 | 1 | 1 | 1 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 0 | 0 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 0 | 1 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 1 | 0 | 0 | B'01 | H'00_0000_0000 | |
| 1 | 1 | 0 | 1 | 1 | 0 | B'01 | H'00_0000_0000 | |
| 1 | 1 | 1 | 0 | 0 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 0 | 1 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 1 | 0 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 1 | 1 | 0 | B'10 | H'00_0000_0000 | |

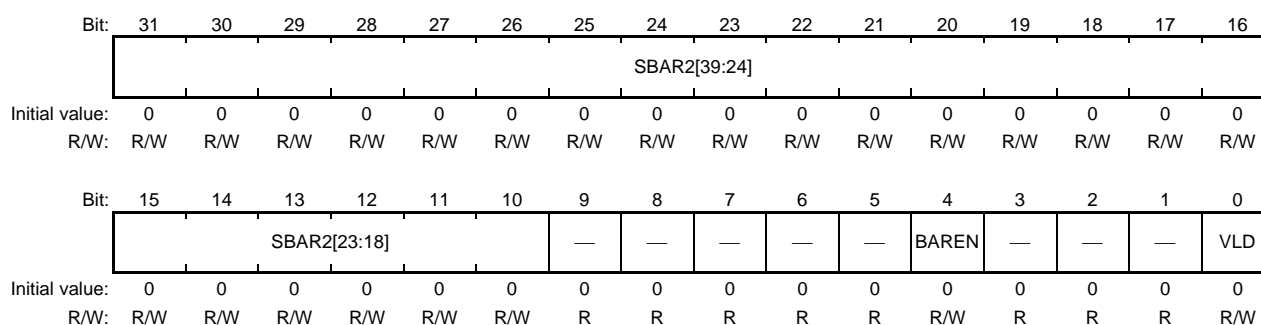
17.2.7 Cortex-A53 Boot Address Register2 (CA53BAR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | √ |

CA53BAR2 is a 32-bit readable/writable register, which can be accessed only in longwords. This register can be changed only in the secure mode.

This register specifies the boot space of the System CPU (Cortex-A53).

This register is initialized by PRESET# when WDTRSTCR.RESBAR2S = 0. This register is initialized by PRESET# and Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) when WDTRSTCR.RESBAR2S = 1.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|------------------|-----|---|
| 31 to 10 | SBAR2[39:18] | {H'0_0000, B'00} | R/W | System CPU (Cortex-A53) Boot Address2 When System CPU (Cortex-A53) accesses to the range of physical address from H'00_0000_0000 to H'00_0003_FFFF, the address bit in [39:18] is replaced by SBAR2[39:18]. |
| 9 to 5 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 4 | BAREN* | B'0 | R/W | BAREN bit 0: SBAR2 is not valid. 1: SBAR2 is valid. |
| 3 to 1 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 0 | VLD | B'0 | R/W | VALID bit When 1 is set to this bit, System CPU (Cortex-A53) starts only from the address set to SBAR2. 0: System CPU (Cortex-A53) starts from the address set to SBAR. 1: System CPU (Cortex-A53) starts from the address set to SBAR2. |

Note: * Do not rewrite BAREN = 1 simultaneously with SBAR2 and VLD. After completing the setting of SBAR2 and VLD with BAREN = 0, set only BAREN = 1.

For example: Change SBAR2 [39:18] from A to B, and VLD=1

Step 1: Write SBAR2[39:18] = B, BAREN = 0, VLD = 1

Step 2: Write SBAR2[39:18] = B, BAREN = 1, VLD = 1

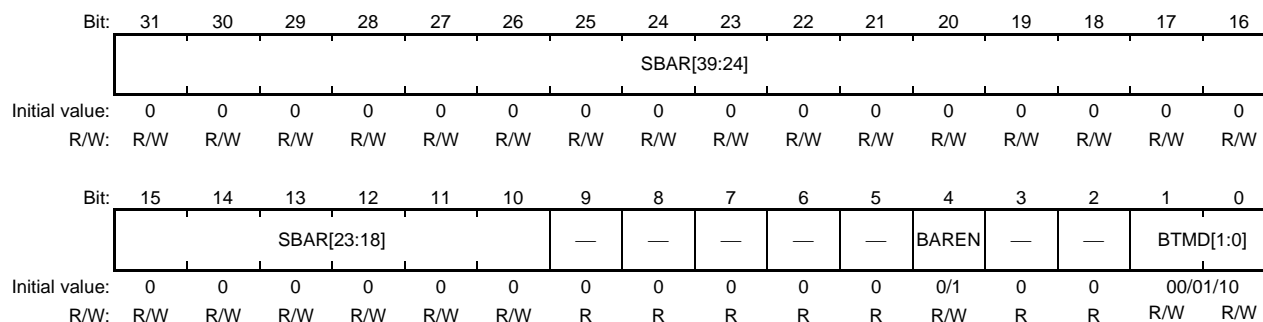
17.2.8 Cortex-A57 Boot Address Register (CA57BAR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

CA57BAR is a 32-bit readable/writable register, which can be accessed only in longwords.

This register specifies the boot space of the System CPU (Cortex-A57).

This register is initialized by PRESET# and Soft Power On Reset (WDT reset or SRESCR.SPRES = 1).



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|--------------------|-----|--|
| 31 to 10 | SBAR[39:18] | {H'0_0000, B'00} | R/W | System CPU (Cortex-A57) Boot Address When System CPU (Cortex-A57) accesses to the range of physical address from H'00_0000_0000 to H'00_0003_FFFF, the address bit in [39:18] is replaced by SBAR[39:18]. |
| 9 to 5 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 4 | BAREN*2 | B'0/ B'1*1 | R/W | BAREN bit 0: SBAR is not valid. 1: SBAR is valid. |
| 3, 2 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 1, 0 | BTMD[1:0] | B'00/ B'01/ B'10*1 | R/W | Specifies the Boot area of System CPU (Cortex-A57) During Hardware Power On Reset the initial value of these bits depends upon MD4, MD3, MD2 and MD1 pin setting. 00: SBAR[39:18] is assigned to Boot address, set SBAR[17:0] to 0. B'01: Prohibited. B'10: Boots from BootROM. B'11: Prohibited. |

Notes: 1. Initial value depends upon MD4, MD3, MD2 and MD1 pin setting. Refer to the Table 17.6.

2. Do not rewrite BAREN = 1 simultaneously with SBAR and BTMD. After completing the setting of SBAR and BTMD with BAREN = 0, set only BAREN = 1.

For example: Change SBAR[39:18] from A to B, and BTMD[1:0] = B'00

Step 1: Write SBAR[39:18] = B, BAREN = 0, BTMD[1:0] = B'00

Step 2: Write SBAR[39:18] = B, BAREN = 1, BTMD[1:0] = B'00

Table 17.6 Initial values of BAREN, BTMD[1:0], and boot address

| Mode pin setting | | | | | Initial values | | Boot address | Note |
|------------------|-----|-----|-----|-----|----------------|-----------|----------------|----------------------------|
| MD15 | MD4 | MD3 | MD2 | MD1 | BAREN | BTMD[1:0] | | |
| 0 | 0 | 0 | 0 | 0 | 0 | B'00 | H'00_0000_0000 | |
| 0 | 0 | 0 | 0 | 1 | 1 | B'01 | H'00_0800_0000 | |
| 0 | 0 | 0 | 1 | 0 | 1 | B'10 | H'00_EB10_0000 | |
| 0 | 0 | 0 | 1 | 1 | 1 | B'10 | H'00_EB10_0000 | |
| 0 | 0 | 1 | 0 | 0 | 1 | B'10 | H'00_EB10_0000 | |
| 0 | 0 | 1 | 0 | 1 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 0 | 1 | 1 | 0 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 0 | 1 | 1 | 1 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 1 | 0 | 0 | 0 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 1 | 0 | 0 | 1 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 1 | 0 | 1 | 0 | 1 | B'01 | H'00_0800_0000 | |
| 0 | 1 | 0 | 1 | 1 | 1 | B'01 | H'00_0800_0000 | |
| 0 | 1 | 1 | 0 | 0 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 1 | 1 | 0 | 1 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 1 | 1 | 1 | 0 | 1 | B'10 | H'00_EB10_0000 | This setting is prohibited |
| 0 | 1 | 1 | 1 | 1 | 1 | B'10 | H'00_EB10_0000 | |
| 1 | 0 | 0 | 0 | 0 | 0 | B'00 | H'00_0000_0000 | |
| 1 | 0 | 0 | 0 | 1 | 0 | B'01 | H'00_0000_0000 | |
| 1 | 0 | 0 | 1 | 0 | 0 | B'10 | H'00_0000_0000 | |
| 1 | 0 | 0 | 1 | 1 | 0 | B'10 | H'00_0000_0000 | |
| 1 | 0 | 1 | 0 | 0 | 0 | B'10 | H'00_0000_0000 | |
| 1 | 0 | 1 | 0 | 1 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 0 | 1 | 1 | 0 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 0 | 1 | 1 | 1 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 0 | 0 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 0 | 1 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 1 | 0 | 0 | B'01 | H'00_0000_0000 | |
| 1 | 1 | 0 | 1 | 1 | 0 | B'01 | H'00_0000_0000 | |
| 1 | 1 | 1 | 0 | 0 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 0 | 1 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 1 | 0 | 0 | B'10 | H'00_0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 1 | 1 | 0 | B'10 | H'00_0000_0000 | |

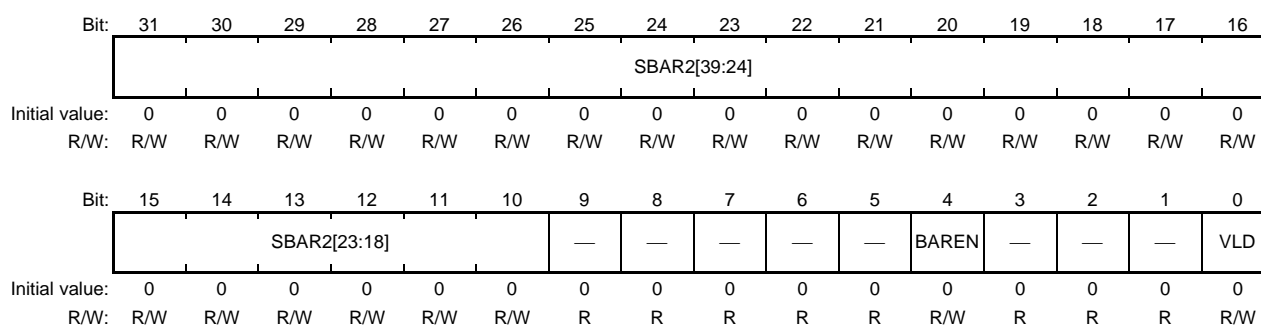
17.2.9 Cortex-A57 Boot Address Register2 (CA57BAR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

CA57BAR2 is a 32-bit readable/writable register, which can be accessed only in longwords. This register can be changed only in the secure mode.

This register specifies the boot space of the System CPU (Cortex-A57).

This register is initialized by PRESET# when WDTRSTCR.RESBAR2S = 0. This register is initialized by PRESET# and Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) when WDTRSTCR.RESBAR2S = 1.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|------------------|-----|---|
| 31 to 10 | SBAR2[39:18] | {H'0_0000, B'00} | R/W | System CPU (Cortex-A57) Boot Address2 When System CPU (Cortex-A57) accesses to the range of physical address from H'00_0000_0000 to H'00_0003_FFFF, the address bit in [39:18] is replaced by SBAR2[39:18]. |
| 9 to 5 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 4 | BAREN* | B'0 | R/W | BAREN bit 0: SBAR2 is not valid. 1: SBAR2 is valid. |
| 3 to 1 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 0 | VLD | B'0 | R/W | VALID bit When 1 is set to this bit, System CPU (Cortex-A57) starts only from the address set to SBAR2. 0: System CPU (Cortex-A57) starts from the address set to SBAR. 1: System CPU (Cortex-A57) starts from the address set to SBAR2. |

Note: * Do not rewrite BAREN = 1 simultaneously with SBAR2 and VLD. After completing the setting of SBAR2 and VLD with BAREN = 0, set only BAREN = 1.

For example: Change SBAR2[39:18] from A to B, and VLD = 1

Step 1: Write SBAR2[39:18] = B, BAREN = 0, VLD = 1

Step 2: Write SBAR2[39:18] = B, BAREN = 1, VLD = 1

17.2.10 Cortex-A57 CPU_n (n = 0 to 3[RZ/G2H], n = 0 and 1[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]) Boot Address Register for 64-bit mode H (CA57CPU_nBARH)

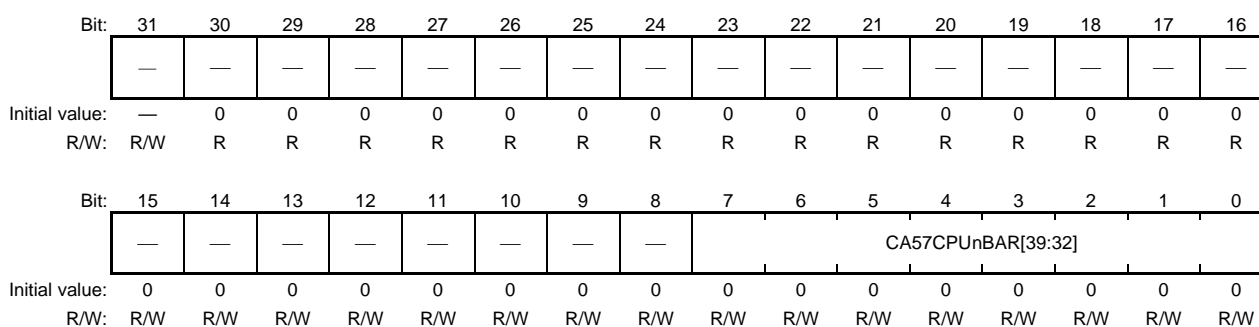
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

CA57CPU_nBARH is a 32-bit readable/writable register, which can be accessed only in longwords. This register can be changed only in the secure mode.

This register specifies the boot space of the Cortex-A57 in 64-bit addressing mode.

This register is initialized by PRESET# when WDTRSTCR.RESBAR2S = 0. This register is initialized by PRESET# and Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) when WDTRSTCR.RESBAR2S = 1.

Note: Each RZ/G Series 2nd Generation product provides these registers as shown in Table 17.2.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------------------|---------------|-----|---|
| 31 | — | — | R/W | Reserved Initial read value is undefined. The write value should always be read value. |
| 30 to 16 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 15 to 8 | — | All 0 | R/W | Reserved When read returns 0. The write value should always be 0. |
| 7 to 0 | CA57CPU _n BAR [39:32] | H'00 | R/W | Cortex-A57 CPU _n Boot Address in 64-bit mode |

17.2.11 Cortex-A57 CPU_n (n = 0 to 3[RZ/G2H], n = 0 and 1[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]) Boot Address Register for 64-bit mode L (CA57CPU_nBARL)

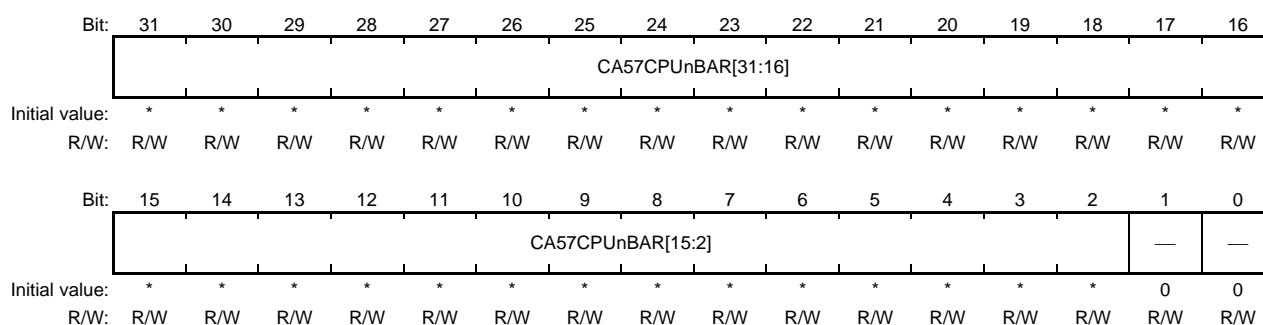
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

CA57CPU_nBARL is a 32-bit readable/writable register, which can be accessed only in longwords. This register can be changed only in the secure mode.

This register specifies the boot space of the Cortex-A57 in 64-bit addressing mode.

This register is initialized by PRESET# when WDTRSTCR.RESBAR2S = 0. This register is initialized by PRESET# and Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) when WDTRSTCR.RESBAR2S = 1.

Note: Each RZ/G Series 2nd Generation product provides these registers as shown in Table 17.2.



Note * Refer to the Table 17.7 and Table 17.8.

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------------------|---------------|-----|---|
| 31 to 2 | CA57CPU _n BAR[31:2] | * | R/W | Cortex-A57 CPU _n Boot Address (32 to 2) in 64-bit mode |
| 1 to 0 | — | All 0 | R/W | Reserved When read, returns 0. The write value should always be 0. |

Note: * Refer to the Table 17.7 and Table 17.8.

Table 17.7 Initial value of CA57CPU_nBARL (n = 0 to 3 [RZ/G2H], n = 0 to 1 [RZ/G2N])

| Mode pin setting | | | | | Initial value of register | | Note | |
|------------------|------------|-----|-----|-----|---------------------------|---|----------------------------|--|
| MD15 | MD4 | MD3 | MD2 | MD1 | CA57CPU0BARL | CA57CPU _n BARL (n = 1 to 3) | | |
| 0 | don't care | | | | | H'0000_0000 | H'0000_0000 | |
| 1 | 0 | 0 | 0 | 0 | H'0000_0000 | H'0000_0000 | | |
| 1 | 0 | 0 | 0 | 1 | H'0800_0000 | H'0000_0000 | | |
| 1 | 0 | 0 | 1 | 0 | H'EB11_2800 | H'0000_0000 | | |
| 1 | 0 | 0 | 1 | 1 | H'EB11_2800 | H'0000_0000 | | |
| 1 | 0 | 1 | 0 | 0 | H'EB11_2800 | H'0000_0000 | | |
| 1 | 0 | 1 | 0 | 1 | H'EB11_2800 | H'0000_0000 | This setting is prohibited | |
| 1 | 0 | 1 | 1 | 0 | H'EB11_2800 | H'0000_0000 | This setting is prohibited | |
| 1 | 0 | 1 | 1 | 1 | H'EB11_2800 | H'0000_0000 | This setting is prohibited | |
| 1 | 1 | 0 | 0 | 0 | H'EB11_2800 | H'0000_0000 | This setting is prohibited | |
| 1 | 1 | 0 | 0 | 1 | H'EB11_2800 | H'0000_0000 | This setting is prohibited | |
| 1 | 1 | 0 | 1 | 0 | H'0800_0000 | H'0000_0000 | | |
| 1 | 1 | 0 | 1 | 1 | H'0800_0000 | H'0000_0000 | | |
| 1 | 1 | 1 | 0 | 0 | H'EB11_2800 | H'0000_0000 | This setting is prohibited | |
| 1 | 1 | 1 | 0 | 1 | H'EB11_2800 | H'0000_0000 | This setting is prohibited | |
| 1 | 1 | 1 | 1 | 0 | H'EB11_2800 | H'0000_0000 | This setting is prohibited | |
| 1 | 1 | 1 | 1 | 1 | H'EB11_2800 | H'0000_0000 | | |

Table 17.8 Initial value of CA57CPU_nBARL (n = 0 to 1) [RZ/G2M V1.3, RZ/G2M V3.0]

| Mode pin setting | | | | | Initial value of register | | Note | |
|------------------|------------|-----|-----|-----|---------------------------|--------------------------------------|-------------|----------------------------|
| MD15 | MD4 | MD3 | MD2 | MD1 | CA57CPU0BARL | CA57CPU _n BARL (n = 1) | | |
| 0 | don't care | | | | | H'0000_0000 | H'0000_0000 | |
| 1 | 0 | 0 | 0 | 0 | | H'0000_0000 | H'0000_0000 | |
| 1 | 0 | 0 | 0 | 1 | | H'0800_0000 | H'0000_0000 | |
| 1 | 0 | 0 | 1 | 0 | | H'EB10_C000 | H'0000_0000 | |
| 1 | 0 | 0 | 1 | 1 | | H'EB10_C000 | H'0000_0000 | |
| 1 | 0 | 1 | 0 | 0 | | H'EB10_C000 | H'0000_0000 | |
| 1 | 0 | 1 | 0 | 1 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 0 | 1 | 1 | 0 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 0 | 1 | 1 | 1 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 0 | 0 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 0 | 1 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 1 | 0 | | H'0800_0000 | H'0000_0000 | |
| 1 | 1 | 0 | 1 | 1 | | H'0800_0000 | H'0000_0000 | |
| 1 | 1 | 1 | 0 | 0 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 0 | 1 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 1 | 0 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 1 | 1 | | H'EB10_C000 | H'0000_0000 | |

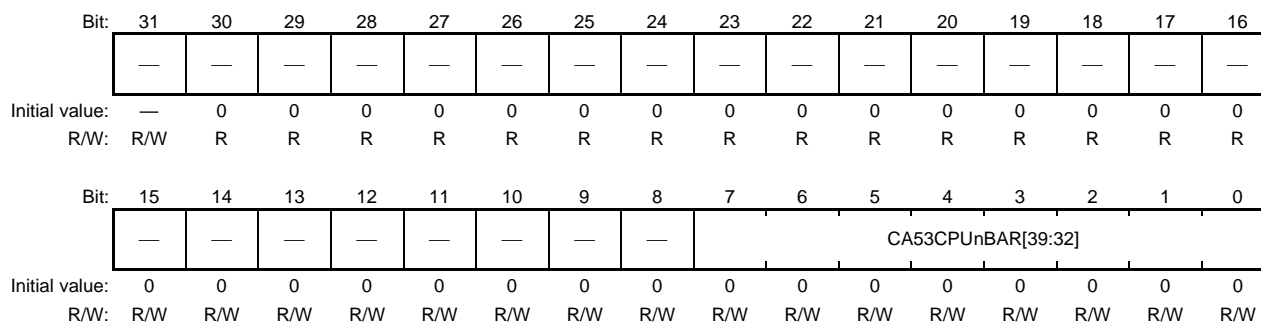
**17.2.12 Cortex-A53 CPU_n (n = 0 to 3[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0], n = 0 and 1[RZ/G2E])
Boot Address Register for 64-bit mode H (CA53CPU_nBARH)**

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | √ |

CA53CPU_nBARH is a 32-bit readable/writable register, which can be accessed only in longwords. This register can be changed only in the secure mode.

This register specifies the boot space of the Cortex-A53 in 64-bit addressing mode.

This register is initialized by PRESET# when WDTRSTCR.RESBAR2S = 0. This register is initialized by PRESET# and Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) when WDTRSTCR.RESBAR2S = 1.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------------------|---------------|-----|---|
| 31 | — | — | R/W | Reserved Initial read value is undefined. The write value should always be read value. |
| 30 to 16 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 15 to 8 | — | All 0 | R/W | Reserved The write value should always be 0. |
| 7 to 0 | CA53CPU _n BAR [39:32] | H'00 | R/W | Cortex-A53 CPU _n Boot Address in 64-bit mode |

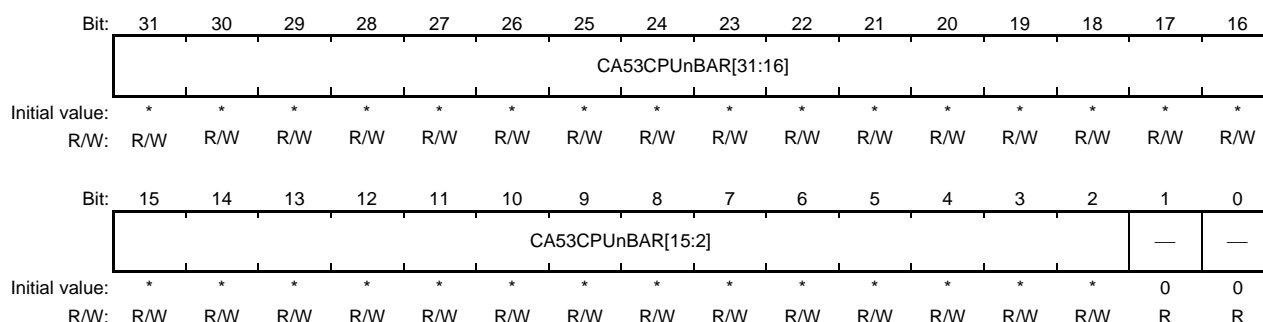
**17.2.13 Cortex-A53 CPU_n (n = 0 to 3[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0], n = 0 and 1[RZ/G2E])
Boot Address Register for 64-bit mode L (CA53CPU_nBARL)**

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | √ |

CA53CPU_nBARL is a 32-bit readable/writable register, which can be accessed only in longwords. This register can be changed only in the secure mode.

This register specifies the boot space of the Cortex-A53 in 64-bit addressing mode.

This register is initialized by PRESET# when WDTRSTCR.RESBAR2S = 0. This register is initialized by PRESET# and Soft Power On Reset (WDT reset or SRESCR.SPRES = 1) when WDTRSTCR.RESBAR2S = 1.



Note: * Refer to the Table 17.9. Initial value of CA53CPU_nBARL_n (n = 0 to 3 [RZ/G2H], n = 0 and 1[RZ/G2E])
Refer to the Table 17.10. Initial value of CA53CPU_nBARL_n (n = 0 to 3 [RZ/G2M V1.3, RZ/G2M V3.0])

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------------------|---------------|-----|---|
| 31 to 2 | CA53CPU _n BAR [31:2] | * | R/W | Cortex-A53 CPU _n Boot Address in 64-bit mode |
| 1 to 0 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |

Note: * Refer to the Table 17.9 and Table 17.10.

Table 17.9 Initial value of CA53CPU_nBARL (n = 0 to 3 [RZ/G2H], n = 0 to 1 [RZ/G2E])

| Mode pin setting | | | | | Initial value of register | | Note |
|------------------|------------|-----|-----|-----|---------------------------|---|----------------------------|
| MD15 | MD4 | MD3 | MD2 | MD1 | CA53CPU0BARL | CA53CPU _n BARL (n = 1 to 3) | |
| 0 | don't care | | | | H'0000_0000 | H'0000_0000 | |
| 1 | 0 | 0 | 0 | 0 | H'0000_0000 | H'0000_0000 | |
| 1 | 0 | 0 | 0 | 1 | H'0800_0000 | H'0000_0000 | |
| 1 | 0 | 0 | 1 | 0 | H'EB11_2800 | H'0000_0000 | |
| 1 | 0 | 0 | 1 | 1 | H'EB11_2800 | H'0000_0000 | |
| 1 | 0 | 1 | 0 | 0 | H'EB11_2800 | H'0000_0000 | |
| 1 | 0 | 1 | 0 | 1 | H'EB11_2800 | H'0000_0000 | This setting is prohibited |
| 1 | 0 | 1 | 1 | 0 | H'EB11_2800 | H'0000_0000 | This setting is prohibited |
| 1 | 0 | 1 | 1 | 1 | H'EB11_2800 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 0 | 0 | H'EB11_2800 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 0 | 1 | H'EB11_2800 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 1 | 0 | H'0800_0000 | H'0000_0000 | |
| 1 | 1 | 0 | 1 | 1 | H'0800_0000 | H'0000_0000 | |
| 1 | 1 | 1 | 0 | 0 | H'EB11_2800 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 0 | 1 | H'EB11_2800 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 1 | 0 | H'EB11_2800 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 1 | 1 | H'EB11_2800 | H'0000_0000 | |

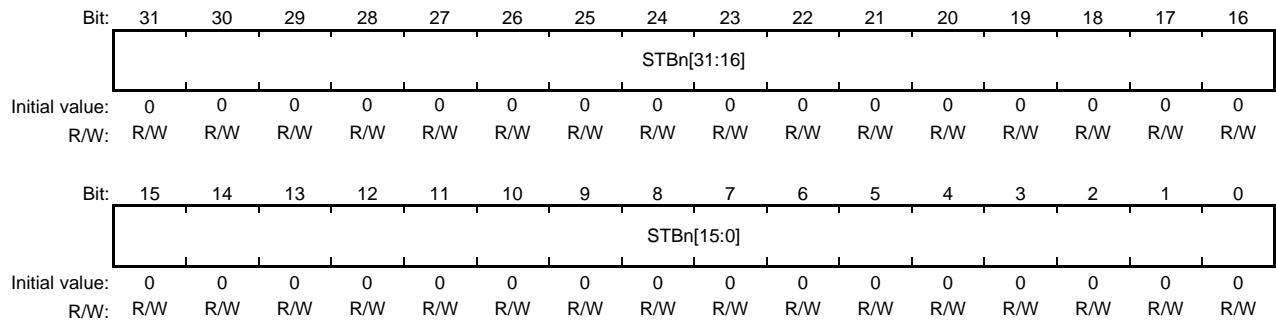
Table 17.10 Initial value of CA53CPU_nBARL (n = 0 to 3 [RZ/G2M V1.3, RZ/G2M V3.0])

| Mode pin setting | | | | | Initial value of register | | Note | |
|------------------|------------|-----|-----|-----|---------------------------|---|-------------|----------------------------|
| MD15 | MD4 | MD3 | MD2 | MD1 | CA53CPU0BARL | CA53CPU _n BARL (n = 1 to 3) | | |
| 0 | don't care | | | | | H'0000_0000 | H'0000_0000 | |
| 1 | 0 | 0 | 0 | 0 | | H'0000_0000 | H'0000_0000 | |
| 1 | 0 | 0 | 0 | 1 | | H'0800_0000 | H'0000_0000 | |
| 1 | 0 | 0 | 1 | 0 | | H'EB10_C000 | H'0000_0000 | |
| 1 | 0 | 0 | 1 | 1 | | H'EB10_C000 | H'0000_0000 | |
| 1 | 0 | 1 | 0 | 0 | | H'EB10_C000 | H'0000_0000 | |
| 1 | 0 | 1 | 0 | 1 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 0 | 1 | 1 | 0 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 0 | 1 | 1 | 1 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 0 | 0 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 0 | 1 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 0 | 1 | 0 | | H'0800_0000 | H'0000_0000 | |
| 1 | 1 | 0 | 1 | 1 | | H'0800_0000 | H'0000_0000 | |
| 1 | 1 | 1 | 0 | 0 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 0 | 1 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 1 | 0 | | H'EB10_C000 | H'0000_0000 | This setting is prohibited |
| 1 | 1 | 1 | 1 | 1 | | H'EB10_C000 | H'0000_0000 | |

17.2.14 Standby Flag Register n (STBCHRn) (n = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

STBCHRn are 32-bit readable/writable registers. STBCHR0 to 3 can be changed only in the secure mode. STBCHR4 to 7 can be changed even if in the non-secure mode (no access protection). These registers are defined purely for software purpose. Setting these registers doesn't influence the LSI operation. As these bits are initialized only by PRESET#, the programmer can use this register at their convenience.

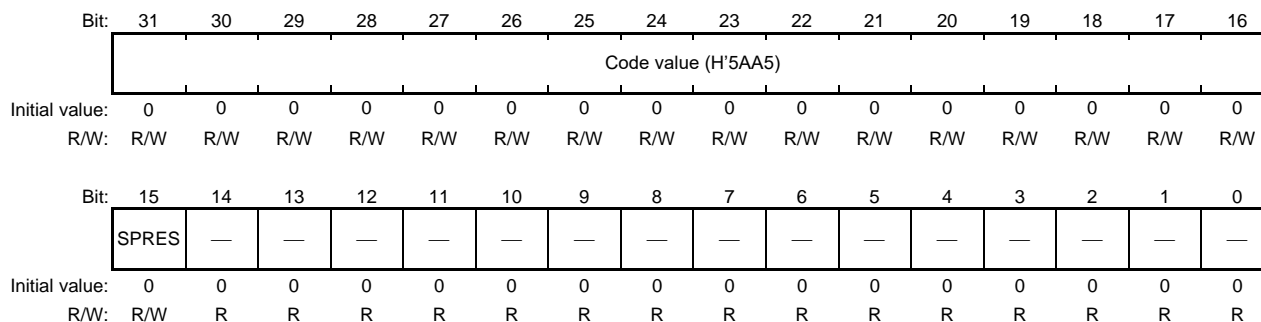


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|-------------------------------------|
| 31 to 0 | STBn[31:0] | H'0000_0000 | R/W | defined purely for software purpose |

17.2.15 Soft Power On Reset Control Register (SRESR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SRESR is a 32bit readable/writable register. This register can be changed only in the secure mode. This register is initialized only by PRESET#. The upper word of the write value should always be the code value H'5AA5. If the code value is read, it is always read as 0.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | H'0000 | R/W | Code value (H'5AA5) When read, returns 0. |
| 15 | SPRES | B'0 | R/W | Soft Power On Reset Soft Power On Reset is asserted. This bit is initialized on Power On Reset. 0: Ignored 1: Soft Power On Reset is asserted. Note: Soft Power On Reset is asserted when 1 is written to this bit. When Power On Reset process is completed, this bit gets cleared (The range of initialization is same with Soft Power On Reset by WDT) |
| 14 to 0 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |

17.2.16 SYS Reset Flag Register (SRSTFR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SRSTFR is a 32bit readable/writable register. This register can be changed only in the secure mode. Write access is performed only to clear the flag bits. Access this register from System Core (Cortex-A57 or Cortex-A53 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0], Cortex-A53 [RZ/G2E], Cortex-A57 [RZ/G2N]). This register sets the reset flags when Soft Power On Reset (Power On Reset other than PRESET#) is detected. By reading this register at the time of reboot etc., it is known that which kind of reset was detected.

This register is initialized only by PRESET#.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|--------|--------|--------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | RCPRES | RCSWDT | RCRWDT | RPF |
| | | | | | | | | | | | | | S | T | DT | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R(W)* | R(W)* | R(W)* | R(W)* |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-------|--|
| 31 to 4 | — | All 0 | R | Reserved When read, returns 0. The write value should always be 0. |
| 3 | RCPRES | B'0 | R(W)* | Soft Power On Reset Factor Bit 0: SRESCR.SPRES is not Soft Power On Reset factor. 1: SRESCR.SPRES is Soft Power On Reset factor. |
| 2 | RCSWDT | B'0 | R(W)* | Soft Power On Reset Factor Bit 0: System-WDT is not Soft Power On Reset factor. 1: System-WDT is Soft Power On Reset factor. |
| 1 | RCRWDT | B'0 | R(W)* | Soft Power On Reset Factor Bit 0: RWDT is not Soft Power On Reset factor. 1: RWDT is Soft Power On Reset factor. |
| 0 | RPF | B'0 | R(W)* | Soft Power On Reset detection bit 0: Soft Power On Reset is not detected. 1: Soft Power On Reset is detected. |

Note: * Write 0 to clear the flags.

17.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

17.3.1 Power-On Reset by PRESET# Pin

For the power-on oscillation settling time (t_{OSC}), see section 73, Electrical Characteristics.

After the state on the PRESET# pin input is changed from a low level to high level, the internal reset state continues until the flash ROM reset time (longer than $50 \mu s$), the flash ROM holding time (longer than $5 \mu s$) and the reset holding time elapse.

(1) Sequence for Turning on the Power

When the power is turned on, ensure that a low level is input to the PRESET# pin. A low level input is also needed on the TRST# pin.

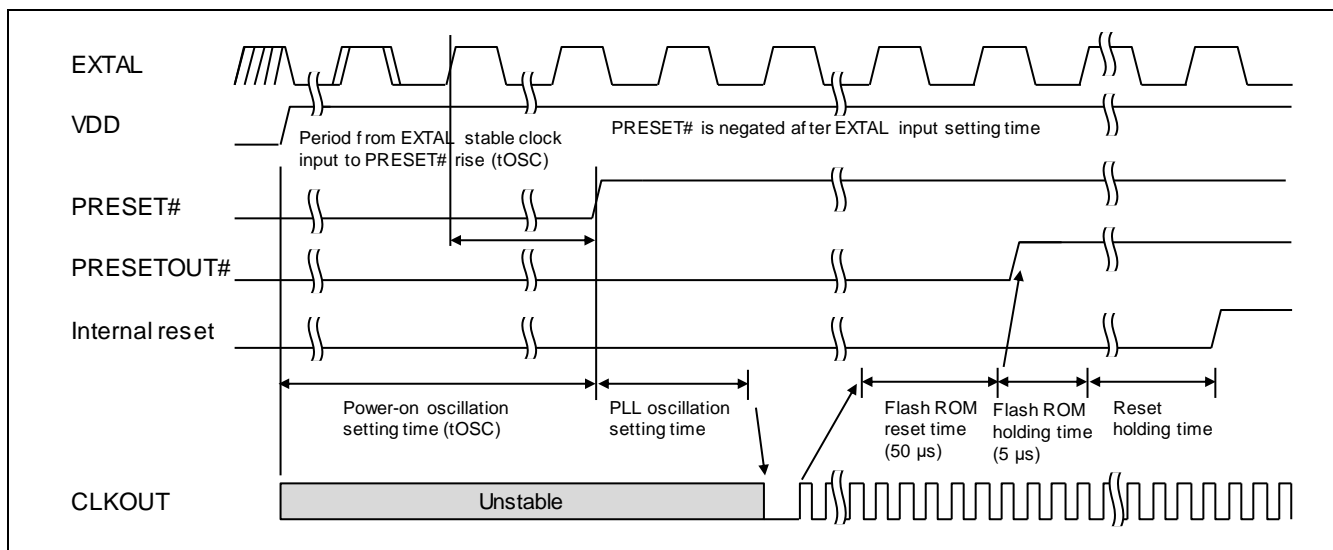


Figure 17.4 Free-Running Mode (MD0 = L)

17.3.2 WDT reset

- WDT resets can be issued by the RWDT and System WDT modules.
- WDTRSTCR controls the permission or prohibition of WDT resets.
- The reset (RST) module is not initialized by a WDT reset.

18. Interrupt Controller (INTC)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

18.1 Overview

This module handles system domain interrupt events.

18.1.1 Features

- INTC-AP handles system domain interrupt events, refer to section 19

18.1.2 Block Diagram

Figure 18.1 shows block diagram of connection between INTC and related modules. INTC module consists of synchronization.

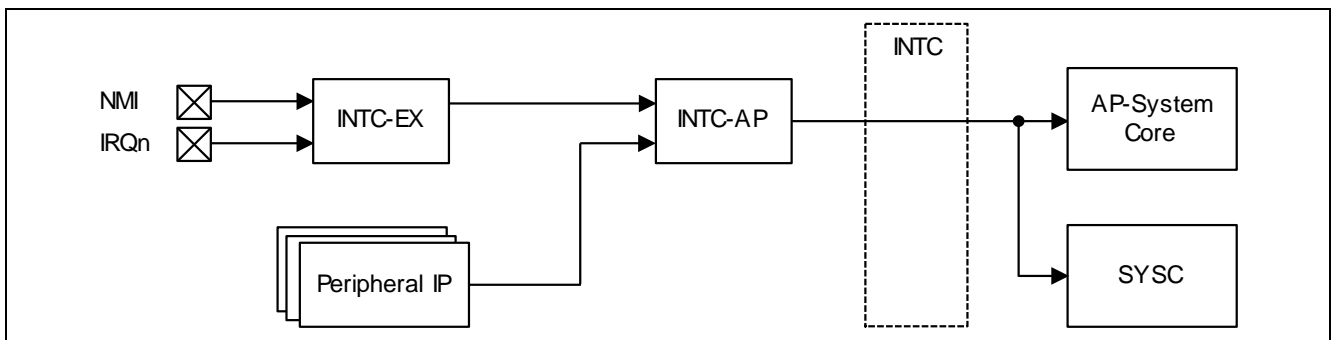


Figure 18.1 Block diagram of connection between INTC and related modules

18.1.3 External Pins

Refer to section 20.

18.1.4 Connected Module

Table 18.1 Connected Module

| Module name | Connected module name | Function of connected module |
|--------------------|------------------------------|--|
| INTC | AP-System Core | Application CPU core block |
| | INTC-EX | Provide interrupt from external device |
| | SYSC | Control the power supply to CPUs |

19. Interrupt Controller (INTC-AP)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

19.1 Overview

This module handles system domain interrupt events. This module is integrated Arm CoreLink GIC-400(r0p0) which is compliant Arm GICv2.

19.1.1 Features

- Interrupt controller for AP-System core
- 32 programmable priority levels
- Arm-IRQ interrupt generation
- INTC-AP handles peripheral interrupt

19.1.2 Block Diagram

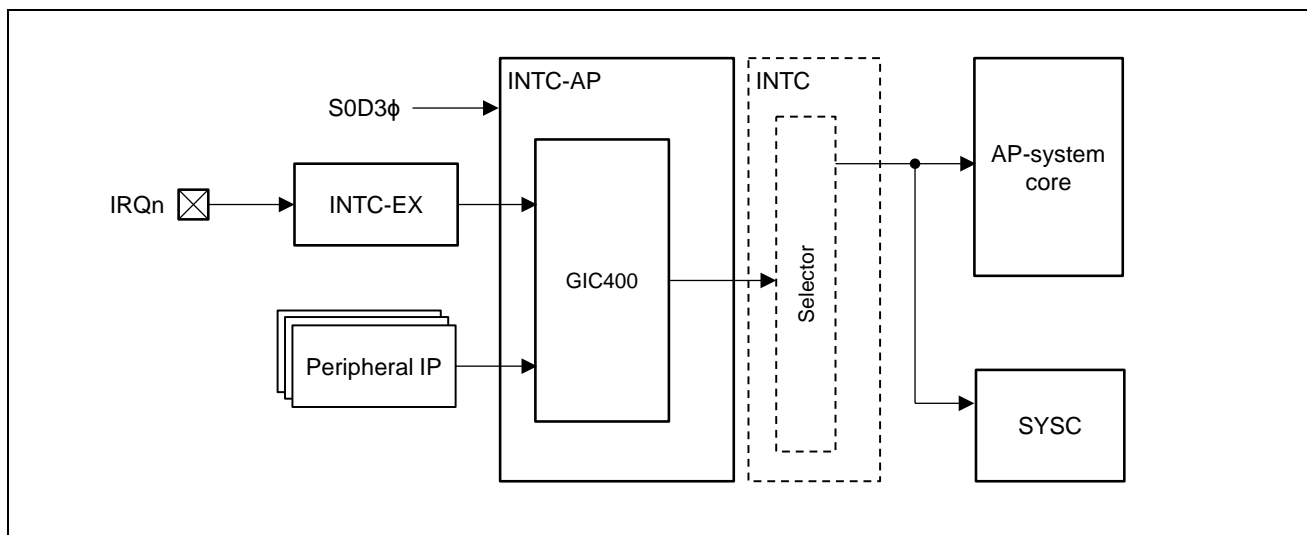


Figure 19.1 Block diagram of INTC-AP

19.1.3 External Pins

Refer to section 20.

19.1.4 Register Configuration

Refer to Arm Cortex-A57/Cortex-A53 Technical Reference Manual and Arm CoreLink GIC-400(r0p0) Generic Interrupt Controller Technical Reference Manual.

19.1.5 Connected Module

Table 19.1 Connected Module

| Module name | Connected module name | Function of connected module |
|--------------------|------------------------------|--|
| INTC-AP | AP-System Core | Application CPU core block |
| | INTC-EX | Provide interrupt from external device |
| | SYSC | Control the power supply to CPUs |

19.2 Register Description

This section describes the registers of INTC-AP. If you want more detail information, refer to Arm Cortex-A57/Cortex-A53 Technical Reference Manual and Arm CoreLink GIC-400(r0p0) Generic Interrupt Controller Technical Reference Manual.

Table 19.2 Register Description

| GIC-400(r0p0) functional block | Address range | Second Generation RZ/G Series Products | | | |
|--|--|--|----------------------------|--------|--------|
| | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| INTC-AP Distributor | H'F101_0000 – H'F101_0FFF | √ | √ | √ | √ |
| INTC-AP CPU Interfaces | H'F102_0000 – H'F102_0FFF*1 H'F103_0000 – H'F103_0FFF*2 | √ | √ | √ | √ |
| INTC-AP Virtual Interface control block, for the processor that is performing the access | H'F104_0000 – H'F104_0FFF | √ | √ | √ | √ |
| INTC-AP Virtual interface control block, for the processor selected by address bits[11:9] | H'F105_0000 – H'F105_01FF Alias for Processor 0 H'F105_0200 – H'F105_03FF Alias for Processor 1 ... H'F105_0E00 – H'F105_0FFF Alias for Processor 7 | √ | √ | √ | √ |
| INTC-AP Virtual CPU interfaces | H'F106_0000 – H'F106_0FFF H'F107_0000 – H'F107_0FFF | √ | √ | √ | √ |

Notes: CPU Interface register base address is H'F102 0000.

1. Offset H'0000 to H'0FFF are assigned.
2. Offset H'1000 to H'1FFF are assigned.

19.3 Operation

RZ/G2H

RZ/G2M V1.3

RZ/G2M V3.0

RZ/G2N

RZ/G2E

INTC-AP is based on Arm Generic Interrupt Controller. For more information, see Arm Generic Interrupt Controller Architecture Specification and section 7, AP-System Core.

19.3.1 INTC-AP Register Configuration and Function Description

480 SPI interrupts are integrated into Interrupt ID [511:32].

The INTC-AP supports 8 CPU cores which are mapped as shown below.

Table 19.3 Port mapping for INTC-AP

| | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|----------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|
| CPU0 interface | AP-System core (Cortex-A57) CPU0 | AP-System core (Cortex-A57) CPU0 | AP-System core (Cortex-A57) CPU0 | AP-System core (Cortex-A53) CPU0 |
| CPU1 interface | AP-System core (Cortex-A57) CPU1 | AP-System core (Cortex-A57) CPU1 | AP-System core (Cortex-A57) CPU1 | AP-System core (Cortex-A53) CPU1 |
| CPU2 interface | AP-System core (Cortex-A57) CPU2 | Reserved | Reserved | Reserved |
| CPU3 interface | AP-System core (Cortex-A57) CPU3 | Reserved | Reserved | Reserved |
| CPU4 interface | AP-System core (Cortex-A53) CPU0 | AP-System core (Cortex-A53) CPU0 | Reserved | Reserved |
| CPU5 interface | AP-System core (Cortex-A53) CPU1 | AP-System core (Cortex-A53) CPU1 | Reserved | Reserved |
| CPU6 interface | AP-System core (Cortex-A53) CPU2 | AP-System core (Cortex-A53) CPU2 | Reserved | Reserved |
| CPU7 interface | AP-System core (Cortex-A53) CPU3 | AP-System core (Cortex-A53) CPU3 | Reserved | Reserved |

For more information of these registers, see Arm Cortex-A57/Cortex-A53 Technical Reference Manual and Arm CoreLink GIC-400(r0p0) Generic Interrupt Controller Technical Reference Manual.

19.3.2 Interrupts Mapping

Table 19.4 shows the AP-system core interrupt controller (INTC-AP) interrupt mapping.

Note: SGI is Software Generated Interrupt, PPI is Private Peripheral Interrupt and SPI is Shared Peripheral Interrupt. All SPI interrupts should be level-sensitive configuration. If you want more detail information, refer to Arm Cortex-A57/Cortex-A53 Technical Reference Manual and Arm CoreLink GIC-400(r0p0) Generic Interrupt Controller Technical Reference Manual.

Table 19.4 INTC-AP Interrupt Mapping

| Interrupt ID | SGI, PPI or SPI No | Source | | | |
|--------------|--------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|
| | | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 0 | SGI 0 | SGI ID0 | SGI ID0 | SGI ID0 | SGI ID0 |
| 1 | SGI 1 | SGI ID1 | SGI ID1 | SGI ID1 | SGI ID1 |
| 2 | SGI 2 | SGI ID2 | SGI ID2 | SGI ID2 | SGI ID2 |
| 3 | SGI 3 | SGI ID3 | SGI ID3 | SGI ID3 | SGI ID3 |
| 4 | SGI 4 | SGI ID4 | SGI ID4 | SGI ID4 | SGI ID4 |
| 5 | SGI 5 | SGI ID5 | SGI ID5 | SGI ID5 | SGI ID5 |
| 6 | SGI 6 | SGI ID6 | SGI ID6 | SGI ID6 | SGI ID6 |
| 7 | SGI 7 | SGI ID7 | SGI ID7 | SGI ID7 | SGI ID7 |
| 8 | SGI 8 | SGI ID8 | SGI ID8 | SGI ID8 | SGI ID8 |
| 9 | SGI 9 | SGI ID9 | SGI ID9 | SGI ID9 | SGI ID9 |
| 10 | SGI 10 | SGI ID10 | SGI ID10 | SGI ID10 | SGI ID10 |
| 11 | SGI 11 | SGI ID11 | SGI ID11 | SGI ID11 | SGI ID11 |
| 12 | SGI 12 | SGI ID12 | SGI ID12 | SGI ID12 | SGI ID12 |
| 13 | SGI 13 | SGI ID13 | SGI ID13 | SGI ID13 | SGI ID13 |
| 14 | SGI 14 | SGI ID14 | SGI ID14 | SGI ID14 | SGI ID14 |
| 15 | SGI 15 | SGI ID15 | SGI ID15 | SGI ID15 | SGI ID15 |
| 16 | PPI 0 | Reserved | Reserved | Reserved | Reserved |
| ... | ... | ... | ... | ... | ... |
| 24 | PPI 8 | Reserved | Reserved | Reserved | Reserved |
| 25 | PPI 9 | Virtual maintenance interrupt | Virtual maintenance interrupt | Virtual maintenance interrupt | Virtual maintenance interrupt |
| 26 | PPI 10 | Hypervisor timer | Hypervisor timer | Hypervisor timer | Hypervisor timer |
| 27 | PPI 11 | Virtual timer | Virtual timer | Virtual timer | Virtual timer |
| 28 | PPI 12 | Legacy FIQ signal | Legacy FIQ signal | Legacy FIQ signal | Legacy FIQ signal |
| 29 | PPI 13 | Secure physical timer | Secure physical timer | Secure physical timer | Secure physical timer |
| 30 | PPI 14 | Non-secure physical timer | Non-secure physical timer | Non-secure physical timer | Non-secure physical timer |
| 31 | PPI 15 | Reserved | Reserved | Reserved | Reserved |
| 32 | SPI 0 | IRQ0 | IRQ0 | IRQ0 | IRQ0 |
| 33 | SPI 1 | IRQ1 | IRQ1 | IRQ1 | IRQ1 |
| 34 | SPI 2 | IRQ2 | IRQ2 | IRQ2 | IRQ2 |
| 35 | SPI 3 | IRQ3 | IRQ3 | IRQ3 | IRQ3 |
| 36 | SPI 4 | GPIO.ch0 | GPIO.ch0 | GPIO.ch0 | GPIO.ch0 |
| 37 | SPI 5 | GPIO.ch1 | GPIO.ch1 | GPIO.ch1 | GPIO.ch1 |
| 38 | SPI 6 | GPIO.ch2 | GPIO.ch2 | GPIO.ch2 | GPIO.ch2 |

| Interrupt ID | SGL, PPI or SPI No | Source | | | |
|--------------|--------------------|--------------------|-----------------------------|--------------------|--------------------|
| | | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 39 | SPI 7 | GPIO.ch3 | GPIO.ch3 | GPIO.ch3 | GPIO.ch3 |
| 40 | SPI 8 | GPIO.ch4 | GPIO.ch4 | GPIO.ch4 | GPIO.ch4 |
| 41 | SPI 9 | GPIO.ch5 | GPIO.ch5 | GPIO.ch5 | GPIO.ch5 |
| 42 | SPI 10 | GPIO.ch6 | GPIO.ch6 | GPIO.ch6 | GPIO.ch6 |
| 43 | SPI 11 | GPIO.ch7 | GPIO.ch7 | GPIO.ch7 | Reserved |
| 44 | SPI 12 | Reserved | Reserved | Reserved | Reserved |
| 45 | SPI 13 | Reserved | Reserved | Reserved | Reserved |
| 46 | SPI 14 | Reserved | Reserved | Reserved | Reserved |
| 47 | SPI 15 | Reserved | Reserved | Reserved | Reserved |
| 48 | SPI 16 | SCIF.ch4 | SCIF.ch4 | SCIF.ch4 | SCIF.ch4 |
| 49 | SPI 17 | SCIF.ch5 | SCIF.ch5 | SCIF.ch5 | SCIF.ch5 |
| 50 | SPI 18 | IRQ4 | IRQ4 | IRQ4 | IRQ4 |
| 51 | SPI 19 | I2C.ch4 | I2C.ch4 | I2C.ch4 | I2C.ch4 |
| 52 | SPI 20 | I2C.ch5 | I2C.ch5 | I2C.ch5 | I2C.ch5 |
| 53 | SPI 21 | I2C.ch6 | I2C.ch6 | I2C.ch6 | I2C.ch6 |
| 54 | SPI 22 | Reserved | Reserved | Reserved | Reserved |
| 55 | SPI 23 | SCIF.ch3 | SCIF.ch3 | SCIF.ch3 | SCIF.ch3 |
| 56 | SPI 24 | Reserved | Reserved | Reserved | Reserved |
| 57 | SPI 25 | Reserved | Reserved | Reserved | Reserved |
| 58 | SPI 26 | Reserved | Reserved | Reserved | Reserved |
| 59 | SPI 27 | Reserved | Reserved | Reserved | Reserved |
| 60 | SPI 28 | Reserved | Reserved | Reserved | Reserved |
| 61 | SPI 29 | CAN-FD channel | CAN-FD channel | CAN-FD channel | CAN-FD channel |
| 62 | SPI 30 | CAN-FD global | CAN-FD global | CAN-FD global | CAN-FD global |
| 63 | SPI 31 | Reserved | Reserved | Reserved | Reserved |
| 64 | SPI 32 | Reserved | Reserved | Reserved | Reserved |
| 65 | SPI 33 | Reserved | Reserved | Reserved | Reserved |
| 66 | SPI 34 | Reserved | Reserved | Reserved | Reserved |
| 67 | SPI 35 | Reserved | Reserved | Reserved | Reserved |
| 68 | SPI 36 | Reserved | Reserved | Reserved | I2C.ch7 |
| 69 | SPI 37 | EHCI/OHCI OTG.ch3 | Reserved | Reserved | Reserved |
| 70 | SPI 38 | RPC-IF | RPC-IF | RPC-IF | RPC-IF |
| 71 | SPI 39 | EthernetAVB-IF.ch0 | EthernetAVB-IF.ch0 | EthernetAVB-IF.ch0 | EthernetAVB-IF.ch0 |
| 72 | SPI 40 | EthernetAVB-IF.ch1 | EthernetAVB-IF.ch1 | EthernetAVB-IF.ch1 | EthernetAVB-IF.ch1 |
| 73 | SPI 41 | EthernetAVB-IF.ch2 | EthernetAVB-IF.ch2 | EthernetAVB-IF.ch2 | EthernetAVB-IF.ch2 |
| 74 | SPI 42 | EthernetAVB-IF.ch3 | EthernetAVB-IF.ch3 | EthernetAVB-IF.ch3 | EthernetAVB-IF.ch3 |
| 75 | SPI 43 | EthernetAVB-IF.ch4 | EthernetAVB-IF.ch4 | EthernetAVB-IF.ch4 | EthernetAVB-IF.ch4 |
| 76 | SPI 44 | EthernetAVB-IF.ch5 | EthernetAVB-IF.ch5 | EthernetAVB-IF.ch5 | EthernetAVB-IF.ch5 |
| 77 | SPI 45 | EthernetAVB-IF.ch6 | EthernetAVB-IF.ch6 | EthernetAVB-IF.ch6 | EthernetAVB-IF.ch6 |
| 78 | SPI 46 | EthernetAVB-IF.ch7 | EthernetAVB-IF.ch7 | EthernetAVB-IF.ch7 | EthernetAVB-IF.ch7 |
| 79 | SPI 47 | EthernetAVB-IF.ch8 | EthernetAVB-IF.ch8 | EthernetAVB-IF.ch8 | EthernetAVB-IF.ch8 |
| 80 | SPI 48 | EthernetAVB-IF.ch9 | EthernetAVB-IF.ch9 | EthernetAVB-IF.ch9 | EthernetAVB-IF.ch9 |

| Interrupt ID | SGI, PPI or SPI No | Source | | | |
|--------------|--------------------|--|--|--|---------------------|
| | | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 81 | SPI 49 | EthernetAVB-IF.ch10 | EthernetAVB-IF.ch10 | EthernetAVB-IF.ch10 | EthernetAVB-IF.ch10 |
| 82 | SPI 50 | EthernetAVB-IF.ch11 | EthernetAVB-IF.ch11 | EthernetAVB-IF.ch11 | EthernetAVB-IF.ch11 |
| 83 | SPI 51 | EthernetAVB-IF.ch12 | EthernetAVB-IF.ch12 | EthernetAVB-IF.ch12 | EthernetAVB-IF.ch12 |
| 84 | SPI 52 | EthernetAVB-IF.ch13 | EthernetAVB-IF.ch13 | EthernetAVB-IF.ch13 | EthernetAVB-IF.ch13 |
| 85 | SPI 53 | EthernetAVB-IF.ch14 | EthernetAVB-IF.ch14 | EthernetAVB-IF.ch14 | EthernetAVB-IF.ch14 |
| 86 | SPI 54 | EthernetAVB-IF.ch15 | EthernetAVB-IF.ch15 | EthernetAVB-IF.ch15 | EthernetAVB-IF.ch15 |
| 87 | SPI 55 | EthernetAVB-IF.ch16 | EthernetAVB-IF.ch16 | EthernetAVB-IF.ch16 | EthernetAVB-IF.ch16 |
| 88 | SPI 56 | EthernetAVB-IF.ch17 | EthernetAVB-IF.ch17 | EthernetAVB-IF.ch17 | EthernetAVB-IF.ch17 |
| 89 | SPI 57 | EthernetAVB-IF.ch18 | EthernetAVB-IF.ch18 | EthernetAVB-IF.ch18 | EthernetAVB-IF.ch18 |
| 90 | SPI 58 | EthernetAVB-IF.ch19 | EthernetAVB-IF.ch19 | EthernetAVB-IF.ch19 | EthernetAVB-IF.ch19 |
| 91 | SPI 59 | EthernetAVB-IF.ch20 | EthernetAVB-IF.ch20 | EthernetAVB-IF.ch20 | EthernetAVB-IF.ch20 |
| 92 | SPI 60 | EthernetAVB-IF.ch21 | EthernetAVB-IF.ch21 | EthernetAVB-IF.ch21 | EthernetAVB-IF.ch21 |
| 93 | SPI 61 | EthernetAVB-IF.ch22 | EthernetAVB-IF.ch22 | EthernetAVB-IF.ch22 | EthernetAVB-IF.ch22 |
| 94 | SPI 62 | EthernetAVB-IF.ch23 | EthernetAVB-IF.ch23 | EthernetAVB-IF.ch23 | EthernetAVB-IF.ch23 |
| 95 | SPI 63 | EthernetAVB-IF.ch24 | EthernetAVB-IF.ch24 | EthernetAVB-IF.ch24 | EthernetAVB-IF.ch24 |
| 96 | SPI 64 | Reserved | Reserved | Reserved | Reserved |
| 97 | SPI 65 | Reserved | Reserved | Reserved | Reserved |
| 98 | SPI 66 | Reserved | Reserved | Reserved | Reserved |
| 99 | SPI 67 | Thermal Sensor.ch0 | Thermal Sensor.ch0 | Thermal Sensor.ch0 | Thermal Sensor.ch0 |
| 100 | SPI 68 | Thermal Sensor.ch1 | Thermal Sensor.ch1 | Thermal Sensor.ch1 | Thermal Sensor.ch1 |
| 101 | SPI 69 | Thermal Sensor.ch2 | Thermal Sensor.ch2 | Thermal Sensor.ch2 | Thermal Sensor.ch2 |
| 102 | SPI 70 | Reserved | Reserved | Reserved | Reserved |
| 103 | SPI 71 | Reserved | Reserved | Reserved | Reserved |
| 104 | SPI 72 | AP-System Core. Cortex-A57 core0 pmu | AP-System Core. Cortex-A57 core0 pmu | AP-System Core. Cortex-A57 core0 pmu | Reserved |
| 105 | SPI 73 | AP-System Core. Cortex-A57 core1 pmu | AP-System Core. Cortex-A57 core1 pmu | AP-System Core. Cortex-A57 core1 pmu | Reserved |
| 106 | SPI 74 | AP-System Core. Cortex-A57 core2 pmu | Reserved | Reserved | Reserved |
| 107 | SPI 75 | AP-System Core. Cortex-A57 core3 pmu | Reserved | Reserved | Reserved |
| 108 | SPI 76 | AP-System Core. Cortex-A57 core0 cti | AP-System Core. Cortex-A57 core0 cti | AP-System Core. Cortex-A57 core0 cti | Reserved |
| 109 | SPI 77 | AP-System Core. Cortex-A57 core1 cti | AP-System Core. Cortex-A57 core1 cti | AP-System Core. Cortex-A57 core1 cti | Reserved |
| 110 | SPI 78 | AP-System Core. Cortex-A57 core2 cti | Reserved | Reserved | Reserved |
| 111 | SPI 79 | AP-System Core. Cortex-A57 core3 cti | Reserved | Reserved | Reserved |
| 112 | SPI 80 | AP-System Core. Cortex-A57 core0 comm | AP-System Core. Cortex-A57 core0 comm | AP-System Core. Cortex-A57 core0 comm | Reserved |
| 113 | SPI 81 | AP-System Core. Cortex-A57 core1 comm | AP-System Core. Cortex-A57 core1 comm | AP-System Core. Cortex-A57 core1 comm | Reserved |
| 114 | SPI 82 | AP-System Core. Cortex-A57 core2 comm | Reserved | Reserved | Reserved |

| Interrupt ID | SGI, PPI or SPI No | Source | | | |
|--------------|--------------------|--|--|-------------------------------------|--|
| | | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 115 | SPI 83 | AP-System Core. Cortex-A57 core3 comm | Reserved | Reserved | Reserved |
| 116 | SPI 84 | AP-System Core. Cortex-A53 core0 pmu | AP-System Core. Cortex-A53 core0 pmu | Reserved | AP-System Core. Cortex-A53 core0 pmu |
| 117 | SPI 85 | AP-System Core. Cortex-A53 core1 pmu | AP-System Core. Cortex-A53 core1 pmu | Reserved | AP-System Core. Cortex-A53 core1 pmu |
| 118 | SPI 86 | AP-System Core. Cortex-A53 core2 pmu | AP-System Core. Cortex-A53 core2 pmu | Reserved | Reserved |
| 119 | SPI 87 | AP-System Core. Cortex-A53 core3 pmu | AP-System Core. Cortex-A53 core3 pmu | Reserved | Reserved |
| 120 | SPI 88 | AP-System Core. Cortex-A53 core0 cti | AP-System Core. Cortex-A53 core0 cti | Reserved | AP-System Core. Cortex-A53 core0 cti |
| 121 | SPI 89 | AP-System Core. Cortex-A53 core1 cti | AP-System Core. Cortex-A53 core1 cti | Reserved | AP-System Core. Cortex-A53 core1 cti |
| 122 | SPI 90 | AP-System Core. Cortex-A53 core2 cti | AP-System Core. Cortex-A53 core2 cti | Reserved | Reserved |
| 123 | SPI 91 | AP-System Core. Cortex-A53 core3 cti | AP-System Core. Cortex-A53 core3 cti | Reserved | Reserved |
| 124 | SPI 92 | AP-System Core. Cortex-A53 core0 comm | AP-System Core. Cortex-A53 core0 comm | Reserved | AP-System Core. Cortex-A53 core0 comm |
| 125 | SPI 93 | AP-System Core. Cortex-A53 core1 comm | AP-System Core. Cortex-A53 core1 comm | Reserved | AP-System Core. Cortex-A53 core1 comm |
| 126 | SPI 94 | AP-System Core. Cortex-A53 core2 comm | AP-System Core. Cortex-A53 core2 comm | Reserved | Reserved |
| 127 | SPI 95 | AP-System Core. Cortex-A53 core3 comm | AP-System Core. Cortex-A53 core3 comm | Reserved | Reserved |
| 128 | SPI 96 | LBSC | LBSC | LBSC | LBSC |
| 129 | SPI 97 | Reserved | Reserved | Reserved | Reserved |
| 130 | SPI 98 | Reserved | Reserved | Reserved | Reserved |
| 131 | SPI 99 | Reserved | Reserved | Reserved | Reserved |
| 132 | SPI 100 | Reserved | NAND Flash Controller | NAND Flash Controller | NAND Flash Controller |
| 133 | SPI 101 | USB3.0 Host Controller. ch0 bc | USB3.0 Host Controller. ch0 bc | USB3.0 Host Controller. ch0 bc | USB3.0 Host Controller. ch0 bc |
| 134 | SPI 102 | USB3.0 Host Controller. ch0 host | USB3.0 Host Controller. ch0 host | USB3.0 Host Controller. ch0 host | USB3.0 Host Controller. ch0 host |
| 135 | SPI 103 | USB3.0 Host Controller. ch0 otg | USB3.0 Host Controller. ch0 otg | USB3.0 Host Controller. ch0 otg | USB3.0 Host Controller. ch0 otg |
| 136 | SPI 104 | USB3.0 Host Controller. ch0 peri | USB3.0 Host Controller. ch0 peri | USB3.0 Host Controller. ch0 peri | USB3.0 Host Controller. ch0 peri |
| 137 | SPI 105 | Serial-ATA Gen3 | Reserved | Serial-ATA Gen3 | Reserved |
| 138 | SPI 106 | AP-System Core.CCI | AP-System Core.CCI | Reserved | Reserved |
| 139 | SPI 107 | EHCI/OHCI OTG.ch0 | EHCI/OHCI OTG.ch0 | EHCI/OHCI OTG.ch0 | EHCI/OHCI OTG.ch0 |
| 140 | SPI 108 | EHCI HOST0 | EHCI HOST0 | EHCI HOST0 | EHCI HOST0 |
| 141 | SPI 109 | USB-DMAC. ch0 | USB-DMAC. ch0 | USB-DMAC. ch0 | USB-DMAC. ch0 |
| 142 | SPI 110 | USB-DMAC. ch1 | USB-DMAC. ch1 | USB-DMAC. ch1 | USB-DMAC. ch1 |
| 143 | SPI 111 | USB.DDM | USB.DDM | USB.DDM | USB.DDM |

| Interrupt ID | SGL, PPI or SPI No | Source | | | |
|--------------|--------------------|-----------------------|-----------------------------|-----------------------|-----------------------|
| | | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 144 | SPI 112 | EHCI. HOST1 | EHCI. HOST1 | EHCI. HOST1 | Reserved |
| 145 | SPI 113 | Reserved | Reserved | Reserved | Reserved |
| 146 | SPI 114 | Reserved | Reserved | Reserved | Reserved |
| 147 | SPI 115 | Reserved | Reserved | Reserved | Reserved |
| 148 | SPI 116 | PCIE.ch0 | PCIE.ch0 | PCIE.ch0 | PCIE.ch0 |
| 149 | SPI 117 | PCIE.ch0 dma | PCIE.ch0 dma | PCIE.ch0 dma | PCIE.ch0 dma |
| 150 | SPI 118 | PCIE.ch0 err | PCIE.ch0 err | PCIE.ch0 err | PCIE.ch0 err |
| 151 | SPI 119 | 3DGE | 3DGE | 3DGE | 3DGE |
| 152 | SPI 120 | CMT1.ch0 | CMT1.ch0 | CMT1.ch0 | CMT1.ch0 |
| 153 | SPI 121 | CMT1.ch1 | CMT1.ch1 | CMT1.ch1 | CMT1.ch1 |
| 154 | SPI 122 | CMT1.ch2 | CMT1.ch2 | CMT1.ch2 | CMT1.ch2 |
| 155 | SPI 123 | CMT1.ch3 | CMT1.ch3 | CMT1.ch3 | CMT1.ch3 |
| 156 | SPI 124 | CMT1.ch4 | CMT1.ch4 | CMT1.ch4 | CMT1.ch4 |
| 157 | SPI 125 | CMT1.ch5 | CMT1.ch5 | CMT1.ch5 | CMT1.ch5 |
| 158 | SPI 126 | CMT1.ch6 | CMT1.ch6 | CMT1.ch6 | CMT1.ch6 |
| 159 | SPI 127 | CMT1.ch7 | CMT1.ch7 | CMT1.ch7 | CMT1.ch7 |
| 160 | SPI 128 | TMU.TUNI3 | TMU.TUNI3 | TMU.TUNI3 | TMU.TUNI3 |
| 161 | SPI 129 | TMU.TUNI4 | TMU.TUNI4 | TMU.TUNI4 | TMU.TUNI4 |
| 162 | SPI 130 | TMU.TUNI5 | TMU.TUNI5 | TMU.TUNI5 | TMU.TUNI5 |
| 163 | SPI 131 | TMU.TUNI9 | TMU.TUNI9 | TMU.TUNI9 | TMU.TUNI9 |
| 164 | SPI 132 | TMU.TUNI10 | TMU.TUNI10 | TMU.TUNI10 | TMU.TUNI10 |
| 165 | SPI 133 | TMU.TUNI11 | TMU.TUNI11 | TMU.TUNI11 | TMU.TUNI11 |
| 166 | SPI 134 | System timer | System timer | System timer | System timer |
| 167 | SPI 135 | TPU | TPU | TPU | TPU |
| 168 | SPI 136 | TMU.TUNI0 | TMU.TUNI0 | TMU.TUNI 0 | TMU.TUNI 0 |
| 169 | SPI 137 | TMU.TUNI1 | TMU.TUNI1 | TMU.TUNI1 | TMU.TUNI1 |
| 170 | SPI 138 | TMU.TUNI2 | TMU.TUNI2 | TMU.TUNI2 | TMU.TUNI2 |
| 171 | SPI 139 | System up-time clock | System up-time clock | System up-time clock | System up-time clock |
| 172 | SPI 140 | RCLK Watchdog Timer | RCLK Watchdog Timer | RCLK Watchdog Timer | RCLK Watchdog Timer |
| 173 | SPI 141 | System Watchdog Timer | System Watchdog Timer | System Watchdog Timer | System Watchdog Timer |
| 174 | SPI 142 | CMT0.ch0 | CMT0.ch0 | CMT0.ch0 | CMT0.ch0 |
| 175 | SPI 143 | CMT0.ch1 | CMT0.ch1 | CMT0.ch1 | CMT0.ch1 |
| 176 | SPI 144 | HSCIF.ch2 | HSCIF.ch2 | HSCIF.ch2 | HSCIF.ch2 |
| 177 | SPI 145 | HSCIF.ch3 | HSCIF.ch3 | HSCIF.ch3 | HSCIF.ch3 |
| 178 | SPI 146 | HSCIF.ch4 | HSCIF.ch4 | HSCIF.ch4 | HSCIF.ch4 |
| 179 | SPI 147 | TMU.TICP15 | TMU.TICP15 | TMU.TICP15 | TMU.TICP15 |
| 180 | SPI 148 | PCIE.ch1 | PCIE.ch1 | PCIE.ch1 | Reserved |
| 181 | SPI 149 | PCIE.ch1 dma | PCIE.ch1 dma | PCIE.ch1 dma | Reserved |
| 182 | SPI 150 | PCIE.ch1 err | PCIE.ch1 err | PCIE.ch1 err | Reserved |
| 183 | SPI 151 | Reserved | Reserved | Reserved | Reserved |
| 184 | SPI 152 | SCIF.ch0 | SCIF.ch0 | SCIF.ch0 | SCIF.ch0 |

| Interrupt ID | SGL, PPI or SPI No | Source | | | |
|--------------|--------------------|--------------|-----------------------------|--------------|--------------|
| | | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 185 | SPI 153 | SCIF.ch1 | SCIF.ch1 | SCIF.ch1 | SCIF.ch1 |
| 186 | SPI 154 | HSCIF.ch0 | HSCIF.ch0 | HSCIF.ch0 | HSCIF.ch0 |
| 187 | SPI 155 | HSCIF.ch1 | HSCIF.ch1 | HSCIF.ch1 | HSCIF.ch1 |
| 188 | SPI 156 | MSIOF.ch0 | MSIOF.ch0 | MSIOF.ch0 | MSIOF.ch0 |
| 189 | SPI 157 | MSIOF.ch1 | MSIOF.ch1 | MSIOF.ch1 | MSIOF.ch1 |
| 190 | SPI 158 | MSIOF.ch2 | MSIOF.ch2 | MSIOF.ch2 | MSIOF.ch2 |
| 191 | SPI 159 | MSIOF.ch3 | MSIOF.ch3 | MSIOF.ch3 | MSIOF.ch3 |
| 192 | SPI 160 | Reserved | Reserved | Reserved | Reserved |
| 193 | SPI 161 | IRQ5 | IRQ5 | IRQ5 | IRQ5 |
| 194 | SPI 162 | Reserved | Reserved | Reserved | Reserved |
| 195 | SPI 163 | Reserved | Reserved | Reserved | Reserved |
| 196 | SPI 164 | SCIF.ch2 | SCIF.ch2 | SCIF.ch2 | SCIF.ch2 |
| 197 | SPI 165 | SDHI.ch0 | SDHI.ch0 | SDHI.ch0 | SDHI.ch0 |
| 198 | SPI 166 | SDHI.ch1 | SDHI.ch1 | SDHI.ch1 | SDHI.ch1 |
| 199 | SPI 167 | SDHI.ch2 | SDHI.ch2 | SDHI.ch2 | Reserved |
| 200 | SPI 168 | SDHI.ch3 | SDHI.ch3 | SDHI.ch3 | SDHI.ch3 |
| 201 | SPI 169 | Reserved | Reserved | Reserved | Reserved |
| 202 | SPI 170 | Reserved | Reserved | Reserved | Reserved |
| 203 | SPI 171 | VIN.ch7 | VIN.ch7 | VIN.ch7 | Reserved |
| 204 | SPI 172 | Reserved | Reserved | Reserved | Reserved |
| 205 | SPI 173 | IIC for PMIC | IIC for PMIC | IIC for PMIC | IIC for PMIC |
| 206 | SPI 174 | VIN.ch4 | VIN.ch4 | VIN.ch4 | VIN.ch4 |
| 207 | SPI 175 | VIN.ch5 | VIN.ch5 | VIN.ch5 | VIN.ch5 |
| 208 | SPI 176 | VIN.ch6 | VIN.ch6 | VIN.ch6 | Reserved |
| 209 | SPI 177 | Reserved | Reserved | Reserved | Reserved |
| 210 | SPI 178 | Reserved | Reserved | Reserved | Reserved |
| 211 | SPI 179 | Reserved | Reserved | Reserved | Reserved |
| 212 | SPI 180 | Reserved | Reserved | Reserved | Reserved |
| 213 | SPI 181 | Reserved | Reserved | Reserved | Reserved |
| 214 | SPI 182 | Reserved | Reserved | Reserved | Reserved |
| 215 | SPI 183 | Reserved | Reserved | Reserved | Reserved |
| 216 | SPI 184 | CSI2.ch0 | CSI2.ch0 | CSI2.ch0 | Reserved |
| 217 | SPI 185 | Reserved | Reserved | Reserved | Reserved |
| 218 | SPI 186 | CAN.ch0 | CAN.ch0 | CAN.ch0 | CAN.ch0 |
| 219 | SPI 187 | CAN.ch1 | CAN.ch1 | CAN.ch1 | CAN.ch1 |
| 220 | SPI 188 | VIN.ch0 | VIN.ch0 | VIN.ch0 | Reserved |
| 221 | SPI 189 | VIN.ch1 | VIN.ch1 | VIN.ch1 | Reserved |
| 222 | SPI 190 | VIN.ch2 | VIN.ch2 | VIN.ch2 | Reserved |
| 223 | SPI 191 | VIN.ch3 | VIN.ch3 | VIN.ch3 | Reserved |
| 224 | SPI 192 | Reserved | Reserved | Reserved | Reserved |
| 225 | SPI 193 | Reserved | Reserved | Reserved | Reserved |
| 226 | SPI 194 | Reserved | Reserved | Reserved | Reserved |

| Interrupt ID | SGL, PPI or SPI No | Source | | | |
|--------------|--------------------|----------------|-----------------------------|----------------|----------------|
| | | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 227 | SPI 195 | Reserved | Reserved | Reserved | Reserved |
| 228 | SPI 196 | IPMMU | IPMMU | IPMMU | IPMMU |
| 229 | SPI 197 | IPMMU sec | IPMMU sec | IPMMU sec | IPMMU sec |
| 230 | SPI 198 | Reserved | Reserved | Reserved | Reserved |
| 231 | SPI 199 | SYS-DMAC0 err | SYS-DMAC0 err | SYS-DMAC0 err | SYS-DMAC0 err |
| 232 | SPI 200 | SYS-DMAC0.ch0 | SYS-DMAC0.ch0 | SYS-DMAC0.ch0 | SYS-DMAC0.ch0 |
| 233 | SPI 201 | SYS-DMAC0.ch1 | SYS-DMAC0.ch1 | SYS-DMAC0.ch1 | SYS-DMAC0.ch1 |
| 234 | SPI 202 | SYS-DMAC0.ch2 | SYS-DMAC0.ch2 | SYS-DMAC0.ch2 | SYS-DMAC0.ch2 |
| 235 | SPI 203 | SYS-DMAC0.ch3 | SYS-DMAC0.ch3 | SYS-DMAC0.ch3 | SYS-DMAC0.ch3 |
| 236 | SPI 204 | SYS-DMAC0.ch4 | SYS-DMAC0.ch4 | SYS-DMAC0.ch4 | SYS-DMAC0.ch4 |
| 237 | SPI 205 | SYS-DMAC0.ch5 | SYS-DMAC0.ch5 | SYS-DMAC0.ch5 | SYS-DMAC0.ch5 |
| 238 | SPI 206 | SYS-DMAC0.ch6 | SYS-DMAC0.ch6 | SYS-DMAC0.ch6 | SYS-DMAC0.ch6 |
| 239 | SPI 207 | SYS-DMAC0.ch7 | SYS-DMAC0.ch7 | SYS-DMAC0.ch7 | SYS-DMAC0.ch7 |
| 240 | SPI 208 | SYS-DMAC0.ch8 | SYS-DMAC0.ch8 | SYS-DMAC0.ch8 | SYS-DMAC0.ch8 |
| 241 | SPI 209 | SYS-DMAC0.ch9 | SYS-DMAC0.ch9 | SYS-DMAC0.ch9 | SYS-DMAC0.ch9 |
| 242 | SPI 210 | SYS-DMAC0.ch10 | SYS-DMAC0.ch10 | SYS-DMAC0.ch10 | SYS-DMAC0.ch10 |
| 243 | SPI 211 | SYS-DMAC0.ch11 | SYS-DMAC0.ch11 | SYS-DMAC0.ch11 | SYS-DMAC0.ch11 |
| 244 | SPI 212 | SYS-DMAC0.ch12 | SYS-DMAC0.ch12 | SYS-DMAC0.ch12 | SYS-DMAC0.ch12 |
| 245 | SPI 213 | SYS-DMAC0.ch13 | SYS-DMAC0.ch13 | SYS-DMAC0.ch13 | SYS-DMAC0.ch13 |
| 246 | SPI 214 | SYS-DMAC0.ch14 | SYS-DMAC0.ch14 | SYS-DMAC0.ch14 | SYS-DMAC0.ch14 |
| 247 | SPI 215 | SYS-DMAC0.ch15 | SYS-DMAC0.ch15 | SYS-DMAC0.ch15 | SYS-DMAC0.ch15 |
| 248 | SPI 216 | SYS-DMAC1.ch0 | SYS-DMAC1.ch0 | SYS-DMAC1.ch0 | SYS-DMAC1.ch0 |
| 249 | SPI 217 | SYS-DMAC1.ch1 | SYS-DMAC1.ch1 | SYS-DMAC1.ch1 | SYS-DMAC1.ch1 |
| 250 | SPI 218 | SYS-DMAC1.ch2 | SYS-DMAC1.ch2 | SYS-DMAC1.ch2 | SYS-DMAC1.ch2 |
| 251 | SPI 219 | SYS-DMAC1.ch3 | SYS-DMAC1.ch3 | SYS-DMAC1.ch3 | SYS-DMAC1.ch3 |
| 252 | SPI 220 | SYS-DMAC1 err | SYS-DMAC1 err | SYS-DMAC1 err | SYS-DMAC1 err |
| 253 | SPI 221 | Reserved | Reserved | Reserved | Reserved |
| 254 | SPI 222 | Reserved | Reserved | Reserved | Reserved |
| 255 | SPI 223 | iVDP1C.cmint | iVDP1C.cmint | iVDP1C.cmint | iVDP1C.cmint |
| 256 | SPI 224 | Reserved | Reserved | Reserved | Reserved |
| 257 | SPI 225 | Reserved | Reserved | Reserved | Reserved |
| 258 | SPI 226 | Reserved | Reserved | Reserved | Reserved |
| 259 | SPI 227 | Reserved | Reserved | Reserved | Reserved |
| 260 | SPI 228 | Reserved | Reserved | Reserved | Reserved |
| 261 | SPI 229 | Reserved | Reserved | Reserved | Reserved |
| 262 | SPI 230 | Reserved | Reserved | Reserved | Reserved |
| 263 | SPI 231 | Reserved | Reserved | Reserved | Reserved |

| Interrupt ID | SGI, PPI or SPI No | Source | | | |
|--------------|--------------------|-----------------|-----------------------------|-----------------|-----------------|
| | | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 264 | SPI 232 | Reserved | Reserved | Reserved | Reserved |
| 265 | SPI 233 | Reserved | Reserved | Reserved | Reserved |
| 266 | SPI 234 | CPG.ch2 | CPG.ch2 | CPG.ch2 | CPG.ch2 |
| 267 | SPI 235 | Reserved | Reserved | Reserved | Reserved |
| 268 | SPI 236 | Reserved | Reserved | Reserved | Reserved |
| 269 | SPI 237 | Reserved | Reserved | Reserved | Reserved |
| 270 | SPI 238 | Reserved | Reserved | Reserved | Reserved |
| 271 | SPI 239 | Reserved | Reserved | Reserved | Reserved |
| 272 | SPI 240 | VCP4.vdpb.vint | VCP4.vdpb.vint | VCP4.vdpb.vint | VCP4.vdpb.vint |
| 273 | SPI 241 | VCP4.vdpb.cint | VCP4.vdpb.cint | VCP4.vdpb.cint | VCP4.vdpb.cint |
| 274 | SPI 242 | Reserved | Reserved | Reserved | Reserved |
| 275 | SPI 243 | Reserved | Reserved | Reserved | Reserved |
| 276 | SPI 244 | CPG.ch0 | CPG.ch0 | CPG.ch0 | CPG.ch0 |
| 277 | SPI 245 | CPG.ch1 | CPG.ch1 | CPG.ch1 | CPG.ch1 |
| 278 | SPI 246 | CSI2.ch2 | CSI2.ch2 | CSI2.ch2 | CSI2.ch2 |
| 279 | SPI 247 | Reserved | Reserved | Reserved | Reserved |
| 280 | SPI 248 | GPIO.ch0A | GPIO.ch0A | GPIO.ch0A | GPIO.ch0A |
| 281 | SPI 249 | GPIO.ch1A | GPIO.ch1A | GPIO.ch1A | GPIO.ch1A |
| 282 | SPI 250 | GPIO.ch2A | GPIO.ch2A | GPIO.ch2A | GPIO.ch2A |
| 283 | SPI 251 | GPIO.ch3A | GPIO.ch3A | GPIO.ch3A | GPIO.ch3A |
| 284 | SPI 252 | GPIO.ch4A | GPIO.ch4A | GPIO.ch4A | GPIO.ch4A |
| 285 | SPI 253 | GPIO.ch5A | GPIO.ch5A | GPIO.ch5A | GPIO.ch5A |
| 286 | SPI 254 | GPIO.ch6A | GPIO.ch6A | GPIO.ch6A | GPIO.ch6A |
| 287 | SPI 255 | GPIO.ch7A | GPIO.ch7A | GPIO.ch7A | Reserved |
| 288 | SPI 256 | DU.ch0 | DU.ch0 | DU.ch0 | DU.ch0 |
| 289 | SPI 257 | Reserved | Reserved | Reserved | Reserved |
| 290 | SPI 258 | Reserved | Reserved | Reserved | Reserved |
| 291 | SPI 259 | Reserved | Reserved | Reserved | Reserved |
| 292 | SPI 260 | VCP4.vcplf.vint | VCP4.vcplf.vint | VCP4.vcplf.vint | VCP4.vcplf.vint |
| 293 | SPI 261 | VCP4.vcplf.cint | VCP4.vcplf.cint | VCP4.vcplf.cint | VCP4.vcplf.cint |
| 294 | SPI 262 | FDP1.ch0 | FDP1.ch0 | FDP1.ch0 | FDP1.ch0 |
| 295 | SPI 263 | FDP1.ch1 | Reserved | Reserved | Reserved |
| 296 | SPI 264 | Reserved | Reserved | Reserved | Reserved |
| 297 | SPI 265 | Reserved | Reserved | Reserved | Reserved |
| 298 | SPI 266 | VSP2.vspbd | VSP2.vspbd | VSP2.vspbd | VSP2.vspbd |
| 299 | SPI 267 | Reserved | Reserved | Reserved | Reserved |
| 300 | SPI 268 | DU.ch1 | DU.ch1 | DU.ch1 | DU.ch1 |
| 301 | SPI 269 | Reserved | DU.ch2 | Reserved | Reserved |
| 302 | SPI 270 | DU.ch3 | Reserved | DU.ch3 | Reserved |

| Interrupt ID | SGI, PPI or SPI No | Source | | | |
|--------------|--------------------|-------------------|-----------------------------|-------------------|-------------------|
| | | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 303 | SPI 271 | Reserved | Reserved | Reserved | Reserved |
| 304 | SPI 272 | Reserved | Reserved | Reserved | Reserved |
| 305 | SPI 273 | Reserved | Reserved | Reserved | Reserved |
| 306 | SPI 274 | Reserved | Reserved | Reserved | Reserved |
| 307 | SPI 275 | Reserved | Reserved | Reserved | Reserved |
| 308 | SPI 276 | Reserved | Reserved | Reserved | Reserved |
| 309 | SPI 277 | Reserved | Reserved | Reserved | Reserved |
| 310 | SPI 278 | Reserved | Reserved | Reserved | Reserved |
| 311 | SPI 279 | Reserved | Reserved | Reserved | Reserved |
| 312 | SPI 280 | Reserved | Reserved | Reserved | Reserved |
| 313 | SPI 281 | Reserved | Reserved | Reserved | Reserved |
| 314 | SPI 282 | Reserved | Reserved | Reserved | Reserved |
| 315 | SPI 283 | Reserved | Reserved | Reserved | Reserved |
| 316 | SPI 284 | Reserved | Reserved | Reserved | Reserved |
| 317 | SPI 285 | Reserved | Reserved | Reserved | Reserved |
| 318 | SPI 286 | I2C.ch2 | I2C.ch2 | I2C.ch2 | I2C.ch2 |
| 319 | SPI 287 | I2C.ch0 | I2C.ch0 | I2C.ch0 | I2C.ch0 |
| 320 | SPI 288 | I2C.ch1 | I2C.ch1 | I2C.ch1 | I2C.ch1 |
| 321 | SPI 289 | Reserved | Reserved | Reserved | Reserved |
| 322 | SPI 290 | I2C.ch3 | I2C.ch3 | I2C.ch3 | I2C.ch3 |
| 323 | SPI 291 | Reserved | Reserved | Reserved | Reserved |
| 324 | SPI 292 | Reserved | Reserved | Reserved | Reserved |
| 325 | SPI 293 | Reserved | Reserved | Reserved | Reserved |
| 326 | SPI 294 | Reserved | Reserved | Reserved | Reserved |
| 327 | SPI 295 | Reserved | Reserved | Reserved | Reserved |
| 328 | SPI 296 | Reserved | Reserved | Reserved | Reserved |
| 329 | SPI 297 | System Controller | System Controller | System Controller | System Controller |
| 330 | SPI 298 | Reserved | Reserved | Reserved | Reserved |
| 331 | SPI 299 | Reserved | Reserved | Reserved | Reserved |
| 332 | SPI 300 | Reserved | Reserved | Reserved | Reserved |
| 333 | SPI 301 | Reserved | Reserved | Reserved | Reserved |
| 334 | SPI 302 | Reserved | Reserved | Reserved | Reserved |
| 335 | SPI 303 | TMU.TUNI6 | TMU.TUNI6 | TMU.TUNI6 | TMU.TUNI6 |
| 336 | SPI 304 | TMU.TUNI7 | TMU.TUNI7 | TMU.TUNI7 | TMU.TUNI7 |
| 337 | SPI 305 | TMU.TUNI8 | TMU.TUNI8 | TMU.TUNI8 | TMU.TUNI8 |
| 338 | SPI 306 | TMU.TICP18 | TMU.TICP18 | TMU.TICP18 | TMU.TICP18 |
| 339 | SPI 307 | Reserved | Reserved | Reserved | Reserved |
| 340 | SPI 308 | SYS-DMAC1.ch4 | SYS-DMAC1.ch4 | SYS-DMAC1.ch4 | SYS-DMAC1.ch4 |
| 341 | SPI 309 | SYS-DMAC1.ch5 | SYS-DMAC1.ch5 | SYS-DMAC1.ch5 | SYS-DMAC1.ch5 |
| 342 | SPI 310 | SYS-DMAC1.ch6 | SYS-DMAC1.ch6 | SYS-DMAC1.ch6 | SYS-DMAC1.ch6 |
| 343 | SPI 311 | SYS-DMAC1.ch7 | SYS-DMAC1.ch7 | SYS-DMAC1.ch7 | SYS-DMAC1.ch7 |
| 344 | SPI 312 | SYS-DMAC1.ch8 | SYS-DMAC1.ch8 | SYS-DMAC1.ch8 | SYS-DMAC1.ch8 |

| Interrupt ID | SGL, PPI or SPI No | Source | | | |
|--------------|--------------------|------------------|-----------------------------|------------------|------------------|
| | | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 345 | SPI 313 | SYS-DMAC1.ch9 | SYS-DMAC1.ch9 | SYS-DMAC1.ch9 | SYS-DMAC1.ch9 |
| 346 | SPI 314 | SYS-DMAC1.ch10 | SYS-DMAC1.ch10 | SYS-DMAC1.ch10 | SYS-DMAC1.ch10 |
| 347 | SPI 315 | SYS-DMAC1.ch11 | SYS-DMAC1.ch11 | SYS-DMAC1.ch11 | SYS-DMAC1.ch11 |
| 348 | SPI 316 | SYS-DMAC1.ch12 | SYS-DMAC1.ch12 | SYS-DMAC1.ch12 | SYS-DMAC1.ch12 |
| 349 | SPI 317 | SYS-DMAC1.ch13 | SYS-DMAC1.ch13 | SYS-DMAC1.ch13 | SYS-DMAC1.ch13 |
| 350 | SPI 318 | SYS-DMAC1.ch14 | SYS-DMAC1.ch14 | SYS-DMAC1.ch14 | SYS-DMAC1.ch14 |
| 351 | SPI 319 | SYS-DMAC1.ch15 | SYS-DMAC1.ch15 | SYS-DMAC1.ch15 | SYS-DMAC1.ch15 |
| 352 | SPI 320 | Audio-DMAC0.ch0 | Audio-DMAC0.ch0 | Audio-DMAC0.ch0 | Audio-DMAC0.ch0 |
| 353 | SPI 321 | Audio-DMAC0.ch1 | Audio-DMAC0.ch1 | Audio-DMAC0.ch1 | Audio-DMAC0.ch1 |
| 354 | SPI 322 | Audio-DMAC0.ch2 | Audio-DMAC0.ch2 | Audio-DMAC0.ch2 | Audio-DMAC0.ch2 |
| 355 | SPI 323 | Audio-DMAC0.ch3 | Audio-DMAC0.ch3 | Audio-DMAC0.ch3 | Audio-DMAC0.ch3 |
| 356 | SPI 324 | Audio-DMAC0.ch4 | Audio-DMAC0.ch4 | Audio-DMAC0.ch4 | Audio-DMAC0.ch4 |
| 357 | SPI 325 | Audio-DMAC0.ch5 | Audio-DMAC0.ch5 | Audio-DMAC0.ch5 | Audio-DMAC0.ch5 |
| 358 | SPI 326 | Audio-DMAC0.ch6 | Audio-DMAC0.ch6 | Audio-DMAC0.ch6 | Audio-DMAC0.ch6 |
| 359 | SPI 327 | Audio-DMAC0.ch7 | Audio-DMAC0.ch7 | Audio-DMAC0.ch7 | Audio-DMAC0.ch7 |
| 360 | SPI 328 | Audio-DMAC0.ch8 | Audio-DMAC0.ch8 | Audio-DMAC0.ch8 | Audio-DMAC0.ch8 |
| 361 | SPI 329 | Audio-DMAC0.ch9 | Audio-DMAC0.ch9 | Audio-DMAC0.ch9 | Audio-DMAC0.ch9 |
| 362 | SPI 330 | Audio-DMAC0.ch10 | Audio-DMAC0.ch10 | Audio-DMAC0.ch10 | Audio-DMAC0.ch10 |
| 363 | SPI 331 | Audio-DMAC0.ch11 | Audio-DMAC0.ch11 | Audio-DMAC0.ch11 | Audio-DMAC0.ch11 |
| 364 | SPI 332 | Audio-DMAC0.ch12 | Audio-DMAC0.ch12 | Audio-DMAC0.ch12 | Audio-DMAC0.ch12 |
| 365 | SPI 333 | Audio-DMAC0.ch13 | Audio-DMAC0.ch13 | Audio-DMAC0.ch13 | Audio-DMAC0.ch13 |
| 366 | SPI 334 | Audio-DMAC0.ch14 | Audio-DMAC0.ch14 | Audio-DMAC0.ch14 | Audio-DMAC0.ch14 |
| 367 | SPI 335 | Audio-DMAC0.ch15 | Audio-DMAC0.ch15 | Audio-DMAC0.ch15 | Audio-DMAC0.ch15 |
| 368 | SPI 336 | Audio-DMAC1.ch0 | Audio-DMAC1.ch0 | Audio-DMAC1.ch0 | Reserved |
| 369 | SPI 337 | Audio-DMAC1.ch1 | Audio-DMAC1.ch1 | Audio-DMAC1.ch1 | Reserved |
| 370 | SPI 338 | Audio-DMAC1.ch2 | Audio-DMAC1.ch2 | Audio-DMAC1.ch2 | Reserved |
| 371 | SPI 339 | Audio-DMAC1.ch3 | Audio-DMAC1.ch3 | Audio-DMAC1.ch3 | Reserved |
| 372 | SPI 340 | Audio-DMAC1.ch4 | Audio-DMAC1.ch4 | Audio-DMAC1.ch4 | Reserved |
| 373 | SPI 341 | Audio-DMAC1.ch5 | Audio-DMAC1.ch5 | Audio-DMAC1.ch5 | Reserved |
| 374 | SPI 342 | Audio-DMAC1.ch6 | Audio-DMAC1.ch6 | Audio-DMAC1.ch6 | Reserved |
| 375 | SPI 343 | Audio-DMAC1.ch7 | Audio-DMAC1.ch7 | Audio-DMAC1.ch7 | Reserved |
| 376 | SPI 344 | Audio-DMAC1.ch8 | Audio-DMAC1.ch8 | Audio-DMAC1.ch8 | Reserved |
| 377 | SPI 345 | Audio-DMAC1.ch9 | Audio-DMAC1.ch9 | Audio-DMAC1.ch9 | Reserved |

| Interrupt ID | SGI, PPI or SPI No | Source | | | |
|--------------|--------------------|------------------|-----------------------------|------------------|-----------------|
| | | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 378 | SPI 346 | Audio-DMAC1.ch10 | Audio-DMAC1.ch10 | Audio-DMAC1.ch10 | Reserved |
| 379 | SPI 347 | Audio-DMAC1.ch11 | Audio-DMAC1.ch11 | Audio-DMAC1.ch11 | Reserved |
| 380 | SPI 348 | Audio-DMAC1.ch12 | Audio-DMAC1.ch12 | Audio-DMAC1.ch12 | Reserved |
| 381 | SPI 349 | Audio-DMAC1.ch13 | Audio-DMAC1.ch13 | Audio-DMAC1.ch13 | Reserved |
| 382 | SPI 350 | Audio-DMAC0 err | Audio-DMAC0 err | Audio-DMAC0 err | Audio-DMAC0 err |
| 383 | SPI 351 | Audio-DMAC1 err | Audio-DMAC1 err | Audio-DMAC1 err | Reserved |
| 384 | SPI 352 | SCU.ch0 | SCU.ch0 | SCU.ch0 | SCU.ch0 |
| 385 | SPI 353 | SCU.ch1 | SCU.ch1 | SCU.ch1 | SCU.ch1 |
| 386 | SPI 354 | SCU.ch2 | SCU.ch2 | SCU.ch2 | SCU.ch2 |
| 387 | SPI 355 | SCU.ch3 | SCU.ch3 | SCU.ch3 | SCU.ch3 |
| 388 | SPI 356 | SCU.ch4 | SCU.ch4 | SCU.ch4 | SCU.ch4 |
| 389 | SPI 357 | SCU.ch5 | SCU.ch5 | SCU.ch5 | SCU.ch5 |
| 390 | SPI 358 | SCU.ch6 | SCU.ch6 | SCU.ch6 | SCU.ch6 |
| 391 | SPI 359 | SCU.ch7 | SCU.ch7 | SCU.ch7 | SCU.ch7 |
| 392 | SPI 360 | SCU.ch8 | SCU.ch8 | SCU.ch8 | SCU.ch8 |
| 393 | SPI 361 | SCU.ch9 | SCU.ch9 | SCU.ch9 | SCU.ch9 |
| 394 | SPI 362 | Reserved | Reserved | Reserved | Reserved |
| 395 | SPI 363 | Reserved | Reserved | Reserved | Reserved |
| 396 | SPI 364 | Reserved | Reserved | Reserved | Reserved |
| 397 | SPI 365 | Reserved | Reserved | Reserved | Reserved |
| 398 | SPI 366 | Reserved | Reserved | Reserved | Reserved |
| 399 | SPI 367 | Reserved | Reserved | Reserved | Reserved |
| 400 | SPI 368 | Reserved | Reserved | Reserved | Reserved |
| 401 | SPI 369 | Reserved | Reserved | Reserved | Reserved |
| 402 | SPI 370 | SSI.ch0 | SSI.ch0 | SSI.ch0 | SSI.ch0 |
| 403 | SPI 371 | SSI.ch1 | SSI.ch1 | SSI.ch1 | SSI.ch1 |
| 404 | SPI 372 | SSI.ch2 | SSI.ch2 | SSI.ch2 | SSI.ch2 |
| 405 | SPI 373 | SSI.ch3 | SSI.ch3 | SSI.ch3 | SSI.ch3 |
| 406 | SPI 374 | SSI.ch4 | SSI.ch4 | SSI.ch4 | SSI.ch4 |
| 407 | SPI 375 | SSI.ch5 | SSI.ch5 | SSI.ch5 | SSI.ch5 |
| 408 | SPI 376 | SSI.ch6 | SSI.ch6 | SSI.ch6 | SSI.ch6 |
| 409 | SPI 377 | SSI.ch7 | SSI.ch7 | SSI.ch7 | SSI.ch7 |
| 410 | SPI 378 | SSI.ch8 | SSI.ch8 | SSI.ch8 | SSI.ch8 |
| 411 | SPI 379 | SSI.ch9 | SSI.ch9 | SSI.ch9 | SSI.ch9 |
| 412 | SPI 380 | iVDP1C vint | iVDP1C vint | iVDP1C vint | iVDP1C vint |
| 413 | SPI 381 | iVDP1C cint | iVDP1C cint | iVDP1C cint | iVDP1C cint |
| 414 | SPI 382 | Audio-DMAC1.ch14 | Audio-DMAC1.ch14 | Audio-DMAC1.ch14 | Reserved |
| 415 | SPI 383 | Audio-DMAC1.ch15 | Audio-DMAC1.ch15 | Audio-DMAC1.ch15 | Reserved |

| Interrupt ID | SGL, PPI or SPI No | Source | | | |
|--------------|--------------------|----------------|-----------------------------|----------------|----------------|
| | | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 416 | SPI 384 | Reserved | Reserved | Reserved | Reserved |
| 417 | SPI 385 | Reserved | Reserved | Reserved | Reserved |
| 418 | SPI 386 | Reserved | Reserved | Reserved | Reserved |
| 419 | SPI 387 | Reserved | Reserved | Reserved | Reserved |
| 420 | SPI 388 | Reserved | Reserved | Reserved | Reserved |
| 421 | SPI 389 | HDMI0 master | HDMI0 master | HDMI0 master | Reserved |
| 422 | SPI 390 | Reserved | Reserved | Reserved | Reserved |
| 423 | SPI 391 | HDMI0 esm | HDMI0 esm | HDMI0 esm | Reserved |
| 424 | SPI 392 | HDMI0 rng | HDMI0 rng | HDMI0 rng | Reserved |
| 425 | SPI 393 | Reserved | Reserved | Reserved | Reserved |
| 426 | SPI 394 | Reserved | Reserved | Reserved | Reserved |
| 427 | SPI 395 | Reserved | Reserved | Reserved | Reserved |
| 428 | SPI 396 | Reserved | Reserved | Reserved | Reserved |
| 429 | SPI 397 | SYS-DMAC2.ch15 | SYS-DMAC2.ch15 | SYS-DMAC2.ch15 | SYS-DMAC2.ch15 |
| 430 | SPI 398 | CMT2.ch0 | CMT2.ch0 | CMT2.ch0 | CMT2.ch0 |
| 431 | SPI 399 | CMT2.ch1 | CMT2.ch1 | CMT2.ch1 | CMT2.ch1 |
| 432 | SPI 400 | CMT2.ch2 | CMT2.ch2 | CMT2.ch2 | CMT2.ch2 |
| 433 | SPI 401 | CMT2.ch3 | CMT2.ch3 | CMT2.ch3 | CMT2.ch3 |
| 434 | SPI 402 | CMT2.ch4 | CMT2.ch4 | CMT2.ch4 | CMT2.ch4 |
| 435 | SPI 403 | CMT2.ch5 | CMT2.ch5 | CMT2.ch5 | CMT2.ch5 |
| 436 | SPI 404 | CMT2.ch6 | CMT2.ch6 | CMT2.ch6 | CMT2.ch6 |
| 437 | SPI 405 | CMT2.ch7 | CMT2.ch7 | CMT2.ch7 | CMT2.ch7 |
| 438 | SPI 406 | TMU.TUNI12 | TMU.TUNI12 | TMU.TUNI12 | TMU.TUNI12 |
| 439 | SPI 407 | TMU.TUNI13 | TMU.TUNI13 | TMU.TUNI13 | TMU.TUNI13 |
| 440 | SPI 408 | TMU.TUNI14 | TMU.TUNI14 | TMU.TUNI14 | TMU.TUNI14 |
| 441 | SPI 409 | Reserved | Reserved | Reserved | Reserved |
| 442 | SPI 410 | Reserved | Reserved | Reserved | Reserved |
| 443 | SPI 411 | Reserved | Reserved | Reserved | Reserved |
| 444 | SPI 412 | Reserved | Reserved | Reserved | Reserved |
| 445 | SPI 413 | Reserved | Reserved | Reserved | Reserved |
| 446 | SPI 414 | Reserved | Reserved | Reserved | Reserved |
| 447 | SPI 415 | Reserved | Reserved | Reserved | Reserved |
| 448 | SPI 416 | SYS-DMAC2 err | SYS-DMAC2 err | SYS-DMAC2 err | SYS-DMAC2 err |
| 449 | SPI 417 | SYS-DMAC2.ch0 | SYS-DMAC2.ch0 | SYS-DMAC2.ch0 | SYS-DMAC2.ch0 |
| 450 | SPI 418 | SYS-DMAC2.ch1 | SYS-DMAC2.ch1 | SYS-DMAC2.ch1 | SYS-DMAC2.ch1 |
| 451 | SPI 419 | SYS-DMAC2.ch2 | SYS-DMAC2.ch2 | SYS-DMAC2.ch2 | SYS-DMAC2.ch2 |
| 452 | SPI 420 | SYS-DMAC2.ch3 | SYS-DMAC2.ch3 | SYS-DMAC2.ch3 | SYS-DMAC2.ch3 |
| 453 | SPI 421 | SYS-DMAC2.ch4 | SYS-DMAC2.ch4 | SYS-DMAC2.ch4 | SYS-DMAC2.ch4 |
| 454 | SPI 422 | SYS-DMAC2.ch5 | SYS-DMAC2.ch5 | SYS-DMAC2.ch5 | SYS-DMAC2.ch5 |
| 455 | SPI 423 | SYS-DMAC2.ch6 | SYS-DMAC2.ch6 | SYS-DMAC2.ch6 | SYS-DMAC2.ch6 |
| 456 | SPI 424 | SYS-DMAC2.ch7 | SYS-DMAC2.ch7 | SYS-DMAC2.ch7 | SYS-DMAC2.ch7 |

| Interrupt ID | SGL, PPI or SPI No | Source | | | |
|--------------|--------------------|----------------|-----------------------------|----------------|----------------|
| | | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 457 | SPI 425 | SYS-DMAC2.ch8 | SYS-DMAC2.ch8 | SYS-DMAC2.ch8 | SYS-DMAC2.ch8 |
| 458 | SPI 426 | SYS-DMAC2.ch9 | SYS-DMAC2.ch9 | SYS-DMAC2.ch9 | SYS-DMAC2.ch9 |
| 459 | SPI 427 | SYS-DMAC2.ch10 | SYS-DMAC2.ch10 | SYS-DMAC2.ch10 | SYS-DMAC2.ch10 |
| 460 | SPI 428 | SYS-DMAC2.ch11 | SYS-DMAC2.ch11 | SYS-DMAC2.ch11 | SYS-DMAC2.ch11 |
| 461 | SPI 429 | SYS-DMAC2.ch12 | SYS-DMAC2.ch12 | SYS-DMAC2.ch12 | SYS-DMAC2.ch12 |
| 462 | SPI 430 | SYS-DMAC2.ch13 | SYS-DMAC2.ch13 | SYS-DMAC2.ch13 | SYS-DMAC2.ch13 |
| 463 | SPI 431 | SYS-DMAC2.ch14 | SYS-DMAC2.ch14 | SYS-DMAC2.ch14 | SYS-DMAC2.ch14 |
| 464 | SPI 432 | Reserved | Reserved | Reserved | Reserved |
| 465 | SPI 433 | Reserved | Reserved | Reserved | Reserved |
| 466 | SPI 434 | Reserved | Reserved | Reserved | Reserved |
| 467 | SPI 435 | Reserved | Reserved | Reserved | Reserved |
| 468 | SPI 436 | Reserved | Reserved | Reserved | Reserved |
| 469 | SPI 437 | Reserved | Reserved | Reserved | Reserved |
| 470 | SPI 438 | Reserved | Reserved | Reserved | Reserved |
| 471 | SPI 439 | Reserved | Reserved | Reserved | Reserved |
| 472 | SPI 440 | Reserved | Reserved | Reserved | Reserved |
| 473 | SPI 441 | Reserved | Reserved | Reserved | Reserved |
| 474 | SPI 442 | Reserved | Reserved | Reserved | Reserved |
| 475 | SPI 443 | Reserved | Reserved | Reserved | Reserved |
| 476 | SPI 444 | VSP2.vspi.ch0 | VSP2.vspi.ch0 | VSP2.vspi.ch0 | VSP2.vspi.ch0 |
| 477 | SPI 445 | VSP2.vspi.ch1 | Reserved | Reserved | Reserved |
| 478 | SPI 446 | Reserved | Reserved | Reserved | Reserved |
| 479 | SPI 447 | Reserved | Reserved | Reserved | Reserved |
| 480 | SPI 448 | Reserved | Reserved | Reserved | Reserved |
| 481 | SPI 449 | Reserved | Reserved | Reserved | Reserved |
| 482 | SPI 450 | Reserved | Reserved | Reserved | Reserved |
| 483 | SPI 451 | Reserved | Reserved | Reserved | Reserved |
| 484 | SPI 452 | Reserved | Reserved | Reserved | Reserved |
| 485 | SPI 453 | Reserved | Reserved | Reserved | Reserved |
| 486 | SPI 454 | Reserved | Reserved | Reserved | Reserved |
| 487 | SPI 455 | Reserved | Reserved | Reserved | Reserved |
| 488 | SPI 456 | Reserved | Reserved | Reserved | Reserved |
| 489 | SPI 457 | Reserved | Reserved | Reserved | Reserved |
| 490 | SPI 458 | Reserved | Reserved | Reserved | Reserved |
| 491 | SPI 459 | Reserved | Reserved | Reserved | Reserved |
| 492 | SPI 460 | Reserved | Reserved | Reserved | Reserved |
| 493 | SPI 461 | Reserved | Reserved | Reserved | Reserved |

| Interrupt ID | SGI, PPI or SPI No | Source | | | |
|--------------|--------------------|-------------------|-----------------------------|-------------------|-------------------|
| | | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 494 | SPI 462 | Reserved | Reserved | Reserved | Reserved |
| 495 | SPI 463 | Reserved | Reserved | Reserved | Reserved |
| 496 | SPI 464 | Reserved | Reserved | Reserved | Reserved |
| 497 | SPI 465 | VSP2.vspbc | Reserved | Reserved | Reserved |
| 498 | SPI 466 | VSP2.vspd. ch0 | VSP2.vspd. ch0 | VSP2.vspd. ch0 | VSP2.vspd. ch0 |
| 499 | SPI 467 | VSP2.vspd. ch1 | VSP2.vspd. ch1 | VSP2.vspd. ch1 | VSP2.vspd. ch1 |
| 500 | SPI 468 | Reserved | VSP2.vspd. ch2 | Reserved | Reserved |
| 501 | SPI 469 | Reserved | Reserved | Reserved | Reserved |
| 502 | SPI 470 | CMT3.ch0 | CMT3.ch0 | CMT3.ch0 | CMT3.ch0 |
| 503 | SPI 471 | CMT3.ch1 | CMT3.ch1 | CMT3.ch1 | CMT3.ch1 |
| 504 | SPI 472 | CMT3.ch2 | CMT3.ch2 | CMT3.ch2 | CMT3.ch2 |
| 505 | SPI 473 | CMT3.ch3 | CMT3.ch3 | CMT3.ch3 | CMT3.ch3 |
| 506 | SPI 474 | CMT3.ch4 | CMT3.ch4 | CMT3.ch4 | CMT3.ch4 |
| 507 | SPI 475 | CMT3.ch5 | CMT3.ch5 | CMT3.ch5 | CMT3.ch5 |
| 508 | SPI 476 | CMT3.ch6 | CMT3.ch6 | CMT3.ch6 | CMT3.ch6 |
| 509 | SPI 477 | CMT3.ch7 | CMT3.ch7 | CMT3.ch7 | CMT3.ch7 |
| 510 | SPI 478 | Reserved | Reserved | Reserved | Reserved |
| 511 | SPI 479 | Reserved | Reserved | Reserved | Reserved |

20. Interrupt Controller (INTC-EX)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

20.1 Overview

The INTC-EX is a common sub-module of interrupt controllers. This sub-module provides interrupt from external device and inter-domain communication.

20.1.1 Features

- Common module to handle the external NMI/IRQ inputs
- Edge-triggered on rising, falling or both
- Level-sensitive on high or low values
- IRQ signal debouncing, noise reduction and chattering

20.1.2 Block Diagram

Figure 20.1 and Figure 20.2 are block diagrams of INTC-EX.

The event detector block detects external IRQn, and supports noise reduction and edge detection.

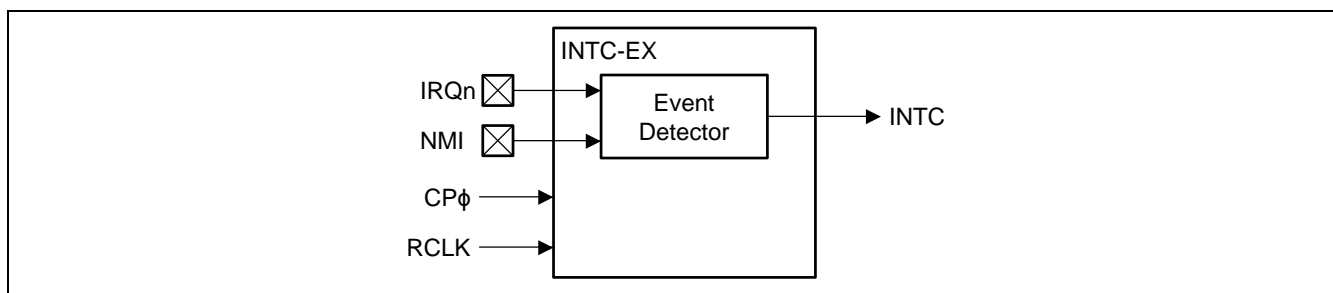


Figure 20.1 INTC-EX Block Diagram

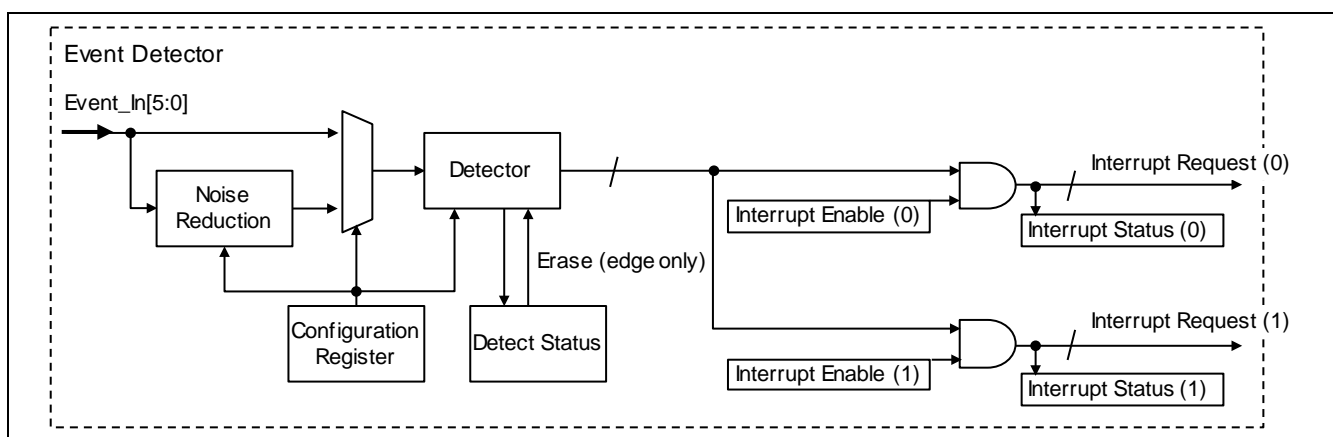


Figure 20.2 INTC-EX Event Detector

20.1.3 External Pins

Table 20.1 shows the INTC-EX pin configuration.

Table 20.1 Pin Configuration

| Pin Name | Function | I/O | Description | Second Generation RZ/G Series Products | | | | |
|----------------------|--|-------|--|---|----------------------------|--------|--------|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| IRQn (n = 0 to 5) | External interrupt input pins | Input | Input of IRQn interrupt request signals from external device. All IRQ signals are always active. | √ | √ | √ | √ | |
| NMI | External non maskable Interrupt input pins | Input | Input of NMI interrupt request signal from external device. | √ | √ | √ | √ | |

20.1.4 Register Configuration

Table 20.2 shows the base address of each block and Table 20.3 to Table 20.5 show register configuration.

Table 20.2 INTC-EX Base Address

| Module Name | Base Address |
|------------------------------|--------------|
| INTC-EX | H'E61C_0000 |
| INTC-EX event detector block | H'E61C_0000 |
| NMI event detector block | H'E61C_0400 |
| NMI mask lock block | H'E61C_0A00 |

Table 20.3 INTC-EX Event Detector Register Configuration

| Register Name | Abbreviation | R/W | Address Offset | Initial value | Size | Second Generation RZ/G Series Products | | | |
|--|---------------|-----------|----------------|---------------|------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Interrupt request status register 0 | INTREQ_STS0 | R | H'000 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Interrupt enable status register 0 | INTEN_STS0 | R/WC 1 | H'004 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Interrupt enable set register 0 | INTEN_SET0 | W | H'008 | — | 32 | √ | √ | √ | √ |
| IRQn detect status register | DETECT_STATUS | R/WC 1 | H'100 | H'0000_0000 | 32 | √ | √ | √ | √ |
| IRQn signal level monitor register | MONITOR | R | H'104 | — | 32 | √ | √ | √ | √ |
| IRQn high level detect status register | HLVL_STS | R | H'108 | H'0000_0000 | 32 | √ | √ | √ | √ |
| IRQn low level detect status register | LLVL_STS | R | H'10C | H'0000_0000 | 32 | √ | √ | √ | √ |

| Register Name | Abbreviation | R/W | Address Offset | Initial value | Size | Second Generation RZ/G Series Products | | | |
|--|--------------|-----|----------------|---------------|------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| IRQn sync rising edge detect status register | S_R_EDGE_STS | R | H'110 | H'0000_0000 | 32 | √ | √ | √ | √ |
| IRQn sync falling edge detect status register | S_F_EDGE_STS | R | H'114 | H'0000_0000 | 32 | √ | √ | √ | √ |
| IRQn async rising edge detect status register | A_R_EDGE_STS | R | H'118 | H'0000_0000 | 32 | √ | √ | √ | √ |
| IRQn async falling edge detect status register | A_F_EDGE_STS | R | H'11C | H'0000_0000 | 32 | √ | √ | √ | √ |
| IRQn chattering reduction status register | CHTEN_STS | R | H'120 | H'0000_0000 | 32 | √ | √ | √ | √ |
| IRQ0 configuration register | CONFIG_0 | R/W | H'180 | H'0000_0000 | 32 | √ | √ | √ | √ |
| IRQ1 configuration register | CONFIG_1 | R/W | H'184 | H'0000_0000 | 32 | √ | √ | √ | √ |
| IRQ2 configuration register | CONFIG_2 | R/W | H'188 | H'0000_0000 | 32 | √ | √ | √ | √ |
| IRQ3 configuration register | CONFIG_3 | R/W | H'18C | H'0000_0000 | 32 | √ | √ | √ | √ |
| IRQ4 configuration register | CONFIG_4 | R/W | H'190 | H'0000_0000 | 32 | √ | √ | √ | √ |
| IRQ5 configuration register | CONFIG_5 | R/W | H'194 | H'0000_0000 | 32 | √ | √ | √ | √ |

Table 20.4 NMI Event Detector Register Configuration

| Register Name | Abbreviation | R/W | Address Offset | Initial value | Size | Second Generation RZ/G Series Products | | | |
|---|-------------------|-------|----------------|---------------|------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| NMI request status register 0 | NMIREQ_STS0 | R | H'000 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI enable status register 0 | NMIEN_STS0 | R/WC1 | H'004 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI enable set register 0 | NMIEN_SET0 | W | H'008 | — | 32 | √ | √ | √ | √ |
| NMI detect status register | DETECT_STATUS_NMI | R/WC1 | H'100 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI signal level monitor register | MONITOR_NMI | R | H'104 | — | 32 | √ | √ | √ | √ |
| NMI high level detect status register | HLVL_STS_NMI | R | H'108 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI low level detect status register | LLVL_STS_NMI | R | H'10C | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI sync rising edge detect status register | S_R_EDGE_STS_NMI | R | H'110 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI sync falling edge detect status register | S_F_EDGE_STS_NMI | R | H'114 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI async rising edge detect status register | A_R_EDGE_STS_NMI | R | H'118 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI async falling edge detect status register | A_F_EDGE_STS_NMI | R | H'11C | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI chattering reduction status register | CHTEN_STS_NMI | R | H'120 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI debounce setting register | DEB_SET_NMI | R/W | H'140 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI configuration 0 register | CONFIG0_NMI | R/W | H'180 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI configuration 1 register | CONFIG1_NMI | R/W | H'184 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI configuration 2 register | CONFIG2_NMI | R/W | H'188 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI configuration 3 register | CONFIG3_NMI | R/W | H'18C | H'0000_0000 | 32 | √ | √ | √ | √ |

| | | | | | | Second Generation RZ/G Series Products | | | |
|------------------------------|--------------|-----|-------------------|---------------|------|---|----------------------------|--------|--------|
| Register Name | Abbreviation | R/W | Address Offset | Initial value | Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| NMI configuration 4 register | CONFIG4_NMI | R/W | H'190 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI configuration 5 register | CONFIG5_NMI | R/W | H'194 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI configuration 6 register | CONFIG6_NMI | R/W | H'198 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI configuration 7 register | CONFIG7_NMI | R/W | H'19C | H'0000_0000 | 32 | √ | √ | √ | √ |

Table 20.5 NMI Lock Register Configuration

| | | | | | | Second Generation RZ/G Series Products | | | |
|-----------------------------------|--------------|------|-------------------|------------------|------|---|----------------------------|--------|--------|
| Register Name | Abbreviation | R/W | Address Offset | Initial value | Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| NMI mask lock set register | NMI_LCK | R/W* | H'000 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI lock code register | NMI_LCKCODE | R/W* | H'004 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI debug control enable register | NMI_DBG | R/W | H'008 | H'0000_0000 | 32 | √ | √ | √ | √ |
| NMI debug code register | NMI_DBGCODE | R/W | H'00C | H'0000_0000 | 32 | √ | √ | √ | √ |

Note: * It cannot update register value during lock.

20.1.5 Connected Module

Table 20.6 Connected modules

| Module name | Connected module name | Function of connected module |
|-------------|-----------------------|---|
| INTC-EX | INTC-AP | Interrupt controller for AP-System Core |

20.2 Register Description

20.2.1 Interrupt Request Status Register0 (INTREQ_STS0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register shows interrupt request status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|--------|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | INTREQ | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|--|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | INTREQ[5:0] | H'00 | R | Interrupt Status 0: Interrupt not generation 1: Interrupt generation |

20.2.2 Interrupt Enable Status Register0 (INTEN_STS0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register shows interrupt enable status and clear interrupt enable.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | INTEN | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-------|---|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | INTEN[5:0] | H'00 | R/WC1 | Interrupt Enable Read 0: Interrupt generation is disabled Read 1: Interrupt generation is enabled Write 0: No functional effect Write 1: Interrupt enable clear |

20.2.3 Interrupt Enable Set Register0 (INTEN_SET0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register set interrupt enable.

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|--------|----|----|----|----|----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | — | — | — | — | — | INTENS | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — | — | — | — | — | |
| R/W: | R | R | R | R | R | R | R | R | R | R | W | W | W | W | W | W | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|--|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | INTENS[5:0] | — | W | Interrupt Enable Set Write 0: No functional effect Write 1: Interrupt enable set |

20.2.4 IRQn Detect Status Register (DETECT_STATUS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register shows IRQn event detection status and provides the function to clear edge-triggered event. The status bit is cleared by writing 1 to the corresponding bit in edge-triggered mode. Writing 0 to this register bits does not affect to the register value.

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|---------|-------|-------|-------|-------|-------|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | — | — | — | — | — | IRQnDET | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-------|---|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | IRQnDET [5:0] | H'00 | R/WC1 | IRQn Event Detection Status Edge-triggered mode: Read 0: No interrupt request occurred Read 1: Interrupt request occurred Write 0: No functional effect Write 1: Cleared detection. Level-sensitive mode: Read 0: No interrupt request occurred. Read 1: Interrupt request occurred Write 0: No functional effect. Write 1: No functional effect. |

Note: n = 5

20.2.5 IRQn Signal Level Monitor Register (MONITOR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides external signal monitor.

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|---------|----|----|----|----|----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | — | — | — | — | — | IRQnMON | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — | — | — | — | — | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | IRQnMON [5:0] | — | R | IRQn External Signal Level Monitor This function show input value when signal is enabled by corresponding configuration register SS bits. So, initial value depends on input value. 0: IRQn is low level 1: IRQn is high level |

Note: n = 5

20.2.6 IRQn High Level Detect Status Register (HLVL_STS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides interrupt detail detect status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----------|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | IRQnHSTS | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | IRQnHSTS[5:0] | H'00 | R | IRQn High Level Interrupt Status 0: IRQn high level interrupt request not occurred 1: IRQn high level interrupt request occurred |

Note: n = 5

20.2.7 IRQn Low Level Detect Status Register (LLVL_STS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides interrupt detail detect status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----------|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | IRQnLSTS | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | IRQnLSTS[5:0] | H'00 | R | IRQn Low Level Interrupt Status 0: IRQn low level interrupt request not occurred 1: IRQn low level interrupt request occurred |

Note: n = 5

20.2.8 IRQn Sync Rising Edge Detect Status Register (S_R_EDGE_STS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides interrupt detail detect status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|-----------|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | IRQnSRSTS | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | IRQnSRSTS [5:0] | H'00 | R | IRQn Synchronous Rise Edge Interrupt Status 0: IRQn rise edge interrupt request not occurred 1: IRQn rise edge interrupt request occurred |

Note: n = 5

20.2.9 IRQn Sync Falling Edge Detect Status Register (S_F_EDGE_STS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides interrupt detail detect status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-----------|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | IRQnSFSTS | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | IRQnSFSTS [5:0] | H'00 | R | IRQn Synchronous Fall Edge Interrupt Status 0: IRQn fall edge interrupt request not occurred 1: IRQn fall edge interrupt request occurred |

Note: n = 5

20.2.10 IRQn Async Rising Edge Detect Status Register (A_R_EDGE_STS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides interrupt detail detect status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-----------|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | IRQnARSTS | | | | | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|--|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | IRQnARSTS [5:0] | H'00 | R | IRQn Asynchronous Rise Edge Interrupt Status 0: IRQn rise edge interrupt request not occurred 1: IRQn rise edge interrupt request occurred |

Note: n = 5

20.2.11 IRQn Async Falling Edge Detect Status Register (A_F_EDGE_STS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides interrupt detail detect status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|-----------|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | IRQnAFSTS | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|--|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | IRQnAFSTS [5:0] | H'00 | R | IRQn Asynchronous Fall Edge Interrupt Status 0: IRQn fall edge interrupt request not occurred 1: IRQn fall edge interrupt request occurred |

Note: n = 5

20.2.12 IRQn Chattering Reduction Status Register (CHTEN_STS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register shows chattering reduction enable status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-------|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | CHTEN | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | CHTEN[5:0] | H'00 | R | Chattering Reduction Enable Status 0: Chattering reduction disabled 1: Chattering reduction enabled |

Note: After you set CHTEN.CONFIG_n bit, these bits show its status.
However, 3 cycles @ RCLK (32 kHz) is necessary to reflect the set value.

20.2.13 IRQn Configuration Register (CONFIG_n)

Note: n = 0 to 5

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides detection mode and chattering reduction setting.

| | | | | | | | | | | | | | | | | |
|----------------|-------|----|----|----|----|----|----|----|-------|-------|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CHTEN | — | — | — | — | — | — | — | ST S1 | ST S2 | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | SS | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | CHTEN | B'0 | R/W | Chattering Reduction Enable 0: Chattering reduction disabled 1: Chattering reduction enabled |
| 30 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23, 22 | ST S1 | B'00 | R/W | IRQn Scan Timing These bits provide chattering reduction timing. B'00: 1 ms B'01: 2 ms B'10: 4 ms B'11: 8 ms |
| 21 to 16 | ST S2 | H'00 | R/W | IRQn Chattering Reduction Period The chattering reduction period is defined by ST S1 × ST S2. H'00: No reduction H'01 to H'3F: ST S1 × ST S2 Notes: 1. These bits should not be set to H'00 when chattering reduction enable. 2. Check chattering reduction is disabled completely with corresponding bit of chattering reduction status register (CHTEN_ST S) when this bit change. |
| 15 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The writing value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 5 to 0 | SS | H'00 | R/W | <p>Sense Selection</p> <p>B'00_0000: Disable event detection</p> <p>B'00_0001: Enable low level sensitive</p> <p>B'00_0010: Enable high level sensitive</p> <p>B'00_0100: Enable synchronous falling edge triggered</p> <p>B'00_1000: Enable synchronous rising edge triggered</p> <p>B'00_1100: Enable synchronous both edge triggered</p> <p>B'01_0000: Enable asynchronous falling edge triggered</p> <p>B'10_0000: Enable asynchronous rising edge triggered</p> <p>B'11_0000: Enable asynchronous both edge triggered</p> <p>Others : setting prohibit</p> <p>Note: The Asynchronous edge triggered can use only as follow conditions.</p> <p>+ Chattering reduction not use.</p> <p>The synchronous edge triggered can use always.</p> |

20.2.14 NMI Request Status Register 0 (NMIREQ_STS0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register shows external NMI request status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | C7STS | C6STS | C5STS | C4STS | C3STS | C2STS | C1STS | C0STS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | C7STS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI Status for CPU7 interface (INTC-AP) 0: not during NMI service 1: during NMI service [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 6 | C6STS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI Status for CPU6 interface (INTC-AP) 0: not during NMI service 1: during NMI service [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 5 | C5STS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI Status for CPU5 interface (INTC-AP) 0: not during NMI service 1: during NMI service [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | C4STS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI Status for CPU4 interface (INTC-AP) 0: not during NMI service 1: during NMI service [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | C3STS | B'0 | R | <p>[RZ/G2H] NMI Status for CPU3 interface (INTC-AP) 0: not during NMI service 1: during NMI service</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0.</p> |
| 2 | C2STS | B'0 | R | <p>[RZ/G2H] NMI Status for CPU2 interface (INTC-AP) 0: not during NMI service 1: during NMI service</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0.</p> |
| 1 | C1STS | B'0 | R | <p>NMI Status for CPU1 interface (INTC-AP) 0: not during NMI service 1: during NMI service</p> |
| 0 | C0STS | B'0 | R | <p>NMI Status for CPU0 interface (INTC-AP) 0: not during NMI service 1: during NMI service</p> |

20.2.15 NMI Enable Status Register 0 (NMIEN_STS0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register shows NMI interrupt enable status and clears NMI interrupt enable.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | C7IEN | C6IEN | C5IEN | C4IEN | C3IEN | C2IEN | C1IEN | C0IEN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-------|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | C7IEN | B'0 | R/WC1 | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] Interrupt Enable for CPU7 interface (INTC-AP) Read 0: NMI interrupt generation is disabled Read 1: NMI interrupt generation is enabled Write 0: No functional effect Write 1: NMI interrupt enable clear |
| 6 | C6IEN | B'0 | R/WC1 | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] Interrupt Enable for CPU6 interface (INTC-AP) Read 0: NMI interrupt generation is disabled Read 1: NMI interrupt generation is enabled Write 0: No functional effect Write 1: NMI interrupt enable clear |
| 5 | C5IEN | B'0 | R/WC1 | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] Interrupt Enable for CPU5 interface (INTC-AP) Read 0: NMI interrupt generation is disabled Read 1: NMI interrupt generation is enabled Write 0: No functional effect Write 1: NMI interrupt enable clear |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 4 | C4IEN | B'0 | R/WC1 | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>Interrupt Enable for CPU4 interface (INTC-AP)</p> <p>Read 0: NMI interrupt generation is disabled</p> <p>Read 1: NMI interrupt generation is enabled</p> <p>Write 0: No functional effect</p> <p>Write 1: NMI interrupt enable clear</p> <hr/> <p>[RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 3 | C3IEN | B'0 | R/WC1 | <p>[RZ/G2H]</p> <p>Interrupt Enable for CPU3 interface (INTC-AP)</p> <p>Read 0: NMI interrupt generation is disabled</p> <p>Read 1: NMI interrupt generation is enabled</p> <p>Write 0: No functional effect</p> <p>Write 1: NMI interrupt enable clear</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 2 | C2IEN | B'0 | R/WC1 | <p>[RZ/G2H]</p> <p>Interrupt Enable for CPU2 interface (INTC-AP)</p> <p>Read 0: NMI interrupt generation is disabled</p> <p>Read 1: NMI interrupt generation is enabled</p> <p>Write 0: No functional effect</p> <p>Write 1: NMI interrupt enable clear</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 1 | C1IEN | B'0 | R/WC1 | <p>Interrupt Enable for CPU1 interface (INTC-AP)</p> <p>Read 0: NMI interrupt generation is disabled</p> <p>Read 1: NMI interrupt generation is enabled</p> <p>Write 0: No functional effect</p> <p>Write 1: NMI interrupt enable clear</p> |
| 0 | C0IEN | B'0 | R/WC1 | <p>Interrupt Enable for CPU0 interface (INTC-AP)</p> <p>Read 0: NMI interrupt generation is disabled</p> <p>Read 1: NMI interrupt generation is enabled</p> <p>Write 0: No functional effect</p> <p>Write 1: NMI interrupt enable clear</p> |

20.2.16 NMI Enable Set Register 0 (NMIEN_SET0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register enables the NMI interrupt to each CPU.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | C7SET | C6SET | C5SET | C4SET | C3SET | C2SET | C1SET | C0SET |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | W | W | W | W | W | W | W | W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | C7SET | — | W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] Interrupt Enable Set for CPU7 interface (INTC-AP) Write 0: No functional effect Write 1: Interrupt enable set [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 6 | C6SET | — | W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] Interrupt Enable Set for CPU6 interface (INTC-AP) Write 0: No functional effect Write 1: Interrupt enable set [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 5 | C5SET | — | W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] Interrupt Enable Set for CPU5 interface (INTC-AP) Write 0: No functional effect Write 1: Interrupt enable set [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | C4SET | — | W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] Interrupt Enable Set for CPU4 interface (INTC-AP) Write 0: No functional effect Write 1: Interrupt enable set [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 3 | C3SET | — | W | <p>[RZ/G2H] Interrupt Enable Set for CPU3 interface (INTC-AP) Write 0: No functional effect Write 1: Interrupt enable set</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0.</p> |
| 2 | C2SET | — | W | <p>[RZ/G2H] Interrupt Enable Set for CPU2 interface (INTC-AP) Write 0: No functional effect Write 1: Interrupt enable set</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0.</p> |
| 1 | C1SET | — | W | <p>Interrupt Enable Set for CPU1 interface (INTC-AP) Write 0: No functional effect Write 1: Interrupt enable set</p> |
| 0 | C0SET | — | W | <p>Interrupt Enable Set for CPU0 interface (INTC-AP) Write 0: No functional effect Write 1: Interrupt enable set</p> |

20.2.17 NMI Detect Status Register (DETECT_STATUS_NMI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register shows NMI event detection status and provides the function to clear edge-triggered event. The status bit is cleared by writing 1 to the corresponding bit in edge-triggered mode. Writing 0 to this register bits does not affect to the register value.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | NMI7D ET | NMI6D ET | NMI5D ET | NMI4D ET | NMI3D ET | NMI2D ET | NMI1D ET | NMI0D ET |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-------|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | NMI7DET | B'0 | R/WC1 | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI7 Event Detection Status for CPU7 interface (INTC-AP) Edge-triggered mode: Read 0: No interrupt request occurred Read 1: Interrupt request occurred Write 0: No functional effect Write 1: Cleared detection. Level-sensitive mode: Read 0: No interrupt request occurred. Read 1: Interrupt request occurred Write 0: No functional effect. Write 1: No functional effect. |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 6 | NMI6DET | B'0 | R/WC1 | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI6 Event Detection Status for CPU6 interface (INTC-AP) Edge-triggered mode: Read 0: No interrupt request occurred Read 1: Interrupt request occurred Write 0: No functional effect Write 1: Cleared detection. Level-sensitive mode: Read 0: No interrupt request occurred. Read 1: Interrupt request occurred Write 0: No functional effect. Write 1: No functional effect. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 5 | NMI5DET | B'0 | R/WC1 | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI5 Event Detection Status for CPU5 interface (INTC-AP) Edge-triggered mode: Read 0: No interrupt request occurred Read 1: Interrupt request occurred Write 0: No functional effect Write 1: Cleared detection. Level-sensitive mode: Read 0: No interrupt request occurred. Read 1: Interrupt request occurred Write 0: No functional effect. Write 1: No functional effect. |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | NMI4DET | B'0 | R/WC1 | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI4 Event Detection Status for CPU4 interface (INTC-AP) Edge-triggered mode: Read 0: No interrupt request occurred Read 1: Interrupt request occurred Write 0: No functional effect Write 1: Cleared detection. Level-sensitive mode: Read 0: No interrupt request occurred. Read 1: Interrupt request occurred Write 0: No functional effect. Write 1: No functional effect. |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 3 | NMI3DET | B'0 | R/WC1 | [RZ/G2H] NMI3 Event Detection Status for CPU3 interface (INTC-AP) Edge-triggered mode: Read 0: No interrupt request occurred Read 1: Interrupt request occurred Write 0: No functional effect Write 1: Cleared detection. Level-sensitive mode: Read 0: No interrupt request occurred. Read 1: Interrupt request occurred Write 0: No functional effect. Write 1: No functional effect. |
| | | | | [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 2 | NMI2DET | B'0 | R/WC1 | <p>[RZ/G2H]</p> <p>NMI2 Event Detection Status for CPU2 interface (INTC-AP)</p> <p>Edge-triggered mode:</p> <p>Read 0: No interrupt request occurred</p> <p>Read 1: Interrupt request occurred</p> <p>Write 0: No functional effect</p> <p>Write 1: Cleared detection.</p> <p>Level-sensitive mode:</p> <p>Read 0: No interrupt request occurred.</p> <p>Read 1: Interrupt request occurred</p> <p>Write 0: No functional effect.</p> <p>Write 1: No functional effect.</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 1 | NMI1DET | B'0 | R/WC1 | <p>NMI1 Event Detection Status for CPU1 interface (INTC-AP)</p> <p>Edge-triggered mode:</p> <p>Read 0: No interrupt request occurred</p> <p>Read 1: Interrupt request occurred</p> <p>Write 0: No functional effect</p> <p>Write 1: Cleared detection.</p> <p>Level-sensitive mode:</p> <p>Read 0: No interrupt request occurred.</p> <p>Read 1: Interrupt request occurred</p> <p>Write 0: No functional effect.</p> <p>Write 1: No functional effect.</p> |
| 0 | NMI0DET | B'0 | R/WC1 | <p>NMI0 Event Detection Status for CPU0 interface (INTC-AP)</p> <p>Edge-triggered mode:</p> <p>Read 0: No interrupt request occurred</p> <p>Read 1: Interrupt request occurred</p> <p>Write 0: No functional effect</p> <p>Write 1: Cleared detection.</p> <p>Level-sensitive mode:</p> <p>Read 0: No interrupt request occurred.</p> <p>Read 1: Interrupt request occurred</p> <p>Write 0: No functional effect.</p> <p>Write 1: No functional effect.</p> |

20.2.18 NMI Signal Level Monitor Register (MONITOR_NMI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides external signal monitor.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | NMI7 MON | NMI6 MON | NMI5 MON | NMI4 MON | NMI3 MON | NMI2 MON | NMI1 MON | NMI0 MON |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | NMI7MON | — | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI7 External Signal Level Monitor for CPU7 interface (INTC-AP) This function show input value when signal is enabled by corresponding configuration register SS bits. So, initial value depends on input value. 0: NMI7 is low level 1: NMI7 is high level |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 6 | NMI6MON | — | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI6 External Signal Level Monitor for CPU6 interface (INTC-AP) This function show input value when signal is enabled by corresponding configuration register SS bits. So, initial value depends on input value. 0: NMI6 is low level 1: NMI6 is high level |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 5 | NMI5MON | — | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI5 External Signal Level Monitor for CPU5 interface (INTC-AP) This function show input value when signal is enabled by corresponding configuration register SS bits. So, initial value depends on input value. 0: NMI5 is low level 1: NMI5 is high level |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | NMI4MON | — | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI4 External Signal Level Monitor for CPU4 interface (INTC-AP) This function show input value when signal is enabled by corresponding configuration register SS bits. So, initial value depends on input value. 0: NMI4 is low level 1: NMI4 is high level |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 3 | NMI3MON | — | R | [RZ/G2H] NMI3 External Signal Level Monitor for CPU3 interface (INTC-AP) This function show input value when signal is enabled by corresponding configuration register SS bits. So, initial value depends on input value. 0: NMI3 is low level 1: NMI3 is high level |
| | | | | [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | NMI2MON | — | R | [RZ/G2H] NMI2 External Signal Level Monitor for CPU2 interface (INTC-AP) This function show input value when signal is enabled by corresponding configuration register SS bits. So, initial value depends on input value. 0: NMI2 is low level 1: NMI2 is high level |
| | | | | [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 1 | NMI1MON | — | R | <p>NMI1 External Signal Level Monitor for CPU1 interface (INTC-AP)</p> <p>This function show input value when signal is enabled by corresponding configuration register SS bits.</p> <p>So, initial value depends on input value.</p> <p>0: NMI1 is low level</p> <p>1: NMI1 is high level</p> |
| 0 | NMI0MON | — | R | <p>NMI0 External Signal Level Monitor for CPU0 interface (INTC-AP)</p> <p>This function show input value when signal is enabled by corresponding configuration register SS bits.</p> <p>So, initial value depends on input value.</p> <p>0: NMI0 is low level</p> <p>1: NMI0 is high level</p> |

20.2.19 NMI High Level Detect Status Register (HLVL_STS_NMI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides interrupt detail detect status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | NMI7H STS | NMI6H STS | NMI5H STS | NMI4H STS | NMI3H STS | NMI2H STS | NMI1H STS | NMI0H STS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The writing value should always be 0. |
| 7 | NMI7HSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI7 High Level Interrupt Status for CPU7 interface (INTC-AP) 0: NMI7 high level interrupt request not occurred 1: NMI7 high level interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |
| 6 | NMI6HSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI6 High Level Interrupt Status for CPU6 interface (INTC-AP) 0: NMI6 high level interrupt request not occurred 1: NMI6 high level interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 5 | NMI5HSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI5 High Level Interrupt Status for CPU5 interface (INTC-AP) 0: NMI5 high level interrupt request not occurred 1: NMI5 high level interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |
| 4 | NMI4HSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI4 High Level Interrupt Status for CPU4 interface (INTC-AP) 0: NMI4 high level interrupt request not occurred 1: NMI4 high level interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |
| 3 | NMI3HSTS | B'0 | R | [RZ/G2H] NMI3 High Level Interrupt Status for CPU3 interface (INTC-AP) 0: NMI3 high level interrupt request not occurred 1: NMI3 high level interrupt request occurred |
| | | | | [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |
| 2 | NMI2HSTS | B'0 | R | [RZ/G2H] NMI2 High Level Interrupt Status for CPU2 interface (INTC-AP) 0: NMI2 high level interrupt request not occurred 1: NMI2 high level interrupt request occurred |
| | | | | [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |
| 1 | NMI1HSTS | B'0 | R | NMI1 High Level Interrupt Status for CPU1 interface (INTC-AP) 0: NMI1 high level interrupt request not occurred 1: NMI1 high level interrupt request occurred |
| 0 | NMI0HSTS | B'0 | R | NMI0 High Level Interrupt Status for CPU0 interface (INTC-AP) 0: NMI0 high level interrupt request not occurred 1: NMI0 high level interrupt request occurred |

20.2.20 NMI Low Level Detect Status Register (LLVL_STS_NMI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides interrupt detail detect status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | NMI7L STS | NMI6L STS | NMI5L STS | NMI4L STS | NMI3L STS | NMI2L STS | NMI1L STS | NMI0L STS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The writing value should always be 0. |
| 7 | NMI7LSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI7 Low Level Interrupt Status for CPU7 interface (INTC-AP) 0: NMI7 low level interrupt request not occurred 1: NMI7 low level interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |
| 6 | NMI6LSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI6 Low Level Interrupt Status for CPU6 interface (INTC-AP) 0: NMI6 low level interrupt request not occurred 1: NMI6 low level interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 5 | NMI5LSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI5 Low Level Interrupt Status for CPU5 interface (INTC-AP) 0: NMI5 low level interrupt request not occurred 1: NMI5 low level interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |
| 4 | NMI4LSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI4 Low Level Interrupt Status for CPU4 interface (INTC-AP) 0: NMI4 low level interrupt request not occurred 1: NMI4 low level interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |
| 3 | NMI3LSTS | B'0 | R | [RZ/G2H] NMI3 Low Level Interrupt Status for CPU3 interface (INTC-AP) 0: NMI3 low level interrupt request not occurred 1: NMI3 low level interrupt request occurred |
| | | | | [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |
| 2 | NMI2LSTS | B'0 | R | [RZ/G2H] NMI2 Low Level Interrupt Status for CPU2 interface (INTC-AP) 0: NMI2 low level interrupt request not occurred 1: NMI2 low level interrupt request occurred |
| | | | | [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |
| 1 | NMI1LSTS | B'0 | R | NMI1 Low Level Interrupt Status for CPU1 interface (INTC-AP) 0: NMI1 low level interrupt request not occurred 1: NMI1 low level interrupt request occurred |
| 0 | NMI0LSTS | B'0 | R | NMI0 Low Level Interrupt Status for CPU0 interface (INTC-AP) 0: NMI0 low level interrupt request not occurred 1: NMI0 low level interrupt request occurred |

20.2.21 NMI Sync Rising Edge Detect Status Register (S_R_EDGE_STS_NMI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides interrupt detail detect status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | NMI7S RSTS | NMI6S RSTS | NMI5S RSTS | NMI4S RSTS | NMI3S RSTS | NMI2S RSTS | NMI1S RSTS | NMI0S RSTS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The writing value should always be 0. |
| 7 | NMI7SRSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI7 Synchronous Rise Edge Interrupt Status for CPU7 interface (INTC-AP) 0: NMI7 rise edge interrupt request not occurred 1: NMI7 rise edge interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |
| 6 | NMI6SRSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI6 Synchronous Rise Edge Interrupt Status for CPU6 interface (INTC-AP) 0: NMI6 rise edge interrupt request not occurred 1: NMI6 rise edge interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------|---------------|-----|---|
| 5 | NMI5SRSTS | B'0 | R | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>NMI5 Synchronous Rise Edge Interrupt Status for CPU5 interface (INTC-AP)</p> <p>0: NMI5 rise edge interrupt request not occurred</p> <p>1: NMI5 rise edge interrupt request occurred</p> <hr/> <p>[RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The writing value should always be 0.</p> |
| 4 | NMI4SRSTS | B'0 | R | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>NMI4 Synchronous Rise Edge Interrupt Status for CPU4 interface (INTC-AP)</p> <p>0: NMI4 rise edge interrupt request not occurred</p> <p>1: NMI4 rise edge interrupt request occurred</p> <hr/> <p>[RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The writing value should always be 0.</p> |
| 3 | NMI3SRSTS | B'0 | R | <p>[RZ/G2H]</p> <p>NMI3 Synchronous Rise Edge Interrupt Status for CPU3 interface (INTC-AP)</p> <p>0: NMI3 rise edge interrupt request not occurred</p> <p>1: NMI3 rise edge interrupt request occurred</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The writing value should always be 0.</p> |
| 2 | NMI2SRSTS | B'0 | R | <p>[RZ/G2H]</p> <p>NMI2 Synchronous Rise Edge Interrupt Status for CPU2 interface (INTC-AP)</p> <p>0: NMI2 rise edge interrupt request not occurred</p> <p>1: NMI2 rise edge interrupt request occurred</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The writing value should always be 0.</p> |
| 1 | NMI1SRSTS | B'0 | R | <p>NMI1 Synchronous Rise Edge Interrupt Status for CPU1 interface (INTC-AP)</p> <p>0: NMI1 rise edge interrupt request not occurred</p> <p>1: NMI1 rise edge interrupt request occurred</p> |
| 0 | NMI0SRSTS | B'0 | R | <p>NMI0 Synchronous Rise Edge Interrupt Status for CPU0 interface (INTC-AP)</p> <p>0: NMI0 rise edge interrupt request not occurred</p> <p>1: NMI0 rise edge interrupt request occurred</p> |

20.2.22 NMI Sync Falling Edge Detect Status Register (S_F_EDGE_STS_NMI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides interrupt detail detect status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | NMI7S FSTS | NMI6S FSTS | NMI5S FSTS | NMI4S FSTS | NMI3S FSTS | NMI2S FSTS | NMI1S FSTS | NMI0S FSTS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The writing value should always be 0. |
| 7 | NMI7SFSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI7 Synchronous Fall Edge Interrupt Status for CPU7 interface (INTC-AP) 0: NMI7 fall edge interrupt request not occurred 1: NMI7 fall edge interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |
| 6 | NMI6SFSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI6 Synchronous Fall Edge Interrupt Status for CPU6 interface (INTC-AP) 0: NMI6 fall edge interrupt request not occurred 1: NMI6 fall edge interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------|---------------|-----|--|
| 5 | NMI5SFSTS | B'0 | R | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>NMI5 Synchronous Fall Edge Interrupt Status for CPU5 interface (INTC-AP)</p> <p>0: NMI5 fall edge interrupt request not occurred 1: NMI5 fall edge interrupt request occurred</p> <hr/> <p>[RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The writing value should always be 0.</p> |
| 4 | NMI4SFSTS | B'0 | R | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>NMI4 Synchronous Fall Edge Interrupt Status for CPU4 interface (INTC-AP)</p> <p>0: NMI4 fall edge interrupt request not occurred 1: NMI4 fall edge interrupt request occurred</p> <hr/> <p>[RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The writing value should always be 0.</p> |
| 3 | NMI3SFSTS | B'0 | R | <p>[RZ/G2H]</p> <p>NMI3 Synchronous Fall Edge Interrupt Status for CPU3 interface (INTC-AP)</p> <p>0: NMI3 fall edge interrupt request not occurred 1: NMI3 fall edge interrupt request occurred</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The writing value should always be 0.</p> |
| 2 | NMI2SFSTS | B'0 | R | <p>[RZ/G2H]</p> <p>NMI2 Synchronous Fall Edge Interrupt Status for CPU2 interface (INTC-AP)</p> <p>0: NMI2 fall edge interrupt request not occurred 1: NMI2 fall edge interrupt request occurred</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The writing value should always be 0.</p> |
| 1 | NMI1SFSTS | B'0 | R | <p>NMI1 Synchronous Fall Edge Interrupt Status for CPU1 interface (INTC-AP)</p> <p>0: NMI1 fall edge interrupt request not occurred 1: NMI1 fall edge interrupt request occurred</p> |
| 0 | NMI0SFSTS | B'0 | R | <p>NMI0 Synchronous Fall Edge Interrupt Status for CPU0 interface (INTC-AP)</p> <p>0: NMI0 fall edge interrupt request not occurred 1: NMI0 fall edge interrupt request occurred</p> |

20.2.23 NMI Async Rising Edge Detect Status Register (A_R_EDGE_STS_NMI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides interrupt detail detect status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | NMI7A RSTS | NMI6A RSTS | NMI5A RSTS | NMI4A RSTS | NMI3A RSTS | NMI2A RSTS | NMI1A RSTS | NMI0A RSTS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The writing value should always be 0. |
| 7 | NMI7ARSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI7 Asynchronous Rise Edge Interrupt Status for CPU7 interface (INTC-AP) 0: NMI7 rise edge interrupt request not occurred 1: NMI7 rise edge interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |
| 6 | NMI6ARSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI6 Asynchronous Rise Edge Interrupt Status for CPU6 interface (INTC-AP) 0: NMI6 rise edge interrupt request not occurred 1: NMI6 rise edge interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------|---------------|-----|---|
| 5 | NMI5ARSTS | B'0 | R | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>NMI5 Asynchronous Rise Edge Interrupt Status for CPU5 interface (INTC-AP)</p> <p>0: NMI5 rise edge interrupt request not occurred 1: NMI5 rise edge interrupt request occurred</p> <hr/> <p>[RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The writing value should always be 0.</p> |
| 4 | NMI4ARSTS | B'0 | R | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>NMI4 Asynchronous Rise Edge Interrupt Status for CPU4 interface (INTC-AP)</p> <p>0: NMI4 rise edge interrupt request not occurred 1: NMI4 rise edge interrupt request occurred</p> <hr/> <p>[RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The writing value should always be 0.</p> |
| 3 | NMI3ARSTS | B'0 | R | <p>[RZ/G2H]</p> <p>NMI3 Asynchronous Rise Edge Interrupt Status for CPU3 interface (INTC-AP)</p> <p>0: NMI3 rise edge interrupt request not occurred 1: NMI3 rise edge interrupt request occurred</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The writing value should always be 0.</p> |
| 2 | NMI2ARSTS | B'0 | R | <p>[RZ/G2H]</p> <p>NMI2 Asynchronous Rise Edge Interrupt Status for CPU2 interface (INTC-AP)</p> <p>0: NMI2 rise edge interrupt request not occurred 1: NMI2 rise edge interrupt request occurred</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]</p> <p>Reserved</p> <p>These bits are always read as 0. The writing value should always be 0.</p> |
| 1 | NMI1ARSTS | B'0 | R | <p>NMI1 Asynchronous Rise Edge Interrupt Status for CPU1 interface (INTC-AP)</p> <p>0: NMI1 rise edge interrupt request not occurred 1: NMI1 rise edge interrupt request occurred</p> |
| 0 | NMI0ARSTS | B'0 | R | <p>NMI0 Asynchronous Rise Edge Interrupt Status for CPU0 interface (INTC-AP)</p> <p>0: NMI0 rise edge interrupt request not occurred 1: NMI0 rise edge interrupt request occurred</p> |

20.2.24 NMI Async Falling Edge Detect Status Register (A_F_EDGE_STS_NMI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides interrupt detail detect status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | NMI7A FSTS | NMI6A FSTS | NMI5A FSTS | NMI4A FSTS | NMI3A FSTS | NMI2A FSTS | NMI1A FSTS | NMI0A FSTS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The writing value should always be 0. |
| 7 | NMI7AFSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI7 Asynchronous Fall Edge Interrupt Status for CPU7 interface (INTC-AP) 0: NMI7 fall edge interrupt request not occurred 1: NMI7 fall edge interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |
| 6 | NMI6AFSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI6 Asynchronous Fall Edge Interrupt Status for CPU6 interface (INTC-AP) 0: NMI6 fall edge interrupt request not occurred 1: NMI6 fall edge interrupt request occurred |
| | | | | [RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------|---------------|-----|--|
| 5 | NMI5AFSTS | B'0 | R | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI5 Asynchronous Fall Edge Interrupt Status for CPU5 interface (INTC-AP) 0: NMI5 fall edge interrupt request not occurred 1: NMI5 fall edge interrupt request occurred</p> <hr/> <p>[RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0.</p> |
| 4 | NMI4AFSTS | B'0 | R | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0] NMI4 Asynchronous Fall Edge Interrupt Status for CPU4 interface (INTC-AP) 0: NMI4 fall edge interrupt request not occurred 1: NMI4 fall edge interrupt request occurred</p> <hr/> <p>[RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0.</p> |
| 3 | NMI3AFSTS | B'0 | R | <p>[RZ/G2H] NMI3 Asynchronous Fall Edge Interrupt Status for CPU3 interface (INTC-AP) 0: NMI3 fall edge interrupt request not occurred 1: NMI3 fall edge interrupt request occurred</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0.</p> |
| 2 | NMI2AFSTS | B'0 | R | <p>[RZ/G2H] NMI2 Asynchronous Fall Edge Interrupt Status for CPU2 interface (INTC-AP) 0: NMI2 fall edge interrupt request not occurred 1: NMI2 fall edge interrupt request occurred</p> <hr/> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] Reserved These bits are always read as 0. The writing value should always be 0.</p> |
| 1 | NMI1AFSTS | B'0 | R | <p>NMI1 Asynchronous Fall Edge Interrupt Status for CPU1 interface (INTC-AP) 0: NMI1 fall edge interrupt request not occurred 1: NMI1 fall edge interrupt request occurred</p> |
| 0 | NMI0AFSTS | B'0 | R | <p>NMI0 Asynchronous Fall Edge Interrupt Status for CPU0 interface (INTC-AP) 0: NMI0 fall edge interrupt request not occurred 1: NMI0 fall edge interrupt request occurred</p> |

20.2.25 NMI Chattering Reduction Status Register (CHTEN_STS_NMI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register shows chattering reduction enable status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CHTEN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | CHTEN | B'0 | R | Chattering Reduction Enable Status 0: Chattering reduction disabled 1: Chattering reduction enabled |

Note: After you set CHTEN.DEB_SET_NMI bit, this bit shows its status. However, 3 cycles @ RCLK (32 kHz) is necessary to reflect the set value.

20.2.26 NMI Debounce Setting Register (DEB_SET_NMI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides chattering reduction setting.

| | | | | | | | | | | | | | | | | |
|----------------|-------|----|----|----|----|----|----|----|------|-----|------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CHTEN | — | — | — | — | — | — | — | STS1 | | STS2 | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | CHTEN | B'0 | R/W | Chattering Reduction Enable 0: Chattering reduction disabled 1: Chattering reduction enabled |
| 30 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23, 22 | STS1 | B'00 | R/W | NMI Scan Timing These bits provide chattering reduction timing. B'00: 1 ms B'01: 2 ms B'10: 4 ms B'11: 8 ms |
| 21 to 16 | STS2 | H'00 | R/W | NMI Chattering Reduction Period The chattering reduction period is defined by STS1 × STS2. H'00: No reduction H'01 to H'3F: STS1 × STS2 Note 1. These bits should not be set H'00 when chattering reduction enable. 2. Check chattering reduction is disabled completely with bit of chattering reduction status register (CHTEN_STS) when this bit change. |
| 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The writing value should always be 0. |

20.2.27 NMI Configuration n Register (CONFIGn_NMI)

- Note: n = 0 CPU0 interface (INTC-AP)
 n = 1 CPU1 interface (INTC-AP)
 n = 2 CPU2 interface (INTC-AP)
 n = 3 CPU3 interface (INTC-AP)
 n = 4 CPU4 interface (INTC-AP)
 n = 5 CPU5 interface (INTC-AP)
 n = 6 CPU6 interface (INTC-AP)
 n = 7 CPU7 interface (INTC-AP)

For the CPU_n interface, see section 19.INTC-AP Register Configuration and Function Description.

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides detection mode and noise reduction setting.

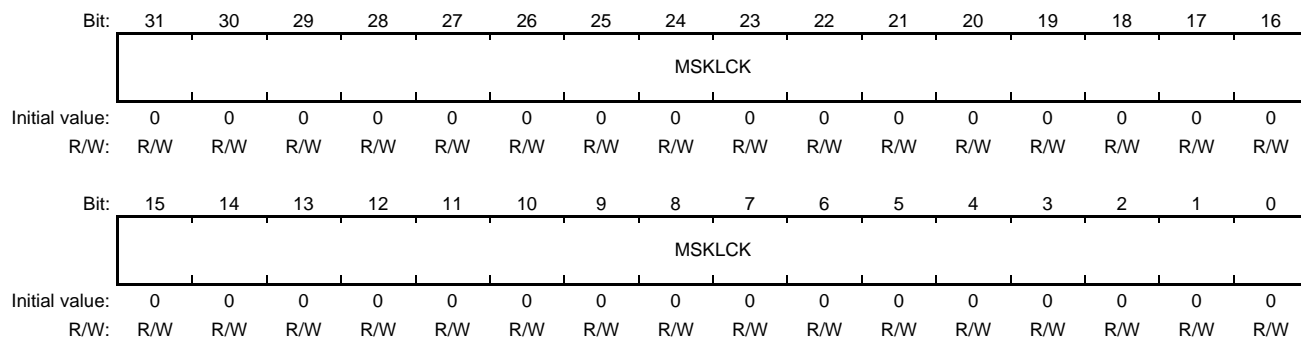
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | SS | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | SS | H'00 | R/W | Sense Selection B'00_0000: Disable event detection B'00_0001: Enable low level sensitive B'00_0010: Enable high level sensitive B'00_0100: Enable synchronous falling edge triggered B'00_1000: Enable synchronous rising edge triggered B'00_1100: Enable synchronous both edge triggered B'01_0000: Enable asynchronous falling edge triggered B'10_0000: Enable asynchronous rising edge triggered B'11_0000: Enable asynchronous both edge triggered Others: setting prohibit Note: The Asynchronous edge triggered can be used only following conditions. - Chattering reduction not be used. - Only low level sensitive can be used for Arm. |

20.2.28 NMI Mask Lock Set Register (NMI_LCK)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides NMI mask locking feature. When set the same value as NMI_LCKCODE, then NMI cannot be masked by software (always accept NMI interrupt).

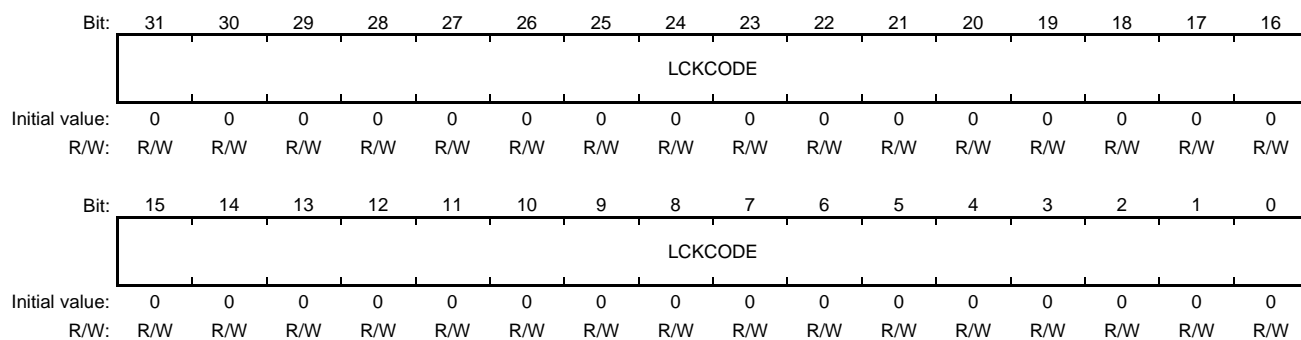


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|-------------------|
| 31 to 0 | MSKLCK | All 0 | R/W | Lock code setting |

20.2.29 NMI Lock Code Register (NMI_LCKCODE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register sets the value for lock code of NMI mask.

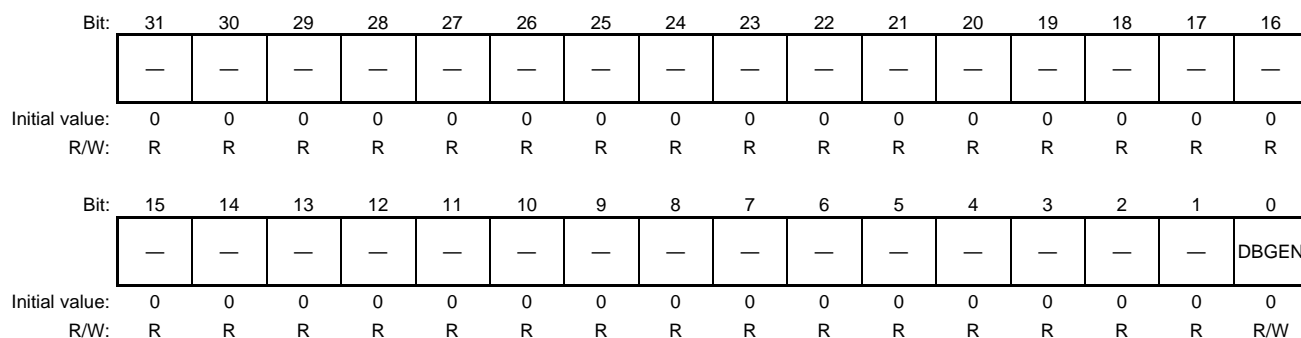


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|----------------------------|
| 31 to 0 | LCKCODE | H'0000_0000 | R/W | NMI mask lock code setting |

20.2.30 NMI Debug Control Enable Register (NMI_DBG)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register enables the debug feature for NMI mask lock.



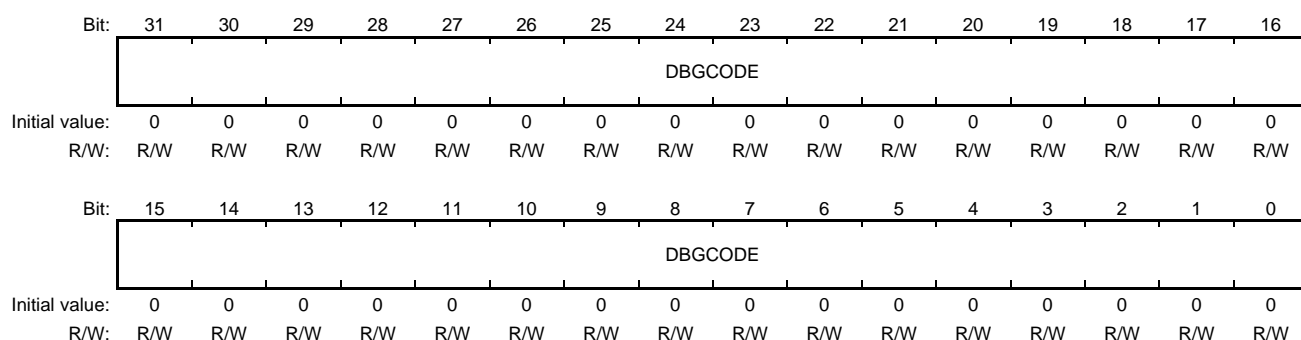
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved |
| 0 | DBGEN | B'0 | R/W | Enable debug for NMI mask lock feature |

20.2.31 NMI Debug Code Register (NMI_DBGCODE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register sets value for debug code of NMI mask lock. When set the following value, then unlock this mask feature.

$DBGCODE[31:0] = \sim(NMI_LCKCODE.LCKCODE[31:0])$ (bit reversed)



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|----------------------------------|
| 31 to 0 | DBGCODE | H'0000_0000 | R/W | NMI mask lock debug code setting |

20.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

20.3.1 NMI Mask Lock Feature

NMI can be masked by setting NMI Mask Lock Set Register. If you set the lock code to NMI lock code register and set the same lock code to NMI lock set register, then NMI cannot be masked by software (locked NMI mask). If you want to unlock NMI mask, you need to set DBGEN bit of NMI debug control register and set the unlock code to the NMI debug code register.

20.3.2 Procedure of NMI Mask Lock

- Lock feature
 1. Write the lock code to NMI lock code register
ex) Write32 (INTC-EX_BASE + H'0A04), H'00A_CCE55
 2. Write the lock code to NMI mask lock set register
ex) Write32 (INTC-EX_BASE + H'0A00), H'00AC_CE55
 3. Write access to NMI lock code register and NMI mask lock set register
— Check whether these registers cannot be updated
- Unlock feature
 1. Set enable bit to NMI debug control register
ex) Write32 (INTC-EX_BASE + H'0A08), H'0000_0001
 2. Write unlock code to NMI debug code register
ex) Write32 (INTC-EX_BASE + H'0A0C), H'558C_CAFF
 3. Write access to NMI lock code register and NMI mask lock register
— Check whether these registers can be updated

21. AXI-bus

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

21.1 Overview

AXI-bus is the On-Chip interconnect bus based on the AMBA® Advanced extensible Interface (AXI) Protocol Specification. AXI-bus has the hierarchical structure and there are the following domains.

- 3DGE (3D-Graphic) domain AXI
- VP (Video Processor) domain AXI (RZ/G2H, RZ/G2N, RZ/G2E)
- VC (Video Codec) domain AXI
- VIO (Video IO) domain AXI
- Peripheral (PeriE/PeriW) domain AXI
- Audio domain AXI
- HC (High Communication) domain AXI
- CPU domain AXI
- Main Memory domain AXI
- Slave Access Bus

21.1.1 Features

- Bus protocol: AXI3 with QoS Control
- Frequency: 266MHz/400MHz/533MHz
- Bus width: 64bits/128bits/256bits/512bits
- Quality-of-Service (QoS)
- Cache snoop function for CPU (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0)
- Provide flexible SDRAM address mapping. Linear mapping allocates a continuous address space to every single memory controller. Split mapping provides an interleaved address space to four memory controllers at 4KB granularity.
- Decompression of visual near lossless compressed image
- Access protection for secure regions
- Low Power function

21.1.2 Block Diagram

Figure 21.1 to Figure 21.4 are block diagrams of the AXI-bus.

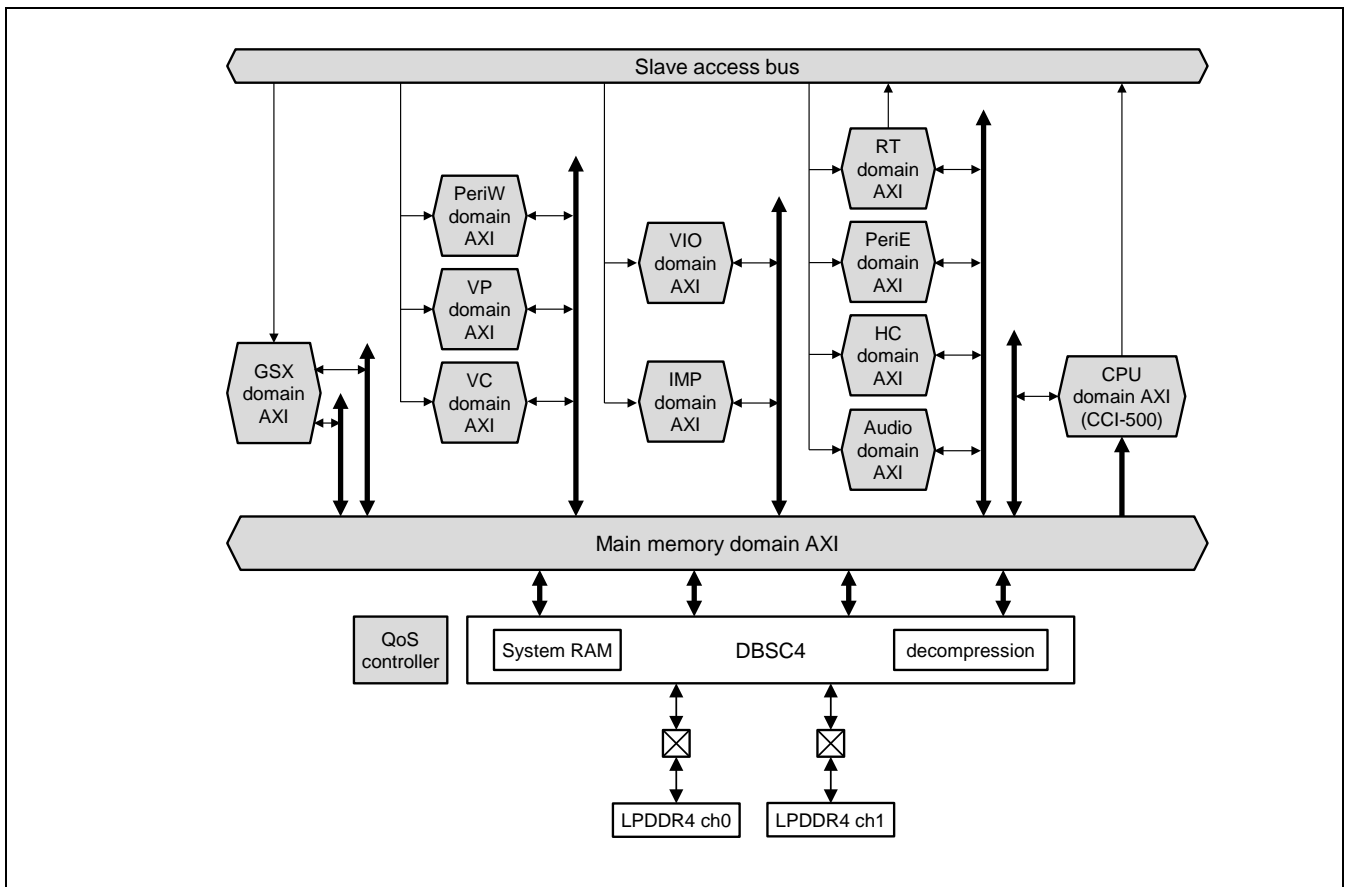


Figure 21.1 Block diagram of AXI-bus [RZ/G2H]

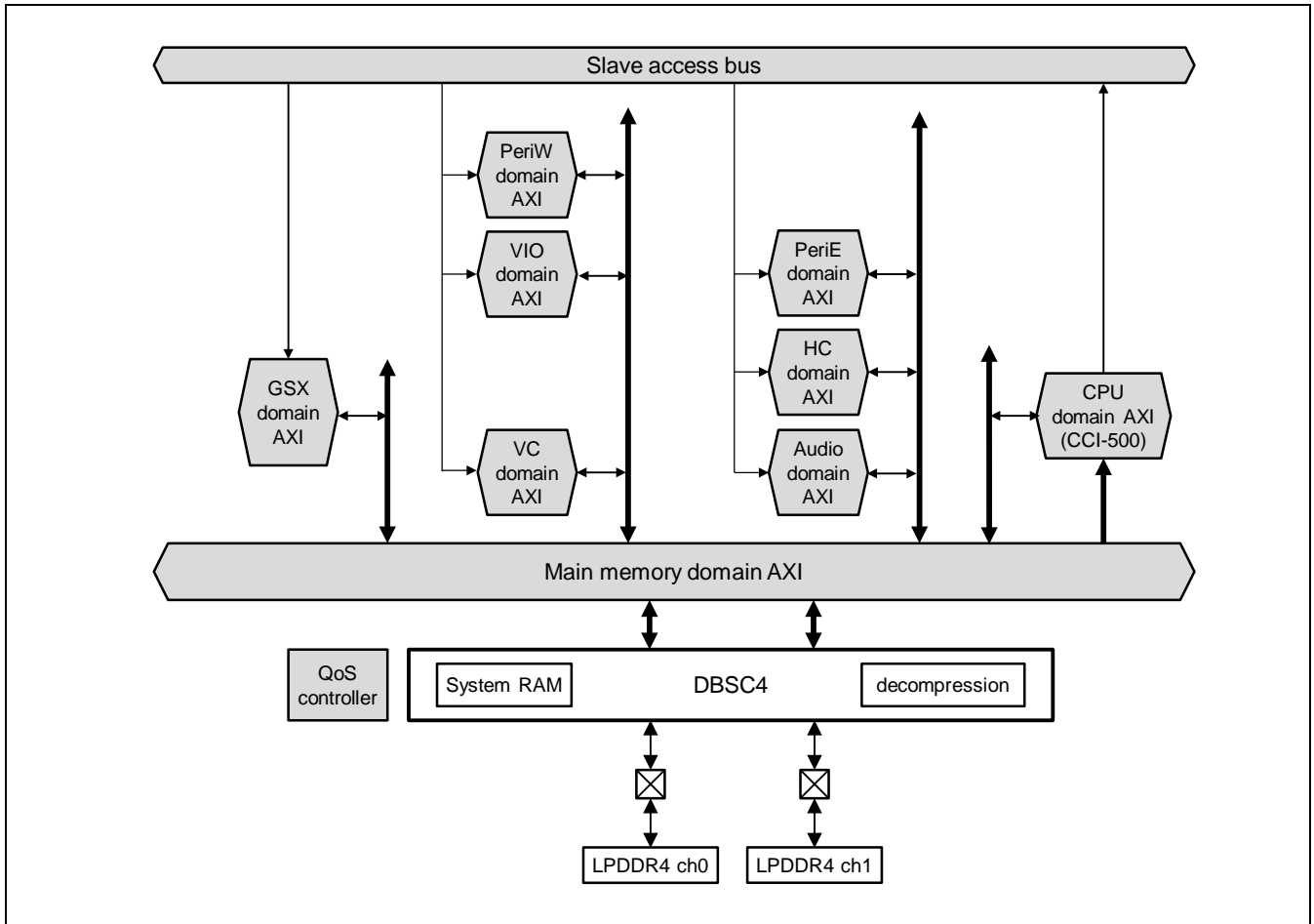


Figure 21.2 Block diagram of AXI-bus [RZ/G2M V1.3, RZ/G2M V3.0]

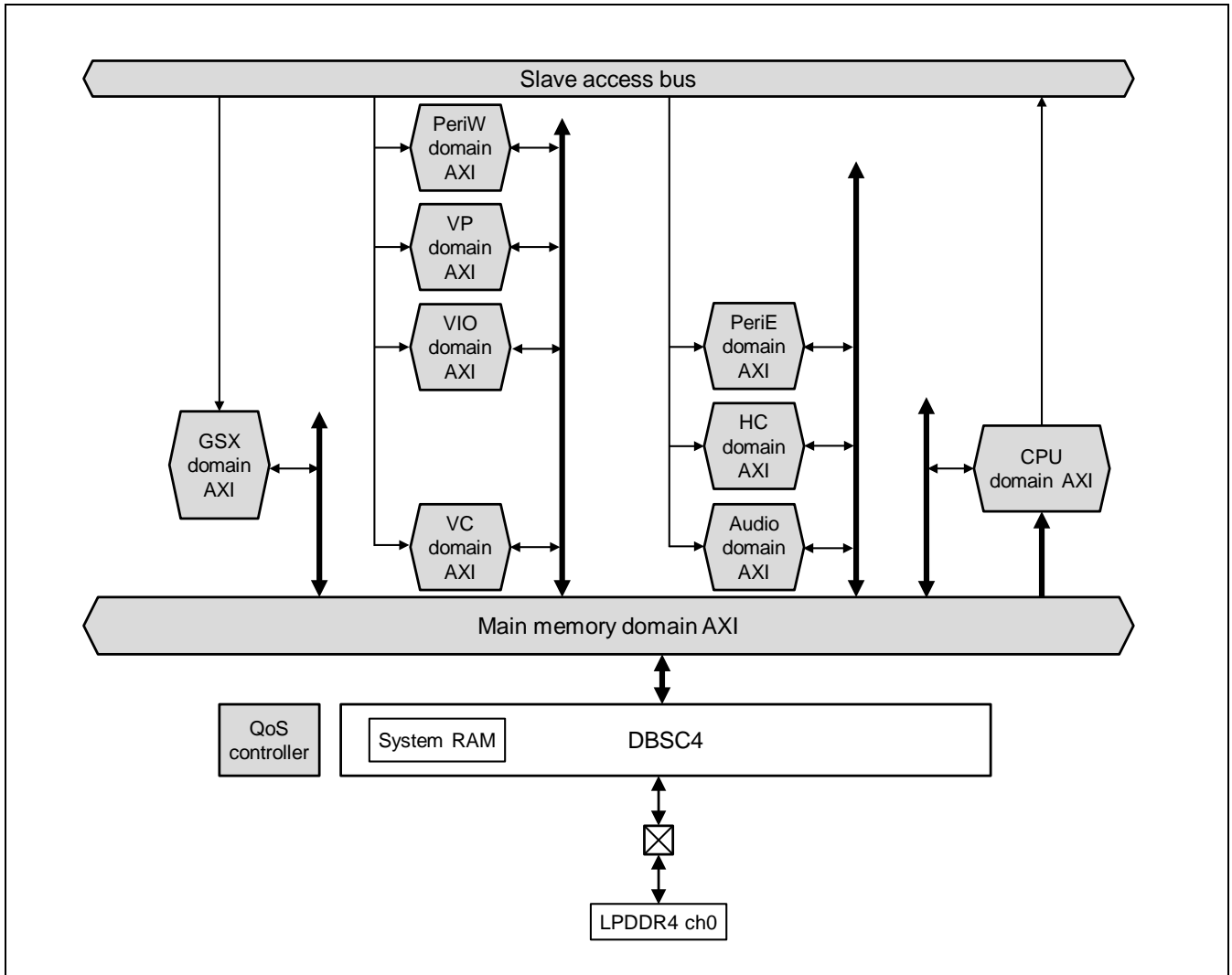


Figure 21.3 Block diagram of AXI-bus [RZ/G2N]

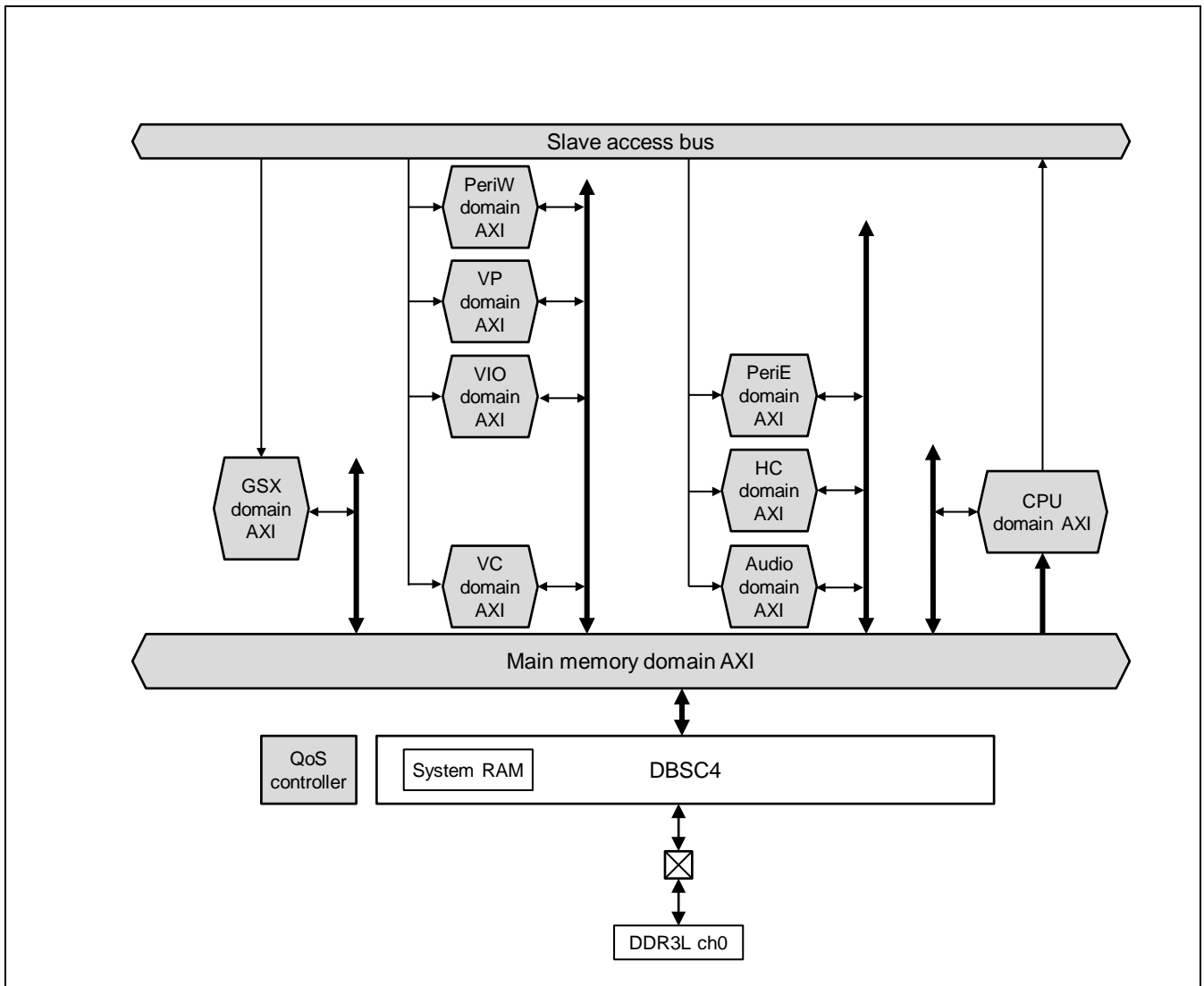


Figure 21.4 Block diagram of AXI-bus [RZ/G2E]

21.1.3 External Pins

No external pins are supported.

21.1.4 Register Configuration

Table 21.1 Register Configurations

| Name | Abbreviation | R/W | Address | Initial value | Size | Second Generation RZ/G Series Products | | | | |
|--|---------------|-----|---------------------|---------------|------|---|-------------|-------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Address Split Control Register0 | ADSPLCR0 | R/W | H'E678_4008 | H'0000_0000 | 32 | √ | √ | √ | — | — |
| Address Split Control Register1 | ADSPLCR1 | R/W | H'E678_400C | H'0000_0000 | 32 | √ | √ | √ | — | — |
| Address Split Control Register2 | ADSPLCR2 | R/W | H'E678_4010 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| Address Split Control Register3 | ADSPLCR3 | R/W | H'E678_4014 | H'0000_0000 | 32 | √ | √ | √ | — | — |
| Decompression Area Configuration Register An (n = 0-15) | DCMPAREACRAn | R/W | H'E678_4100 + 0x8*n | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Decompression Area Configuration Register Bn (n = 0-15) | DCMPAREACRBn | R/W | H'E678_4104 + 0x8*n | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Decompression Shadow Area Configuration Register | DCMPSHDWCR | R/W | H'E678_4280 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Main Memory AXI Control Register | MMCR | R/W | H'E678_4300 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| DRAM Protected Area Division Register n (n = 0-14) | DPTDIVCRn | R/W | H'E678_4400 + 0x4*n | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| DRAM Protected Area Setting Register n (n = 0-15) | DPTCRn | R/W | H'E678_4440 + 0x4*n | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| System RAM Protected Area Division Register n (n = 0-14) | SPTDIVCRn | R/W | H'E678_4500 + 0x4*n | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| System RAM Protected Area Setting Register n (n = 0-15) | SPTCRn | R/W | H'E678_4540 + 0x4*n | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| TR3 Control Register | TR3CR | R/W | H'E67D_100C | H'0000_0000 | 32 | √ | √ | √ | — | — |
| TR4 Control Register | TR4CR | R/W | H'E67D_1014 | H'0000_0000 | 32 | √ | — | — | — | — |
| HC domain Secure Error Upper Address Register | SECADDRU_HC | R | H'E658_0514 | — | 32 | √ | √ | √ | √ | √ |
| HC domain Secure Error Lower Address Register | SECADDRL_HC | R | H'E658_0518 | — | 32 | √ | √ | √ | √ | √ |
| HC domain Secure Error Register | SECERR_HC | R/W | H'E658_051C | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| PeriW domain Secure Error Upper Address Register DM | SECADDRU_PWDM | R | H'E66F_0504 | — | 32 | √ | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial value | Size | Second Generation RZ/G Series Products | | | | |
|---|---------------|-----|-------------|---------------|------|---|-------------|-------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| PeriW domain Secure Error Lower Address Register DM | SECADDRL_PWDM | R | H'E66F_0508 | — | 32 | √ | √ | √ | √ | √ |
| PeriW domain Secure Error Register DM0 | SECERR_PWDM0 | R/W | H'E66F_050C | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| PeriW domain Secure Error Upper Address Register 0 | SECADDRU_PW0 | R | H'E675_0084 | — | 32 | √ | √ | √ | — | — |
| | | | H'E675_0610 | | | — | — | — | — | — |
| | | | H'E675_0514 | | | — | — | — | √ | √ |
| PeriW domain Secure Error Lower Address Register 0 | SECADDRL_PW0 | R | H'E675_0080 | — | 32 | √ | √ | √ | — | — |
| | | | H'E675_0614 | | | — | — | — | — | — |
| | | | H'E675_0518 | | | — | — | — | √ | √ |
| PeriW domain Secure Error Register 00 | SECERR_PW00 | R/W | H'E675_0088 | H'0000_0000 | 32 | √ | √ | √ | — | — |
| | | | H'E675_0600 | | | — | — | — | — | — |
| | | | H'E675_0500 | | | — | — | — | √ | √ |
| PeriW domain Secure Error Register 01 | SECERR_PW01 | R/W | H'E675_0604 | H'0000_0000 | 32 | — | — | — | — | — |
| | | | H'E675_0504 | | | — | — | — | √ | √ |
| PeriW domain Secure Error Register 02 | SECERR_PW02 | R/W | H'E675_0608 | H'0000_0000 | 32 | — | — | — | — | — |
| | | | H'E675_0508 | | | — | — | — | √ | √ |
| PeriW domain Secure Error Register 03 | SECERR_PW03 | R/W | H'E675_050C | H'0000_0000 | 32 | — | — | — | √ | √ |
| PeriW domain Secure Error Upper Address Register 1 | SECADDRU_PW1 | R | H'E675_0614 | — | 32 | √ | √ | √ | — | — |
| PeriW domain Secure Error Lower Address Register 1 | SECADDRL_PW1 | R | H'E675_0514 | — | 32 | √ | √ | √ | — | — |
| PeriW domain Secure Error Register 10 | SECERR_PW10 | R/W | H'E675_0714 | H'0000_0000 | 32 | √ | √ | √ | — | — |
| PeriW domain Secure Error Upper Address Register 2 | SECADDRU_PW2 | R | H'E675_061C | — | 32 | √ | √ | √ | — | — |
| PeriW domain Secure Error Lower Address Register 2 | SECADDRL_PW2 | R | H'E675_051C | — | 32 | √ | √ | √ | — | — |
| PeriW domain Secure Error Register 20 | SECERR_PW20 | R/W | H'E675_071C | H'0000_0000 | 32 | √ | √ | √ | — | — |
| PeriW domain Secure Error Upper Address Register 3 | SECADDRU_PW3 | R | H'E675_0624 | — | 32 | √ | √ | √ | — | — |
| PeriW domain Secure Error Lower Address Register 3 | SECADDRL_PW3 | R | H'E675_0524 | — | 32 | √ | √ | √ | — | — |
| PeriW domain Secure Error Register 30 | SECERR_PW30 | R/W | H'E675_0724 | H'0000_0000 | 32 | √ | √ | √ | — | — |
| PeriE domain Secure Error Upper Address Register DM | SECADDRU_PEDM | R | H'E778_0504 | — | 32 | √ | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial value | Size | Second Generation RZ/G Series Products | | | | |
|---|---------------|-----|-------------|---------------|------|---|-------------|-------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| PeriE domain Secure Error Lower Address Register DM | SECADDRL_PEDM | R | H'E778_0508 | — | 32 | √ | √ | √ | √ | √ |
| PeriE domain Secure Error Register DM0 | SECERR_PEDM0 | R/W | H'E778_050C | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| PeriE domain Secure Error Upper Address Register 0 | SECADDRU_PE0 | R | H'E775_00AC | — | 32 | √ | √ | √ | √ | √ |
| PeriE domain Secure Error Lower Address Register 0 | SECADDRL_PE0 | R | H'E775_00A8 | — | 32 | √ | √ | √ | √ | √ |
| PeriE domain Secure Error Register 00 | SECERR_PE00 | R/W | H'E775_00B0 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| PeriE domain Secure Error Register 01 | SECERR_PE01 | R/W | H'E775_00D4 | H'0000_0000 | 32 | — | — | — | √ | √ |
| PeriE domain Secure Error Register 02 | SECERR_PE02 | R/W | H'E775_00DC | H'0000_0000 | 32 | — | — | — | √ | √ |
| PeriE domain Secure Error Register 03 | SECERR_PE03 | R/W | H'E775_00E4 | H'0000_0000 | 32 | — | — | — | √ | √ |
| PeriE domain Secure Error Register 04 | SECERR_PE04 | R/W | H'E775_00EC | H'0000_0000 | 32 | — | — | — | √ | √ |
| PeriE domain Secure Error Upper Address Register 1 | SECADDRU_PE1 | R | H'E775_0608 | — | 32 | √ | √ | √ | — | — |
| PeriE domain Secure Error Lower Address Register 1 | SECADDRL_PE1 | R | H'E775_0508 | — | 32 | √ | √ | √ | — | — |
| PeriE domain Secure Error Register 10 | SECERR_PE10 | R/W | H'E775_0708 | H'0000_0000 | 32 | √ | √ | √ | — | — |
| PeriE domain Secure Error Upper Address Register 2 | SECADDRU_PE2 | R | H'E775_0610 | — | 32 | √ | √ | √ | — | — |
| PeriE domain Secure Error Lower Address Register 2 | SECADDRL_PE2 | R | H'E775_0510 | — | 32 | √ | √ | √ | — | — |
| PeriE domain Secure Error Register 20 | SECERR_PE20 | R/W | H'E775_0710 | H'0000_0000 | 32 | √ | √ | √ | — | — |
| PeriE domain Secure Error Upper Address Register 3 | SECADDRU_PE3 | R | H'E775_0618 | — | 32 | √ | √ | √ | — | — |
| PeriE domain Secure Error Lower Address Register 3 | SECADDRL_PE3 | R | H'E775_0518 | — | 32 | √ | √ | √ | — | — |
| PeriE domain Secure Error Register 30 | SECERR_PE30 | R/W | H'E775_0718 | H'0000_0000 | 32 | √ | √ | √ | — | — |
| PeriE domain Secure Error Upper Address Register 4 | SECADDRU_PE4 | R | H'E775_0620 | — | 32 | √ | √ | √ | — | — |
| PeriE domain Secure Error Lower Address Register 4 | SECADDRL_PE4 | R | H'E775_0520 | — | 32 | √ | √ | √ | — | — |
| PeriE domain Secure Error Register 40 | SECERR_PE40 | R/W | H'E775_0720 | H'0000_0000 | 32 | √ | √ | √ | — | — |

| Name | Abbreviation | R/W | Address | Initial value | Size | Second Generation RZ/G Series Products | | | | |
|--|---------------|-----|-------------|---------------|------|---|-------------|-------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| MP domain Secure Error Upper Address Register | SECADDRU_MP | R | H'EC60_0134 | — | 32 | √ | √ | √ | √ | √ |
| MP domain Secure Error Lower Address Register | SECADDRL_MP | R | H'EC60_0138 | — | 32 | √ | √ | √ | √ | √ |
| MP domain Secure Error Register 0 | SECERR_MP0 | R/W | H'EC60_0120 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| MP domain Secure Error Register 1 | SECERR_MP1 | R/W | H'EC60_0124 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| MP domain Secure Error Register 2 | SECERR_MP2 | R/W | H'EC60_0128 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| MP domain Secure Error Upper Address Register DM | SECADDRU_MPDM | R | H'EC61_0504 | — | 32 | √ | √ | √ | √ | √ |
| MP domain Secure Error Lower Address Register DM | SECADDRL_MPDM | R | H'EC61_0508 | — | 32 | √ | √ | √ | √ | √ |
| MP domain Secure Error Register DM0 | SECERR_MPDM0 | R/W | H'EC61_050C | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| MP domain Secure Error Register DM1 | SECERR_MPDM1 | R/W | H'EC61_0510 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| MP domain Secure Error Register DM2 | SECERR_MPDM2 | R/W | H'EC61_0514 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| GSX domain Secure Error Upper Address Register | SECADDRU_GSX | R | H'FD81_0514 | — | 32 | √ | √ | √ | √ | √ |
| GSX domain Secure Error Lower Address Register | SECADDRL_GSX | R | H'FD81_0518 | — | 32 | √ | √ | √ | √ | √ |
| GSX domain Secure Error Register 0 | SECERR_GSX0 | R/W | H'FD81_051C | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Main Memory domain Secure Error Upper Address Register | SECADDRU_MM | R | H'E67C_0118 | — | 32 | √ | √ | √ | √ | √ |
| Main Memory domain Secure Error Lower Address Register | SECADDRL_MM | R | H'E67C_011C | — | 32 | √ | √ | √ | √ | √ |
| Main Memory domain Secure Error Register 0 | SECERR_MM0 | R/W | H'E67C_0100 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| VC domain Secure Error Upper Address Register | SECADDRU_VC | R | H'FE68_5008 | — | 32 | √ | √ | √ | √ | √ |
| VC domain Secure Error Lower Address Register | SECADDRL_VC | R | H'FE68_500C | — | 32 | √ | √ | √ | √ | √ |
| VC domain Secure Error Register 0 | SECERR_VC0 | R/W | H'FE68_5000 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| VI domain Secure Error Upper Address Register | SECADDRU_VI | R | H'FEBF_500C | — | 32 | √ | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial value | Size | Second Generation RZ/G Series Products | | | | |
|--|-------------------|-----|-------------|---------------|------|---|-------------|-------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| VI domain Secure Error Lower Address Register | SECADDRL_VI | R | H'FEBF_5010 | — | 32 | √ | √ | √ | √ | √ |
| VI domain Secure Error Register 0 | SECERR_VI0 | R/W | H'FEBF_5000 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| VI domain Secure Error Register 1 | SECERR_VI1 | R/W | H'FEBF_5004 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| VP domain Secure Error Upper Address Register | SECADDRU_VP | R | H'FE9D_5008 | — | 32 | √ | — | — | √ | √ |
| VP domain Secure Error Lower Address Register | SECADDRL_VP | R | H'FE9D_500C | — | 32 | √ | — | — | √ | √ |
| VP domain Secure Error Register 0 | SECERR_VP0 | R/W | H'FE9D_5000 | H'0000_0000 | 32 | √ | — | — | √ | √ |
| Slave Access Bus Secure Error Upper Address Register | SECADDRU_SLV | R | H'FF81_0608 | — | 32 | √ | √ | √ | √ | √ |
| Slave Access Bus Secure Error Lower Address Register | SECADDRL_SLV | R | H'FF81_0508 | — | 32 | √ | √ | √ | √ | √ |
| Slave Access Bus Secure Error Register 0 | SECERR_SLV0 | R/W | H'FF81_0708 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| QOSCTRL SL INIT Register | QOSCTRL_SL_INIT | R/W | H'E67E_8000 | H'0307_00FA | 32 | √ | √ | √ | √ | √ |
| QOSCTRL REF ARS Register | QOSCTRL_REF_ARS | R/W | H'E67E_8004 | H'0064_0000 | 32 | √ | √ | √ | √ | √ |
| QOSCTRL REF ENBL Register | QOSCTRL_REF_ENBL | R/W | H'E67E_8044 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| QOSCTRL STATQC Register | QOSCTRL_STATQC | R/W | H'E67E_8008 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| QOSCTRL MEMBANK Register | QOSCTRL_MEMBANK | R/W | H'E67E_800C | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| QOSCTRL RAS Register | QOSCTRL_RAS | R/W | H'E67F_0000 | H'0000_0020 | 32 | √ | √ | √ | √ | √ |
| QOSCTRL DANT Register | QOSCTRL_DANT | R/W | H'E67F_0038 | H'0020_1008 | 32 | √ | √ | √ | √ | √ |
| QOSCTRL DANN LOW Register | QOSCTRL_DANN_LOW | R/W | H'E67F_0030 | H'0402_0201 | 32 | √ | √ | √ | √ | √ |
| QOSCTRL DANN HIGH Register | QOSCTRL_DANN_HIGH | R/W | H'E67F_0034 | H'0402_0201 | 32 | √ | √ | √ | √ | √ |
| QOSCTRL EMS LOW Register | QOSCTRL_EMS_LOW | R/W | H'E67F_0040 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| QOSCTRL EMS HIGH Register | QOSCTRL_EMS_HIGH | R/W | H'E67F_0044 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| QOSCTRL EARLYR Register | QOSCTRL_EARLYR | R/W | H'E67F_0060 | H'0000_0000 | 32 | √ | — | — | √ | √ |
| QOSCTRL RACNT0 Register | QOSCTRL_RACNT0 | R/W | H'E67F_0080 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial value | Size | Second Generation RZ/G Series Products | | | | |
|--|-------------------------|-----------|---------------------------------|---------------|------|---|-------------|-------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| QOSCTRL INFC Register | QOSCTRL_ INFC | R/W | H'E67F_0050 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| QOSCTRL RAEN Register | QOSCTRL_RAEN | R/W | H'E67F_0018 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| QOSCTRL STATGEN0 Register | QOSCTRL_ STATGEN0 | R/W | H'E67F_0088 | H'0000_0000 | 32 | √ | — | √ | — | — |
| QOSWT WTEN Register | QOSWT_WTEN | R/W | H'E67E_8030 | H'0000_0000 | 32 | √ | √ | √ | √ | — |
| QOSWT WTREF Register | QOSWT_WTREF | R/W | H'E67E_8034 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| QOSWT WTSET0 Register | QOSWT_ WTSET0 | R/W | H'E67E_8038 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| QOSWT WTSET1 Register | QOSWT_ WTSET1 | R/W | H'E67E_803C | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| QOSBW FIX QOS BANK0 Register | QOSBW_FIX_ QOS_BANK0 | R/W | H'E67E_0000 ~ H'E67E_0FFF | — | 64 | √ | √ | √ | √ | √ |
| QOSBW FIX QOS BANK1 Register | QOSBW_FIX_ QOS_BANK1 | R/W | H'E67E_1000 ~ H'E67E_1FFF | — | 64 | √ | √ | √ | √ | √ |
| QOSBW BE QOS BANK0 Register | QOSBW_BE_ QOS_BANK0 | R/W | H'E67E_2000 ~ H'E67E_2FFF | — | 64 | √ | √ | √ | √ | √ |
| QOSBW BE QOS BANK1 Register | QOSBW_BE_ QOS_BANK1 | R/W | H'E67E_3000 ~ H'E67E_3FFF | — | 64 | √ | √ | √ | √ | √ |
| CPU Domain Activate Register 0 | CPU_ACT0 | R/W | 0xF130_0800 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| CPU Domain Activate Register 1 | CPU_ACT1 | R/W | 0xF134_0800 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| CPU Domain Activate Register 2 | CPU_ACT2 | R/W | 0xF138_0800 | H'0000_0000 | 32 | √ | √ | √ | — | — |
| CPU Domain Activate Register 3 | CPU_ACT3 | R/W | 0xF13C_0800 | H'0000_0000 | 32 | √ | √ | √ | — | — |
| Main Memory Secure Error Status Register | MMSECERRST | R/WC 1 | H'E678_4600 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Main Memory Secure Error Address Register0 | MMSECERRADD R0 | R | H'E678_4614 | H'0000_0000 | 32 | √ | √ | √ | √ | √ |
| Main Memory Secure Error Address Register1 | MMSECERRADD R1 | R | H'E678_461C | H'0000_0000 | 32 | √ | √ | √ | √ | √ |

21.2 Register Description

21.2.1 Address Split Control Register0 (ADSPLCR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | — |

| | | | | | | | | | | | | | | | | |
|----------------|-------------|-----|-----|-----------|-----|-----|-----|-----|---------------|-----|----------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | ADRM ODE | — | — | — | — | — | — | — | SPLITSEL[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | AREA[4:0] | | | | — | — | — | SWP[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | ADRMODE | B'0 | R/W | Select address mapping mode 0: default mode(Gen2) 1: Gen1 mode Set 0 on this bit. |
| 30 to 24 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | SPLITSEL | H'00 | R/W | Specify the areas in 4ch split mode translated through SWP. Each bit of SPLITSEL represents the number of area (0-7) Set below value on these bits: H'00 |
| 15 to 13 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 8 | AREA | H'00 | R/W | Indicate the address bit which divides DBSC address space into 8 areas. Set below value on these bits: RZ/G2H: H'1C RZ/G2M V1.3: H'00 RZ/G2M V3.0: H'00 |
| 7 to 5 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | SWP | H'00 | R/W | Indicate the address bit to interleave DBSCs in 4ch split mode. Set below value on these bits: RZ/G2H: H'00 RZ/G2M V1.3: H'00 RZ/G2M V3.0: H'00 |

21.2.2 Address Split Control Register1 (ADSPLCR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | — |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----------|-----|-----|-----|-----|---------------|-----|----------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | SPLITSEL[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | AREA[4:0] | | | | — | — | — | SWP[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 24 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | SPLITSEL | H'00 | R/W | Specify the areas in 2ch split mode translated through SWP. Each bit of SPLITSEL represents the number of area (0-7) Set H'FF on these bits. |
| 15 to 13 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 8 | AREA | H'00 | R/W | Indicate the address bit which divides DBSC address space into 8 areas. Set H'1C on these bits. |
| 7 to 5 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | SWP | H'00 | R/W | Indicate the address bit to interleave DBSCs in 2ch split mode. Set H'0C on these bits. |

21.2.3 Address Split Control Register2 (ADSPLCR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | | |
|----------------|-----|-----------|-----|-----|-----|-----------|-----|-----|--------------|-----------|-----|-----|--------------|-----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | B13EO REN | B13EORSEL | | | B12EO REN | B12EORSEL | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | B15SWPSEL | | | — | B14SWPSEL | | | — | B13SWPSEL | | | — | B12SWPSEL | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 24 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 | B13EOREN | B'0 | R/W | Address bit13 EOR Enable 0: Disable 1: Enable Set 0 on this bit. |
| 22 to 20 | B13EORSEL | B'000 | R/W | Specify the EOR address bit to bit13 B'000: bit15 B'001: bit16 B'010: bit17 Others: Reserved Set B'000 on these bits. |
| 19 | B12EOREN | B'0 | R/W | Address bit12 EOR Enable 0: Disable 1: Enable Set 0 on this bit. |
| 18 to 16 | B12EORSEL | B'000 | R/W | Specify the EOR address bit to bit12 B'000: bit15 B'001: bit16 B'010: bit17 Others: Reserved Set B'000 on these bits. |
| 15 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 14 to 12 | B15SWPSEL | B'000 | R/W | Specify the swap address bit to bit15 B'000: bit15 B'001: bit12 B'010: bit13 B'011: bit14 B'100: bit15 B'101: bit16 B'110: bit17 Set below value on these bits: RZ/G2N: B'000 Others: B'001 |
| 11 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 to 8 | B14SWPSEL | B'000 | R/W | Specify the swap address bit to bit14 B'000: bit14 B'001: bit12 B'010: bit13 B'011: bit14 B'100: bit15 B'101: bit16 B'110: bit17 Set B'000 on these bits. |
| 7 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 to 4 | B13SWPSEL | B'000 | R/W | Specify the swap address bit to bit13 B'000: bit13 B'001: bit12 B'010: bit13 B'011: bit14 B'100: bit15 B'101: bit16 B'110: bit17 Set B'000 on these bits. |
| 3 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 to 0 | B12SWPSEL | B'000 | R/W | Specify the swap address bit to bit12 B'000: bit12 B'001: bit12 B'010: bit13 B'011: bit14 B'100: bit15 B'101: bit16 B'110: bit17 Set below value on these bits: RZ/G2N: B'000 Others: B'100 |

21.2.4 Address Split Control Register3 (ADSPLCR3)

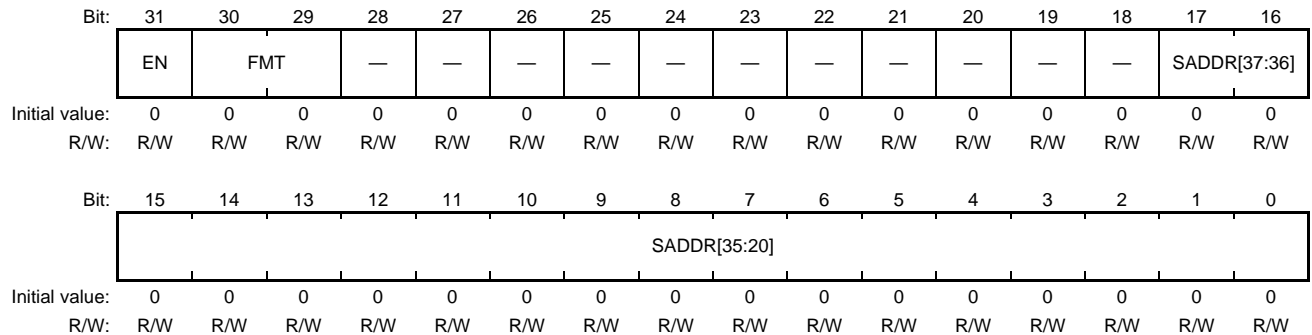
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | — |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----------|-----|-----|-----|-----|---------------|-----|----------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | SPLITSEL[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | AREA[4:0] | | | | — | — | — | SWP[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 24 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | SPLITSEL | H'00 | R/W | Specify the areas in Extra split mode translated through SWP. Each bit of SPLITSEL represents the number of area (0-7) Set H'00 on these bits |
| 15 to 13 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 8 | AREA | H'00 | R/W | Indicate the address bit which divides DBSC address space into 8 areas. Set H'00 on these bits. |
| 7 to 5 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | SWP | H'00 | R/W | Indicate the address bit to interleave DBSCs in Extra split mode. Set H'00 on these bits. |

21.2.5 Decompression Area Configuration Register An (n = 0-15) (DCMPAREACRAn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | EN | B'0 | R/W | Decompression Enable for this Area 0: Disable 1: Enable |
| 30, 29 | FMT | B'00 | R/W | Specify the Decompression Format in this Area B'00: YUV planar B'01: YUV422 Interleave B'10: ARGB8888 B'11: Reserved |
| 28 to 18 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 17 to 0 | SADDR | H'0_0000 | R/W | Start Address[37:20] Target area is equal to or more than Start Address. These bits specified bits 37 to 20 of target area start address value. Bits 19 to 0 of target address are not considered on this function. |

21.2.6 Decompression Area Configuration Register Bn (n = 0-15) (DCMPAREACRBn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|--------------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | EADDR[37:36] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | EADDR[35:20] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 18 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 17 to 0 | EADDR | H'0_0000 | R/W | End Address[37:20] Target area is less than End Address. These bits specified bits 37 to 20 of target area end address value. Bits 19 to 0 of target address are not considered on this function. |

21.2.7 Decompression Shadow Area Configuration Register (DCMPSHDWCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------|-----|-----|-----|-----|-----|----------|-------------|-----|-----|-----|-----|-----|-----|----------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | AREA3 EN | — | — | — | — | — | AREA3FMT | AREA2 EN | — | — | — | — | — | — | AREA2FMT | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | AREA1 EN | — | — | — | — | — | AREA1FMT | AREA0 EN | — | — | — | — | — | — | AREA0FMT | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | AREA3EN | B'0 | R/W | Shadow Area3 Decompression Enable 0: Disable 1: Enable |
| 30 to 26 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 25, 24 | AREA3FMT | B'00 | R/W | Specify the Shadow Area3 Decompression Format B'00: YUV planar B'01: YUV422 Interleave B'10: ARGB8888 B'11: Reserved |
| 23 | AREA2EN | B'0 | R/W | Shadow Area2 Decompression Enable 0: Disable 1: Enable |
| 22 to 18 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 17, 16 | AREA2FMT | B'00 | R/W | Specify the Shadow Area2 Decompression Format B'00: YUV planar B'01: YUV422 Interleave B'10: ARGB8888 B'11: Reserved |
| 15 | AREA1EN | B'0 | R/W | Shadow Area1 Decompression Enable 0: Disable 1: Enable |
| 14 to 10 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 to 8 | AREA1FMT | B'00 | R/W | Specify the Shadow Area1 Decompression Format B'00: YUV planar B'01: YUV422 Interleave B'10: ARGB8888 B'11: Reserved |
| 7 | AREA0EN | B'0 | R/W | Shadow Area0 Decompression Enable 0: Disable 1: Enable |
| 6 to 2 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 1, 0 | AREA0FMT | B'00 | R/W | Specify the Shadow Area0 Decompression Format B'00: YUV planar B'01: YUV422 Interleave B'10: ARGB8888 B'11: Reserved |

21.2.8 Main Memory AXI Control Register (MMCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-------|-----|-----|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CTRL 31 | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CTRL 16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CTRL3 | — | — | CTRL0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | CTRL31 | B'0 | R/W | Bus Control 31 Set 0 on this bit. |
| 30 to 17 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | CTRL16 | B'0 | R/W | Bus Control 16 Set below value on this bit: RZ/G2H: 1 Others: 0 |
| 15 to 4 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 | CTRL3 | B'0 | R/W | Bus Control 3 Set below value on this bit: RZ/G2H: 1 Others: 0 |
| 2, 1 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | CTRL0 | B'0 | R/W | Bus Control 0 Set below value on this bit: RZ/G2M V1.3/RZ/G2M V3.0: 1 Others: 0 |

21.2.9 DRAM Protected Area Division Register n (n = 0-14) (DPTDIVCRn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----------------|-----|-----|-----|----------------|----------------|----------------|----------------|-----|-----|----------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | SECG0R EGWP | SECG1R EGWP | SECG2R EGWP | SECG3R EGWP | — | — | DIVADDR[37:16] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DIVADDR[37:16] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 28 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 | SECG0REGWP | B'0 | R/W | Secure Group0 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 26 | SECG1REGWP | B'0 | R/W | Secure Group1 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 25 | SECG2REGWP | B'0 | R/W | Secure Group2 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 24 | SECG3REGWP | B'0 | R/W | Secure Group3 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 23, 22 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 to 0 | DIVADDR | H'00_0000 | R/W | Protection area division physical address [37:16] |

21.2.10 DRAM Protected Area Setting Register n (n = 0-15) (DPTCRn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|----------------|----------------|----------------|----------------|-----|-----|-----|-----|-------------|-------------|-------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | SECG0R EGWP | SECG1R EGWP | SECG2R EGWP | SECG3R EGWP | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | SECG0R RP | SECG1R RP | SECG2R RP | SECG3R RP | — | — | — | — | SECG0 WP | SECG1 WP | SECG2 WP | SECG3 WP |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 28 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 | SECG0REGWP | B'0 | R/W | Secure Group0 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 26 | SECG1REGWP | B'0 | R/W | Secure Group1 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 25 | SECG2REGWP | B'0 | R/W | Secure Group2 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 24 | SECG3REGWP | B'0 | R/W | Secure Group3 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 23 to 12 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 | SECG0RP | B'0 | R/W | Secure Group0 Read Privilege Setting 0: Has the privilege to read the relevant DRAM area 1: Does not have the privilege to read the relevant DRAM area Secure Master is not restricted by this setting. |
| 10 | SECG1RP | B'0 | R/W | Secure Group1 Read Privilege Setting 0: Has the privilege to read the relevant DRAM area 1: Does not have the privilege to read the relevant DRAM area Secure Master is not restricted by this setting. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 9 | SECG2RP | B'0 | R/W | Secure Group2 Read Privilege Setting 0: Has the privilege to read the relevant DRAM area 1: Does not have the privilege to read the relevant DRAM area Secure Master is not restricted by this setting. |
| 8 | SECG3RP | B'0 | R/W | Secure Group3 Read Privilege Setting 0: Has the privilege to read the relevant DRAM area 1: Does not have the privilege to read the relevant DRAM area Secure Master is not restricted by this setting. |
| 7 to 4 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 | SECG0WP | B'0 | R/W | Secure Group0 Write Privilege Setting 0: Has the privilege to write the relevant DRAM area 1: Does not have the privilege to write the relevant DRAM area Secure Master is not restricted by this setting. |
| 2 | SECG1WP | B'0 | R/W | Secure Group1 Write Privilege Setting 0: Has the privilege to write the relevant DRAM area 1: Does not have the privilege to write the relevant DRAM area Secure Master is not restricted by this setting. |
| 1 | SECG2WP | B'0 | R/W | Secure Group2 Write Privilege Setting 0: Has the privilege to write the relevant DRAM area 1: Does not have the privilege to write the relevant DRAM area Secure Master is not restricted by this setting. |
| 0 | SECG3WP | B'0 | R/W | Secure Group3 Write Privilege Setting 0: Has the privilege to write the relevant DRAM area 1: Does not have the privilege to write the relevant DRAM area Secure Master is not restricted by this setting. |

21.2.11 System RAM Protected Area Division Register n (n = 0-14) (SPTDIVCRn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----------------|-----|-----|-----|----------------|----------------|----------------|----------------|-----|-----|-----|-----|----------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | SECG0R EGWP | SECG1R EGWP | SECG2R EGWP | SECG3R EGWP | — | — | — | — | DIVADDR[31:12] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DIVADDR[31:12] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 28 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 | SECG0REGWP | B'0 | R/W | Secure Group0 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 26 | SECG1REGWP | B'0 | R/W | Secure Group1 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 25 | SECG2REGWP | B'0 | R/W | Secure Group2 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 24 | SECG3REGWP | B'0 | R/W | Secure Group3 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 23 to 20 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 to 0 | DIVADDR | All 0 | R/W | Protection area division physical address [31:12] |

21.2.12 System RAM Protected Area Setting Register n (n = 0-15) (SPTCRn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|----------------|----------------|----------------|----------------|-----|-----|-----|-----|-------------|-------------|-------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | SECG0R EGWP | SECG1R EGWP | SECG2R EGWP | SECG3R EGWP | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | SECG0R P | SECG1R P | SECG2R P | SECG3R P | — | — | — | — | SECG0 WP | SECG1 WP | SECG2 WP | SECG3 WP |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 28 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 | SECG0REGWP | B'0 | R/W | Secure Group0 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 26 | SECG1REGWP | B'0 | R/W | Secure Group1 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 25 | SECG2REGWP | B'0 | R/W | Secure Group2 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 24 | SECG3REGWP | B'0 | R/W | Secure Group3 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 23 to 12 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 | SECG0RP | B'0 | R/W | Secure Group0 Read Privilege Setting 0: Has the privilege to read the relevant System RAM area 1: Does not have the privilege to read the relevant System RAM area Secure Master is not restricted by this setting. |
| 10 | SECG1RP | B'0 | R/W | Secure Group1 Read Privilege Setting 0: Has the privilege to read the relevant System RAM area 1: Does not have the privilege to read the relevant System RAM area Secure Master is not restricted by this setting. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 9 | SECG2RP | B'0 | R/W | Secure Group2 Read Privilege Setting 0: Has the privilege to read the relevant System RAM area 1: Does not have the privilege to read the relevant System RAM area Secure Master is not restricted by this setting. |
| 8 | SECG3RP | B'0 | R/W | Secure Group3 Read Privilege Setting 0: Has the privilege to read the relevant System RAM area 1: Does not have the privilege to read the relevant System RAM area Secure Master is not restricted by this setting. |
| 7 to 4 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 | SECG0WP | B'0 | R/W | Secure Group0 Write Privilege Setting 0: Has the privilege to write the relevant System RAM area 1: Does not have the privilege to write the relevant System RAM area Secure Master is not restricted by this setting. |
| 2 | SECG1WP | B'0 | R/W | Secure Group1 Write Privilege Setting 0: Has the privilege to write the relevant System RAM area 1: Does not have the privilege to write the relevant System RAM area Secure Master is not restricted by this setting. |
| 1 | SECG2WP | B'0 | R/W | Secure Group2 Write Privilege Setting 0: Has the privilege to write the relevant System RAM area 1: Does not have the privilege to write the relevant System RAM area Secure Master is not restricted by this setting. |
| 0 | SECG3WP | B'0 | R/W | Secure Group3 Write Privilege Setting 0: Has the privilege to write the relevant System RAM area 1: Does not have the privilege to write the relevant System RAM area Secure Master is not restricted by this setting. |

21.2.13 TR3 Control Register (TR3CR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | — |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | TRCTRL16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 17 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | TRCTRL16 | B'0 | R/W | TR Control 16 Set below value on this bit: RZ/G2H: 1 Others: 0 |
| 15 to 0 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |

21.2.14 TR4 Control Register (TR4CR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | — | — |

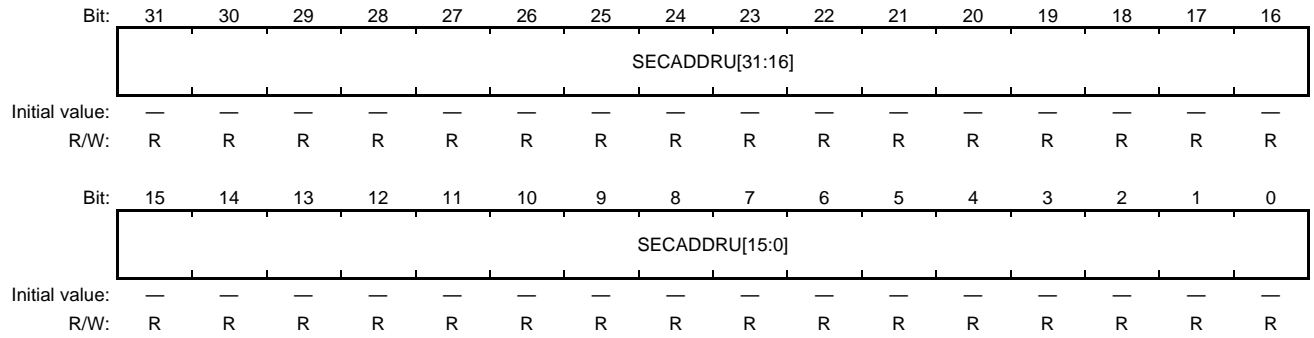
| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | TRCTRL16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 17 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | TRCTRL16 | B'0 | R/W | TR Control 16 Set below value on this bit: RZ/G2H: 1 Others: 0 |
| 15 to 0 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |

21.2.15 Secure Error Register Upper Address Register (SECADDRU_X)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

X indicates each domain related to this register. Refer to Table 21.1 for X.

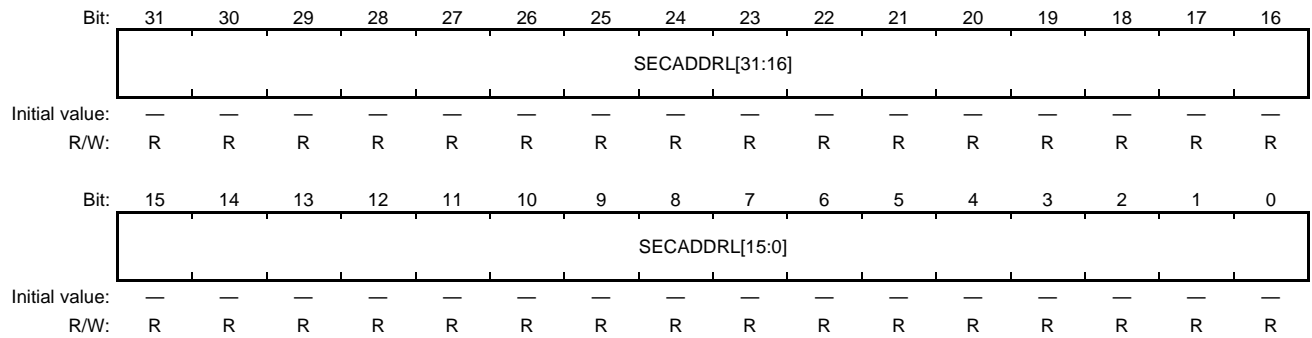


This register is security use only. Keep initial value.

21.2.16 Secure Error Register Lower Address Register (SECADDRL_X)

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

X indicates each domain related to this register. Refer to Table 21.1 for X.

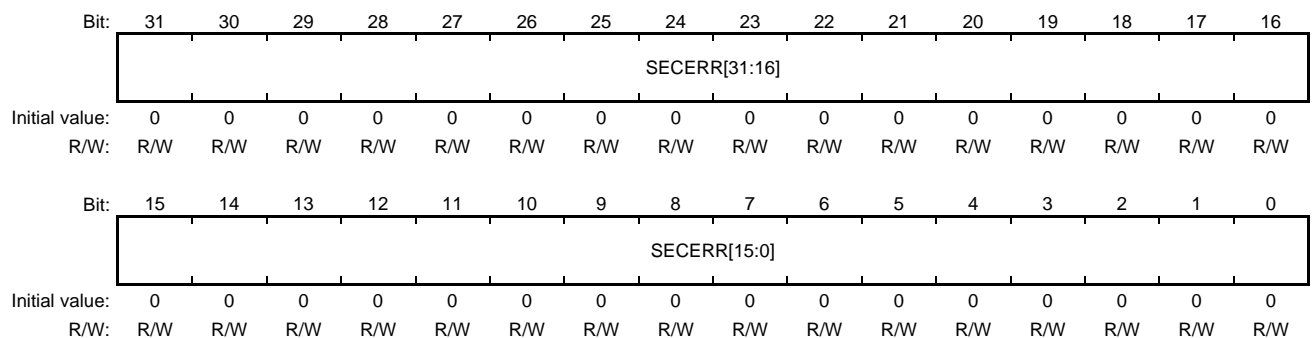


This register is security use only. Keep initial value.

21.2.17 Secure Error Register (SECERR_X)

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

X indicates each domain related to this register. Refer to Table 21.1 for X.



This register is security use only. Keep initial value.

21.2.18 QOSCTRL SL INIT Register (QOSCTRL_SL_INIT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: There are unsupported products depending on its refresh interval period;

1.89 usec: RZ/G2E is not supported.

7.80 usec: RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N are not supported.

In this case, sslotclk field is always read as 0 and the write value should always be 0.

| | | | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|----------|-----|-----|----------|-----|-----|-----|-----|-----------|-----|-----|-----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | refsslot | | | | — | — | — | — | slotsslot | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | | |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | sslotclk | | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | | |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 28 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 to 24 | refsslot | H'3 | R/W | refsslot Setting. Set 0x3 on these bits. A value other than above must not be set to these bits. These bits are used when refresh_mode bit of QOSCTRL_STATQC is 0. |
| 23 to 20 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 to 16 | slotsslot | H'7 | R/W | slotsslot Setting. Set 0x5 on these bits. A value other than above must not be set to these bits. |
| 15 to 9 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 to 0 | sslotclk | H'0FA | R/W | sslotclk Setting. Set below values depending on DDR refresh interval period. 1.89 usec refresh interval: H'07D 3.78 usec refresh interval: H'0FB [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] H'0AE [RZ/G2E] 7.80 usec refresh interval: H'15D [RZ/G2E] |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.19 QOSCTRL REF ARS Register (QOSCTRL_REF_ARS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: There are unsupported products depending on its refresh interval period;

1.89 usec: RZ/G2E is not supported.

7.80 usec: RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N are not supported.

In this case, arbstopcycle field value is always read as 0 and the write value should always be 0.

| | | | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|--------------|-----|-----|-----|-----|-----|-----|-----|-----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | arbstopcycle | | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | | |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 25 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 to 16 | arbstopcycle | H'064 | R/W | arbstopcycle Setting. Set below values depending on DDR refresh interval period. 1.89 usec refresh interval: H'078 3.78 usec refresh interval: H'0F6 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 3.90 usec: refresh interval: H'102 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] H'0A9 [RZ/G2E] 7.80 usec: refresh interval: H'158 [RZ/G2E] |
| 15 to 0 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.20 QOSCTRL REF ENBL Register (QOSCTRL_REF_ENBL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | refssloten | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | refssloten | H'0000 | R/W | refssloten Setting. Set H'0000 on these bits. These bits are used when refresh_mode bit of QOSCTRL_STATQC is 1. A value other than above must not be set to these bits. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.21 QOSCTRL_STATQC Register (QOSCTRL_STATQC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|--------------|-----|-----|-----|-----|-----|-----|-----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | refresh_mode | — | — | — | — | — | — | — | statqen |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|---|
| 31 to 9 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | refresh_mode | B'0 | R/W | refresh_mode setting. Set 0x0 on this bit. This bit must be set with statqen bit. |
| 7 to 1 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | statqen | B'0 | R/W | QOS control Enable 0: Disable 1: Enable. |

21.2.22 QOSCTRL MEMBANK Register (QOSCTRL_MEMBANK)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----------------|-----|-----|-----|-----|-----|-----|-----|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | exe_me mbank | — | — | — | — | — | — | — | mem bank |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|--|
| 31 to 9 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | exe_membank | B'0 | R/W | Current using bank of QOSBW Status. 0: using QOSBW_FIX_QOS_BANK0 and QOSBW_BE_QOS_BANK0 1: using QOSBW_FIX_QOS_BANK1 and QOSBW_BE_QOS_BANK1 |
| 7 to 1 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | membank | B'0 | R/W | Bank of QOSBW_FIX_QOS_BANK0, QOSBW_FIX_QOS_BANK1, QOSBW_BE_QOS_BANK0 and QOSBW_BE_QOS_BANK1 registers to control QOS Setting.* 0: use QOSBW_FIX_QOS_BANK0 and QOSBW_BE_QOS_BANK0 1: use QOSBW_FIX_QOS_BANK1 and QOSBW_BE_QOS_BANK1 |

Notes: * Bit [8](exe_membank) specifies "Current Bank" of QOSBW_FIX_QOS_BANK and QOSBW_BE_QOS_BANK. Shifting QOSBW_FIX_QOS_BANK and QOSBW_BE_QOS_BANK, write opposite value of read value of Bit [8](exe_membank) to Bit [0](membank). Subsequently, set equivalent QOSBW_FIX_QOS_BANK and QOSBW_BE_QOS_BANK.
(ex. Bit [8] = 1'b0; write 1'b1 to Bit [0] and set QOSBW_FIX_QOS_BANK1 and QOSBW_BE_QOS_BANK1, vice versa.)
Do not write any value other than mentioned above.
This register can be written regardless of statqen bit value of QOSCTRL_STATQC register.

21.2.23 QOSCTRL RAS Register (QOSCTRL_RAS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|----------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | EnumInit | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | EnumInit | H'20 | R/W | EnumInit Setting. Set below value on each product: RZ/G2H: H'44 RZ/G2M V1.3, RZ/G2M V3.0: H'44 RZ/G2N: H'28 RZ/G2E: H'20 Notice: Above value for , RZ/G2H and RZ/G2M V1.3 and RZ/G2M V3.0 are based on below condition. - Number of DRAM CH: RZ/G2H: 32bits × 2CH RZ/G2M V1.3, RZ/G2M V3.0: 32bits × 2CH - Split Mapping: All area is set as split mapping. For other configuration case, please contact Renesas. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.24 QOSCTRL DANT Register (QOSCTRL_DANT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|--------------------|-----|-----|-----|-----|-----|-----|--------------------|--------------------|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | DAccNodeThreshold2 | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | DAccNodeThreshold1 | | | | | | — | DAccNodeThreshold0 | | | | | | | |
| Initial value: | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------|---------------|-----|---|
| 31 to 23 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 22 to 16 | DAccNodeThreshold2 | H'20 | R/W | DAccNodeThreshold2 Setting. Set below value on these bits: RZ/G2H, RZ/G2M Ver1.3, RZ/G2M Ver.3.0: H'20 Others: H'10 A value other than above must not be set to these bits. |
| 15 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 14 to 8 | DAccNodeThreshold1 | H'10 | R/W | DAccNodeThreshold1 Setting. Set below value on these bits. RZ/G2H, RZ/G2M Ver1.3, RZ/G2M Ver.3.0: H'10 Others: H'08 A value other than above must not be set to these bits. |
| 7 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 to 0 | DAccNodeThreshold0 | H'08 | R/W | DAccNodeThreshold0 Setting. Set below value on these bits. RZ/G2H, RZ/G2M Ver1.3, RZ/G2M Ver.3.0: H'0A Others: H'04 A value other than above must not be set to these bits. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.25 QOSCTRL DANN LOW Register (QOSCTRL_DANN_LOW)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|------------------|-----|-----|-----|-----|-----|-----|------------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | DAccNodeNum_FIX3 | | | | — | — | — | DAccNodeNum_FIX2 | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | DAccNodeNum_FIX1 | | | | — | — | — | DAccNodeNum_FIX0 | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|--|
| 31 to 29 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 24 | DAccNodeNum_FIX3 | H'04 | R/W | DAccNodeNum_FIX3 Setting. Set H'02 on these bits. These bits setting range is 0, 1, 2, 4, 8 or 16. |
| 23 to 21 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 to 16 | DAccNodeNum_FIX2 | H'02 | R/W | DAccNodeNum_FIX2 Setting. Set H'02 on these bits. These bits setting range is 0, 1, 2, 4, 8 or 16. |
| 15 to 13 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 8 | DAccNodeNum_FIX1 | H'02 | R/W | DAccNodeNum_FIX1 Setting. Set H'02 on these bits. These bits setting range is 0, 1, 2, 4, 8 or 16. |
| 7 to 5 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | DAccNodeNum_FIX0 | H'01 | R/W | DAccNodeNum_FIX0 Setting. Set H'01 on these bits. These bits setting range is 0, 1, 2, 4, 8 or 16. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.26 QOSCTRL DANN HIGH Register (QOSCTRL_DANN_HIGH)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----------------|-----|-----|-----|-----|-----|-----|-----------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | DAccNodeNum_BE3 | | | | — | — | — | DAccNodeNum_BE2 | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | DAccNodeNum_BE1 | | | | — | — | — | DAccNodeNum_BE0 | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 to 29 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 24 | DAccNodeNum_BE3 | H'04 | R/W | DAccNodeNum_BE3 Setting. Set H'04 on these bits. These bits setting range is 0, 1, 2, 3, 4, 5, 8, 9 or 16. |
| 23 to 21 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 to 16 | DAccNodeNum_BE2 | H'02 | R/W | DAccNodeNum_BE2 Setting. Set below value on these bits: Other than RZ/G2N: H'04 RZ/G2N: H'02 These bits setting range is 0, 1, 2, 3, 4, 5, 8, 9 or 16. |
| 15 to 13 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 8 | DAccNodeNum_BE1 | H'02 | R/W | DAccNodeNum_BE1 Setting. Set below value on these bits: RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E: H'2 RZ/G2N: H'00 These bits setting range is 0, 1, 2, 3, 4, 5, 8, 9 or 16. |
| 7 to 5 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | DAccNodeNum_BE0 | H'01 | R/W | DAccNodeNum_BE0 Setting. Set H'00 on these bits. These bits setting range is 0, 1, 2, 3, 4, 5, 8, 9 or 16. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.27 QOSCTRL EMS LOW Register (QOSCTRL_EMS_LOW)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|---------------|-----|-----|-----|---------------|-----|-----|-----|---------------|-----|-----|-----|---------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | emaxschedule7 | | | — | emaxschedule6 | | | — | emaxschedule5 | | | — | emaxschedule4 | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | emaxschedule3 | | | — | emaxschedule2 | | | — | emaxschedule1 | | | — | emaxschedule0 | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

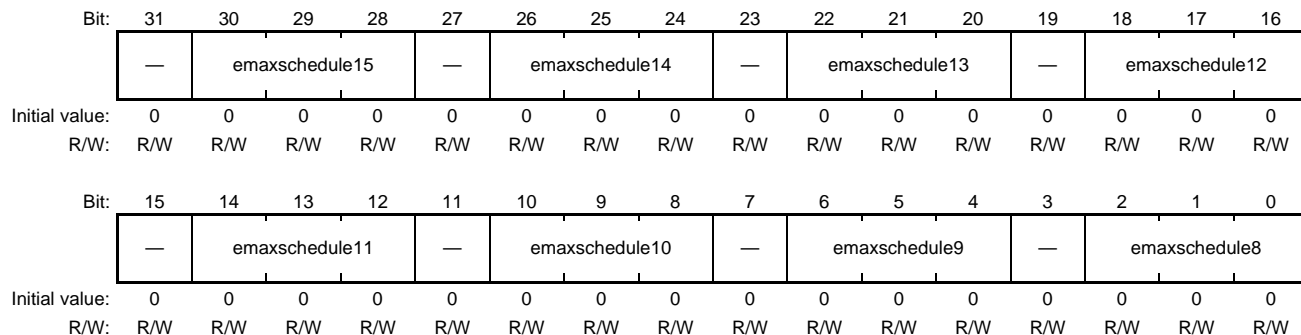
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 30 to 28 | emaxschedule7 | B'000 | R/W | emaxschedule7 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |
| 27 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 26 to 24 | emaxschedule6 | B'000 | R/W | emaxschedule6 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |
| 23 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 22 to 20 | emaxschedule5 | B'000 | R/W | emaxschedule5 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |
| 19 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 18 to 16 | emaxschedule4 | B'000 | R/W | emaxschedule4 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |
| 15 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 14 to 12 | emaxschedule3 | B'000 | R/W | emaxschedule3 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |
| 11 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 10 to 8 | emaxschedule2 | B'000 | R/W | emaxschedule2 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |
| 7 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 to 4 | emaxschedule1 | B'000 | R/W | emaxschedule1 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |
| 3 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 to 0 | emaxschedule0 | B'000 | R/W | emaxschedule0 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.28 QOSCTRL EMS HIGH Register (QOSCTRL_EMS_HIGH)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 30 to 28 | emaxschedule15 | B'000 | R/W | emaxschedule15 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |
| 27 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 26 to 24 | emaxschedule14 | B'000 | R/W | emaxschedule14 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |
| 23 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 22 to 20 | emaxschedule13 | B'000 | R/W | emaxschedule13 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |
| 19 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 18 to 16 | emaxschedule12 | B'000 | R/W | emaxschedule12 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |
| 15 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 14 to 12 | emaxschedule11 | B'000 | R/W | emaxschedule11 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |
| 11 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 to 8 | emaxschedule10 | B'000 | R/W | emaxschedule10 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------|---------------|-----|--|
| 7 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 to 4 | emaxschedule9 | B'000 | R/W | emaxschedule9 Setting. Set B'000 on these bits. The range of these bits is B'000 to B'100. |
| 3 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 to 0 | emaxschedule8 | B'000 | R/W | emaxschedule8 Setting. Set B'000 on these bits The range of these bits is B'000 to B'100. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.29 QOSCTRL EARLYR Register (QOSCTRL_EARLYR)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | √ | √ |

[RZ/G2H]

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | early return mode |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

[RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | early_return mode |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

[RZ/G2H]

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------------|---------------|-----|--|
| 31 to 1 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | early_return_mode | B'0 | R/W | early_return_mode Setting. Set 0 on this bit. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

[RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------------|---------------|-----|--|
| 31 to 2 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 1, 0 | early_return_mode | B'00 | R/W | early_return_mode Setting. Set below value on these bits: RZ/G2N: B'01 Others: B'00 |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.30 QOSCTRL RACNT0 Register (QOSCTRL_RACNT0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|--------------------|-----|-----|-----|-----|-----|--------------------------|-----|-----|----------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | gpu_ticket mode | — | — | — | — | — | be_ticket_mask _cycle | — | — | ticket_mask cycle |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

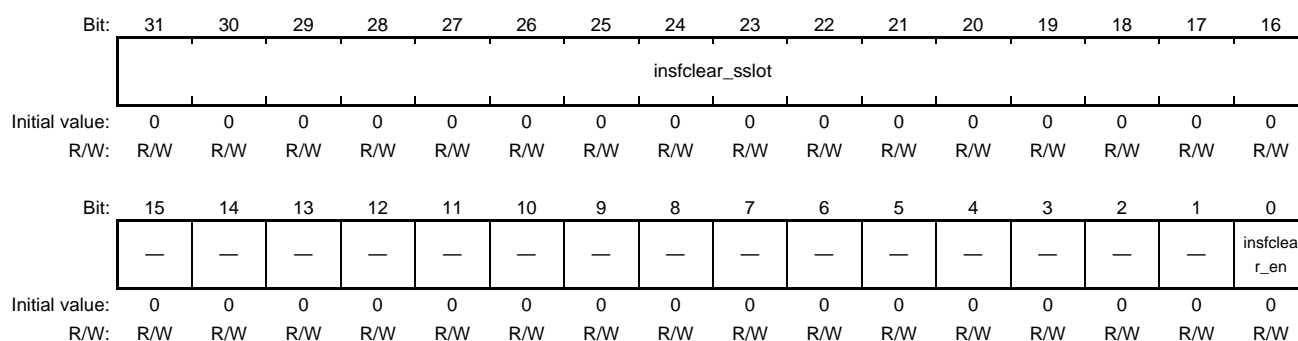
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------------|---------------|-----|---|
| 31 to 26 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 25, 24 | gpu_ticket_mode | B'00 | R/W | [RZ/G2M V1.3, RZ/G2M V3.0] gpu_ticket_mode Setting. Set B'10 on these bits. A value other than above must not be set to these bits. [Other Product] Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 20 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 19, 18 | be_ticket_mask_cycle | B'00 | R/W | [RZ/G2N, RZ/G2E] be_ticket_mask_cycle Setting. Set B'00 on these bits. A value other than above must not be set to these bits. [Other Product] Reserved These bits are always read as 0. The write value should always be 0. |
| 17, 16 | ticket_mask_cycle | B'00 | R/W | ticket_mask_cycle Setting. Set B'01 on these bits. A value other than above must not be set to these bits. |
| 15 to 2 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | swapin_inform_disable | B'0 | R/W | swapin_inform_disable Setting. Set 1 on this bit. A value other than above must not be set to this bit. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------------------|---------------|-----|---|
| 0 | beticket_mask_disable | B'0 | R/W | beticket_mask_disable Setting. Set B'1 on this bit. A value other than above must not be set to this bit. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.31 QOSCTRL INSFC Register (QOSCTRL_INSFC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 to 16 | insfclear_sslot | H'0000 | R/W | insfclear_sslot Setting. Set H'0633 on these bits. A value other than above must not be set to these bits. |
| 15 to 1 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | insfclear_en | B'0 | R/W | insfclear_en Enable. Set 1 on this bit. A value other than above must not be set to this bit. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.32 QOSCTRL RAEN Register (QOSCTRL_RAEN)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | RAlloc Enable |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|--|
| 31 to 1 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | RAllocEnable | B'0 | R/W | RAllocEnable Enable. This bit must be set 1 before QOSCTRL_STATQC setting.. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.33 QOSCTRL_STATGEN0 Register (QOSCTRL_STATGEN0)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — | — |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | gpu ticket mode |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|--|
| 31 to 1 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | gpu_ticket_mode | B'0 | R/W | gpu_ticket_mode Setting Set 1 on this bit. A value other than above must not be set to this bit. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.34 QOSWT WTEN Register (QOSWT_WTEN)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

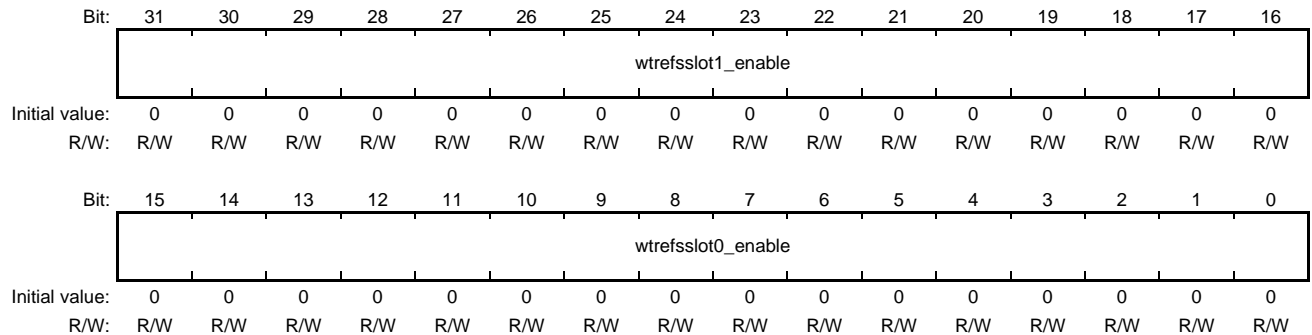
| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | wtbank_en |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | wtbank_en | B'0 | R/W | wtbank_en Enable. Using the Write DQ training, set this bit "1" after setting PMSTREN bit of DBDFIPMSTRCNF "1", and WTMODE of DBDFIPMSTRCNF "B'01". 0: Disable 1: Enable |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.35 QOSWT WTREF Register (QOSWT_WTREF)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------|---------------|-----|---|
| 31 to 16 | wtrefslot1_enable | H'0000 | R/W | wtrefslot1_enable Setting. Set below value on these bits: RZ/G2E: H'0000 Others: H'0001 A value other than above must not be set to these bits. |
| 15 to 0 | wtrefslot0_enable | H'0000 | R/W | wtrefslot0_enable Setting. Set below value on these bits: RZ/G2E: H'0000 Others: H'0001 A value other than above must not be set to these bits. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.36 QOSWT WTSET0 Register (QOSWT_WTSET0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|-----|-----|-----|----------|-----|-----|-----|-----|-----|-----|-----|--------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | wtperiod0 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | | | | wtsslot0 | | | | — | | | | wtslotsslot0 | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 16 | wtperiod0 | H'0000 | R/W | wtperiod0 Setting. Set below value on these bits: RZ/G2E: H'0000 Others: 1.89 usec refresh interval: H'2955 3.78 usec refresh interval: H'14AA A value other than above must not be set to these bits. |
| 15 to 12 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 8 | wtsslot0 | H'0 | R/W | wtsslot0 Setting. Set below value on these bits: RZ/G2E: H'0 Others: H'4 A value other than above must not be set to these bits. |
| 7 to 4 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | wtslotsslot0 | H'0 | R/W | wtslotsslot0 Setting. Set below value on these bits: RZ/G2E: H'0 Others: H'B A value other than above must not be set to these bits. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.37 QOSWT WTSET1 Register (QOSWT_WTSET1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|-----|-----|-----|----------|-----|-----|-----|-----|-----|-----|-----|--------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | wtperiod1 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | | | | wtsslot1 | | | | — | | | | wtslotsslot1 | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 16 | wtperiod1 | H'0000 | R/W | wtperiod1 Setting. Set below value on these bits: RZ/G2E: H'0000 Others: 1.89 usec refresh interval: H'2955 3.78 usec refresh interval: H'14AA A value other than above must not be set to these bits. |
| 15 to 12 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 8 | wtsslot1 | H'0 | R/W | wtsslot1 Setting. Set below value on these bits: RZ/G2E: H'0 Others: H'4 A value other than above must not be set to these bits. |
| 7 to 4 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | wtslotsslot1 | H'0 | R/W | wtslotsslot1 Setting. Set below value on these bits: RZ/G2E: H'0 Others: H'B A value other than above must not be set to these bits. |

Note: This register must be written during statqen bit of QOSCTRL_STATQC register is 0.

21.2.38 QOSBW FIX QOS BANK0 Register n (QOSBW_FIX_QOS_BANK0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[RZ/G2H] n: 0 to 365, Address range: H'E67E_0000 to H'E67E_0B68

[RZ/G2M V1.3, RZ/G2M V3.0] n: 0 to 362, Address range: H'E67E_0000 to H'E67E_0B50

[RZ/G2N] n: 0 to 370, Address range: H'E67E_0000 to H'E67E_0B90

[RZ/G2E] n: 0 to 255, Address range: H'E67E_0000 to H'E67E_07F8

| | | | | | | | | | | | | | | | | |
|----------------|----------------|-----|-----|-----|-----|-----|----------------|-----|----------------|-----|-----|----------------|-----|----------------|-----|-----|
| Bit: | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 | 48 |
| | — | — | — | — | — | — | — | — | — | — | — | FIXQOS_BANK0_5 | | FIXQOS_BANK0_4 | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 47 | 46 | 45 | 44 | 43 | 42 | 41 | 40 | 39 | 38 | 37 | 36 | 35 | 34 | 33 | 32 |
| | FIXQOS_BANK0_4 | | | | | | FIXQOS_BANK0_3 | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | FIXQOS_BANK0_2 | | | | | | | | FIXQOS_BANK0_1 | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | FIXQOS_BANK0_0 | | | | | | | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 63 to 57 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 56 to 53 | — | — | R/W | Reserved The write value should always be 0. |
| 52 to 50 | FIXQOS_BANK0_5 | — | R/W | FIXQOS_BANK0_5 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. Setting value must be 0, 3, 4 or 5. |
| 49 to 42 | FIXQOS_BANK0_4 | — | R/W | FIXQOS_BANK0_4 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |
| 41 to 32 | FIXQOS_BANK0_3 | — | R/W | FIXQOS_BANK0_3 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 24 | FIXQOS_BANK0_2 | — | R/W | FIXQOS_BANK0_2 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |
| 23 to 16 | FIXQOS_BANK0_1 | — | R/W | FIXQOS_BANK0_1 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |
| 15 to 0 | FIXQOS_BANK0_0 | — | R/W | FIXQOS_BANK0_0 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |

Notes: This register must be accessed by 8byte access.

This register must be written during statqen bit of QOSCTRL_STATQC register is 0 or exe_membank of QOSCTRL_MEMBANK is 1.

Refer to the Table 21.2 for setting values of QOSBW_FIX_QOS_BANK0 registers.

Table 21.2 Setting values of QOSBW_FIX_QOS_BANK0 registers (1/2)

| Address | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|--------|
| H'E67E_0000 to H'E67E_0350 | *1 | *1 | *1 | *1 |
| H'E67E_0358 to H'E67E_0368 | *1 | H'0000_ 0000_ 0000_ 0000 | *1 | *1 |
| H'E67E_0370 to H'E67E_0390 | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 | *1 | *1 |
| H'E67E_0830 | H'0010_ 0404_ 0000_ C010 | H'0010_ 0404_ 0000_ C010 | H'0010_ 0432_ 0000_ C010 | — |
| H'E67E_0838 | H'0010_ 0807_ 0000_ C010 | H'0010_ 0404_ 0000_ C010 | H'0010_ 0432_ 0000_ C010 | — |
| H'E67E_0840 | H'0014_ 1007_ 0000_ FFF0 | H'0014_ 1409_ 0000_ FFF0 | H'0014_ 0C5D_ 0000_ FFF0 | — |
| H'E67E_0858 | H'0014_ 100D_ 0000_ C010 | H'0014_ 0C0A_ 0000_ C010 | H'0014_ 0C94_ 0000_ C010 | — |
| H'E67E_0860 | H'0014_ 100D_ 0000_ C010 | H'0014_ 0C0A_ 0000_ C010 | H'0014_ 0C94_ 0000_ C010 | — |
| H'E67E_0878 | H'0000_ 0000_ 0000_ 0000 | H'0010_ 0403_ 0000_ C010 | H'0000_ 0000_ 0000_ 0000 | — |
| H'E67E_0888 | H'0104_ 1007_ 0000_ FF00 | H'0014_ 1409_ 0000_ FFF0 | H'0014_ 041F_ 0000_ FFF0 | — |
| H'E67E_0890 | H'0000_ 0000_ 0000_ 0000 | H'0014_ 0807_ 0000_ C010 | H'0000_ 0000_ 0000_ 0000 | — |
| H'E67E_0A60 | H'000C_ 0802_ 0000_ FFF0 | H'000C_ 0802_ 0000_ FFF0 | H'000C_ 0815_ 0000_ FFF0 | — |

Table 21.2 Setting values of QOSBW_FIX_QOS_BANK0 registers (2/2)

| Address | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|------------------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|
| H'E67E_0A68 | H'0014_ 0801_ 0000_ FFF0 | H'0014_ 0801_ 0000_ FFF0 | H'0014_ 0802_ 0000_ FFF0 | — |
| H'E67E_0A70 | H'0014_ 0401_ 0000_ FFF0 | H'0014_ 0401_ 0000_ FFF0 | H'0014_ 0401_ 0000_ FFF0 | — |
| H'E67E_0A78 | H'000C_ 0401_ 0000_ FFF0 | H'000C_ 0401_ 0000_ FFF0 | H'000C_ 0409_ 0000_ FFF0 | — |
| H'E67E_0A90 | H'0014_ 0801_ 0000_ FFF0 | H'0014_ 0801_ 0000_ FFF0 | H'0014_ 0802_ 0000_ FFF0 | — |
| H'E67E_0A98 | H'0014_ 0401_ 0000_ FFF0 | H'0014_ 0401_ 0000_ FFF0 | H'0014_ 0401_ 0000_ FFF0 | — |
| Other Address | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 |

Notes: — : Not supported

*1: Set values generated by a tool released from Renesas

21.2.39 QOSBW FIX QOS BANK1 Register n (QOSBW_FIX_QOS_BANK1)

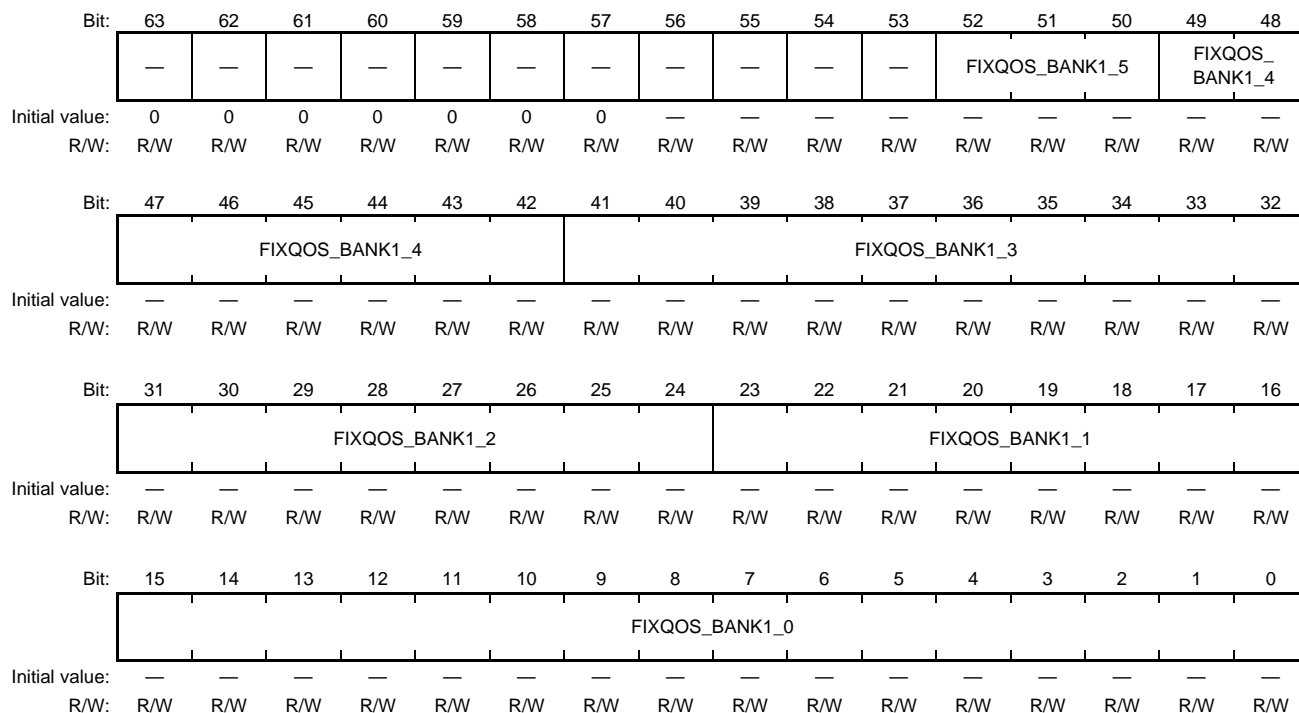
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[RZ/G2H] n: 0 to 365, Address range: H'E67E_1000 to H'E67E_1B68

[RZ/G2M V1.3, RZ/G2M V3.0] n: 0 to 362, Address range: H'E67E_1000 to H'E67E_1B50

[RZ/G2N] n: 0 to 370, Address range: H'E67E_1000 to H'E67E_1B90

[RZ/G2E] n: 0 to 255, Address range: H'E67E_1000 to H'E67E_17F8



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 63 to 57 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 56 to 53 | — | — | R/W | Reserved The write value should always be 0. |
| 52 to 50 | FIXQOS_BANK1_5 | — | R/W | FIXQOS_BANK1_5 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. Setting value must be 0, 3, 4 or 5. |
| 49 to 42 | FIXQOS_BANK1_4 | — | R/W | FIXQOS_BANK1_4 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |
| 41 to 32 | FIXQOS_BANK1_3 | — | R/W | FIXQOS_BANK1_3 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 24 | FIXQOS_BANK1_2 | — | R/W | FIXQOS_BANK1_2 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |
| 23 to 16 | FIXQOS_BANK1_1 | — | R/W | FIXQOS_BANK1_1 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |
| 15 to 0 | FIXQOS_BANK1_0 | — | R/W | FIXQOS_BANK1_0 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |

Notes: This register must be accessed by 8byte access.

This register must be written during statqen bit of QOSCTRL_STATQC register is 0 or exe_membank of QOSCTRL_MEMBANK is 0.

Refer to the Table 21.3 for setting values of QOSBW_FIX_QOS_BANK1 registers.

Table 21.3 Setting values of QOSBW_FIX_QOS_BANK1 registers (1/2)

| Address | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|--------|
| H'E67E_1000 to H'E67E_1350 | *1 | *1 | *1 | *1 |
| H'E67E_1358 to H'E67E_1368 | *1 | H'0000_ 0000_ 0000_ 0000 | *1 | *1 |
| H'E67E_1370 to H'E67E_1390 | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 | *1 | *1 |
| H'E67E_1830 | H'0010_ 0404_ 0000_ C010 | H'0010_ 0404_ 0000_ C010 | H'0010_ 0432_ 0000_ C010 | — |
| H'E67E_1838 | H'0010_ 0807_ 0000_ C010 | H'0010_ 0404_ 0000_ C010 | H'0010_ 0432_ 0000_ C010 | — |
| H'E67E_1840 | H'0014_ 1007_ 0000_ FFF0 | H'0014_ 1409_ 0000_ FFF0 | H'0014_ 0C5D_ 0000_ FFF0 | — |
| H'E67E_1858 | H'0014_ 100D_ 0000_ C010 | H'0014_ 0C0A_ 0000_ C010 | H'0014_ 0C94_ 0000_ C010 | — |
| H'E67E_1860 | H'0014_ 100D_ 0000_ C010 | H'0014_ 0C0A_ 0000_ C010 | H'0014_ 0C94_ 0000_ C010 | — |
| H'E67E_1878 | H'0000_ 0000_ 0000_ 0000 | H'0010_ 0403_ 0000_ C010 | H'0000_ 0000_ 0000_ 0000 | — |
| H'E67E_1888 | H'0104_ 1007_ 0000_ FF00 | H'0014_ 1409_ 0000_ FFF0 | H'0014_ 041F_ 0000_ FFF0 | — |
| H'E67E_1890 | H'0000_ 0000_ 0000_ 0000 | H'0014_ 0807_ 0000_ C010 | H'0000_ 0000_ 0000_ 0000 | — |
| H'E67E_1A60 | H'000C_ 0802_ 0000_ FFF0 | H'000C_ 0802_ 0000_ FFF0 | H'000C_ 0815_ 0000_ FFF0 | — |

Table 21.3 Setting values of QOSBW_FIX_QOS_BANK1 registers (2/2)

| Address | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|------------------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|
| H'E67E_1A68 | H'0014_ 0801_ 0000_ FFF0 | H'0014_ 0801_ 0000_ FFF0 | H'0014_ 0802_ 0000_ FFF0 | — |
| H'E67E_1A70 | H'0014_ 0401_ 0000_ FFF0 | H'0014_ 0401_ 0000_ FFF0 | H'0014_ 0401_ 0000_ FFF0 | — |
| H'E67E_1A78 | H'000C_ 0401_ 0000_ FFF0 | H'000C_ 0401_ 0000_ FFF0 | H'000C_ 0409_ 0000_ FFF0 | — |
| H'E67E_1A90 | H'0014_ 0801_ 0000_ FFF0 | H'0014_ 0801_ 0000_ FFF0 | H'0014_ 0802_ 0000_ FFF0 | — |
| H'E67E_1A98 | H'0014_ 0401_ 0000_ FFF0 | H'0014_ 0401_ 0000_ FFF0 | H'0014_ 0401_ 0000_ FFF0 | — |
| Other Address | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 |

Notes: — : Not supported

*1: Set values generated by a tool released from Renesas

21.2.40 QOSBW BE QOS BANK0 Register n

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[RZ/G2H] n: 0 to 365, Address range: H'E67E_2000 to H'E67E_2B68

[RZ/G2M V1.3, RZ/G2M V3.0] n: 0 to 362, Address range: H'E67E_2000 to H'E67E_2B50

[RZ/G2N] n: 0 to 370, Address range: H'E67E_2000 to H'E67E_2B90

[RZ/G2E] n: 0 to 255, Address range: H'E67E_2000 to H'E67E_27F8

| | | | | | | | | | | | | | | | | |
|----------------|---------------|-----|-----|-----|---------------|-----|---------------|-----|-----|-----|---------------|-----|---------------|-----|-----|-----|
| Bit: | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 | 48 |
| | — | — | — | — | — | — | — | — | — | — | — | — | BEQOS_BANK0_4 | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 47 | 46 | 45 | 44 | 43 | 42 | 41 | 40 | 39 | 38 | 37 | 36 | 35 | 34 | 33 | 32 |
| | — | — | — | — | BEQOS_BANK0_3 | | | | | | BEQOS_BANK0_2 | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | BEQOS_BANK0_2 | | | | | | | | | | BEQOS_BANK0_1 | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | BEQOS_BANK0_1 | | | | | | BEQOS_BANK0_0 | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 63 to 57 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 56 to 54 | — | — | R/W | Reserved The write value should always be 0. |
| 53, 52 | BEQOS_BANK0_5 | — | R/W | BEQOS_BANK0_5 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. The range of these bits is B'00 to B'10. |
| 51 to 48 | BEQOS_BANK0_4 | — | R/W | BEQOS_BANK0_4 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |
| 47 to 44 | — | — | R/W | Reserved The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 43 to 36 | BEQOS_BANK0_3 | — | R/W | BEQOS_BANK0_3 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |
| 35 to 20 | BEQOS_BANK0_2 | — | R/W | BEQOS_BANK0_2 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |
| 19 to 10 | BEQOS_BANK0_1 | — | R/W | BEQOS_BANK0_1 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |
| 9 to 0 | BEQOS_BANK0_0 | — | R/W | BEQOS_BANK0_0 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |

Notes: This register must be accessed by 8byte access.

This register must be written during statqen bit of QOSCTRL_STATQC register is 0 or exe_membank of QOSCTRL_MEMBANK is 1.

Refer to the Table 21.4 for setting values of QOSBW BE QOS BANK0 registers.

Table 21.4 Setting values of QOSBW_BE_QOS_BANK0 registers

| Address | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|
| H'E67E_2000 to H'E67E_2350 | *1 | *1 | *1 | *1 |
| H'E67E_2358 to H'E67E_2368 | *1 | H'0000_ 0000_ 0000_ 0000 | *1 | *1 |
| H'E67E_2370 to H'E67E_2390 | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 | *1 | *1 |
| Other Address | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 |

Notes: *1: Set values generated by a tool released from Renesas

21.2.41 QOSBW BE QOS BANK1 Register n

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[RZ/G2H] n: 0 to 365, Address range: H'E67E_3000 to H'E67E_3B68

[RZ/G2M V1.3, RZ/G2M V3.0] n: 0 to 362, Address range: H'E67E_3000 to H'E67E_3B50

[RZ/G2N] n: 0 to 370, Address range: H'E67E_3000 to H'E67E_3B90

[RZ/G2E] n: 0 to 255, Address range: H'E67E_3000 to H'E67E_37F8

| | | | | | | | | | | | | | | | | |
|----------------|---------------|-----|-----|-----|---------------|-----|---------------|-----|-----|-----|---------------|-----|-----|-----|-----|-----|
| Bit: | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 | 48 |
| | — | — | — | — | — | — | — | — | — | — | BEQOS_BANK1_5 | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 47 | 46 | 45 | 44 | 43 | 42 | 41 | 40 | 39 | 38 | 37 | 36 | 35 | 34 | 33 | 32 |
| | — | — | — | — | BEQOS_BANK1_3 | | | | | | BEQOS_BANK1_2 | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | BEQOS_BANK1_2 | | | | | | | | | | BEQOS_BANK1_1 | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | BEQOS_BANK1_1 | | | | | | BEQOS_BANK1_0 | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 63 to 57 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 56 to 54 | — | — | R/W | Reserved The write value should always be 0. |
| 53, 52 | BEQOS_BANK1_5 | — | R/W | BEQOS_BANK1_5 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. The range of these bits is B'00 to B'10. |
| 51 to 48 | BEQOS_BANK1_4 | — | R/W | BEQOS_BANK1_4 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |
| 47 to 44 | — | — | R/W | Reserved The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 43 to 36 | BEQOS_BANK1_3 | — | R/W | BEQOS_BANK1_3 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |
| 35 to 20 | BEQOS_BANK1_2 | — | R/W | BEQOS_BANK1_2 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |
| 19 to 10 | BEQOS_BANK1_1 | — | R/W | BEQOS_BANK1_1 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |
| 9 to 0 | BEQOS_BANK1_0 | — | R/W | BEQOS_BANK1_0 Setting. These bits should be a value generated by a tool released from Renesas. A value other than above must not be set to these bits. |

Notes: This register must be accessed by 8byte access.

This register must be written during statqen bit of QOSCTRL_STATQC register is 0 or exe_membank of QOSCTRL_MEMBANK is 0.

Refer to the Table 21.5 for setting values of QOSBW BE QOS BANK1 registers.

Table 21.5 Setting values of QOSBW_BE_QOS_BANK1 registers

| Address | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|
| H'E67E_3000 to H'E67E_3350 | *1 | *1 | *1 | *1 |
| H'E67E_3358 to H'E67E_3368 | *1 | H'0000_ 0000_ 0000_ 0000 | *1 | *1 |
| H'E67E_3370 to H'E67E_3390 | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 | *1 | *1 |
| Other Address | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 | H'0000_ 0000_ 0000_ 0000 |

Notes: *1: Set values generated by a tool released from Renesas

21.2.42 CPU Domain Active Register n (CPU_ACTn) (n: 0 to 3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CPU_ACT2 and 3 are not supported by RZ/G2N and RZ/G2E

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | ACT_CYCLE | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 4 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0 |
| 3 to 0 | ACT_CYCLE | H'0 | R/W | Request signal extension Set H'3 on these bits. Notice: This register can not be accessed from other Master which is belonging to non-CPU domain on RZ/G2H. |

21.2.43 Main Memory Secure Error Status Register (MMSECERRST)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|---------------|---------------|----------------|----------------|----------------|----------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | SECG0R EGWP | SECG1R EGWP | SECG2R EGWP | SECG3R EGWP | — | — | — | — | — | — | LOCK1 | LOCK0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R/W1C | R/W1C |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | SECER R6_R | SECER R6_W | SECER R5_R | SECER R5_W | SECER R4_R | SECER R4_W | SECER R3_R | SECER R3_W | SECER R2_R | SECER R2_W | SECER R1_R | SECER R1_W | SECER R0_R | SECER R0_W |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W1C | R/W1C | R/W1C | R/W1C | R/W1C | R/W1C | R/W1C | R/W1C | R/W1C | R/W1C | R/W1C | R/W1C | R/W1C | R/W1C |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-------|--|
| 31 to 28 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0 |
| 27 | SECG0REGWP | B'0 | R/W | Secure Group0 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 26 | SECG1REGWP | B'0 | R/W | Secure Group1 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 25 | SECG2REGWP | B'0 | R/W | Secure Group2 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 24 | SECG3REGWP | B'0 | R/W | Secure Group3 Write Privilege Setting for this register 0: Has the privilege to write to this register 1: Does not have the privilege write to this register Secure Master is not restricted by this setting. |
| 23 to 18 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0 |
| 17 | LOCK1 | B'0 | R/W1C | Secure error status and information for main memory AXI I/F1-6 is locked |
| 16 | LOCK0 | B'0 | R/W1C | Secure error status and information for main memory AXI I/F0 is locked |
| 15, 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 | SECERR6_R | B'0 | R/W1C | Secure Error detect of main memory AXI I/F6 read access for DRAM/System RAM |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------|---------------|-------|--|
| 12 | SECERR6_W | B'0 | R/W1C | Secure Error detect of main memory AXI I/F6 write access for DRAM/System RAM |
| 11 | SECERR5_R | B'0 | R/W1C | Secure Error detect of main memory AXI I/F5 read access for DRAM/System RAM |
| 10 | SECERR5_W | B'0 | R/W1C | Secure Error detect of main memory AXI I/F5 write access for DRAM/System RAM |
| 9 | SECERR4_R | B'0 | R/W1C | Secure Error detect of main memory AXI I/F4 read access for DRAM/System RAM |
| 8 | SECERR4_W | B'0 | R/W1C | Secure Error detect of main memory AXI I/F4 write access for DRAM/System RAM |
| 7 | SECERR3_R | B'0 | R/W1C | Secure Error detect of main memory AXI I/F3 read access for DRAM/System RAM |
| 6 | SECERR3_W | B'0 | R/W1C | Secure Error detect of main memory AXI I/F3 write access for DRAM/System RAM |
| 5 | SECERR2_R | B'0 | R/W1C | Secure Error detect of main memory AXI I/F2 read access for DRAM/System RAM |
| 4 | SECERR2_W | B'0 | R/W1C | Secure Error detect of main memory AXI I/F2 write access for DRAM/System RAM |
| 3 | SECERR1_R | B'0 | R/W1C | Secure Error detect of main memory AXI I/F1 read access for DRAM/System RAM |
| 2 | SECERR1_W | B'0 | R/W1C | Secure Error detect of main memory AXI I/F1 write access for DRAM/System RAM |
| 1 | SECERR0_R | B'0 | R/W1C | Secure Error detect of main memory AXI I/F0 read access for DRAM/System RAM |
| 0 | SECERR0_W | B'0 | R/W1C | Secure Error detect of main memory AXI I/F0 write access for DRAM/System RAM |

21.2.44 Main Memory Secure Error Address Register0 (MMSECERRADDR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SECERRADDR0[37:22] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SECERRADDR0[21:6] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|--|
| 31 to 0 | SECERRADDR0 | H'0000_0000 | R | Secure error address [37:6] of main memory AXI I/F0 Write 1 to MMSECERRST.LOCK0 for clear |

21.2.45 Main Memory Secure Error Address Register1 (MMSECERRADDR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SECERRADDR1[37:22] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SECERRADDR1[21:6] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|--|
| 31 to 0 | SECERRADDR1 | H'0000_0000 | R | Secure error address [37:6] of main memory AXI I/F1-6 Write 1 to MMSECERRST.LOCK1 for clear |

21.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

21.3.1 DRAM Split/Linear Address mapping

The AXI-bus provides the mixed address space interleaved address mapping (Split mapping) and no-interleaved address mapping (Linear mapping) on memory controllers through **ADSPLCRn** register. The split mapping mode enhances DRAM access efficiency and bandwidth via parallel access to memory controllers. In this mode, allocated to each DBSC channel at the specified granularity. The linear mapping mode allocate a contiguous space to each memory controller. It can be divided into 8 areas. And also, Linear or Split mapping can be chosen in each area. Figure 21.5 and Figure 21.6 shown Split/Linear Address mapping.

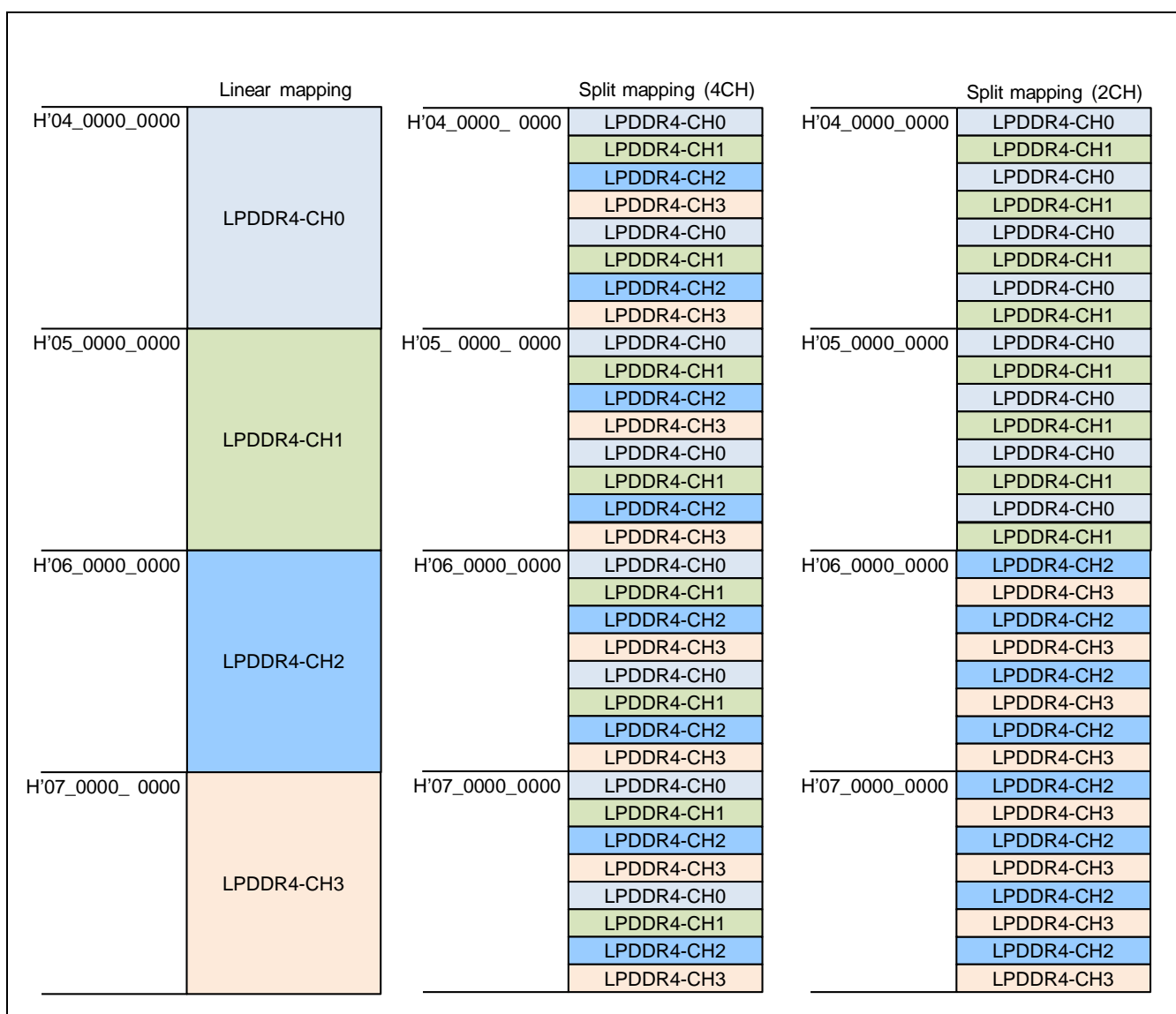


Figure 21.5 Concept of Split/Linear Address mapping [RZ/G2H]

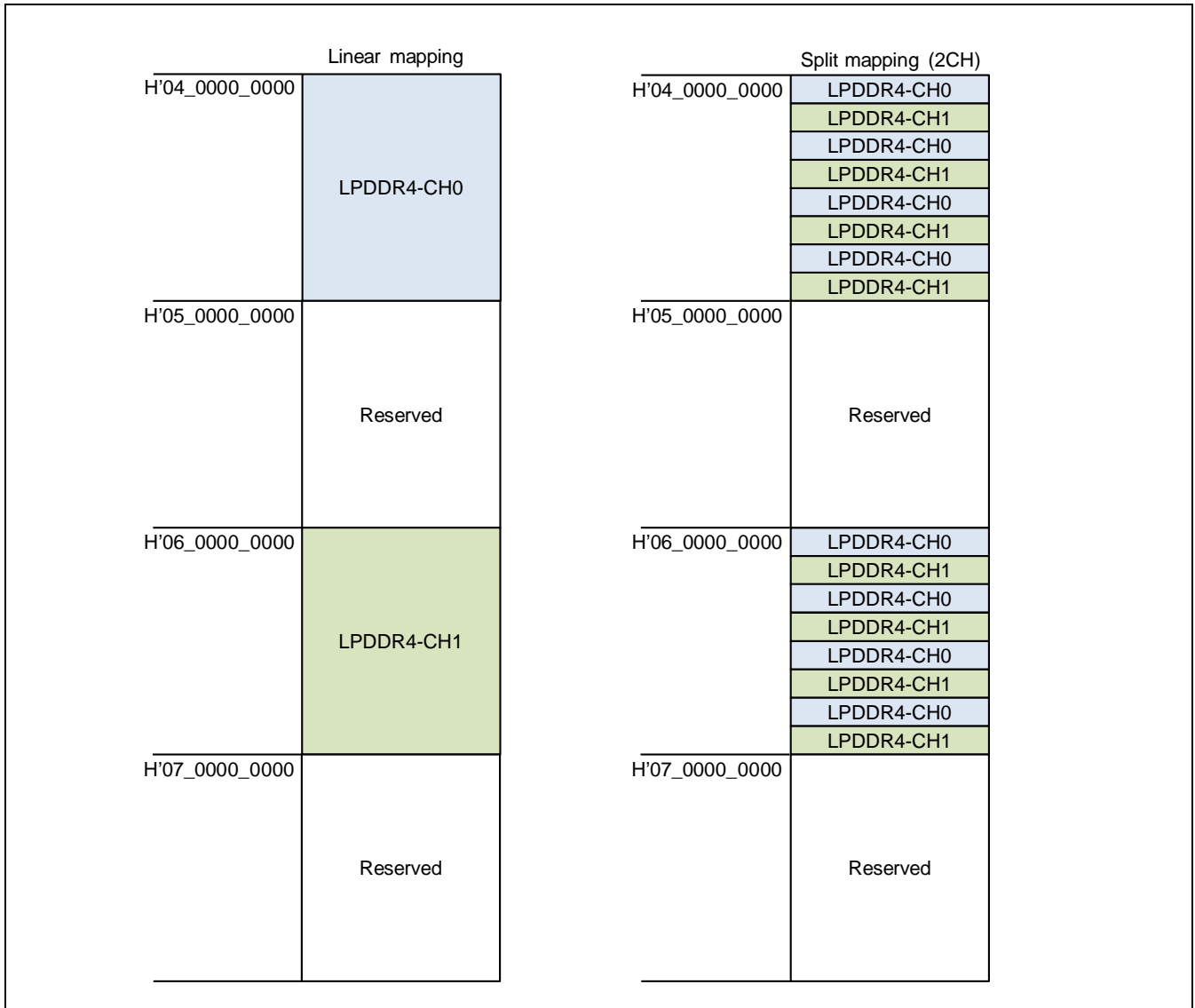


Figure 21.6 Concept of Split/Linear Address mapping [RZ/G2M V1.3, RZ/G2M V3.0]

21.3.2 Decompression of visual near lossless compressed image

AXI-bus supports decompression of visual near lossless compressed image. IPs can read compressed data by accessing to setting area through **DCMPAREACRAn/DCMPAREACRBn** or **DCMPSHDWCR** register.

21.3.3 QoS arbitration

AXI-bus has the QoS arbitration to guarantee latency and bandwidth. A concept of the time shared control which is being called time division slot allocation is introduced. The arbitration can be controlled by assigning bandwidth of each IP to time slot.

21.3.4 Access protection for secure/safety regions

21.3.4.1 Memory Protection

Main Memory domain AXI-bus has memory protected area function. This function as shown in Figure 21.7 is that DRAM or System RAM area can be divided into 16 areas and each area and each area can be set access privilege to 4 secure groups.

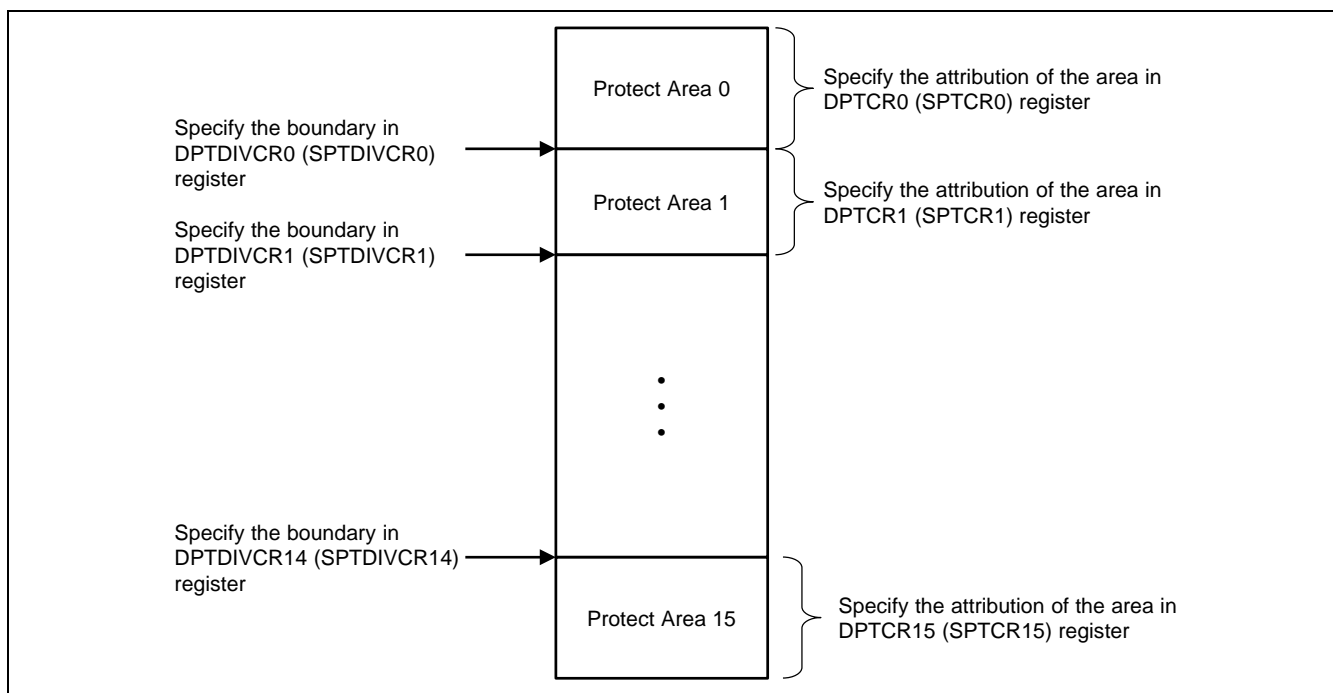


Figure 21.7 Memory Protected Area Function Concept

21.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

21.4.1 Change of the initial value of the register

The initial value of the following private register should be changed before using the LSI.

Write H'0000_000A to the address field H'E67F_0048 with 32-bit access.

Note: The initial value of this address field is H'0080_000A.

22. IPMMU

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

22.1 Overview

The IPMMU is a Memory Management Unit (MMU) which provides address translation and access protection functionalities to processing units and interconnect networks.

22.1.1 Features

IPMMU includes the following main features.

- MMU architecture compatible with Armv8 VMSAv8-64 and VMSAv8-32 including the Large Physical Address Extension (LPAE), the Security Extensions and the Multiprocessing Extensions. IPMMU supports eight sets of TLBs and these can be managed independently.
- PMB* address translation functionality. IPMMU supports sixteen domains.
- Caching recently used page table entries in translation look-aside buffer (TLB)
- Performance monitoring for Armv8 VMSAv8-64 and VMSAv8-32 LPAE
- DVM messages are not supported.

For more information about Armv8 VMSA, see Arm Architecture Reference Manual Armv8, for Armv8-A architecture profile.

In VMSAv8-64, supported memory translation granule size is 4KB and 64KB. 16KB granule size is not supported in IPMMU. And 8bit ASID size is only supported. (not support 16bit size)

In VMSAv8-32, 40bit IPA can be handled by IPMMU.

Note: * PMB (Privileged space Mapping Buffer) is the register specified address translation mechanism. It supports up to sixteen translation tables and four types of tables (16 MB, 64 MB, 128 MB and 521 MB) can be set.

22.1.2 Block Diagram

Figure 22.1 shows the IPMMU block diagram. When a master (a processing unit or an interconnect) issues a transaction, a micro-TLB connected to the master sends a virtual address to the IPMMU cache which is connected to the same bus domain. If the virtual address is not cached in the IPMMU cache, the IPMMU performs a page table walk to the main IPMMU which is located at the main memory bus domain through an interconnect and returns a physical address to the micro-TLB.

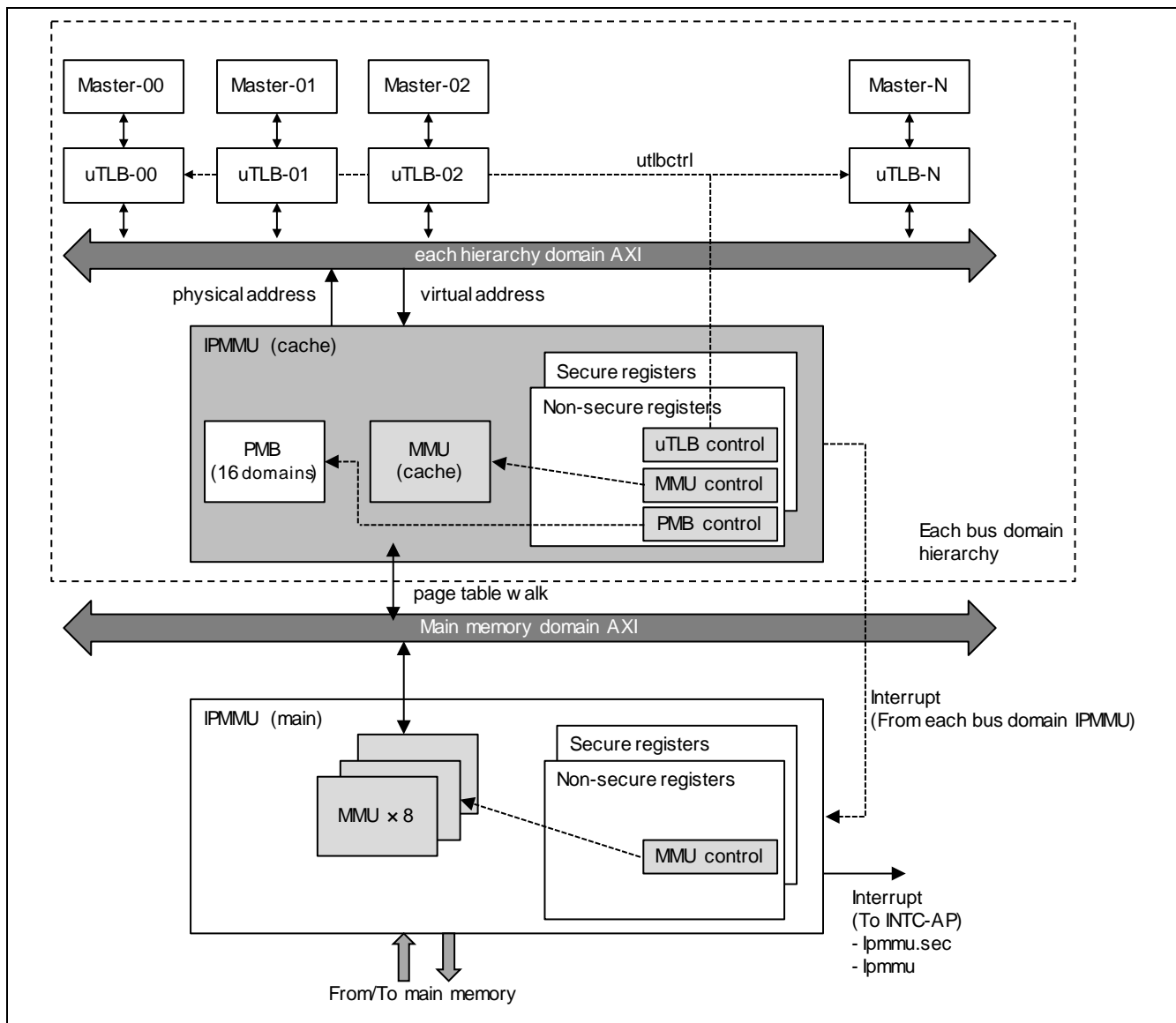


Figure 22.1 IPMMU System Integration

22.1.3 External Pins

There are no external pins in this module.

22.1.4 Register Configuration

Table 22.1 shows the register address map of IPMMU. These registers are 32-bit access registers. When modifying reserved bits or read-only bits, write 0 to the bits.

Table 22.1 Register Configurations

| | | | | | | Second Generation RZ/G Series Products | | | | |
|--|--------------|-----|----------------------|-----------------|------|---|-------------|-------------|--------|--------|
| Register Name | Abbreviation | R/W | Offset Address | Initial Value | Size | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| MMU Control Register n (n = 0 to 7) | IMCTRn*1 | R/W | H'0000 + H'40 × n | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Translation Table Base Control Register n | IMTTBCRn*1 | R/W | H'0008 + H'40 × n | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Translation Table Lower Base Register 0n | IMTTLBR0n*1 | R/W | H'0010 + H'40 × n | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Translation Table Upper Base Register 0n | IMTTUBR0n*1 | R/W | H'0014 + H'40 × n | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Translation Table Lower Base Register 1n | IMTTLBR1n*1 | R/W | H'0018 + H'40 × n | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Translation Table Upper Base Register 1n | IMTTUBR1n*1 | R/W | H'001C + H'40 × n | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Error Status Register n | IMSTRn*1 | R/W | H'0020 + H'40 × n | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Memory Attribute Indirection Register 0n | IMMAIR0n*1 | R/W | H'0028 + H'40 × n | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Memory Attribute Indirection Register 1n | IMMAIR1n*1 | R/W | H'002C + H'40 × n | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Error Lower Address Register n | IMELARn*1 | R | H'0030 + H'40 × n | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Error Upper Address Register n | IMEUARn*1 | R | H'0034 + H'40 × n | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| PMB Control Register | IMPCTR*1 | R/W | H'0200 | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| PMB Status Register | IMPSTR*1 | R/W | H'0208 | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| PMB Error Address Register | IMPEAR*1 | R | H'020C | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| PMB Address Array n (n = 0 to 15) | IMPMBAn*1 | R/W | H'0280 to H'02BC | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| PMB Data Array n | IMPMBDn*1 | R/W | H'02C0 to H'02FC | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| uTLB Control Register n (n = 0 to 31) | IMUCTRn*1 | R/W | H'0300 + H'10 × n | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |

| | | | | | | Second Generation RZ/G Series Products | | | | |
|---|----------------|-----|-----------------------------|-----------------|------|---|-------------|-------------|--------|--------|
| Register Name | Abbreviation | R/W | Offset Address | Initial Value | Size | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| uTLB Control Register n (n = 32 to 47) | IMUCTRn*1 | R/W | H'0600 + H'10 × (n - 32) | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| uTLB ASID Register n (n = 0 to 31) | IMUASIDn*1 | R/W | H'0308 + H'10 × n | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| uTLB ASID Register n (n = 32 to 47) | IMUASIDn*1 | R/W | H'0608 + H'10 × (n - 32) | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU System Control Register | IMSCTLR*3 | R/W | H'0500 | H'0000_ 0000 | 32 | — | √ | — | — | — |
| MMU System Control Register | IMSCTLR*3 | R/W | H'0500 | — | 32 | √ | — | √ | √ | √ |
| MMU Auxiliary Control Register | IMSAXCTL R*3 | R/W | H'0504 | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Interrupt Status Register | IMSSTR | R | H'0540 | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Performance Monitor Control Register | IMPFCMTR*4 | R/W | H'0580 | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Performance Monitor Total Translation Counter | IMPFCM TOTAL*4 | R | H'0590 | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Performance Monitor Hit Counter | IMPFCMHIT*4 | R | H'0594 | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Performance Monitor L3 Miss Counter | IMPFCML3MIS S | R | H'0598 | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Performance Monitor L2 Miss Counter | IMPFCML2MIS S | R | H'059C | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |
| MMU Performance Monitor Miss Counter | IMPFCMMISS | R | H'0598 | H'0000_ 0000 | 32 | √ | — | √ | √ | √ |
| Address alias of Non-secure registers *2 | — | R/W | H'0800 to H'0FFC | H'0000_ 0000 | 32 | √ | √ | √ | √ | √ |

- Notes: 1. These registers are composed of the registers banked for security. This provides a separate secure and non-secure copies of the registers.
2. Non-secure registers (H'0800 to H'0FFC) can be accessed through this address space in Secure mode. In Non-secure mode, this address space is always read as 0.
3. These registers can be accessed by secure mode only.
4. IMPFCMTR, IMPFCMTOTAL, IMPFCMHIT, IMPFCMMISS are not supported in IPMMU-cache of RZ/G2M V1.3, RZ/G2M V3.0.

Table 22.2 shows the each IPMMU base address and the power domain which belong to. This table also shows which IPMMU is supported to each RZ/G product. Basically, each IPMMU is connected to the each BUS modules. But the number of IPMMU is different between each RZ/G product. Except the IPMMU in Always-On domain, power-up sequence is required before initialization of IPMMU. About the power domain and the procedure for power-up sequence of the corresponding power domain, see the specification of SYSC.

Table 22.2 IPMMU Base Address and Power domain

| MMU | Power Domain | Base Address | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----------|--------------|--------------|--------|-------------|-------------|--------|--------|
| IPMMU-VI0 | Always-On | H'FEBD_0000 | √ | √ | √ | √ | √ |
| IPMMU-VI1 | Always-On | H'FEBE_0000 | √ | — | — | — | — |
| IPMMU-VP0 | A3VP | H'FE99_0000 | √ | — | — | √ | — |
| | Always-On | | — | — | — | — | √ |
| IPMMU-VP1 | A3VP | H'FE98_0000 | √ | — | — | — | — |
| IPMMU-VC0 | A3VC | H'FE6B_0000 | √ | √ | √ | √ | √ |
| | Always-On | | — | — | — | — | — |
| IPMMU-VC1 | A3VC | H'FE6F_0000 | √ | — | — | — | — |
| IPMMU-PV0 | Always-On | H'FD80_0000 | √ | √ | √ | √ | √ |
| IPMMU-PV1 | Always-On | H'FD95_0000 | √ | √ | √ | — | — |
| IPMMU-PV2 | Always-On | H'FD96_0000 | √ | — | — | — | — |
| IPMMU-PV3 | Always-On | H'FD97_0000 | √ | — | — | — | — |
| IPMMU-IR | A3IR | H'FF8B_0000 | √ | √ | √ | — | — |
| IPMMU-HC | Always-On | H'E657_0000 | √ | √ | √ | √ | √ |
| IPMMU-MP | Always-On | H'EC67_0000 | √ | √ | √ | √ | √ |
| IPMMU-DS0 | Always-On | H'E674_0000 | √ | √ | √ | √ | √ |
| IPMMU-DS1 | Always-On | H'E774_0000 | √ | √ | √ | √ | √ |
| IPMMU-MM | Always-On | H'E67B_0000 | √ | √ | √ | √ | √ |

Note: √: Supported, —: Not Supported

Table 22.3 shows the register state in each processing mode.

Table 22.3 Register State in Each Processing Mode

| Register Abbreviation | Reset*1 | Initial value | Second Generation RZ/G Series Products | | | | |
|-----------------------|-------------|---------------|--|-------------|-------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| IMCTRn | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMTTBCRn | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMTTLBR0n | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMTTUBR0n | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMTTLBR1n | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMTTUBR1n | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMSTRn | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMMAIR0n | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMMAIR1n | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMELARn | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMEUARn | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMPCTR | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMPSTR | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMPEAR | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMPMBAA00 to 15 | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMPMBD00 to 15 | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMUCTRn | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMUASIDn | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMSCTLR | Initialized | H'0000_0000 | — | √ | — | — | — |
| IMSCTLR | Initialized | — | √ | — | √ | √ | √ |
| IMSAUXCTLR | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMSSTR | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMPFMCTR*2 | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMPFMTOTAL*2 | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMPFMHIT*2 | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMPFML3MISS | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMPFML2MISS | Initialized | H'0000_0000 | √ | √ | √ | √ | √ |
| IMPFMMISS*2 | Initialized | H'0000_0000 | √ | — | √ | √ | √ |

Notes: 1. Refer to section 17, RESET (RST).

2. IMPFMCTR, IMPFMTOTAL, IMPFMHIT, IMPFMMISS are not supported in IPMMU-cache of RZ/G2M V1.3, RZ/G2M V3.0.

As described in Figure 22.1, IPMMU is composed of two parts. IPMMU in each bus domain hierarchy is caching pre-decoded address translation information. And also managed the microTLB and support PMB function. On the other hand, IPMMU in main memory bus domain is managed MMU function only. Table 22.4 shows the registers which each IPMMU supports.

Table 22.4 Supported Registers

| Register Abbreviation | IPMMU(cache) | IPMMU(main) |
|-----------------------|--------------|-------------|
| IMCTRn | √ | √ |
| IMTTBCRn | — | √ |
| IMTTLBR0n | — | √ |
| IMTTUBR0n | — | √ |
| IMTTLBR1n | — | √ |
| IMTTUBR1n | — | √ |
| IMSTRn | — | √ |
| IMMAIR0n | — | √ |
| IMMAIR1n | — | √ |
| IMELARn | — | √ |
| IMEUARn | — | √ |
| IMPCTR | √ | — |
| IMPSTR | √ | — |
| IMPEAR | √ | — |
| IMPMBAA00 to 15 | √ | — |
| IMPMBD00 to 15 | √ | — |
| IMUCTRn | √ | — |
| IMUASIDn | √ | — |
| IMSCTLR | √ | √ |
| IMSAUXCTLR | — | √ |
| IMSSTR | — | √ |
| IMPFMCTR* | √ | √ |
| IMPFMTOTAL* | √ | √ |
| IMPFMHIT* | √ | √ |
| IMPFML3MISS | — | √ |
| IMPFML2MISS | — | √ |
| IMPFMMISS* | √ | — |

Note: √: Supported, —: Not Supported

IPMMU (cache): IPMMU-VPn, VIn, VCn, PVn, MP, IR, HC, DSn

IPMMU (main): IPMMU-MM

Note: * IMPFMCTR, IMPFMTOTAL, IMPFMHIT, IMPFMMISS are not supported in IPMMU-cache of RZ/G2M V1.3, RZ/G2M V3.0.

22.1.5 Connected Module

Table 22.5 shows the connected modules to IPMMU. As for the feature of the connected modules, refer to the chapter of each modules.

Table 22.5 Connected Modules

| Module Name | Connected Module Name | Connected Module Function |
|--------------------|------------------------------|----------------------------------|
| IPMMU | AXI-BUS | Bus |
| | SYSC | Clock Supply |
| | RST | Module Reset |
| | INTC-AP | Interrupt handling |

22.2 Register Description

22.2.1 MMU Control Register n (IMCTRn)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

Note: n = 0 to 7

This register controls the behavior of the MMU.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|------|----|----|----|----|----|----|------------|-----|------|-------|-------|-----------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | VA64 | — | — | — | — | — | — | — | — | — | — | — | TRE | AFE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | RTSEL[2:0] | | TREN | INTEN | FLUSH | MMUE N | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 | VA64* | B'0 | R/W | AArch64 support 1: VMSAv8-64 mode 0: VMSAv8-32 mode This register can be set IPMMU-MM only |
| 28 to 18 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 17 | TRE* | B'0 | R/W | TEX Remap Enable This field is used when EAE is 0. 0: TEX remap disabled. 1: TEX remap enabled. This register can be set IPMMU-MM only |
| 16 | AFE* | B'0 | R/W | Access Flag Enable 0: Behave as if AF bit is always set to 1. 1: Enable software management of the Access Flag. This register can be set IPMMU-MM only |
| 15 to 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 to 4 | RTSEL[2:0]* | B'000 | R/W | Retranslation Table Select When TREN is 1, RTSEL indicates the table number to retranslate. This register can be set IPMMU-MM only |
| 3 | TREN* | B'0 | R/W | MMU Retranslation Enable 0: Output PA as a physical address. 1: Output PA as an intermediate physical address to retranslate through 40-bit TLB. This register can be set IPMMU-MM only |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 2 | INTEN | B'0 | R/W | Interrupt Enable 0: Don't assert an interrupt when an error occurs. 1: Assert an interrupt when an error occurs. |
| 1 | FLUSH | B'0 | R/W | TLB Invalidate This bit is automatically cleared to 0. So read value is always 0. 1: Invalidate all TLB entries instantaneously. |
| 0 | MMUEN | B'0 | R/W | MMU Enable. If 0 is set, IPMMU and micro-TLBs which are belong to this IPMMU are disabled. 0: MMU disabled 1: MMU enabled |

Note: * These fields are implemented in IPMMU-MM only

22.2.2 MMU Translation Table Base Control Register n (IMTTBCRn)

| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|-------------|-------------|--------|--------|
| √ | √ | √ | √ | √ |

Note: n = 0 to 7

This register controls the attribute of TLB managed by each MMU.

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|-----|---------------|----------|-----|------------|-----|------------|-----|------|------|-----------|-----|-----|-----|-----|-----|
| | EAE | PMB/ BYPEN | SH1[1:0] | | ORGN1[1:0] | | IRGN1[1:0] | | PGSZ | SCSZ | TSZ1[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------------|----|----|----------|-----|------------|-----|------------|-----|---------|-----|-----------|-----|-----|-----|-----|-----|
| | — | — | SH0[1:0] | | ORGN0[1:0] | | IRGN0[1:0] | | SL[1:0] | | TSZ0[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------------|---------------|-----|--|
| 31 | EAE*1 | B'0 | R/W | Extended Address Enable 0: Enable the 32-bit translation system with the Armv8 Short-descriptor translation table format or PMB address translation. 1: Enable the 40-bit translation System with the Armv8 Long-descriptor translation table format. |
| 30 | PMB/ BYPEN | B'0 | R/W | [EAE = 0] PMB Enable 0: Disable PMB address translation. 1: Enable PMB address translation in the virtual address range from H'8000_0000 to H'BFFF_FFFF. This field is used when EAE is 0. [EAE = 1] Bypass stage 1 translation 0: Not bypass stage 1 translation when stage 2 translation is performed (default) 1: Bypass stage 1 translation when only stage 2 translation is performed. The meaning of this field is different in case that EAE bit is set. For details, see 22.3.3 Address Space (4) stage 2 translation. |
| 29, 28 | SH1[1:0]*4 | B'00 | R/W | Share ability attributes for the memory associated with the translation table walks using TTBR1n. B'00: Non-shareable B'01: Reserved B'10: Outer shareable B'11: Inner shareable |
| 27, 26 | ORGN1[1:0] *4 | B'00 | R/W | Outer Cache ability attributes for the memory associated with the translation table walks using TTBR1n. B'00: Normal memory, Outer Non-cacheable B'01: Normal memory, Outer Write-Back Write-Allocate Cacheable B'10: Normal memory, Outer Write-Through Cacheable B'11: Normal memory, Outer Write-Back no Write-Allocate Cacheable |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|---|
| 25, 24 | IRGN1[1:0] *4 | B'00 | R/W | Inner Cache ability attributes for the memory associated with the translation table walks using TTBR1n. B'00: Normal memory, Inner Non-cacheable B'01: Normal memory, Inner Write-Back Write-Allocate Cacheable B'10: Normal memory, Inner Write-Through Cacheable B'11: Normal memory, Inner Write-Back no Write-Allocate Cacheable |
| 23 | PGSZ | B'0 | R/W | [VMSAv8-32, EAE = 0] 2nd level Page Size Select 1: Support large page size (64KB page) 0: Support small page size (4KB page) [VMSAv8-64] 3rd level Page Size Select 1: 64KB page granule 0: 4KB page granule (default) |
| 22 | SCSZ | B'0 | R/W | [VMSAv8-32, EAE = 0] 1st level page size select 1: Support super section size (16MB) 0: Support section size (1MB) |
| 21 to 16 | TSZ1[5:0]*3 | B'00_0000 | R/W | [VMSAv8-64] - TSZ1[5:0] is used for the size offset of the TTBR0n addressed region, encoded as a 6-bit unsigned number, giving the size of the region as $2^{(64-TSZ1)}$. [VMSAv8-32, stage 1 translation]: - TSZ1[5:3] is reserved. This bit is always read as 0. The write value should always be 0. - TSZ1[2:0] is used for the size offset of the TTBR0n addressed region, encoded as a 3-bit unsigned number, giving the size of the region as $2^{(32-TSZ1)}$. [VMSAv8-32, stage 2 translation]: - TSZ1[5:0] is reserved. This bit is always read as 0. The write value should always be 0. |
| 15, 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13, 12 | SH0[1:0] *4 | B'00 | R/W | Share ability attributes for the memory associated with the translation table walks using TTBR0n. B'00: Non-shareable B'01: Reserved B'10: Outer shareable B'11: Inner shareable |
| 11, 10 | ORGN0[1:0] *4 | B'00 | R/W | Outer Cache ability attributes for the memory associated with the translation table walks using TTBR0n. B'00: Normal memory, Outer Non-cacheable B'01: Normal memory, Outer Write-Back Write-Allocate Cacheable B'10: Normal memory, Outer Write-Through Cacheable B'11: Normal memory, Outer Write-Back no Write-Allocate Cacheable |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------------|---------------|-----|--|
| 9, 8 | IRGN0[1:0] *4 | B'00 | R/W | <p>Inner Cache ability attributes for the memory associated with the translation table walks using TTBR0n.</p> <p>B'00: Normal memory, Inner Non-cacheable B'01: Normal memory, Inner Write-Back Write-Allocate Cacheable B'10: Normal memory, Inner Write-Through Cacheable B'11: Normal memory, Inner Write-Back no Write-Allocate Cacheable</p> |
| 7, 6 | SL[1:0]*2 | B'00 | R/W | <p>Starting level for translation table walks.</p> <p>[VMSAv8-32] SL[1]: 1: Start at first level 0: Start at second level.</p> <p>[VMSAv8-64, PGSZ = 0] SL[1:0] B'00: Start at second level B'01: Start at first level</p> <p>[VMSAv8-64, PGSZ = 1] SL[1:0] B'00: Start at third level B'01: Start at second level</p> |
| 5 to 0 | TSZ0[5:0] | B'00_0000 | R/W | <p>[VMSAv8-64].</p> <ul style="list-style-type: none"> - TSZ0[5:0] is used for the size offset of the TTBR0n addressed region, encoded as a 6-bit unsigned number, giving the size of the region as $2^{(64-TSZ0)}$. <p>[VMSAv8-32, stage 1 translation]:</p> <ul style="list-style-type: none"> - TSZ0[5:3] is reserved. This bit is always read as 0. The write value should always be 0. - TSZ0[2:0] is used for the size offset of the TTBR0n addressed region, encoded as a 3-bit unsigned number, giving the size of the region as $2^{(32-TSZ0)}$. <p>[VMSAv8-32, stage 2 translation]:</p> <ul style="list-style-type: none"> - TSZ0[5:4] is reserved. This bit is always read as 0. The write value should always be 0. - TSZ0[3:0] is used for the size offset of the TTBR0n addressed region, encoded as a 4-bit signed number, giving the size of the region as $2^{(32-TSZ0)}$. |

- Notes:
1. Even if IMCTRn.VA64 bit is set, this field need to be set in case of using long descriptor format.
 2. About Start Level, refer to "22.5.3.4 Start Level for Page Table Walk".
 3. If TSZ1 = 24 is set, all page table walk request will be issued via TTBR1. This setting is prohibited. From the hardware point of view, each IP (except ARM-CPU) can be treated the VA up to 40-bit. In case that TSZ0 and TSZ1 are set to "24" at the same time, IPMMU cannot recognize which TTBR is used for page table walk.
 4. Changing the initial value of these fields has no effect to the system because cache snoop transaction for page table walk request is not supported in RZ/G Series, 2nd Generation. Not configuring to these fields are recommended.

22.2.3 MMU Translation Table Upper Base Register 0/1n (IMTTUBR0/1n)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

Note: n = 0 to 7

This register indicates the base address of the TLB managed by each MMU.

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------------|-----|-----|-----|-----|-----|-----|-----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | TTBR[39:32] | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | TTBR[39:32] | H'00 | R/W | Bits [39:32] of translation table base address This field is used when EAE is 1. |

22.2.4 MMU Translation Table Lower Base Register 0/1n (IMTTLBR0/1n)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

Note: n = 0 to 7

This register indicates the base address of the TLB managed by each MMU.

| | | | | | | | | | | | | | | | | |
|----------------|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TTBR[31:16] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TTBR[15:12] | | | | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 12 | TTBR[31:12] | H'0_0000 | R/W | Bits [31:12] of translation table base address |
| 11 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

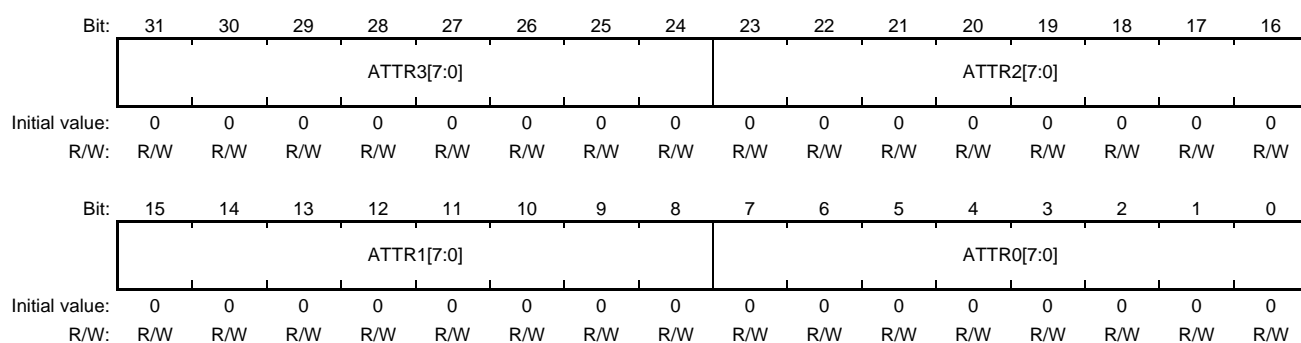
22.2.5 MMU Memory Attribute Indirection Register 0n (IMMAIR0n)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

Note: n = 0 to 7

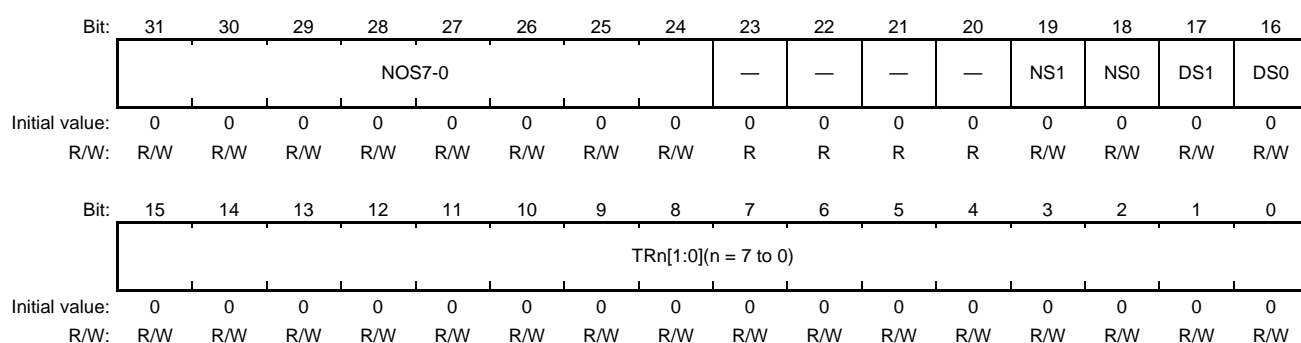
When using the Long-descriptor translation format, this register works as MAIR0, Memory Attribute Indirection Register in Armv8 VMSAv8-64 and VMSAv8-32. When using the Short-descriptor translation format, this register works as PRRR, Primary Region Remap Register in Armv8 VMSAv8-32.

(Long-descriptor translation format)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 24 | ATTR3[7:0] | H'00 | R/W | The memory attribute encoding for an AttrIdx[2:0] entry. |
| 23 to 16 | ATTR2[7:0] | H'00 | R/W | |
| 15 to 8 | ATTR1[7:0] | H'00 | R/W | |
| 7 to 0 | ATTR0[7:0] | H'00 | R/W | |

(Short-descriptor translation format)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 24 | NOS7-0 | H'00 | R/W | Outer Shareable property mapping for memory attributes n. 0: Memory region is outer shareable. 1: Memory region is inner shareable. |
| 23 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------------|---------------|-----|---|
| 19 | NS1 | B'0 | R/W | Mapping of S = 1 attribute for Normal memory. 0: Region is not shareable. 1: Region is shareable. |
| 18 | NS0 | B'0 | R/W | Mapping of S = 0 attribute for Normal memory. 0: Region is not shareable. 1: Region is shareable. |
| 17 | DS1 | B'0 | R/W | Mapping of S = 1 attribute for Device memory. 0: Region is not shareable. 1: Region is shareable. |
| 16 | DS0 | B'0 | R/W | Mapping of S = 0 attribute for Device memory. 0: Region is not shareable. 1: Region is shareable. |
| 15 to 0 | TRn[1:0] (n = 7 to 0) | B'00 | R/W | Primary TEX mapping for memory attributes n. n is the value of TEX[0], C and B bits. B'00: Strongly-ordered B'01: Device B'10: Normal memory B'11: Reserved |

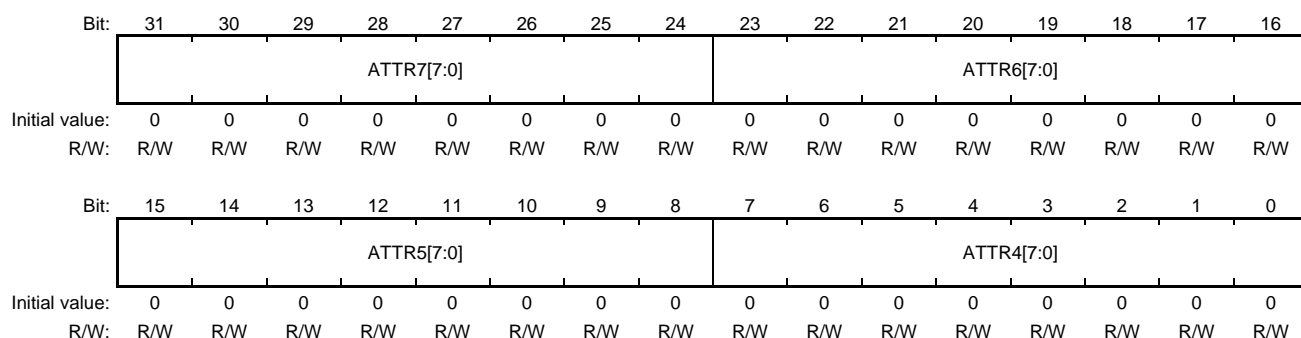
22.2.6 MMU Memory Attribute Indirection Register 1n (IMMAIR1n)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

Note: n = 0 to 7

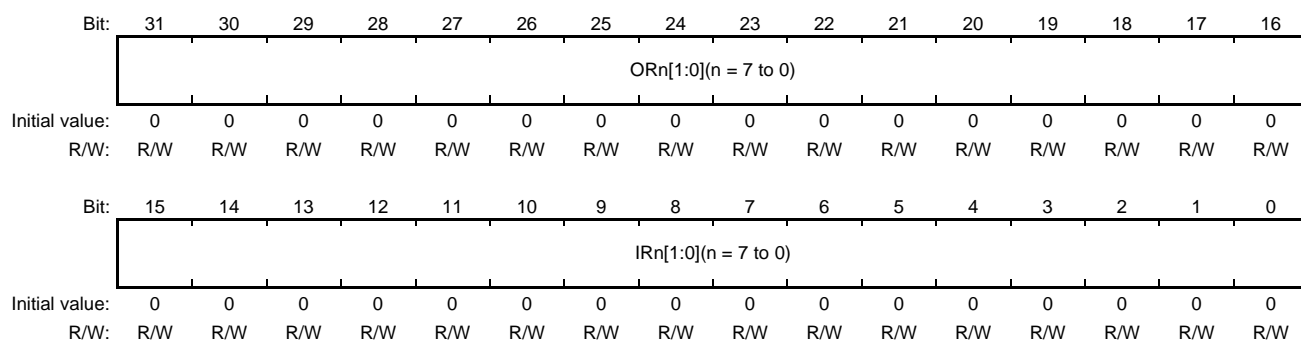
When using the Long-descriptor translation format, this register works as MAIR1, Memory Attribute Indirection Register in Armv8 VMSAv8-64 and VMSAv8-32. When using the Short-descriptor translation format, this register works as NMRR, Normal Memory Remap Register in Armv8 VMSAv8-32.

(Long-descriptor translation format)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 24 | ATTR7[7:0] | H'00 | R/W | The memory attribute encoding for an AttrIdx[2:0] entry. |
| 23 to 16 | ATTR6[7:0] | H'00 | R/W | |
| 15 to 8 | ATTR5[7:0] | H'00 | R/W | |
| 7 to 0 | ATTR4[7:0] | H'00 | R/W | |

(Short-descriptor translation format)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------------|---------------|-----|---|
| 31 to 16 | ORn[1:0] (n = 7 to 0) | B'00 | R/W | Outer Cacheable property mapping for memory attributes n. B'00: Region is non-cacheable. B'01: Region is write-back, write-allocate. B'10: Region is write-through, no write-allocate. B'11: Region is write-back, no write-allocate. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------------|---------------|-----|---|
| 15 to 0 | IRn[1:0] (n = 7 to 0) | B'00 | R/W | Inner Cacheable property mapping for memory attributes n. B'00: Region is non-cacheable. B'01: Region is write-back, write-allocate. B'10: Region is write-through, no write-allocate. B'11: Region is write-back, no write-allocate. |

22.2.7 MMU Error Status Register n (IMSTRn)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

Note: n = 0 to 7

This register indicates the error status of during address translation. About more detail information, refer to section 22.3.8.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|-------------|----|----|----|----|----|----|----|------|-----|-------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | ERRLVL[1:0] | — | — | — | — | — | — | — | MHIT | — | ABORT | PF | TF | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13, 12 | ERRLVL [1:0] | B'00 | R | indicate which level of page table walk caused the error. B'00: Level1 page table walk B'01: Level2 page table walk B'10: Level3 page table walk |
| 11 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 10 to 8 | ERRCODE [2:0] | B'000 | R | Indicate error type. ERRCODE is set only during page table walk. B'001: Translation Fault B'100: Access Flag Fault / Permission Fault B'101: Secure Access Fault Others: reserved |
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | MHIT | B'0 | R/W | TLB Conflict Fault Indicate that multiple TLB hits occurred. |
| 3 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 2 | ABORT | B'0 | R/W | This bit is set to 1 when the IPMMU received an error response during a page table walk. |
| 1 | PF | B'0 | R/W | Permission Fault This bit is set to 1 when an access right violation occurred. |
| 0 | TF | B'0 | R/W | Translation Fault This bit is set to 1 when a translation fault occurred during a page table walk. |

22.2.8 MMU Error Lower Address Register n (IMELARn)

| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|-------------|-------------|--------|--------|
| √ | √ | √ | √ | √ |

Note: n = 0 to 7

This register indicates the address which an address translation error occurred.

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | EAR[31:16] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | EAR[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | EAR[31:0] | All 0 | R | The faulting virtual address is set when an address translation error occurred. |

22.2.9 MMU Error Upper Address Register n (IMEUARn)

| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|-------------|-------------|--------|--------|
| √ | √ | √ | √ | √ |

Note: n = 0 to 7

This register indicates the address which an address translation error occurred.

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|----|----|----|----|----|----|----|----|------------|----|----|----|----|----|----|----|
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | EAR[39:32] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | EAR[39:32] | H'00 | R | The faulting virtual address is set when an address translation error occurred. |

22.2.10 PMB Control Register (IMPCTR)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

This register controls the behavior of the PMB function.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|------------|-----|------|-------|-----|-------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | TTSEL[2:0] | | TTEN | INTEN | — | PMBEN | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 to 4 | TTSEL[2:0] | B'000 | R/W | Translation Table Select Indicates the table number (MMU0 - MMU7). This field is used when IMPCTR.TTEN = 1. |
| 3 | TTEN | B'0 | R/W | TLB Translation Enable 0: Output PPN as a physical address. 1: Output PPN as an intermediate physical address to retranslate through 40-bit TLB. PPN[39:32] is not used in retranslation. |
| 2 | INTEN | B'0 | R/W | Interrupt Enable 0: Don't assert an interrupt when an error occurred. 1: Assert an interrupt when an error occurred. |
| 1 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 0 | PMBEN | B'0 | R/W | PMB Enable 0: PMB disabled 1: PMB enabled |

22.2.11 PMB Address Array n (IMPMBAn)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

Note: n = 0 to 15

This register is used for setting the virtual page number.

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | VPN[31:24] | | | | | | | | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | V | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R |

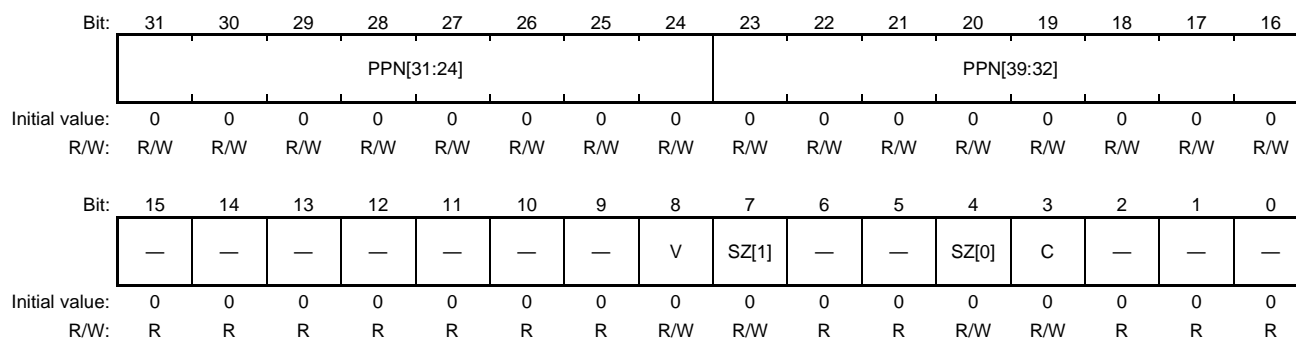
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 24 | VPN[31:24] | H'00 | R/W | Virtual Page Number For 16-Mbyte page, VPN[31:24] is used. For 64-Mbyte page, VPN[31:26] is used. For 128-Mbyte page, VPN[31:27] is used. For 512-Mbyte page, VPN[31:29] is used. |
| 23 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | V | B'0 | R/W | Enable this page translation. |
| 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

22.2.12 PMB Data Array n (IMPMBDn)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

Note: n = 0 to 15

This register is used for setting the physical page number and the memory size which PMB managed.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 24 | PPN[31:24] | H'00 | R/W | Physical Page Number For 16-Mbyte page, PPN[31:24] is used. For 64-Mbyte page, PPN[31:26] is used. For 128-Mbyte page, PPN[31:27] is used. For 512-Mbyte page, PPN[31:29] is used. |
| 23 to 16 | PPN[39:32] | H'00 | R/W | Upper Physical Page Number |
| 15 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | V | B'0 | R/W | Enable this page translation. |
| 7 | SZ[1] | B'0 | R/W | This bit and SZ[0] (bit 4) specify the page size. SZ[1:0] = 00: 16-Mbyte page 01: 64-Mbyte page 10: 128-Mbyte page 11: 512-Mbyte page |
| 6, 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | SZ[0] | B'0 | R/W | Page Size See the description of SZ[1] (bit 7). |
| 3 | C | B'0 | R/W | Cache bit When this bit is set, the request after address translation can be treated as "cacheable" request. |
| 2 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

22.2.13 PMB Status Register (IMPSTR)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

This register indicates the error status of the address translation by PMB.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|------|----|----|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | MHIT | — | — | — | TF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | MHIT | B'0 | R/W | Multiple hit Indicate that multiple PMB hits occurred. |
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | TF | B'0 | R/W | Translation Fault This bit is set to 1 when a translation fault occurred during a PMB translation. |

22.2.14 PMB Error Address Register (IMPEAR)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

This register indicates the address which an address translation error occurred.

| | | | | | | | | | | | | | | | | |
|----------------|------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | EAR[31:16] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | EAR[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | EAR[31:0] | H'0000_0000 | R | The faulting virtual address is set when an address translation error occurred. |

22.2.15 uTLB Control Register n (IMUCTRn)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

Note: n = 0 to 47

This register controls the behavior of each uTLB.

| | | | | | | | | | | | | | | | | | | |
|----------------|--------------|----|----|----|----|----|----|----|---------------|-----|-----|-----|-----|-----|-------|-----------|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | FIXAD DEN | — | — | — | — | — | — | — | FIXADD[39:32] | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R/W | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | TTSEL[3:0] | | | | — | — | FLUSH | MMUE N | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R/W | R/W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 | FIXADDEN | B'0 | R/W | Fix the upper 8 bits of physical address Always output the upper 8 bits of physical address as FIXADD[39:32]. This bit must be used only when IMTTBCRn.EAE is 0. 0: Disable FIXADD[39:32] 1: Enable FIXADD[39:32] IMUCTRn.FIXADD[39:32] cannot be used with stage 2 translation as part of stage 1 translation to form upper 8 bits of IPA. |
| 30 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | FIXADD [39:32] | All 0 | R/W | When FIXADDEN is 1, the upper 8bit of physical address is FIXADD[39:32]. This bit must be used only when IMTTBCRn.EAE is 0. |
| 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 4 | TTSEL[3:0] | B'0000 | R/W | Translation Table B'0000 to B'0111: MMU0 to MMU7 B'1000: PMB B'1001 to B'1111: Reserved |
| 3, 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | FLUSH | B'0 | R/W | micro-TLB Invalidate Invalidate all entries in the micro-TLB. This bit is automatically cleared to 0. So read value is always 0. 1: Invalidate all entries instantaneously. |
| 0 | MMUEN | B'0 | R/W | Address Translation Enable 0: Disable address translation 1: Enable address translation |

22.2.16 uTLB ASID Register n (IMUASIDn)

| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|-------------|-------------|--------|--------|
| √ | √ | √ | √ | √ |

Note: n = 0 to 47

This register is used for setting ASID of each uTLB.

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------------|------------|-----|-----|-----|-----|-----|-----|-----|------------|-----|-----|-----|-----|-----|-----|-----|
| | ASID1[7:0] | | | | | | | | ASID0[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 8 | ASID1[7:0] | H'00 | R/W | ASID1 This field indicates the ASID which the micro-TLB uses in the stage 2 translation. |
| 7 to 0 | ASID0[7:0] | H'00 | R/W | ASID0 This field indicates the ASID which the micro-TLB uses in the stage 1 translation. |

Note: ASID is not supported in RZ/G2. See “22.5.3.6 Treatments of ASID feature”.

22.2.17 MMU System Control Register (IMSCTLR)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

This register controls the behavior of IPMMU function.

[RZ/G2M V1.3]

| | | | | | | | | | | | | | | | | |
|----------------|----------|---------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | diswprot | nsaccen | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 | diswprot | B'0 | R/W | Read and Write protection of MMU System Control Register and MMU auxiliary Control register 1: Enable read and write access permission 0: Disable read and write access permission, read value is all 0 (default) |
| 30 | nsaccen | B'0 | R/W | Non-secure access enable for MMU System Control Register and MMU auxiliary Control register 1: Enable non-secure access 0: Disable non-secure access (default) |
| 29 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0 |

[RZ/G2H, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| | | | | | | | | | | | | | | | | |
|----------------|----------|---------|--------|------------|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | diswprot | nsaccen | dismmu | use_secgrp | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | — | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 | diswprot | B'0 | R/W | Read and Write protection of MMU System Control Register and MMU auxiliary Control register 1: Enable read and write access permission 0: Disable read and write access permission, read value is all 0 (default) |
| 30 | nsaccen | B'0 | R/W | Non-secure access enable for MMU System Control Register and MMU auxiliary Control register 1: Enable non-secure access 0: Disable non-secure access (default) |
| 29 | dismmu | B'0 | R/W | Disable IPMMU cache 1: Disable IPMMU cache 0: Enable IPMMU cache This bit can be set except IPMMU-MM |
| 28 | use_secgrp | — | R/W | Use security group to judge Secure/Non-secure access 1: Group 3 or 2 access is judged as Secure access 0: Only secure access is judged as Secure access |
| 27 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0 |

22.2.18 MMU Auxiliary Control Register (IMSAUXCTLR)

| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|-------------|-------------|--------|--------|
| √ | √ | √ | √ | √ |

This register controls the behavior of IPMMU function. This register is supported main IPMMU (IPMMU-MM) only.

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|----|----|----|----|----|----|---------|----------|----|----|----|----|----|----|----|----------|
| | — | — | — | — | — | — | nhint40 | nmerge40 | — | — | — | — | — | — | — | nmerge32 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R | R | R/W |

| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------------|----|----|----|----|----|----|---|---|---|---|---|-------|-----|---|---|---|
| | — | — | — | — | — | — | — | — | — | — | — | s2pte | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 26 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0 |
| 25 | nhint40 | B'0 | R/W | Disable the "hint" field in long descriptor format 0: Enable the hint field (default) 1: Disable the hint field Hint feature is only applied for 16 adjacent translation table entries. |
| 24 | nmerge40* | B'0 | R/W | Disable merge for the adjacent TLB entry in long descriptor format 0: Enable merge for the adjacent TLB entry (default) 1: Disable merge for the adjacent TLB entry |
| 23 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0 |
| 16 | nmerge32* | B'0 | R/W | Disable merge for the adjacent TLB entry in short descriptor format 0: Enable merge for the adjacent TLB entry (default) 1: Disable merge for the adjacent TLB entry |
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0 |
| 3 | s2pte | B'0 | R/W | Support stage 2 translation table format 0: Use stage 1 translation table format when stage 2 translation enabled (default) 1: Use stage 2 translation table format when stage 2 translation enabled |
| 2 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0 |

Note: * In case that the successive two 4KB page has the same MMU attribute like shareability, nG, IPMMU can be merged these two pages and can be treated them as one 8KB page. This is our special setting for optimizing the page table walk. RZ/G2M V1.3 supports this function but RZ/G2M V3.0 does not.

22.2.19 MMU Interrupt Status Register (IMSSTR)

| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|-------------|-------------|--------|--------|
| √ | √ | √ | √ | √ |

This register indicates the interrupt status from each IPMMU. This register can be read in non-secure mode only. In secure mode, refer to the IMSTRn and IMPSTRn in the IPMMU where secure master is managed.

[RZ/G2H]

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|-----|-----|-----|-----|----|----|-----|-----|-----|-----|----|----|----|----|-----|-----|
| | — | — | — | — | — | — | — | — | — | — | — | — | — | MM | VP1 | VP0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | VI1 | VI0 | VC1 | VC0 | — | — | PV3 | PV2 | PV1 | PV0 | — | MP | IR | HC | DS1 | DS0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 19 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 18 | MM | B'0 | R | Interrupt status of IPMMU-MM |
| 17 | VP1 | B'0 | R | Interrupt status of IPMMU-VP1 |
| 16 | VP0 | B'0 | R | Interrupt status of IPMMU-VP0 |
| 15 | VI1 | B'0 | R | Interrupt status of IPMMU-VI1 |
| 14 | VI0 | B'0 | R | Interrupt status of IPMMU-VI0 |
| 13 | VC1 | B'0 | R | Interrupt status of IPMMU-VC1 |
| 12 | VC0 | B'0 | R | Interrupt status of IPMMU-VC0 |
| 11 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 | — | B'0 | R | Reserved |
| 9 | PV3 | B'0 | R | Interrupt status of IPMMU-PV3 |
| 8 | PV2 | B'0 | R | Interrupt status of IPMMU-PV2 |
| 7 | PV1 | B'0 | R | Interrupt status of IPMMU-PV1 |
| 6 | PV0 | B'0 | R | Interrupt status of IPMMU-PV0 |
| 5 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | MP | B'0 | R | Interrupt status of IPMMU-MP |
| 3 | IR | B'0 | R | Interrupt status of IPMMU-IR |
| 2 | HC | B'0 | R | Interrupt status of IPMMU-HC |
| 1 | DS1 | B'0 | R | Interrupt status of IPMMU-DS1. |
| 0 | DS0 | B'0 | R | Interrupt status of IPMMU-DS0. 0: Interrupt is not accepted 1: Interrupt is accepted. |

[RZ/G2M V1.3, RZ/G2M V3.0]

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-----|-----|----|-----|-----|----|----|----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | MM | VI0 | VC0 | — | PV1 | PV0 | MP | IR | HC | DS1 | DS0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 11 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 | MM | B'0 | R | Interrupt status of IPMMU-MM |
| 9 | VI0 | B'0 | R | Interrupt status of IPMMU-VI0 |
| 8 | VC0 | B'0 | R | Interrupt status of IPMMU-VC0 |
| 7 | — | B'0 | R | Reserved |
| 6 | PV1 | B'0 | R | Interrupt status of IPMMU-PV1 |
| 5 | PV0 | B'0 | R | Interrupt status of IPMMU-PV0 |
| 4 | MP | B'0 | R | Interrupt status of IPMMU-MP |
| 3 | IR | B'0 | R | Interrupt status of IPMMU-IR |
| 2 | HC | B'0 | R | Interrupt status of IPMMU-HC |
| 1 | DS1 | B'0 | R | Interrupt status of IPMMU-DS1. |
| 0 | DS0 | B'0 | R | Interrupt status of IPMMU-DS0 0: Interrupt is not accepted 1: Interrupt is accepted. |

[RZ/G2E]

| | | | | | | | | | | | | | | | | |
|----------------|----|-----|----|-----|----|----|----|----|----|-----|----|----|----|----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | MM | — | VP0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | VI0 | — | VC0 | — | — | — | — | — | PV0 | — | MP | — | HC | DS1 | DS0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 19 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 18 | MM | B'0 | R | Interrupt status of IPMMU-MM |
| 17 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | VP0 | B'0 | R | Interrupt status of IPMMU-VP0 |
| 15 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 14 | VI0 | B'0 | R | Interrupt status of IPMMU-VI0 |
| 13 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | VC0 | B'0 | R | Interrupt status of IPMMU-VC0 |
| 11 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 | — | B'0 | R | Reserved |
| 9 to 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 | PV0 | B'0 | R | Interrupt status of IPMMU-PV0 |
| 5 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | MP | B'0 | R | Interrupt status of IPMMU-MP |
| 3 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | HC | B'0 | R | Interrupt status of IPMMU-HC |
| 1 | DS1 | B'0 | R | Interrupt status of IPMMU-DS1. |
| 0 | DS0 | B'0 | R | Interrupt status of IPMMU-DS0 0: Interrupt is not accepted 1: Interrupt is accepted. |

[RZ/G2N]

| | | | | | | | | | | | | | | | | |
|----------------|----|-----|----|-----|----|----|----|----|----|-----|----|----|----|----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | MM | — | VP0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | VI0 | — | VC0 | — | — | — | — | — | PV0 | — | MP | — | HC | DS1 | DS0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 19 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 18 | MM | B'0 | R | Interrupt status of IPMMU-MM |
| 17 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | VP0 | B'0 | R | Interrupt status of IPMMU-VP0 |
| 15 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 14 | VI0 | B'0 | R | Interrupt status of IPMMU-VI0 |
| 13 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | VC0 | B'0 | R | Interrupt status of IPMMU-VC0 |
| 11 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 | — | B'0 | R | Reserved |
| 9 to 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 | PV0 | B'0 | R | Interrupt status of IPMMU-PV0 |
| 5 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | MP | B'0 | R | Interrupt status of IPMMU-MP |
| 3 | — | B'0 | R | Reserved |
| 2 | HC | B'0 | R | Interrupt status of IPMMU-HC |
| 1 | DS1 | B'0 | R | Interrupt status of IPMMU-DS1. |
| 0 | DS0 | B'0 | R | Interrupt status of IPMMU-DS0 0: Interrupt is not accepted 1: Interrupt is accepted. |

22.2.20 MMU Performance Monitor Control Register (IMPFMCTR)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

[RZ/G2M V1.3, RZ/G2M V3.0]

| | | | | | | | | | | | | | | | | |
|----------------|----|----|---------|-----|----|----|----|----|----|----|----|----|----|----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | MD[1:0] | — | — | — | — | — | — | — | — | — | — | — | RST | EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13, 12 | MD[1:0] | B'00 | R/W | Monitor Mode B'00: Monitor all 40-bit MMUs B'01: Monitor all 32-bit MMUs B'10: Setting prohibited B'11: Setting prohibited |
| 11 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | RST | B'0 | R/W | Reset all status and counter values. This bit is valid when EN = 1. |
| 0 | EN | B'0 | R/W | Performance Monitor Enable 0: Stop to count 1: Start to count All counters stop when TOTAL bits get full. |

Note: This register is not supported in IPMMU-cache of RZ/G2M V1.3, RZ/G2M V3.0.

[RZ/G2H, RZ/G2N, RZ/G2E]

| | | | | | | | | | | | | | | | | |
|----------------|----|----|---------|-----|----------|-----|-----|-----|----|----|----|----|----|----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | MD[1:0] | — | SEL[2:0] | | | — | — | — | — | — | — | — | RST | EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R | R/W | R/W | R/W | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13, 12 | MD[1:0] | B'00 | R/W | Monitor Mode B'00: Monitor all 40-bit MMUs B'01: Monitor all 32-bit MMUs B'10: Setting prohibited B'11: Setting prohibited |
| 11 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 10 to 8 | SEL[2:0] | B'000 | R/W | When MD is B'11, SEL indicates the MMU table number to be monitored. |
| 7 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | RST | B'0 | R/W | Reset all status and counter values. This bit is valid when EN = 1. |
| 0 | EN | B'0 | R/W | Performance Monitor Enable 0: Stop to count 1: Start to count All counters stop when TOTAL bits get full. |

22.2.21 MMU Performance Monitor Total Translation Counter (IMPFMTOTAL)

| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|-------------|-------------|--------|--------|
| √ | √ | √ | √ | √ |

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|-------------|----|----|----|----|----|----|----|--------------|----|----|----|----|----|----|----|
| | — | — | — | — | — | — | — | — | TOTAL[23:16] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TOTAL[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | TOTAL[23:0] | H'00_0000 | R | The total number of translation requests |

Note: This register is not supported in IPMMU-cache of RZ/G2M V1.3, RZ/G2M V3.0.

22.2.22 MMU Performance Monitor Hit Counter (IMPFMHIT)

| | | | | |
|--------|-------------|-------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----|----|----|----|----|------------|----|----|----|----|----|----|----|--|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | | |
| | — | — | — | — | — | — | — | — | HIT[23:16] | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | |
| | HIT[15:0] | | | | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | HIT[23:0] | H'00_0000 | R | The total number of TLB hit requests = L3 TLB hit |

Note: This register is not supported in IPMMU-cache of RZ/G2M V1.3, RZ/G2M V3.0.

22.2.23 MMU Performance Monitor L3 Miss Counter (IMPFML3MISS)

| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|-------------|-------------|--------|--------|
| √ | √ | √ | √ | √ |

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|--------------|----|----|----|----|----|----|----|---------------|----|----|----|----|----|----|----|
| | — | — | — | — | — | — | — | — | L3MISS[23:16] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | L3MISS[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | L3MISS [23:0] | H'00_0000 | R | The total number of L3 miss requests (not including L2 miss and L1 miss) = L2 TLB hit |

22.2.24 MMU Performance Monitor L2 Miss Counter (IMPFML2MISS)

| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|-------------|-------------|--------|--------|
| √ | √ | √ | √ | √ |

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|--------------|----|----|----|----|----|----|----|---------------|----|----|----|----|----|----|----|
| | — | — | — | — | — | — | — | — | L2MISS[23:16] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | L2MISS[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | L2MISS [23:0] | H'00_0000 | R | The total number of L2 miss requests (not including L1 miss) = Page Table Walk |

22.2.25 MMU Performance Monitor Miss Counter (IMPFMMISS)

| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|-------------|-------------|--------|--------|
| √ | — | √ | √ | √ |

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|------------|----|----|----|----|----|----|----|-------------|----|----|----|----|----|----|----|
| | — | — | — | — | — | — | — | — | MISS[23:16] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MISS[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | MISS [23:0] | H'00_0000 | R | The total number of miss requests |

Note: This register is not supported in IPMMU-cache of RZ/G2M V1.3, RZ/G2M V3.0.

22.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

22.3.1 Address Translation Sequence in micro-TLBs

Each micro-TLB has independent entries and can translate from a virtual address to a physical address without sending translation requests to the IPMMU if the virtual address is cached in the micro-TLB.

When a translation error occurred in address translation, the micro-TLB blocks subsequent transactions until a valid page entry is registered. Set IMUCTRn.FLUSH = 1 after the entry registration.

Figure 22.2 shows the micro-TLB address translation sequence.

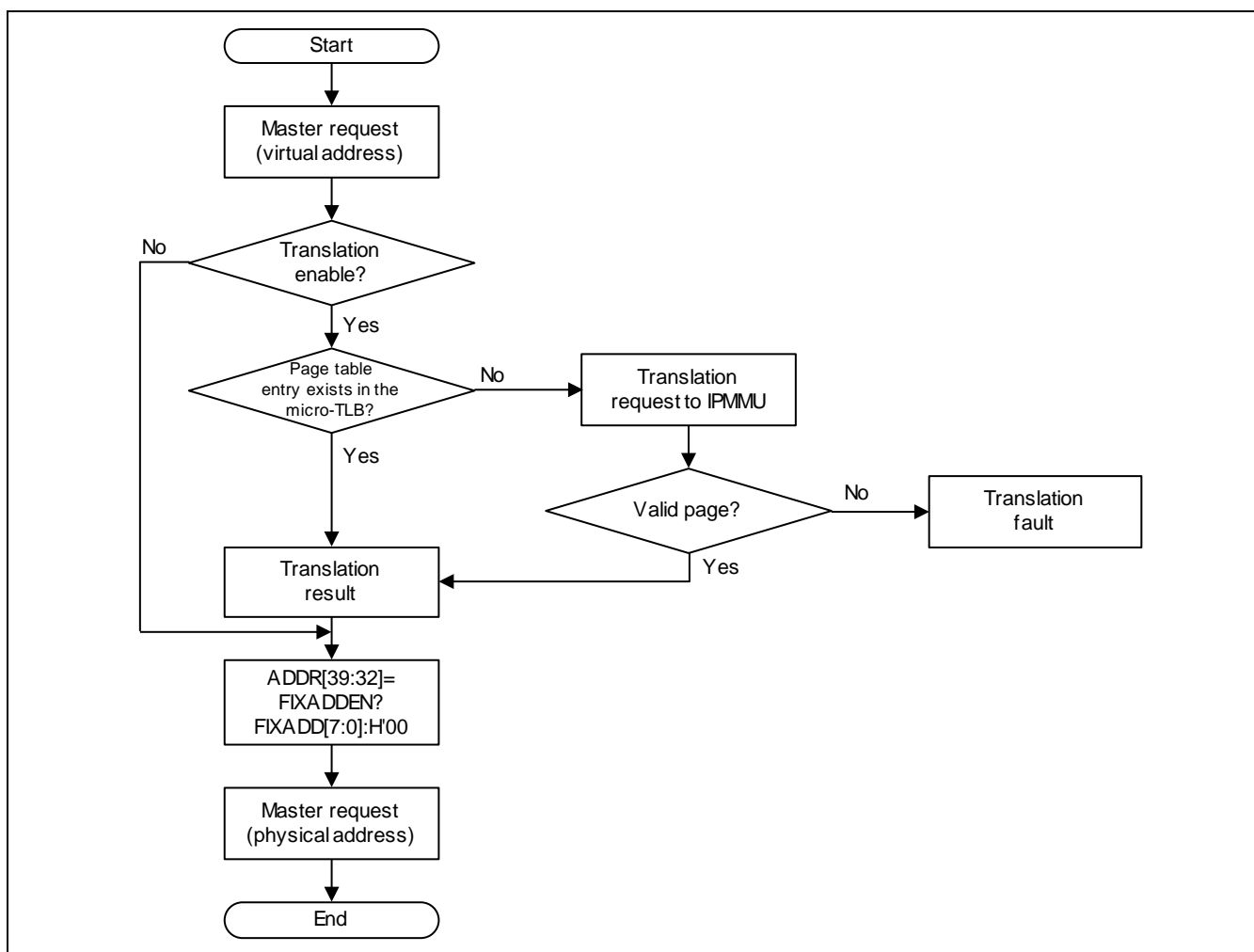


Figure 22.2 micro-TLB Address Translation Sequence

22.3.2 Address Translation Sequence in the IPMMU

(1) MMU Address Translation

The IPMMU has independent eight non-secure page tables and eight secure page tables which support two page table formats, the Short-descriptor format and the Long-descriptor format. When the IPMMU receives an address translation request from a micro-TLB, the IPMMU starts address translation sequence based on the page table specified by IMUCTRn.TTSEL. In case that the micro-TLB is enabled and MMU is selected in IMUCTRn.TTSEL, then the corresponding IMCTRn.MMUEN also need to be set. In case that IMCTRn.MMUEN is disabled, the corresponding micro-TLB must not be enabled.

Figure 22.3 shows the MMU address translation sequence.

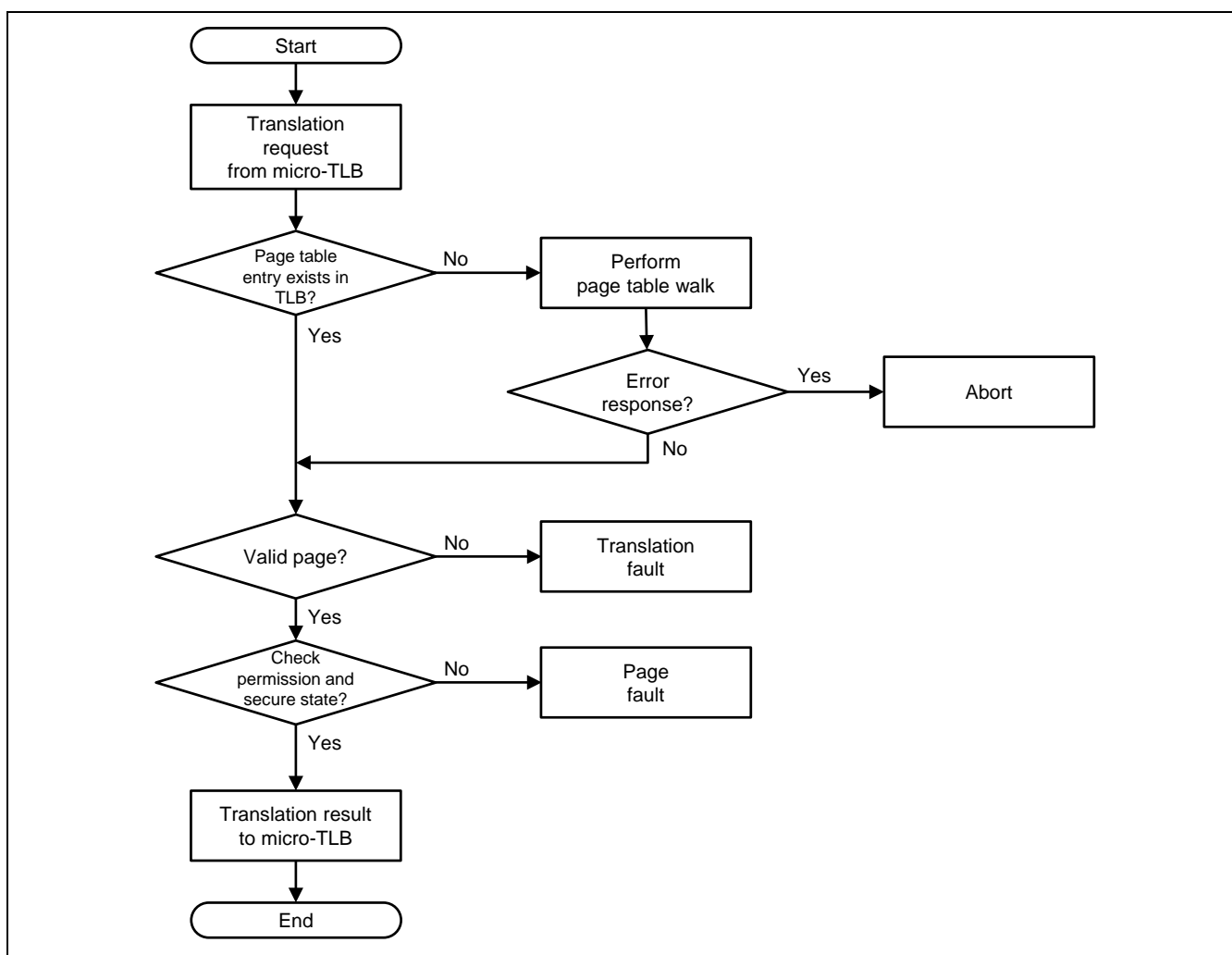


Figure 22.3 MMU Address Translation Sequence

(2) PMB Address Translation

PMB address translation is performed in two ways. When PMB address translation is chosen in IMUCTRn.TTSEL, all virtual addresses are translated by PMB. When PMB address translation is performed in a part of MMU translation (IMTTBCRn.PMB = 1), virtual addresses within H'8000_0000 to H'BFFF_FFFF are translated to physical addresses by PMB. In case that the micro-TLB is enabled and PMB is selected in IMUCTRn.TTSEL, then the corresponding IMPCTR.PMBEN also need to be set. In case that IMPCTR.PMBEN is disabled, the corresponding micro-TLB must not be enabled.

Figure 22.4 shows the PMB translation flow.

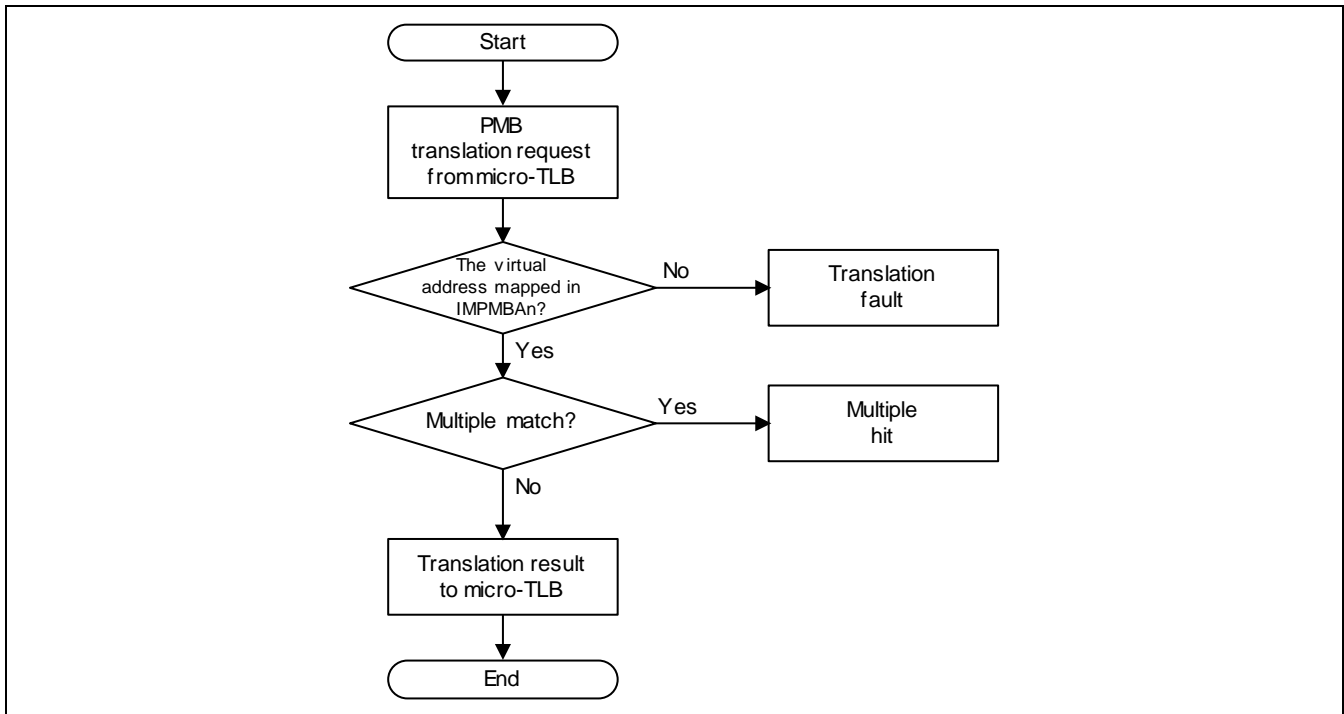


Figure 22.4 PMB Translation Flow

22.3.3 Address Space

(1) Short-descriptor

In the short-descriptor format, 32-bit virtual address space is translated to 32-bit physical address space managed by TTBR0/1. When IMTTBCRn.PMB = 1, translation address space is compatible with our previous products. In this mode, a virtual address within H'8000_0000 to H'BFFF_FFFF is translated by PMB and a virtual address within H'C000_0000 to H'FFFF_FFFF is mapped to a physical address of the same value.

Figure 22.5 shows the Short-descriptor address space.

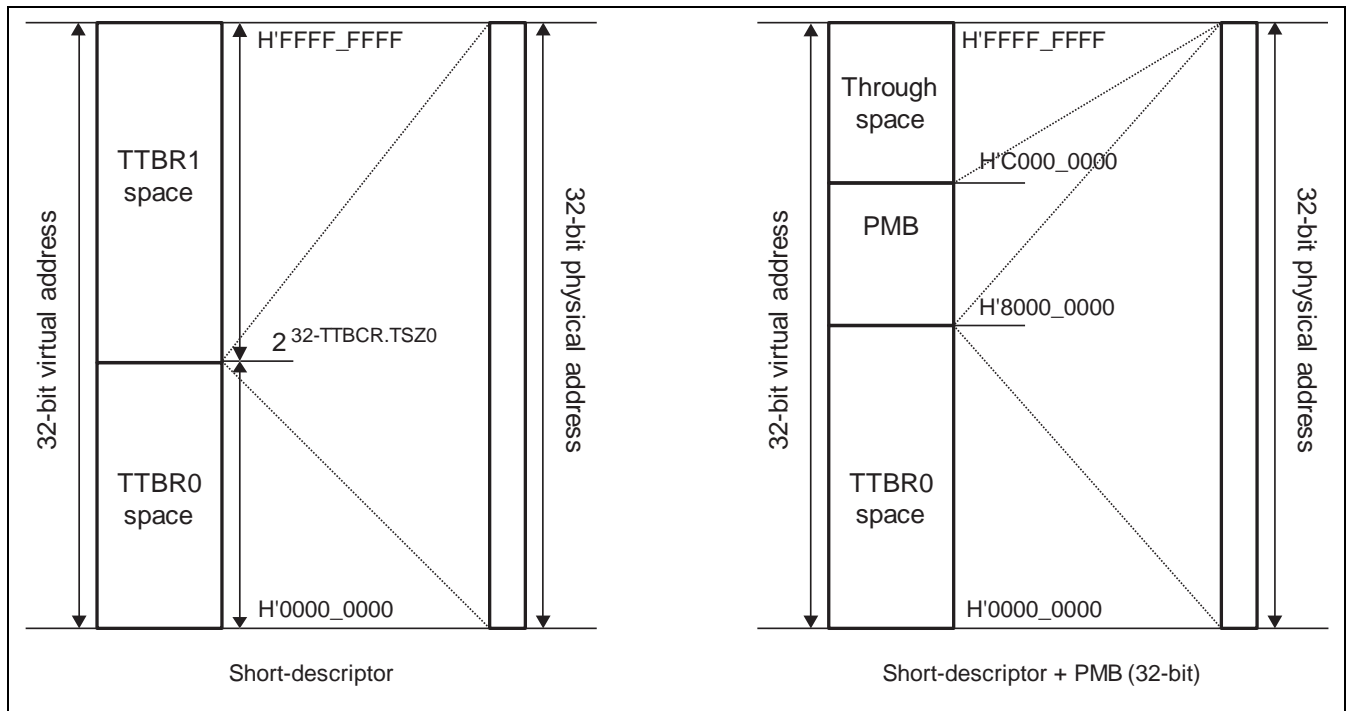


Figure 22.5 Short-descriptor Address Space

(2) Long-descriptor

In the long-descriptor format, 32-bit and 40-bit virtual address space is translated to 40-bit physical address space managed by TTBR0/1. The IPMMU supports the Secure PL1 and PL0 stage 1 translation and the Non-secure PL1 and PL0 stage 1 translation.

Figure 22.6 and Figure 22.7 shows the Long-descriptor address space for VMSAv8-32 and VMSAv8-64.

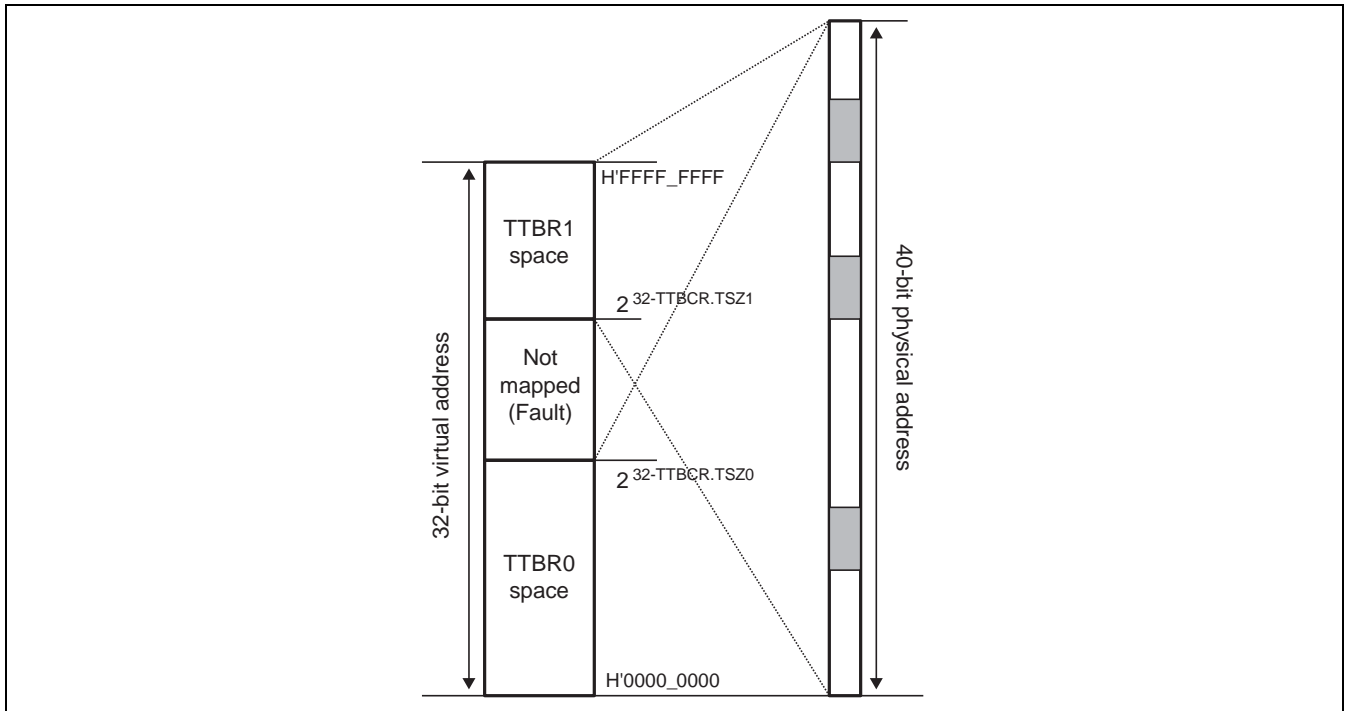


Figure 22.6 Long-descriptor Address Space (VMSAv8-32)

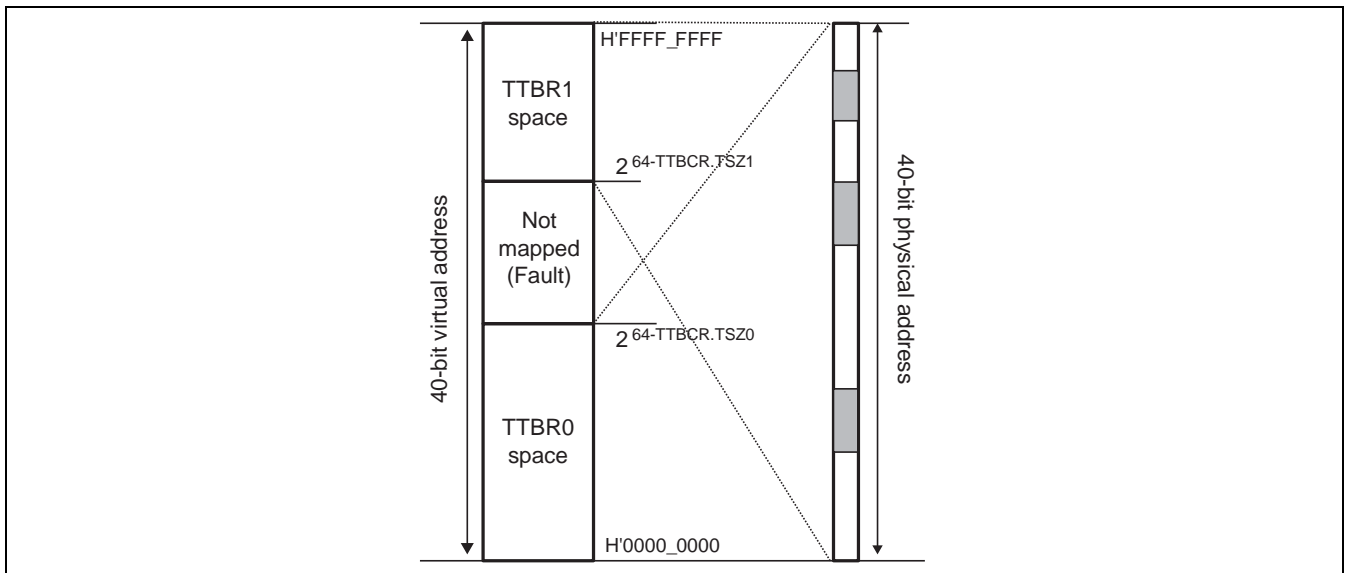


Figure 22.7 Long-descriptor Address Space (VMSAv8-64)

(3) PMB

The IPMMU supports the PMB address translation system. Figure 22.8 shows the PMB address space.

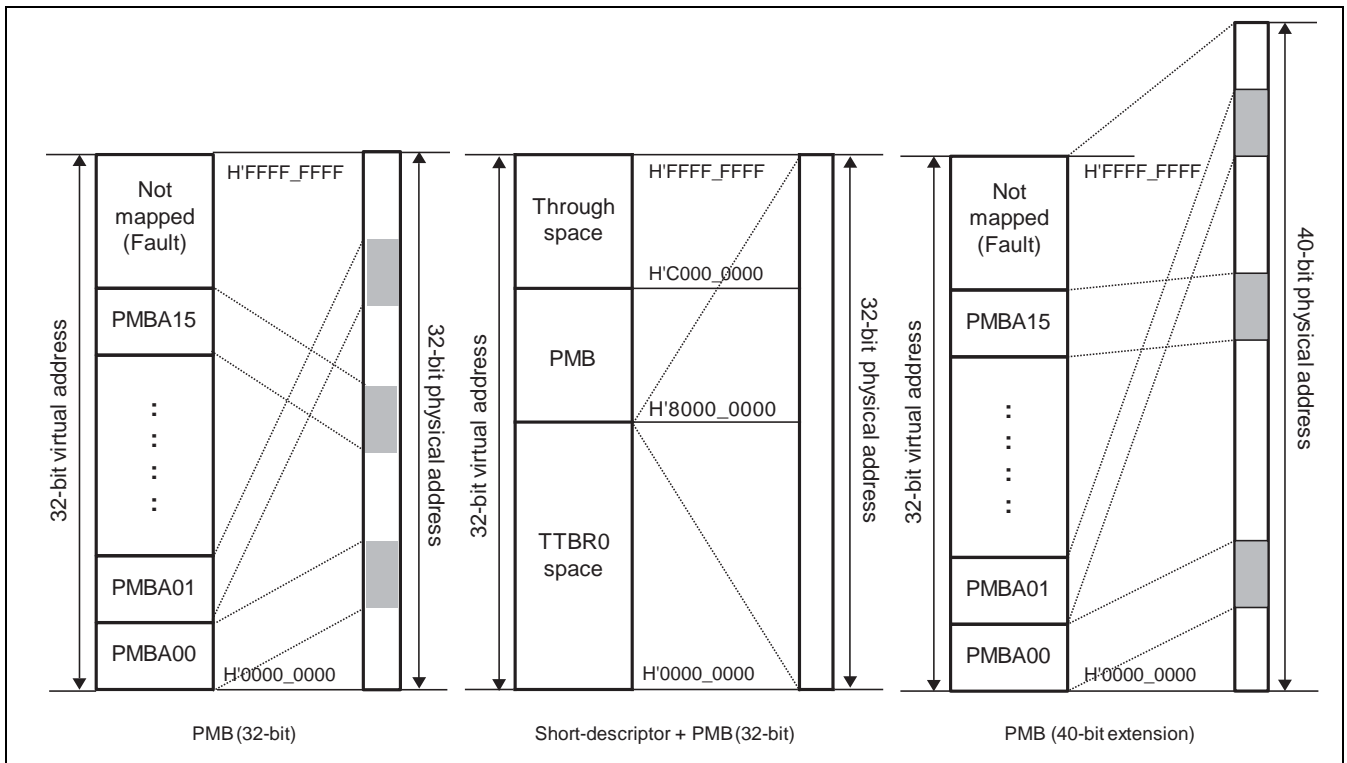


Figure 22.8 PMB Address Space

(4) Stage 2 Translation [Except RZ/G2M V1.3, RZ/G2M V3.0]

The IPMMU supports stage 2 translation. At the stage 1, 32-bit address space is translated to 32-bit or 40-bit intermediate physical address (IPA) space. At the stage 2, 32-bit or 40-bit IPA space is translated to 40-bit physical address space. In order to enable the stage 2 translation, set 1 to IMCTRn.TREN and set its stage 2 page table to IMCTRn.RTSEL. In default, IPMMU use stage 1 translation table format even if performing stage 2 translation. In default, IPMMU use stage 1 translation table format even if performing stage 2 translation. In case of using stage 2 translation table format in stage 2 translation, set 1 to IMSAUXCTLR.S2PTE. In some hypervisor system, only address translation from IPA to PA is supported by stage 2 translation. By setting 1 to IMTTBCRn.BYPEN, IPMMU doesn't perform stage 1 address translation and start stage 2 address translation.*

Note: * IMSAUXCTLR.S2PTE and IMTTBCRn.BYPEN are supported by RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E.

Figure 22.9 shows the stage 2 translation address space.

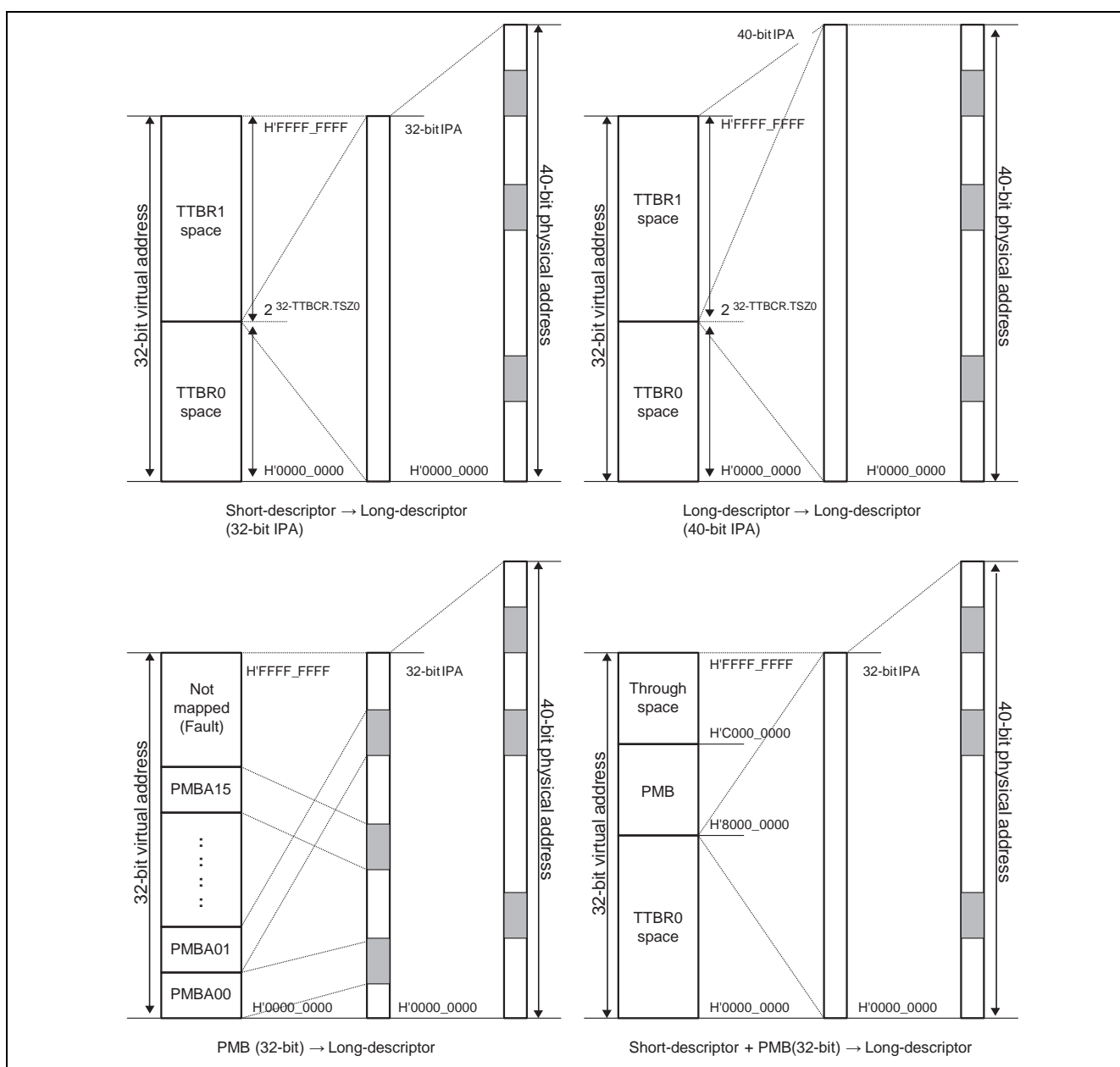


Figure 22.9 Stage 2 Translation Address Space

22.3.4 Initialization Sequence

This section describes the initialization sequence of the IPMMU.

(1) Module Stop

The IPMMU doesn't support module stop control. The IPMMU can be used whenever a master requires address translation through the IPMMU. When a master doesn't require address translation, the corresponding IMUCTRn MMUEN must be disabled.

(2) micro-TLB

Before a master access, the corresponding micro-TLB registers which includes the translation mode and ASIDs must be configured.

Figure 22.10 and Figure 22.11 shows the example of the procedure for using MMU and PMB function.

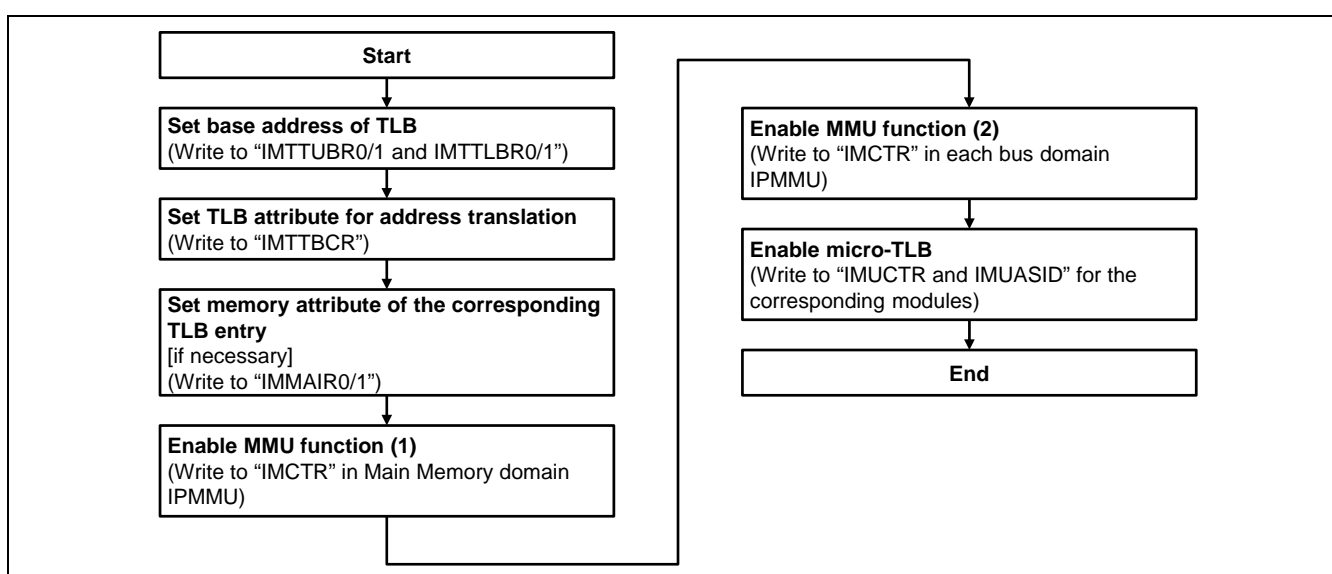


Figure 22.10 initialization sequence for using MMU function

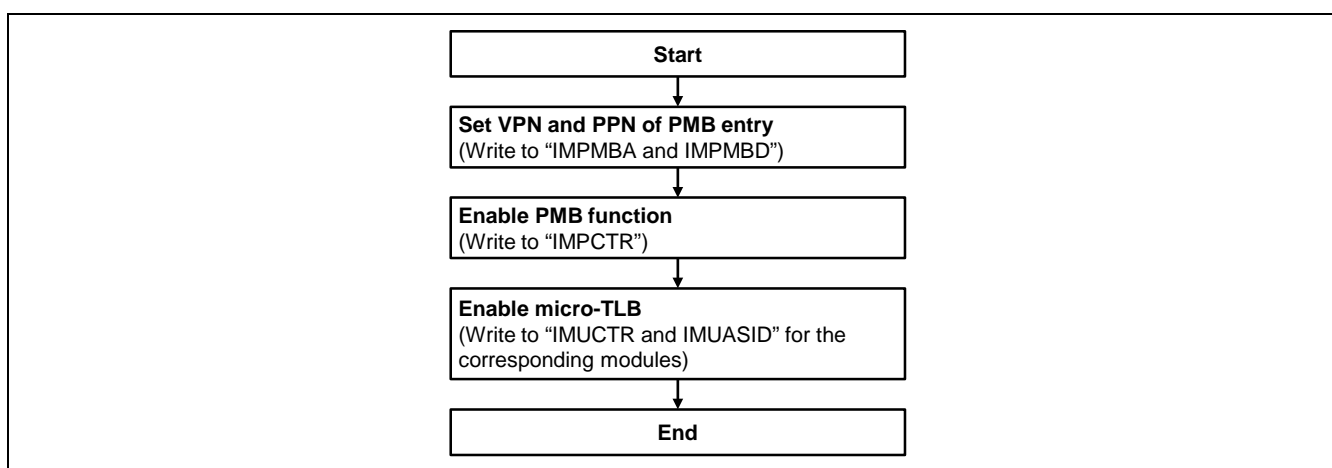


Figure 22.11 initialization sequence for using PMB function

22.3.5 TLB Maintenance

Some TLB maintenances between an IPMMU and the connected micro-TLBs are performed automatically by hardware. In case that the flush operation is performed via IMCTRn (IMCTRn FLUSH = 1), all the microTLB which is related to the same MMU tables are applied the TLB flush and clear the all TLB entry in microTLB.

Table 22.6 TLB Maintenance

| Category | Software operation | TLB maintenance in IPMMU | TLB maintenance in micro-TLB |
|----------------|----------------------------|--|---------------------------------|
| MMU Page Table | First entry registration | Not required | Not required |
| | Entry update | Invalidate the target TLB (IMCTRn.FLUSH = 1) | Automatically invalidated |
| | Entry release | Invalidate the target TLB (IMCTRn.FLUSH = 1) | Automatically invalidated |
| ASID | ASID update | Automatically invalidated | Automatically invalidated |
| MMU registers | MMU register values update | Invalidate the target TLB (IMCTRn.FLUSH = 1) | Automatically invalidated |
| PMB | PMB entry registration | Not required | Not required |
| | PMB entry update | Not required | Invalidate the target micro-TLB |
| | PMB entry release | Not required | Invalidate the target micro-TLB |

IPMMU-main cannot issue the flush operation to each IPMMU-cache and microTLB. In case of replace the MMU table, the flush operation of both IPMMU-main and IPMMU-cache via IMCTRn (IMCTRn.FLUSH = 1) is required. TLB invalidation is instantaneous so software doesn't need to check IMUCTRn.FLUSH or IMCTRn.FLUSH whether TLB invalidation is completed. Figure 22.12 shows the example of the procedure for updating TLB entry

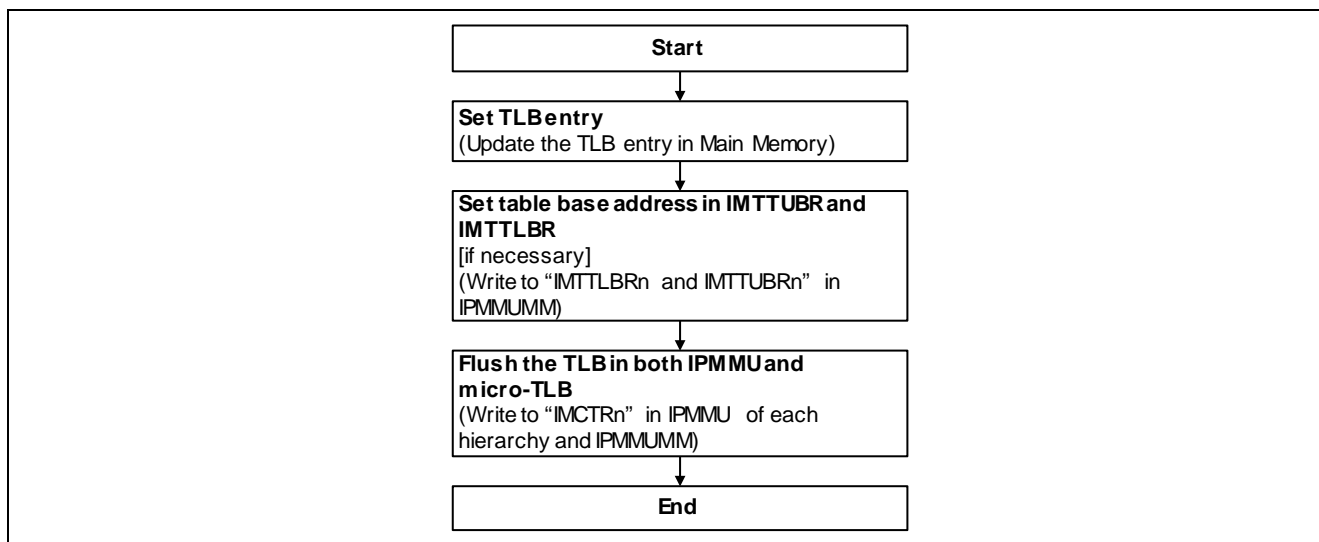


Figure 22.12 update sequence of TLB entry

22.3.6 Security Support

The IPMMU supports the security function based on Armv8 and additional extensions.

(1) Banked Registers

The IPMMU has two copies of a register at the same address. They are banked with secure and non-secure. A Non-secure access can read and write only non-secure registers and a secure access can read and write secure registers. A secure access can also read and write non-secure registers through the alias address space to non-secure registers (from H'0800 to H'0FFC).

(2) Secure-Group

Besides Secure access, IPMMUs support another secure access. The IPMMU which belongs to either secure group2 or secure group3 is treated as a secure, and performs its page table walk in secure mode. Table 22.7 shows the supported secure mode of IPMMU.

Table 22.7 Supported Secure Mode

| IPMMU | Secure | Secure group control | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----------|--------|----------------------|--------|-----------------------------|--------|--------|
| IPMMU-VI0 | √ | √ | √ | √ | √ | √ |
| IPMMU-VI1 | √ | √ | √ | — | — | — |
| IPMMU-VP0 | √ | √ | √ | — | √ | √ |
| IPMMU-VP1 | √ | √ | √ | — | — | — |
| IPMMU-VC0 | √ | √ | √ | √ | √ | √ |
| IPMMU-VC1 | √ | √ | √ | — | — | — |
| IPMMU-PV0 | √ | √ | √ | √ | √ | √ |
| IPMMU-PV1 | √ | √ | √ | √ | — | — |
| IPMMU-PV2 | √ | √ | √ | — | — | — |
| IPMMU-PV3 | √ | √ | √ | — | — | — |
| IPMMU-IR | √ | √ | √ | √ | √ | — |
| IPMMU-HC | √ | √ | √ | √ | √ | √ |
| IPMMU-MP | √ | √ | √ | √ | √ | √ |
| IPMMU-DS0 | √ | √ | √ | √ | √ | √ |
| IPMMU-DS1 | √ | √ | √ | √ | √ | √ |
| IPMMU-MM | √ | √ | √ | √ | √ | √ |

Note: √: Supported, —: Not Supported

(3) Address Translation

When a Secure/Accelerator master sends a translation request to an IPMMU through a micro-TLB, Secure MMU registers are referred to perform its page table walk. The Secure page table walk can read secure page table entries (NS = 0). In the case of a Non-secure translation request, Non-secure MMU registers are referred and the access to secure page table entries is ignored.

22.3.7 OS-ID support

IPMMU-PV0/1/2/3 and IPMMU-VC0/1 [RZ/G2H, RZ/G2M V3.0] support the OS-ID which is used for specifying the operating system. In order to use this feature, the microTLB which corresponds to the OS-ID need to be used. Table 22.8 shows the micro-TLB assignment for each OS-ID.

Table 22.8 micro-TLB assignment for OS-ID support.

| OS-ID | Micro-TLB assignment |
|-------|----------------------------------|
| ID 0 | uTLB0, uTLB8, uTLB16 and uTLB24 |
| ID 1 | uTLB1, uTLB9, uTLB17 and uTLB25 |
| ID 2 | uTLB2, uTLB10, uTLB18 and uTLB26 |
| ID 3 | uTLB3, uTLB11, uTLB19 and uTLB27 |
| ID 4 | uTLB4, uTLB12, uTLB20 and uTLB28 |
| ID 5 | uTLB5, uTLB13, uTLB21 and uTLB29 |
| ID 6 | uTLB6, uTLB14, uTLB22 and uTLB30 |
| ID 7 | uTLB7, uTLB15, uTLB23 and uTLB31 |

22.3.8 Error Handling

(1) IPMMU Interrupt Requests

Table 22.9 shows the error events which can cause interrupts. They are all errors which IPMMU supports. The errors are set in IMSTRn registers.

Table 22.9 IPMMU Error Events

| Category | Error | Description |
|-------------------------------------|---------------------|--|
| Long-descriptor Short-descriptor | Translation Fault | A virtual address was not mapped in the region defined by TTBR0 and TTBR1.* |
| | | A descriptor was neither Table nor Block format |
| | Page Fault | AP (access permissions) / AF (access flag) bit mismatch occurred. NS bit mismatch occurred. Non-secure master read a Secure page table entry. |
| | Abort | The IPMMU got an error response from an Interconnect during a page table walk. |
| PMB | Multiple Hit (MHIT) | Multiple TLB hits occurred. |
| | Translation Fault | A virtual address was not mapped in IMPMBAn. |
| | Multiple Hit (MHIT) | Multiple PMB hits occurred. |

Note: * This factor is only for long-descriptor.

(2) Error Handling Flow

When the micro-TLB cannot translate its virtual address because of some IPMMU errors, the micro-TLB stops to issue the following transactions. When the IPMMU detects an error, IMCTRn.FLUSH must be set after the IMSTRn register is cleared and page table entries are updated. After the TLB invalidation, the micro-TLB resumes to issue transactions.

Figure 22.13 shows the error handling flow.

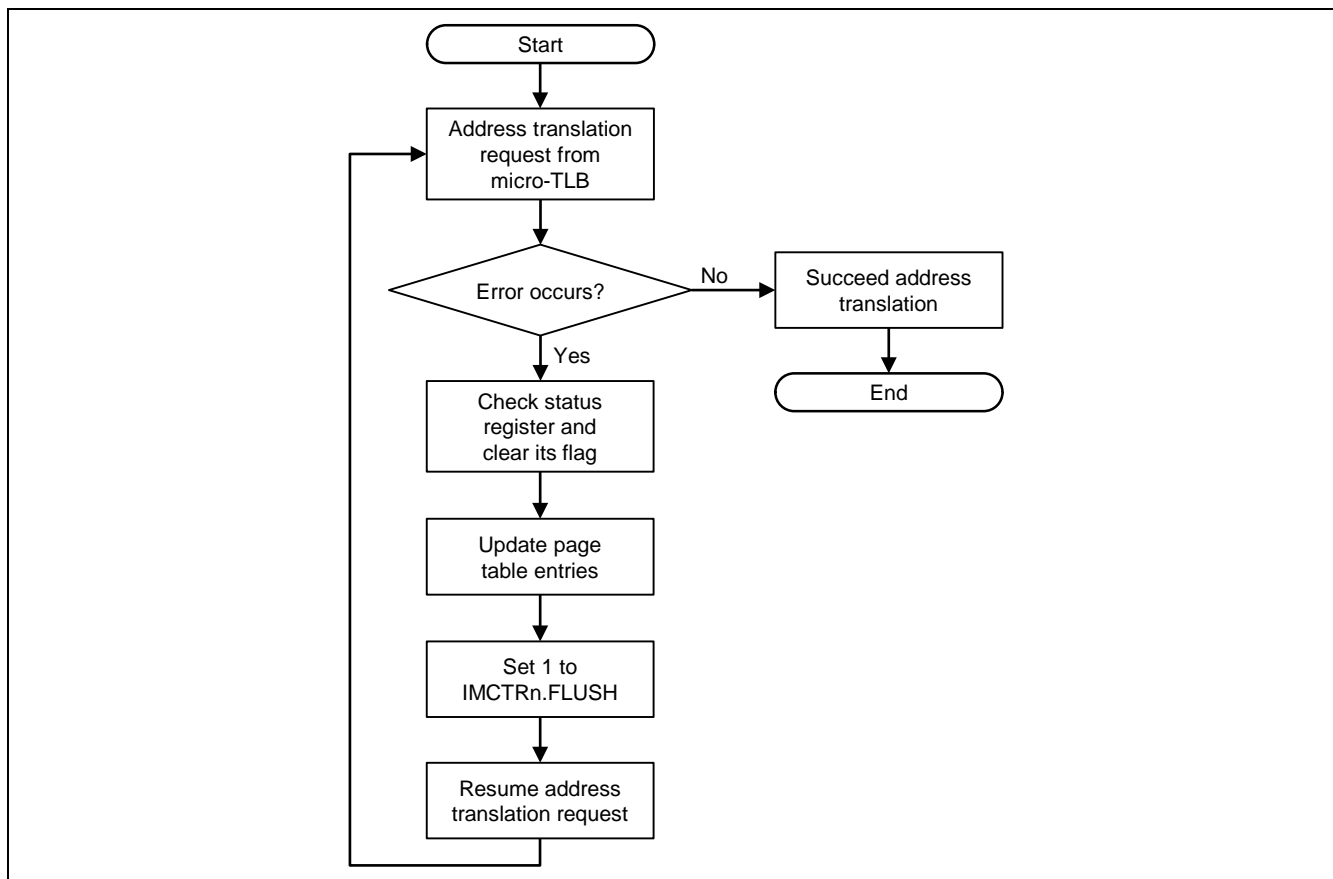


Figure 22.13 Error Handling Flow

22.3.9 Performance monitor

In order to check the number of the page hit or miss, performance monitor is helpful for this purpose. When you use performance monitor, set the IMPFMCTR.MD[1:0] bits. After that set 1 to IMPFMCTR.EN bit. After you set 1 to IMPFMCTR.EN, IMPFMTOTAL, IMPFMHIT, IMPFML3MISS and IMPFML2MISS registers start counting.

22.4 micro-TLB Assignment

The assignment of micro-TLB varies with the RZ/G series products. Refer to the following sections.

22.4.1 micro-TLB Assignment

Table 22.10 micro-TLB Assignment in IPMMUVI0

| micro-TLB | Master Module | | | | |
|-----------|---------------|---------------|---------------|-----------|-----------|
| | [RZ/G2H] | [RZ/G2M V1.3] | [RZ/G2M V3.0] | [RZ/G2N] | [RZ/G2E] |
| 00 | VIN ch0-3 | VIN ch0-3 | VIN ch0-3 | VIN ch0-3 | — |
| 01 | — | VIN ch4-7 | VIN ch4-7 | VIN ch4-7 | VIN ch4-5 |
| 02 | — | — | — | — | — |
| 03 | — | — | — | — | — |
| 04 | — | — | — | — | — |
| 05 | — | FCPVB ch0 | FCPVB ch0 | — | — |
| 06 | — | — | — | — | — |
| 07 | — | — | — | — | — |
| 08 | FCPVD ch0 | FCPVD ch0 | FCPVD ch0 | FCPVD ch0 | FCPVD ch0 |
| 09 | FCPVD ch1 | FCPVD ch1 | FCPVD ch1 | FCPVD ch1 | FCPVD ch1 |
| 10 | — | FCPVD ch2 | FCPVD ch2 | — | — |
| 11 | — | FCPVD ch3 | FCPVD ch3 | — | — |
| 12 | — | HDMI ch0 | HDMI ch0 | HDMI ch0 | — |
| 13 | — | — | — | — | — |
| 14 | — | — | — | — | — |
| 16 | — | — | — | — | — |
| 33 | — | — | — | — | — |

Table 22.11 micro-TLB Assignment in IPMMUVI1

| micro-TLB | Master Module | | | | |
|-----------|---------------|---------------|---------------|----------|----------|
| | [RZ/G2H] | [RZ/G2M V1.3] | [RZ/G2M V3.0] | [RZ/G2N] | [RZ/G2E] |
| 00 | — | — | — | — | — |
| 01 | VIN ch4-7 | — | — | — | — |
| 08 | — | — | — | — | — |
| 09 | — | — | — | — | — |
| 10 | — | FCPVD ch2 | — | — | — |
| 12 | HDMI ch0 | — | — | — | — |
| 13 | — | — | — | — | — |

Table 22.12 micro-TLB Assignment in IPMMUVP0

| micro-TLB | Master Module | | | | |
|-----------|---------------|---------------|---------------|-----------|-----------|
| | [RZ/G2H] | [RZ/G2M V1.3] | [RZ/G2M V3.0] | [RZ/G2N] | [RZ/G2E] |
| 00 | FCPF ch0 | — | — | FCPF ch0 | FCPF ch0 |
| 01 | — | — | — | — | — |
| 05 | FCPVB ch0 | — | — | FCPVB ch0 | FCPVB ch0 |
| 07 | — | — | — | — | — |
| 08 | FCPVI ch0 | — | — | FCPVI ch0 | FCPVI ch0 |
| 09 | — | — | — | — | — |

Table 22.13 micro-TLB Assignment in IPMMUVP1

| micro-TLB | Master Module | | | | |
|-----------|---------------|---------------|---------------|----------|----------|
| | [RZ/G2H] | [RZ/G2M V1.3] | [RZ/G2M V3.0] | [RZ/G2N] | [RZ/G2E] |
| 00 | — | — | — | — | — |
| 01 | FCPF ch1 | — | — | — | — |
| 05 | — | — | — | — | — |
| 07 | FCPVB ch1 | — | — | — | — |
| 08 | — | — | — | — | — |
| 09 | FCPVI ch1 | — | — | — | — |

Table 22.14 micro-TLB Assignment in IPMMUVC0

| micro-TLB | Master Module | | | | |
|-----------|----------------------|---------------|----------------------|----------------------|----------------------|
| | [RZ/G2H] | [RZ/G2M V1.3] | [RZ/G2M V3.0] | [RZ/G2N] | [RZ/G2E] |
| 00 | — | — | — | — | — |
| 01 | — | — | FCPCS (OS-ID = 0) | — | — |
| 02 | — | — | — | — | — |
| 03 | — | — | — | — | — |
| 04 | — | FCPCI ch0 | — | — | — |
| 05 | — | — | FCPCS (OS-ID = 4) | — | — |
| 06 | — | — | — | — | — |
| 07 | — | — | — | — | — |
| 08 | FCPCS (OS-ID = 0) | FCPCS | FCPCS (OS-ID = 0) | FCPCS (OS-ID = 0) | FCPCS (OS-ID = 0) |
| 12 | FCPCS (OS-ID = 4) | — | FCPCS (OS-ID = 4) | FCPCS (OS-ID = 4) | FCPCS (OS-ID = 4) |
| 16 | — | FCPF ch0 | FCPF ch0 | — | — |
| 19 | — | FCPVI ch0 | FCPVI ch0 | — | — |

Table 22.15 micro-TLB Assignment in IPMMUVC1

| micro-TLB | Master Module | | | | |
|-----------|----------------------|---------------|---------------|----------|----------|
| | [RZ/G2H] | [RZ/G2M V1.3] | [RZ/G2M V3.0] | [RZ/G2N] | [RZ/G2E] |
| 00 | — | — | — | — | — |
| 01 | — | — | — | — | — |
| 04 | — | — | — | — | — |
| 05 | — | — | — | — | — |
| 08 | FCPCS (OS-ID = 0) | — | — | — | — |
| 12 | FCPCS (OS-ID = 4) | — | — | — | — |

Table 22.16 micro-TLB Assignment in IPMMUPV0

| micro-TLB | Master Module | | | | |
|-----------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|----------------------------------|
| | [RZ/G2H] | [RZ/G2M V1.3] | [RZ/G2M V3.0] | [RZ/G2N] | [RZ/G2E] |
| 00 | 3DGE Master I/F 0 (OS-ID = 0) | 3DGE Master I/F 0 (OS-ID = 0) | 3DGE Master I/F 0 (OS-ID = 0) | 3DGE Master I/F 0 (OS-ID = 0) | 3DGE Master I/F 0 (OS-ID = 0) |
| 01 | 3DGE Master I/F 0 (OS-ID = 1) | 3DGE Master I/F 0 (OS-ID = 1) | 3DGE Master I/F 0 (OS-ID = 1) | 3DGE Master I/F 0 (OS-ID = 1) | 3DGE Master I/F 0 (OS-ID = 1) |
| 02 | 3DGE Master I/F 0 (OS-ID = 2) | 3DGE Master I/F 0 (OS-ID = 2) | 3DGE Master I/F 0 (OS-ID = 2) | 3DGE Master I/F 0 (OS-ID = 2) | 3DGE Master I/F 0 (OS-ID = 2) |
| 03 | 3DGE Master I/F 0 (OS-ID = 3) | 3DGE Master I/F 0 (OS-ID = 3) | 3DGE Master I/F 0 (OS-ID = 3) | 3DGE Master I/F 0 (OS-ID = 3) | 3DGE Master I/F 0 (OS-ID = 3) |
| 04 | 3DGE Master I/F 0 (OS-ID = 4) | 3DGE Master I/F 0 (OS-ID = 4) | 3DGE Master I/F 0 (OS-ID = 4) | 3DGE Master I/F 0 (OS-ID = 4) | 3DGE Master I/F 0 (OS-ID = 4) |
| 05 | 3DGE Master I/F 0 (OS-ID = 5) | 3DGE Master I/F 0 (OS-ID = 5) | 3DGE Master I/F 0 (OS-ID = 5) | 3DGE Master I/F 0 (OS-ID = 5) | 3DGE Master I/F 0 (OS-ID = 5) |
| 06 | 3DGE Master I/F 0 (OS-ID = 6) | 3DGE Master I/F 0 (OS-ID = 6) | 3DGE Master I/F 0 (OS-ID = 6) | 3DGE Master I/F 0 (OS-ID = 6) | 3DGE Master I/F 0 (OS-ID = 6) |
| 07 | 3DGE Master I/F 0 (OS-ID = 7) | 3DGE Master I/F 0 (OS-ID = 7) | 3DGE Master I/F 0 (OS-ID = 7) | 3DGE Master I/F 0 (OS-ID = 7) | 3DGE Master I/F 0 (OS-ID = 7) |

Table 22.17 micro-TLB Assignment in IPMMUPV1

| micro-TLB | Master Module | | | | |
|-----------|----------------------------------|----------------------------------|----------------------------------|----------|----------|
| | [RZ/G2H] | [RZ/G2M V1.3] | [RZ/G2M V3.0] | [RZ/G2N] | [RZ/G2E] |
| 00 | 3DGE Master I/F 1 (OS-ID = 0) | 3DGE Master I/F 1 (OS-ID = 0) | 3DGE Master I/F 1 (OS-ID = 0) | — | — |
| 01 | 3DGE Master I/F 1 (OS-ID = 1) | 3DGE Master I/F 1 (OS-ID = 1) | 3DGE Master I/F 1 (OS-ID = 1) | — | — |
| 02 | 3DGE Master I/F 1 (OS-ID = 2) | 3DGE Master I/F 1 (OS-ID = 2) | 3DGE Master I/F 1 (OS-ID = 2) | — | — |
| 03 | 3DGE Master I/F 1 (OS-ID = 3) | 3DGE Master I/F 1 (OS-ID = 3) | 3DGE Master I/F 1 (OS-ID = 3) | — | — |
| 04 | 3DGE Master I/F 1 (OS-ID = 4) | 3DGE Master I/F 1 (OS-ID = 4) | 3DGE Master I/F 1 (OS-ID = 4) | — | — |
| 05 | 3DGE Master I/F 1 (OS-ID = 5) | 3DGE Master I/F 1 (OS-ID = 5) | 3DGE Master I/F 1 (OS-ID = 5) | — | — |
| 06 | 3DGE Master I/F 1 (OS-ID = 6) | 3DGE Master I/F 1 (OS-ID = 6) | 3DGE Master I/F 1 (OS-ID = 6) | — | — |
| 07 | 3DGE Master I/F 1 (OS-ID = 7) | 3DGE Master I/F 1 (OS-ID = 7) | 3DGE Master I/F 1 (OS-ID = 7) | — | — |

Table 22.18 micro-TLB Assignment in IPMMUPV2

| micro-TLB | Master Module | | | | |
|-----------|----------------------------------|---------------|---------------|----------|----------|
| | [RZ/G2H] | [RZ/G2M V1.3] | [RZ/G2M V3.0] | [RZ/G2N] | [RZ/G2E] |
| 00 | 3DGE Master I/F 2 (OS-ID = 0) | — | — | — | — |
| 01 | 3DGE Master I/F 2 (OS-ID = 1) | — | — | — | — |
| 02 | 3DGE Master I/F 2 (OS-ID = 2) | — | — | — | — |
| 03 | 3DGE Master I/F 2 (OS-ID = 3) | — | — | — | — |
| 04 | 3DGE Master I/F 2 (OS-ID = 4) | — | — | — | — |
| 05 | 3DGE Master I/F 2 (OS-ID = 5) | — | — | — | — |
| 06 | 3DGE Master I/F 2 (OS-ID = 6) | — | — | — | — |
| 07 | 3DGE Master I/F 2 (OS-ID = 7) | — | — | — | — |

Table 22.19 micro-TLB Assignment in IPMMUPV3

| micro-TLB | Master Module | | | | |
|-----------|----------------------------------|---------------|---------------|----------|----------|
| | [RZ/G2H] | [RZ/G2M V1.3] | [RZ/G2M V3.0] | [RZ/G2N] | [RZ/G2E] |
| 00 | 3DGE Master I/F 3 (OS-ID = 0) | — | — | — | — |
| 01 | 3DGE Master I/F 3 (OS-ID = 1) | — | — | — | — |
| 02 | 3DGE Master I/F 3 (OS-ID = 2) | — | — | — | — |
| 03 | 3DGE Master I/F 3 (OS-ID = 3) | — | — | — | — |
| 04 | 3DGE Master I/F 3 (OS-ID = 4) | — | — | — | — |
| 05 | 3DGE Master I/F 3 (OS-ID = 5) | — | — | — | — |
| 06 | 3DGE Master I/F 3 (OS-ID = 6) | — | — | — | — |
| 07 | 3DGE Master I/F 3 (OS-ID = 7) | — | — | — | — |

Table 22.20 micro-TLB Assignment in IPMMUHC

| micro-TLB | Master Module | | | | |
|-----------|---|---|---|---|---|
| | [RZ/G2H] | [RZ/G2M V1.3] | [RZ/G2M V3.0] | [RZ/G2N] | [RZ/G2E] |
| 00 | PCIe ch0 | PCIe ch0 | PCIe ch0 | PCIe ch0 | PCIe ch0 |
| 01 | PCIe ch1 | PCIe ch1 | PCIe ch1 | PCIe ch1 | — |
| 02 | Serial-ATA Gen2* | — | — | Serial-ATA Gen2* | — |
| 04 | USB2.0 Host ch0 | USB2.0 Host ch0 | USB2.0 Host ch0 | USB2.0 Host ch0 | USB2.0 Host ch0 |
| 05 | USB2.0 Host ch1 | USB2.0 Host ch1 | USB2.0 Host ch1 | USB2.0 Host ch1 | — |
| 06 | — | — | — | — | — |
| 07 | — | — | — | — | — |
| 08 | USB-DMAC DDM | USB-DMAC DDM | USB-DMAC DDM | USB-DMAC DDM | USB-DMAC DDM |
| 09 | USB-DMAC ch0 | USB-DMAC ch0 | USB-DMAC ch0 | USB-DMAC ch0 | USB-DMAC ch0 |
| 10 | USB-DMAC ch1 | USB-DMAC ch1 | USB-DMAC ch1 | USB-DMAC ch1 | USB-DMAC ch1 |
| 12 | USB3.0 Host Controller /Host ch0 | USB3.0 Host Controller /Host ch0 | USB3.0 Host Controller /Host ch0 | USB3.0 Host Controller /Host ch0 | USB3.0 Host Controller /Host ch0 |
| 13 | USB3.0 Host Controller /Peripheral ch0 | USB3.0 Host Controller /Peripheral ch0 | USB3.0 Host Controller /Peripheral ch0 | USB3.0 Host Controller /Peripheral ch0 | USB3.0 Host Controller /Peripheral ch0 |
| 14 | — | — | — | — | — |
| 15 | — | — | — | — | — |

Note: * Serial-ATA Generation 2

Table 22.21 micro-TLB Assignment in IPMMUMP

| micro-TLB | Master Module | | | | |
|-----------|-----------------|-----------------|-----------------|-----------------|-----------------|
| | [RZ/G2H] | [RZ/G2M V1.3] | [RZ/G2M V3.0] | [RZ/G2N] | [RZ/G2E] |
| 00 | Audio-DMAC ch0 | Audio-DMAC ch0 | Audio-DMAC ch0 | Audio-DMAC ch0 | Audio-DMAC ch0 |
| 01 | Audio-DMAC ch1 | Audio-DMAC ch1 | Audio-DMAC ch1 | Audio-DMAC ch1 | Audio-DMAC ch1 |
| 02 | Audio-DMAC ch2 | Audio-DMAC ch2 | Audio-DMAC ch2 | Audio-DMAC ch2 | Audio-DMAC ch2 |
| 03 | Audio-DMAC ch3 | Audio-DMAC ch3 | Audio-DMAC ch3 | Audio-DMAC ch3 | Audio-DMAC ch3 |
| 04 | Audio-DMAC ch4 | Audio-DMAC ch4 | Audio-DMAC ch4 | Audio-DMAC ch4 | Audio-DMAC ch4 |
| 05 | Audio-DMAC ch5 | Audio-DMAC ch5 | Audio-DMAC ch5 | Audio-DMAC ch5 | Audio-DMAC ch5 |
| 06 | Audio-DMAC ch6 | Audio-DMAC ch6 | Audio-DMAC ch6 | Audio-DMAC ch6 | Audio-DMAC ch6 |
| 07 | Audio-DMAC ch7 | Audio-DMAC ch7 | Audio-DMAC ch7 | Audio-DMAC ch7 | Audio-DMAC ch7 |
| 08 | Audio-DMAC ch8 | Audio-DMAC ch8 | Audio-DMAC ch8 | Audio-DMAC ch8 | Audio-DMAC ch8 |
| 09 | Audio-DMAC ch9 | Audio-DMAC ch9 | Audio-DMAC ch9 | Audio-DMAC ch9 | Audio-DMAC ch9 |
| 10 | Audio-DMAC ch10 | Audio-DMAC ch10 | Audio-DMAC ch10 | Audio-DMAC ch10 | Audio-DMAC ch10 |
| 11 | Audio-DMAC ch11 | Audio-DMAC ch11 | Audio-DMAC ch11 | Audio-DMAC ch11 | Audio-DMAC ch11 |
| 12 | Audio-DMAC ch12 | Audio-DMAC ch12 | Audio-DMAC ch12 | Audio-DMAC ch12 | Audio-DMAC ch12 |
| 13 | Audio-DMAC ch13 | Audio-DMAC ch13 | Audio-DMAC ch13 | Audio-DMAC ch13 | Audio-DMAC ch13 |
| 14 | Audio-DMAC ch14 | Audio-DMAC ch14 | Audio-DMAC ch14 | Audio-DMAC ch14 | Audio-DMAC ch14 |
| 15 | Audio-DMAC ch15 | Audio-DMAC ch15 | Audio-DMAC ch15 | Audio-DMAC ch15 | Audio-DMAC ch15 |
| 16 | Audio-DMAC ch16 | Audio-DMAC ch16 | Audio-DMAC ch16 | Audio-DMAC ch16 | — |
| 17 | Audio-DMAC ch17 | Audio-DMAC ch17 | Audio-DMAC ch17 | Audio-DMAC ch17 | — |
| 18 | Audio-DMAC ch18 | Audio-DMAC ch18 | Audio-DMAC ch18 | Audio-DMAC ch18 | — |
| 19 | Audio-DMAC ch19 | Audio-DMAC ch19 | Audio-DMAC ch19 | Audio-DMAC ch19 | — |
| 20 | Audio-DMAC ch20 | Audio-DMAC ch20 | Audio-DMAC ch20 | Audio-DMAC ch20 | — |
| 21 | Audio-DMAC ch21 | Audio-DMAC ch21 | Audio-DMAC ch21 | Audio-DMAC ch21 | — |
| 22 | Audio-DMAC ch22 | Audio-DMAC ch22 | Audio-DMAC ch22 | Audio-DMAC ch22 | — |
| 23 | Audio-DMAC ch23 | Audio-DMAC ch23 | Audio-DMAC ch23 | Audio-DMAC ch23 | — |
| 24 | Audio-DMAC ch24 | Audio-DMAC ch24 | Audio-DMAC ch24 | Audio-DMAC ch24 | — |
| 25 | Audio-DMAC ch25 | Audio-DMAC ch25 | Audio-DMAC ch25 | Audio-DMAC ch25 | — |
| 26 | Audio-DMAC ch26 | Audio-DMAC ch26 | Audio-DMAC ch26 | Audio-DMAC ch26 | — |
| 27 | Audio-DMAC ch27 | Audio-DMAC ch27 | Audio-DMAC ch27 | Audio-DMAC ch27 | — |
| 28 | Audio-DMAC ch28 | Audio-DMAC ch28 | Audio-DMAC ch28 | Audio-DMAC ch28 | — |
| 29 | Audio-DMAC ch29 | Audio-DMAC ch29 | Audio-DMAC ch29 | Audio-DMAC ch29 | — |
| 30 | Audio-DMAC ch30 | Audio-DMAC ch30 | Audio-DMAC ch30 | Audio-DMAC ch30 | — |
| 31 | Audio-DMAC ch31 | Audio-DMAC ch31 | Audio-DMAC ch31 | Audio-DMAC ch31 | — |

Table 22.22 micro-TLB Assignment in IPMMUDS0

| micro-TLB | Master Module | | | | |
|-----------|---------------|---------------|---------------|---------------|---------------|
| | [RZ/G2H] | [RZ/G2M V1.3] | [RZ/G2M V3.0] | [RZ/G2N] | [RZ/G2E] |
| 00 | SYS-DMAC ch0 | SYS-DMAC ch0 | SYS-DMAC ch0 | SYS-DMAC ch0 | SYS-DMAC ch0 |
| 01 | SYS-DMAC ch1 | SYS-DMAC ch1 | SYS-DMAC ch1 | SYS-DMAC ch1 | SYS-DMAC ch1 |
| 02 | SYS-DMAC ch2 | SYS-DMAC ch2 | SYS-DMAC ch2 | SYS-DMAC ch2 | SYS-DMAC ch2 |
| 03 | SYS-DMAC ch3 | SYS-DMAC ch3 | SYS-DMAC ch3 | SYS-DMAC ch3 | SYS-DMAC ch3 |
| 04 | SYS-DMAC ch4 | SYS-DMAC ch4 | SYS-DMAC ch4 | SYS-DMAC ch4 | SYS-DMAC ch4 |
| 05 | SYS-DMAC ch5 | SYS-DMAC ch5 | SYS-DMAC ch5 | SYS-DMAC ch5 | SYS-DMAC ch5 |
| 06 | SYS-DMAC ch6 | SYS-DMAC ch6 | SYS-DMAC ch6 | SYS-DMAC ch6 | SYS-DMAC ch6 |
| 07 | SYS-DMAC ch7 | SYS-DMAC ch7 | SYS-DMAC ch7 | SYS-DMAC ch7 | SYS-DMAC ch7 |
| 08 | SYS-DMAC ch8 | SYS-DMAC ch8 | SYS-DMAC ch8 | SYS-DMAC ch8 | SYS-DMAC ch8 |
| 09 | SYS-DMAC ch9 | SYS-DMAC ch9 | SYS-DMAC ch9 | SYS-DMAC ch9 | SYS-DMAC ch9 |
| 10 | SYS-DMAC ch10 | SYS-DMAC ch10 | SYS-DMAC ch10 | SYS-DMAC ch10 | SYS-DMAC ch10 |
| 11 | SYS-DMAC ch11 | SYS-DMAC ch11 | SYS-DMAC ch11 | SYS-DMAC ch11 | SYS-DMAC ch11 |
| 12 | SYS-DMAC ch12 | SYS-DMAC ch12 | SYS-DMAC ch12 | SYS-DMAC ch12 | SYS-DMAC ch12 |
| 13 | SYS-DMAC ch13 | SYS-DMAC ch13 | SYS-DMAC ch13 | SYS-DMAC ch13 | SYS-DMAC ch13 |
| 14 | SYS-DMAC ch14 | SYS-DMAC ch14 | SYS-DMAC ch14 | SYS-DMAC ch14 | SYS-DMAC ch14 |
| 15 | SYS-DMAC ch15 | SYS-DMAC ch15 | SYS-DMAC ch15 | SYS-DMAC ch15 | SYS-DMAC ch15 |
| 16 | Ethernet AVB | Ethernet AVB | Ethernet AVB | Ethernet AVB | Ethernet AVB |
| 17 | — | — | — | — | — |
| 18 | TDDMAC | TDDMAC | TDDMAC | TDDMAC | — |
| 19 | — | — | — | — | — |
| 20 | — | — | — | — | — |
| 21 | — | — | — | — | — |
| 22 | — | — | — | — | — |
| 23 | — | — | — | — | — |

Table 22.23 micro-TLB Assignment in IPMMUDS1

| micro-TLB | Master Module | | | | |
|-----------|--------------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| | [RZ/G2H] | [RZ/G2M V1.3] | [RZ/G2M V3.0] | [RZ/G2N] | [RZ/G2E] |
| 00 | SYS-DMAC ch16 | SYS-DMAC ch16 | SYS-DMAC ch16 | SYS-DMAC ch16 | SYS-DMAC ch16 |
| 01 | SYS-DMAC ch17 | SYS-DMAC ch17 | SYS-DMAC ch17 | SYS-DMAC ch17 | SYS-DMAC ch17 |
| 02 | SYS-DMAC ch18 | SYS-DMAC ch18 | SYS-DMAC ch18 | SYS-DMAC ch18 | SYS-DMAC ch18 |
| 03 | SYS-DMAC ch19 | SYS-DMAC ch19 | SYS-DMAC ch19 | SYS-DMAC ch19 | SYS-DMAC ch19 |
| 04 | SYS-DMAC ch20 | SYS-DMAC ch20 | SYS-DMAC ch20 | SYS-DMAC ch20 | SYS-DMAC ch20 |
| 05 | SYS-DMAC ch21 | SYS-DMAC ch21 | SYS-DMAC ch21 | SYS-DMAC ch21 | SYS-DMAC ch21 |
| 06 | SYS-DMAC ch22 | SYS-DMAC ch22 | SYS-DMAC ch22 | SYS-DMAC ch22 | SYS-DMAC ch22 |
| 07 | SYS-DMAC ch23 | SYS-DMAC ch23 | SYS-DMAC ch23 | SYS-DMAC ch23 | SYS-DMAC ch23 |
| 08 | SYS-DMAC ch24 | SYS-DMAC ch24 | SYS-DMAC ch24 | SYS-DMAC ch24 | SYS-DMAC ch24 |
| 09 | SYS-DMAC ch25 | SYS-DMAC ch25 | SYS-DMAC ch25 | SYS-DMAC ch25 | SYS-DMAC ch25 |
| 10 | SYS-DMAC ch26 | SYS-DMAC ch26 | SYS-DMAC ch26 | SYS-DMAC ch26 | SYS-DMAC ch26 |
| 11 | SYS-DMAC ch27 | SYS-DMAC ch27 | SYS-DMAC ch27 | SYS-DMAC ch27 | SYS-DMAC ch27 |
| 12 | SYS-DMAC ch28 | SYS-DMAC ch28 | SYS-DMAC ch28 | SYS-DMAC ch28 | SYS-DMAC ch28 |
| 13 | SYS-DMAC ch29 | SYS-DMAC ch29 | SYS-DMAC ch29 | SYS-DMAC ch29 | SYS-DMAC ch29 |
| 14 | SYS-DMAC ch30 | SYS-DMAC ch30 | SYS-DMAC ch30 | SYS-DMAC ch30 | SYS-DMAC ch30 |
| 15 | SYS-DMAC ch31 | SYS-DMAC ch31 | SYS-DMAC ch31 | SYS-DMAC ch31 | SYS-DMAC ch31 |
| 16 | SYS-DMAC ch32 | SYS-DMAC ch32 | SYS-DMAC ch32 | SYS-DMAC ch32 | SYS-DMAC ch32 |
| 17 | SYS-DMAC ch33 | SYS-DMAC ch33 | SYS-DMAC ch33 | SYS-DMAC ch33 | SYS-DMAC ch33 |
| 18 | SYS-DMAC ch34 | SYS-DMAC ch34 | SYS-DMAC ch34 | SYS-DMAC ch34 | SYS-DMAC ch34 |
| 19 | SYS-DMAC ch35 | SYS-DMAC ch35 | SYS-DMAC ch35 | SYS-DMAC ch35 | SYS-DMAC ch35 |
| 20 | SYS-DMAC ch36 | SYS-DMAC ch36 | SYS-DMAC ch36 | SYS-DMAC ch36 | SYS-DMAC ch36 |
| 21 | SYS-DMAC ch37 | SYS-DMAC ch37 | SYS-DMAC ch37 | SYS-DMAC ch37 | SYS-DMAC ch37 |
| 22 | SYS-DMAC ch38 | SYS-DMAC ch38 | SYS-DMAC ch38 | SYS-DMAC ch38 | SYS-DMAC ch38 |
| 23 | SYS-DMAC ch39 | SYS-DMAC ch39 | SYS-DMAC ch39 | SYS-DMAC ch39 | SYS-DMAC ch39 |
| 24 | SYS-DMAC ch40 | SYS-DMAC ch40 | SYS-DMAC ch40 | SYS-DMAC ch40 | SYS-DMAC ch40 |
| 25 | SYS-DMAC ch41 | SYS-DMAC ch41 | SYS-DMAC ch41 | SYS-DMAC ch41 | SYS-DMAC ch41 |
| 26 | SYS-DMAC ch42 | SYS-DMAC ch42 | SYS-DMAC ch42 | SYS-DMAC ch42 | SYS-DMAC ch42 |
| 27 | SYS-DMAC ch43 | SYS-DMAC ch43 | SYS-DMAC ch43 | SYS-DMAC ch43 | SYS-DMAC ch43 |
| 28 | SYS-DMAC ch44 | SYS-DMAC ch44 | SYS-DMAC ch44 | SYS-DMAC ch44 | SYS-DMAC ch44 |
| 29 | SYS-DMAC ch45 | SYS-DMAC ch45 | SYS-DMAC ch45 | SYS-DMAC ch45 | SYS-DMAC ch45 |
| 30 | SYS-DMAC ch46 | SYS-DMAC ch46 | SYS-DMAC ch46 | SYS-DMAC ch46 | SYS-DMAC ch46 |
| 31 | SYS-DMAC ch47 | SYS-DMAC ch47 | SYS-DMAC ch47 | SYS-DMAC ch47 | SYS-DMAC ch47 |
| 32 | SDHI ch0 | SDHI ch0 | SDHI ch0 | SDHI ch0 | SDHI ch0 |
| 33 | SDHI ch1 | SDHI ch1 | SDHI ch1 | SDHI ch1 | SDHI ch1 |
| 34 | SDHI ch2 (MMC-IF ch0) | SDHI ch2 (MMC-IF ch0) | SDHI ch2 (MMC-IF ch0) | SDHI ch2 (MMC-IF ch0) | — |
| 35 | SDHI ch3 (MMC-IF ch1) | SDHI ch3 (MMC-IF ch1) | SDHI ch3 (MMC-IF ch1) | SDHI ch3 (MMC-IF ch1) | SDHI ch3 (MMC-IF ch1) |
| 36 | — | — | — | — | — |
| 37 | — | — | — | — | — |
| 38 | — | R-NANDC | R-NANDC | R-NANDC | R-NANDC |

22.5 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

22.5.1 IPMMU naming

Table 22.24 shows the correspondence between each IPMMU and each bus domain.

Table 22.24 IPMMU naming

| IPMMU | Bus Domain | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----------|------------------------------------|--------|-------------|-------------|--------|--------|
| IPMMU-VI0 | VIO (Video IO) domain AXI | √ | √ | √ | √ | √ |
| IPMMU-VI1 | VIO (Video IO) domain AXI | √ | — | — | — | — |
| IPMMU-VP0 | VP (Video Processor) domain AXI | √ | — | — | √ | √ |
| IPMMU-VP1 | VP (Video Processor) domain AXI | √ | — | — | — | — |
| IPMMU-VC0 | VC (Video Codec) domain AXI | √ | √ | √ | √ | √ |
| IPMMU-VC1 | VC (Video Codec) domain AXI | √ | — | — | — | — |
| IPMMU-PV0 | 3DG (3D-Graphics) domain AXI | √ | √ | √ | √ | √ |
| IPMMU-PV1 | 3DG (3D-Graphics) domain AXI | √ | √ | √ | — | — |
| IPMMU-PV2 | 3DG (3D-Graphics) domain AXI | √ | — | — | — | — |
| IPMMU-PV3 | 3DG (3D-Graphics) domain AXI | √ | — | — | — | — |
| IPMMU-IR | IMP domain AXI | √ | √ | √ | √ | — |
| IPMMU-HC | HC (High Communication) domain AXI | √ | √ | √ | √ | √ |
| IPMMU-MP | Audio domain AXI | √ | √ | √ | √ | √ |
| IPMMU-DS0 | Peripheral domain AXI | √ | √ | √ | √ | √ |
| IPMMU-DS1 | Peripheral domain AXI | √ | √ | √ | √ | √ |
| IPMMU-MM | Main Memory domain AXI | √ | √ | √ | √ | √ |

22.5.2 IPMMU configuration

22.5.2.1 IPMMU configuration for FCPCS [RZ/G2H, RZ/G2M V3.0]

In case that the VDPB, VCPLF and iVDP1C use the address translation function of IPMMU, uTLB8 and uTLB12 of both IPMMU-VC0 and IPMMU-VC1 need to be set. Because the request from each master IP is split to the two master ports at FCPCS. Each master port is connected to the each IPMMU. (IPMMU-VC0 and IPMMU-VC1). Figure 22.14 shows the connection of FCPCS and each IPMMU. When the MMU configuration is changed, TLB flush operation is required to IPMMU-VC0 and IPMMU-VC1 as well as each uTLB of both IPMMUs.

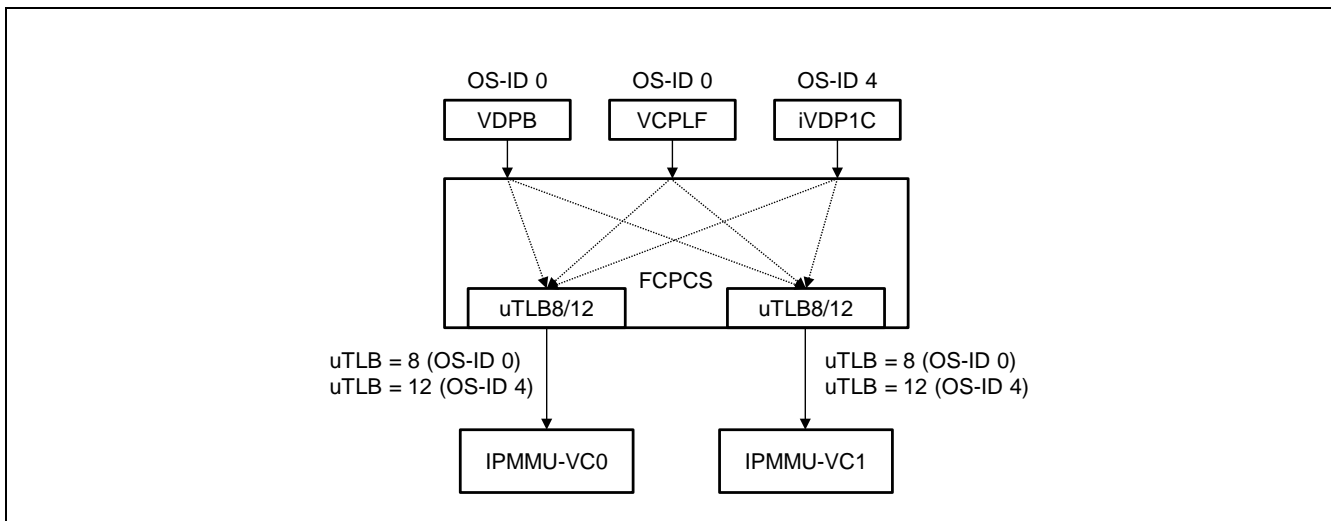


Figure 22.14 connection of FCPCS and IPMMU

For RZ/G2M V3.0, uTLB1, uTLB5, uTLB8, and uTLB12 of IPMMU-VC0 need to be set. Each mater port is connected to the IPMMU-VC0. Figure 22.15 shows the connection of FCPCS and IPMMU.

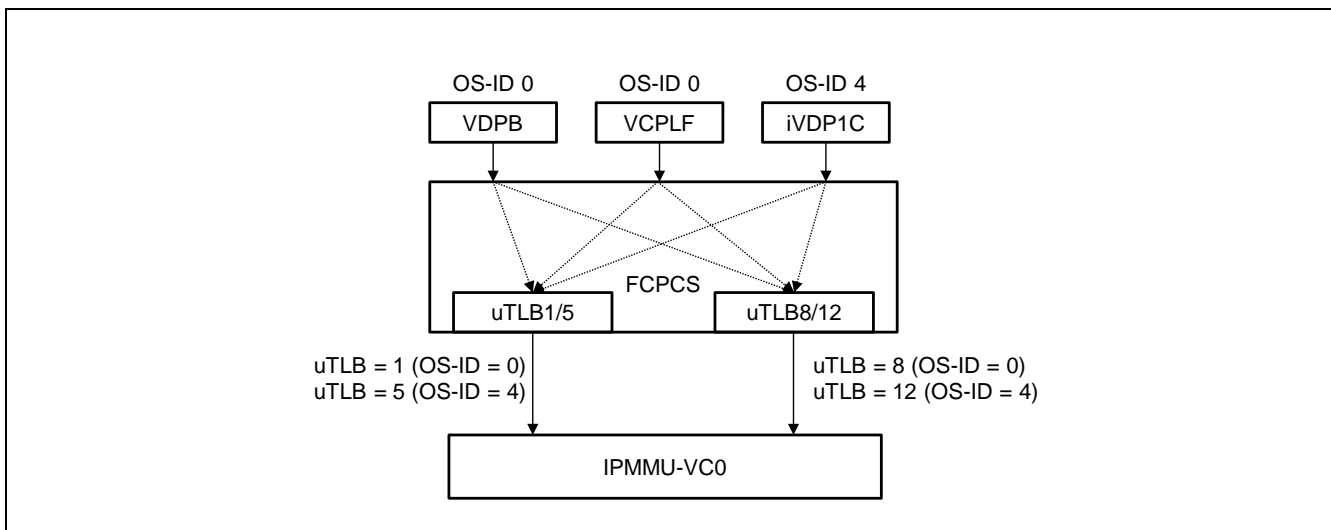


Figure 22.15 connection of FCPCS and IPMMU

22.5.2.2 IPMMU configuration for 3DG [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]

In case that the 3DG engine use the address translation function of IPMMU, uTLB corresponding to OS-ID of IPMMU-PVn (n = 0, 1, 2, 3 [RZ/G2H], n = 0, 1 [RZ/G2M V1.3, RZ/G2M V3.0]) need to be set as well. Figure 22.16 and Figure 22.17 show the connection of 3DG engine and each IPMMU. When the MMU configuration is changed, TLB flush operation is required to IPMMU-PVn as well as each uTLB of both IPMMUs.

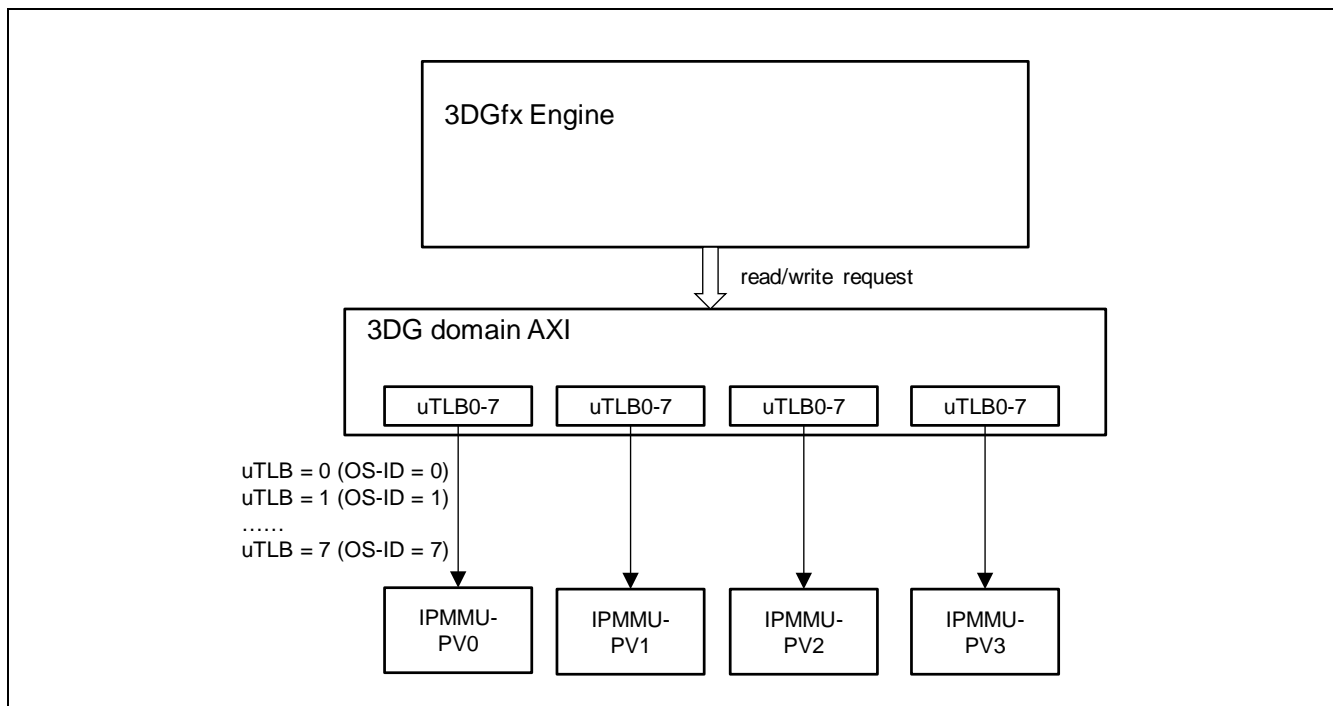


Figure 22.16 connection of 3DG engine and IPMMU (RZ/G2H)

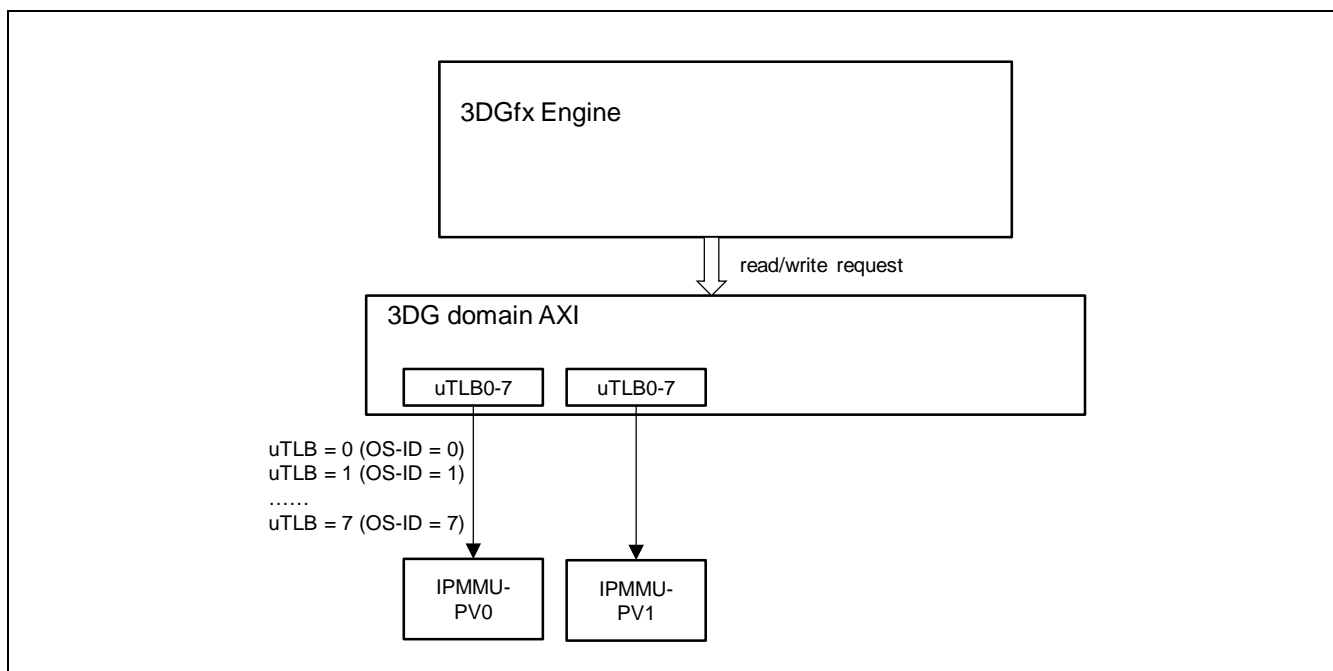


Figure 22.17 connection of 3DG engine and IPMMU (RZ/G2M V1.3, RZ/G2M V3.0)

22.5.3 Restrictions for IPMMU

22.5.3.1 Page size and Section size usage in VMSAv8-32

IPMMU cannot handle both 4KB page and 64KB page at the same MMU context in VMSAv8-32. In case of using the 64KB page, IMTTBCRn.PGSZ = 1 is required. Also 1MB section and 16MB super-section cannot be handled. In case of using 16MB section, IMTTBCRn.SCSZ = 1 is required.

22.5.3.2 4KB page usage in VMSAv8-64

Four level translation is not supported in IPMMU. In case of using 4KB page granule, we should set "0x1" to "SL" field. (start from 1st level) and use concatenated translation table. Architecturally, this table optimization is supported in stage 2 translation, but IPMMU is supported this mode in stage 1 translation as well in order to reduce the number of page table walk. In table lookup, MSB of input address (IA[39]) is used for the page table entry selection for upper or lower table.

22.5.3.3 Error Handling in VMSAv8-64

Architecturally, Address Size Fault is described in VMSAv8-64. But this fault is not able to handle in IPMMU. On the other hand, the NS bit in a Non-secure translation table descriptor need to be set in IPMMU although it is ignored in VMSAv8-64. In order to avoid "Secure Access Fault", set "NS bit" in a Non-secure translation table descriptor.

22.5.3.4 Start Level for Page Table Walk

Start Level for page table walk need to be set under below condition. Refer to Table 22.25.

Table 22.25 Start Level for Page Table Walk

| IMCTRn.VA64 = 1 IMTTBCRn.PGSZ = 0 (use 4KB page granule) | | |
|--|--|--|
| Look Up | stage 1 Translation | stage 2 Translation |
| Level 0 * | No support | No Support |
| Level 1 | base address[39:x]: Input Address[y:30] if IMTTBCRn SL[1:0] = 2'b01 $x = (37 - \text{TSZn} (n = 0, 1)) [24^* \leq \text{TSZn} \leq 33]$ $y = x + 26$ Other values go to Level 2 | base address[39:x]: Input Address[y:30] if IMTTBCRn SL[1:0] = 2'b01 $x = (37 - \text{TSZn} (n = 0, 1)) [24^* \leq \text{TSZn} \leq 33]$ $y = x + 26$ Other values go to Level 2 |
| Level 2 | base address [39:x], Input Address[y:21] if IMTTBCRn SL[1:0] = 2'b00 $x = (46 - \text{TSZn} (n = 0, 1)) [30 \leq \text{TSZn} \leq 39]$ $y = x + 17$ Other values $x = (46 - \text{TSZn} (n = 0, 1)) [34 \leq \text{TSZn} \leq 39]$ $y = x + 17$ | base address[39:x]: Input Address[y:21] if IMTTBCRn.SL[1:0] = 2'b00 $x = (46 - \text{TSZn} (n = 0, 1)) [30 \leq \text{TSZn} \leq 39]$ $y = x + 17$ Other values No support |
| IMCTRn.VA64 = 1 IMTTBCRn.PGSZ = 1 (use 64KB page granule) | | |
| Level 1 | No Support | No Support |
| Level 2 | base address [39:x], Input Address[y:29] if IMTTBCRn SL[1:0] = 2'b01 $x = (38 - \text{TSZn} (n = 0, 1)) [24 \leq \text{TSZn} \leq 34]$ $y = x + 25$ Other values go to Level 3 | base address [39:x], Input Address[y:29] if IMTTBCRn SL[1:0] = 2'b01 $x = (38 - \text{TSZn} (n = 0, 1)) [24 \leq \text{TSZn} \leq 34]$ $y = x + 25$ Other values go to Level 3 |
| Level 3 | base address [39:x], Input Address [y:16] if IMTTBCRn.SL[1:0] = 2'b00 $x = (51 - \text{TSZn} (n = 0, 1)) [31 \leq \text{TSZn} \leq 39]$ $y = x + 12$ Other values $x = (51 - \text{TSZn} (n = 0, 1)) [35 \leq \text{TSZn} \leq 39]$ $y = x + 12$ | base address [39:x], Input Address [y:16] if IMTTBCRn.SL[1:0] = 2'b00 $x = (51 - \text{TSZn} (n = 0, 1)) [31 \leq \text{TSZn} \leq 39]$ $y = x + 12$ Other values $x = (51 - \text{TSZn} (n = 0, 1)) [35 \leq \text{TSZn} \leq 39]$ $y = x + 12$ |
| IMCTRn.VA64 = 0 | | |
| Level 1 | base address[39:x]: Input Address[y:30] if IMTTBCRn SL[1] = 1'b1 $x = (5 - \text{TSZn} (n = 0, 1)) [0 \leq \text{TSZn} \leq 1]$ $y = x + 26$ Other values go to Level 2 | base address[39:x]: Input Address[y:30] if IMTTBCRn.SL[1] = 1'b1 $x = (5 - \text{TSZ0}) [-2 \leq \text{TSZ0} \leq 1]$ $y = x + 26$ Other values $x = (5 - \text{TSZ0}) [-8 \leq \text{TSZ0} \leq -3]$ $y = x + 26$ |
| Level 2 | base address[39:x]: Input Address[y:21] if IMTTBCRn.SL[1] = 1'b0 $x = (14 - \text{TSZn} (n = 0, 1)) [2 \leq \text{TSZn} \leq 7]$ $y = x + 17$ Other values No support | base address[39:x]: Input Address[y:21] if IMTTBCRn.SL[1] = 1'b0 $x = (14 - \text{TSZ0}) [-2 \leq \text{TSZ0} \leq 7]$ $y = x + 17$ Other values No support |

Note: * Not support level 0 page table walk. Refer to 22.5.3.2 4KB page usage in VMSAv8-64.

22.5.3.5 Virtual address space range

In case of using MMU, Virtual address space range cannot be configured less than or equal 128MB. In case of supporting this address range, set virtual address space range to 256MB by setting TSZ0/TSZ1 and unnecessary area is not mapped by MMU.

22.5.3.6 Treatments of ASID feature

In RZ/G2, IPMMU is not supported ASID feature. Software should not set the value except “0” in IMUASID register.

22.5.3.7 Disabling the address translation by IPMMU

Even if a master IP is disabling the address translation by IPMMU, IMUCTRn.TTSEL[2:0] needs to be set unused MMU context number. It means that only seven MMU context can be used for the address translation by IPMMU in this case.

22.5.3.8 Restriction for DMAC transfer [RZ/G2M V1.3 only]

SYS-DMAC cannot support read interleaving feature. In order to avoid this occurrence, it is required to apply the following restriction in case that SYS-DMAC need to access address space exceeding 32-bit.

1. Don't set the transfer size exceeding 16 bytes in case of using IPMMU
2. Use 1GB page in order to avoid TLB miss.
3. Don't use IPMMU. Use extended address function which is implemented in SYS-DMAC.

22.5.3.9 Restriction for OS-ID support [RZ/G2M V1.3 only]

In order to perform address translation for 3DG Engine by IPMMU normally, context switching by OS-ID cannot be used. It is required to set the same MMU context number to each uTLB for 3DG Engine.

22.5.3.10 Restriction for TLB location [RZ/G2M V1.3 only]

TLB for address translation by IPMMU cannot be located at System-RAM. It is required to put TLB in the main memory.

22.5.3.11 Restriction for TLB flush operation [RZ/G2M V1.3 only]

In order to perform address translation by IPMMU normally, it is limited to replace the MMU context dynamically so as not to perform TLB flush operation. Use 1GB page to avoid this occurrence.

22.5.3.12 Restriction for IPMMU cache [RZ/G2M V1.3, RZ/G2N, RZ/G2E]

In order to perform address translation by IPMMU normally, it is limited not to use IPMMU cache. In RZ/G2N and RZ/G2E, it is required to disable IPMMU cache by setting “IMSCTLR.dismmu” to 1. In RZ/G2M V1.3, IPMMU cache is always disabled by hardware. IPMMU cache cannot be used.

22.5.3.13 Restriction for PMB

In order to perform address translation by IPMMU normally, it is limited not to use PMB mode in secure mode. Use PMB mode only in non-secure mode.

22.5.3.14 Restriction for PMB with Stage 2 Translation [RZ/G2M V1.3 only]

In order to perform address translation by IPMMU normally, it is limited not to use PMB mode with Stage 2 Translation. Use MMU mode only in Stage 2 Translation.

23. Direct Memory Access Controller for System (SYS-DMAC)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

This LSI includes a direct memory access controller for system (SYS-DMAC). The SYS-DMAC can be used in place of the CPU to handle high-speed data transfer to and from an external memory, the on-chip memory, memory-mapped external devices, or on-chip peripheral modules.

23.1 Overview

23.1.1 Features

- Up to 48 channels are available.
- 1-T byte physical address space
- Transfer data length: Byte, word (2 bytes), longword (4 bytes), 8 bytes, 16 bytes, 32 bytes, and 64 bytes
- Maximum number of transfer times: 16,777,215
- Address mode: Dual address mode
- Transfer requests:

Requests from on-chip peripheral modules or auto requests can be selected. The following modules can issue on-chip peripheral module requests.

IIC-PMIC, TPU0, HSCIF0/1/2/3/4, MSIOF0/1/2/3, SCIF0/1/2/3/4/5, CANFD, I2C0/1/2/3/4/5/6.

Note: The availability of these modules and channels depends on the product. Refer to Table 23.5 for available modules in each product.

- Selectable bus modes:
Normal speed mode or slow mode can be selected for each channel.
- Either fixed priority or round-robin arbitration can be selected for use in arbitration among the transfer channels.
- Interrupt request:

The SYS-DMAC can be set up to generate an interrupt request for the CPU upon completion of transfer under the control of one stage of the descriptor memory, at the end of the data transfer, in response to an MMU error, and in response to an address error.

- Descriptor memory function:

Up to 128 sets of the settings for the source address register, destination address register, and transfer count register are available (if use of the descriptor memory is only enabled for one channel) for use in setting up consecutive DMA transfers (the memory can hold register values for up to 256 stages of transfer when the external memory is selected). An infinite repeat mode is also available.

Figure 23.1 shows the block diagram of the SYS-DMAC.

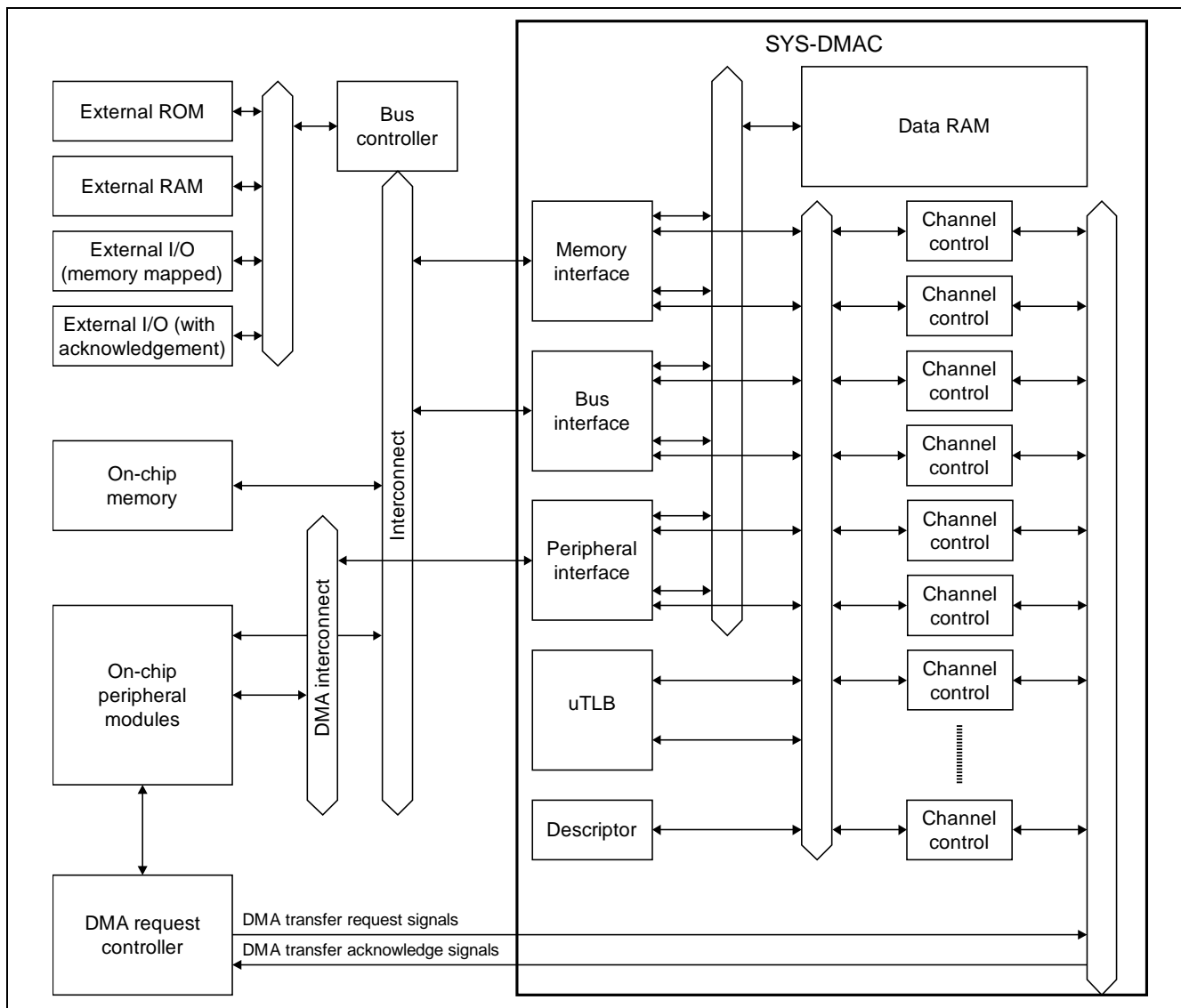


Figure 23.1 Block Diagram of the SYS-DMAC

23.1.2 External Pins

There are no external pins relevant to the SYS-DMAC.

23.1.3 Register Configuration

Table 23.1 lists the registers of the SYS-DMAC. Table 23.2 shows the register states of the SYS-DMAC in each operating mode.

Each product of the Second Generation RZ/G series has the same configuration of registers for the SYS-DMAC.

Table 23.1 Register Configuration of the SYS-DMAC

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--|--------------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA interrupt status register (for channels 0 to 15) | DMAISTA_0 | R | H'E670_0020 | 32 | √ | √ | √ | √ |
| DMA secure control register (for channels 0 to 15) | DMASEC_0 | R/W | H'E670_0030 | 32 | √ | √ | √ | √ |
| DMA operation register (for channels 0 to 15) | DMAOR_0 | R/W | H'E670_0060 | 16 | √ | √ | √ | √ |
| DMA channel clear register (for channels 0 to 15) | DMACHCLR_0 | W | H'E670_0080 | 32 | √ | √ | √ | √ |
| DPRAM secure control register (for channels 0 to 15) | DMADPSEC_0 | R/W | H'E670_00A0 | 32 | √ | √ | √ | √ |
| DMA source address register_0 | DMASAR_0 | R/W | H'E670_8000 H'E670_8020* | 32 | √ | √ | √ | √ |
| DMA destination address register_0 | DMADAR_0 | R/W | H'E670_8004 H'E670_8024* | 32 | √ | √ | √ | √ |
| DMA transfer count register_0 | DMATCR_0 | R/W | H'E670_8008 | 32 | √ | √ | √ | √ |
| DMA transfer size register_0 | DMATSR_0 | R/W | H'E670_8028* | 32 | √ | √ | √ | √ |
| DMA channel control register_0 | DMACHCR_0 | R/W | H'E670_800C H'E670_802C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_0 | DMATCRB_0 | R/W | H'E670_8018 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_0 | DMATSRB_0 | R/W | H'E670_8038* | 32 | √ | √ | √ | √ |
| DMA channel control register B_0 | DMACHCRB_0 | R/W | H'E670_801C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_0 | DMARS_0 | R/W | H'E670_8040 | 16 | √ | √ | √ | √ |
| DMA buffer control register_0 | DMABUFCR_0 | R/W | H'E670_8048 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_0 | DMADPBASE_0 | R/W | H'E670_8050 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_0 | DMADPCR_0 | R/W | H'E670_8054 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_0 | DMAFIXSAR_0 | R/W | H'E670_8010 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_0 | DMAFIXDAR_0 | R/W | H'E670_8014 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_0 | DMAFIXDP BASE_0 | R/W | H'E670_8060 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--|----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA source address register_1 | DMASAR_1 | R/W | H'E670_8080 H'E670_80A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_1 | DMADAR_1 | R/W | H'E670_8084 H'E670_80A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_1 | DMATCR_1 | R/W | H'E670_8088 | 32 | √ | √ | √ | √ |
| DMA transfer size register_1 | DMATSR_1 | R/W | H'E670_80A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_1 | DMACHCR_1 | R/W | H'E670_808C H'E670_80AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_1 | DMATCRB_1 | R/W | H'E670_8098 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_1 | DMATSRB_1 | R/W | H'E670_80B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_1 | DMACHCRB_1 | R/W | H'E670_809C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_1 | DMARS_1 | R/W | H'E670_80C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_1 | DMABUFCR_1 | R/W | H'E670_80C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_1 | DMADPBASE_1 | R/W | H'E670_80D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_1 | DMADPCR_1 | R/W | H'E670_80D4 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_1 | DMAFIXDPBASE_1 | R/W | H'E670_80E0 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_1 | DMAFIXSAR_1 | R/W | H'E670_8090 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_1 | DMAFIXDAR_1 | R/W | H'E670_8094 | 32 | √ | √ | √ | √ |
| DMA source address register_2 | DMASAR_2 | R/W | H'E670_8100 H'E670_8120* | 32 | √ | √ | √ | √ |
| DMA destination address register_2 | DMADAR_2 | R/W | H'E670_8104 H'E670_8124* | 32 | √ | √ | √ | √ |
| DMA transfer count register_2 | DMATCR_2 | R/W | H'E670_8108 | 32 | √ | √ | √ | √ |
| DMA transfer size register_2 | DMATSR_2 | R/W | H'E670_8128* | 32 | √ | √ | √ | √ |
| DMA channel control register_2 | DMACHCR_2 | R/W | H'E670_810C H'E670_812C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_2 | DMATCRB_2 | R/W | H'E670_8118 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_2 | DMATSRB_2 | R/W | H'E670_8138* | 32 | √ | √ | √ | √ |
| DMA channel control register B_2 | DMACHCRB_2 | R/W | H'E670_811C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_2 | DMARS_2 | R/W | H'E670_8140 | 16 | √ | √ | √ | √ |
| DMA buffer control register_2 | DMABUFCR_2 | R/W | H'E670_8148 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_2 | DMADPBASE_2 | R/W | H'E670_8150 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--|----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA descriptor control register_2 | DMADPCR_2 | R/W | H'E670_8154 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_2 | DMAFIXSAR_2 | R/W | H'E670_8110 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_2 | DMAFIXDAR_2 | R/W | H'E670_8114 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_2 | DMAFIXDPBASE_2 | R/W | H'E670_8160 | 32 | √ | √ | √ | √ |
| DMA source address register_3 | DMASAR_3 | R/W | H'E670_8180 H'E670_81A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_3 | DMADAR_3 | R/W | H'E670_8184 H'E670_81A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_3 | DMATCR_3 | R/W | H'E670_8188 | 32 | √ | √ | √ | √ |
| DMA transfer size register_3 | DMATSR_3 | R/W | H'E670_81A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_3 | DMACHCR_3 | R/W | H'E670_818C H'E670_81AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_3 | DMATCRB_3 | R/W | H'E670_8198 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_3 | DMATSRB_3 | R/W | H'E670_81B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_3 | DMACHCRB_3 | R/W | H'E670_819C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_3 | DMARS_3 | R/W | H'E670_81C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_3 | DMABUFCR_3 | R/W | H'E670_81C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_3 | DMADPBASE_3 | R/W | H'E670_81D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_3 | DMADPCR_3 | R/W | H'E670_81D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_3 | DMAFIXSAR_3 | R/W | H'E670_8190 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_3 | DMAFIXDAR_3 | R/W | H'E670_8194 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_3 | DMAFIXDPBASE_3 | R/W | H'E670_81E0 | 32 | √ | √ | √ | √ |
| DMA source address register_4 | DMASAR_4 | R/W | H'E670_8200 H'E670_8220* | 32 | √ | √ | √ | √ |
| DMA destination address register_4 | DMADAR_4 | R/W | H'E670_8204 H'E670_8224* | 32 | √ | √ | √ | √ |
| DMA transfer count register_4 | DMATCR_4 | R/W | H'E670_8208 | 32 | √ | √ | √ | √ |
| DMA transfer size register_4 | DMATSR_4 | R/W | H'E670_8228* | 32 | √ | √ | √ | √ |
| DMA channel control register_4 | DMACHCR_4 | R/W | H'E670_820C H'E670_822C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_4 | DMATCRB_4 | R/W | H'E670_8218 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_4 | DMATSRB_4 | R/W | H'E670_8238* | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--|----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA channel control register B_4 | DMACHCRB_4 | R/W | H'E670_821C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_4 | DMARS_4 | R/W | H'E670_8240 | 16 | √ | √ | √ | √ |
| DMA buffer control register_4 | DMABUFCR_4 | R/W | H'E670_8248 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_4 | DMADPBASE_4 | R/W | H'E670_8250 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_4 | DMADPCR_4 | R/W | H'E670_8254 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_4 | DMAFIXSAR_4 | R/W | H'E670_8210 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_4 | DMAFIXDAR_4 | R/W | H'E670_8214 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_4 | DMAFIXDPBASE_4 | R/W | H'E670_8260 | 32 | √ | √ | √ | √ |
| DMA source address register_5 | DMASAR_5 | R/W | H'E670_8280 H'E670_82A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_5 | DMADAR_5 | R/W | H'E670_8284 H'E670_82A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_5 | DMATCR_5 | R/W | H'E670_8288 | 32 | √ | √ | √ | √ |
| DMA transfer size register_5 | DMATSR_5 | R/W | H'E670_82A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_5 | DMACHCR_5 | R/W | H'E670_828C H'E670_82AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_5 | DMATCRB_5 | R/W | H'E670_8298 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_5 | DMATSRB_5 | R/W | H'E670_82B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_5 | DMACHCRB_5 | R/W | H'E670_829C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_5 | DMARS_5 | R/W | H'E670_82C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_5 | DMABUFCR_5 | R/W | H'E670_82C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_5 | DMADPBASE_5 | R/W | H'E670_82D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_5 | DMADPCR_5 | R/W | H'E670_82D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_5 | DMAFIXSAR_5 | R/W | H'E670_8290 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_5 | DMAFIXDAR_5 | R/W | H'E670_8294 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_5 | DMAFIXDPBASE_5 | R/W | H'E670_82E0 | 32 | √ | √ | √ | √ |
| DMA source address register_6 | DMASAR_6 | R/W | H'E670_8300 H'E670_8320* | 32 | √ | √ | √ | √ |
| DMA destination address register_6 | DMADAR_6 | R/W | H'E670_8304 H'E670_8324* | 32 | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--|----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA transfer count register_6 | DMATCR_6 | R/W | H'E670_8308 | 32 | √ | √ | √ | √ |
| DMA transfer size register_6 | DMATSR_6 | R/W | H'E670_8328* | 32 | √ | √ | √ | √ |
| DMA channel control register_6 | DMACHCR_6 | R/W | H'E670_830C H'E670_832C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_6 | DMATCRB_6 | R/W | H'E670_8318 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_6 | DMATSRB_6 | R/W | H'E670_8338* | 32 | √ | √ | √ | √ |
| DMA channel control register B_6 | DMACHCRB_6 | R/W | H'E670_831C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_6 | DMARS_6 | R/W | H'E670_8340 | 16 | √ | √ | √ | √ |
| DMA buffer control register_6 | DMABUFCR_6 | R/W | H'E670_8348 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_6 | DMADPBASE_6 | R/W | H'E670_8350 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_6 | DMADPCR_6 | R/W | H'E670_8354 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_6 | DMAFIXSAR_6 | R/W | H'E670_8310 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_6 | DMAFIXDAR_6 | R/W | H'E670_8314 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_6 | DMAFIXDPBASE_6 | R/W | H'E670_8360 | 32 | √ | √ | √ | √ |
| DMA source address register_7 | DMASAR_7 | R/W | H'E670_8380 H'E670_83A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_7 | DMADAR_7 | R/W | H'E670_8384 H'E670_83A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_7 | DMATCR_7 | R/W | H'E670_8388 | 32 | √ | √ | √ | √ |
| DMA transfer size register_7 | DMATSR_7 | R/W | H'E670_83A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_7 | DMACHCR_7 | R/W | H'E670_838C H'E670_83AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_7 | DMATCRB_7 | R/W | H'E670_8398 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_7 | DMATSRB_7 | R/W | H'E670_83B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_7 | DMACHCRB_7 | R/W | H'E670_839C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_7 | DMARS_7 | R/W | H'E670_83C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_7 | DMABUFCR_7 | R/W | H'E670_83C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_7 | DMADPBASE_7 | R/W | H'E670_83D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_7 | DMADPCR_7 | R/W | H'E670_83D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_7 | DMAFIXSAR_7 | R/W | H'E670_8390 | 32 | √ | √ | √ | √ |

| | | | | | Second Generation RZ/G Series Products | | | |
|--|----------------|-----|-----------------------------|----------------|---|----------------------------|--------|--------|
| Name | Abbreviation | R/W | Address | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| | | | | | √ | √ | √ | √ |
| DMA fixed destination address register_7 | DMAFIXDAR_7 | R/W | H'E670_8394 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_7 | DMAFIXDPBASE_7 | R/W | H'E670_83E0 | 32 | √ | √ | √ | √ |
| DMA source address register_8 | DMASAR_8 | R/W | H'E670_8400 H'E670_8420* | 32 | √ | √ | √ | √ |
| DMA destination address register_8 | DMADAR_8 | R/W | H'E670_8404 H'E670_8424* | 32 | √ | √ | √ | √ |
| DMA transfer count register_8 | DMATCR_8 | R/W | H'E670_8408 | 32 | √ | √ | √ | √ |
| DMA transfer size register_8 | DMATSR_8 | R/W | H'E670_8428* | 32 | √ | √ | √ | √ |
| DMA channel control register_8 | DMACHCR_8 | R/W | H'E670_840C H'E670_842C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_8 | DMATCRB_8 | R/W | H'E670_8418 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_8 | DMATSRB_8 | R/W | H'E670_8438* | 32 | √ | √ | √ | √ |
| DMA channel control register B_8 | DMACHCRB_8 | R/W | H'E670_841C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_8 | DMARS_8 | R/W | H'E670_8440 | 16 | √ | √ | √ | √ |
| DMA buffer control register_8 | DMABUFCR_8 | R/W | H'E670_8448 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_8 | DMADPBASE_8 | R/W | H'E670_8450 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_8 | DMADPCR_8 | R/W | H'E670_8454 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_8 | DMAFIXSAR_8 | R/W | H'E670_8410 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_8 | DMAFIXDAR_8 | R/W | H'E670_8414 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_8 | DMAFIXDPBASE_8 | R/W | H'E670_8460 | 32 | √ | √ | √ | √ |
| DMA source address register_9 | DMASAR_9 | R/W | H'E670_8480 H'E670_84A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_9 | DMADAR_9 | R/W | H'E670_8484 H'E670_84A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_9 | DMATCR_9 | R/W | H'E670_8488 | 32 | √ | √ | √ | √ |
| DMA transfer size register_9 | DMATSR_9 | R/W | H'E670_84A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_9 | DMACHCR_9 | R/W | H'E670_848C H'E670_84AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_9 | DMATCRB_9 | R/W | H'E670_8498 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_9 | DMATSRB_9 | R/W | H'E670_84B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_9 | DMACHCRB_9 | R/W | H'E670_849C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_9 | DMARS_9 | R/W | H'E670_84C0 | 16 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA buffer control register_9 | DMABUFCR_9 | R/W | H'E670_84C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_9 | DMADPBASE_9 | R/W | H'E670_84D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_9 | DMADPCR_9 | R/W | H'E670_84D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_9 | DMAFIXSAR_9 | R/W | H'E670_8490 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_9 | DMAFIXDAR_9 | R/W | H'E670_8494 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_9 | DMAFIXDPBASE_9 | R/W | H'E670_84E0 | 32 | √ | √ | √ | √ |
| DMA source address register_10 | DMASAR_10 | R/W | H'E670_8500 H'E670_8520* | 32 | √ | √ | √ | √ |
| DMA destination address register_10 | DMADAR_10 | R/W | H'E670_8504 H'E670_8524* | 32 | √ | √ | √ | √ |
| DMA transfer count register_10 | DMATCR_10 | R/W | H'E670_8508 | 32 | √ | √ | √ | √ |
| DMA transfer size register_10 | DMATSR_10 | R/W | H'E670_8528* | 32 | √ | √ | √ | √ |
| DMA channel control register_10 | DMACHCR_10 | R/W | H'E670_850C H'E670_852C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_10 | DMATCRB_10 | R/W | H'E670_8518 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_10 | DMATSRB_10 | R/W | H'E670_8538* | 32 | √ | √ | √ | √ |
| DMA channel control register B_10 | DMACHCRB_10 | R/W | H'E670_851C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_10 | DMARS_10 | R/W | H'E670_8540 | 16 | √ | √ | √ | √ |
| DMA buffer control register_10 | DMABUFCR_10 | R/W | H'E670_8548 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_10 | DMADPBASE_10 | R/W | H'E670_8550 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_10 | DMADPCR_10 | R/W | H'E670_8554 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_10 | DMAFIXSAR_10 | R/W | H'E670_8510 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_10 | DMAFIXDAR_10 | R/W | H'E670_8514 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_10 | DMAFIXDPBASE_10 | R/W | H'E670_8560 | 32 | √ | √ | √ | √ |
| DMA source address register_11 | DMASAR_11 | R/W | H'E670_8580 H'E670_85A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_11 | DMADAR_11 | R/W | H'E670_8584 H'E670_85A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_11 | DMATCR_11 | R/W | H'E670_8588 | 32 | √ | √ | √ | √ |
| DMA transfer size register_11 | DMATSR_11 | R/W | H'E670_85A8* | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA channel control register_11 | DMACHCR_11 | R/W | H'E670_858C H'E670_85AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_11 | DMATCRB_11 | R/W | H'E670_8598 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_11 | DMATSRB_11 | R/W | H'E670_85B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_11 | DMACHCRB_11 | R/W | H'E670_859C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_11 | DMARS_11 | R/W | H'E670_85C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_11 | DMABUFCR_11 | R/W | H'E670_85C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_11 | DMADPBASE_11 | R/W | H'E670_85D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_11 | DMADPCR_11 | R/W | H'E670_85D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_11 | DMAFIXSAR_11 | R/W | H'E670_8590 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_11 | DMAFIXDAR_11 | R/W | H'E670_8594 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_11 | DMAFIXDPBASE_11 | R/W | H'E670_85E0 | 32 | √ | √ | √ | √ |
| DMA source address register_12 | DMASAR_12 | R/W | H'E670_8600 H'E670_8620* | 32 | √ | √ | √ | √ |
| DMA destination address register_12 | DMADAR_12 | R/W | H'E670_8604 H'E670_8624* | 32 | √ | √ | √ | √ |
| DMA transfer count register_12 | DMATCR_12 | R/W | H'E670_8608 | 32 | √ | √ | √ | √ |
| DMA transfer size register_12 | DMATSR_12 | R/W | H'E670_8628* | 32 | √ | √ | √ | √ |
| DMA channel control register_12 | DMACHCR_12 | R/W | H'E670_860C H'E670_862C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_12 | DMATCRB_12 | R/W | H'E670_8618 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_12 | DMATSRB_12 | R/W | H'E670_8638* | 32 | √ | √ | √ | √ |
| DMA channel control register B_12 | DMACHCRB_12 | R/W | H'E670_861C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_12 | DMARS_12 | R/W | H'E670_8640 | 16 | √ | √ | √ | √ |
| DMA buffer control register_12 | DMABUFCR_12 | R/W | H'E670_8648 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_12 | DMADPBASE_12 | R/W | H'E670_8650 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_12 | DMADPCR_12 | R/W | H'E670_8654 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_12 | DMAFIXSAR_12 | R/W | H'E670_8610 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_12 | DMAFIXDAR_12 | R/W | H'E670_8614 | 32 | √ | √ | √ | √ |

| | | | | | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| Name | Abbreviation | R/W | Address | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA fixed descriptor base address register_12 | DMAFIXDPBASE_12 | R/W | H'E670_8660 | 32 | √ | √ | √ | √ |
| DMA source address register_13 | DMASAR_13 | R/W | H'E670_8680 H'E670_86A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_13 | DMADAR_13 | R/W | H'E670_8684 H'E670_86A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_13 | DMATCR_13 | R/W | H'E670_8688 | 32 | √ | √ | √ | √ |
| DMA transfer size register_13 | DMATSR_13 | R/W | H'E670_86A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_13 | DMACHCR_13 | R/W | H'E670_868C H'E670_86AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_13 | DMATCRB_13 | R/W | H'E670_8698 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_13 | DMATSRB_13 | R/W | H'E670_86B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_13 | DMACHCRB_13 | R/W | H'E670_869C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_13 | DMARS_13 | R/W | H'E670_86C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_13 | DMABUFCR_13 | R/W | H'E670_86C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_13 | DMADPBASE_13 | R/W | H'E670_86D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_13 | DMADPCR_13 | R/W | H'E670_86D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_13 | DMAFIXSAR_13 | R/W | H'E670_8690 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_13 | DMAFIXDAR_13 | R/W | H'E670_8694 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_13 | DMAFIXDPBASE_13 | R/W | H'E670_86E0 | 32 | √ | √ | √ | √ |
| DMA source address register_14 | DMASAR_14 | R/W | H'E670_8700 H'E670_8720* | 32 | √ | √ | √ | √ |
| DMA destination address register_14 | DMADAR_14 | R/W | H'E670_8704 H'E670_8724* | 32 | √ | √ | √ | √ |
| DMA transfer count register_14 | DMATCR_14 | R/W | H'E670_8708 | 32 | √ | √ | √ | √ |
| DMA transfer size register_14 | DMATSR_14 | R/W | H'E670_8728* | 32 | √ | √ | √ | √ |
| DMA channel control register_14 | DMACHCR_14 | R/W | H'E670_870C H'E670_872C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_14 | DMATCRB_14 | R/W | H'E670_8718 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_14 | DMATSRB_14 | R/W | H'E670_8738* | 32 | √ | √ | √ | √ |
| DMA channel control register B_14 | DMACHCRB_14 | R/W | H'E670_871C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_14 | DMARS_14 | R/W | H'E670_8740 | 16 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-------------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA buffer control register_14 | DMABUFCR_14 | R/W | H'E670_8748 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_14 | DMADPBASE_14 | R/W | H'E670_8750 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_14 | DMADPCR_14 | R/W | H'E670_8754 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_14 | DMAFIXSAR_14 | R/W | H'E670_8710 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_14 | DMAFIXDAR_14 | R/W | H'E670_8714 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_14 | DMAFIXDPBASE_14 | R/W | H'E670_8760 | 32 | √ | √ | √ | √ |
| DMA source address register_15 | DMASAR_15 | R/W | H'E670_8780 H'E670_87A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_15 | DMADAR_15 | R/W | H'E670_8784 H'E670_87A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_15 | DMATCR_15 | R/W | H'E670_8788 | 32 | √ | √ | √ | √ |
| DMA transfer size register_15 | DMATSR_15 | R/W | H'E670_87A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_15 | DMACHCR_15 | R/W | H'E670_878C H'E670_87AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_15 | DMATCRB_15 | R/W | H'E670_8798 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_15 | DMATSRB_15 | R/W | H'E670_87B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_15 | DMACHCRB_15 | R/W | H'E670_879C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_15 | DMARS_15 | R/W | H'E670_87C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_15 | DMABUFCR_15 | R/W | H'E670_87C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_15 | DMADPBASE_15 | R/W | H'E670_87D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_15 | DMADPCR_15 | R/W | H'E670_87D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_15 | DMAFIXSAR_15 | R/W | H'E670_8790 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_15 | DMAFIXDAR_15 | R/W | H'E670_8794 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_15 | DMAFIXDPBASE_15 | R/W | H'E670_87E0 | 32 | √ | √ | √ | √ |
| Descriptor memory (for channels 0 to 15) | DescriptorMEM | R/W | H'E670_A000 to H'E670_A7FC | 32 | √ | √ | √ | √ |
| Secure function Secure Status register (for channels 0 to 15) | DMASES_0 | R/W | H'E670_00C0 | 32 | √ | √ | √ | √ |
| Secure function Slave Error Address register (for channels 0 to 15) | DMASEA_0 | R | H'E670_00C4 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Secure function Error Master ID register (for channels 0 to 15) | DMAEMID_0 | R/W | H'E670_00C8 | 32 | √ | √ | √ | √ |
| DMA interrupt status register (for channels 16 to 31) | DMAISTA_1 | R | H'E730_0020 | 32 | √ | √ | √ | √ |
| DMA secure control register (for channels 16 to 31) | DMASEC_1 | R/W | H'E730_0030 | 32 | √ | √ | √ | √ |
| DMA operation register (for channels 16 to 31) | DMAOR_1 | R/W | H'E730_0060 | 16 | √ | √ | √ | √ |
| DMA channel clear register (for channels 16 to 31) | DMACHCLR_1 | W | H'E730_0080 | 32 | √ | √ | √ | √ |
| DPRAM secure control register (for channels 16 to 31) | DMADPSEC_1 | R/W | H'E730_00A0 | 32 | √ | √ | √ | √ |
| DMA source address register_16 | DMASAR_16 | R/W | H'E730_8000 H'E730_8020* | 32 | √ | √ | √ | √ |
| DMA destination address register_16 | DMADAR_16 | R/W | H'E730_8004 H'E730_8024* | 32 | √ | √ | √ | √ |
| DMA transfer count register_16 | DMATCR_16 | R/W | H'E730_8008 | 32 | √ | √ | √ | √ |
| DMA transfer size register_16 | DMATSR_16 | R/W | H'E730_8028* | 32 | √ | √ | √ | √ |
| DMA channel control register_16 | DMACHCR_16 | R/W | H'E730_800C H'E730_802C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_16 | DMATCRB_16 | R/W | H'E730_8018 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_16 | DMATSRB_16 | R/W | H'E730_8038 | 32 | √ | √ | √ | √ |
| DMA channel control register B_16 | DMACHCRB_16 | R/W | H'E730_801C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_16 | DMARS_16 | R/W | H'E730_8040 | 16 | √ | √ | √ | √ |
| DMA buffer control register_16 | DMABUFCR_16 | R/W | H'E730_8048 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_16 | DMADPBASE_16 | R/W | H'E730_8050 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_16 | DMADPCR_16 | R/W | H'E730_8054 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_16 | DMAFIXSAR_16 | R/W | H'E730_8010 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_16 | DMAFIXDAR_16 | R/W | H'E730_8014 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_16 | DMAFIXDPBASE_16 | R/W | H'E730_8060 | 32 | √ | √ | √ | √ |
| DMA source address register_17 | DMASAR_17 | R/W | H'E730_8080 H'E730_80A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_17 | DMADAR_17 | R/W | H'E730_8084 H'E730_80A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_17 | DMATCR_17 | R/W | H'E730_8088 | 32 | √ | √ | √ | √ |
| DMA transfer size register_17 | DMATSR_17 | R/W | H'E730_80A8* | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA channel control register_17 | DMACHCR_17 | R/W | H'E730_808C H'E73080AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_17 | DMATCRB_17 | R/W | H'E730_8098 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_17 | DMATSRB_17 | R/W | H'E730_80B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_17 | DMACHCRB_17 | R/W | H'E730_809C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_17 | DMARS_17 | R/W | H'E730_80C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_17 | DMABUFCR_17 | R/W | H'E730_80C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_17 | DMADPBASE_17 | R/W | H'E730_80D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_17 | DMADPCR_17 | R/W | H'E730_80D4 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_17 | DMAFIXDPBASE_17 | R/W | H'E730_80E0 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_17 | DMAFIXSAR_17 | R/W | H'E730_8090 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_17 | DMAFIXDAR_17 | R/W | H'E730_8094 | 32 | √ | √ | √ | √ |
| DMA source address register_18 | DMASAR_18 | R/W | H'E730_8100 H'E730_8120* | 32 | √ | √ | √ | √ |
| DMA destination address register_18 | DMADAR_18 | R/W | H'E730_8104 H'E730_8124* | 32 | √ | √ | √ | √ |
| DMA transfer count register_18 | DMATCR_18 | R/W | H'E730_8108 | 32 | √ | √ | √ | √ |
| DMA transfer size register_18 | DMATSR_18 | R/W | H'E730_8128* | 32 | √ | √ | √ | √ |
| DMA channel control register_18 | DMACHCR_18 | R/W | H'E730_810C H'E730_812C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_18 | DMATCRB_18 | R/W | H'E730_8118 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_18 | DMATSRB_18 | R/W | H'E730_8138* | 32 | √ | √ | √ | √ |
| DMA channel control register B_18 | DMACHCRB_18 | R/W | H'E730_811C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_18 | DMARS_18 | R/W | H'E730_8140 | 16 | √ | √ | √ | √ |
| DMA buffer control register_18 | DMABUFCR_18 | R/W | H'E730_8148 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_18 | DMADPBASE_18 | R/W | H'E730_8150 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_18 | DMADPCR_18 | R/W | H'E730_8154 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_18 | DMAFIXSAR_18 | R/W | H'E730_8110 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_18 | DMAFIXDAR_18 | R/W | H'E730_8114 | 32 | √ | √ | √ | √ |

| | | | | | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| Name | Abbreviation | R/W | Address | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA fixed descriptor base address register_18 | DMAFIXDPBASE_18 | R/W | H'E730_8160 | 32 | √ | √ | √ | √ |
| DMA source address register_19 | DMASAR_19 | R/W | H'E730_8180 H'E730_81A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_19 | DMADAR_19 | R/W | H'E730_8184 H'E730_81A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_19 | DMATCR_19 | R/W | H'E730_8188 | 32 | √ | √ | √ | √ |
| DMA transfer size register_19 | DMATSR_19 | R/W | H'E730_81A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_19 | DMACHCR_19 | R/W | H'E730_818C H'E730_81AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_19 | DMATCRB_19 | R/W | H'E730_8198 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_19 | DMATSRB_19 | R/W | H'E730_81B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_19 | DMACHCRB_19 | R/W | H'E730_819C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_19 | DMARS_19 | R/W | H'E730_81C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_19 | DMABUFCR_19 | R/W | H'E730_81C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_19 | DMADPBASE_19 | R/W | H'E730_81D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_19 | DMADPCR_19 | R/W | H'E730_81D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_19 | DMAFIXSAR_19 | R/W | H'E730_8190 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_19 | DMAFIXDAR_19 | R/W | H'E730_8194 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_19 | DMAFIXDPBASE_19 | R/W | H'E730_81E0 | 32 | √ | √ | √ | √ |
| DMA source address register_20 | DMASAR_20 | R/W | H'E730_8200 H'E730_8220* | 32 | √ | √ | √ | √ |
| DMA destination address register_20 | DMADAR_20 | R/W | H'E730_8204 H'E730_8224* | 32 | √ | √ | √ | √ |
| DMA transfer count register_20 | DMATCR_20 | R/W | H'E730_8208 | 32 | √ | √ | √ | √ |
| DMA transfer size register_20 | DMATSR_20 | R/W | H'E730_8228* | 32 | √ | √ | √ | √ |
| DMA channel control register_20 | DMACHCR_20 | R/W | H'E730_820C H'E730_822C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_20 | DMATCRB_20 | R/W | H'E730_8218 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_20 | DMATSRB_20 | R/W | H'E730_8238* | 32 | √ | √ | √ | √ |
| DMA channel control register B_20 | DMACHCRB_20 | R/W | H'E730_821C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_20 | DMARS_20 | R/W | H'E730_8240 | 16 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA buffer control register_20 | DMABUFCR_20 | R/W | H'E730_8248 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_20 | DMADPBASE_20 | R/W | H'E730_8250 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_20 | DMADPCR_20 | R/W | H'E730_8254 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_20 | DMAFIXSAR_20 | R/W | H'E730_8210 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_20 | DMAFIXDAR_20 | R/W | H'E730_8214 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_20 | DMAFIXDPBASE_20 | R/W | H'E730_8260 | 32 | √ | √ | √ | √ |
| DMA source address register_21 | DMASAR_21 | R/W | H'E730_8280 H'E730_82A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_21 | DMADAR_21 | R/W | H'E730_8284 H'E730_82A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_21 | DMATCR_21 | R/W | H'E730_8288 | 32 | √ | √ | √ | √ |
| DMA transfer size register_21 | DMATSR_21 | R/W | H'E730_82A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_21 | DMACHCR_21 | R/W | H'E730_828C H'E730_82AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_21 | DMATCRB_21 | R/W | H'E730_8298 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_21 | DMATSRB_21 | R/W | H'E730_82B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_21 | DMACHCRB_21 | R/W | H'E730_829C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_21 | DMARS_21 | R/W | H'E730_82C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_21 | DMABUFCR_21 | R/W | H'E730_82C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_21 | DMADPBASE_21 | R/W | H'E730_82D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_21 | DMADPCR_21 | R/W | H'E730_82D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_21 | DMAFIXSAR_21 | R/W | H'E730_8290 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_21 | DMAFIXDAR_21 | R/W | H'E730_8294 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_21 | DMAFIXDPBASE_21 | R/W | H'E730_82E0 | 32 | √ | √ | √ | √ |
| DMA source address register_22 | DMASAR_22 | R/W | H'E730_8300 H'E730_8320* | 32 | √ | √ | √ | √ |
| DMA destination address register_22 | DMADAR_22 | R/W | H'E730_8304 H'E730_8324* | 32 | √ | √ | √ | √ |
| DMA transfer count register_22 | DMATCR_22 | R/W | H'E730_8308 | 32 | √ | √ | √ | √ |
| DMA transfer size register_22 | DMATSR_22 | R/W | H'E730_8328* | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA channel control register_22 | DMACHCR_22 | R/W | H'E730_830C H'E730_832C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_22 | DMATCRB_22 | R/W | H'E730_8318 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_22 | DMATSRB_22 | R/W | H'E730_8338* | 32 | √ | √ | √ | √ |
| DMA channel control register B_22 | DMACHCRB_22 | R/W | H'E730_831C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_22 | DMARS_22 | R/W | H'E730_8340 | 16 | √ | √ | √ | √ |
| DMA buffer control register_22 | DMABUFCR_22 | R/W | H'E730_8348 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_22 | DMADPBASE_22 | R/W | H'E730_8350 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_22 | DMADPCR_22 | R/W | H'E730_8354 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_22 | DMAFIXSAR_22 | R/W | H'E730_8310 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_22 | DMAFIXDAR_22 | R/W | H'E730_8314 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_22 | DMAFIXDPBASE_22 | R/W | H'E730_8360 | 32 | √ | √ | √ | √ |
| DMA source address register_23 | DMASAR_23 | R/W | H'E730_8380 H'E730_83A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_23 | DMADAR_23 | R/W | H'E730_8384 H'E730_83A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_23 | DMATCR_23 | R/W | H'E730_8388 | 32 | √ | √ | √ | √ |
| DMA transfer size register_23 | DMATSR_23 | R/W | H'E730_83A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_23 | DMACHCR_23 | R/W | H'E730_838C H'E730_83AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_23 | DMATCRB_23 | R/W | H'E730_8398 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_23 | DMATSRB_23 | R/W | H'E730_83B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_23 | DMACHCRB_23 | R/W | H'E730_839C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_23 | DMARS_23 | R/W | H'E730_83C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_23 | DMABUFCR_23 | R/W | H'E730_83C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_23 | DMADPBASE_23 | R/W | H'E730_83D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_23 | DMADPCR_23 | R/W | H'E730_83D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_23 | DMAFIXSAR_23 | R/W | H'E730_8390 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_23 | DMAFIXDAR_23 | R/W | H'E730_8394 | 32 | √ | √ | √ | √ |

| | | | | | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| Name | Abbreviation | R/W | Address | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA fixed descriptor base address register_23 | DMAFIXDPBASE_23 | R/W | H'E730_83E0 | 32 | √ | √ | √ | √ |
| DMA source address register_24 | DMASAR_24 | R/W | H'E730_8400 H'E730_8420* | 32 | √ | √ | √ | √ |
| DMA destination address register_24 | DMADAR_24 | R/W | H'E730_8404 H'E730_8424* | 32 | √ | √ | √ | √ |
| DMA transfer count register_24 | DMATCR_24 | R/W | H'E730_8408 | 32 | √ | √ | √ | √ |
| DMA transfer size register_24 | DMATSR_24 | R/W | H'E730_8428* | 32 | √ | √ | √ | √ |
| DMA channel control register_24 | DMACHCR_24 | R/W | H'E730_840C H'E730_842C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_24 | DMATCRB_24 | R/W | H'E730_8418 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_24 | DMATSRB_24 | R/W | H'E730_8438* | 32 | √ | √ | √ | √ |
| DMA channel control register B_24 | DMACHCRB_24 | R/W | H'E730_841C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_24 | DMARS_24 | R/W | H'E730_8440 | 16 | √ | √ | √ | √ |
| DMA buffer control register_24 | DMABUFCR_24 | R/W | H'E730_8448 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_24 | DMADPBASE_24 | R/W | H'E730_8450 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_24 | DMADPCR_24 | R/W | H'E730_8454 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_24 | DMAFIXSAR_24 | R/W | H'E730_8410 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_24 | DMAFIXDAR_24 | R/W | H'E730_8414 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_24 | DMAFIXDPBASE_24 | R/W | H'E730_8460 | 32 | √ | √ | √ | √ |
| DMA source address register_25 | DMASAR_25 | R/W | H'E730_8480 H'E730_84A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_25 | DMADAR_25 | R/W | H'E730_8484 H'E730_84A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_25 | DMATCR_25 | R/W | H'E730_8488 | 32 | √ | √ | √ | √ |
| DMA transfer size register_25 | DMATSR_25 | R/W | H'E730_84A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_25 | DMACHCR_25 | R/W | H'E730_848C H'E730_84AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_25 | DMATCRB_25 | R/W | H'E730_8498 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_25 | DMATSRB_25 | R/W | H'E730_84B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_25 | DMACHCRB_25 | R/W | H'E730_849C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_25 | DMARS_25 | R/W | H'E730_84C0 | 16 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA buffer control register_25 | DMABUFCR_25 | R/W | H'E730_84C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_25 | DMADPBASE_25 | R/W | H'E730_84D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_25 | DMADPCR_25 | R/W | H'E730_84D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_25 | DMAFIXSAR_25 | R/W | H'E730_8490 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_25 | DMAFIXDAR_25 | R/W | H'E730_8494 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_25 | DMAFIXDPBASE_25 | R/W | H'E730_84E0 | 32 | √ | √ | √ | √ |
| DMA source address register_26 | DMASAR_26 | R/W | H'E730_8500 H'E730_8520* | 32 | √ | √ | √ | √ |
| DMA destination address register_26 | DMADAR_26 | R/W | H'E730_8504 H'E730_8524* | 32 | √ | √ | √ | √ |
| DMA transfer count register_26 | DMATCR_26 | R/W | H'E730_8508 | 32 | √ | √ | √ | √ |
| DMA transfer size register_26 | DMATSR_26 | R/W | H'E730_8528* | 32 | √ | √ | √ | √ |
| DMA channel control register_26 | DMACHCR_26 | R/W | H'E730_850C H'E730_852C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_26 | DMATCRB_26 | R/W | H'E730_8518 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_26 | DMATSRB_26 | R/W | H'E730_8538* | 32 | √ | √ | √ | √ |
| DMA channel control register B_26 | DMACHCRB_26 | R/W | H'E730_851C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_26 | DMARS_26 | R/W | H'E730_8540 | 16 | √ | √ | √ | √ |
| DMA buffer control register_26 | DMABUFCR_26 | R/W | H'E730_8548 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_26 | DMADPBASE_26 | R/W | H'E730_8550 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_26 | DMADPCR_26 | R/W | H'E730_8554 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_26 | DMAFIXSAR_26 | R/W | H'E730_8510 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_26 | DMAFIXDAR_26 | R/W | H'E730_8514 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_26 | DMAFIXDPBASE_26 | R/W | H'E730_8560 | 32 | √ | √ | √ | √ |
| DMA source address register_27 | DMASAR_27 | R/W | H'E730_8580 H'E730_85A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_27 | DMADAR_27 | R/W | H'E730_8584 H'E730_85A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_27 | DMATCR_27 | R/W | H'E730_8588 | 32 | √ | √ | √ | √ |
| DMA transfer size register_27 | DMATSR_27 | R/W | H'E730_85A8* | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA channel control register_27 | DMACHCR_27 | R/W | H'E730_858C H'E730_85AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_27 | DMATCRB_27 | R/W | H'E730_8598 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_27 | DMATSRB_27 | R/W | H'E730_85B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_27 | DMACHCRB_27 | R/W | H'E730_859C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_27 | DMARS_27 | R/W | H'E730_85C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_27 | DMABUFCR_27 | R/W | H'E730_85C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_27 | DMADPBASE_27 | R/W | H'E730_85D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_27 | DMADPCR_27 | R/W | H'E730_85D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_27 | DMAFIXSAR_27 | R/W | H'E730_8590 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_27 | DMAFIXDAR_27 | R/W | H'E730_8594 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_27 | DMAFIXDPBASE_27 | R/W | H'E730_85E0 | 32 | √ | √ | √ | √ |
| DMA source address register_28 | DMASAR_28 | R/W | H'E730_8600 H'E730_8620* | 32 | √ | √ | √ | √ |
| DMA destination address register_28 | DMADAR_28 | R/W | H'E730_8604 H'E730_8624* | 32 | √ | √ | √ | √ |
| DMA transfer count register_28 | DMATCR_28 | R/W | H'E730_8608 | 32 | √ | √ | √ | √ |
| DMA transfer size register_28 | DMATSR_28 | R/W | H'E730_8628* | 32 | √ | √ | √ | √ |
| DMA channel control register_28 | DMACHCR_28 | R/W | H'E730_860C H'E730_862C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_28 | DMATCRB_28 | R/W | H'E730_8618 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_28 | DMATSRB_28 | R/W | H'E730_8638* | 32 | √ | √ | √ | √ |
| DMA channel control register B_28 | DMACHCRB_28 | R/W | H'E730_861C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_28 | DMARS_28 | R/W | H'E730_8640 | 16 | √ | √ | √ | √ |
| DMA buffer control register_28 | DMABUFCR_28 | R/W | H'E730_8648 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_28 | DMADPBASE_28 | R/W | H'E730_8650 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_28 | DMADPCR_28 | R/W | H'E730_8654 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_28 | DMAFIXSAR_28 | R/W | H'E730_8610 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_28 | DMAFIXDAR_28 | R/W | H'E730_8614 | 32 | √ | √ | √ | √ |

| | | | | | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| Name | Abbreviation | R/W | Address | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA fixed descriptor base address register_28 | DMAFIXDPBASE_28 | R/W | H'E730_8660 | 32 | √ | √ | √ | √ |
| DMA source address register_29 | DMASAR_29 | R/W | H'E730_8680 H'E730_86A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_29 | DMADAR_29 | R/W | H'E730_8684 H'E730_86A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_29 | DMATCR_29 | R/W | H'E730_8688 | 32 | √ | √ | √ | √ |
| DMA transfer size register_29 | DMATSR_29 | R/W | H'E730_86A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_29 | DMACHCR_29 | R/W | H'E730_868C H'E730_86AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_29 | DMATCRB_29 | R/W | H'E730_8698 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_29 | DMATSRB_29 | R/W | H'E730_86B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_29 | DMACHCRB_29 | R/W | H'E730_869C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_29 | DMARS_29 | R/W | H'E730_86C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_29 | DMABUFCR_29 | R/W | H'E730_86C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_29 | DMADPBASE_29 | R/W | H'E730_86D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_29 | DMADPCR_29 | R/W | H'E730_86D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_29 | DMAFIXSAR_29 | R/W | H'E730_8690 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_29 | DMAFIXDAR_29 | R/W | H'E730_8694 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_29 | DMAFIXDPBASE_29 | R/W | H'E730_86E0 | 32 | √ | √ | √ | √ |
| DMA source address register_30 | DMASAR_30 | R/W | H'E730_8700 H'E730_8720* | 32 | √ | √ | √ | √ |
| DMA destination address register_30 | DMADAR_30 | R/W | H'E730_8704 H'E730_8724* | 32 | √ | √ | √ | √ |
| DMA transfer count register_30 | DMATCR_30 | R/W | H'E730_8708 | 32 | √ | √ | √ | √ |
| DMA transfer size register_30 | DMATSR_30 | R/W | H'E730_8728* | 32 | √ | √ | √ | √ |
| DMA channel control register_30 | DMACHCR_30 | R/W | H'E730_870C H'E730_872C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_30 | DMATCRB_30 | R/W | H'E730_8718 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_30 | DMATSRB_30 | R/W | H'E730_8738* | 32 | √ | √ | √ | √ |
| DMA channel control register B_30 | DMACHCRB_30 | R/W | H'E730_871C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_30 | DMARS_30 | R/W | H'E730_8740 | 16 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--|-----------------|-----|-------------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA buffer control register_30 | DMABUFCR_30 | R/W | H'E730_8748 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_30 | DMADPBASE_30 | R/W | H'E730_8750 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_30 | DMADPCR_30 | R/W | H'E730_8754 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_30 | DMAFIXSAR_30 | R/W | H'E730_8710 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_30 | DMAFIXDAR_30 | R/W | H'E730_8714 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_30 | DMAFIXDPBASE_30 | R/W | H'E730_8760 | 32 | √ | √ | √ | √ |
| DMA source address register_31 | DMASAR_31 | R/W | H'E730_8780 H'E730_87A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_31 | DMADAR_31 | R/W | H'E730_8784 H'E730_87A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_31 | DMATCR_31 | R/W | H'E730_8788 | 32 | √ | √ | √ | √ |
| DMA transfer size register_31 | DMATSR_31 | R/W | H'E730_87A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_31 | DMACHCR_31 | R/W | H'E730_878C H'E730_87AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_31 | DMATCRB_31 | R/W | H'E730_8798 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_31 | DMATSRB_31 | R/W | H'E730_87B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_31 | DMACHCRB_31 | R/W | H'E730_879C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_31 | DMARS_31 | R/W | H'E730_87C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_31 | DMABUFCR_31 | R/W | H'E730_87C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_31 | DMADPBASE_31 | R/W | H'E730_87D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_31 | DMADPCR_31 | R/W | H'E730_87D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_31 | DMAFIXSAR_31 | R/W | H'E730_8790 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_31 | DMAFIXDAR_31 | R/W | H'E730_8794 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_31 | DMAFIXDPBASE_31 | R/W | H'E730_87E0 | 32 | √ | √ | √ | √ |
| Descriptor memory (for channels 16 to 31) | DescriptorMEM | R/W | H'E730 A000 to H'E730 A7FC | 32 | √ | √ | √ | √ |
| Secure function Secure Status register (for channels 16 to 31) | DMASES_1 | R/W | H' E730_00C0 | 32 | √ | √ | √ | √ |
| Secure function Slave Error Address register (for channels 16 to 31) | DMASEA_1 | R | H' E730_00C4 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Secure function Error Master ID register (for channels 16 to 31) | DMAEMID_1 | R/W | H'E730_00C8 | 32 | √ | √ | √ | √ |
| DMA interrupt status register (for channels 32 to 47) | DMAISTA_2 | R | H'E731_0020 | 32 | √ | √ | √ | √ |
| DMA secure control register (for channels 32 to 47) | DMASEC_2 | R/W | H'E731_0030 | 32 | √ | √ | √ | √ |
| DMA operation register (for channels 32 to 47) | DMAOR_2 | R/W | H'E731_0060 | 16 | √ | √ | √ | √ |
| DMA channel clear register (for channels 32 to 47) | DMACHCLR_2 | W | H'E731_0080 | 32 | √ | √ | √ | √ |
| DPRAM secure control register (for channels 32 to 47) | DMADPSEC_2 | R/W | H'E731_00A0 | 32 | √ | √ | √ | √ |
| DMA source address register_32 | DMASAR_32 | R/W | H'E731_8000 H'E731_8020* | 32 | √ | √ | √ | √ |
| DMA destination address register_32 | DMADAR_32 | R/W | H'E731_8004 H'E731_8024* | 32 | √ | √ | √ | √ |
| DMA transfer count register_32 | DMATCR_32 | R/W | H'E731_8008 | 32 | √ | √ | √ | √ |
| DMA transfer size register_32 | DMATSR_32 | R/W | H'E731_8028* | 32 | √ | √ | √ | √ |
| DMA channel control register_32 | DMACHCR_32 | R/W | H'E731_800C H'E731_802C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_32 | DMATCRB_32 | R/W | H'E731_8018 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_32 | DMATSRB_32 | R/W | H'E731_8038* | 32 | √ | √ | √ | √ |
| DMA channel control register B_32 | DMACHCRB_32 | R/W | H'E731_801C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_32 | DMARS_32 | R/W | H'E731_8040 | 16 | √ | √ | √ | √ |
| DMA buffer control register_32 | DMABUFCR_32 | R/W | H'E731_8048 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_32 | DMADPBASE_32 | R/W | H'E731_8050 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_32 | DMADPCR_32 | R/W | H'E731_8054 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_32 | DMAFIXSAR_32 | R/W | H'E731_8010 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_32 | DMAFIXDAR_32 | R/W | H'E731_8014 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_32 | DMAFIXDPBASE_32 | R/W | H'E731_8060 | 32 | √ | √ | √ | √ |
| DMA source address register_33 | DMASAR_33 | R/W | H'E731_8080 H'E731_80A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_33 | DMADAR_33 | R/W | H'E731_8084 H'E731_80A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_33 | DMATCR_33 | R/W | H'E731_8088 | 32 | √ | √ | √ | √ |
| DMA transfer size register_33 | DMATSR_33 | R/W | H'E731_80A8* | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA channel control register_33 | DMACHCR_33 | R/W | H'E731_808C H'E731_80AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_33 | DMATCRB_33 | R/W | H'E731_8098 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_33 | DMATSRB_33 | R/W | H'E731_80B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_33 | DMACHCRB_33 | R/W | H'E731_809C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_33 | DMARS_33 | R/W | H'E731_80C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_33 | DMABUFCR_33 | R/W | H'E731_80C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_33 | DMADPBASE_33 | R/W | H'E731_80D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_33 | DMADPCR_33 | R/W | H'E731_80D4 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_33 | DMAFIXDPBASE_33 | R/W | H'E731_80E0 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_33 | DMAFIXSAR_33 | R/W | H'E731_8090 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_33 | DMAFIXDAR_33 | R/W | H'E731_8094 | 32 | √ | √ | √ | √ |
| DMA source address register_34 | DMASAR_34 | R/W | H'E731_8100 H'E731_8120* | 32 | √ | √ | √ | √ |
| DMA destination address register_34 | DMADAR_34 | R/W | H'E731_8104 H'E731_8124* | 32 | √ | √ | √ | √ |
| DMA transfer count register_34 | DMATCR_34 | R/W | H'E731_8108 | 32 | √ | √ | √ | √ |
| DMA transfer size register_34 | DMATSR_34 | R/W | H'E731_8128* | 32 | √ | √ | √ | √ |
| DMA channel control register_34 | DMACHCR_34 | R/W | H'E731_810C H'E731_812C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_34 | DMATCRB_34 | R/W | H'E731_8118 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_34 | DMATSRB_34 | R/W | H'E731_8138* | 32 | √ | √ | √ | √ |
| DMA channel control register B_34 | DMACHCRB_34 | R/W | H'E731_811C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_34 | DMARS_34 | R/W | H'E731_8140 | 16 | √ | √ | √ | √ |
| DMA buffer control register_34 | DMABUFCR_34 | R/W | H'E731_8148 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_34 | DMADPBASE_34 | R/W | H'E731_8150 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_34 | DMADPCR_34 | R/W | H'E731_8154 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_34 | DMAFIXSAR_34 | R/W | H'E731_8110 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_34 | DMAFIXDAR_34 | R/W | H'E731_8114 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA fixed descriptor base address register_34 | DMAFIXDPBASE_34 | R/W | H'E731_8160 | 32 | √ | √ | √ | √ |
| DMA source address register_35 | DMASAR_35 | R/W | H'E731_8180 H'E731_81A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_35 | DMADAR_35 | R/W | H'E731_8184 H'E731_81A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_35 | DMATCR_35 | R/W | H'E731_8188 | 32 | √ | √ | √ | √ |
| DMA transfer size register_35 | DMATSR_35 | R/W | H'E731_81A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_35 | DMACHCR_35 | R/W | H'E731_818C H'E731_81AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_35 | DMATCRB_35 | R/W | H'E731_8198 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_35 | DMATSRB_35 | R/W | H'E731_81B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_35 | DMACHCRB_35 | R/W | H'E731_819C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_35 | DMARS_35 | R/W | H'E731_81C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_35 | DMABUFCR_35 | R/W | H'E731_81C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_35 | DMADPBASE_35 | R/W | H'E731_81D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_35 | DMADPCR_35 | R/W | H'E731_81D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_35 | DMAFIXSAR_35 | R/W | H'E731_8190 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_35 | DMAFIXDAR_35 | R/W | H'E731_8194 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_35 | DMAFIXDPBASE_35 | R/W | H'E731_81E0 | 32 | √ | √ | √ | √ |
| DMA source address register_36 | DMASAR_36 | R/W | H'E731_8200 H'E731_8220* | 32 | √ | √ | √ | √ |
| DMA destination address register_36 | DMADAR_36 | R/W | H'E731_8204 H'E731_8224* | 32 | √ | √ | √ | √ |
| DMA transfer count register_36 | DMATCR_36 | R/W | H'E731_8208 | 32 | √ | √ | √ | √ |
| DMA transfer size register_36 | DMATSR_36 | R/W | H'E731_8228* | 32 | √ | √ | √ | √ |
| DMA channel control register_36 | DMACHCR_36 | R/W | H'E731_820C H'E731_822C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_36 | DMATCRB_36 | R/W | H'E731_8218 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_36 | DMATSRB_36 | R/W | H'E731_8238* | 32 | √ | √ | √ | √ |
| DMA channel control register B_36 | DMACHCRB_36 | R/W | H'E731_821C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_36 | DMARS_36 | R/W | H'E731_8240 | 16 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA buffer control register_36 | DMABUFCR_36 | R/W | H'E731_8248 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_36 | DMADPBASE_36 | R/W | H'E731_8250 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_36 | DMADPCR_36 | R/W | H'E731_8254 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_36 | DMAFIXSAR_36 | R/W | H'E731_8210 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_36 | DMAFIXDAR_36 | R/W | H'E731_8214 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_36 | DMAFIXDPBASE_36 | R/W | H'E731_8260 | 32 | √ | √ | √ | √ |
| DMA source address register_37 | DMASAR_37 | R/W | H'E731_8280 H'E731_82A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_37 | DMADAR_37 | R/W | H'E731_8284 H'E731_82A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_37 | DMATCR_37 | R/W | H'E731_8288 | 32 | √ | √ | √ | √ |
| DMA transfer size register_37 | DMATSR_37 | R/W | H'E731_82A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_37 | DMACHCR_37 | R/W | H'E731_828C H'E731_82AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_37 | DMATCRB_37 | R/W | H'E731_8298 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_37 | DMATSRB_37 | R/W | H'E731_82B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_37 | DMACHCRB_37 | R/W | H'E731_829C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_37 | DMARS_37 | R/W | H'E731_82C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_37 | DMABUFCR_37 | R/W | H'E731_82C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_37 | DMADPBASE_37 | R/W | H'E731_82D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_37 | DMADPCR_37 | R/W | H'E731_82D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_37 | DMAFIXSAR_37 | R/W | H'E731_8290 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_37 | DMAFIXDAR_37 | R/W | H'E731_8294 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_37 | DMAFIXDPBASE_37 | R/W | H'E731_82E0 | 32 | √ | √ | √ | √ |
| DMA source address register_38 | DMASAR_38 | R/W | H'E731_8300 H'E731_8320* | 32 | √ | √ | √ | √ |
| DMA destination address register_38 | DMADAR_38 | R/W | H'E731_8304 H'E731_8324* | 32 | √ | √ | √ | √ |
| DMA transfer count register_38 | DMATCR_38 | R/W | H'E731_8308 | 32 | √ | √ | √ | √ |
| DMA transfer size register_38 | DMATSR_38 | R/W | H'E731_8328* | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA channel control register_38 | DMACHCR_38 | R/W | H'E731_830C H'E731_832C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_38 | DMATCRB_38 | R/W | H'E731_8318 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_38 | DMATSRB_38 | R/W | H'E731_8338* | 32 | √ | √ | √ | √ |
| DMA channel control register B_38 | DMACHCRB_38 | R/W | H'E731_831C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_38 | DMARS_38 | R/W | H'E731_8340 | 16 | √ | √ | √ | √ |
| DMA buffer control register_38 | DMABUFCR_38 | R/W | H'E731_8348 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_38 | DMADPBASE_38 | R/W | H'E731_8350 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_38 | DMADPCR_38 | R/W | H'E731_8354 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_38 | DMAFIXSAR_38 | R/W | H'E731_8310 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_38 | DMAFIXDAR_38 | R/W | H'E731_8314 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_38 | DMAFIXDPBASE_38 | R/W | H'E731_8360 | 32 | √ | √ | √ | √ |
| DMA source address register_39 | DMASAR_39 | R/W | H'E731_8380 H'E731_83A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_39 | DMADAR_39 | R/W | H'E731_8384 H'E731_83A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_39 | DMATCR_39 | R/W | H'E731_8388 | 32 | √ | √ | √ | √ |
| DMA transfer size register_39 | DMATSR_39 | R/W | H'E731_83A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_39 | DMACHCR_39 | R/W | H'E731_838C H'E731_83AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_39 | DMATCRB_39 | R/W | H'E731_8398 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_39 | DMATSRB_39 | R/W | H'E731_83B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_39 | DMACHCRB_39 | R/W | H'E731_839C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_39 | DMARS_39 | R/W | H'E731_83C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_39 | DMABUFCR_39 | R/W | H'E731_83C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_39 | DMADPBASE_39 | R/W | H'E731_83D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_39 | DMADPCR_39 | R/W | H'E731_83D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_39 | DMAFIXSAR_39 | R/W | H'E731_8390 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_39 | DMAFIXDAR_39 | R/W | H'E731_8394 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA fixed descriptor base address register_39 | DMAFIXDPBASE_39 | R/W | H'E731_83E0 | 32 | √ | √ | √ | √ |
| DMA source address register_40 | DMASAR_40 | R/W | H'E731_8400 H'E731_8420* | 32 | √ | √ | √ | √ |
| DMA destination address register_40 | DMADAR_40 | R/W | H'E731_8404 H'E731_8424* | 32 | √ | √ | √ | √ |
| DMA transfer count register_40 | DMATCR_40 | R/W | H'E731_8408 | 32 | √ | √ | √ | √ |
| DMA transfer size register_40 | DMATSR_40 | R/W | H'E731_8428* | 32 | √ | √ | √ | √ |
| DMA channel control register_40 | DMACHCR_40 | R/W | H'E731_840C H'E731_842C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_40 | DMATCRB_40 | R/W | H'E731_8418 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_40 | DMATSRB_40 | R/W | H'E731_8438* | 32 | √ | √ | √ | √ |
| DMA channel control register B_40 | DMACHCRB_40 | R/W | H'E731_841C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_40 | DMARS_40 | R/W | H'E731_8440 | 16 | √ | √ | √ | √ |
| DMA buffer control register_40 | DMABUFCR_40 | R/W | H'E731_8448 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_40 | DMADPBASE_40 | R/W | H'E731_8450 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_40 | DMADPCR_40 | R/W | H'E731_8454 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_40 | DMAFIXSAR_40 | R/W | H'E731_8410 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_40 | DMAFIXDAR_40 | R/W | H'E731_8414 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_40 | DMAFIXDPBASE_40 | R/W | H'E731_8460 | 32 | √ | √ | √ | √ |
| DMA source address register_41 | DMASAR_41 | R/W | H'E731_8480 H'E731_84A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_41 | DMADAR_41 | R/W | H'E731_8484 H'E731_84A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_41 | DMATCR_41 | R/W | H'E731_8488 | 32 | √ | √ | √ | √ |
| DMA transfer size register_41 | DMATSR_41 | R/W | H'E731_84A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_41 | DMACHCR_41 | R/W | H'E731_848C H'E731_84AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_41 | DMATCRB_41 | R/W | H'E731_8498 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_41 | DMATSRB_41 | R/W | H'E731_84B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_41 | DMACHCRB_41 | R/W | H'E731_849C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_41 | DMARS_41 | R/W | H'E731_84C0 | 16 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA buffer control register_41 | DMABUFCR_41 | R/W | H'E731_84C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_41 | DMADPBASE_41 | R/W | H'E731_84D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_41 | DMADPCR_41 | R/W | H'E731_84D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_41 | DMAFIXSAR_41 | R/W | H'E731_8490 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_41 | DMAFIXDAR_41 | R/W | H'E731_8494 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_41 | DMAFIXDPBASE_41 | R/W | H'E731_84E0 | 32 | √ | √ | √ | √ |
| DMA source address register_42 | DMASAR_42 | R/W | H'E731_8500 H'E731_8520* | 32 | √ | √ | √ | √ |
| DMA destination address register_42 | DMADAR_42 | R/W | H'E731_8504 H'E731_8524* | 32 | √ | √ | √ | √ |
| DMA transfer count register_42 | DMATCR_42 | R/W | H'E731_8508 | 32 | √ | √ | √ | √ |
| DMA transfer size register_42 | DMATSR_42 | R/W | H'E731_8528* | 32 | √ | √ | √ | √ |
| DMA channel control register_42 | DMACHCR_42 | R/W | H'E731_850C H'E731_852C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_42 | DMATCRB_42 | R/W | H'E731_8518 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_42 | DMATSRB_42 | R/W | H'E731_8538* | 32 | √ | √ | √ | √ |
| DMA channel control register B_42 | DMACHCRB_42 | R/W | H'E731_851C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_42 | DMARS_42 | R/W | H'E731_8540 | 16 | √ | √ | √ | √ |
| DMA buffer control register_42 | DMABUFCR_42 | R/W | H'E731_8548 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_42 | DMADPBASE_42 | R/W | H'E731_8550 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_42 | DMADPCR_42 | R/W | H'E731_8554 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_42 | DMAFIXSAR_42 | R/W | H'E731_8510 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_42 | DMAFIXDAR_42 | R/W | H'E731_8514 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_42 | DMAFIXDPBASE_42 | R/W | H'E731_8560 | 32 | √ | √ | √ | √ |
| DMA source address register_43 | DMASAR_43 | R/W | H'E731_8580 H'E731_85A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_43 | DMADAR_43 | R/W | H'E731_8584 H'E731_85A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_43 | DMATCR_43 | R/W | H'E731_8588 | 32 | √ | √ | √ | √ |
| DMA transfer size register_43 | DMATSR_43 | R/W | H'E731_85A8* | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA channel control register_43 | DMACHCR_43 | R/W | H'E731_858C H'E731_85AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_43 | DMATCRB_43 | R/W | H'E731_8598 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_43 | DMATSRB_43 | R/W | H'E731_85B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_43 | DMACHCRB_43 | R/W | H'E731_859C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_43 | DMARS_43 | R/W | H'E731_85C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_43 | DMABUFCR_43 | R/W | H'E731_85C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_43 | DMADPBASE_43 | R/W | H'E731_85D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_43 | DMADPCR_43 | R/W | H'E731_85D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_43 | DMAFIXSAR_43 | R/W | H'E731_8590 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_43 | DMAFIXDAR_43 | R/W | H'E731_8594 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_43 | DMAFIXDPBASE_43 | R/W | H'E731_85E0 | 32 | √ | √ | √ | √ |
| DMA source address register_44 | DMASAR_44 | R/W | H'E731_8600 H'E731_8620* | 32 | √ | √ | √ | √ |
| DMA destination address register_44 | DMADAR_44 | R/W | H'E731_8604 H'E731_8624* | 32 | √ | √ | √ | √ |
| DMA transfer count register_44 | DMATCR_44 | R/W | H'E731_8608 | 32 | √ | √ | √ | √ |
| DMA transfer size register_44 | DMATSR_44 | R/W | H'E731_8628* | 32 | √ | √ | √ | √ |
| DMA channel control register_44 | DMACHCR_44 | R/W | H'E731_860C H'E731_862C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_44 | DMATCRB_44 | R/W | H'E731_8618 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_44 | DMATSRB_44 | R/W | H'E731_8638* | 32 | √ | √ | √ | √ |
| DMA channel control register B_44 | DMACHCRB_44 | R/W | H'E731_861C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_44 | DMARS_44 | R/W | H'E731_8640 | 16 | √ | √ | √ | √ |
| DMA buffer control register_44 | DMABUFCR_44 | R/W | H'E731_8648 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_44 | DMADPBASE_44 | R/W | H'E731_8650 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_44 | DMADPCR_44 | R/W | H'E731_8654 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_44 | DMAFIXSAR_44 | R/W | H'E731_8610 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_44 | DMAFIXDAR_44 | R/W | H'E731_8614 | 32 | √ | √ | √ | √ |

| | | | | | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| Name | Abbreviation | R/W | Address | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA fixed descriptor base address register_44 | DMAFIXDPBASE_44 | R/W | H'E731_8660 | 32 | √ | √ | √ | √ |
| DMA source address register_45 | DMASAR_45 | R/W | H'E731_8680 H'E731_86A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_45 | DMADAR_45 | R/W | H'E731_8684 H'E731_86A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_45 | DMATCR_45 | R/W | H'E731_8688 | 32 | √ | √ | √ | √ |
| DMA transfer size register_45 | DMATSR_45 | R/W | H'E731_86A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_45 | DMACHCR_45 | R/W | H'E731_868C H'E731_86AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_45 | DMATCRB_45 | R/W | H'E731_8698 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_45 | DMATSRB_45 | R/W | H'E731_86B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_45 | DMACHCRB_45 | R/W | H'E731_869C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_45 | DMARS_45 | R/W | H'E731_86C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_45 | DMABUFCR_45 | R/W | H'E731_86C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_45 | DMADPBASE_45 | R/W | H'E731_86D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_45 | DMADPCR_45 | R/W | H'E731_86D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_45 | DMAFIXSAR_45 | R/W | H'E731_8690 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_45 | DMAFIXDAR_45 | R/W | H'E731_8694 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_45 | DMAFIXDPBASE_45 | R/W | H'E731_86E0 | 32 | √ | √ | √ | √ |
| DMA source address register_46 | DMASAR_46 | R/W | H'E731_8700 H'E731_8720* | 32 | √ | √ | √ | √ |
| DMA destination address register_46 | DMADAR_46 | R/W | H'E731_8704 H'E731_8724* | 32 | √ | √ | √ | √ |
| DMA transfer count register_46 | DMATCR_46 | R/W | H'E731_8708 | 32 | √ | √ | √ | √ |
| DMA transfer size register_46 | DMATSR_46 | R/W | H'E731_8728* | 32 | √ | √ | √ | √ |
| DMA channel control register_46 | DMACHCR_46 | R/W | H'E731_870C H'E731_872C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_46 | DMATCRB_46 | R/W | H'E731_8718 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_46 | DMATSRB_46 | R/W | H'E731_8738* | 32 | √ | √ | √ | √ |
| DMA channel control register B_46 | DMACHCRB_46 | R/W | H'E731_871C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_46 | DMARS_46 | R/W | H'E731_8740 | 16 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--|-----------------|-----|-------------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA buffer control register_46 | DMABUFCR_46 | R/W | H'E731_8748 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_46 | DMADPBASE_46 | R/W | H'E731_8750 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_46 | DMADPCR_46 | R/W | H'E731_8754 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_46 | DMAFIXSAR_46 | R/W | H'E731_8710 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_46 | DMAFIXDAR_46 | R/W | H'E731_8714 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_46 | DMAFIXDPBASE_46 | R/W | H'E731_8760 | 32 | √ | √ | √ | √ |
| DMA source address register_47 | DMASAR_47 | R/W | H'E731_8780 H'E731_87A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_47 | DMADAR_47 | R/W | H'E731_8784 H'E731_87A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_47 | DMATCR_47 | R/W | H'E731_8788 | 32 | √ | √ | √ | √ |
| DMA transfer size register_47 | DMATSR_47 | R/W | H'E731_87A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_47 | DMACHCR_47 | R/W | H'E731_878C H'E731_87AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_47 | DMATCRB_47 | R/W | H'E731_8798 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_47 | DMATSRB_47 | R/W | H'E731_87B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_47 | DMACHCRB_47 | R/W | H'E731_879C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_47 | DMARS_47 | R/W | H'E731_87C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_47 | DMABUFCR_47 | R/W | H'E731_87C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_47 | DMADPBASE_47 | R/W | H'E731_87D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_47 | DMADPCR_47 | R/W | H'E731_87D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_47 | DMAFIXSAR_47 | R/W | H'E731_8790 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_47 | DMAFIXDAR_47 | R/W | H'E731_8794 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_47 | DMAFIXDPBASE_47 | R/W | H'E731_87E0 | 32 | √ | √ | √ | √ |
| Descriptor memory (for channels 32 to 47) | DescriptorMEM | R/W | H'E731 A000 to H'E731 A7FC | 32 | √ | √ | √ | √ |
| Secure function Secure Status register (for channels 32 to 47) | DMASES_2 | R/W | H'E731_00C0 | 32 | √ | √ | √ | √ |
| Secure function Slave Error Address register (for channels 32 to 47) | DMASEA_2 | R | H'E731_00C4 | 32 | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--|--------------|-----|--------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Secure function Error Master ID register (for channels 32 to 47) | DMAEMID_2 | R/W | H' E731_00C8 | 32 | √ | √ | √ | √ |

Notes: The base address of registers for the channels of SYS-DMAC0 (0 to 15) is H'E670_0000.

The base address of registers for the channels of SYS-DMAC1 (16 to 31) is H'E730_0000.

The base address of registers for the channels of SYS-DMAC2 (32 to 47) is H'E731_0000.

* This address is used in total size transmission (see section 23.3.6, Total Size Transmission).

Table 23.2 States of SYS-DMAC Registers in each Operating Mode

| Abbreviation | Power-On Reset | Module Standby |
|-----------------------------------|----------------|----------------|
| DMAISTA_0/DMAISTA_1/DMAISTA_2 | Initialized | Retained |
| DMASEC_0/DMASEC_1/DMASEC_2 | Initialized | Retained |
| DMAOR_0/DMAOR_1/DMAOR_2 | Initialized | Retained |
| DMACHCLR_0/DMACHCLR_1/DMACHCLR_2 | Initialized | Retained |
| DMADPSEC_0/DMADPSEC_1/DMADPSEC_2 | Initialized | Retained |
| DMASAR_0 to DMASAR_47 | Initialized | Retained |
| DMADAR_0 to DMADAR_47 | Initialized | Retained |
| DMATCR_0 to DMATCR_47 | Initialized | Retained |
| DMATSR_0 to DMATSR_47 | Initialized | Retained |
| DMACHCR_0 to DMACHCR_47 | Initialized | Retained |
| DMATCRB_0 to DMATCRB_47 | Initialized | Retained |
| DMATSRB_0 to DMATSRB_47 | Initialized | Retained |
| DMACHCRB_0 to DMACHCRB_47 | Initialized | Retained |
| DMABUFCR_0 to DMABUFCR_47 | Initialized | Retained |
| DMARS_0 to DMARS_47 | Initialized | Retained |
| DMADPBASE_0 to DMADPBASE_47 | Initialized | Retained |
| DMADPCR_0 to DMADPCR_47 | Initialized | Retained |
| DMAFIXSAR_0 to DMAFIXSAR_47 | Initialized | Retained |
| DMAFIXDAR_0 to DMAFIXDAR_47 | Initialized | Retained |
| DMAFIXDPBASE_0 to DMAFIXDPBASE_47 | Initialized | Retained |
| DMASES_0 to DMASES_2 | Initialized | Retained |
| DMASEA_0 to DMASEA_2 | Initialized | Retained |
| DMAEMID_0 to DMAEMID_2 | Initialized | Retained |
| DescriptorMEM | Undefined | Retained |

23.1.4 Connected Module

Table 23.3 shows the connected modules to SYS-DMAC. As for the feature of the connected modules, refer to the chapter of each module.

Table 23.3 Connected Modules

| Module name | Connected module name | Function of connected module |
|-------------|-----------------------|------------------------------|
| SYS-DMAC | CPG | Output clocks |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |
| | INTC-SYS | Control to interrupt |

23.2 Register Description

23.2.1 DMA Interrupt Status Register for channels 0 to 15 (DMAISTA_0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMAISTA_0 is a 32-bit readable register that indicates the states of the interrupt signals for each of the lower-numbered channels.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | I15 | I14 | I13 | I12 | I11 | I10 | I9 | I8 | I7 | I6 | I5 | I4 | I3 | I2 | I1 | I0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | I15 | B'0 | R | Interrupt State in Channel 15 0: An interrupt is not present. 1: An interrupt is present. |
| 14 | I14 | B'0 | R | Interrupt State in Channel 14 0: An interrupt is not present. 1: An interrupt is present. |
| 13 | I13 | B'0 | R | Interrupt State in Channel 13 0: An interrupt is not present. 1: An interrupt is present. |
| 12 | I12 | B'0 | R | Interrupt State in Channel 12 0: An interrupt is not present. 1: An interrupt is present. |
| 11 | I11 | B'0 | R | Interrupt State in Channel 11 0: An interrupt is not present. 1: An interrupt is present. |
| 10 | I10 | B'0 | R | Interrupt State in Channel 10 0: An interrupt is not present. 1: An interrupt is present. |
| 9 | I9 | B'0 | R | Interrupt State in Channel 9 0: An interrupt is not present. 1: An interrupt is present. |
| 8 | I8 | B'0 | R | Interrupt State in Channel 8 0: An interrupt is not present. 1: An interrupt is present. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|--|
| 7 | I7 | B'0 | R | Interrupt State in Channel 7 0: An interrupt is not present. 1: An interrupt is present. |
| 6 | I6 | B'0 | R | Interrupt State in Channel 6 0: An interrupt is not present. 1: An interrupt is present. |
| 5 | I5 | B'0 | R | Interrupt State in Channel 5 0: An interrupt is not present. 1: An interrupt is present. |
| 4 | I4 | B'0 | R | Interrupt State in Channel 4 0: An interrupt is not present. 1: An interrupt is present. |
| 3 | I3 | B'0 | R | Interrupt State in Channel 3 0: An interrupt is not present. 1: An interrupt is present. |
| 2 | I2 | B'0 | R | Interrupt State in Channel 2 0: An interrupt is not present. 1: An interrupt is present. |
| 1 | I1 | B'0 | R | Interrupt State in Channel 1 0: An interrupt is not present. 1: An interrupt is present. |
| 0 | I0 | B'0 | R | Interrupt State in Channel 0 0: An interrupt is not present. 1: An interrupt is present. |

23.2.2 DMA Interrupt Status Register for channels 16 to 31 (DMAISTA_1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMAISTA_1 is a 32-bit readable register that indicates the states of the interrupt signals for each of the higher-numbered channels.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | I31 | I30 | I29 | I28 | I27 | I26 | I25 | I24 | I23 | I22 | I21 | I20 | I19 | I18 | I17 | I16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | I31 | B'0 | R | Interrupt State in Channel 31 0: An interrupt is not present. 1: An interrupt is present. |
| 14 | I30 | B'0 | R | Interrupt State in Channel 30 0: An interrupt is not present. 1: An interrupt is present. |
| 13 | I29 | B'0 | R | Interrupt State in Channel 29 0: An interrupt is not present. 1: An interrupt is present. |
| 12 | I28 | B'0 | R | Interrupt State in Channel 28 0: An interrupt is not present. 1: An interrupt is present. |
| 11 | I27 | B'0 | R | Interrupt State in Channel 27 0: An interrupt is not present. 1: An interrupt is present. |
| 10 | I26 | B'0 | R | Interrupt State in Channel 26 0: An interrupt is not present. 1: An interrupt is present. |
| 9 | I25 | B'0 | R | Interrupt State in Channel 25 0: An interrupt is not present. 1: An interrupt is present. |
| 8 | I24 | B'0 | R | Interrupt State in Channel 24 0: An interrupt is not present. 1: An interrupt is present. |
| 7 | I23 | B'0 | R | Interrupt State in Channel 23 0: An interrupt is not present. 1: An interrupt is present. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|---|
| 6 | I22 | B'0 | R | Interrupt State in Channel 22 0: An interrupt is not present. 1: An interrupt is present. |
| 5 | I21 | B'0 | R | Interrupt State in Channel 21 0: An interrupt is not present. 1: An interrupt is present. |
| 4 | I20 | B'0 | R | Interrupt State in Channel 20 0: An interrupt is not present. 1: An interrupt is present. |
| 3 | I19 | B'0 | R | Interrupt State in Channel 19 0: An interrupt is not present. 1: An interrupt is present. |
| 2 | I18 | B'0 | R | Interrupt State in Channel 18 0: An interrupt is not present. 1: An interrupt is present. |
| 1 | I17 | B'0 | R | Interrupt State in Channel 17 0: An interrupt is not present. 1: An interrupt is present. |
| 0 | I16 | B'0 | R | Interrupt State in Channel 16 0: An interrupt is not present. 1: An interrupt is present. |

23.2.3 DMA Interrupt Status Register for channels 32 to 47 (DMAISTA_2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMAISTA_2 is a 32-bit readable register that indicates the states of the interrupt signals for each of the higher-numbered channels.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | I47 | I46 | I45 | I44 | I43 | I42 | I41 | I40 | I39 | I38 | I37 | I36 | I35 | I34 | I33 | I32 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | I47 | B'0 | R | Interrupt State in Channel 47 0: An interrupt is not present. 1: An interrupt is present. |
| 14 | I46 | B'0 | R | Interrupt State in Channel 46 0: An interrupt is not present. 1: An interrupt is present. |
| 13 | I45 | B'0 | R | Interrupt State in Channel 45 0: An interrupt is not present. 1: An interrupt is present. |
| 12 | I44 | B'0 | R | Interrupt State in Channel 44 0: An interrupt is not present. 1: An interrupt is present. |
| 11 | I43 | B'0 | R | Interrupt State in Channel 43 0: An interrupt is not present. 1: An interrupt is present. |
| 10 | I42 | B'0 | R | Interrupt State in Channel 42 0: An interrupt is not present. 1: An interrupt is present. |
| 9 | I41 | B'0 | R | Interrupt State in Channel 41 0: An interrupt is not present. 1: An interrupt is present. |
| 8 | I40 | B'0 | R | Interrupt State in Channel 40 0: An interrupt is not present. 1: An interrupt is present. |
| 7 | I39 | B'0 | R | Interrupt State in Channel 39 0: An interrupt is not present. 1: An interrupt is present. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|---|
| 6 | I38 | B'0 | R | Interrupt State in Channel 38 0: An interrupt is not present. 1: An interrupt is present. |
| 5 | I37 | B'0 | R | Interrupt State in Channel 37 0: An interrupt is not present. 1: An interrupt is present. |
| 4 | I36 | B'0 | R | Interrupt State in Channel 36 0: An interrupt is not present. 1: An interrupt is present. |
| 3 | I35 | B'0 | R | Interrupt State in Channel 35 0: An interrupt is not present. 1: An interrupt is present. |
| 2 | I34 | B'0 | R | Interrupt State in Channel 34 0: An interrupt is not present. 1: An interrupt is present. |
| 1 | I33 | B'0 | R | Interrupt State in Channel 33 0: An interrupt is not present. 1: An interrupt is present. |
| 0 | I32 | B'0 | R | Interrupt State in Channel 32 0: An interrupt is not present. 1: An interrupt is present. |

23.2.4 DMA Secure Control Register for channels 0 to 15 (DMASEC_0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASEC_0 is a 32-bit readable/writable register that controls the security attribute of each of the lower-numbered channels. Only the initiator in the secure mode can change the setting of this register.

Only secure access is allowed to register of channels with the secure mode setting. The following registers are protected by the secure mode.

DMASAR_0, DMADAR_0, DMATCR_0, DMATSR_0, DMACHCR_0, DMATCRB_0, DMATSRB_0, DMACHCRB_0, DMARS_0, DMABUFCR_0, DMADPBASE_0, DMADPCR_0, DMAFIXSAR_0, DMAFIXDAR_0, and DMAFIXDPBASE_0.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | S15 | S14 | S13 | S12 | S11 | S10 | S9 | S8 | S7 | S6 | S5 | S4 | S3 | S2 | S1 | S0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | S15 | B'0 | R/W | Secure Mode Setting for Channel 15 0: Non-secure mode 1: Secure mode |
| 14 | S14 | B'0 | R/W | Secure Mode Setting for Channel 14 0: Non-secure mode 1: Secure mode |
| 13 | S13 | B'0 | R/W | Secure Mode Setting for Channel 13 0: Non-secure mode 1: Secure mode |
| 12 | S12 | B'0 | R/W | Secure Mode Setting for Channel 12 0: Non-secure mode 1: Secure mode |
| 11 | S11 | B'0 | R/W | Secure Mode Setting for Channel 11 0: Non-secure mode 1: Secure mode |
| 10 | S10 | B'0 | R/W | Secure Mode Setting for Channel 10 0: Non-secure mode 1: Secure mode |
| 9 | S9 | B'0 | R/W | Secure Mode Setting for Channel 9 0: Non-secure mode 1: Secure mode |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|---|
| 8 | S8 | B'0 | R/W | Secure Mode Setting for Channel 8 0: Non-secure mode 1: Secure mode |
| 7 | S7 | B'0 | R/W | Secure Mode Setting for Channel 7 0: Non-secure mode 1: Secure mode |
| 6 | S6 | B'0 | R/W | Secure Mode Setting for Channel 6 0: Non-secure mode 1: Secure mode |
| 5 | S5 | B'0 | R/W | Secure Mode Setting for Channel 5 0: Non-secure mode 1: Secure mode |
| 4 | S4 | B'0 | R/W | Secure Mode Setting for Channel 4 0: Non-secure mode 1: Secure mode |
| 3 | S3 | B'0 | R/W | Secure Mode Setting for Channel 3 0: Non-secure mode 1: Secure mode |
| 2 | S2 | B'0 | R/W | Secure Mode Setting for Channel 2 0: Non-secure mode 1: Secure mode |
| 1 | S1 | B'0 | R/W | Secure Mode Setting for Channel 1 0: Non-secure mode 1: Secure mode |
| 0 | S0 | B'0 | R/W | Secure Mode Setting for Channel 0 0: Non-secure mode 1: Secure mode |

23.2.5 DMA Secure Control Register for channels 16 to 31 (DMASEC_1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASEC_1 is a 32-bit readable/writable register that controls the security attribute of each of the higher-numbered channels. Only the initiator in the secure mode can change the setting of this register.

Only secure access is allowed to register of channels with the secure mode setting. The following registers are protected by the secure mode.

DMASAR_1, DMADAR_1, DMATCR_1, DMATSR_1, DMACHCR_1, DMATCRB_1, DMATSRB_1, DMACHCRB_1, DMARS_1, DMABUFCR_1, DMADPBASE_1, DMADPCR_1, DMAFIXSAR_1, DMAFIXDAR_1, and DMAFIXDPBASE_1.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | S31 | S30 | S29 | S28 | S27 | S26 | S25 | S24 | S23 | S22 | S21 | S20 | S19 | S18 | S17 | S16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | S31 | B'0 | R/W | Secure Mode Setting for Channel 31 0: Non-secure mode 1: Secure mode |
| 14 | S30 | B'0 | R/W | Secure Mode Setting for Channel 30 0: Non-secure mode 1: Secure mode |
| 13 | S29 | B'0 | R/W | Secure Mode Setting for Channel 29 0: Non-secure mode 1: Secure mode |
| 12 | S28 | B'0 | R/W | Secure Mode Setting for Channel 28 0: Non-secure mode 1: Secure mode |
| 11 | S27 | B'0 | R/W | Secure Mode Setting for Channel 27 0: Non-secure mode 1: Secure mode |
| 10 | S26 | B'0 | R/W | Secure Mode Setting for Channel 26 0: Non-secure mode 1: Secure mode |
| 9 | S25 | B'0 | R/W | Secure Mode Setting for Channel 25 0: Non-secure mode 1: Secure mode |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|--|
| 8 | S24 | B'0 | R/W | Secure Mode Setting for Channel 24 0: Non-secure mode 1: Secure mode |
| 7 | S23 | B'0 | R/W | Secure Mode Setting for Channel 23 0: Non-secure mode 1: Secure mode |
| 6 | S22 | B'0 | R/W | Secure Mode Setting for Channel 22 0: Non-secure mode 1: Secure mode |
| 5 | S21 | B'0 | R/W | Secure Mode Setting for Channel 21 0: Non-secure mode 1: Secure mode |
| 4 | S20 | B'0 | R/W | Secure Mode Setting for Channel 20 0: Non-secure mode 1: Secure mode |
| 3 | S19 | B'0 | R/W | Secure Mode Setting for Channel 19 0: Non-secure mode 1: Secure mode |
| 2 | S18 | B'0 | R/W | Secure Mode Setting for Channel 18 0: Non-secure mode 1: Secure mode |
| 1 | S17 | B'0 | R/W | Secure Mode Setting for Channel 17 0: Non-secure mode 1: Secure mode |
| 0 | S16 | B'0 | R/W | Secure Mode Setting for Channel 16 0: Non-secure mode 1: Secure mode |

23.2.6 DMA Secure Control Register for channels 32 to 47 (DMASEC_2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASEC_2 is a 32-bit readable/writable register that controls the security attribute of each of the higher-numbered channels. Only the initiator in the secure mode can change the setting of this register.

Only secure access is allowed to register of channels with the secure mode setting. The following registers are protected by the secure mode.

DMASAR_2, DMADAR_2, DMATCR_2, DMATSR_2, DMACHCR_2, DMATCRB_2, DMATSRB_2, DMACHCRB_2, DMARS_2, DMABUFCR_2, DMADPBASE_2, DMADPCR_2, DMAFIXSAR_2, DMAFIXDAR_2, and DMAFIXDPBASE_2.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | S47 | S46 | S45 | S44 | S43 | S42 | S41 | S40 | S39 | S38 | S37 | S36 | S35 | S34 | S33 | S32 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | S47 | B'0 | R/W | Secure Mode Setting for Channel 47 0: Non-secure mode 1: Secure mode |
| 14 | S46 | B'0 | R/W | Secure Mode Setting for Channel 46 0: Non-secure mode 1: Secure mode |
| 13 | S45 | B'0 | R/W | Secure Mode Setting for Channel 45 0: Non-secure mode 1: Secure mode |
| 12 | S44 | B'0 | R/W | Secure Mode Setting for Channel 44 0: Non-secure mode 1: Secure mode |
| 11 | S43 | B'0 | R/W | Secure Mode Setting for Channel 43 0: Non-secure mode 1: Secure mode |
| 10 | S42 | B'0 | R/W | Secure Mode Setting for Channel 42 0: Non-secure mode 1: Secure mode |
| 9 | S41 | B'0 | R/W | Secure Mode Setting for Channel 41 0: Non-secure mode 1: Secure mode |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|--|
| 8 | S40 | B'0 | R/W | Secure Mode Setting for Channel 40 0: Non-secure mode 1: Secure mode |
| 7 | S39 | B'0 | R/W | Secure Mode Setting for Channel 39 0: Non-secure mode 1: Secure mode |
| 6 | S38 | B'0 | R/W | Secure Mode Setting for Channel 38 0: Non-secure mode 1: Secure mode |
| 5 | S37 | B'0 | R/W | Secure Mode Setting for Channel 37 0: Non-secure mode 1: Secure mode |
| 4 | S36 | B'0 | R/W | Secure Mode Setting for Channel 36 0: Non-secure mode 1: Secure mode |
| 3 | S35 | B'0 | R/W | Secure Mode Setting for Channel 35 0: Non-secure mode 1: Secure mode |
| 2 | S34 | B'0 | R/W | Secure Mode Setting for Channel 34 0: Non-secure mode 1: Secure mode |
| 1 | S33 | B'0 | R/W | Secure Mode Setting for Channel 33 0: Non-secure mode 1: Secure mode |
| 0 | S32 | B'0 | R/W | Secure Mode Setting for Channel 32 0: Non-secure mode 1: Secure mode |

23.2.7 DMA Operation Register for channels 0 to 15 (DMAOR_0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMAOR_0 is a 16-bit readable/writable register that enables DMA transfer on all lower-numbered channels and specifies the method used to determine the priority levels for all lower-numbered DMA channels. This register also indicates address errors.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---------|-----|---|---|---|---|---|--------|---|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | PR[1:0] | — | — | — | — | — | — | AE | — | DME |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R/(W)* | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|------------|---|
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9, 8 | PR[1:0] | B'00 | R/W | Priority Mode Select the method for setting the order of priority of channels when transfer requests for multiple channels arrive simultaneously. B'00: Fixed CH0 > CH1 > ... > CH14 > CH15 B'11: Round-robin priority Other than above: Setting prohibited |
| 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | AE | B'0 | R/(W) * | Address Error Flag Indicates that an address error interrupt occurred during DMA transfer. This bit is set under the following conditions: The value set in DMASAR_0 or DMADAR_0 does not fall on a boundary corresponding to the transfer size. The source or destination for transfer is in an invalid space. The source or destination for transfer is in module stop mode. If this bit is set, DMA transfer through the channel is not possible even if the DE bit is set to 1. To clear the AE bit, write 0 to the AE bit after reading 1 from it or clear the CAE bit for each channel for which it is set. Clearing the AE bit clears the channel address error bits for all channels. 0: A SYS-DMAC address error interrupt is not present. [Clearing condition] (*) Writing CAE = 0 after reading CAE = 1 1: A SYS-DMAC address error interrupt being generated during DMA transfer. |
| 1 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|---|
| 0 | DME | B'0 | R/W | <p>DMA Master Enable</p> <p>Enables or disables DMA transfer on all channels. If the DME bit and the DE bit in DMACHCR_0 are both set to 1, DMA transfer is enabled. At this time all AE bits in DMAOR_0 must have the value 0. For DMA transfer on a channel to then proceed, the TE bit in DMACHCR_0 for the channel must also have the value 0. Clearing this bit during transfer aborts transfer on all channels.</p> <p>0: Disables DMA transfers on all channels 1: Enables DMA transfers on all channels</p> |

23.2.8 DMA Operation Register for channels 16 to 31 (DMAOR_1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMAOR_1 is a 16-bit readable/writable register that enables DMA transfer on all higher-numbered channels and specifies the method used to determine the priority levels for all higher-numbered DMA channels. This register also indicates address errors.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---------|-----|---|---|---|---|---|--------|---|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | PR[1:0] | — | — | — | — | — | — | AE | — | DME |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R/(W)* | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|------------|---|
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9, 8 | PR[1:0] | B'00 | R/W | Priority Mode Select the method for setting the order of priority of channels when transfer requests for multiple channels arrive simultaneously. B'00: Fixed CH16 > CH17 > ... > CH30 > CH31 B'11: Round-robin priority Other than above: Setting prohibited |
| 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | AE | B'0 | R/(W) * | Address Error Flag Indicates that an address error interrupt occurred during DMA transfer. This bit is set under the following conditions: The value set in DMASAR_1 or DMADAR_1 does not fall on a boundary corresponding to the transfer size. The source or destination for transfer is in an invalid space. The source or destination for transfer is in module stop mode. If this bit is set, DMA transfer through the channel is not possible even if the DE bit is set to 1. To clear the AE bit, write 0 to the AE bit after reading 1 from it or clear the CAE bit for each channel for which it is set. Clearing the AE bit clears the channel address error bits for all channels. 0: A SYS-DMAC address error interrupt is not present. [Clearing condition] (*) Writing CAE = 0 after reading CAE = 1 1: A SYS-DMAC address error interrupt being generated during DMA transfer. |
| 1 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|---|
| 0 | DME | B'0 | R/W | <p>DMA Master Enable</p> <p>Enables or disables DMA transfer on all channels. If the DME bit and the DE bit in DMACHCR_1 are both set to 1, DMA transfer is enabled. At this time all AE bits in DMAOR_1 must have the value 0. For DMA transfer on a channel to then proceed, the TE bit in DMACHCR_1 for the channel must also have the value 0. Clearing this bit during transfer aborts transfer on all channels.</p> <p>0: Disables DMA transfers on all channels 1: Enables DMA transfers on all channels</p> |

23.2.9 DMA Operation Register for channels 32 to 47 (DMAOR_2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMAOR_2 is a 16-bit readable/writable register that enables DMA transfer on all higher-numbered channels and specifies the method used to determine the priority levels for all higher-numbered DMA channels. This register also indicates address errors.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---------|-----|---|---|---|---|---|--------|---|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | PR[1:0] | — | — | — | — | — | — | AE | — | DME |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R/(W)* | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|--------|--|
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9, 8 | PR[1:0] | B'00 | R/W | Priority Mode Select the method for setting the order of priority of channels when transfer requests for multiple channels arrive simultaneously. B'00: Fixed CH32 > CH33 > ... > CH46 > CH47 B'11: Round-robin priority Other than above: Setting prohibited |
| 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | AE | B'0 | R/(W)* | Address Error Flag Indicates that an address error interrupt occurred during DMA transfer. This bit is set under the following conditions: The value set in DMASAR_2 or DMADAR_2 does not fall on a boundary corresponding to the transfer size. The source or destination for transfer is in an invalid space. The source or destination for transfer is in module stop mode. If this bit is set, DMA transfer through the channel is not possible even if the DE bit is set to 1. To clear the AE bit, write 0 to the AE bit after reading 1 from it or clear the CAE bit for each channel for which it is set. Clearing the AE bit clears the channel address error bits for all channels. 0: A SYS-DMAC address error interrupt is not present. [Clearing condition] (*) Writing CAE = 0 after reading CAE = 1 1: A SYS-DMAC address error interrupt being generated during DMA transfer. |
| 1 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|---|
| 0 | DME | B'0 | R/W | <p>DMA Master Enable</p> <p>Enables or disables DMA transfer on all channels. If the DME bit and the DE bit in DMACHCR_2 are both set to 1, DMA transfer is enabled. At this time all AE bits in DMAOR_2 must have the value 0. For DMA transfer on a channel to then proceed, the TE bit in DMACHCR_2 for the channel must also have the value 0. Clearing this bit during transfer aborts transfer on all channels.</p> <p>0: Disables DMA transfers on all channels 1: Enables DMA transfers on all channels</p> |

23.2.10 DMA Channel Clear Register for channels 0 to 15 (DMACHCLR_0)

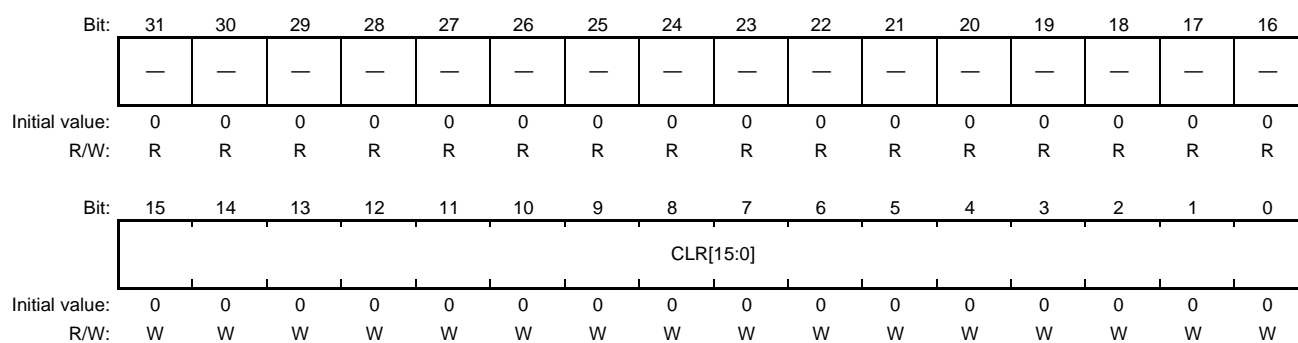
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMACHCLR_0 is a 32-bit writable register that initializes each of the lower-numbered channels.

When a bit of this register is set, the state of the corresponding channel is completely initialized.

This includes initialization of the following registers.

DMASAR_0, DMADAR_0, DMATCR_0, DMATSR_0, DMACHCR_0, DMATCRB_0, DMATSRB_0, DMACHCRB_0, DMARS_0, DMABUFCR_0, DMADPBASE_0, DMADPCR_0, DMAFIXSAR_0, DMAFIXDAR_0, and DMAFIXDPBASE_0



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | CLR[15:0] | H'0000 | W | Writing to a bit leads to clearing of all registers for the corresponding channel. CLR[0] 0: Ignored 1: All registers for channel 0 are cleared. CLR[1] 0: Ignored 1: All registers for channel 1 are cleared. CLR[2] 0: Ignored 1: All registers for channel 2 are cleared. ... CLR[15] 0: Ignored 1: All registers for channel 15 are cleared. When writing to this register, confirm that the DE bit is set to 0. |

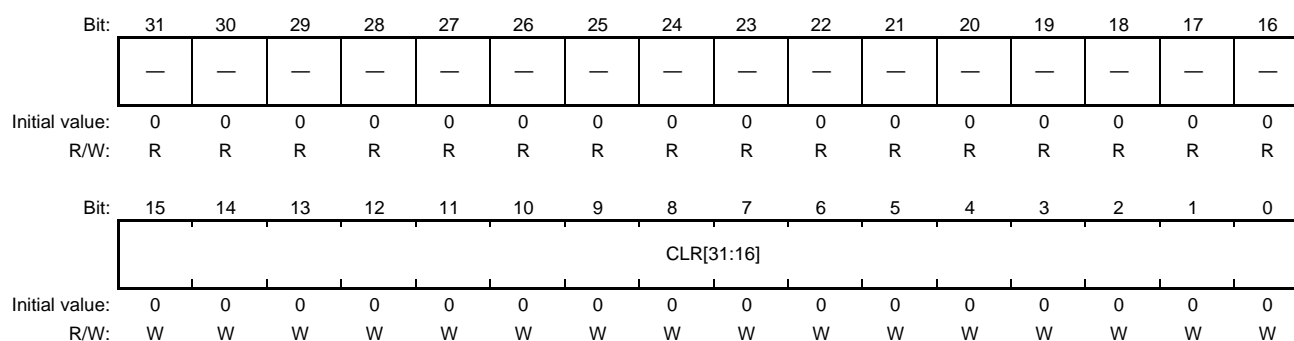
23.2.11 DMA Channel Clear Register for channels 16 to 31 (DMACHCLR_1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMACHCLR_1 is a 32-bit writable register that initializes each of the higher-numbered channels.

When a bit of this register is set, the state of the corresponding channel is completely initialized. This includes initialization of the following registers.

DMASAR_1, DMADAR_1, DMATCR_1, DMATSR_1, DMACHCR_1, DMATCRB_1, DMATSRB_1, DMACHCRB_1, DMARS_1, DMABUFCR_1, DMADPBASE_1, DMADPCR_1, DMAFIXSAR_1, DMAFIXDAR_1, and DMAFIXDPBASE_1.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | CLR[31:16] | H'0000 | W | Writing to a bit leads to clearing of all registers for the corresponding channel. CLR[16] 0: Ignored 1: All registers for channel 16 are cleared. CLR[17] 0: Ignored 1: All registers for channel 17 are cleared. ... CLR[31] 0: Ignored 1: All registers for channel 31 are cleared. When writing to this register, confirm that the DE bit is set to 0. |

23.2.12 DMA Channel Clear Register for channels 32 to 47 (DMACHCLR_2)

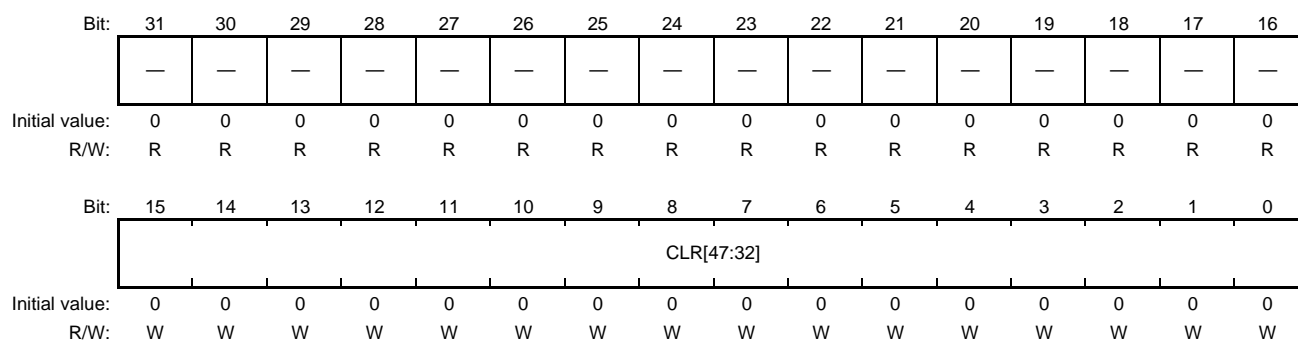
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMACHCLR_2 is a 32-bit writable register that initializes each of the higher-numbered channels.

When a bit of this register is set, the state of the corresponding channel is completely initialized.

This includes initialization of the following registers.

DMASAR_2, DMADAR_2, DMATCR_2, DMATSR_2, DMACHCR_2, DMATCRB_2, DMATSRB_2, DMACHCRB_2, DMARS_2, DMABUF_2, DMADPBASE_2, DMADPCR_2, DMAFIXSAR_2, DMAFIXDAR_2, and DMAFIXDPBASE_2.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | CLR[47:32] | H'0000 | W | Writing to a bit leads to clearing of all registers for the corresponding channel. CLR[32] 0: Ignored 1: All registers for channel 32 are cleared. CLR[33] 0: Ignored 1: All registers for channel 33 are cleared. ... CLR[47] 0: Ignored 1: All registers for channel 47 are cleared. When writing to this register, confirm that the DE bit is set to 0. |

23.2.13 DPRAM Secure Control Register for channels 0 to 15 (DMADPSEC_0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMADPSEC_0 is a 32-bit readable/writable register that controls the security attribute of the descriptor memory. Only the initiator in the secure mode can change the setting of this register.

| | | | | | | | | | | | | | | | | | | | |
|----------------|-----|----|----|----|----|----|----|---------|-----|-----|-----|-----|-----|-----|-----|-----|--|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | | |
| | SEC | — | — | — | — | — | — | SA[8:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R/W | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | |
| | — | — | — | — | — | — | — | SM[8:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | SEC | B'0 | R/W | Security Attribute Setting for Descriptor Memory Specifies the security attribute of the address space used for the descriptor memory. 0: Non-secure 1: Secure |
| 30 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 to 16 | SA[8:0] | H'000 | R/W | Security Attribute Setting for Base Address of Descriptor Memory Specify the base address of the descriptor memory to be assigned the security attribute. H'000: H'A000 H'001: H'A004 ... H'1FF: H'A7FC |
| 15 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 to 0 | SM[8:0] | H'000 | R/W | Security Attribute Setting for Base Address Mask of Descriptor Memory Specify the security attribute base address mask of the descriptor memory. The range of memory to be assigned the security attribute is specified by this register. See Figure 23.4. |

23.2.14 DPRAM Secure Control Register for channels 16 to 31 (DMADPSEC_1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMADPSEC_1 is a 32-bit readable/writable register that controls the security attribute of the descriptor memory. Only the initiator in the secure mode can change the setting of this register.

| | | | | | | | | | | | | | | | | | | | |
|----------------|-----|----|----|----|----|----|----|---------|-----|-----|-----|-----|-----|-----|-----|-----|--|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | | |
| | SEC | — | — | — | — | — | — | SA[8:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R/W | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | |
| | — | — | — | — | — | — | — | SM[8:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | SEC | B'0 | R/W | Security Attribute Setting for Descriptor Memory Specifies the security attribute of the address space used for the descriptor memory. 0: Non-secure 1: Secure |
| 30 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 to 16 | SA[8:0] | H'000 | R/W | Security Attribute Setting for Base Address of Descriptor Memory Specify the base address of the descriptor memory to be assigned the security attribute. H'000: H'A000 H'001: H'A004 ... H'1FF: H'A7FC |
| 15 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 to 0 | SM[8:0] | H'000 | R/W | Security Attribute Setting for Base Address Mask of Descriptor Memory Specify the security attribute base address mask of the descriptor memory. The range of memory to be assigned the security attribute is specified by this register. See Figure 23.4. |

23.2.15 DPRAM Secure Control Register for channels 32 to 47 (DMADPSEC_2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMADPSEC_2 is a 32-bit readable/writable register that controls the security attribute of the descriptor memory. Only the initiator in the secure mode can change the setting of this register.

| | | | | | | | | | | | | | | | | | | | |
|----------------|-----|----|----|----|----|----|----|---------|-----|-----|-----|-----|-----|-----|-----|-----|--|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | | |
| | SEC | — | — | — | — | — | — | SA[8:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R/W | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | |
| | — | — | — | — | — | — | — | SM[8:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |

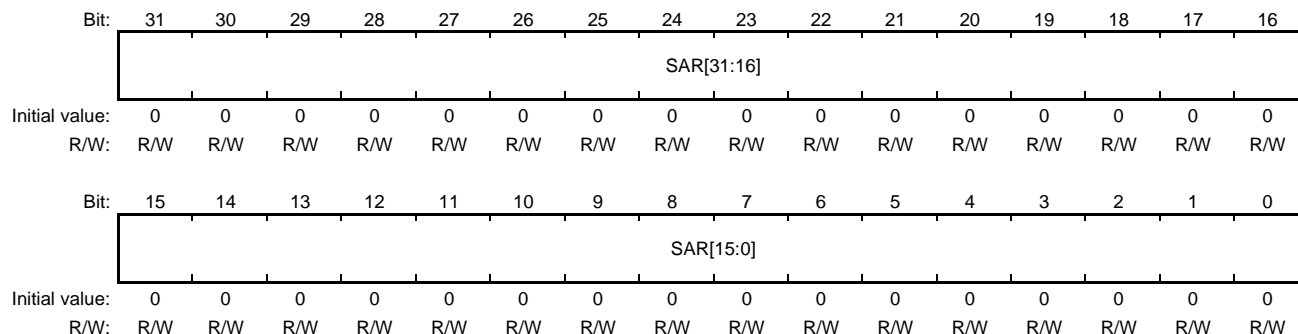
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | SEC | B'0 | R/W | Security Attribute Setting for Descriptor Memory Specifies the security attribute of the address space used for the descriptor memory. 0: Non-secure 1: Secure |
| 30 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 to 16 | SA[8:0] | H'000 | R/W | Security Attribute Setting for Base Address of Descriptor Memory Specify the base address of the descriptor memory to be assigned the security attribute. H'000: H'A000 H'001: H'A004 ... H'1FF: H'A7FC |
| 15 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 to 0 | SM[8:0] | H'000 | R/W | Security Attribute Setting for Base Address Mask of Descriptor Memory Specify the security attribute base address mask of the descriptor memory. The range of memory to be assigned the security attribute is specified by this register. See Figure 23.4. |

23.2.16 DMA Source Address Registers 0 to 47 (DMASAR_0 to DMASAR_47)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASAR is a 32-bit readable/writable register that specifies the source address of a DMA transfer. While a DMA transfer is in progress, this register indicates the next source address.

When the address mode is incremental, sources in memory only have byte boundaries. For details, refer to Table 23.4.



23.2.17 DMA Destination Address Registers 0 to 47 (DMADAR_0 to DMADAR_47)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMADAR is 32-bit readable/writable register that specify the destination address of a DMA transfer. While a DMA transfer is in progress, this register indicates the next destination address.

When the address mode is incremental, destinations in memory only have byte boundaries. For details, refer to Table 23.4.

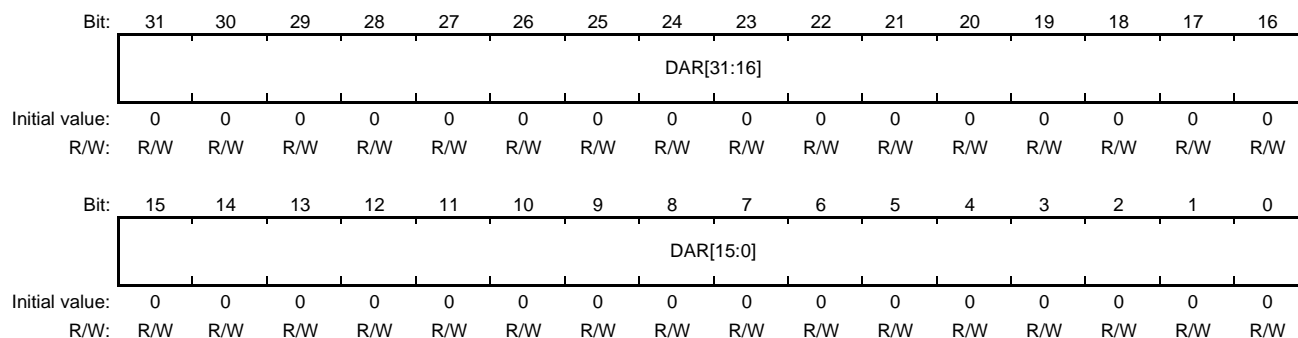


Table 23.4 SAR/DAR Address Restriction

| Resource Selection | Address Mode | Restriction |
|--|--------------|---|
| Auto request | Incremental | No restriction (byte boundaries) |
| | Others | Boundary corresponding to the DMA transfer size |
| On-chip peripheral module request Transmission/DAR, Reception/SAR | All | Boundary corresponding to the DMA transfer size |
| On-chip peripheral module request Transmission/SAR, Reception/DAR | Incremental | No restriction (byte boundaries) |
| | Others | Boundary corresponding to the DMA transfer size |

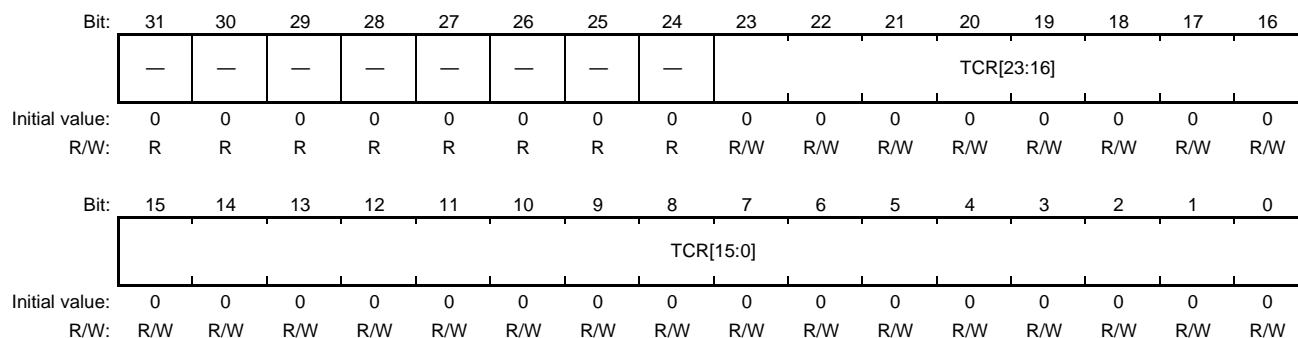
23.2.18 DMA Transfer Count Registers 0 to 47 (DMATCR_0 to DMATCR_47)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMATCR is a 32-bit readable/writable register that specifies the number of rounds of DMA transfer. The number of rounds of DMA transfer is 1 when the setting is H'0000_0001, 16,777,215 when the setting is H'00FF_FFFF (the maximum). During a DMA transfer, this register indicates the remaining number of rounds of transfer.

The SYS-DMAC includes independent data buffers for reading and writing. Therefore, the read transfer counter and write transfer counter have different values. This register indicates the counter value used in reading.

The eight higher-order bits of DMATCR are always read as 0, and the write value should always be 0.



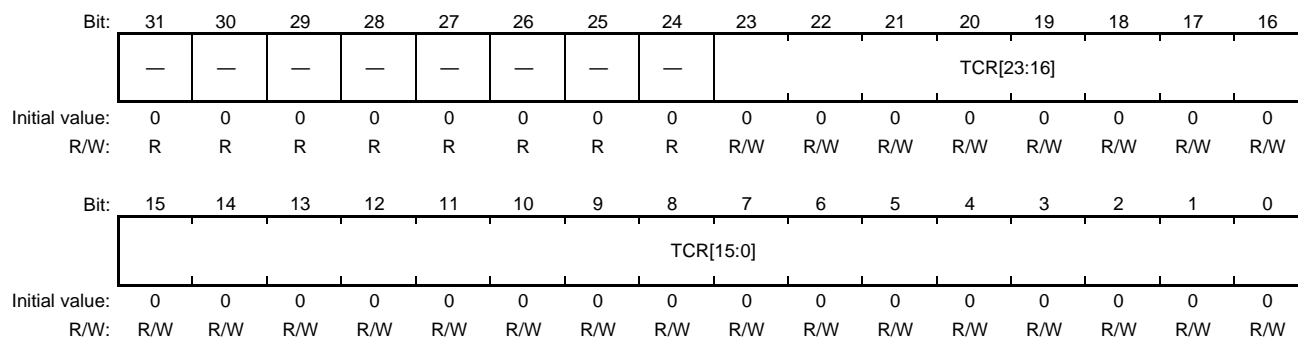
23.2.19 DMA Transfer Count Registers B_0 to 47 (DMATCRB_0 to DMATCRB_47)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMATCRB is a 32-bit readable/writable register that specifies the number of rounds of DMA transfer. The number of rounds of DMA transfer is 1 when the setting is H'0000_0001, 16,777,215 when the setting is H'00FF_FFFF (the maximum). During a DMA transfer, this register indicates the remaining number of rounds of transfer.

The SYS-DMAC includes independent data buffers for reading and writing. Therefore, the read transfer counter and write transfer counter have different values. This register indicates the counter value used in writing.

The eight higher-order bits of DMATCRB are always read as 0, and the write value should always be 0.

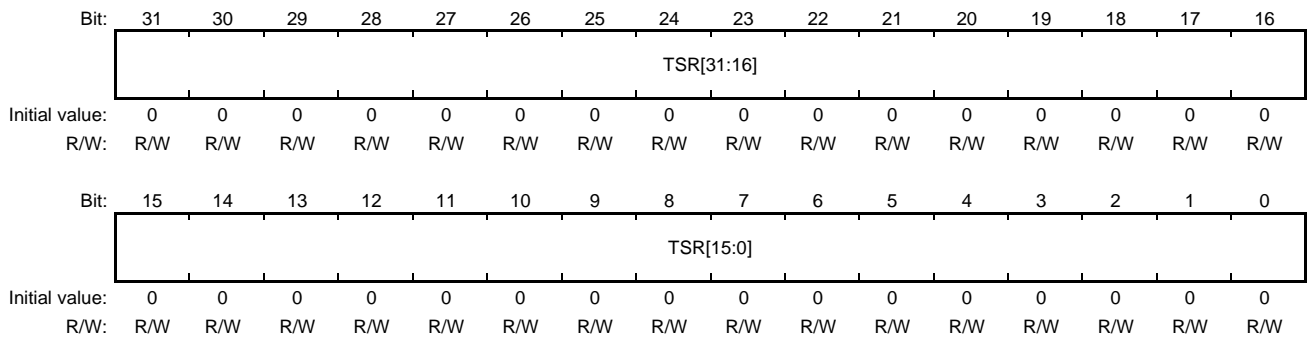


23.2.20 DMA Transfer Size Registers 0 to 47 (DMATSR_0 to DMATSR_47)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMATSR is a 32-bit readable/writable register that specifies a total amount of memory to be transferred. The total size of DMA transfer is 1 byte when the setting is H'0000_0001, 4,294,967,295 bytes when the setting is H'FFFF_FFFF, and 4,294,967,296 bytes (the maximum) when the setting is H'0000_0000. During a DMA transfer, this register indicates the remaining amount of memory to be transferred. This register is used in total size transmission.

The SYS-DMAC includes independent data buffers for reading and writing. Therefore, reading and writing will have different transfer sizes. This register indicates the value of the read transfer size.

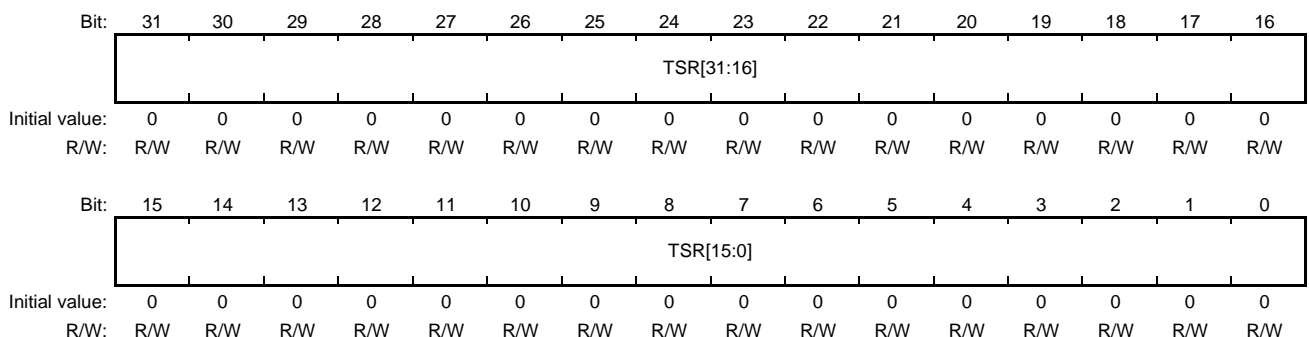


23.2.21 DMA Transfer Size Registers B_0 to 47 (DMATSRB_0 to DMATSRB_47)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMATSRB is a 32-bit readable/writable register that specifies a total amount of memory to be transferred. The total size of DMA transfer is 1 byte when the setting is H'0000_0001, 4,294,967,295 bytes when the setting is H'FFFF_FFFF, and 4,294,967,296 bytes (the maximum) when the setting is H'0000_0000. During a DMA transfer, this register indicates the remaining amount of memory to be transferred. This register is used in total size transmission.

The SYS-DMAC includes independent data buffers for reading and writing. Therefore, reading and writing will have different transfer sizes. This register indicates the value of the write transfer size.



23.2.22 DMA Channel Control Registers 0 to 47 (DMACHCR_0 to DMACHCR_47)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMACHCR is a 32-bit readable/writable register that controls the DMA transfer mode.

| | | | | | | | | | | | | | | | | |
|----------------|---------|------|----------|-----|----------|-----|-----|-----|----|-----|---------|-----|--------|------|--------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CAE | CAIE | DPM[1:0] | | RPT[2:0] | | | — | — | DPB | TS[3:2] | | DSE | DSIE | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/(W)* | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/(W)* | R/W | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DM[1:0] | | SM[1:0] | | RS[3:0] | | | — | — | — | TS[1:0] | | IE | TE | DE | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/(W)* | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|------------|--|
| 31 | CAE | B'0 | R/(W) * | <p>Channel Address Error Flag</p> <p>Indicates that an address error interrupt occurred during DMA transfer.</p> <p>This bit is set under the following conditions:</p> <ul style="list-style-type: none"> The value set in DMASAR or DMADAR does not fall on a boundary corresponding to the transfer size. The source or destination for transfer is in an invalid space. The source or destination for transfer is in module stop mode. <p>If this bit is set, DMA transfer through the channel is not possible even if the DE bit is set to 1.</p> <p>To clear the CAE bit, write 0 to the CAE bit after reading 1 from it or clear the AE bit in DMAOR.</p> <p>(When the AE bit in DMAOR is cleared, CAE bit in all channels are cleared.)</p> <p>0: A SYS-DMAC address error interrupt is not present. [Clearing condition] Writing CAE = 0 after reading CAE = 1</p> <p>1: A SYS-DMAC address error interrupt being generated during DMA transfer.</p> |
| 30 | CAIE | B'0 | R/W | <p>Channel Address Error Interrupt Enable</p> <p>Enables or disables the generation of interrupt requests for the CPU when address errors occur. When the CAIE bit is set to 1, if the CAE bit is also set, an interrupt (DEI 0 to 47) from the corresponding channel will be generated for the CPU in response to address errors.</p> <p>Note: An address error interrupt (DADERR) is also asserted simultaneously. See section 19, Interrupt Controller (INTC-AP) for more details.</p> <p>0: Interrupt requests are disabled. 1: Interrupt requests are enabled.</p> |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|------------|--|
| 29, 28 | DPM[1:0] | B'00 | R/W | <p>Operating Mode of Descriptor Memory</p> <p>Enable or disable the descriptor memory and specify its operating mode.</p> <p>B'00: Disabled (normal use)</p> <p>B'01: Enabled (normal mode)</p> <p>B'10: Enabled (repeat mode)</p> <p>B'11: Enabled (read-out interrupt mode, infinite repeat mode)</p> |
| 27 to 25 | RPT[2:0] | B'000 | R/W | <p>Descriptor Setting Update</p> <p>Specify the parameters to be updated from the descriptor memory.</p> <p>RPT[2]: Enables or disables updating of the source address register</p> <p>RPT[1]: Enables or disables updating of the destination address register</p> <p>RPT[0]: Enables or disables updating of the transfer count register</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 24, 23 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 22 | DPB | B'0 | R/W | <p>Descriptor Start</p> <p>Specifies configuration to be loaded when transfer under control of the descriptor memory begins.</p> <p>This bit is cleared after the descriptor memory is read.</p> <p>0: Processing starts with the values in DMASAR, DMADAR, and DMATCR.</p> <p>1: Processing starts after the first set of descriptors is read out.</p> |
| 21, 20 | TS[3:2] | B'00 | R/W | <p>DMA Transfer Size</p> <p>In combination with TS[1:0], these bits specify the DMA transfer size. When the transfer source or transfer destination is a register of an on-chip peripheral module for which a transfer size is specified, be sure to select the specified transfer size. For the transfer source or destination address specified by DMASAR or DMADAR, an appropriate boundary address should be set according to the transfer data size.</p> <p>TS[3:2] + TS[1:0] (“+” here indicates concatenation, not addition)</p> <p>B'0000: Transfer is in byte units.</p> <p>B'0001: Transfer is in word (2-byte) units.</p> <p>B'0010: Transfer is in longword (4-byte) units.</p> <p>B'0011: Transfer is in 16-byte units.</p> <p>B'0100: Transfer is in 32-byte units.</p> <p>B'0101: Transfer is in 64-byte units.</p> <p>B'0111: Transfer is in 8-byte units.</p> <p>Other than above: Setting prohibited</p> |
| 19 | DSE | B'0 | R/(W) * | <p>Descriptor Stage End</p> <p>(*) When the DSIE bit is set to 1 and the descriptor memory is enabled, the DSE bit is set to 1 on completion of the DMA transfer. This bit is not set when the DPM bit is set to 0 (descriptors are disabled). To clear the DSE bit, start by reading it as 1, and then write 0 to the bit.</p> <p>0: DMA transfer is still running or has been aborted.</p> <p>1: Transfer under the control of one stage of the descriptor memory has been completed.</p> |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|----------|---------------|-----|--|
| 18 | DSIE | B'0 | R/W | <p>Descriptor Stage End Interrupt Enable</p> <p>Specifies whether an interrupt request is generated for the CPU on completion of transfer under the control of one stage of the descriptor memory. When this bit is set to 1, an interrupt (DEI) is generated for the CPU whenever the DSE is set to 1.</p> <p>0: Interrupt requests are disabled. 1: Interrupt requests are enabled.</p> |
| 17, 16 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 15, 14 | DM[1:0] | B'00 | R/W | <p>Destination Address Mode</p> <p>Specify whether the DMA destination address is incremented, fixed, or decremented. The unit of transfer (transfer size) determines the size of the increment.</p> <p>B'00: Destination address is fixed. B'01: Destination addresses are incremented.</p> <ul style="list-style-type: none"> + 1 when transfer is in byte units. + 2 when transfer is in word units. + 4 when transfer is in longword units. + 8 when transfer is in 8-byte units. + 16 when transfer is in 16-byte units. + 32 when transfer is in 32-byte units. + 64 when transfer is in 64-byte units. <p>B'10: Destination addresses are decremented.</p> <ul style="list-style-type: none"> – 1 when transfer is in byte units. – 2 when transfer is in word units. – 4 when transfer is in longword units. <p>Setting prohibited when transfer is in 8-, 16-, 32-, or 64-byte units.</p> <p>B'11: Setting Prohibited</p> |
| 13, 12 | SM[1:0] | B'00 | R/W | <p>Source Address Mode</p> <p>Specify whether the DMA source address is incremented, fixed, or decremented. The unit of transfer (transfer size) determines the size of the increment.</p> <p>00: Source address is fixed. B'01: Source addresses are incremented.</p> <ul style="list-style-type: none"> + 1 when transfer is in byte units. + 2 when transfer is in word units. + 4 when transfer is in longword units. + 8 when transfer is in 8-byte units. + 16 when transfer is in 16-byte units. + 32 when transfer is in 32-byte units. + 64 when transfer is in 64-byte units. <p>B'10: Source addresses are decremented.</p> <ul style="list-style-type: none"> – 1 when transfer is in byte units. – 2 when transfer is in word units. – 4 when transfer is in longword units. <p>Setting prohibited when transfer is in 8-, 16-, 32-, or 64-byte units.</p> <p>B'11: Setting Prohibited</p> |
| 11 to 8 | RS[3:0] | B'0000 | R/W | <p>Resource Selection</p> <p>Specify the source of transfer requests. Only change the transfer request source while the DMA enable bit (DE) is set to 0.</p> <p>B'0100: Auto request B'1000: Source is selected by the DMA extended resource selector. Other than above: Settings prohibited</p> |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|--------|----------|---------------|------------|--|
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4, 3 | TS[1:0] | B'00 | R/W | DMA Transfer Size See the description of TS[3:2] (bits 21 and 20). |
| 2 | IE | B'0 | R/W | Interrupt Enable Specifies whether or not an interrupt request is generated for the CPU on completion of DMA transfer. When this bit is set to 1, an interrupt request (DEI) for the CPU is generated whenever the TE bit is set to 1. 0: Interrupt request is disabled. 1: Interrupt request is enabled. |
| 1 | TE | B'0 | R/(W) * | Transfer End Flag When the descriptor memory is not in use, the TE bit is set to 1 when DMATCR becomes 0 on completion of the DMA transfer. When the descriptor memory is in use, the TE bit is set to 1 on completion of all transfers set up in the descriptor memory. The TE bit is not set to 1 in the following cases. <ul style="list-style-type: none"> • DMA transfer ends due to a DMA address error before DMATCR becomes 0. • DMA transfer is aborted by clearing the DE and DME bits in DMAOR. To clear the TE bit, start by reading it as 1, and then write 0 to it. When the TE bit is set to 1, transfer is not possible even if the DE bit is set to 1. 0: DMA transfer is in progress or was aborted [Clearing condition] Writing of 0 after reading of 1 1: DMA transfer ended on the specified count (TCR = 0) |
| 0 | DE | B'0 | R/W | DMA Enable Enables or disables DMA transfer. In the auto request mode, a DMA transfer is started by setting the DE and DME bits in DMAOR to 1. At this time, the setting of both the AE and TE bits in DMAOR must be 0. In a peripheral module request, a DMA transfer starts if the transfer request is generated by the selected device or on-chip peripheral module after setting the DE and DME bits to 1. In this case too, the settings of both the TE and AE bits must be 0. Clearing the DE bit to 0 aborts all DMA transfer. Note: Ensure that the setting of the DE bit is actually 0 after clearing it. 0: DMA transfer is disabled. 1: DMA transfer is enabled. |

Note: * Writing 0 is possible to clear the flag.

23.2.23 DMA Channel Control Register B_0 to 47 (DMACHCRB_0 to DMACHCRB_47)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMACHCRB is a 32-bit readable/writable register that controls the DMA transfer mode.

| | | | | | | | | | | | | | | | | |
|----------------|-----------|-----|-----|-----|-----|-----|-----|-----|-----------|-----|-----|----------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | DCNT[7:0] | | | | | | | | DPTR[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DRST | — | — | — | — | — | — | DTS | SLM[3:0] | | | PRI[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | W | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|-----------|---------------|-----|--|
| 31 to 24 | DCNT[7:0] | H'00 | R/W | Number of Stages of Descriptor Memory Specify the number of stages of the descriptor memory as DCNT + 1. When the descriptor memory is enabled, a transfer end (TE) interrupt is only generated for the CPU on completion of transfer under control of the specified number of stages. |
| 23 to 16 | DPTR[7:0] | H'00 | R | Descriptor Pointer This bit indicates the pointer to the next descriptor to be read. It is cleared to 0 when the last descriptor of the number of stages specified by DCNT[7:0] is read. It is also cleared to 0 when 1 is written to DRST. |
| 15 | DRST | B'0 | W | Descriptor Reset Resets the descriptor pointer. Before the descriptor memory is used, the pointer must be reset by writing 1 to this bit. This bit is always read as 0. |
| 14 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | DTS | B'0 | R/W | Total Size Transmission under Descriptor Control This bit is only effective when total size transmission is selected. 0: The TCR fields of the descriptors are used as transfer count settings. 1: The TCR fields of the descriptors are used as total size settings. |
| 7 to 4 | SLM[3:0] | B'0000 | R/W | DMA Transfer Low-Speed Mode Specify the number of cycles of AXI-bus clock (266.66 MHz) for the DMA transfer. One round of DMA transfer is executed in the number of cycles of the clock specified by this bit. B'0000: Normal mode B'1000: On round in 256 cycles of the clock. B'1001: On round in 512 cycles of the clock. B'1010: On round in 1024 cycles of the clock. : B'1111: On round in 32768 cycles of the clock. Other than above: Setting prohibited |

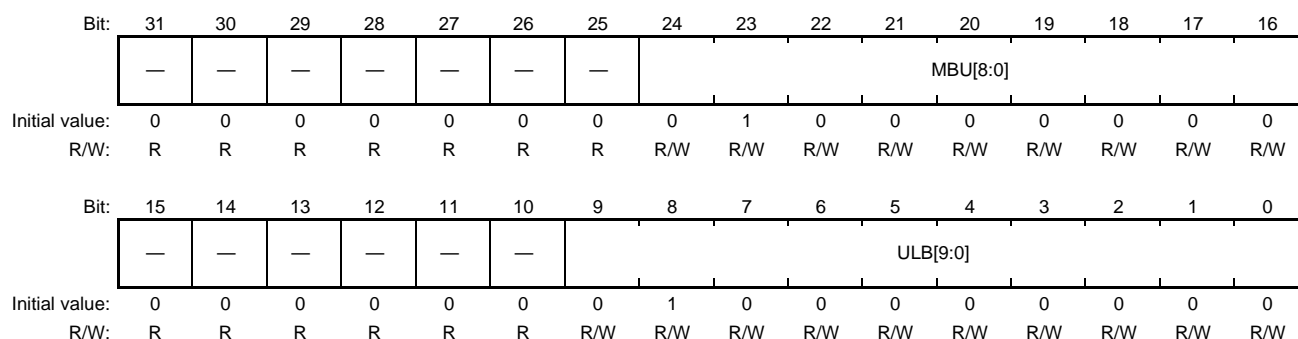
| Bit | Bit Name | Initial Value | R/W | Descriptions |
|--------|----------|---------------|-----|--|
| 3 to 0 | PRI[3:0] | B'0000 | R/W | Channel Request Priority Setting These bits should be written by 0. |

23.2.24 DMA Buffer Control Registers 0 to 47 (DMABUFCR_0 to DMABUFCR_47)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMABUFCR is a 32-bit readable/writable register that controls the upper limit on buffer size in and burst unit for the SDRAM.

Use this register when the upper limit on buffering requires control.



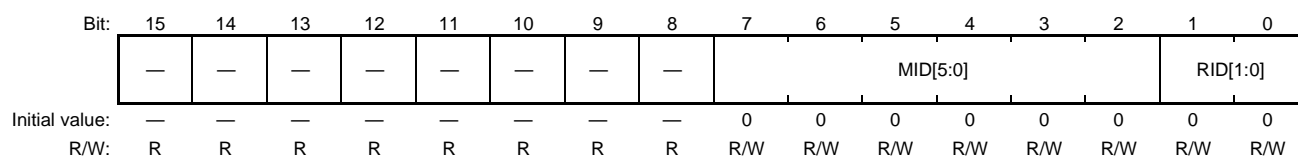
| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|---|
| 31 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 to 16 | MBU[8:0] | H'080 | R/W | Maximum Burst Unit for SDRAM This register is only effective for SDRAM access, and everything other than that is under control of the transfer size (unit). Settings bigger than ULB are prohibited. Power-of-two settings are recommended. Maximum value is 256 (bytes). |
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 to 0 | ULB[9:0] | H'100 | R/W | Upper Limit on Buffer Size This register controls the upper limit value for buffering. Power-of-two settings are recommended. Maximum value is 512 (bytes). |

23.2.25 DMA Extended Resource Selectors 0 to 47 (DMARS_0 to DMARS_47)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMARS is a 16-bit readable/writable register that specifies the on-chip peripheral module to be the source of the DMA transfer request for the given channel. DMARS_0 specifies the source for channel 0, DMARS_1 specifies the source for channel 1 and so on.

When bits MID and RID are set to a value other than the values listed in Table 23.5, the operation of this LSI is not guaranteed. Transfer requests from the source selected in DMARS are only valid when the resource selection bits (RS[3:0]) in DMACHCR have been set to B'1000. Otherwise, even if DMARS has been set, requests from the corresponding transfer request source are not accepted.

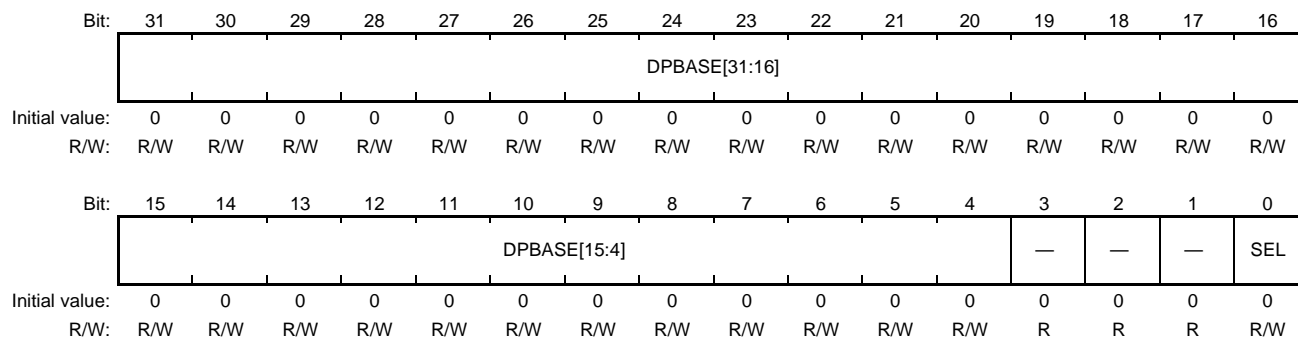


| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|----------|---------------|-----|---|
| 15 to 8 | — | — | R | Reserved These bits are always read as an undefined value. The write value should always be 0. |
| 7 to 2 | MID[5:0] | B'00_0000 | R/W | DMA Request Source Adoption MID[5] to MID[0] (MID) See Table 23.5. |
| 1, 0 | RID[1:0] | B'00 | R/W | DMA Request Source Adoption RID[1] and RID[0] (RID) See Table 23.5. |

23.2.26 DMA Descriptor Base Address Registers 0 to 47 (DMADPBASE_0 to DMADPBASE_47)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMADPBASE specifies the base address of the descriptor memory. The address range of the descriptor memory is specified by setting this register.

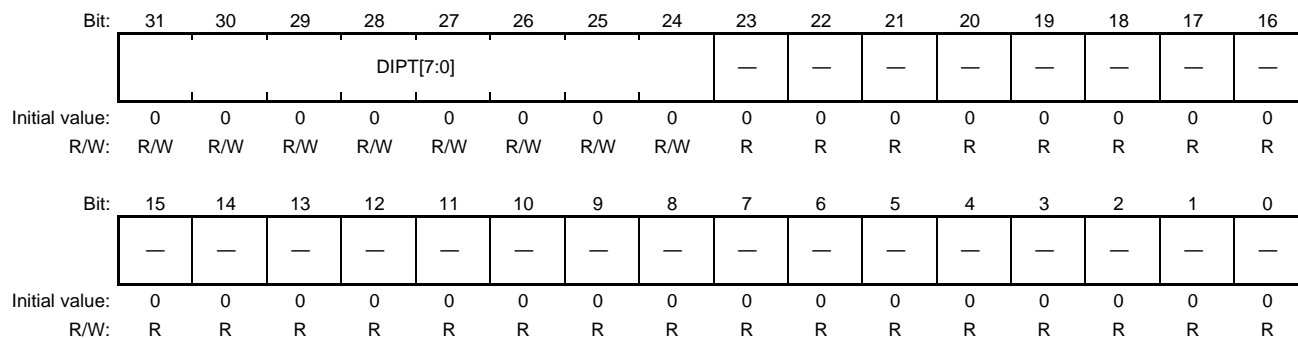


| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|--------------|---------------|-----|---|
| 31 to 4 | DPBASE[31:4] | H'000_0000 | R/W | Base Address of Descriptor Memory Place each stage of the descriptor memory on a 16-byte boundary. Setting example: When Built-in memory is used, [SYS-DMAC0]: H'E67_0A00 to H'E67_0A7F [SYS-DMAC1]: H'E73_0A00 to H'E73_0A7F [SYS-DMAC2]: H'E73_1A00 to H'E73_1A7F When External memory is used, Other memory area on a 16-byte boundary |
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | SEL | B'0 | R/W | Descriptor Memory Selection Selects the memory to be used as descriptor memory. The transfer control registers can be updated at high speed by using the built-in descriptor memory. 0: Setting Prohibited. 1: Built-in memory or External memory is used. |

23.2.27 DMA Descriptor Control Registers 0 to 47 (DMADPCR_0 to DMADPCR_47)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMADPCR is a 32-bit readable/writable register that controls the timing with which interrupts are output in read-out interrupt mode (descriptor mode 3).



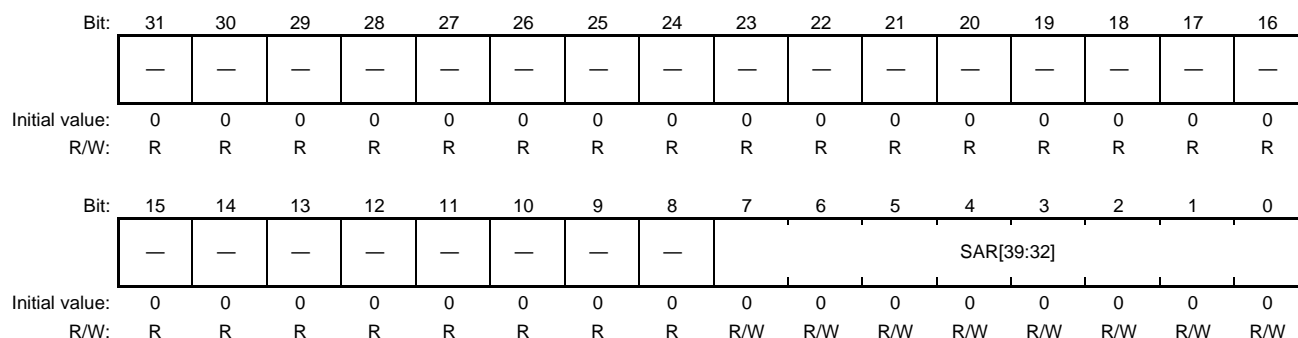
| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|-----------|---------------|-----|--|
| 31 to 24 | DIPT[7:0] | H'00 | R/W | Descriptor Read-out Interrupt Pointer The number of stages for which descriptor read-out interrupts are generated in descriptor mode 3. DIPT + 1 specifies the number of descriptor stages. |
| 23 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

23.2.28 DMA Fixed Source Address Registers 0 to 47 (DMAFIXSAR_0 to DMAFIXSAR_47)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMAFIXSAR is a 32-bit readable/writable register that specifies the most significant 8 bits of the 40-bit source address for a DMA transfer.

This register is not incremented by carrying when DMASAR overflows. And when uTLB function is effective, this register is invalid.



23.2.29 DMA Fixed Destination Address Registers 0 to 47 (DMAFIXDAR_0 to DMAFIXDAR_47)

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

DMAFIXDAR is a 32-bit readable/writable register that specifies the most significant 8 bits of the 40-bit destination address for a DMA transfer.

This register is not incremented by carrying when DMADAR overflows. And when uTLB function is effective, this register is invalid.

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------------|----|----|----|----|----|----|---|---|------------|-----|-----|-----|-----|-----|-----|-----|
| | — | — | — | — | — | — | — | — | DAR[39:32] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

23.2.30 DMA Fixed Descriptor Base Address Registers 0 to 47 (DMAFIXDPBASE_0 to DMAFIXDPBASE_47)

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

DMAFIXDPBASE is a 32-bit readable/writable register that specifies the most significant 8 bits of the 40-bit descriptor base address for a DMA transfer. And when uTLB function is effective, this register is invalid.

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------------|----|----|----|----|----|----|---|---|---------------|-----|-----|-----|-----|-----|-----|-----|
| | — | — | — | — | — | — | — | — | DPBASE[39:32] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

23.2.31 Descriptor Memory (DescriptorMEM)

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

See section 23.3.4, Descriptor Memory.

23.2.32 Secure Function Secure Status Register for channel 0 to 15 (DMASES_0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASES_0 is a 32-bit readable/writable register that contain error status of Secure function of channel 0 to 15.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | Error_0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W* |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|------|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | Error_0 | B'0 | R/W* | Error status of channel 15 to 0 Read: 0: No Error detected. 1: Error detected. Write*: 0: No operation 1: Clear DMASES_0, DMAEMID_0, DMASEA_0 register. |

23.2.33 Secure Function Secure Status Register for channel 16 to 31 (DMASES_1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASES_1 is a 32-bit readable/writable register that contain error status of Secure function of channel 16 to 31.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | Error_1 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W* |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|------|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | Error_1 | B'0 | R/W* | Error status of channel 31 to 16 Read: 0: No Error detected. 1: Error detected. Write*: 0: No operation 1: Clear DMASES_1, DMAEMID_1, DMASEA_1 register. |

23.2.34 Secure Function Secure Status Register for channel 32 to 47 (DMASES_2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASES_2 is a 32-bit readable/writable register that contain error status of Secure function of channel 32 to 47.

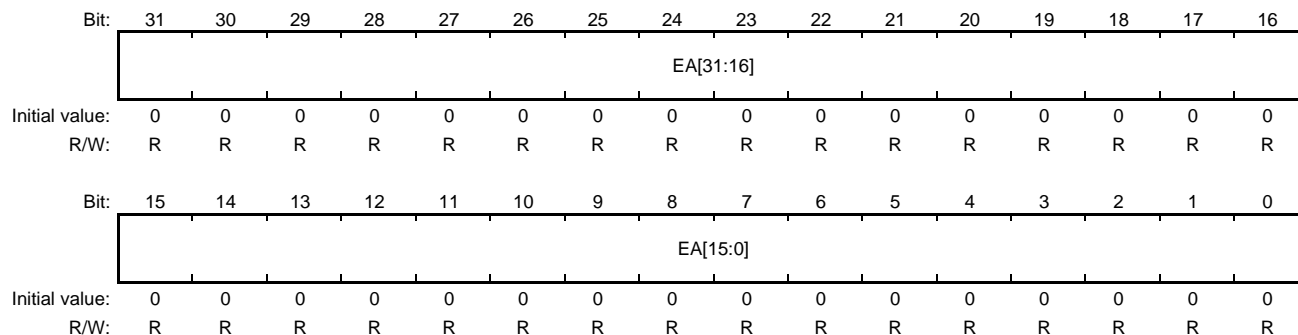
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | Error_2 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W* |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|------|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | Error_2 | B'0 | R/W* | Error status of channel 47 to 32 Read: 0: No Error detected. 1: Error detected. Write*: 0: No operation 1: Clear DMASES_2, DMAEMID_2, DMASEA_2 register. |

23.2.35 Secure Function Slave Error Address Register for channel 0 to 15 (DMASEA_0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASEA_0 is a 32-bit readable register that contain first error address when access to channel 0 to 15 was denied.

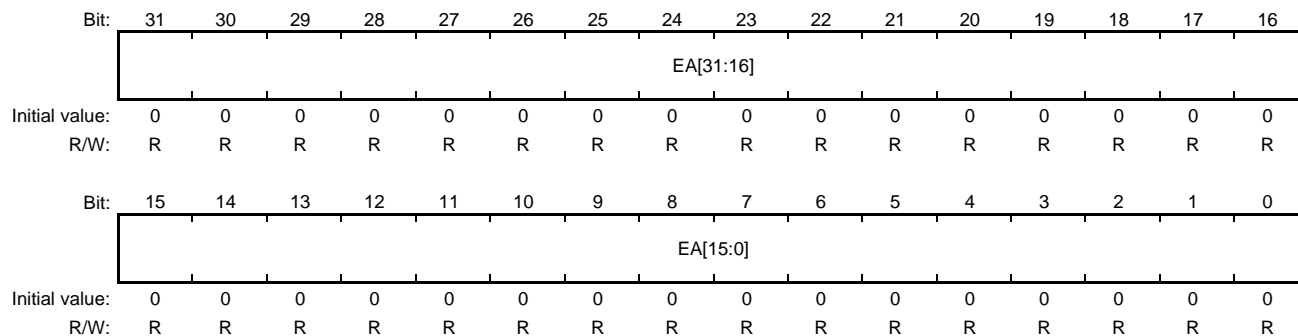


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | EA[31:0] | H'0000_0000 | R | First error address when access to channel 0 to 15 was denied. |

23.2.36 Secure Function Slave Error Address Register for channel 16 to 31 (DMASEA_1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASEA_1 is a 32-bit readable register that contain first error address when access to channel 16 to 31 was denied.

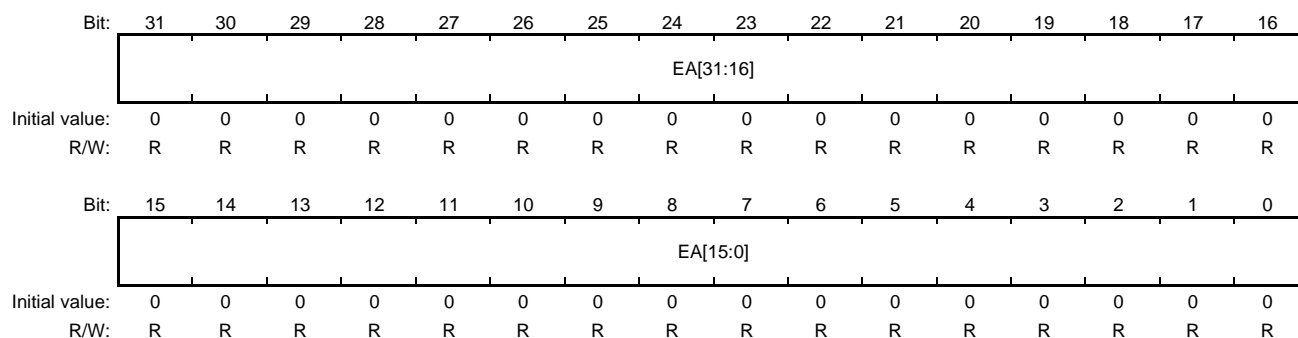


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | EA[31:0] | H'0000_0000 | R | First error address when access to channel 16 to 31 was denied. |

23.2.37 Secure Function Slave Error Address Register for channel 32 to 47 (DMASEA_2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASEA_2 is a 32-bit readable register that contain first error address when access to channel 32 to 47 was denied.

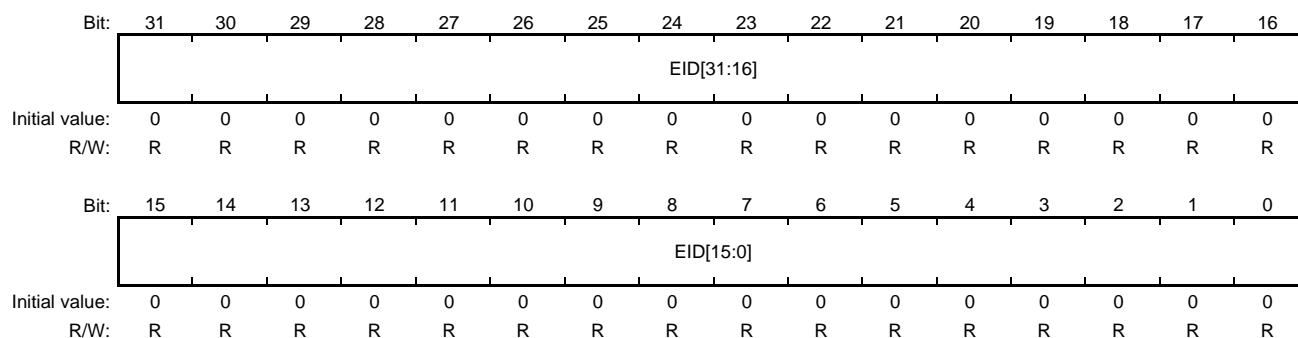


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | EA[31:0] | H'0000_0000 | R | First error address when access to channel 32 to 47 was denied. |

23.2.38 Secure Function Error Master ID Register for channel 0 to 15 (DMAEMID_0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMAEMID_0 is a 32-bit readable register that contain first error ID when access to channel 0 to 15 was denied.

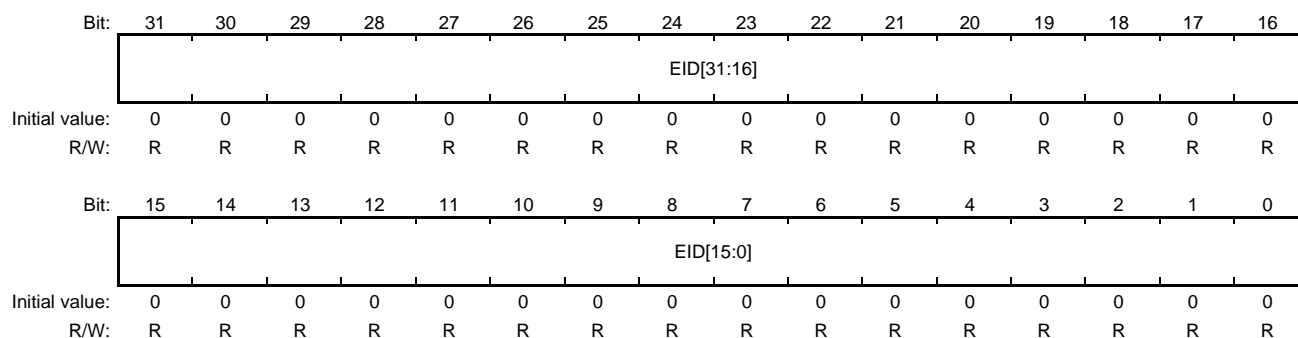


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | EID[31:0] | H'0000_0000 | R | First error ID when access to channel 0 to 15 was denied. |

23.2.39 Secure Function Error Master ID Register for channel 16 to 31 (DMAEMID_1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMAEMID_1 is a 32-bit readable register that contain first error ID when access to channel 16 to 31 was denied.

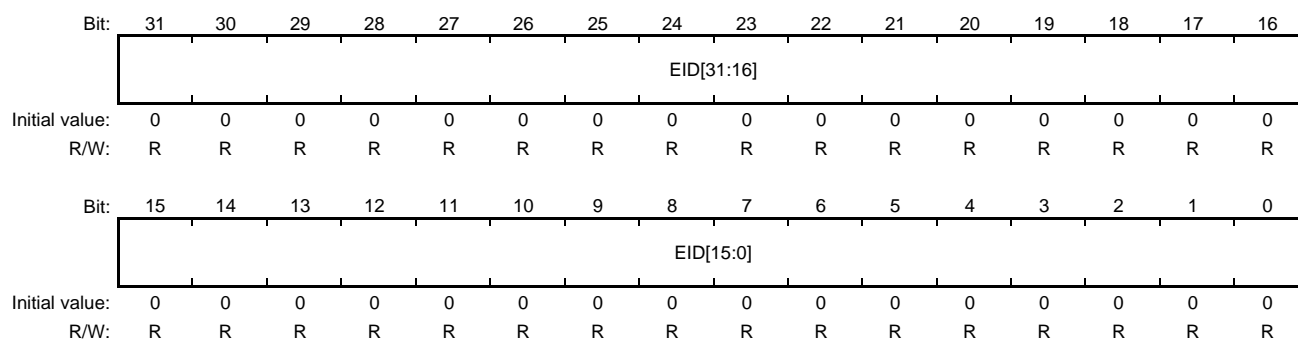


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 0 | EID[31:0] | H'0000_0000 | R | First error ID when access to channel 16 to 31 was denied. |

23.2.40 Secure Function Error Master ID Register for channel 32 to 47 (DMAEMID_2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMAEMID_2 is a 32-bit readable register that contain first error ID when access to channel 32 to 47 was denied.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 0 | EID[31:0] | H'0000_0000 | R | First error ID when access to channel 32 to 47 was denied. |

23.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

When there is a DMA transfer request, the DMAC starts the transfer according to the predetermined channel priority; when the transfer end conditions are satisfied, it ends the transfer. Transfers can be requested in two modes: auto request and on-chip peripheral module request. The bus mode can be selected from normal speed mode and slow speed mode).

23.3.1 DMA Transfer Requests

Most commonly, DMA transfer requests are generated by either the source or destination for transfer, but they can also be generated by on-chip peripheral modules that are neither the source nor the destination. Transfers can be requested in two modes: auto request, and on-chip peripheral module request. The request mode is selected for each channel by DMARS.

(1) Auto-Request Mode

When there is no transfer request signal from an external source, as in a memory-to-memory transfer or a transfer between memory and an on-chip peripheral module unable to request a transfer, auto-request mode allows the SYS-DMAC to automatically generate a transfer request signal internally. When the DE bit in DMACHCR and the DME bit in DMAOR are set to 1 for the target channel, the transfer begins so long as the CAE bit in DMACHCR is 0.

(2) On-Chip Peripheral Module Request Mode

This mode is in case of DMACHCR.RS=1000.

In this mode, a transfer is performed at the transfer request signal of an on-chip peripheral module. The source (on-chip peripheral module) of the DMA transfer request is specified by DMARS.

When this mode is selected, if the DMA transfer is enabled (DE = 1, DME = 1, TE = 0, CAE = 0), a transfer is performed upon the input of a transfer request signal.

When a transmit data empty transfer request of the peripheral is set as the transfer request, the transfer destination must be the peripheral transmit data register. Likewise, when receive data full transfer request of the peripheral is set as the transfer request, the transfer source must be the peripheral receive data register. These conditions also apply to the other on-chip peripheral modules.

The number of the receive FIFO triggers can be set as a transfer request depending on an on-chip peripheral module. Data needs to be read after the DMA transfer is ended, because data may be left in the receive FIFO when the receive FIFO trigger condition is not satisfied.

Table 23.5 Selecting On-Chip Peripheral Module Request Modes

| | | | | | | Second Generation RZ/G Series Products | | | | |
|------------------|-----------|-----------------------------|-----------------------------|---------------|---------------|---|----------------------------|--------|--------|--|
| DMARS | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Channels Support | MID + RID | DMA Transfer Request Source | DMA Transfer Request Signal | Source | Destination | | | | | |
| 0 to 15 | H'10 | IIC-PMIC Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'11 | IIC-PMIC Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 16 to 47 | H'17 | TPU0 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 16 to 47 | H'30 | HSCIF0 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'31 | HSCIF0 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 16 to 47 | H'32 | HSCIF1 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'33 | HSCIF1 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 16 to 47 | H'34 | HSCIF2 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'35 | HSCIF2 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'36 | HSCIF3 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'37 | HSCIF3 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'38 | HSCIF4 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'39 | HSCIF4 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 16 to 47 | H'40 | MSIOF0 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'41 | MSIOF0 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 16 to 47 | H'42 | MSIOF1 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'43 | MSIOF1 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'44 | MSIOF2 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'45 | MSIOF2 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'46 | MSIOF3 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'47 | MSIOF3 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 16 to 47 | H'50 | SCIF0 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'51 | SCIF0 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 16 to 47 | H'52 | SCIF1 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'53 | SCIF1 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 16 to 47 | H'12 | SCIF2 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'13 | SCIF2 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'56 | SCIF3 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'57 | SCIF3 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'58 | SCIF4 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'59 | SCIF4 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'5A | SCIF5 Reception | Receive request | Data register | Arbitrary | — | — | — | √ | |
| 0 to 15 | H'5B | SCIF5 Transmission | Transmit request | Arbitrary | Data register | — | — | — | √ | |
| 16 to 47 | H'5A | SCIF5 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | — | |
| 16 to 47 | H'5B | SCIF5 Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | — | |
| 0 to 15 | H'70 | CANFD_RF0 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |

| Channels Support | DMARS MID + RID | DMA Transfer Request Source | DMA Transfer Request Signal | Source | Destination | Second Generation RZ/G Series Products | | | | |
|------------------|--------------------|-----------------------------|-----------------------------|---------------|---------------|---|----------------------------|--------|--------|--|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| 0 to 15 | H'72 | CANFD_RF1 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'74 | CANFD_RF2 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'76 | CANFD_RF3 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'78 | CANFD_RF4 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'7A | CANFD_RF5 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'7C | CANFD_RF6 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'7E | CANFD_RF7 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'80 | CANFD_CF0 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'82 | CANFD_CF1 Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'90 | I2C0(Master) Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'91 | I2C0(Master) Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 16 to 47 | H'92 | I2C1(Master) Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'93 | I2C1(Master) Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 16 to 47 | H'94 | I2C2(Master) Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'95 | I2C2(Master) Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'96 | I2C3(Master) Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'97 | I2C3(Master) Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'98 | I2C4(Master) Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'99 | I2C4(Master) Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'9A | I2C5(Master) Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'9B | I2C5(Master) Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'9C | I2C6(Master) Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'9D | I2C6(Master) Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |

| | | | | | | Second Generation RZ/G Series Products | | | | |
|------------------|-----------|-----------------------------|-----------------------------|---------------|---------------|---|----------------------------|--------|--------|--|
| DMARS | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Channels Support | MID + RID | DMA Transfer Request Source | DMA Transfer Request Signal | Source | Destination | | | | | |
| 16 to 47 | H'A0 | I2C0(Slave) Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'A1 | I2C0(Slave) Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 16 to 47 | H'A2 | I2C1(Slave) Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'A3 | I2C1(Slave) Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 16 to 47 | H'A4 | I2C2(Slave) Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 16 to 47 | H'A5 | I2C2(Slave) Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'A6 | I2C3(Slave) Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'A7 | I2C3(Slave) Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'A8 | I2C4(Slave) Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'A9 | I2C4(Slave) Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'AA | I2C5(Slave) Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'AB | I2C5(Slave) Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |
| 0 to 15 | H'AC | I2C6(Slave) Reception | Receive request | Data register | Arbitrary | √ | √ | √ | √ | |
| 0 to 15 | H'AD | I2C6(Slave) Transmission | Transmit request | Arbitrary | Data register | √ | √ | √ | √ | |

Table 23.6 Data Length of DMA Transfer for Each of the On-Chip Peripheral Modules

| Module* | 1 Byte | 2 Bytes | 4 Bytes | 8 Bytes | 16 Bytes | 32 Bytes | 64 Bytes | Second Generation RZ/G Series Products | | | | |
|------------------|--------|---------|---------|---------|----------|----------|----------|---|----------------------------|--------|--------|--|
| | | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| IIC-PMIC | √ | | | | | | | √ | √ | √ | √ | |
| TPU0 | | √ | | | | | | √ | √ | √ | √ | |
| HSCIF0/1/2/3/4 | √ | | | | | | | √ | √ | √ | √ | |
| MSIOF0/1/2/3 | | | √ | | | | | √ | √ | √ | √ | |
| SCIF0/1/2/3/4/5 | √ | | | | | | | √ | √ | √ | √ | |
| CANFD | | | | | | √ | | √ | √ | √ | √ | |
| I2C0/1/2/3/4/5/6 | √ | | | | | | | √ | √ | √ | √ | |

Note: * For the availability of module channels of each product, refer to Table 23.5.

23.3.2 Channel Priority

When the SYS-DMAC receives simultaneous transfer requests on two or more channels, it transfers data according to a predetermined priority. Two modes (fixed mode and round-robin mode) are selected by the PR[1:0] bits in DMAOR.

Fixed Mode

In this mode, the priority levels among the channels remain fixed.

CH0 > CH1 > ... > CH14 > CH15, CH16 > CH17 > ... > C30 > C31, C32 > C 33 > ... > CH46 > CH47

Round-Robin Mode

In round-robin mode, each time data of one transfer unit (byte, word, longword, 8-byte, or 16-byte units) is transferred on one channel, the priority is rotated. The channel on which the transfer was just finished rotates to the bottom of the priority. The priority of round-robin mode is CH0 > CH1 > ... > CH14 > CH15, CH16 > CH17 > ... > C30 > C31, C32 > C33 > ... > CH46 > CH47 immediately after reset.

23.3.3 Slow Speed Mode

In the slow-speed mode, a single round of DMA transfer is performed every time the number of clock cycles specified by the SLM bits in DMACHCRB elapse. This mode can be selected per DMA channel. Transfer on other channels can proceed after each round of transfer for a channel in the slow-speed mode is completed.

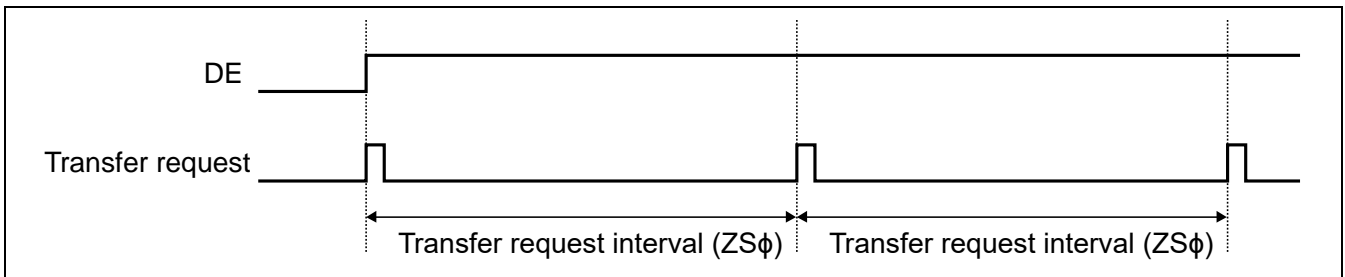


Figure 23.2 Slow Speed Mode

23.3.4 Descriptor Memory

The descriptor memory function is selected by setting the DPM[1:0] bits in DMACHCR to B'01, B'10, or B'11. When DMATCR is set to 0 and the DMA transfer is completed, the next set of settings is read, and if only a single channel is enabled, the contents defined by up to 128 stages of descriptor memory can be consecutively transferred when the built-in descriptor memory is used. External memory can also be used as the descriptor memory. In that case, the contents defined by up to 256 stages of descriptor memory can be transferred.

The following initial settings are required to use the descriptor memory.

- Set the base address of the descriptor memory for the DMA transfer in DMADPBASE.
- Set the DRST bit in DMACHCRB to reset the descriptor memory.
- Set the number of stages of the descriptor memory in the DCNT bits of DMACHCRB.

The descriptor memory is shared between all channels. Ensure that the areas of descriptor memory for use by each of the channels do not overlap.

It is necessary to arrange each stage of the descriptor memory on a 16-byte boundary.

There are two methods to activate the descriptor memory as follows.

- Specify the first DMA transfer settings in DMASAR, DMADAR, and DMATCR, and specify the subsequent settings in the descriptor memory. Then, set the DPB bit in DMACHCR to 0 to activate the descriptor memory. In this case, after completion of the transfer specified in DMASAR, DMADAR, and DMATCR, transfer continues after new settings are read from the descriptor memory. Note, however, that when the operating mode of the descriptor memory is set to the repeat mode, the values specified in DMASAR, DMADAR, and DMATCR are not read, and the transfer starts and is repeated from the head of the descriptor memory.
- Write the DMA transfer settings to the descriptor memory, and write 1 to the DPB bit in DMACHCR to activate the descriptor memory. In this case, the DMA transfer starts from the first settings in the descriptor memory.

There are three operating modes of the descriptor memory, which can be selected by setting the DPM bits in DMACHCR.

For details on these operating modes, see the descriptions of each operating mode in this section.

(1) Configuration of Descriptor Memory

Figure 23.3 shows the configuration of the built-in descriptor memory.

The capacity of the built-in descriptor memory is 16 bytes per stage × 128 stages.

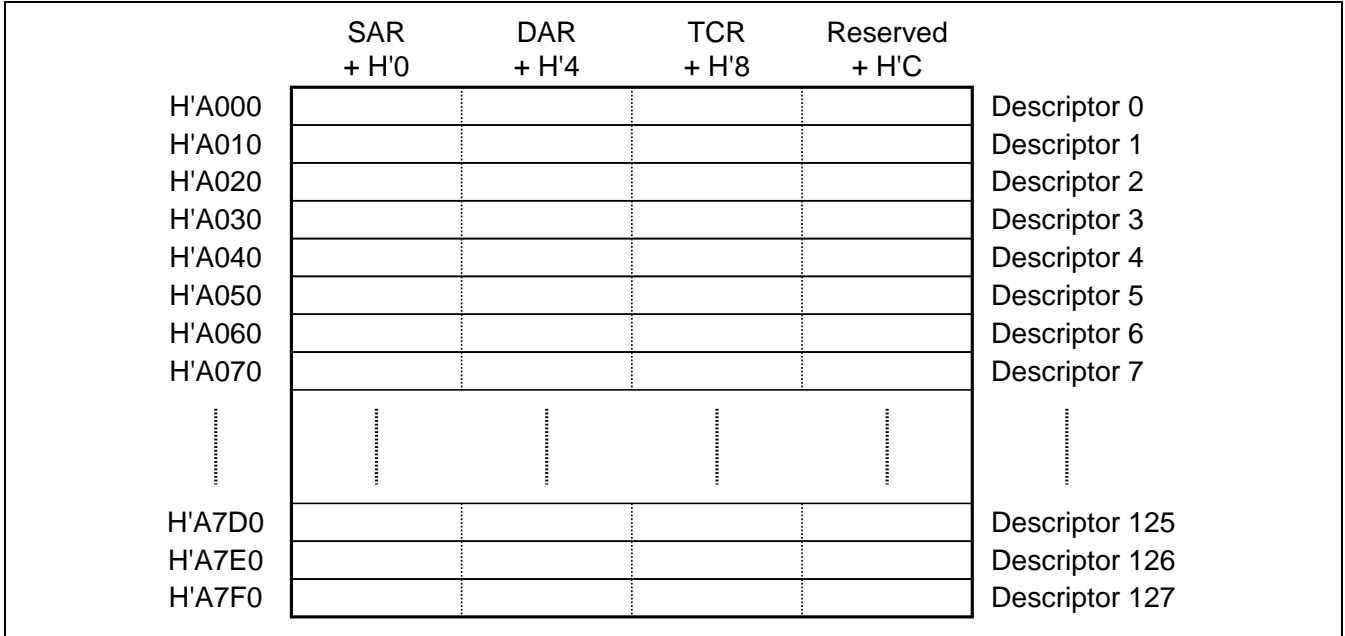


Figure 23.3 Configuration of Built-in Descriptor Memory

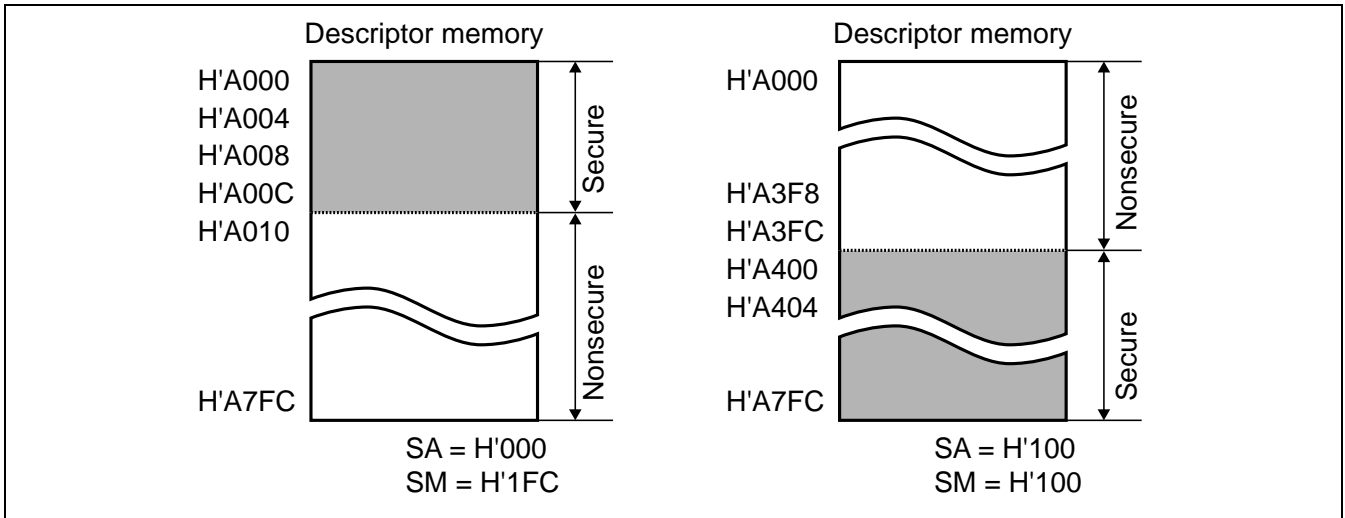


Figure 23.4 Example of DMADPSEC Setting

(2) Flow of Updating from Descriptor Memory

The RPT bits in DMACHCR can be used to specify which registers are to be updated from the descriptor memory.

The DPTR bits in DMACHCRB are incremented when updating from the descriptor memory is completed. If the DPTR value matches the DCNT value, the DPTR value is reset to 0.

This flow is automatically processed by hardware.

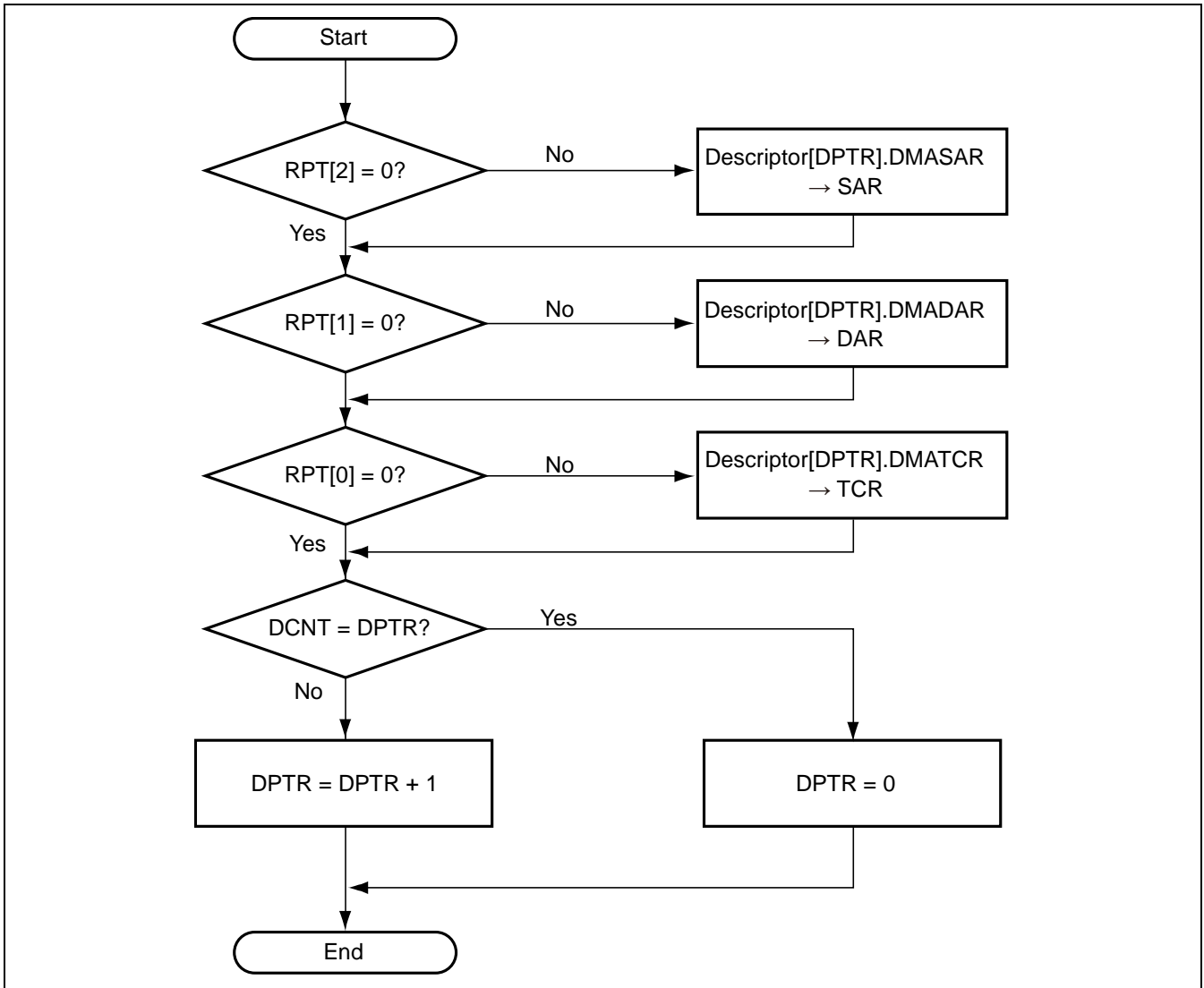


Figure 23.5 Flow of Updating from Descriptor Memory

(3) Operating Mode 1 of Descriptor Memory

Set the DPM bits in DMACHCR to B'01 to select operating mode 1 (normal mode). This mode allows consecutive transfer under control of the descriptor memory.

In this mode, the DMA transfer is complete when the TE bit in DMACHCR is set to 1 after transfer under control of the number of stages of the descriptor memory specified in the DCNT bits in DMACHCRB.

When the DSIE bit in DMACHCR is set to 1, a DSE interrupt is generated every time transfer under the control of one stage of descriptor memory is complete. If a first DSE interrupt has not been processed when a further DSE interrupt is generated, the contents of the next stage of the descriptor memory are not read and the DMA transfer is aborted. Clearing the DSE bit only causes clearing of the first interrupt and reading of the next stage of the descriptor memory.

Figure 23.6 is an example of transfer when operating mode 1 is selected and the TE and DSE bits are set to 1.

Figure 23.7 is an example of transfer when there is a delay in processing of the DSE interrupt when operating mode 1 is selected and the TE and DSE bits are set to 1.

Figure 23.8 is an example of transfer when operating mode 1 is selected and the TE bit is set to 1.

In each example, there are four descriptor stages.

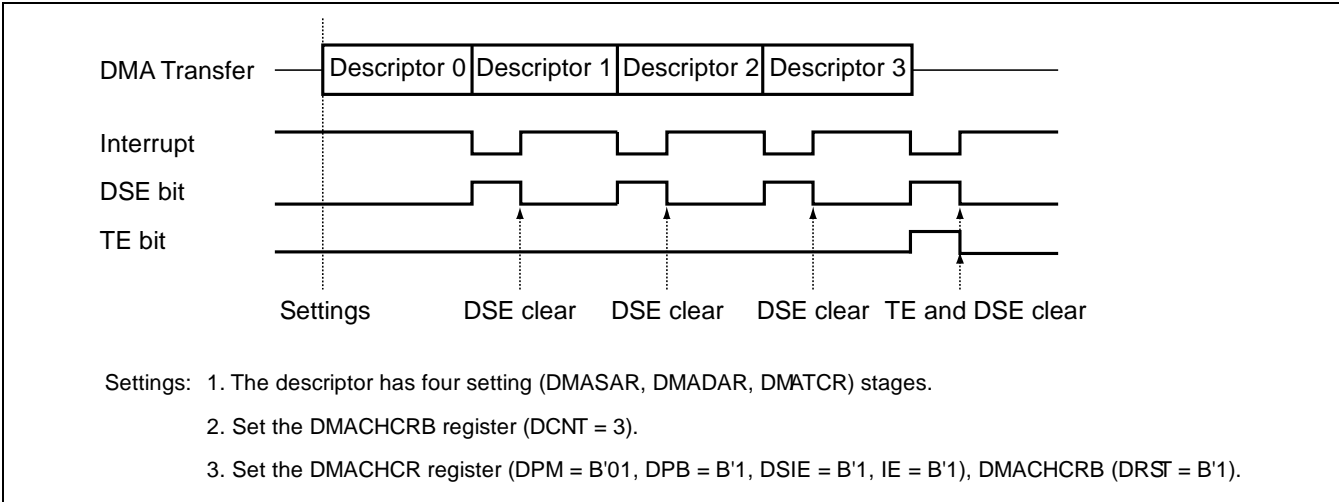


Figure 23.6 Operating Mode 1 (Example 1)

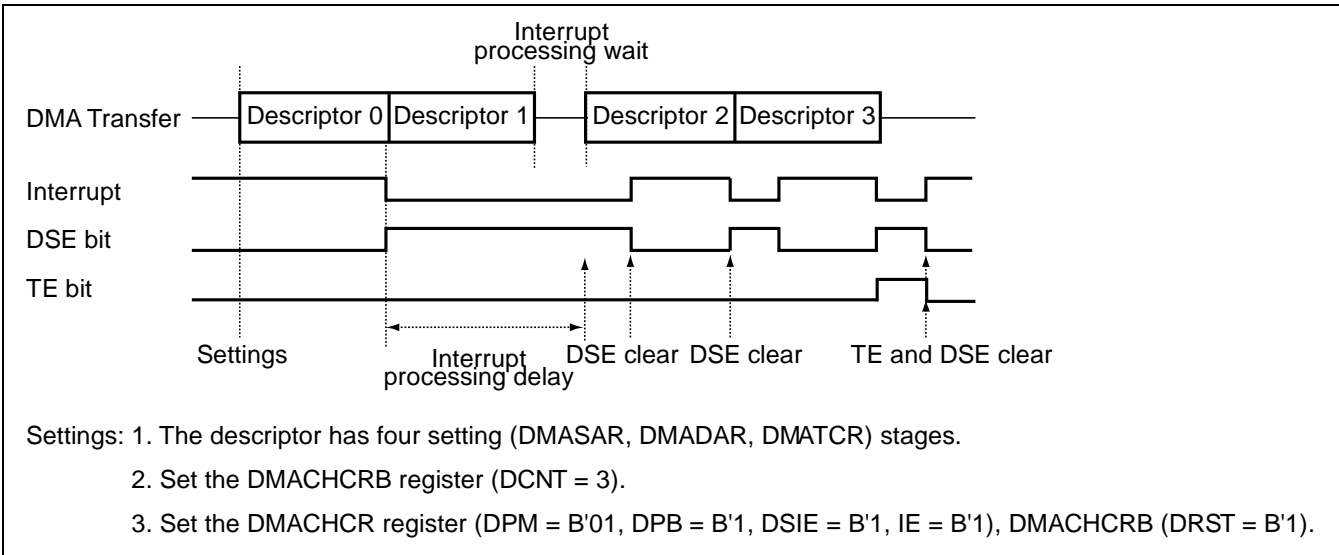


Figure 23.7 Operating Mode 1 (Example 2)

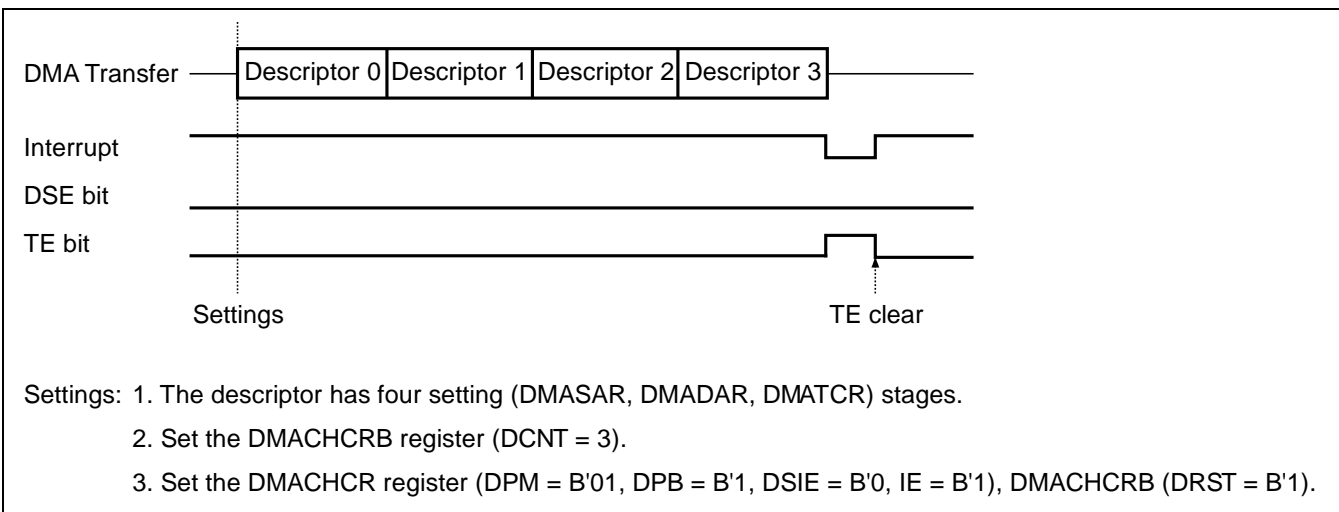


Figure 23.8 Operating Mode 1 (Example 3)

(4) Operating Mode 2 of Descriptor Memory

Set the DPM bits in DMACHCR to B'10 to select operating mode 2 (repeat mode). This mode allows consecutive transfer under control of the descriptor memory.

In this mode, after transfer under control of the number of stages of descriptor memory specified in the DCNT bits in DMACHCRB, the TE bit in DMACHCR is set to 1. This operation is then repeated from the head of the descriptor memory.

When the DSIE bit in DMACHCR is set to 1, a DSE interrupt is generated every time transfer under the control of one stage of descriptor memory is complete. If a first DSE interrupt has not been processed when a further DSE interrupt is generated, the contents of the next stage of the descriptor memory are not read and the DMA transfer is aborted. Clearing the DSE bit only causes clearing of the first interrupt and reading of the next stage of the descriptor memory.

When the DSIE bit in DMACHCR is set to 0, after transfer under control of all stages of the descriptor memory is complete, the TE bit in DMACHCR is set to 1 and a TE interrupt is generated. If a TE interrupt has not been processed when a further DSE interrupt is generated, the contents of the next stage of the descriptor memory are not read and the DMA transfer is aborted. Clearing the TE bit only causes clearing of the first interrupt and reading of the next stage of the descriptor memory.

To end operation in mode 2, change the mode to mode 1 by using the TE interrupt processing. When the mode is changed to mode 1, the DMA transfer is completed when the next TE interrupt is generated.

Figure 23.9 is an example of transfer when operating mode 2 is selected and the TE and DSE bits are set to 1.

Figure 23.10 is an example of transfer when there is a delay in processing of the DSE interrupt when operating mode 2 is selected and the TE and DSE bits are set to 1.

Figure 23.11 is an example of transfer when operating mode 2 is selected and the TE bit is set to 1.

Figure 23.12 is an example of transfer when there is a delay in processing of the DSE interrupt when operating mode 2 is selected and the TE is set to 1.

In each example, there are four descriptor stages.

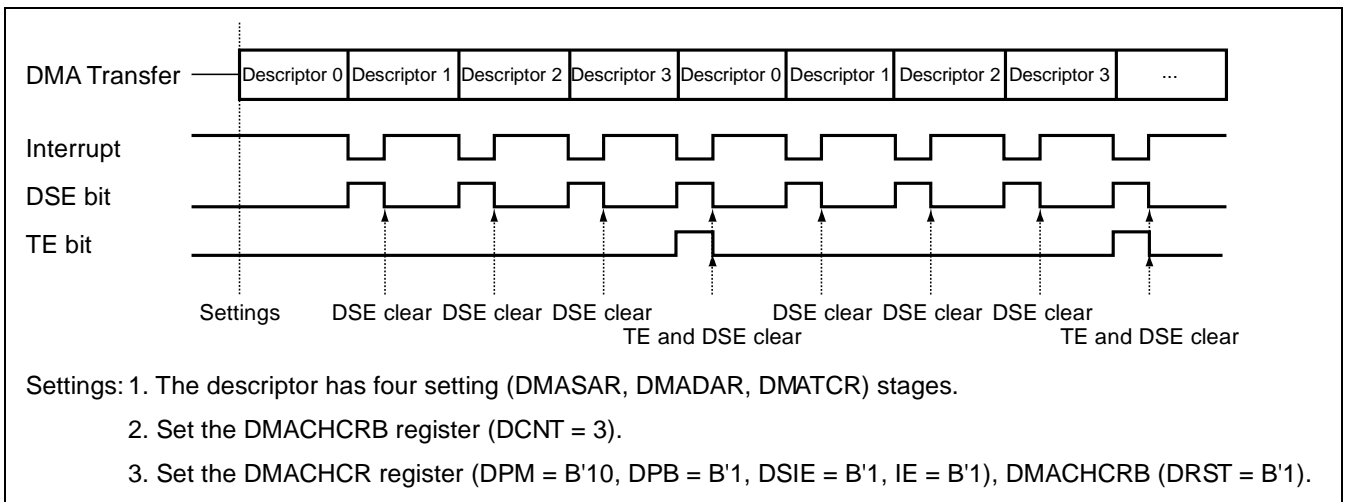


Figure 23.9 Operating Mode 2 (Example 1)

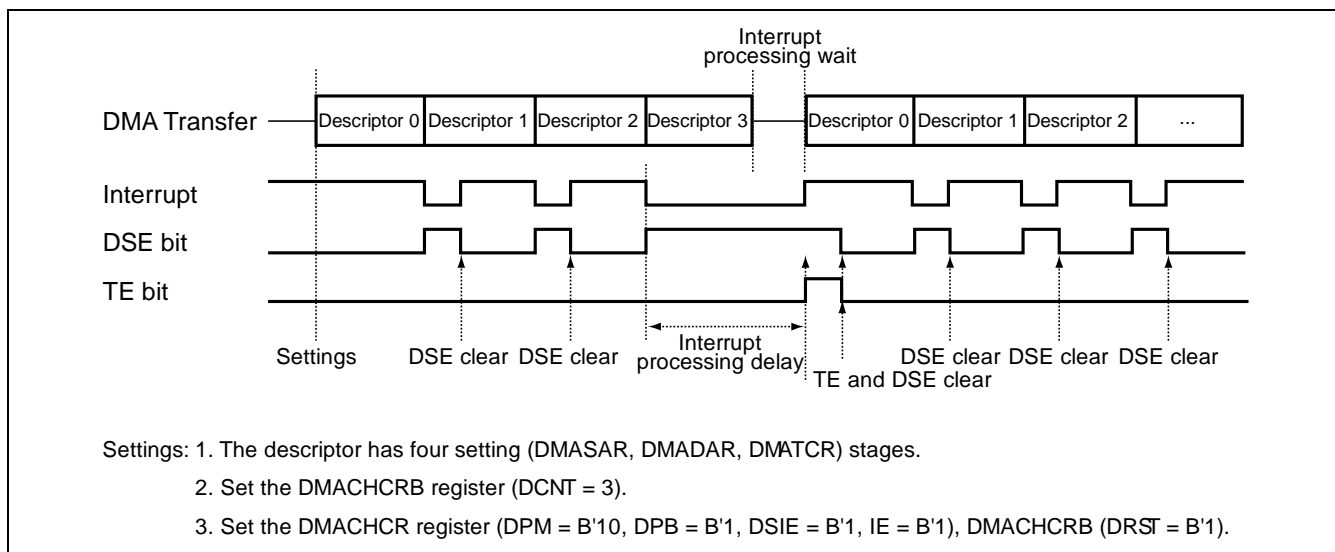


Figure 23.10 Operating Mode 2 (Example 2)

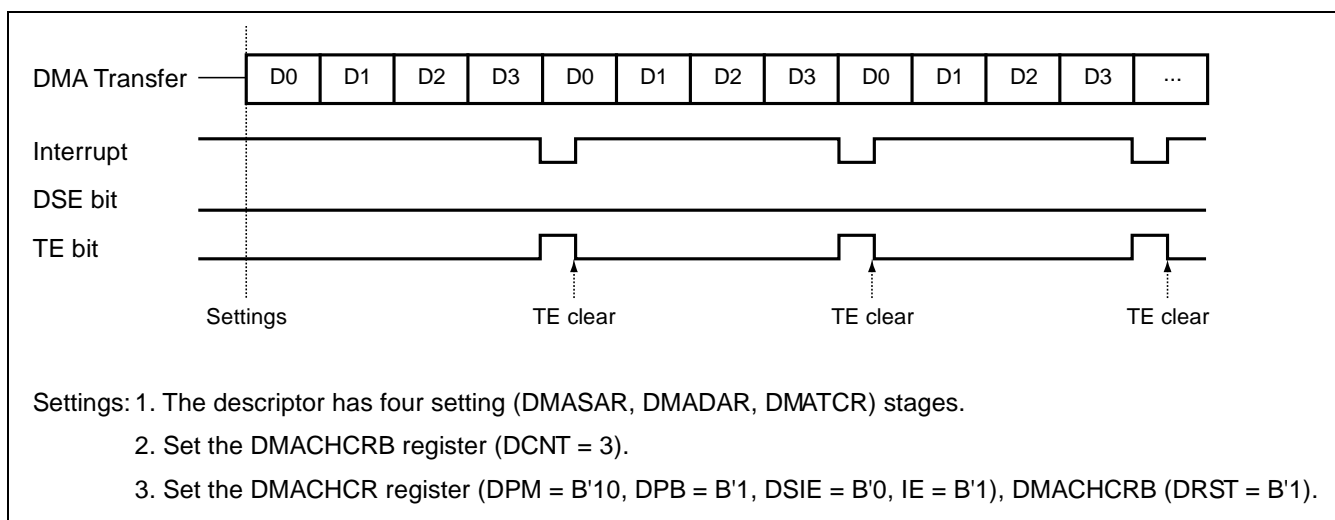


Figure 23.11 Operating Mode 2 (Example 3)

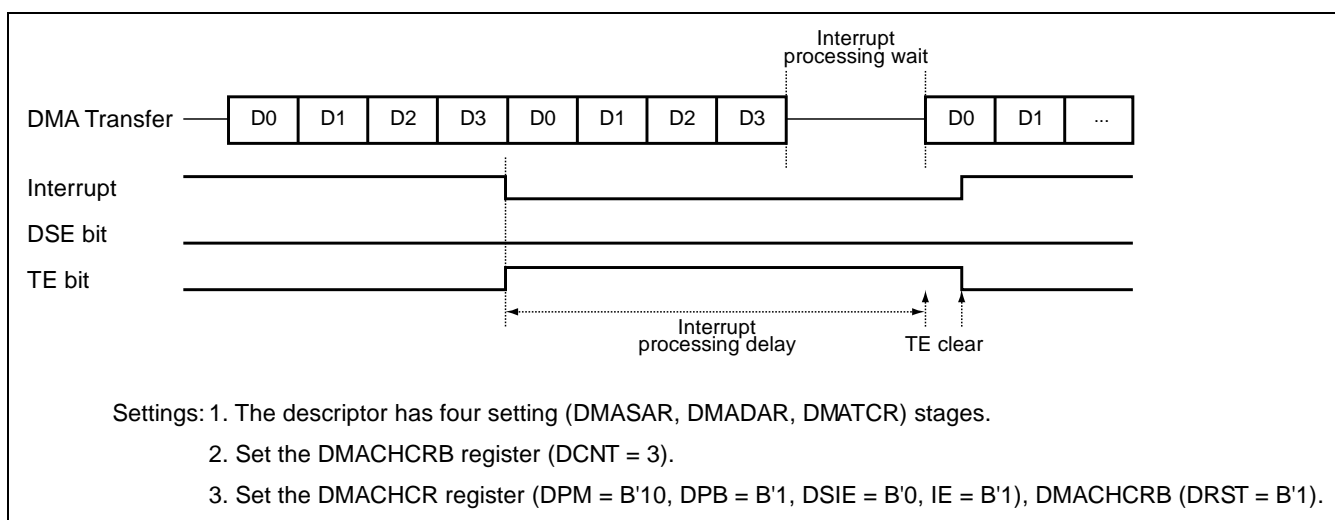


Figure 23.12 Operating Mode 2 (Example 4)

(5) Operating Mode 3 of Descriptor Memory

Set the DPM bits in DMACHCR to B'11 to select operating mode 3 (infinite repeat mode). This mode allows consecutive transfer under control of the descriptor memory.

In this mode, after transfer under control of the number of stages of descriptor memory specified in the DCNT bits in DMACHCRB, the TE bit in DMACHCR is set to 1. This operation is then repeated from the head of the descriptor memory.

When the DSIE bit in DMACHCR is set to 1, a DSE interrupt is generated every time transfer under the control of one stage of descriptor memory is complete. Even if a first DSE interrupt has not been processed when a further DSE interrupt is generated, the DMA transfer is not aborted. Regardless of the number of DSE interrupts that have been generated, the TE bit can be cleared by writing to it once. Similarly, even if a first TE interrupt has not been processed when a further TE interrupt is generated, the DMA transfer is not aborted. Regardless of the number of TE interrupts that have been generated, the TE bit can be cleared by writing to it once.

Figure 23.13 is an example of transfer when infinite repeat mode is selected.

Figure 23.14 is an example of transfer when read-out interrupt mode is selected.

In each example, there are four descriptor stages.

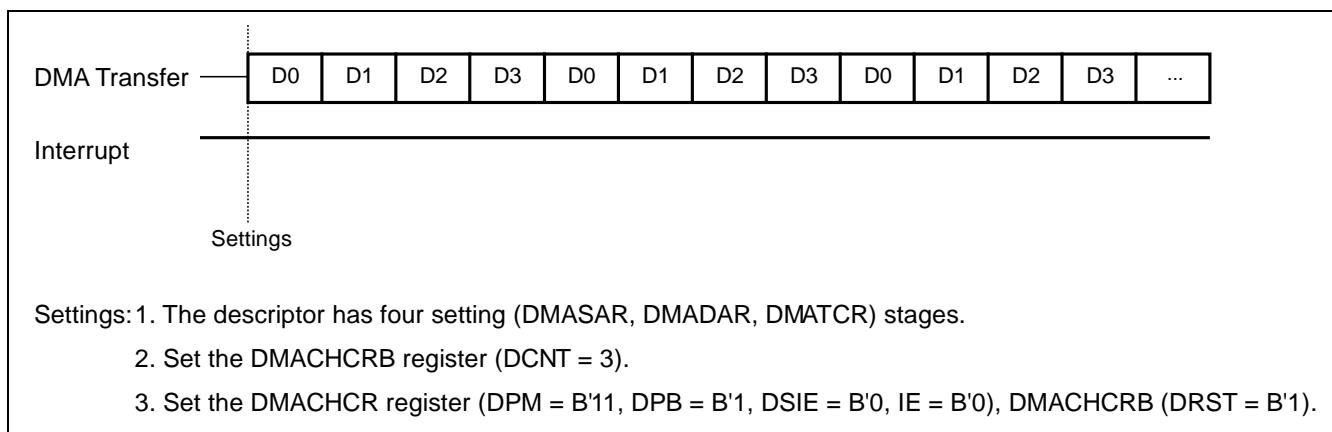


Figure 23.13 Operating Mode 3 (Infinite Repeat Mode)

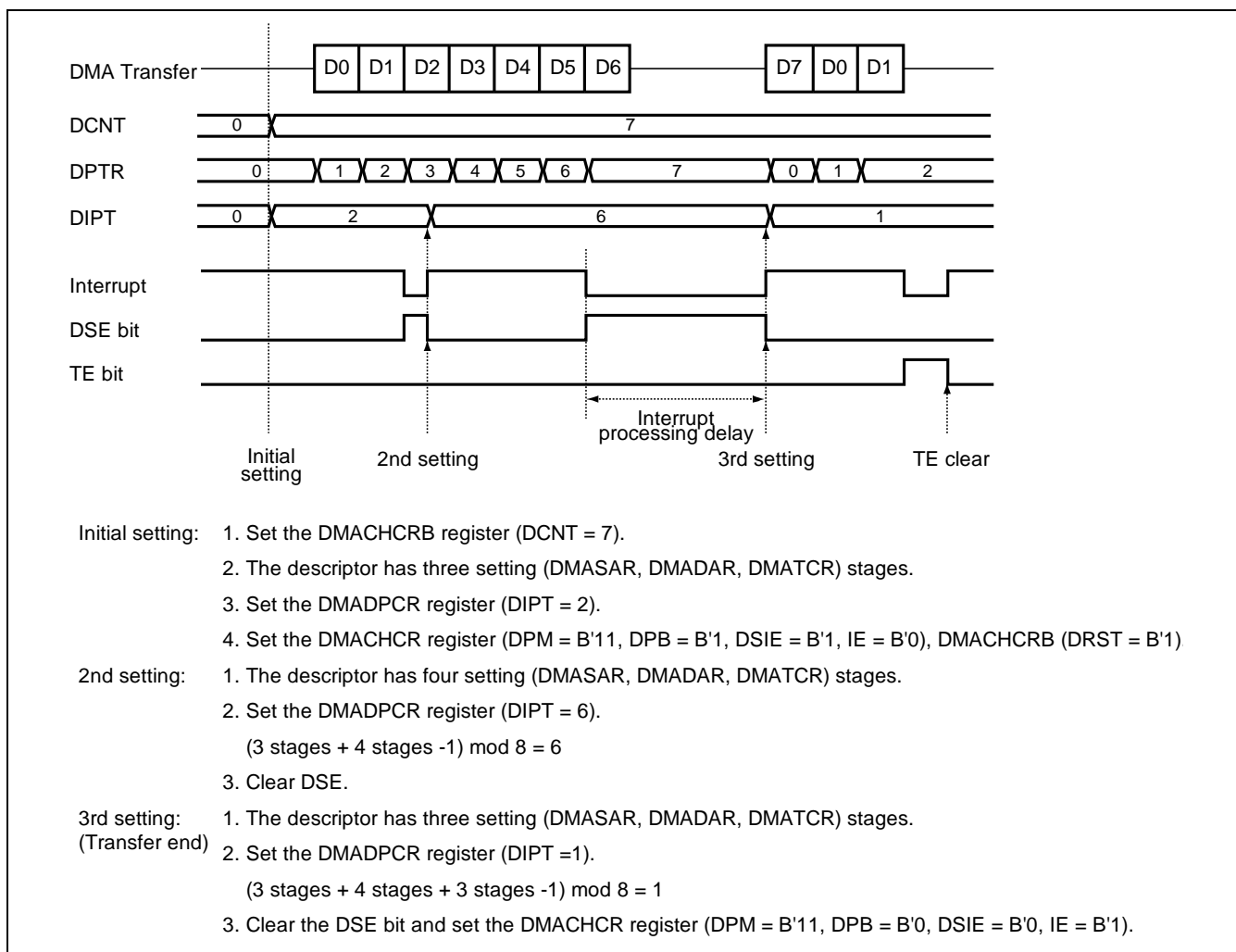


Figure 23.14 Operating Mode 3 (Read-Out Interrupt Mode)

(6) Using the Descriptor Memory for Double Buffering

To use the descriptor memory for double buffering, set the number of stages of descriptor memory to 2, set the buffer configuration to the descriptor memory, and activate the memory in operating mode 2. The DSE interrupt is used in double buffering. To end the use of double buffering, disable the descriptor operating mode, which stops the transfer on completion of the transfer currently in progress.

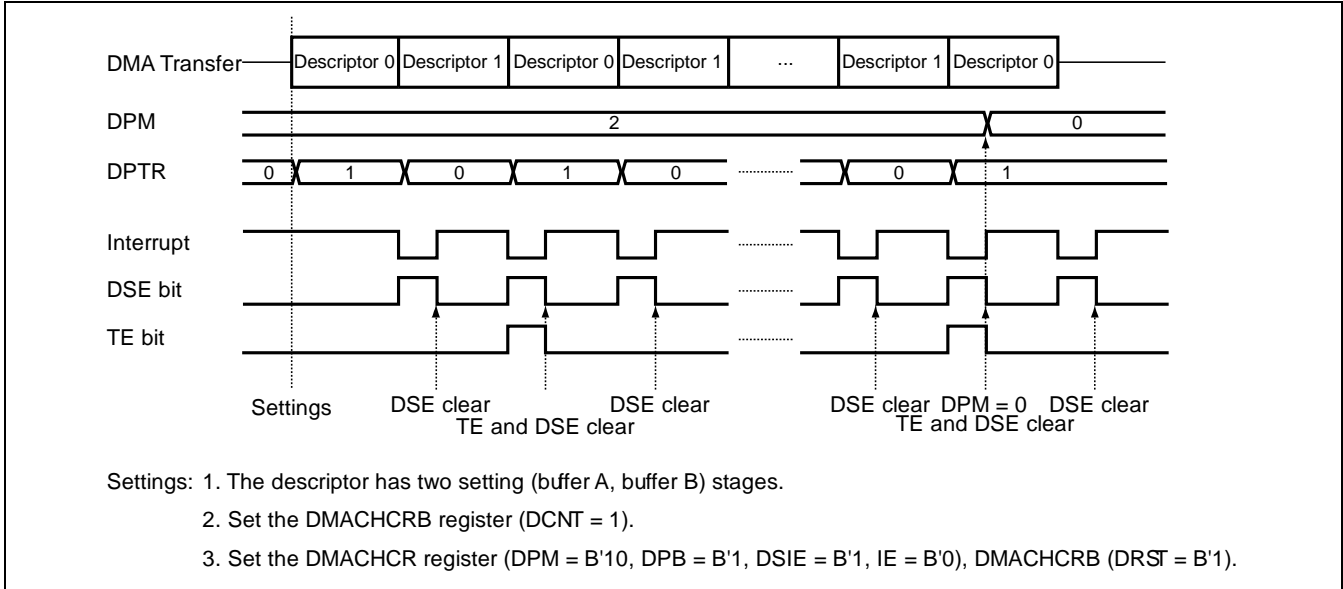


Figure 23.15 Using the Descriptor Memory for Double Buffering

23.3.5 Transmission Flow

Set the transfer conditions as required in the following registers:

DMA source address register (DMASAR), DMA destination address register (DMADAR), DMA transfer count register (DMATCR), DMA Channel control register (DMACHCR), DMA operation register (DMAOR) and DMA extended resource selector (DMARS)

The DMAC then transmits data in the following order.

- A transfer request is generated and the controller checks whether the transfer is allowed (DE = 1, DME = 1, TE = 0, DSE = 0, CAE = 0). When a channel is in the auto request mode, the transfer starts automatically.
- The controller checks whether updating from the descriptor memory is required.
- Updating from the descriptor memory proceeds when the DPB bit in DMACHCR is set to 1 or the descriptor memory is enabled, if DMATCR is set to 0.
- For updating by using the descriptor memory, see section 23.3.4, Descriptor Memory.
- Check whether address translation by the IPMMU is required.
- Address translation by the IPMMU proceeds if the address exceeds the effective size for address translation when the DE bit in DMACHCR is enabled and updating of transfer settings from the descriptor memory is executed.
- Each time a transfer request is generated, the amount of data for a single round of transfer (specified by the TS[3:0] bits) is transmitted. The value in DMATCR is decremented by 1 every time the DMA transfer is completed.
- When the specified number of rounds of transfer are completed (the value in DMATCR is set to 0), the transfer ends normally. A TE interrupt is generated for the CPU upon the end of the transfer if the IE bit in DMACHCR is set to 1. If the descriptor memory is enabled, the processing differs with the mode of the descriptor memory. For more details, see section 23.3.4, Descriptor Memory.
- Transfer is aborted when the DMAC encounters an address error. Transfer is also aborted when the DE bit in DMACHCR or the DME bit in DMAOR is set to 0.

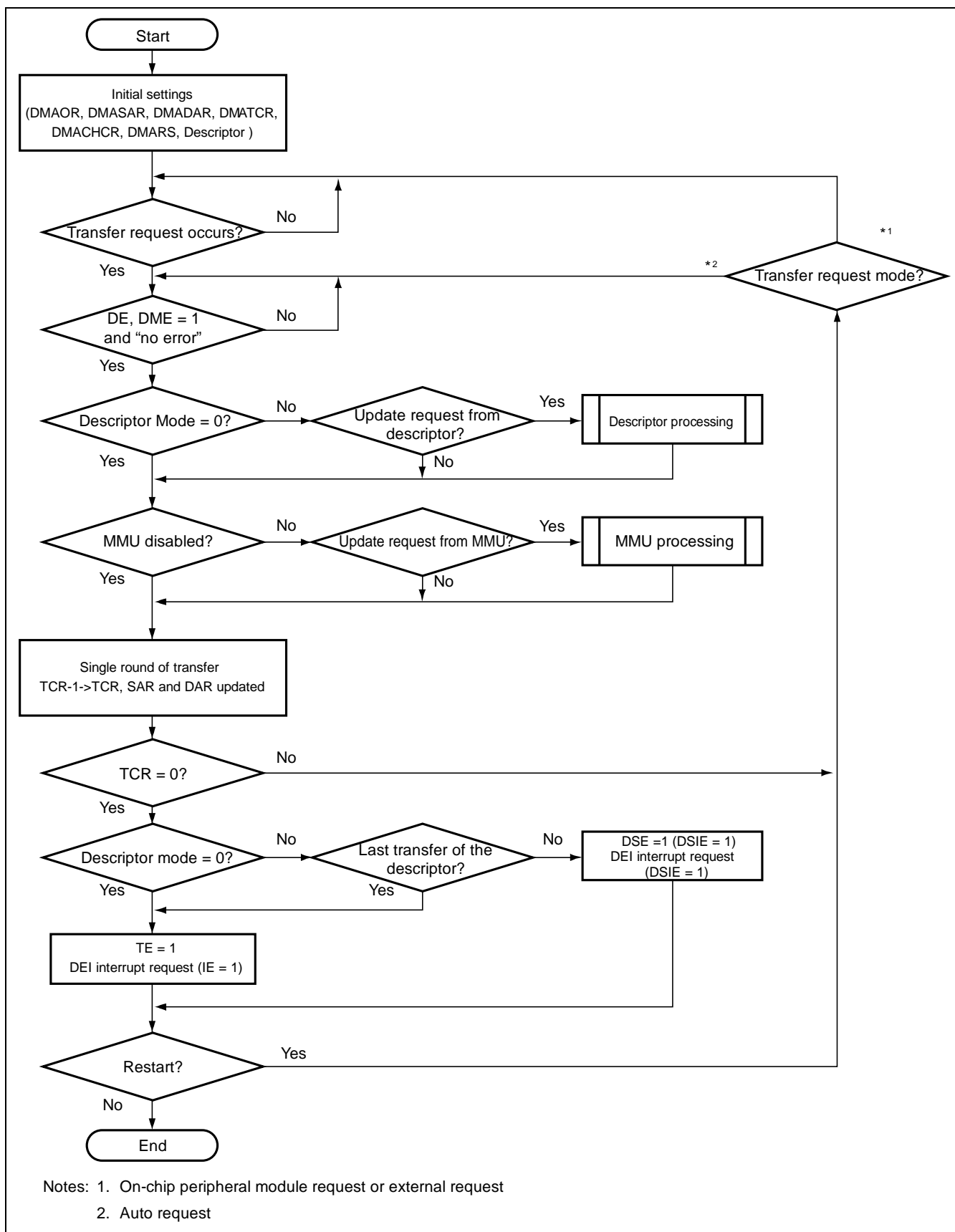


Figure 23.16 Transmission Flow

23.3.6 Total Size Transmission

The amount of data for total size transmission can be set in DMATSR (in bytes), and this setting is effective regardless of the size specified in the TS bits in DMACHCR. Thereby, the desired size can be transmitted in a single round of DMA transfer.

To use this function, make the settings for total size transmission in DMASAR, DMADAR, DMATSR, DMATSRB and DMACHCR.

Total size is set in the TCR (TSR) field of descriptors and 1 is set in DMACHCRB.DTS when total size transmission and descriptors are in use.

23.3.7 Rate access control

In the rate access control mode, the performance of DMA transfer will decrease depending on value of the rate access counter which is set up by CNT bits in WRADM_CONTROL register or RRADM_CONTROL register. When to be enable rate access bit which is set up to 1 by EN bit in WRADM_CONTROL register or RRADM_CONTROL register, the rate access counter starts to count. Whenever value of CNT bits matches with value of counter register, then enable control bit will be toggled after 3 clock cycles.

Period of time of a transaction in the rate access control is calculated as following:

$$\text{Period of time of a transaction} = (\text{CNT} + 1) * 2 + 3 \text{ (clock cycle)}$$

23.4 Secure function

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

23.4.1 Secure function

Security of each channel of SYS-DMAC is protected by DMASEC_* register. Refer description of DMA Secure Control Register for each channel.

In case a channel is in secure mode, when Non-secure CPU access to that channel, the access must be denied, then DMAC do these actions:

- (1) Issue error status and store into DMASES_* register.
- (2) Log error ID of the first access into DMAEMID_* register.
- (3) Log error address of the first access into DMASEA_* register.

23.5 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Pay attention to the following notes when using the SYS-DMAC.

(1) DMA Transfer for Peripheral Modules

When executing DMA transfer for an on-chip peripheral module, set addresses on the appropriate boundary (in terms of the amount of data for each round of transfer) for the transfer source and destination addresses. Otherwise, an address error may occur.

SYS-DMAC channels 0-15 and channels 16-47 are assigned to be used by different Peripheral IPs. Regarding access to CRC module, SYS-DMAC channels 0-15 cannot access CRC module. On the other hand, SYS-DMAC channel 16-47 can access CRC module.

(2) Module Stop

While the SYS-DMAC is operating, the module stop register (MSTPCR2) should not be set to stop the SYS-DMAC. If the SYS-DMAC is stopped in this way, results of the transfer that was in progress cannot be guaranteed.

(3) Address Error

When a DMA address error is generated, reset the registers of the channel on which the error has occurred and then start transfer anew.

(4) Aborting DMA Transfer

To abort a DMA transfer, disable the interrupt signal and set the DE bit in the DMA channel control register (DMACHCR) to 0 to disable the DMA transfer. If the TE and DSE bits are set when DMA transfer is aborted, these bits should be initialized. There is a possibility that TE and DSE will not be set with synchronized timing after transmission, so it's necessary to recognize the following three possibilities and take measures accordingly.

1. The DSE and TE bits are set to 1 before initialization of DMACHCR, but after the interrupt was disabled.
The TE and DSE bits are initialized within the DMA transfer initialization sequence.
2. The DSE and TE bits are set to 1 and the controller fails to abort the transfer, after the interrupt was disabled.
When the DE bit has become 0 after DMA transfer initialization, check the TE and DSE bits. If the TE or DSE bit is 1, go through the DMA transfer initialization process.
3. The TE and DSE bits are not cleared to 0 but the transfer is aborted, after the interrupt is disabled.
The TE or DSE bits are not set because data for transfer still remain after transfer is aborted.

Note: Initialization of DMA transfer during execution of the last transfer leads to a delay in setting of the TE and DSE bits, so include a dummy read.

Figure 23.17 shows an example of processing to abort DMA transfer.

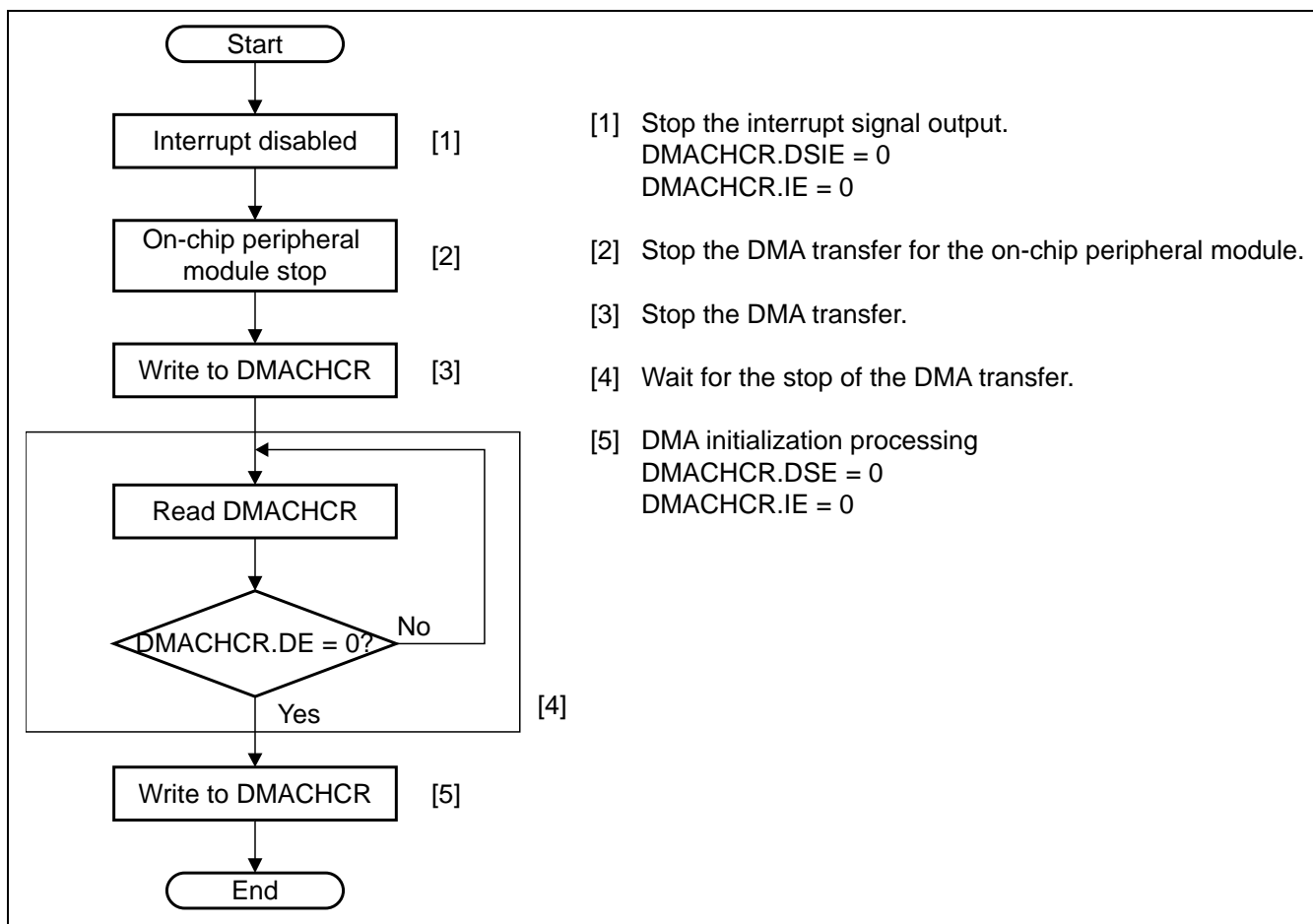


Figure 23.17 Example of Processing to Abort DMA Transfer

24. Booting

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

24.1 Overview

This product allows selection of a master boot processor through the MD7 and MD6 pins. When Cortex-A53, Cortex-A57 is selected as a master boot processor, setting the MD[4:1] pins enables on-chip ROM boot, external ROM boot or XIP(eXecute In Place) mode boot. This module is provided for all RZ/G Series 2nd Generation products.

24.1.1 Features

- Either Cortex-A53, Cortex-A57 can be selected as a master boot processor through the MD7 and MD6 pins. Selection of a master boot processor is described in Table 24.2 Mode Pin Configuration (2).
- The following boot devices are supported through MD[4:1] pins. Selection of a boot device is described in Table 24.1 Mode Pin Configuration (1). Check supported boot device for each product.
 - External ROM (Area 0, local bus)
 - HyperFlash ROM (Max. 320Mbps per pin) and Serial Flash ROM
 - eMMC
 - USB download
 - SCIF as serial downloader
- The boot sequence of HyperFlash, Serial Flash ROM and eMMC using DMA. Cortex-A53, Cortex-A57 master boot processor executes the instructions in on-chip ROM. Data including Boot ROM parameters, address information and data size information of IPL* are transferred from the Serial/HyperFlash ROM and eMMC to System RAM (on-chip RAM). Then the amount of data specified by data size information is transferred to the on-chip RAM. After the transfer, the processor jumps to the address specified by address information of IPL.
- XIP (eXecute In Place) mode of Hyper Flash and Serial Flash boot is supported.
- The initial execution state of Cortex-A57 and Cortex-A53 can be selected through MD15 pin.

Note: * IPL: Initial Program Loader. IPL can be selected by Boot ROM parameter except for eMMC boot.

24.1.2 External Pins

24.1.2.1 MD Pins

Boot device, master boot processor and AArch32/64 can be selected through MD pins.

Table 24.1 Mode Pin Configuration (1)

| Name | Pin Name | I/O | Function | RZ/G Series 2nd Generation | | | | | |
|----------|------------|-------|-----------------------|----------------------------|--|--------|----------------------------|--------|--------|
| | | | | MD[4:1] | Description | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Mode pin | MD4 to MD1 | Input | Select a boot device. | | | | | | |
| | | | | B'0000 | External ROM boot (area 0) | √ | √ | √ | √ |
| | | | | B'0001 | Reserved | — | — | — | — |
| | | | | B'0010 | HyperFlash ROM boot at 160MHz using DMA | √ | √ | √ | — |
| | | | | B'0010 | HyperFlash ROM boot at 150MHz using DMA | — | — | — | √ |
| | | | | B'0011 | HyperFlash ROM boot at 80MHz using DMA | √ | √ | √ | √ |
| | | | | B'0100 | Serial Flash ROM boot at single read 40MHz using DMA | √ | √ | √ | √ |
| | | | | B'0101 | Reserved | — | — | — | — |
| | | | | B'0110 | Reserved | — | — | — | — |
| | | | | B'0111 | Reserved | — | — | — | — |
| | | | | B'1000 | Reserved | — | — | — | — |
| | | | | B'1001 | Reserved | — | — | — | — |
| | | | | B'1010 | HyperFlash ROM at 160MHz(320Mbps) using XIP mode | √ | √ | √ | — |
| | | | | B'1010 | HyperFlash ROM at 150MHz(300Mbps) using XIP mode | — | — | — | √ |
| | | | | B'1011 | HyperFlash ROM at 80MHz using XIP mode | √ | √ | √ | √ |
| | | | | B'1100 | Reserved | √ | √ | √ | √ |
| | | | | B'1101 | eMMC boot at 50MHz x8 bus widths using DMA | √ | √ | √ | √ |
| | | | | B'1110 | USB download mode | √ | — | √ | √ |
| | | | | B'1111 | SCIF download mode | √ | √ | √ | √ |

Table 24.2 Mode Pin Configuration (2)

| Name | Pin Name | I/O | Function | RZ/G Series 2nd Generation | | | | |
|----------|-------------|---------------------|---|----------------------------|------------------------------------|---------------|---------------|--|
| Mode Pin | MD5 | Input | Reserved | | | | | |
| | MD5 | Description | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| | 0 | Setting prohibited. | | √ | √ | √ | √ | |
| | 1 | Normal boot | | √ | √ | √ | √ | |
| Mode pin | MD7 and MD6 | Input | Select a master boot processor. | RZ/G Series 2nd Generation | | | | |
| | MD7 | MD6 | Description | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| | 0 | 0 | Booted through CPU0 in Cortex-A57. | √ | √ | √ | — | |
| | 0 | 1 | Booted through CPU0 in Cortex-A53. | √ | √ | — | √ | |
| | 1 | 0 | Reserved | — | — | — | — | |
| | 1 | 0 | Reserved | — | — | — | — | |
| | 1 | 1 | Reserved | — | — | — | — | |
| Mode pin | MD15 | Input | Select the initial Armv8 execution state of Cortex-A57 and Cortex-A53 | RZ/G Series 2nd Generation | | | | |
| | MD15 | Description | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| | 0 | AArch32 | | √ | √ | √ | √ | |
| | 1 | AArch64 | | √ | √ | √ | √ | |

24.1.2.2 HyperFlash Boot Pins

Table 24.3 HyperFlash Pin Configuration

| | | | | RZ/G Series 2nd Generation | | | |
|--------------|----------------|--------|----------------------------|----------------------------|----------------------------|--------|--------|
| Name | Pin Name | I/O | Function | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Serial clock | QSPI0_SPCLK | Output | Differential clock output+ | √ | √ | √ | √ |
| Serial clock | QSPI1_SPCLK | Output | Differential clock output- | √ | √ | √ | √ |
| Chip select | QSPI0_SSL | Output | Chip select signal. | √ | √ | √ | √ |
| Data0 | QSPI0_MOSI_IO0 | I/O | Data 0 | √ | √ | √ | √ |
| Data1 | QSPI0_MISO_IO1 | I/O | Data 1 | √ | √ | √ | √ |
| Data2 | QSPI0_IO2 | I/O | Data 2 | √ | √ | √ | √ |
| Data3 | QSPI0_IO3 | I/O | Data 3 | √ | √ | √ | √ |
| Data4 | QSPI1_MOSI_IO0 | I/O | Data 4 | √ | √ | √ | √ |
| Data5 | QSPI1_MISO_IO1 | I/O | Data 5 | √ | √ | √ | √ |
| Data6 | QSPI1_IO2 | I/O | Data 6 | √ | √ | √ | √ |
| Data7 | QSPI1_IO3 | I/O | Data 7 | √ | √ | √ | √ |
| RDS | QSPI1_SSL | Input | Read write data strobe | √ | √ | √ | √ |
| Flash Reset | RPC_RESET# | Output | Reset signal output | √ | √ | √ | √ |

24.1.2.3 Serial Flash Boot Pins

Table 24.4 Serial Flash pin Configuration

| | | | | RZ/G Series 2nd Generation | | | |
|---------------|----------------|--------|-----------------------|----------------------------|----------------------------|--------|--------|
| Name | Pin Name | I/O | Function | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Serial clock | QSPI0_SPCLK | Output | Clock output. | √ | √ | √ | √ |
| Chip select | QSPI0_SSL | Output | Chip select signal | √ | √ | √ | √ |
| Transmit data | QSPI0_MOSI_IO0 | Output | Outputs transmit data | √ | √ | √ | √ |
| Receive data | QSPI0_MISO_IO1 | Input | Inputs receive data | √ | √ | √ | √ |
| Flash Reset | RPC_RESET# | Output | Reset signal output | √ | √ | √ | — |

24.1.2.4 SCIF and HSCIF Download Mode Pins

Table 24.5 SCIF2_A Pin Configuration

| | | | | RZ/G Series 2nd Generation | | | |
|---------------|----------|--------|----------------------|----------------------------|----------------------------|--------|--------|
| Name | Pin Name | I/O | Function | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Transmit data | TX2_A | Output | Transmit data output | √ | √ | √ | √ |
| Receive data | RX2_A | Input | Receive data input | √ | √ | √ | √ |

24.1.2.5 eMMC Boot Pins

Table 24.6 MMC1(SDHI3) pin Configuration

| | | | | RZ/G Series 2nd Generation | | | |
|------------------|----------|--------|------------------|----------------------------|----------------------------|--------|--------|
| Name | Pin Name | I/O | Function | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Clock | SD3_CLK | Output | MMC clock | √ | √ | √ | √ |
| Command/response | SD3_CMD | I/O | Command/response | √ | √ | √ | √ |
| Data0 | SD3_DAT0 | I/O | Data [bit 0] | √ | √ | √ | √ |
| Data1 | SD3_DAT1 | I/O | Data [bit 1] | √ | √ | √ | √ |
| Data2 | SD3_DAT2 | I/O | Data [bit 2] | √ | √ | √ | √ |
| Data3 | SD3_DAT3 | I/O | Data [bit 3] | √ | √ | √ | √ |
| Data4 | SD3_DAT4 | I/O | Data [bit 4] | √ | √ | √ | √ |
| Data5 | SD3_DAT5 | I/O | Data [bit 5] | √ | √ | √ | √ |
| Data6 | SD3_DAT6 | I/O | Data [bit 6] | √ | √ | √ | √ |
| Data7 | SD3_DAT7 | I/O | Data [bit 7] | √ | √ | √ | √ |

24.1.2.6 USB Download Mode Pins

| | | | | RZ/G Series 2nd Generation | | | |
|-------------|----------|-----|--|----------------------------|----------------------------|--------|--------|
| Name | Pin Name | I/O | Function | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| USB D+ data | DP0 | I/O | D+ Input/output of the on-chip transceiver. Connect this pin to the D+ pin of the USB bus. | √ | — | √ | √ |
| USB D- data | DM0 | I/O | D- Input/output of the on-chip transceiver. Connect this pin to the D- pin of the USB bus. | √ | — | √ | √ |
| USB VBUS | VBUS0 | I | Connect to USB VBUS with external 30kΩ±1% series resistor. | √ | — | √ | √ |

24.1.3 Connected Module

| Module name | Connected module name | Function of connected module |
|-------------|--------------------------------|---------------------------------|
| Booting | AP-System Core | Boot processor |
| | RESET | Boot address control |
| | System RAM | The destination of IPL transfer |
| | PFC | Pin function |
| | RPC-IF | HyperFlash/Serial Flash IF |
| | SCIF | SCIF download mode |
| | SYS- DMAC | DMA using in Boot ROM code |
| | SD Card/MMC Interfaces | eMMC Alternative Boot operation |
| | USB High-Speed Module (HS-USB) | USB download mode |

24.2 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

24.2.1 Boot Operation Overview

The boot address of the master boot processor is controlled by RESET module. It depends on boot device selection through MD[4:1] pins and AArch64/32 selection through MD15 pin.

If external ROM or XIP mode boot is selected, the processor can directly fetch from the target device. If the boot device except for external ROM and XIP mode is selected, the processor boots from on-chip Boot ROM and executes the boot sequence described “HyperFlash/Serial Flash ROM using DMA Boot sequence” and “eMMC using DMA boot sequence” section. 5KB or 3KB data are transferred from the target boot device to System RAM. Then IPL is transferred to System RAM. Then the processor jumps to the specified address.

If MD[4:1] reserved setting is selected, error message can be outputted via SCIF.

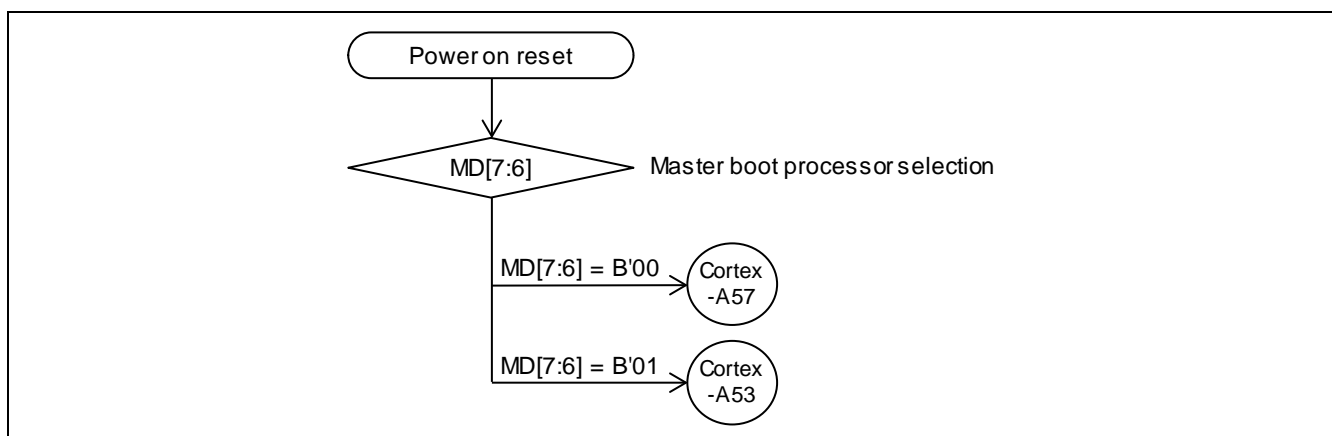


Figure 24.1 Master Boot processor selection

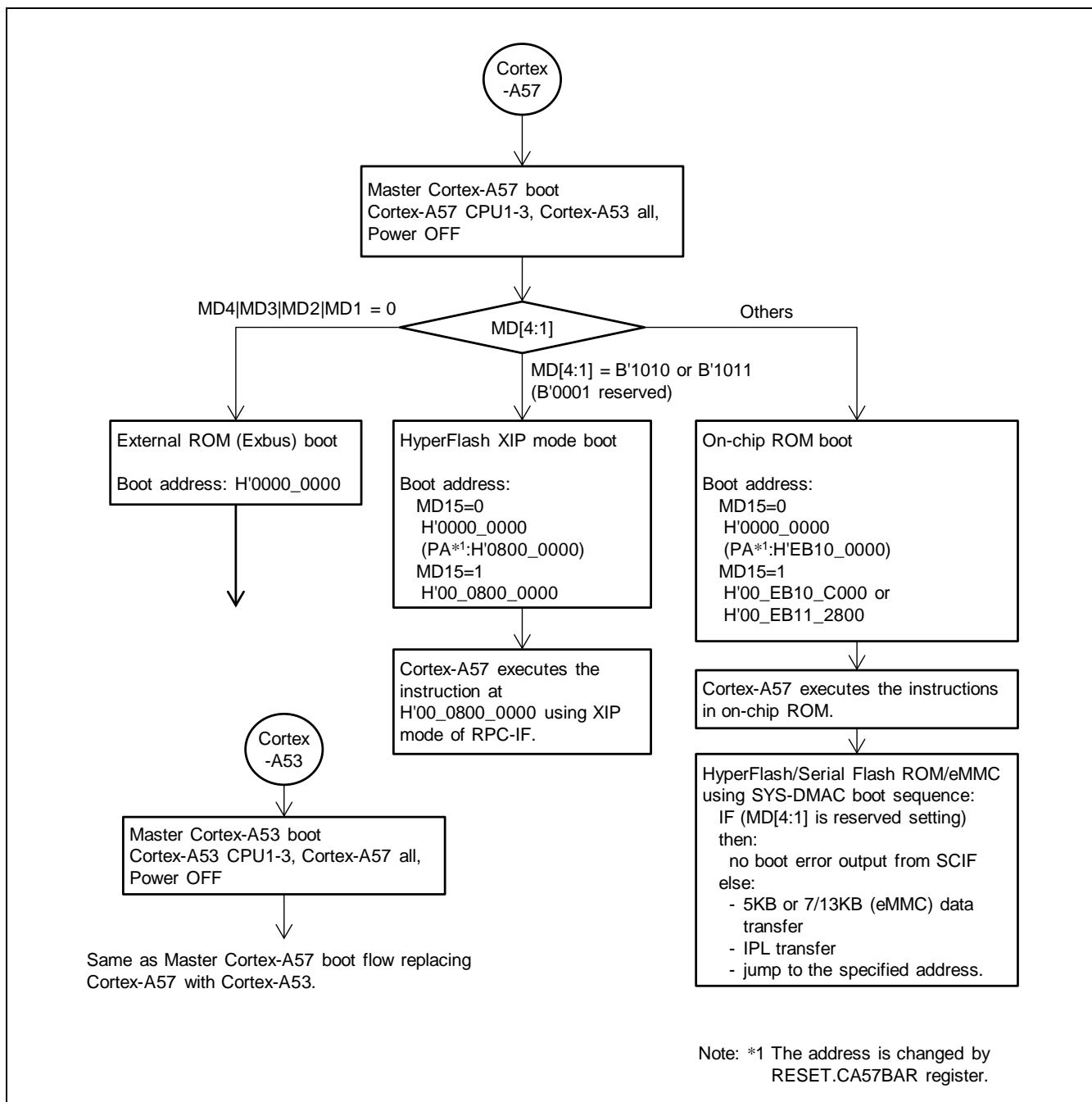


Figure 24.2 Cortex-A57, Cortex-A53, boot operation overview

24.2.2 Boot Address Control

Boot address is controlled by RESET module. If Cortex-A57 at AArch64 boot is selected through MD15 and MD[7:6] pins, RESET.CA57CPU0BARH/L register function is used to specify the boot address. If Cortex-A57 at AArch32 and on-chip ROM boot is selected, RESET.CA57BAR register function is used to specify the boot address. Cortex-A53's boot address is the same as replacing Cortex-A57 with Cortex-A53.

24.2.3 Boot Parameters

1) Boot ROM parameters (External Flash ROM Address H'0000_0000)

Boot ROM parameters are 4bytes and it should be on the top address of serial flash. It controls boot flow.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----------------|----------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 ¹ | 0 ¹ | 0 |
| R/W: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | IPLAB SELR |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 1 | — | All 0 | — | Reserved |
| 0 | IPLABSELR | B'0 | R/W | IPL A/B-Side Select bit. 0: IPL A-Side is loaded. 1: IPL B-Side is loaded. |

eMMC boot do not support this function.

24.2.4 Hyper/Serial ROM Using DMA Boot Sequence

1. The master boot processor jumps to H'EB10_0000 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E AArch32) or H'00_EB10_C000(RZ/G2M V1.3, RZ/G2M V3.0 AArch64) or H'00_EB11_2800 (RZ/G2H, RZ/G2N, RZ/G2E AArch64). The processor executes the instructions in Boot ROM and 5KB data are transferred from Serial Flash, HyperFlash to System RAM using SYS-DMAC channel 0.
2. These 5KB data contain Boot parameters, address information and data size information of IPL-A and IPL-B. Either IPL-A or IPL-B can be selected by Boot ROM parameters.
3. The amount of data specified by data size information is transferred to System RAM. After the transfer, the master processor jumps to the address specified by address information of IPL.

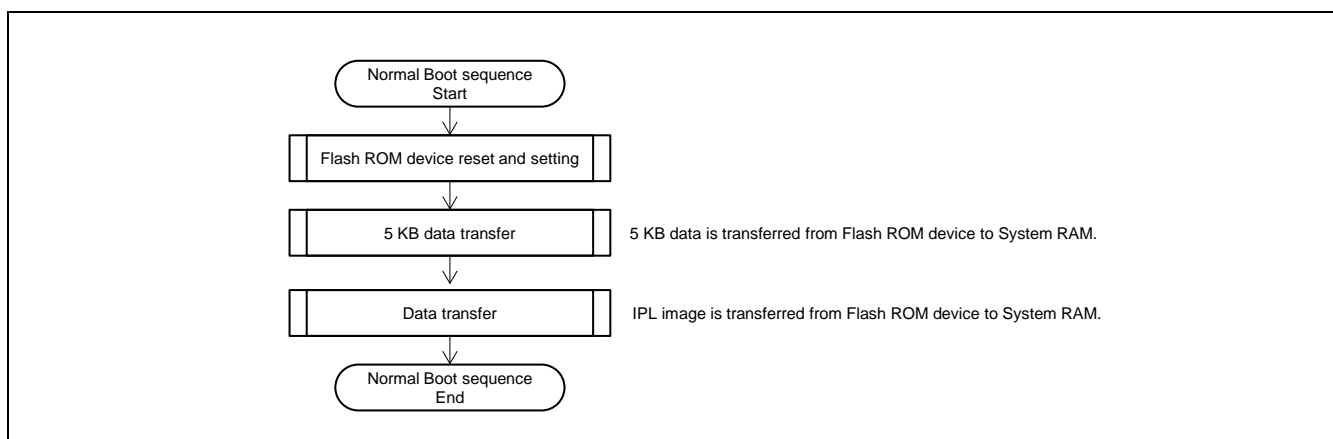


Figure 24.3 Hyper/Serial Flash using SYS-DMA boot sequence

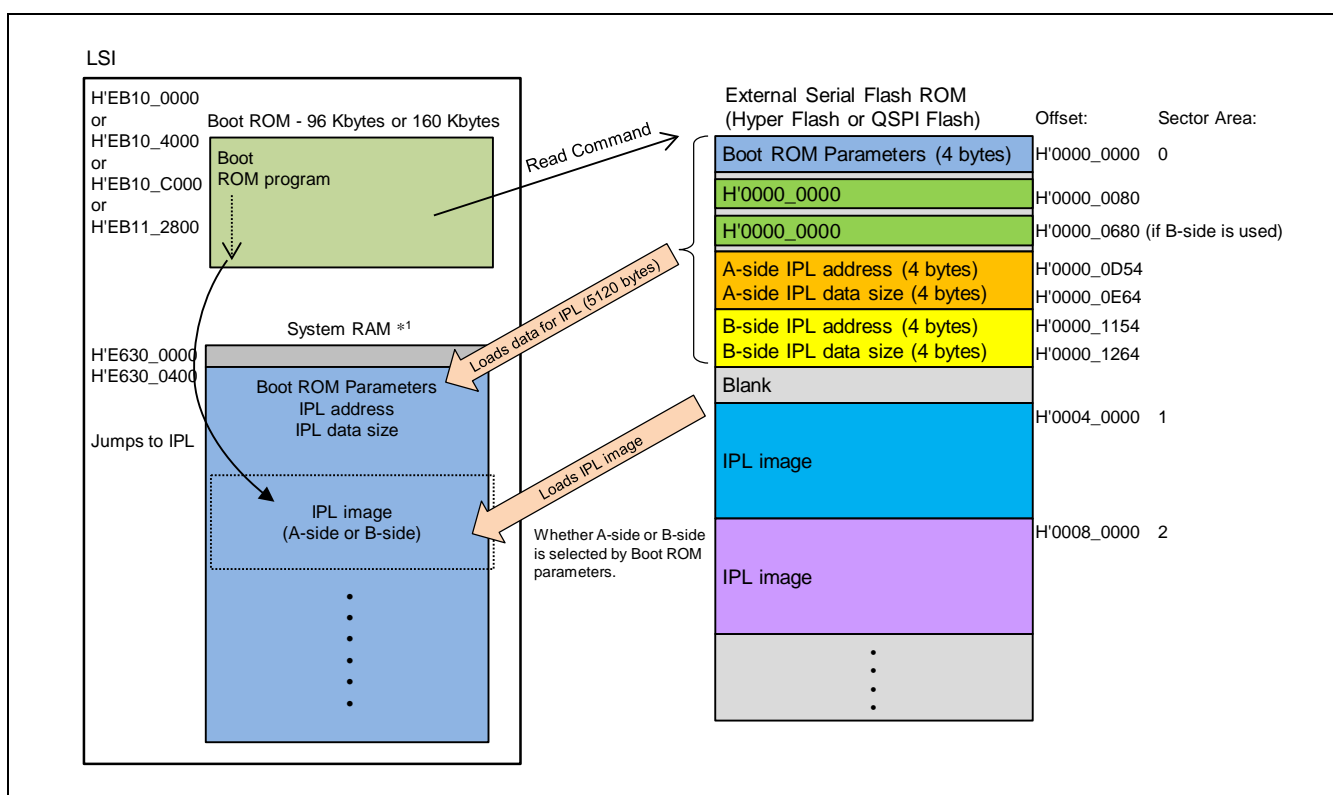


Figure 24.4 IPL transfer and Program image

24.2.5 eMMC Using DMA Boot Sequence

1. The master boot processor jumps to H'EB10_0000 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E AArch32) or H'00_EB10_C000(RZ/G2M V1.3, RZ/G2M V3.0 AArch64) or H'00_EB11_2800 (RZ/G2H, RZ/G2N, RZ/G2E AArch64). The processor executes the instructions in Boot ROM and 3KB data are transferred from eMMC to System RAM using MMC's DMA with 512-Byte transfer size.
2. These 3KB data contain Boot parameters, address information and data size information of IPL.
3. The amount of data specified by data size information is transferred to System RAM. After the transfer, the master processor jumps to the address specified by address information of IPL.

If boot operation from partition 1 fails, switching from boot partition 1 to partition 2 (dual boot mode) is performed.

Note: eMMC device must meet the following conditions.

- eMMC device that conforms to JEDEC eMMC standard 4.4 or higher
- Set the EXT_CSD register byte179, PARTITION_CONFIG bit[6] = H'0(No boot acknowledge sent).
- Set the EXT_CSD register byte179, PARTITION_CONFIG bit[5:3] = H'1 (Boot Area partition1).
- Set the EXT_CSD register byte177, BOOT_BUS_CONDITIONS bit[4:3] = H'1(50MHz SDR).
- Set the EXT_CSD register byte177, BOOT_BUS_CONDITIONS bit[1:0] = H'2(x8 bus widths).
- eMMC device's IO voltage should be 1.8V.
- Place the boot data in eMMC boot area Partition 1 and 2 if dual boot mode is used.

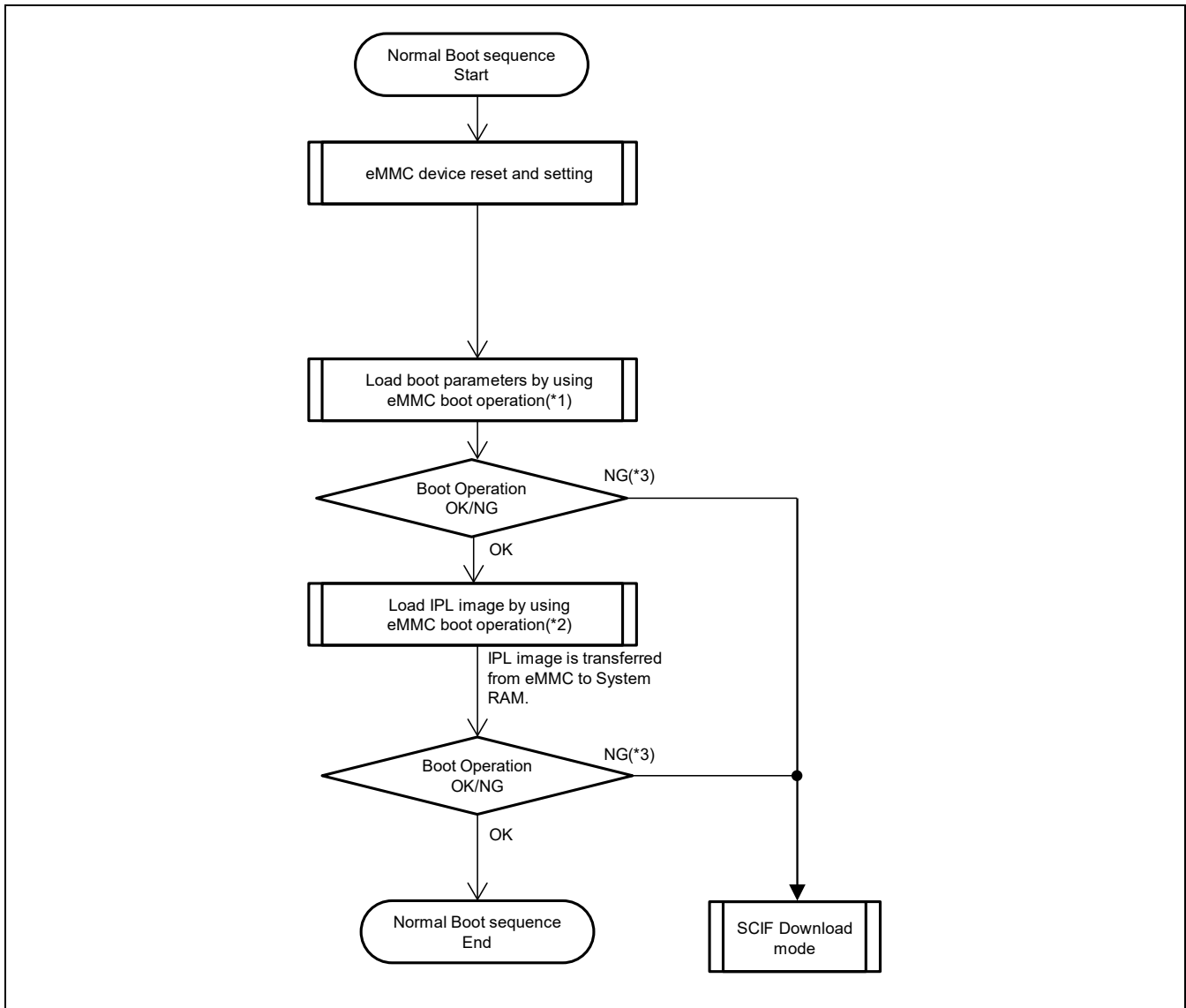


Figure 24.5 eMMC using DMA boot sequence

(*1) CMD0 is issued three times to transfer boot parameters, with argument + H'F0F0_F0F0, + H'FFFF_FFFA, + H'0000_0000.

(*2) As with (*1), CMD0 is also used during IPL image transfer.

(*3) If there is a parameter error in eMMC Boot, an error message will be output and SCIF Download mode will start. Below is an example of error message.

```

Transfer Error!
0x00000000 0x0009A17A 0x00000008 0x00000019
SCIF Download mode (w/o verification)
(C) Renesas Electronics Corp.
-- Load Program to System RAM ----- please send !
  
```

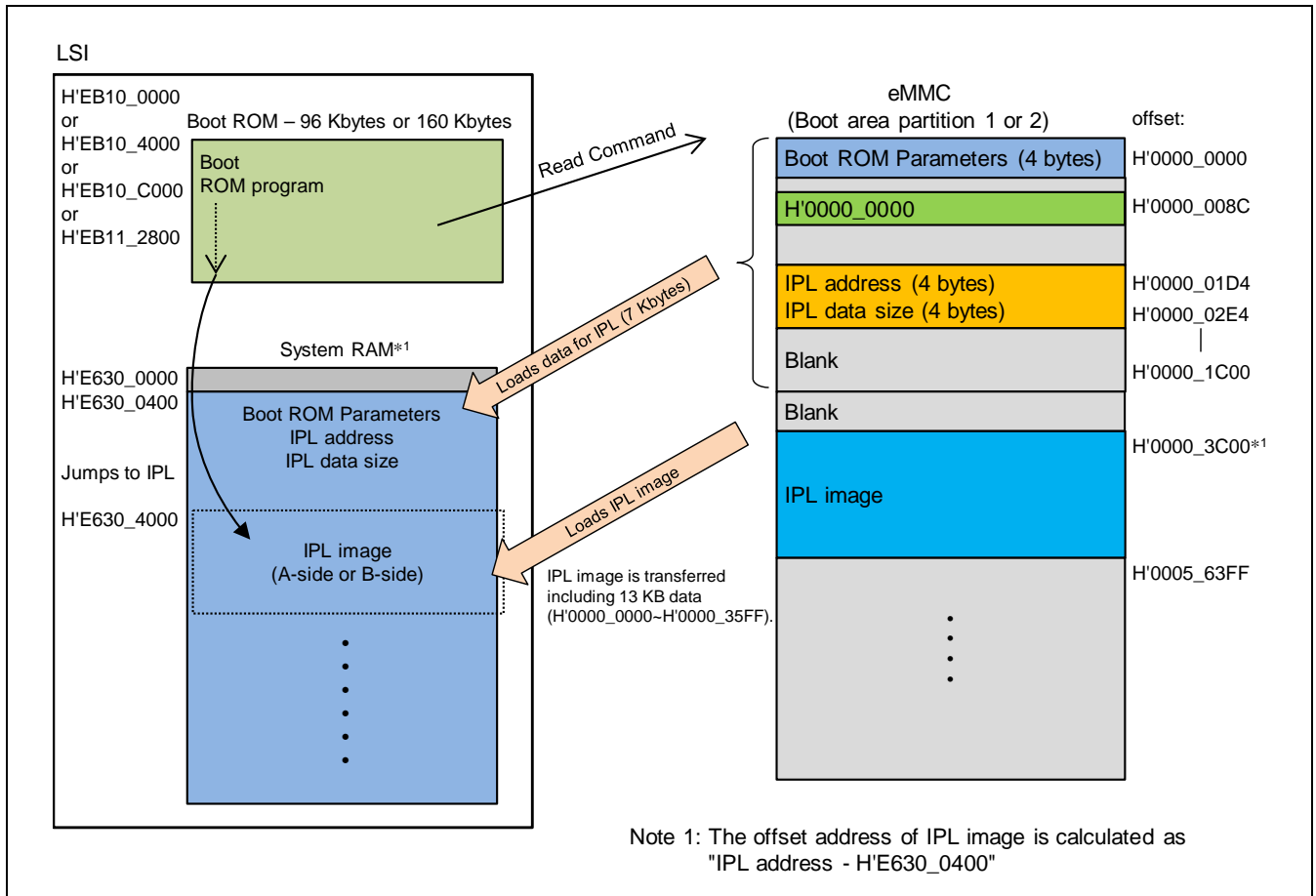


Figure 24.6 IPL transfer and Program image(eMMC)

24.2.6 SCIF Download Mode

SCIF download mode uses SCIF2_A serial ports. Data from Host PC is transferred to System RAM. The transfer specification is below.

- Baud rate: 115200bps
- Data length: 8bit
- Parity: none
- Stop bit: 1bit
- Flow control]: none
- Data format: Motorola S-record

The download sequence and data format are below.

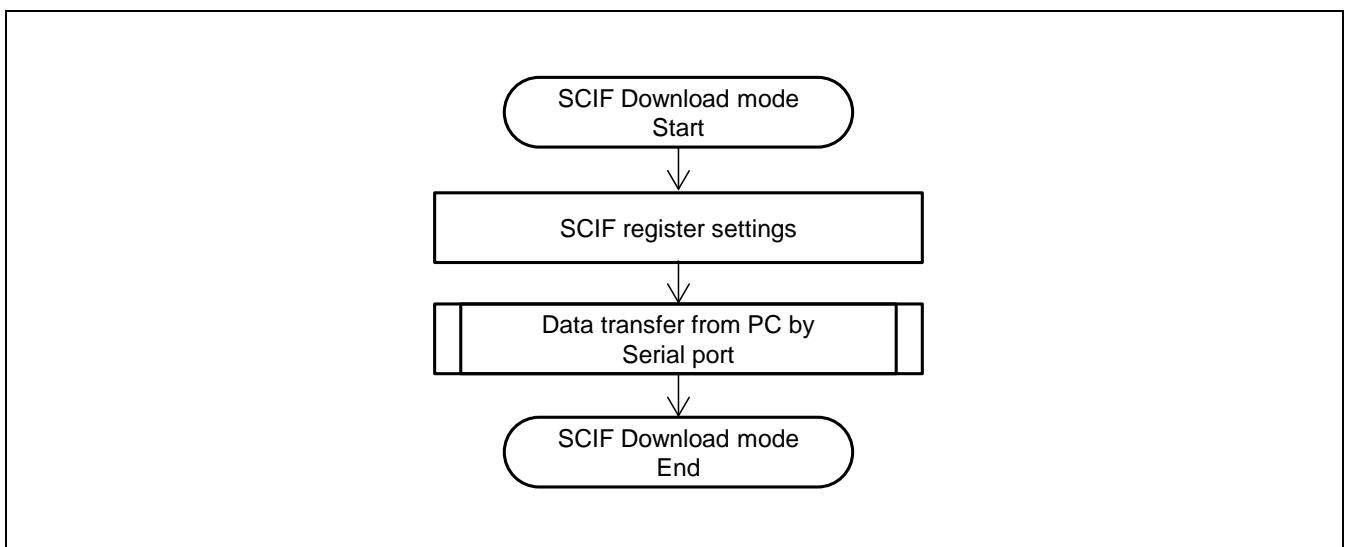


Figure 24.7 SCIF download mode sequence

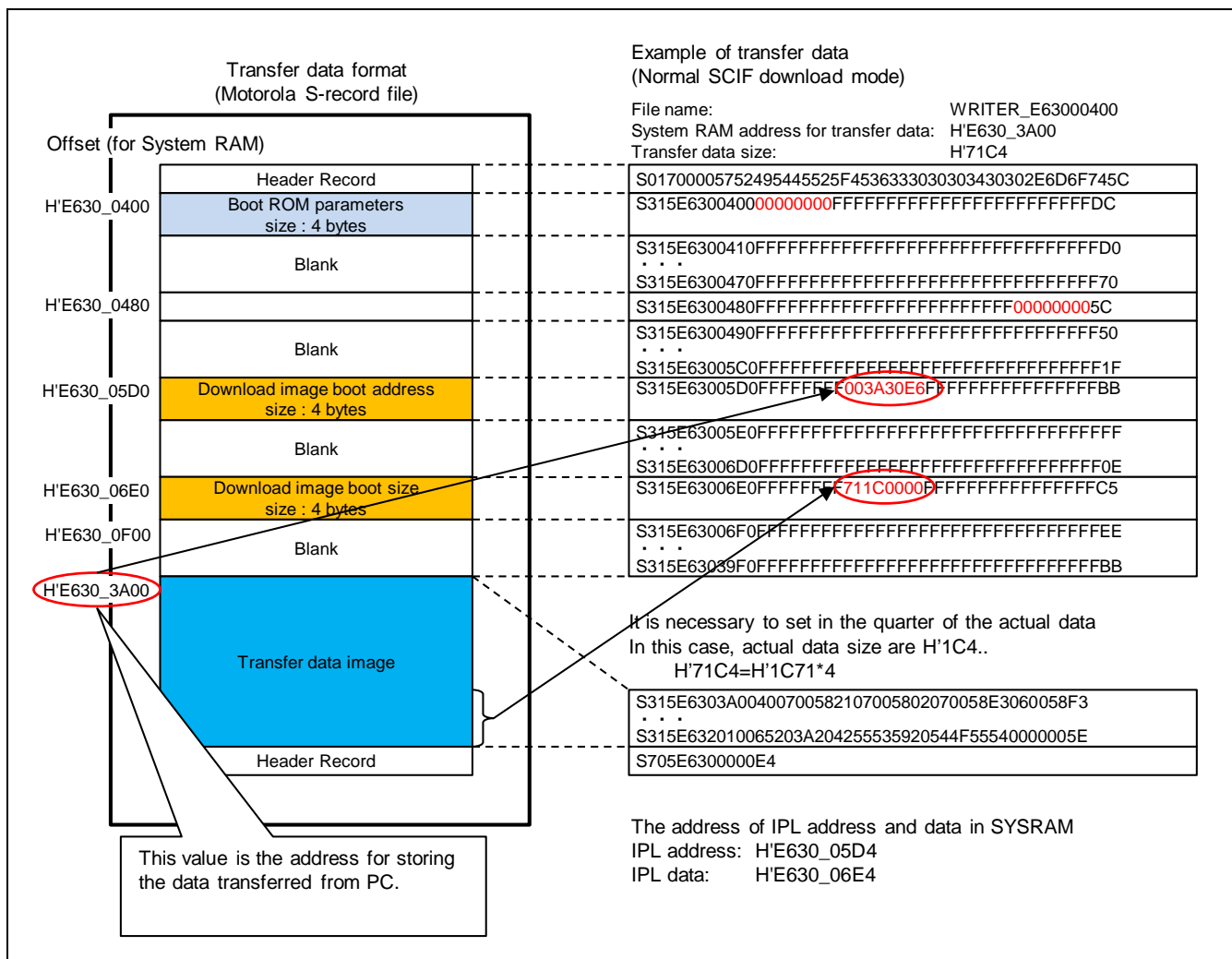


Figure 24.8 SCIF data format for transfer data

24.2.7 USB download mode

USB download mode uses the USB High-Speed port, to transfer the data from Host PC to System RAM. The transfer data format is the same as SCIF download mode's. The transfer specification is below.

- Baud rate: up to USB 2.0 High Speed
- Data format: Motorola S-record
- USB Vendor ID: H'045B
- Product ID: RZ/G2N H'248, RZ/G2E H'024D
- Device Class: Communication Device Class (Abstract Control Model)

24.2.8 Boot ROM

The capacity is 96 Kbytes (RZ/G2M V1.3, RZ/G2M V3.0) or 160 Kbytes (RZ/G2H, RZ/G2N, RZ/G2E). AArch32, AArch64 codes are included.

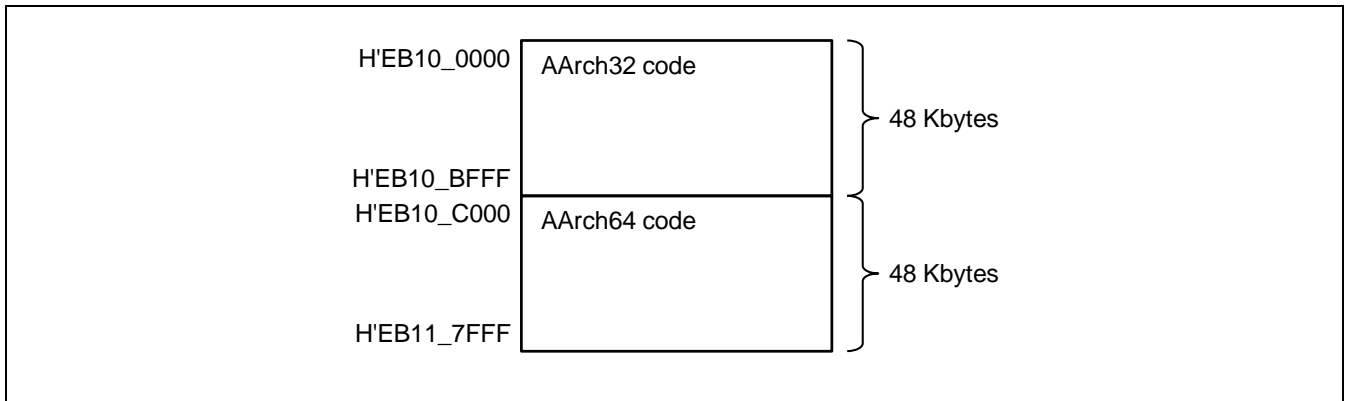


Figure 24.9 Boot ROM memory map (RZ/G2M V1.3, RZ/G2M V3.0)

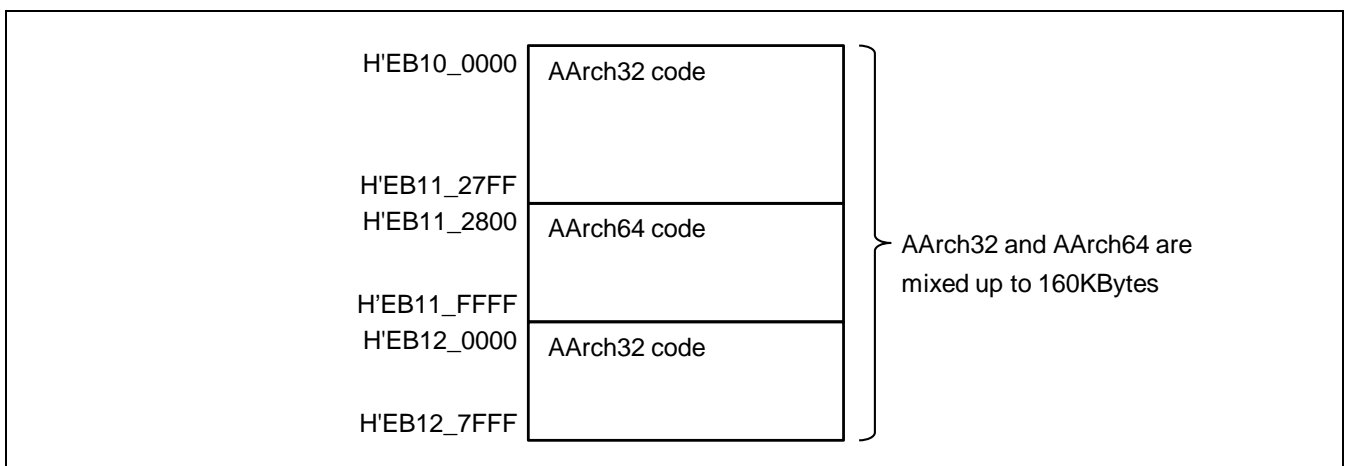


Figure 24.10 Boot ROM memory map (RZ/G2H, RZ/G2N, RZ/G2E)

24.3 Usage Note

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

24.3.1 IPL Address and Data Size Limitation

The IPL end address can be calculated by IPL address and data size. It should be lower than the address below.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: H'E635_67FF
 RZ/G2E: H'E635_7FFF

The IPL address of HyperFlash and Serial Flash ROM boot requires 64-byte alignment. And the IPL address of eMMC boot requires 512-byte alignment.

24.3.2 Cleanup

Before jumping to the IPL, Boot ROM program cleans hardware registers used for loading the IPL and the master boot processor, as listed below.

AArch32 Cortex-A57/Cortex-A53

- Disable I-cache.

AArch64 Cortex-A57/Cortex-A53

- SYS-DMAC registers are set to hardware initial value.
- Disables Stack Alignment check. Disables Alignment fault checking. Disable I-cache.

Note

- The PFC setting is not cleared after jumping to the IPL.
- The settings of RPC-IF module for the transfer is automatically set by MD pin settings. Therefore, registers of RPC-IF module are not cleaned in Boot ROM.
- In the case of eMMC boot, initialize the SDHI interface registers.
- IRQ and FIQ are disabled.

24.3.3 Pull-up/down settings

Boot ROM code supports booting from single channel serial flash and does NOT support booting from dual channel serial Flash. Dual channel serial Flash boot can be selected by setting the BSZ bits of CMNCR register in RPC to B'01 after IPL loading.

| MD[4:1] | Descriptions | QSPI0_IO2/QSPI0_IO3 | QSPI1_SPCLK | QSPI1_IO0/QSPI1_IO1/ QSPI1_IO2/QSPI1_IO3 |
|---------|--------------------|---------------------|-------------|---|
| B'0100 | Serial Flash 40MHz | Pull-up | Pull-down | Pull-up |

24.3.4 Dual channel Serial Flash(Dual QSPI) boot

Boot ROM code supports booting from single channel serial flash and does NOT support booting from dual channel serial Flash. Dual channel serial Flash boot can be selected by setting the BSZ bits of CMNCR register in RPC to B'01 after IPL loading.

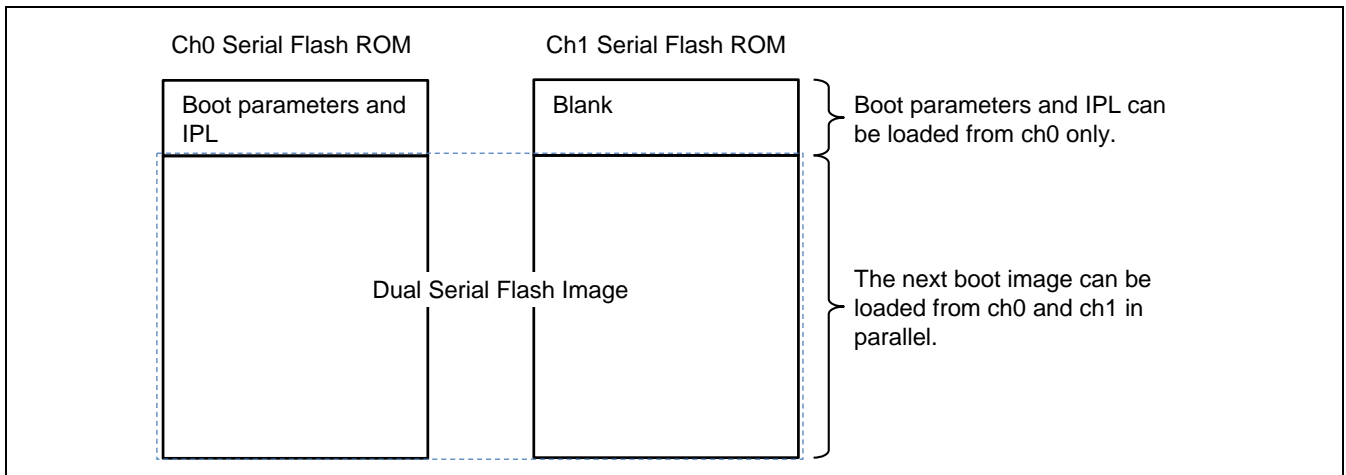


Figure 24.11 Flash image of dual channel Serial Flash booting

25. External Bus Controller for EX-Bus (LBSC)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

25.1 Overview

The LBSC performs bus arbitration and necessary interface conversion for the accesses from the CPU (AXI bus) and outputs them to the external buses. Further, for external bus access, various settings can be specified in the LBSC control registers for the selection of a connection interface type for each area on the external bus or for the adjustment of number of setup/hold cycles on addresses and chip select signals with respect to read/write enable signals. Thus, the LBSC configuration allows diversity in methodology for accessing various external devices that are assigned to their corresponding areas.

The frequency of the external bus clock CLKOUT signal is up to 66.6 MHz. The LBSC outputs bus signals in synchronization with the external bus clock.

25.1.1 Features

The key features of the LBSC include:

- Support for areas 0, 1
 - Each area is allocated to EX-BUS, and SRAM, or byte-control SRAM bus protocol can be selected.
 - Interface, bus size, and wait-cycle insertion can be controlled in each area.
 - Provides bus signals in synchronization with the CLKOUT signal (66.6 MHz/ 44.44 MHz).
- SRAM interface
 - Wait-cycle insertion can be controlled through register settings.
 - Wait cycles can be inserted with the EX_WAIT pins.
 - Connectable bus size: 16 or 8 bits.
- Burst ROM interface (area 0 and CPU access only)
 - Wait-cycle insertion can be controlled through register settings.
 - Burst count can be specified through register settings (cases where this reaches an address branching point are automatically detected, after which the access is broken off).
 - Connectable bus size: 16 or 8 bits
- Byte-control SRAM interface (areas 1 only)
 - SRAM interface with byte control
 - Wait-cycle insertion can be controlled through register settings.
 - Wait cycles can be inserted with the EX_WAIT pins.
 - Connectable bus size: 16 or 8 bits
- Wait timeout
 - Wait timeout detection

Detection time in ns = period in ns at EX-BUS operating frequency × (the time specified in the EX-BUS wait timeout base counter register (EXBCT) and EX-BUS wait timeout detection counter register (EXTCT))

25.1.2 Block Diagram

Figure 25.1 is a block diagram of the LBSC. Placed on the AXI bus, the LBSC outputs accesses from the CPU in sequence to an external bus according to the settings that are provided in internal registers of the LBSC. For the external bus EX-BUS, either of the SRAM bus protocol can be selected the LBSC has an external wait control input that controls the pulse width. When a request for access is received by an external device, the external device uses this to control the wait for a response to suit the situation at the time of access-request reception.

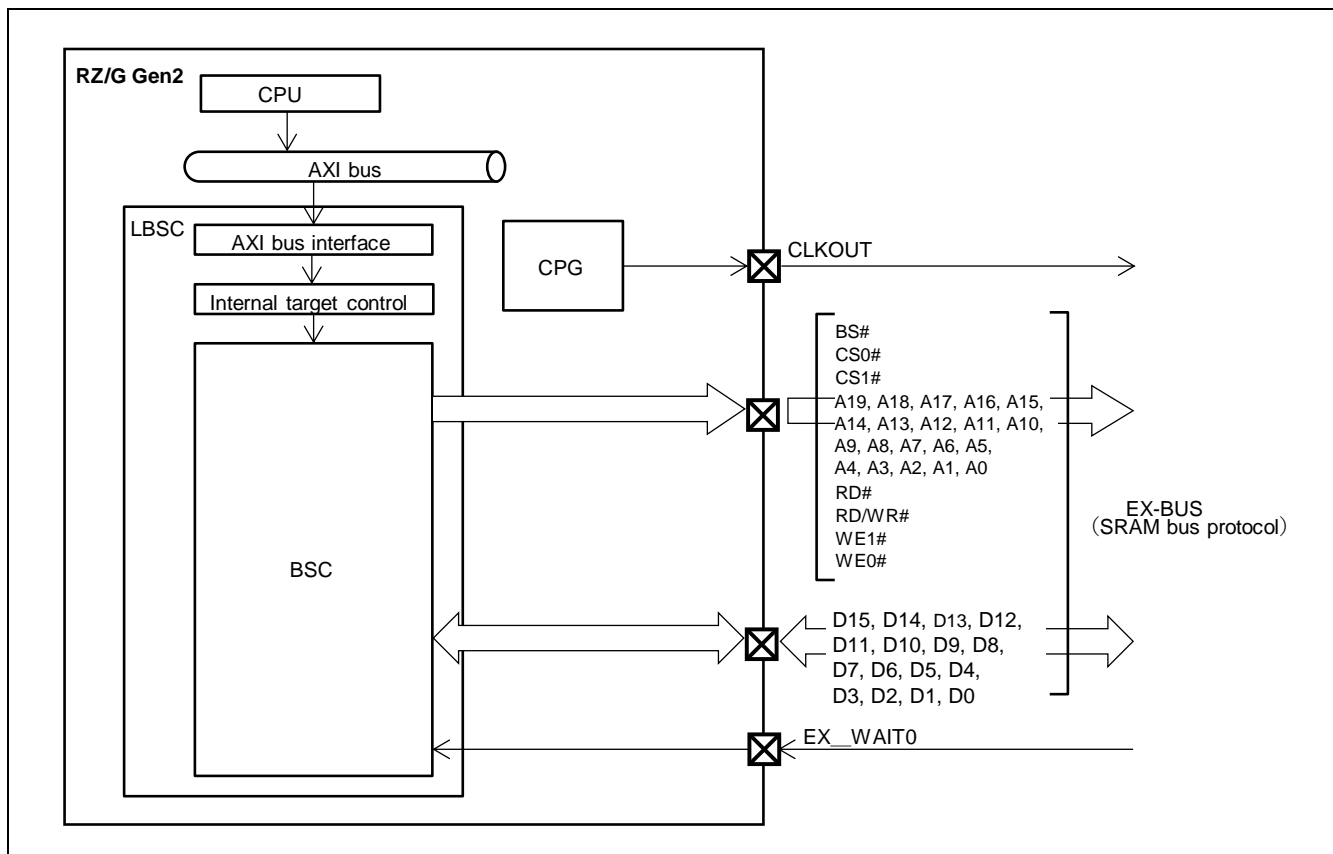


Figure 25.1 Block Diagram of LBSC

25.1.3 LBSC Areas

25.1.3.1 LBSC Support Areas

Figure 25.2 shows LBSC response support areas from the CPU.

The basic configuration of the LBSC supports area 0 and area 1 as external spaces (1Mbytes for each area); Every 1MB address space from H'0010_0000 to H'03FF_FFFF can be used as shadowed area of 1MB from the begging of Area0 (H'0000_0000 – H'000F_FFFF). And every 1MB address space from H'0410_0000 to H'07FF_FFFF can be used as shadowed area of 1MB from the begging of Area1 (H'0400_0000 – H'040F_FFFF). Registers of the BSC is provided in the internal register space of the LBSC.

| LBSC response support area | |
|----------------------------------|--------------------------------------|
| H'0000_0000 to H'03FF_FFFF | Area0 CS0 1 Mbytes |
| H'0400_0000 to H'07FF_FFFF | Area1 CS1 1 Mbytes |
| | Other module space |
| H'EE22_0000 to H'EE22_FFFF | Space for LBSC internal registers |

When access CS1 (Area 1) then CS1# will be toggle.

Figure 25.2 LBSC Response Support Areas from the CPU

25.1.3.2 Functionality Supported in Each Area

Table 25.1 lists the functions supported by the LBSC in each area on the EX-BUS.

Table 25.1 Functions Supported in Each Area on EX-BUS

| Area | Bus | Capacity | Operating Mode | Guard interval | WAIT Function |
|------|-----------|---------------------|---------------------------|----------------|---------------|
| 0 | 8/16 bits | 1 Mbytes selectable | SRAM Burst ROM | Disabled | Enabled |
| 1 | 8/16 bits | Fixed to 1 Mbytes | SRAM Byte-control SRAM | Enabled | Enabled |

Notes: 1. The bus size for area 0 is specified with the LSI mode pins (MD8 = 0: 8 bits, MD8 = 1: 16 bits).
2. When accessing through EX-BUS, A[0] is output as byte address even if 16-bit bus is selected.

25.1.4 Register Configuration

The LBSC set registers to control the interface, bus size, RD/WE# signal pulse cycles, and setup and hold cycles for the CS# signal with respect to the RD/WE# signal, for each of externally connected devices. Table 25.2 lists registers of the LBSC. Note that correct operation is not guaranteed in principle if each register is modified during external bus access (for register modification in other cases, refer to the note under specific register description).

Table 25.2 LBSC Register Configuration

| Register Name | Abbreviation | Access Type | Value after Power-On Reset | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---------------------------------------|--------------|-------------|----------------------------|-------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Area 0 control register | CSOCTRL | R/W | Undefined | H'EE22_0200 | 32 | √ | √ | √ | √ |
| Area 1 control register | CS1CTRL | R/W | H'0000_0020 | H'EE22_0204 | 32 | √ | √ | √ | √ |
| Area 0 RD/WE pulse control register | CSWCR0 | R/W | H'FF70_FF70 | H'EE22_0230 | 32 | √ | √ | √ | √ |
| Area 1 RD/WE pulse control register | CSWCR1 | R/W | H'FF70_FF70 | H'EE22_0234 | 32 | √ | √ | √ | √ |
| Area 0 external wait control register | CSPWCR0 | R/W | H'0000_0000 | H'EE22_0280 | 32 | √ | √ | √ | √ |
| Area 1 external wait control register | CSPWCR1 | R/W | H'0000_0000 | H'EE22_0284 | 32 | √ | √ | √ | √ |
| External wait input control register | EXWTSYNC | R/W | H'0000_0000 | H'EE22_02A0 | 32 | √ | √ | √ | √ |
| Area 0 burst control register | CS0BSTCTL | R/W | H'0000_0000 | H'EE22_02B0 | 32 | √ | √ | √ | √ |
| Area 0 burst pitch set register | CS0BTPH | R/W | H'0000_00F7 | H'EE22_02B4 | 32 | √ | √ | √ | √ |
| Area 1 guard setting register | CS1GDST | R/W | H'0000_0000 | H'EE22_02C0 | 32 | √ | √ | √ | √ |

| Register Name | Abbreviation | Access Type | Value after Power-On Reset | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|--------------|-------------|----------------------------|-------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| BSC interrupt source status register | BCINTSR | R | H'0000_0000 | H'EE22_0330 | 32 | √ | √ | √ | √ |
| BSC interrupt source clear register | BCINTCR | —/WC1 | H'0000_0000 | H'EE22_0334 | 32 | √ | √ | √ | √ |
| BSC interrupt enable register | BCINTMR | R/W | H'0000_0000 | H'EE22_0338 | 32 | √ | √ | √ | √ |
| External wait status register | EXWTSTS | R | Undefined | H'EE22_0344 | 32 | √ | √ | √ | √ |
| EX-BUS wait timeout detection base counter register | EXBCT | R/W | H'0000_0000 | H'EE22_03C0 | 32 | √ | √ | √ | √ |
| EX-BUS wait timeout detection counter register | EXTCT | R/W | H'0000_0000 | H'EE22_03C4 | 32 | √ | √ | √ | √ |
| EX-BUS wait timeout detection access source indication register | EXTSR | R/WC1 | H'0000_0000 | H'EE22_0010 | 32 | √ | √ | √ | √ |
| EX-BUS wait timeout detection address indication register | EXTADR | R/W | H'0000_0000 | H'EE22_0014 | 32 | √ | √ | √ | √ |

Note: Do not write to any addresses other than listed above. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed above are undefined.

25.2 Register Description

Registers in the LBSC are allocated to and arranged in the address space of the internal bus.

Legend for Register Description

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

R: Read-only. When the bit is reserved, the write value should always be 0.

Writing 1 to these bits can cause a malfunction of LBSC.

—/W: Write-only. The read value is undefined.

—/WC1: Write-only. Writing 1 initializes the bit. Writing 0 is ignored.

25.2.1 Area 0 Control Register (CS0CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CS0CTRL specifies the interface in area 0 (EX-BUS).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-------|----|----|----|-------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | CS0SZ | | — | — | CS0IF | |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | — | B'1 | R | Reserved These bits are always read as 1. The write value should always be 1. |
| 14 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 7, 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5, 4 | CS0SZ | — | R | Area 0 Bus Size Indication (These bits indicate the value specified by the LSI mode pin MD8.) B'00: Setting prohibited B'01: 8 bits B'10: 16 bits B'11: Setting prohibited |
| 3, 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1, 0 | CS0IF | B'00 | R/W | Area 0 Interface Selection B'00: Standard (SRAM) B'01: Burst ROM B'10: Setting prohibited B'11: Setting prohibited |

- Notes:
- Even when the burst ROM interface is selected by setting CS0IF = B'01, burst ROM operation is not available unless appropriate setting is made in CS0BSTCTL. Be sure to specify both CS0BSTCTL and CS0BTPH before using the burst ROM interface.
 - Setting of burst ROM is valid for CPU access only.

25.2.2 Area 1 Control Register (CS1CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CS1CTRL specifies the interface in area 1 (EX-BUS).

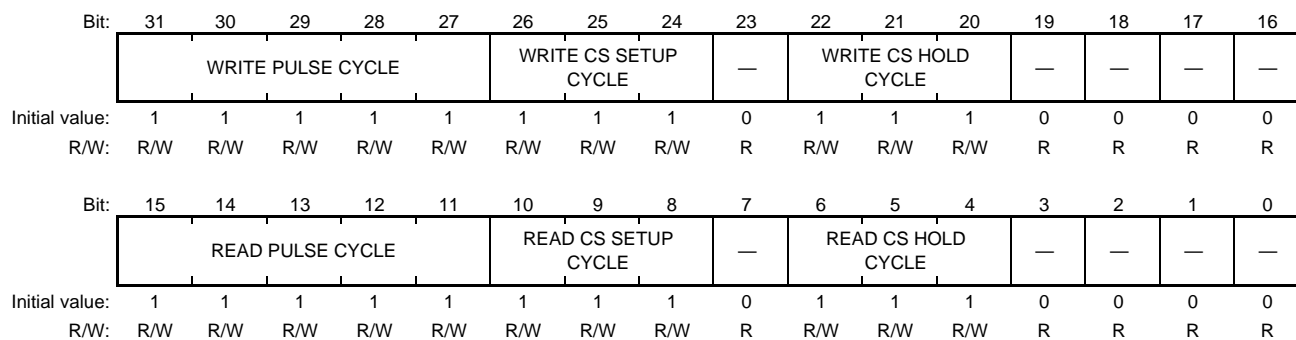
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-------|-----|----|--------|-----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | CS1SZ | — | — | CS1BRM | — | CS1IF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 15 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5, 4 | CS1SZ | B'10 | R/W | Area 1 Bus Size Selection B'00: Setting prohibited B'01: 8 bits B'10: 16 bits B'11: Setting prohibited |
| 3 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 2 | CS1BRM | B'0 | R/W | Area 1 Byte-Control SRAM Mode Selection (valid only when CS1IF = 01) 0: Same cycle as CS# 1: Same cycle as RD# |
| 1, 0 | CS1IF | B'00 | R/W | Area 1 Interface Selection B'00: Standard (SRAM) B'01: Byte-control SRAM B'10: Setting prohibited B'11: Setting prohibited |

25.2.3 Area 0 RD/WE Pulse Control Register (CSWCR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CSWCR0 specifies the RD/WE# pulse cycles and setup and hold cycles for the CS# signal and address during access to area 0 (EX-BUS). (The settings for read access are ignored when the burst ROM interface is selected.)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------|---------------|-----|--|
| 31 to 27 | WRITE PULSE CYCLE | B'1_1111 | R/W | These bits specify the WE# pulse cycles during writing to area 0. B'0_0000: Setting prohibited B'0_0001: 1-cycle pulse B'0_0010: 2-cycle pulse B'0_0011: 3-cycle pulse B'0_0100: 4-cycle pulse : B'1_1101: 29-cycle pulse B'1_1110: 30-cycle pulse B'1_1111: 31-cycle pulse |
| 26 to 24 | WRITE CS SETUP CYCLE | B'111 | R/W | These bits specify the CS# and address setup cycles with respect to the WE# signal during writing to area 0. B'000: 0 cycle for setup period B'001: 1 cycle for setup period B'010: 2 cycles for setup period : B'110: 6 cycles for setup period B'111: 7 cycles for setup period |
| 23 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 22 to 20 | WRITE CS HOLD CYCLE | B'111 | R/W | These bits specify the CS# and address hold cycles with respect to the WE# signal during writing to area 0. B'000: 0 cycle for hold period B'001: 1 cycle for hold period B'010: 2 cycles for hold period : B'110: 6 cycles for hold period B'111: 7 cycles for hold period |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|--|
| 19 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 11 | READ PULSE CYCLE | B'1_1111 | R/W | These bits specify the RD# pulse cycles during reading from area 0. B'0_0000: Setting prohibited B'0_0001: 1-cycle pulse B'0_0010: 2-cycle pulse B'0_0011: 3-cycle pulse B'0_0100: 4-cycle pulse : B'1_1101: 29-cycle pulse B'1_1110: 30-cycle pulse B'1_1111: 31-cycle pulse |
| 10 to 8 | READ CS SETUP CYCLE | B'111 | R/W | These bits specify the CS# and address setup cycles with respect to the RD# signal during reading from area 0. B'000: 0 cycle for setup period B'001: 1 cycle for setup period B'010: 2 cycles for setup period : B'110: 6 cycles for setup period B'111: 7 cycles for setup period |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 to 4 | READ CS HOLD CYCLE | B'111 | R/W | These bits specify the CS# and address hold cycles with respect to the RD# signal during reading from area 0. B'000: 0 cycle for hold period B'001: 1 cycle for hold period B'010: 2 cycles for hold period : B'110: 6 cycles for hold period B'111: 7 cycles for hold period |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

- Notes:
1. A minimum of two clock cycles are required for one EX-BUS access cycle and therefore, the setting must satisfy this lower limit.
Setting less than two clock cycles for one access cycle is not expected in the design of this LSI, and if attempted correct operation is not guaranteed.
Example: When CSSetupCycle = B'000, CSHoldCycle = B'000, and PulseCycle = B'0_0001, correct operation is not guaranteed.
 2. When controlling wait insertion through LSI external pins (EX_WAIT0), set PulseCycle to B'0_0010 or a larger value. If B'0_0001 or a smaller value is specified, wait insertion through an external pin is disabled.
 3. When the burst ROM interface is specified for area 0, the read access-related settings for area 0 in this register is ignored and settings in CS0BTPH are enabled.
 4. For details, refer to section 25.3.1, SRAM Interface (Basic Functionality).

25.2.4 Area 1 RD/WE Pulse Control Register (CSWCR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CSWCR1 specifies the RD/WE# pulse cycles and setup and hold cycles for the CS# signal and address during access to area 1 (EX-BUS).

| | | | | | | | | | | | | | | | | |
|----------------|-------------------|-----|-----|-----|-----|----------------------|-----|-----|----|---------------------|-----|-----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | WRITE PULSE CYCLE | | | | | WRITE CS SETUP CYCLE | | | — | WRITE CS HOLD CYCLE | | | — | — | — | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | READ PULSE CYCLE | | | | | READ CS SETUP CYCLE | | | — | READ CS HOLD CYCLE | | | — | — | — | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------|---------------|-----|---|
| 31 to 27 | WRITE PULSE CYCLE | B'1_1111 | R/W | These bits specify the WE pulse cycles during writing to area 1. B'0_0000: Setting prohibited B'0_0001: 1-cycle pulse B'0_0010: 2-cycle pulse B'0_0011: 3-cycle pulse B'0_0100: 4-cycle pulse : B'1_1101: 29-cycle pulse B'1_1110: 30-cycle pulse B'1_1111: 31-cycle pulse |
| 26 to 24 | WRITE CS SETUP CYCLE | B'111 | R/W | These bits specify the CS# and address setup cycles with respect to the WE# signal during writing to area 1. B'000: 0 cycle for setup period B'001: 1 cycle for setup period B'010: 2 cycles for setup period : B'110: 6 cycles for setup period B'111: 7 cycles for setup period |
| 23 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 22 to 20 | WRITE CS HOLD CYCLE | B'111 | R/W | These bits specify the CS# and address hold cycles with respect to the WE# signal during writing to area 1. B'000: 0 cycle for hold period B'001: 1 cycle for hold period B'010: 2 cycles for hold period : B'110: 6 cycles for hold period B'111: 7 cycles for hold period |
| 19 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|--|
| 15 to 11 | READ PULSE CYCLE | B'1_1111 | R/W | These bits specify the RD# pulse cycles during reading from area 1. B'0_0000: Setting prohibited B'0_0001: 1-cycle pulse B'0_0010: 2-cycle pulse B'0_0011: 3-cycle pulse B'0_0100: 4-cycle pulse : B'1_1101: 29-cycle pulse B'1_1110: 30-cycle pulse B'1_1111: 31-cycle pulse |
| 10 to 8 | READ CS SETUP CYCLE | B'111 | R/W | These bits specify the CS# and address setup cycles with respect to the RD# signal during reading from area 1. B'000: 0 cycle for setup period B'001: 1 cycle for setup period B'010: 2 cycles for setup period : B'110: 6 cycles for setup period B'111: 7 cycles for setup period |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 to 4 | READ CS HOLD CYCLE | B'111 | R/W | These bits specify the CS# and address hold cycles with respect to the RD# signal during reading from area 1. B'000: 0 cycle for hold period B'001: 1 cycle for hold period B'010: 2 cycles for hold period : B'110: 6 cycles for hold period B'111: 7 cycles for hold period |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

- Notes:
1. A minimum of two clock cycles are required for one EX-BUS access cycle and therefore, the setting must satisfy this lower limit.
Setting less than two clock cycles for one access cycle is not expected in the design of this LSI, and if attempted correct operation is not guaranteed.
Example: When CSSetupCycle = B'000, CSHoldCycle = B'000, and PulseCycle = B'0_0001, correct operation is not guaranteed.
 2. When controlling wait insertion through LSI external pins (EX_WAIT0), set PulseCycle to B'0_0010 or a larger value. If B'0_0001 or a smaller value is specified, wait insertion through an external pin is disabled.
 3. For details, refer to section 25.3.1, SRAM Interface (Basic Functionality).

25.2.5 Area 0 External Wait Control Register (CSPWCR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CSPWCR0 makes settings for external wait signal during access to area 0 (EX-BUS) (the settings are ignored when the burst ROM interface is selected).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-----|-----|------|-----|-----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | V | RB | WINV | — | — | EXWT0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 | V | B'0 | R/W | Area 0 External Wait Signal Enable/Disable 0: Disabled 1: Enabled |
| 4 | RB | B'0 | R/W | Area 0 READY/BUSY Logic Selection 0: BUSY logic 1: READY logic |
| 3 | WINV | B'0 | R/W | Area 0 External Wait Signal Polarity 0: Does not invert the polarity of the area 0 external wait signal. 1: Inverts the polarity of the area 0 external wait signal. |
| 2 | — | B'0 | R/W | Reserved The write value must always be 0. Operation is not guaranteed if setting 1. |
| 1 | — | B'0 | R/W | Reserved The write value must always be 0. Operation is not guaranteed if setting 1. |
| 0 | EXWT0 | B'0 | R/W | Area 0 EX_WAIT0 Enable 0: Disables EX_WAIT0 for area 0. 1: Enables EX_WAIT0 for area 0. |

- Notes:
- When this register setting is made valid (bit V = 1), any one of bits EXWT0 must be set to 1.
 - When bit V = 0, the settings in EXWT0 are ignored. In area 0, this register setting is ignored in read access when the burst ROM interface is selected. For details on wait control, refer to section 25.3.1, SRAM Interface (Basic Functionality).

25.2.6 Area 1 External Wait Control Register (CSPWCR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CSPWCR1 makes settings for external wait input pins during access to area 1 (EX-BUS).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-----|-----|------|-----|-----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | V | RB | WINV | — | — | EXWTO |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 | V | B'0 | R/W | Area 1 External Wait Signal Enable/Disable 0: Disabled 1: Enabled |
| 4 | RB | B'0 | R/W | Area 1 READY/BUSY Logic Selection 0: BUSY logic 1: READY logic |
| 3 | WINV | B'0 | R/W | Area 1 External Wait Signal Polarity 0: Does not invert the polarity of the area 1 external wait signal. 1: Inverts the polarity of the area 1 external wait signal. |
| 2 | — | B'0 | R/W | Reserved The write value must always be 0. Operation is not guaranteed if setting 1. |
| 1 | — | B'0 | R/W | Reserved The write value must always be 0. Operation is not guaranteed if setting 1. |
| 0 | EXWTO | B'0 | R/W | Area 1 EX_WAIT0 Enable 0: Disables EX_WAIT0 for area 1. 1: Enables EX_WAIT0 for area 1. |

- Notes:
1. When this register setting is made valid (bit V = 1), any one of bits EXWTO must be set to 1.
 2. When bit V = 0, the settings in EXWTO are ignored. For details on wait control, refer to section 25.3.1, SRAM Interface (Basic Functionality).

25.2.7 External Wait Input Control Register (EXWTSYNC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: EXWTSYNC controls whether or not to synchronize the external wait pins (EXWAIT0).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|-----|----|-----|-----|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | EXWT SYNC0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | — | B'0 | R/W | Reserved The write value should always be 0. |
| 3 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 2 | — | B'0 | R/W | Reserved The write value must always be 0. Operation is not guaranteed if setting 1. |
| 1 | — | B'0 | R/W | Reserved The write value must always be 0. Operation is not guaranteed if setting 1. |
| 0 | EXWTSYNC0 | B'0 | R/W | 0: Does not synchronize EX_WAIT0 (the original EX_WAIT0 is synchronous with CLKOUT). 1: Synchronizes EX_WAIT0 (the original EX_WAIT0 is asynchronous with CLKOUT). |

Note: For details on wait control, refer to section 25.3.1, SRAM Interface (Basic Functionality).

25.2.8 Area 0 Burst Control Register (CS0BSTCTL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CS0BSTCTL specifies the burst length for area 0 when the burst ROM interface is selected.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|------------|-----|-----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | A0BST[2:0] | | | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 to 11 | A0BST[2:0] | B'000 | R/W | Area 0 Burst Length for Burst ROM Interface B'001: 4 access cycles B'010: 8 access cycles B'011: 16 access cycles B'100: 32 access cycles Others: No burst transfer |
| 10 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

- Notes:
1. This register is valid only when the burst ROM interface is selected (CS0IF = B'01 in CS0CTRL).
 2. Set bits A0BST2 to A0BST0 to an appropriate value so that (area 0 bus size) x (burst length set in this register) becomes 32 bytes or less.
 3. For details, refer to section 25.3.2, CPU (AXI Bus) → Burst ROM Interface.

25.2.9 Area 0 Burst Pitch Set Register (CS0BTPH)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CS0BTPH specifies the burst pitches for the first access cycle and the second and later cycles for area 0 when the burst ROM interface is selected.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----|----------|-----|-----|-----|----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | A0H | A0W[3:0] | | | — | A0B[2:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | A0H | B'0 | R/W | Specifies the CS# and address hold cycles with respect to the RD# signal for area 0 in the burst ROM interface. 0: 0 cycle for hold period 1: 1 cycle for hold period |
| 7 to 4 | A0W[3:0] | B'1111 | R/W | These bits specify the burst pitch (wait cycles to be inserted) after the first burst cycle for area 0 in the burst ROM interface. B'0000: Setting prohibited B'0001: Setting prohibited B'0010: 2 cycles B'0011: 3 cycles B'0100: 4 cycles : B'1101: 13 cycles B'1110: 14 cycles B'1111: 15 cycles |
| 3 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 2 to 0 | A0B[2:0] | B'111 | R/W | These bits specify the burst pitch (wait cycles to be inserted) after the second burst cycle for area 0 in the burst ROM interface. B'000: Setting prohibited B'001: 1 cycle B'010: 2 cycles : B'110: 6 cycles B'111: 7 cycles |

Notes: 1. Be sure to specify this register before specifying CS0BSTCTL.
2. For details, refer to section 25.3.2, CPU (AXI Bus) → Burst ROM Interface.

25.2.10 Area 1 Guard Setting Register (CS1GDST)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CS1GDST specifies the guard interval (period of access prohibition) between sequential access cycles in area 1.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|-------|-----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | CS1GD | TIMER_SET | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | CS1GD | B'0 | R/W | 0: Makes the TIMER_SET setting invalid. 1: Makes the TIMER_SET setting valid. |
| 3 to 0 | TIMER_SET | B'0000 | R/W | Guard Interval (Period of Access Prohibition) between Sequential Access Cycles for Area 1 B'0000: 0 clock cycle B'0001: 1 clock cycle B'0010: 2 clock cycles B'0011: 3 clock cycles B'0100: 4 clock cycles : B'1101: 13 clock cycles B'1110: 14 clock cycles B'1111: 15 clock cycles |

- Notes:
- The TIMER_SET setting is ignored when CS1GD = 0.
 - This register must not be dynamically modified regardless of whether area 1 is being accessed.
 - The actual guard interval between sequential access cycles on the EX-BUS is (register setting) + (idle cycles due to hardware processing).
 - For details, refer to section 25.3.1 (4), Controlling Guard Intervals

25.2.11 BSC Interrupt Source Status Register (BCINTSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: BCINTSR indicates the status of the BSC interrupt source.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | EXWTE | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | EXWTE | B'0 | R | EX-BUS Wait Timeout Error Status 0: The EX-BUS is working correctly. 1: An EX-BUS timeout error has occurred. (A timeout error occurs when an EX-BUS clock (CLKOUT) cycle of EXBCT and EXTCT setting values have elapsed.) |
| 0 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

25.2.12 BSC Interrupt Source Clear Register (BCINTCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: BCINTCR clears the state of the BSC interrupt indicator.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | EXWTE C | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | —/WC1 | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-------|--|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | EXWTEC | B'0 | —/WC1 | EX-BUS Wait Timeout Error Status Clear 0: Writing 0 is ignored. 1: Clears the EX-BUS wait timeout error state. |
| 0 | — | B'0 | R | Reserved The write value should always be 0. |

Note: This register is always read as 0.

25.2.13 BSC Interrupt Enable Register (BCINTMR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: BCINTMR enables or disables the BSC interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | EXWTE M | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | EXWTEM | B'0 | R/W | EX-BUS Wait Timeout Error Interrupt Enable 0: Disables output of an interrupt signal for this interrupt source. 1: Enables output of an interrupt signal for this interrupt source. |
| 0 | — | B'0 | R | Reserved The write value must always be 0. Operation is not guaranteed if setting 1. |

25.2.14 External Wait Status Register (EXWTSTS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: EXWTSTS indicates the state of the external wait pins EX_WAIT0.

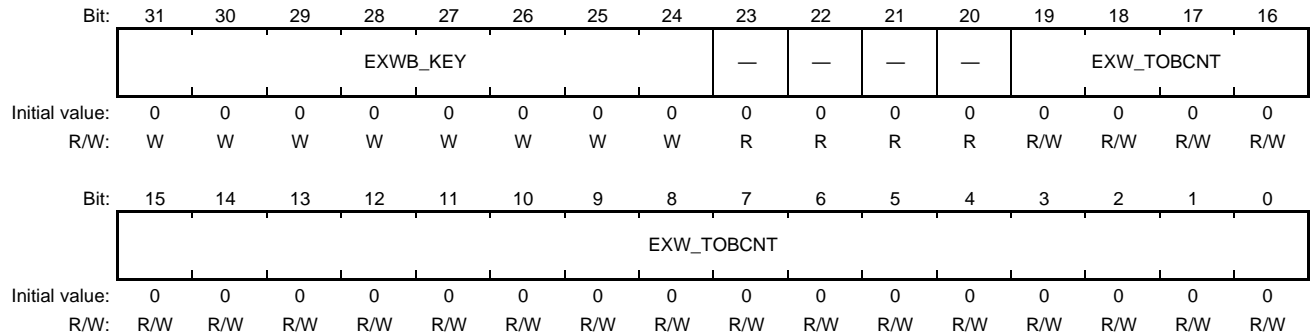
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | EXWT0 STS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | — | — | R | Reserved Writing to this bit is prohibited. |
| 1 | — | — | R | Reserved Writing to this bit is prohibited. |
| 0 | EXWT0STS | — | R | Indicates the EX_WAIT0 pin state. |

25.2.15 EX-BUS Wait Timeout Detection Base Counter Register (EXBCT)

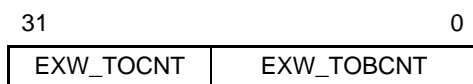
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: EXBCT specifies the lower-order part of the value for counting to detect a timeout in waiting for access to the EX-BUS, which is monitored through pins EX_WAIT0.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 24 | EXWB_KEY | H'00 | W | EX-BUS Wait Timeout Detection Base Counter Register Write Key For writing to this register to activate, the value H'A5 must be written to these bits. Values read from these bits are meaningless. |
| 23 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 to 0 | EXW_TOBCNT | H'0_0000 | R/W | EX-BUS Wait Timeout Counter Setting Maximum value: H'0_0000 Minimum value: H'0_0001 |

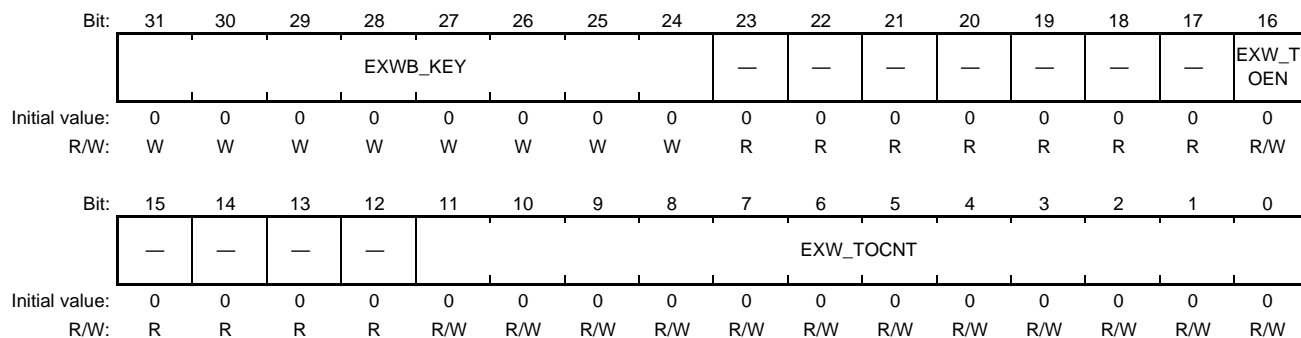
Note: Counting to detect a timeout in waiting for access to the EX-BUS is handled by a 32-bit counter, which is formed by the EXW_TOBCNT bits in the EXBCT register and the EXW_TOCNT bits in the EXTCT register as shown below.



25.2.16 EX-BUS Wait Timeout Detection Counter Register (EXTCT)

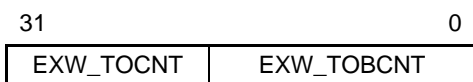
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: EXTCT specifies the higher-order part of the value for counting to detect a timeout in waiting for access to the EX-BUS, which is monitored through pins EX_WAIT0, and enables or disables the detection.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 24 | EXWB_KEY | H'00 | W | EX-BUS Wait Timeout Detection Counter Register Write Key For writing to this register to be effective, the value H'5A must be written to these bits. Values read from these bits are meaningless. |
| 23 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | EXW_TOEN | B'0 | R/W | EX-BUS Wait Timeout Enable 0: Timeout in waiting for access to the EX-BUS has not been detected. 1: Timeout in waiting for access to the EX-BUS has been detected. |
| 15 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 0 | EXW_TOCNT | H'000 | R/W | EX-BUS Wait Timeout Counter Setting Maximum value: H'000 Minimum value: H'001 |

Note: Counting to detect a timeout in waiting for access to the EX-BUS is handled by a 32-bit counter, which is formed by the EXW_TOBCNT bits in the EXBCT register and the EXW_TOCNT bits in the EXTCT register as shown below.



25.2.17 EX-BUS Wait Timeout Detection Access Source Indication Register (EXTSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: EXTSR indicates the source of attempted access to the EX-BUS that reached timeout as detected through the EX_WAIT0 pins.

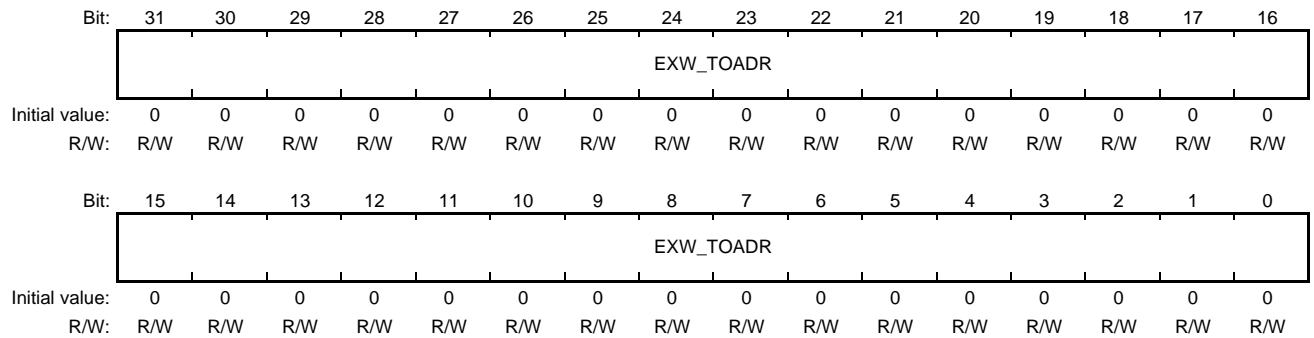
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | EXW_T OSHW |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/WC1 |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-------|--|
| 31 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | EXW_TOSHW | B'0 | R/WC1 | Indication of timeout in waiting for access to the EX-BUS from the AXI bus. 0: Timeout has not been reached. 1: Timeout has been reached. To clear this bit, write 1 to it. |
| 15 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

25.2.18 EX-BUS Wait Timeout Detection Address Indication Register (EXTADR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: EXTADR indicates the address at which access to the EX-BUS was attempted but timeout was detected through pins EX_WAIT0.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | EXW_TOADR | H'0000_0000 | R/W | Indication of the address for which attempted access to the EX-BUS by the AXI bus reached timeout |

Note: This register is readable and writable. To clear this register, write 0s to all bits.

25.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

25.3.1 SRAM Interface (Basic Functionality)

The BSC reads access requests from the CPU stored in the FIFO mounted in the LBSC module and writes them to the EX-BUS. By default, all spaces, areas 0 and 1 are all set for SRAM interface. The pulse width for access signals in this SRAM access interface can be varied according to register settings. Thus, the SRAM interface has functionality for easily accommodating devices with various access specifications, connected on the EX-BUS. In addition, to support low-speed external devices, the guard-interval control functionality is provided for insertion of the appropriate interval for each bus access; and the SRAM interface receives wait-for-response requests (or access complete signals) from external devices on a synchronous/asynchronous-selectable and polarity-selectable basis, thus ensuring flexibility in bus design. Figure 25.3 and Figure 25.4 show SRAM interface timing charts for AXI → EX-BUS conversions.

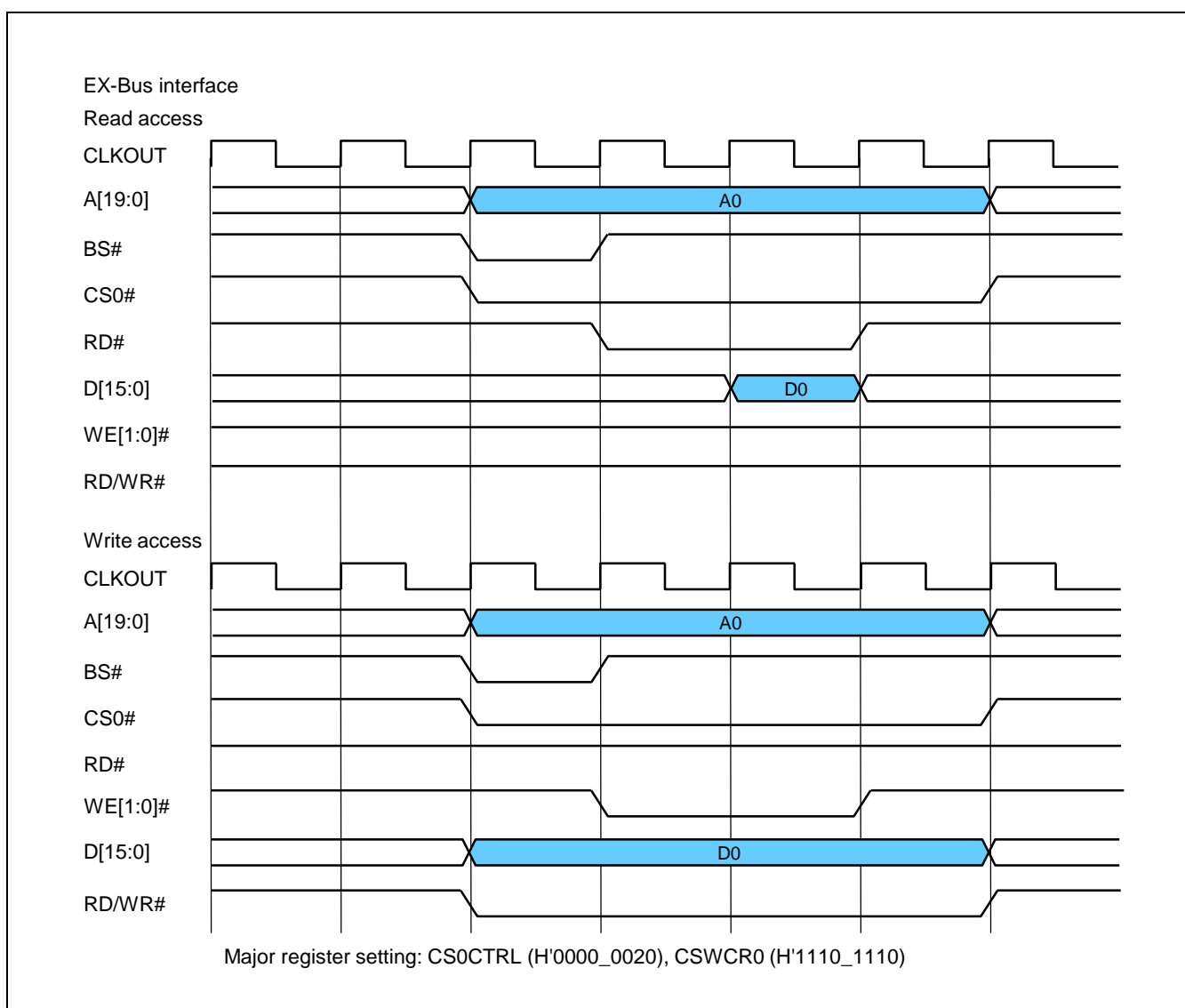
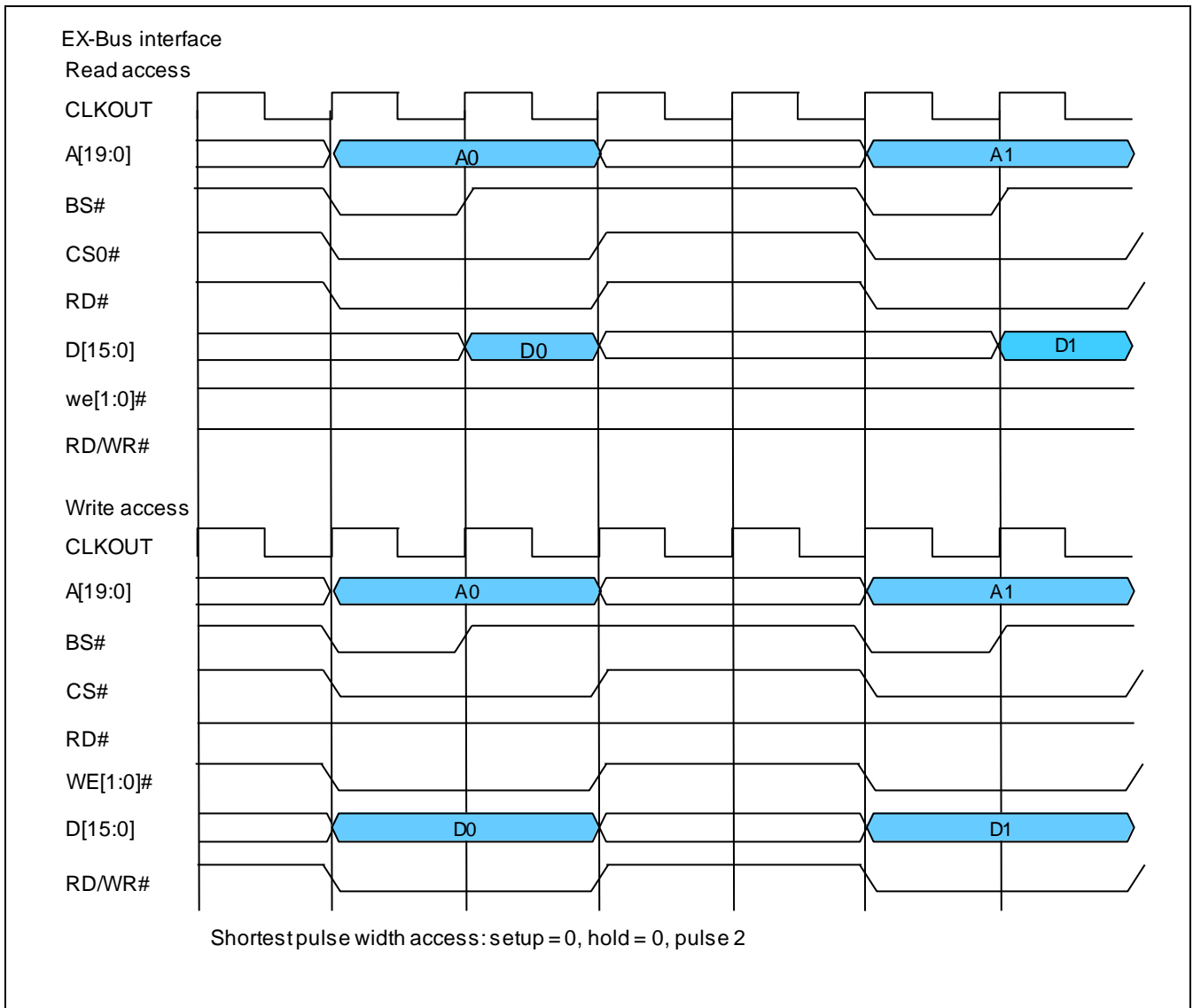


Figure 25.3 Basic Timing Chart for Access from AXI to SRAM (Area 0)



**Figure 25.4 Basic Timing Chart for Access from AXI to SRAM
(Shortest Pulse Width × Shortest PIO Consecutive Accesses)**

Figure 25.4 is an example of the basic SRAM interface waveform with the shortest pulse width, wherein the same waveform occurs twice in succession.

If consecutive PIO access requests are made from the CPU (AXI bus), the access interval will be 2 clock cycles, as indicated in the above waveforms, regardless of whether the preceding or succeeding access request is for the same area or for different areas. The access interval can be extended by means of the guard setting registers.

(1) Address Generation/Alignment

When making access to the EX-BUS as an SRAM interface Figure 25.5 provides an overview of address generation and data alignment/write enable conversions.

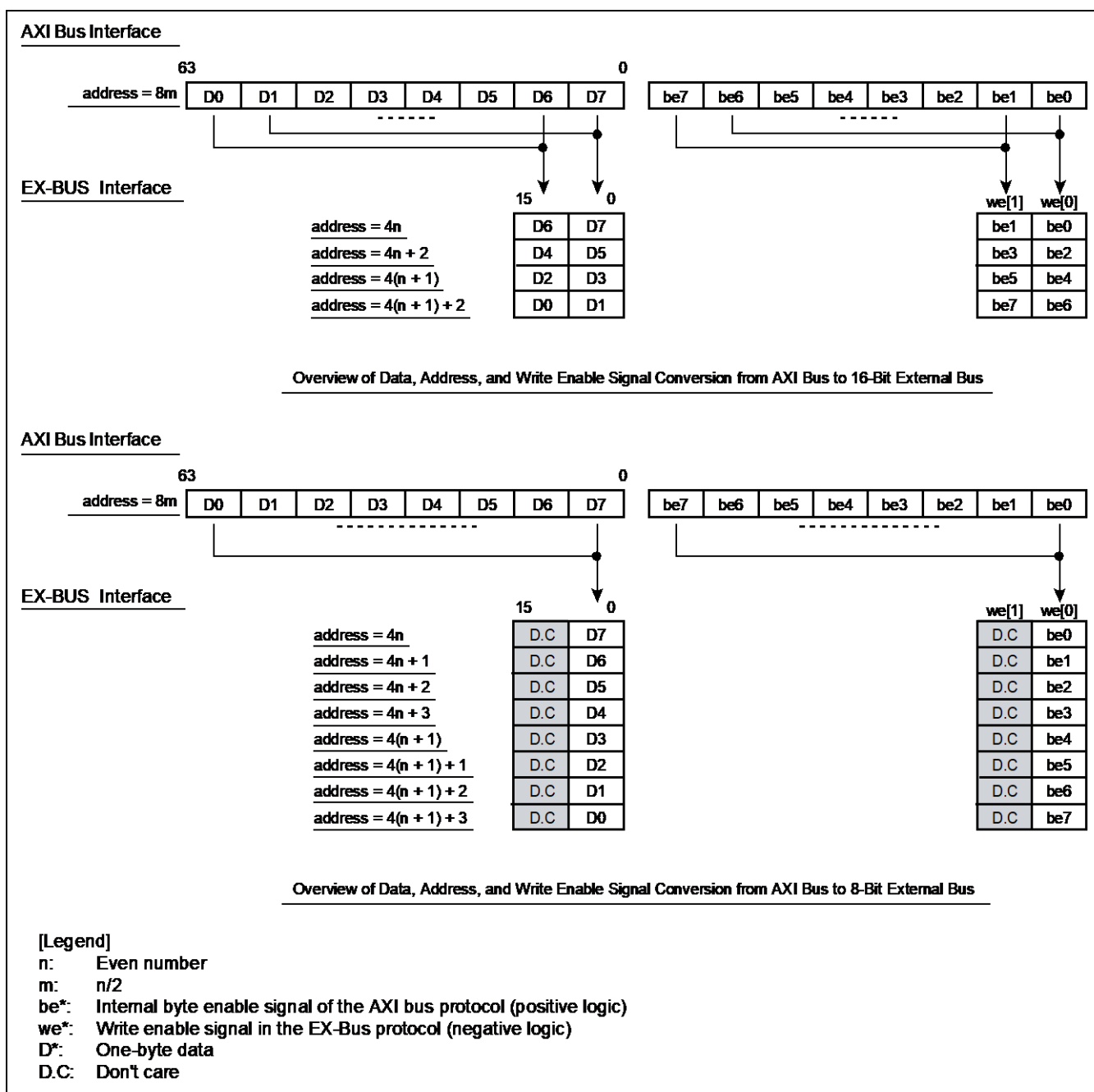


Figure 25.5 Overview of Data, Address, and Write Enable Signal Conversion from CPU (64-Bit AXI Bus) to External Bus

(2) Setting the Pulse Width for an Access Signal

When making access to the EX-BUS, the LBSC can set the setup or hold time for addresses, RD/WR#, and CS# signals based on WE# signals and RD# signals, and the pulse width for WE# and RD# signals in units of clock cycles, according to the values that are set for each area by means of the CSWCR0, CSWCR1 registers. (During a burst ROM read operation, however, the LBSC conforms to CS0BTPH settings rather than CSWCR0 settings. The pulse width for WE# or RD# signal that is stored in a register can be extended by an externally supplied EX_WAIT signal. The minimum total value that is set should be two clock cycles.

(3) External Wait Control

The LBSC controls the external wait signal (EX_WAIT) from a device connected to the EX-BUS based on settings that are provided on external wait control registers (CSPWCR0, CSPWCR1) and the external wait input control register (EXWTSYNC). The external wait control registers permit the selection of the four types of interfaces to accommodate various types of specifications, whether the wait signal input from an external device is based on the READY logic (posting a READY status) or BUSY logic (posting a BUSY status), or which signal polarity is in effect.

Figure 25.6 shows waveforms for the four wait signal patterns that can be input and the waveforms for the wait signals internal to the LBSC after conversion according to the register settings.

The external wait input control register allows the switching between the synchronous/asynchronous handling of external wait input signals. The default is to treat such signals on a synchronous basis. Figure 25.7 shows external wait input timings for synchronization/asynchronization. In the figure, the position indicated by the symbol * represents the point at which the LBSC determines whether or not external wait is in effect. Specifically, for synchronization, the position is one clock cycle before the point at which the WE# and RD# signals would normally be negated by pulse width settings. If external wait is in effect at this position, the pulse width for the WE# or RD# signal continues to extend until the wait status becomes the ready status. If the pulse width for the WE# or RD# signal is extended by the EX_WAIT signal, the address and the RD/WR# and CS# signals are negated when the hold time is satisfied. If the synchronization setting is switched to the asynchronization setting, any external wait is nullified unless there is an external wait two clock cycles before the * position, or three clock cycles before the point in which the WE# and RD# signals would normally be negated.

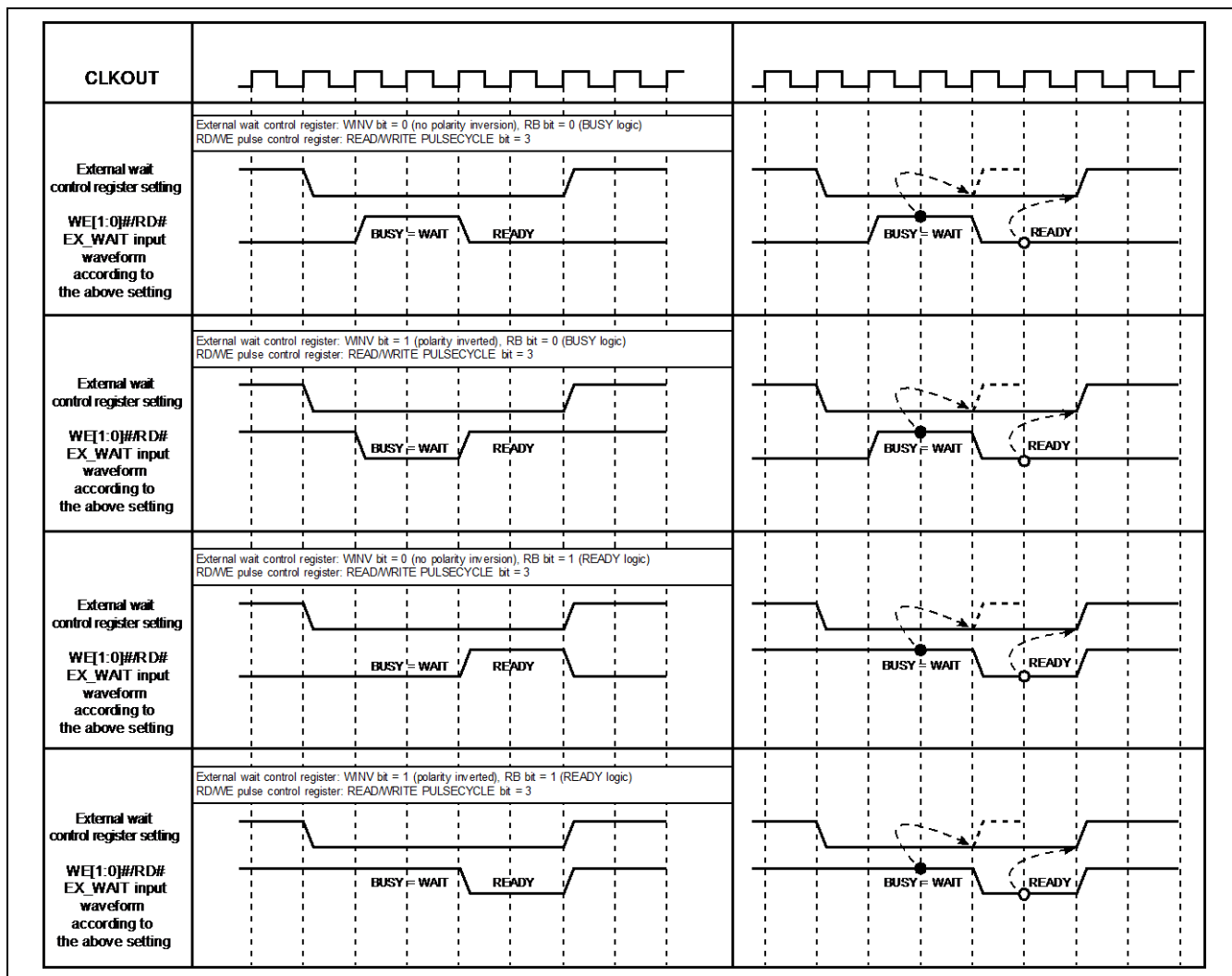


Figure 25.6 Waveforms of Converted External Wait Interface

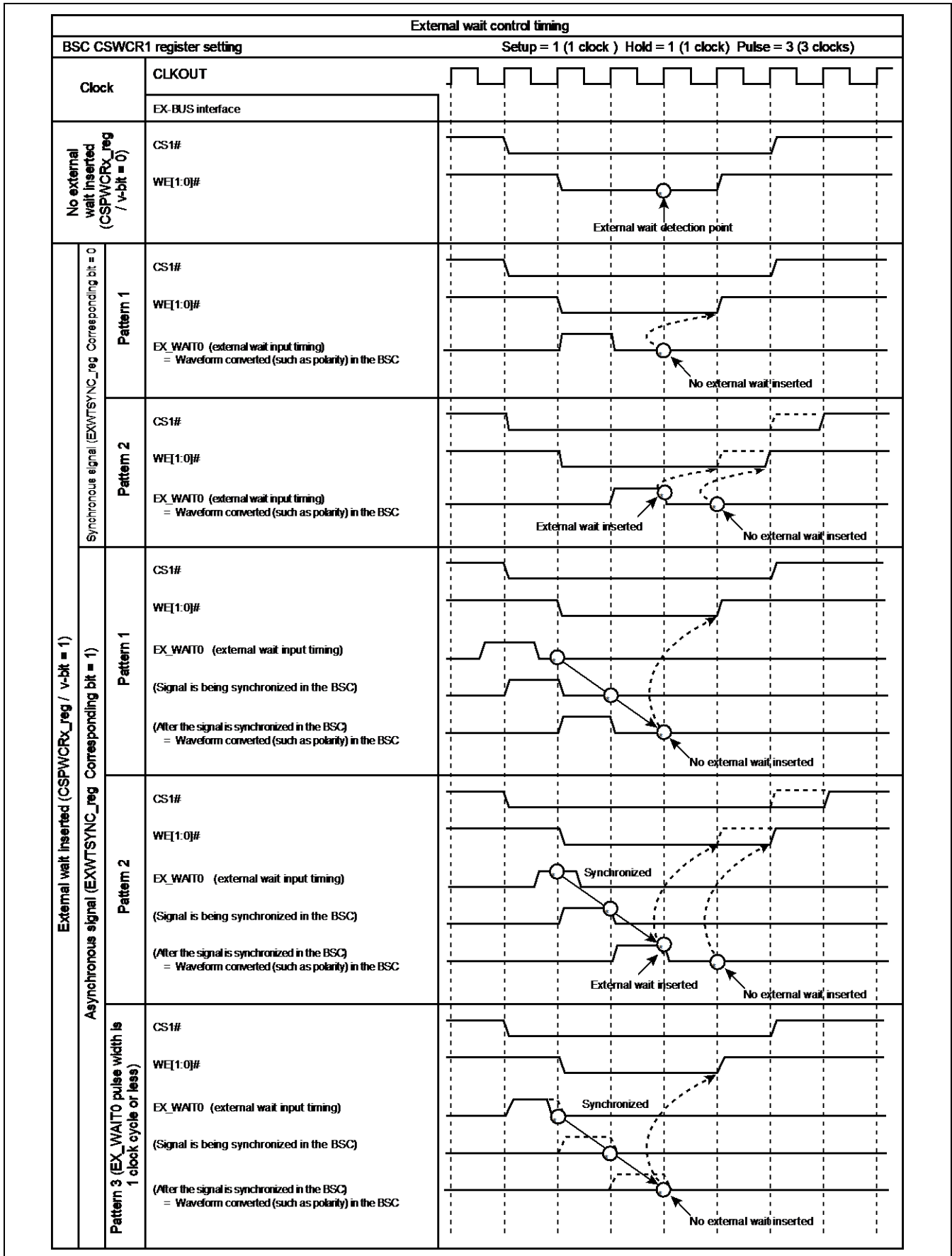


Figure 25.7 Waveforms of External Wait Input Signals

(4) Controlling Guard Intervals

For some external devices, the transition of the data lines to the high-Z state after completion of read access takes a relatively long time. To prevent any contention for the data bus when further access over the EX-BUS proceeds immediately after a read operation for such a device, the LBSC supports a function to guard against further access over a given interval following each bus access. The function is also effective for external devices that have difficulty coping with consecutive reception of access requests. Specifically, set the number of clock cycles over which guarding against access applies in the guard setting register (CS1GDST,) for the given area. (Even if the value assigned to the guard setting register is 0, hardware processing before a further bus access request is issued requires at least 2 clock cycles between PIO and PIO. Therefore, in PIO accesses, the actual interval on the EX-BUS is 2 clock cycles when value in the guard setting register is 0 or 1. And it is 1 clock cycles plus the setting for number of clock cycles in the guard setting register when the setting value is 2 or more.)

After access to a given area, the guard interval is assigned corresponding to the register for that area, regardless of the kind of access or target space of the next access request. Note, however, that there is no guard setting register for area 0. Therefore, guard intervals are not set up for access that immediately follows access to this area. Since area 0 is supposed to be for the connection of general-purpose memory such as ROM, SRAM, or flash-ROM, guard intervals should not be necessary for this area. Figure 25.8 is a basic timing chart to illustrate the concept of guard-interval control.

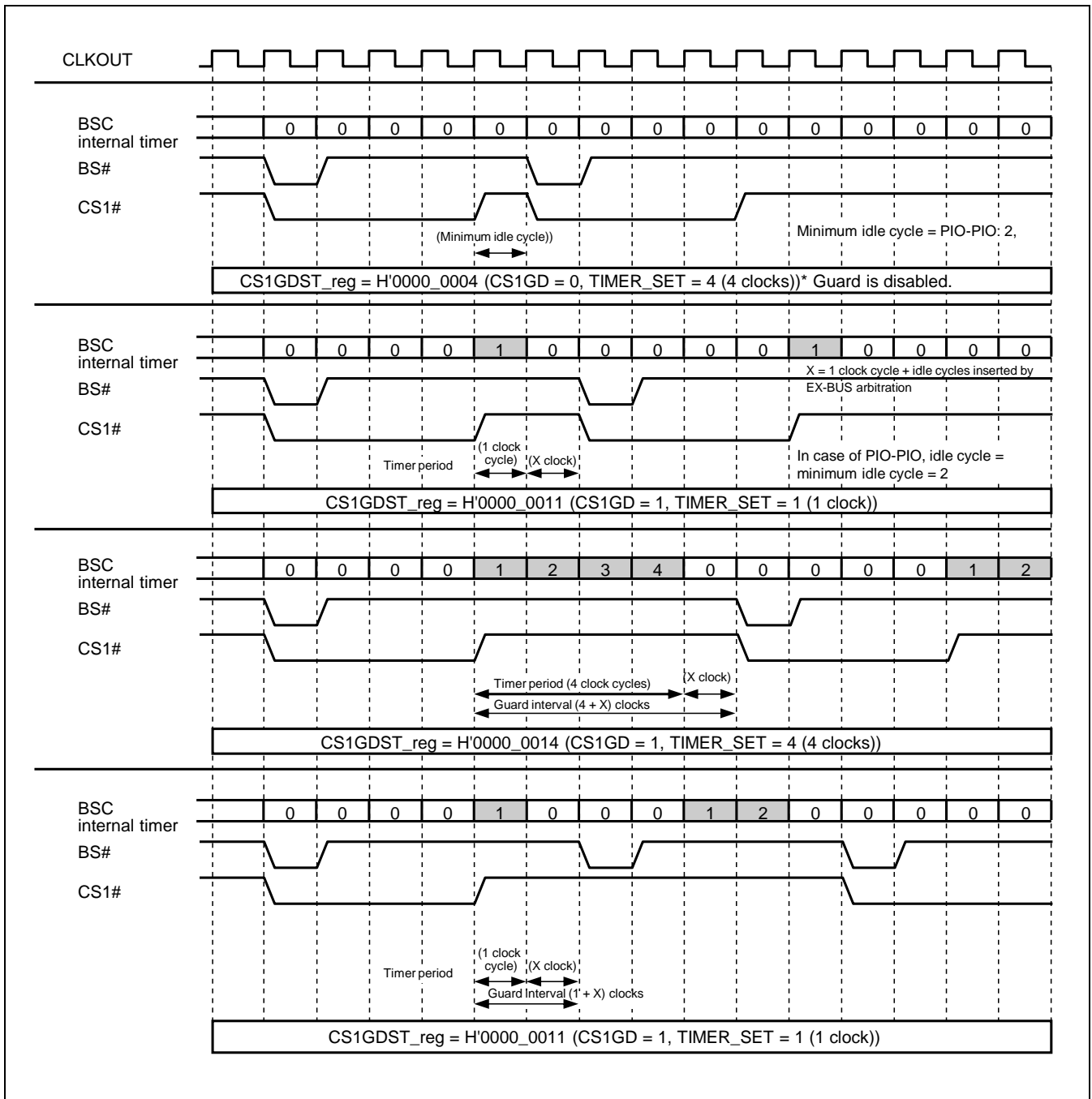


Figure 25.8 Concept of Guard-Interval Control

25.3.2 CPU (AXI Bus) → Burst ROM Interface

For area 0 on the EX-BUS, the BSC supports a page-mode read burst ROM interface. Switching to the burst ROM interface is performed by CS0CTRL settings. Because burst ROM access operations begin immediately after setting CS0CTRL, values must be pre-assigned to the CS0BTPH and CS0BSTCTL. In CS0BTPH, burst ROM access waveforms can be assigned, such as a first-cycle burst pitch, second and subsequent cycle burst pitches, and CS# signal hold cycles on the RD# signal. A burst access count (burst length) can be assigned to CS0BSTCTL. A burst access count should be assigned according to the component specifications for the actual external ROM that is connected.

In two kinds of exceptional cases, however, the BSC handles access with a burst count less than the specified burst count. The first are those cases where the CPU requests access to fewer data than the specified burst count. In such cases, access is terminated on completion of processing for the number of bytes requested by the CPU, even though the specified burst count has not been reached. The second are those cases where a variable burst address boundary is crossed in the midst of a specified burst count. In such cases, access is terminated immediately before the boundary is crossed, and the burst access is divided up, with the remainder of the access executed in a second round. The reason for this behavior is that the variable address boundaries for burst ROM devices, by which the LBSC must abide, are defined. Table 25.3 shows the LBSC's methodology for breaking off burst access.

Table 25.3 Methodology for Breaking off Bust Access

| Burst Count Assigned in CS0BSTCTL | Bus Width | |
|--------------------------------------|--------------------------|---|
| | 8-Bit Bus Width | 16-Bit Bus Width |
| 4 | Break at a change of A2 | Break at a change of A3 (A0 not connected) |
| 8 | Break at a change of A3 | Break at a change of A4 (A0 not connected) |
| 16 | Break at a change of A4 | Break at a change of A5* (A0 not connected) |
| 32 | Break at a change of A5* | (Setting prohibited) |

Note: * Since A5 never changes in the midst of a 32-byte burst operation (a 32-byte boundary), access is not broken off in the case indicated by *, and the specified number of access operations proceed.

As described above, in burst ROM access, the LBSC performs a maximum of 32-byte access to accommodate CPU cache fill while performing access breakup. However, because this operation involves access on 32-byte boundaries, for setting a burst ROM device, the user should specify wrap-around settings instead of continuous burst operations.

Depending on the mode of burst ROM connection, burst operations can also be accommodated in the synchronous burst mode. When using the LBSC in synchronous burst mode, the user should not receive the Wait signals that are output by the burst ROM. Instead, a first-word data output latency should be assigned to the configuration register of the burst ROM, and CS0BTPH should be set as the pitch for the first word and this latency. For the second and subsequent words, a pitch count of 2 should be assigned to both the configuration register of the burst ROM and CS0BTPH. Subject to the AC characteristics of the burst ROM and given frequency for the EX-BUS, if a pitch count of 1 is assigned to, data reception timing can lag behind due to the fact that the burst ROM has a data output delay of approximately 13ns. Figure 25.9 shows a basic timing chart for the burst ROM interface.

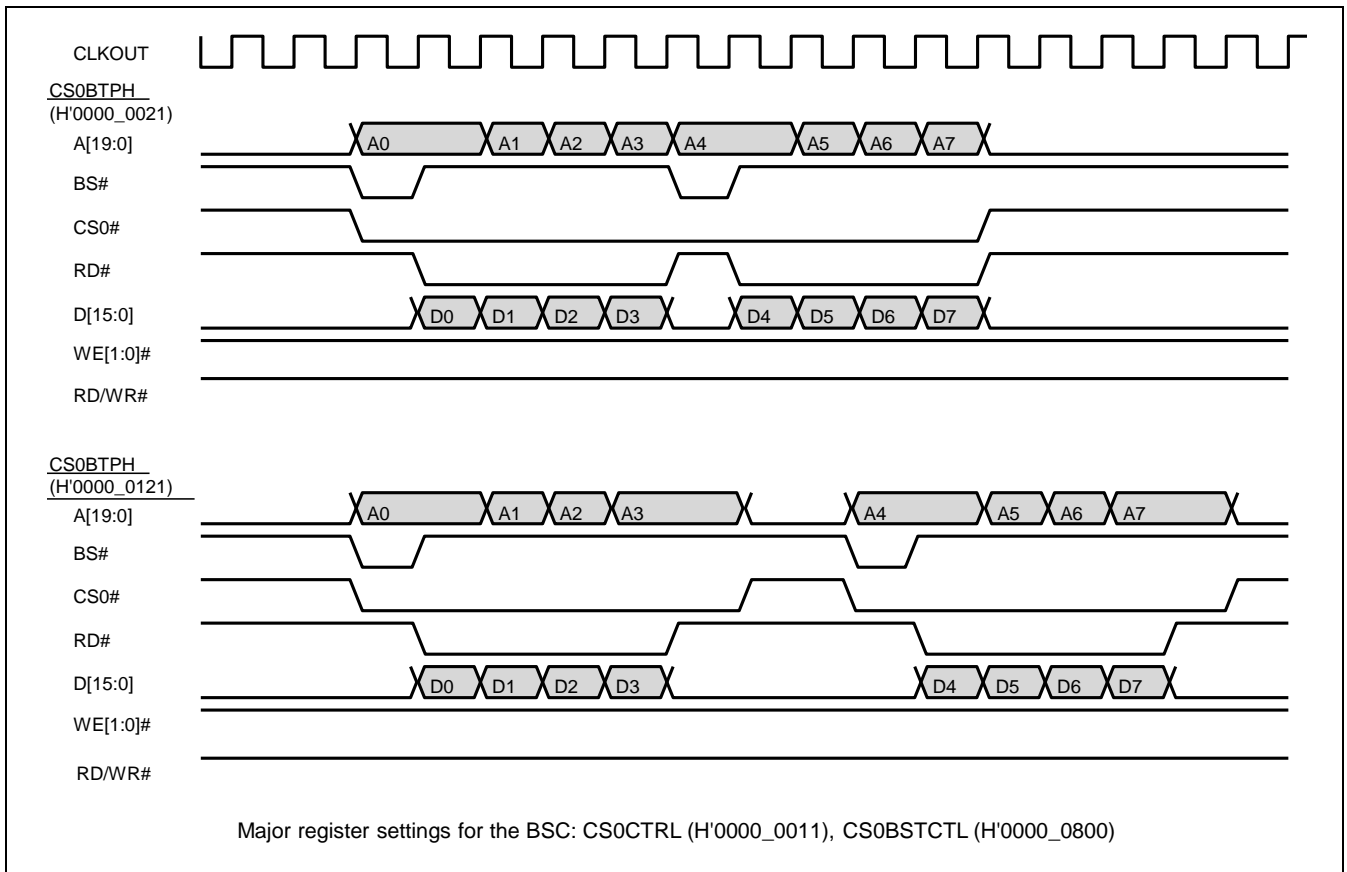


Figure 25.9 Timing Chart for Burst ROM Interface

25.3.3 CPU (AXI Bus) → Byte Control SRAM Interface

With respect to EX-BUS area 1, the LBSC supports byte-control SRAM interface from the CPU (AXI bus). Byte-control SRAM interface is a memory interface in which byte select strobe signal WE# is output in both read and write cycles. Write timing for byte-control SRAM interface is the same as that for SRAM interface. In read access, however, WE# is output at a different timing. In read access, WE# is output for a read byte only. The assertion timing of WE# can be selected with the CS1BRM bit in the CS1CTRL register to be the same as that of CS# or RD#. In default setting, SRAM operating mode is set for each area. Accordingly, along with the setting of assertion timing, byte-control SRAM mode should be set in the registers corresponding to a given area (CS1CTRL).

(1) Byte-control SRAM Interface

- The WE# assertion period can be selected by the CS1BRM bit to be the same as that of CS# or RD#.
- In read access, WE# is asserted only for a valid access byte. (For example, when a byte is read from a device with the byte-control SRAM interface and a bus width of 16 bits, any one of the WE# bits is asserted.)
- The waveform in write access is the same as that for SRAM interface.

(2) Basic Timing Charts

Figure 25.10 shows the basic timing chart for byte-control SRAM interface.

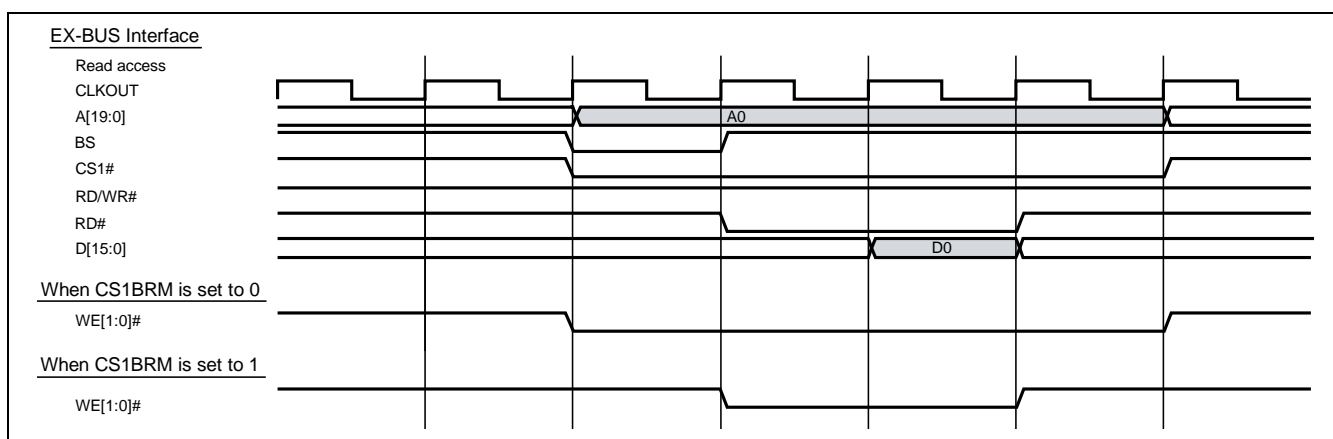


Figure 25.10 Waveforms for Byte-Control SRAM Interface

25.3.4 Wait Timeout

For the detection of timeout in waiting for access to the EX-BUS, the BSC monitors the states of the EX_WAIT0 signals* from external devices. When an external device is waiting for more than a certain length of time, this is detected as an EX-BUS wait timeout error. The error is reflected in a register within the BSC, and forces the termination of access to the EX-BUS interface.

The value of the timeout counter is specified in a total of 32 bits, consisting of the EXW_TOBCNT bits in the EX-TIME wait timeout detection base counter register (EXBCT) and the EXW_TOCNT bits in the EX-BUS wait timeout detection counter register (EXTCT). EXW_TOBCNT is a 20-bit counter that operates at the EX-BUS operating frequency (CLKOUT). EXW_TOCNT is a 12-bit counter that counts up by one every time the EXW_TOBCNT bits overflow.

Writing 0s to all of the bits specifies the maximum time until overflow. The required detection time depends on the EX-BUS operating frequency as defined in the following formula.

$$\text{Detection time in ns} = \text{period in ns at EX-BUS operating frequency} \times \{\text{EXW_TOCNT}, \text{EXW_TOBCNT}\}$$

25.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Keep the following notes in mind when using this LBSC.

25.4.1 Pin Multiplexing

When starting this LSI, be sure to set multiplexed pins appropriately using the LSI pin multiplexing set register.

Also, make pull-up resistor settings using the LSI pin pull-up control registers.

For details on the specific set values, refer to section 8, Pin Function Controller (PFC).

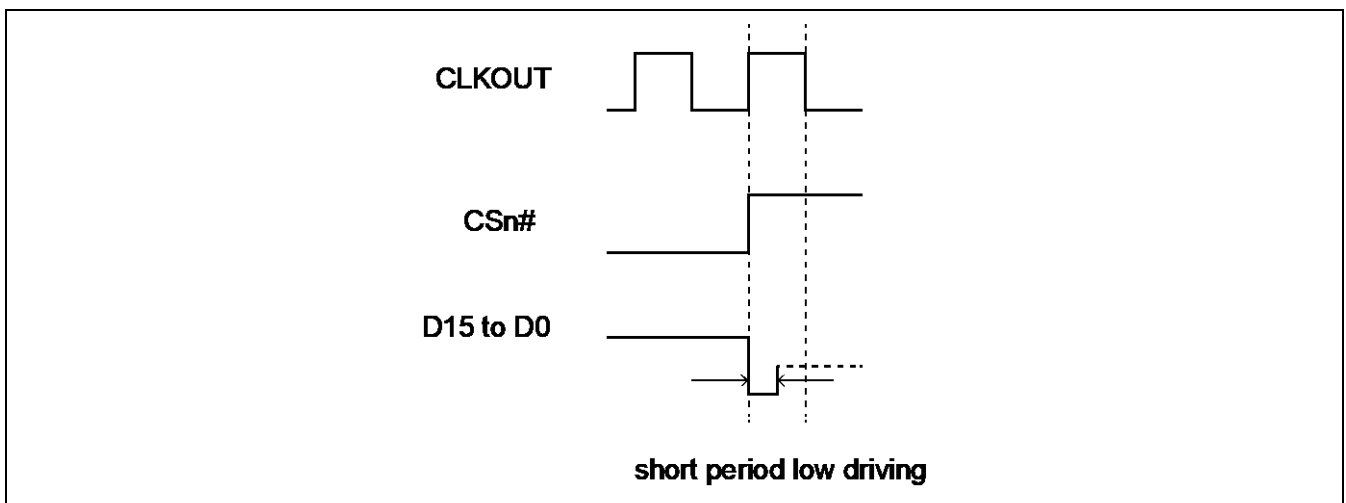


Figure 25.11

When using QSPI or HSCIF booting by mode pins setting (MD [3:1]), the area 0 beginning 256-kByte space (H'00_0000 0000 to H'00_0003_FFFF) cannot be accessed from the LBSC even after booting.

26. External Bus Controller for LPDDR4/DDR3L SDRAM (DBSC4)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

26.1 Overview

The external bus controller (DBSC4) supports DDR3L SDRAM and LPDDR4 SDRAM. The DBSC4 contains AXI bus controller, cache, a scheduler, device controller, and a register controller. The AXI bus controller receives requests from multiple AXI buses and performs protocol conversion to suit the bus in the DBSC4. The scheduler makes schedules for the requests stored in the cache block. The device controller converts requests that follow the internal bus protocol to command-level requests, and reads from and writes to the SDRAM in response to commands. The register controller reads from and writes to the control registers that set the operations of the DBSC4.

The DBSC4 enables the maximum use of SDRAM bus bandwidth using the following functions:

- (1) Multibank operation that improves the page hit rate.
- (2) Preceding execution of the bank activate commands for subsequent requests.

26.1.1 Definition of Terms and Symbols

The following terms and symbols are used throughout the DBSC4 specifications:

- In some cases, DDR3L/ LPDDR4 SDRAM is abbreviated as SDRAM.
- $\text{ceil}(x)$ is the smallest integer which is greater than or equal to real number x .
- $\text{floor}(x)$ is the greatest integer which is less than or equal to real number x .

RZ/G2H has the following two modes.

- (1) Memory area swap mode.
- (2) Memory area non-swap mode.

These modes are abbreviated as follows in this section.

- (1) RZ/G2H: Memory area swap and non-swap mode. Both modes are common.
- (2) RZ/G2H (Swap): Memory area swap mode.
- (3) RZ/G2H (Non-swap): Memory area non-swap mode.

Table 26.1 DBSC4 Functions

| Item | Function |
|--|--|
| Multibank supported | Supports 8-bank multibank operation per ranks per channels |
| Number of banks | Supports eight banks |
| External bus width | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]: 32 bits x 2 channels [RZ/G2N]: 32 bits x 1 channel [RZ/G2E]: 32 bits x 1 channel, 16-bit bus width x 1channel |
| Preceding activate functions | Determines the content of subsequent requests in a request queue and performs preceding activate processing for the bank to be accessed during an empty command cycle upon page-miss. |
| Operating modes | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]: Burst length: LPDDR4:16 (fixed) Burst type: Sequential (fixed) [RZ/G2E]: Burst length: DDR3L:8 (fixed) Burst type: Sequential (fixed) DLL-off mode for the DDR3L-SDRAM is not supported. |
| Power-down mode | Supports self-refresh mode and precharged power-down mode. Combined use of the power-down mode and address translation function allows the SDRAM partial array self-refresh function to be effectively implemented. |
| Address order (address translation function) | The addresses arranged in descending order are: rank address, row address, bank address, and column address. (The 3-bit bank address can be divided into two parts. One can be placed in the upper half of the row address and the other in the lower half. Register settings can be used to determine how to divide the bank address.) |

| Item | Function |
|--------------------------|---|
| Timing setting | <p>The following timing settings can be specified:</p> <p>CAS latency, CAS write latency, ACT to READ or ACT to WRITE minimum period, PRE period, ACT to ACT or ACT to REF minimum period, ACT to PRE minimum period, ACT(A) to ACT(B) minimum period, four active window minimum period, READ to PRE minimum period, write recovery period, READ to WRITE minimum period, WRITE to READ minimum period, REF to ACT or REF to REF (all banks) minimum period, minimum period over which CKE is held high, and minimum period over which CKE is held low</p> <p>Only 0 is supported for additive latency (AL).</p> |
| Refreshes | Average interval and the maximum pulling-in count are set by the register. If an empty cycle for request is found, preceding refresh can be performed. |
| ZQ calibration operation | Supports the automatic ZQCS/ZQCL issuing function. After auto-refresh, the ZQCS/ZQCL command is issued to DDR3L-SDRAM. The frequency of issuing ZQCS/ZQCL can be specified through register settings in units of the auto-refresh count. The intervals between auto-refresh and ZQCS/ZQCL and between ZQCS/ZQCL and auto-refresh can also be specified through register settings. |
| PHY interface | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]:</p> <p>32-bit bus width x 2channels</p> <p>[RZ/G2N]:</p> <p>32-bit bus width x 1channel</p> <p>[RZ/G2E]:</p> <p>32-bit bus width x 1channel, 16-bit bus width x 1channel</p> |
| Memory to be connected | <p>[RZ/G2H]:</p> <p>LPDDR4-SDRAM compliant with JEDEC JESD209-4A. (Supports memory with sizes from 4 Gbits to 16 Gbits. One LPDDR4-SDRAM can be connected to one channel)</p> <p>[RZ/G2M V1.3, RZ/G2M V3.0]:</p> <p>LPDDR4-SDRAM compliant with JEDEC JESD209-4A. (Supports memory with sizes from 4 Gbits to 16 Gbits. One LPDDR4-SDRAM can be connected to one channel)</p> <p>2 ranks/channel support</p> <p>[RZ/G2N]:</p> <p>LPDDR4-SDRAM compliant with JEDEC JESD209-4A. (Supports memory with sizes from 4 Gbits to 16 Gbits. One LPDDR4-SDRAM can be connected to one channel)</p> <p>[RZ/G2E]:</p> <p>DDR3L-SDRAM compliant with JEDEC JESD79-3-1A.01. (In the case of SDRAM with a data bus width of 8 bits, memory with sizes from 512 Mbits to 4 Gbits size are supported and four DDR3L- SDRAM can be connected to one channel. In the case of SDRAM with a data bus width of 16 bits, memory with sizes from 512 Mbits to 8 Gbits is supported and one or two DDR3L- SDRAM can be connected to one channel. Note that SDRAM with a data bus width of 4 bits is not supported.)</p> <p>1 rank/channel support.</p> |
| DBI (Data Bus Inversion) | LPDDR4 function support. |
| Cache | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]:</p> <p>64KB x 2</p> <p>[RZ/G2N, RZ/G2E]:</p> <p>64KB x 1</p> |

26.1.2 Block Diagram

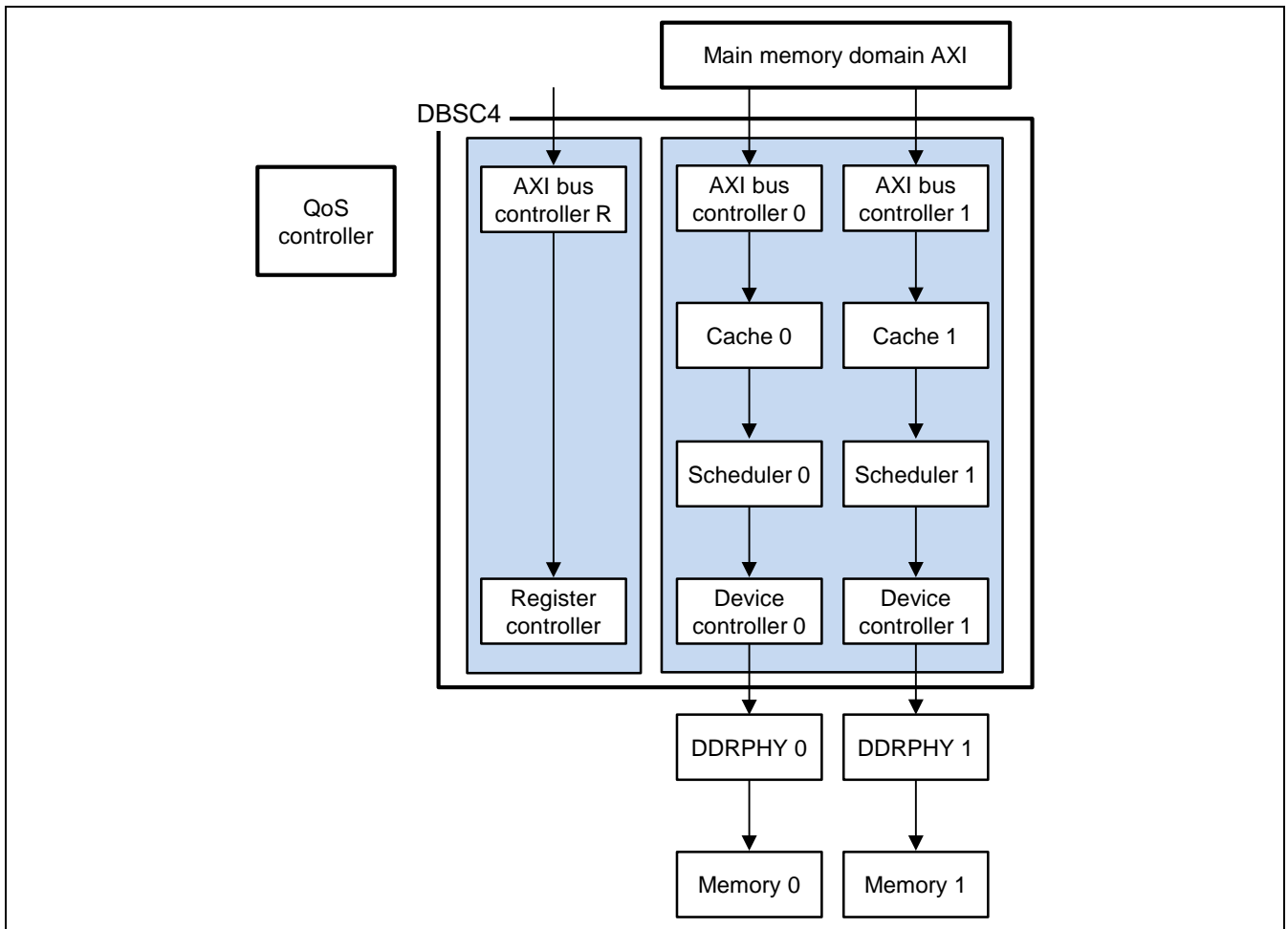


Figure 26.1 Block Diagram of the DBSC4 for RZ/G2M V1.3, RZ/G2M V3.0

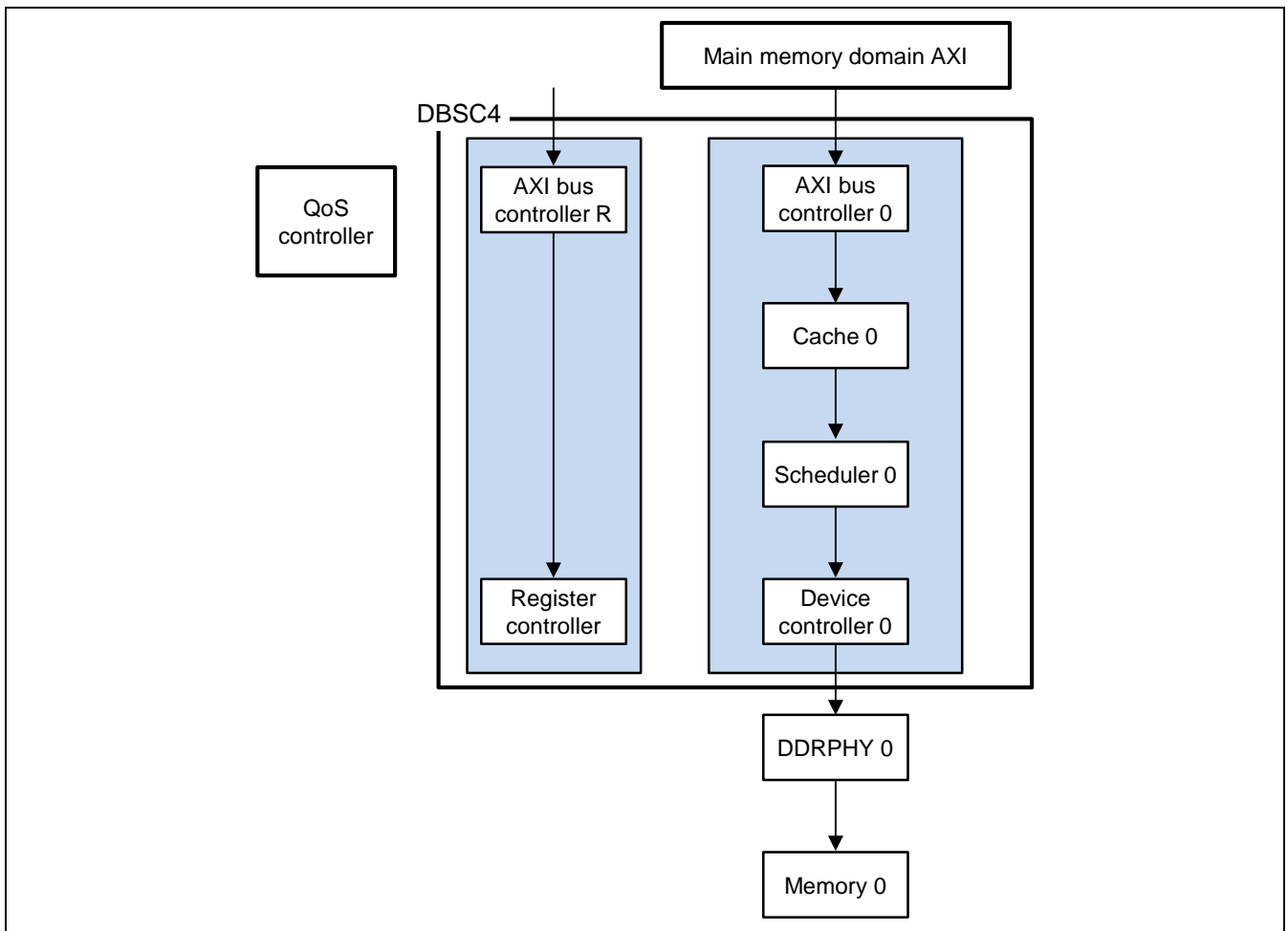


Figure 21.2 Block Diagram of the DBSC4 for RZ/G2N, RZ/G2E

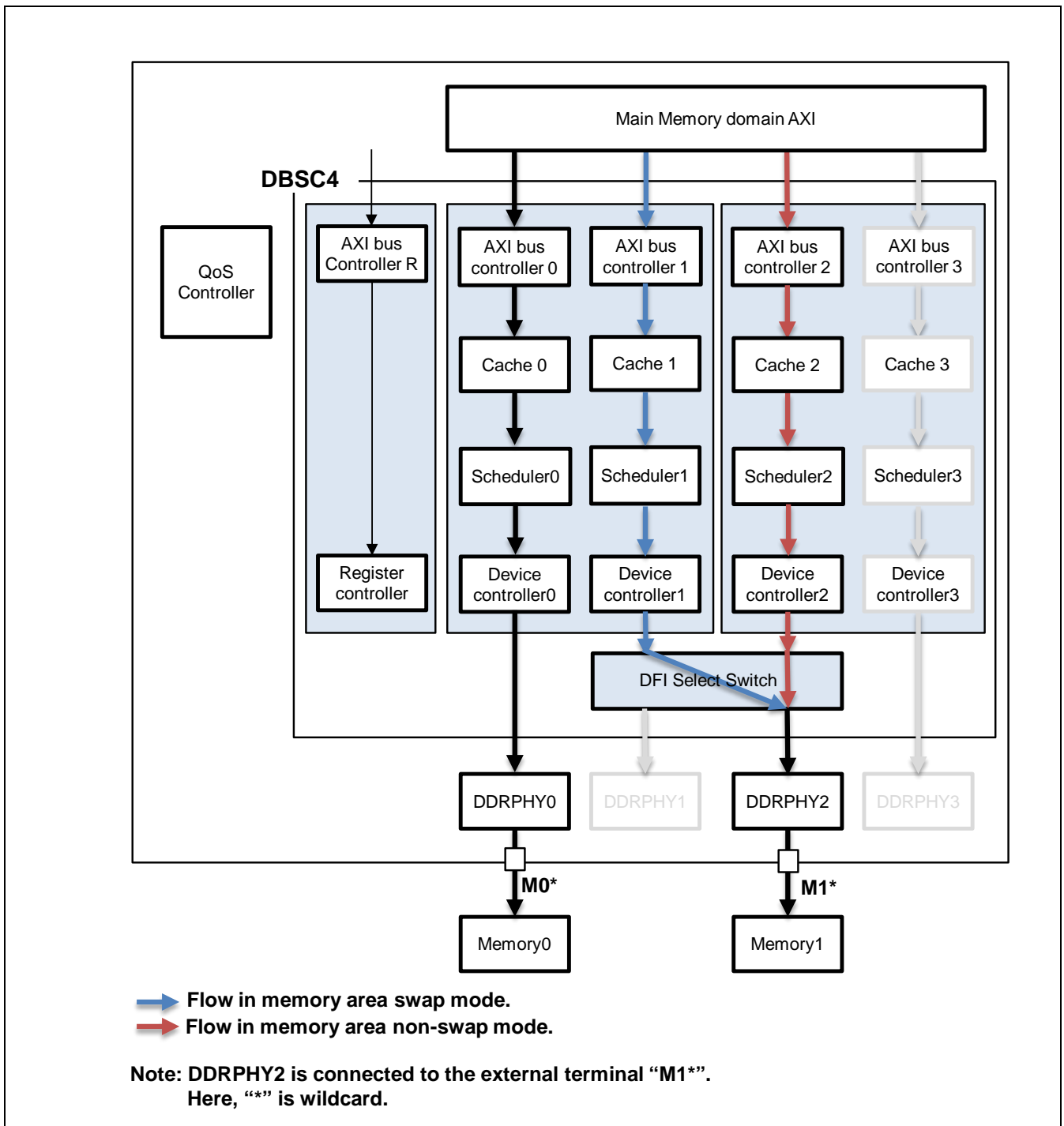


Figure 26.3 Block Diagram of the DBSC4 for RZ/G2H

26.2 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

26.2.1 Activation Sequence

26.2.1.1 Overview

(1) LPDDR4

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | — |

1. After stable power supply, reset release
2. DDRPHY register initialization
3. DDRPHY of PLL / DLL lock start
4. DDRPHY of PLL / DLL lock
5. DDRPHY training operation start
6. DDRPHY training Exit
7. DBSC register initialization
8. SDRAM register initialization
9. Auto-refresh operation start
10. Initialization Exit

(2) DDR3L

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| — | — | — | √ |

1. After stable power supply, reset release
2. DDRPHY register initialization
3. DDRPHY of PLL / DLL lock start
4. DDRPHY of PLL / DLL lock
5. DDRPHY training operation start
6. DDRPHY training Exit
7. DBSC register initialization
8. SDRAM register initialization
9. Auto-refresh operation start
10. Initialization Exit

26.2.2 Self-Refresh Sequence

26.2.2.1 Overview

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

1. Disable DRAM access
2. Precharge all DRAM Banks
3. Issue Self-Refresh Entry Command
4. Disable Auto-Refresh

Use the following procedure to release self-refresh mode.

1. Enable Auto-Refresh
2. Issue Self-Refresh Exit Command
3. Enable DRAM access

26.2.3 SDRAM Power-Supply Backup Mode

26.2.3.1 Overview

(1) M*BKUP control sequence

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | — | √ | — |

1. Disable DRAM access
2. Precharge all DRAM Banks
3. Issue Self-Refresh Entry Command
4. Disable Auto-Refresh
5. Set the M*BKUP signal to high level. Wait for at least 1us.
6. You can turn off the unnecessary power, other than the DDR-IO power.

(2) MBKPRST# control sequence

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| — | — | — | √ |

1. Disable DRAM access
2. Precharge all DRAM Banks
3. Issue Self-Refresh Entry Command
4. Disable Auto-Refresh
5. Set the MBKPRST# signal to low level. Wait for at least 1us.
6. You can turn off the unnecessary power, other than the DDR-IO power.

26.2.4 Recovery from SDRAM Power-Supply Backup Mode

26.2.4.1 Overview

(1) LPDDR4

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | — |

1. After stable power supply, reset release
2. DDRPHY register initialization
3. DDRPHY of PLL / DLL lock start
4. DDRPHY of PLL / DLL lock
5. Set the M*BKUP signal to low level.
6. DDRPHY training operation start
7. DDRPHY training Exit
8. DBSC register initialization
9. SDRAM register initialization
10. Auto-refresh operation start
11. Recovery Exit

(2) DDR3L MBKPRST# control sequence

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| — | — | — | √ |

1. After stable power supply, reset release
2. DDRPHY register initialization
3. DDRPHY of PLL / DLL lock start
4. DDRPHY of PLL / DLL lock
5. Set the MBKPRST# signal to high level.
6. DDRPHY training operation start
7. DDRPHY training Exit
8. DBSC register initialization
9. SDRAM register initialization
10. Auto-refresh operation start
11. Recovery Exit

26.3 Clock Supply Path

Figure 26.4 shows clock supply path to the external memory. The clock to external memory is generated by ZB3φ. ZB3φ is multiplied by 2 in DDRPHY. Refer to 11. Clock Pulse Generator about ZB3φ.

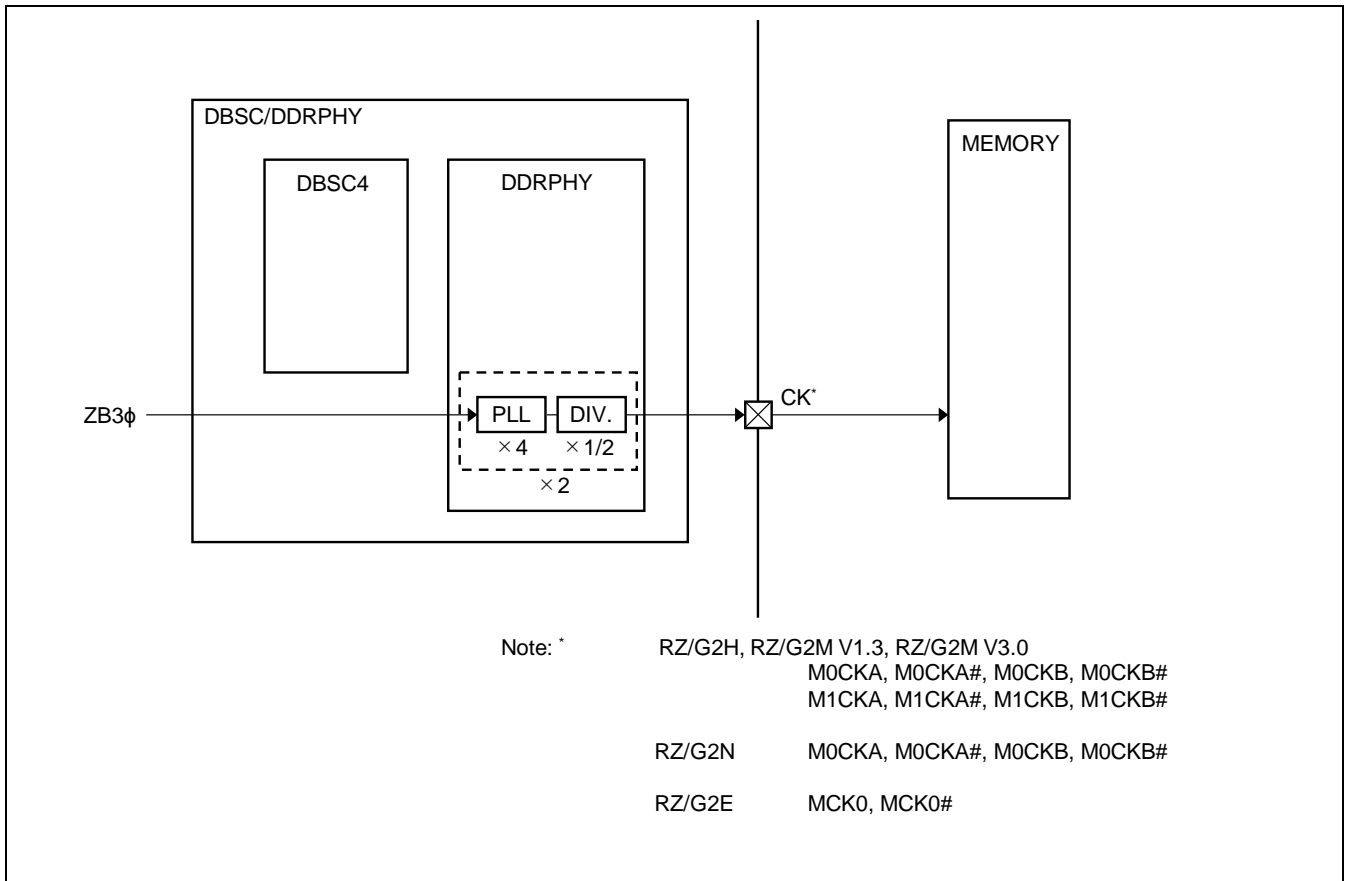


Figure 26.4 Clock supply path to the external memory

27. System RAM

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

27.1 Overview

RZ/G series 2nd generation products, RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E incorporate System RAM.

27.1.1 Features

- Memory size: 384Kbytes in RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, and RZ/G2N
- 128Kbytes in RZ/G2E
- The data of System RAM is undefined on power-on reset and retained on other resets.
- System RAM takes charge of buffer between CPUs.

27.1.2 Block Diagram

System RAM is embedded in DBSC4.

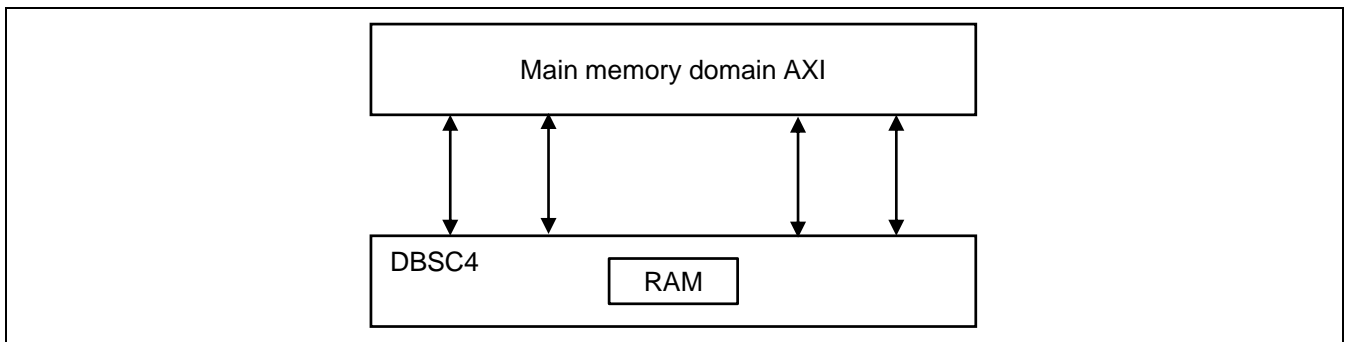


Figure 27.1 Block Diagram

27.2 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

27.2.1 Address of System RAM

Address of System RAM is shown in the table below.

Table 27.1 Address of System RAM

| Address | Notice |
|--|--|
| RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, and RZ/G2N H'E630_0000 to H'E635_FFFF | H'E633_0000 to H'E635_FFFF is shadow RAM area of H'E63A_0000 to H'E63C_FFFF. |
| RZ/G2E H'E630_0000 to H'E631_FFFF | There is no shadow RAM area. |

Note: Do not write to any addresses other than listed above. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed above are undefined.

28. 3D Graphics Engine (3DGE)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

28.1 Overview

The PowerVR Series6XT Rogue architecture builds on the Series6 architecture that consumes the lowest memory bandwidth in the industry while delivering the best performance per mm² and per mW. The Series6XT architecture features market-leading scalability, supporting implementations up to eight compute clusters that scale linearly in GFLOPS and texturing rates. With OpenGL ES support across the range, Series6XT provides among the highest performance OpenGL ES GPUs in the industry.

The PowerVR Series7XE family is a range of ultra-efficient, highly-featured GPUs. The Series7XE range enables the latest games and apps on devices which require high quality UIs at optimum price points.

The PowerVR Series8XE family of GPUs drives cost reduction in entry-level and mass market devices optimized to deliver the best user experience in a limited silicon area budget. The GPU family enables low-cost devices to benefit from the latest apps by supporting OpenGL ES.

- RZ/G2H, RZ/G2M V1.3 and RZ/G2M V3.0 integrate PowerVR GX6650 and GX6250 in a family of PowerVR Series6XT GPU cores respectively
- PowerVR GX6250 and GX6650 are fully-featured implementations with two and six shading clusters respectively.
- System Level Cache provides caching of all types of workload data. 128 KB capacity.
- RZ/G2N integrates PowerVR GE7800 in a family of PowerVR Series7XE GPU cores.
- System Level Cache provides caching of all types of workload data. 64 KB capacity.
- RZ/G2E integrates PowerVR GE8300 in a family of PowerVR Series8XE GPU cores.
- System Level Cache provides caching of all types of workload data. 64 KB capacity.
- OpenGL ES 3.1 supported

28.1.1 Features

- Tile-based deferred rendering architecture with concurrent processing of multiple tiles
- Multi-threaded Unified Shading Cluster (USC) engine incorporating pixel shader, vertex shader, and GP-GPU (compute shader) functionality
- USC incorporates an ALU architecture with high SIMD efficiency
- Fully virtualized memory addressing, supporting unified memory architecture
- Fine-grained task switching, workload balancing and power management
- Advanced DMA driven operation for minimum host CPU interaction
- Dedicated texture load DMA engine and basic ROP engine
- Programmable high quality image anti-aliasing
- Lossless texture and parameter data compression
- Dedicated processor for RogueXT/XE core firmware execution

28.1.2 Block Diagram

The PowerVR GX6650 has six shading clusters. The six shading clusters and other processing blocks are divided into five power-domains (RGX-A, RGX-B, RGX-C, RGX-D and RGX-E). Figure 28.1 shows the block diagram of the PowerVR GX6650 and the corresponding power-domains.

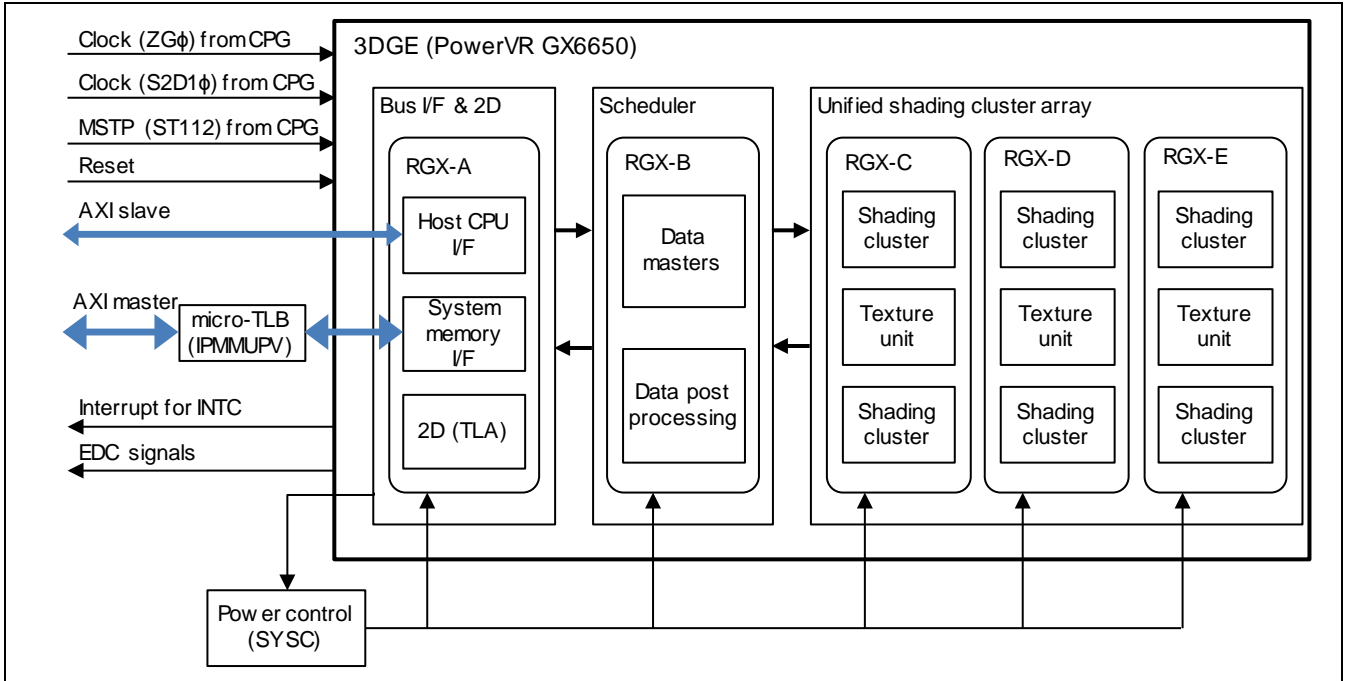


Figure 28.1 Block Diagram of PowerVR GX6650

The PowerVR GX6250 has two shading clusters. The two shading clusters and other processing blocks are divided into two power-domains (RGX-A and RGX-B). Figure 28.2 shows the block diagram of the PowerVR GX6250 and the corresponding power-domains.

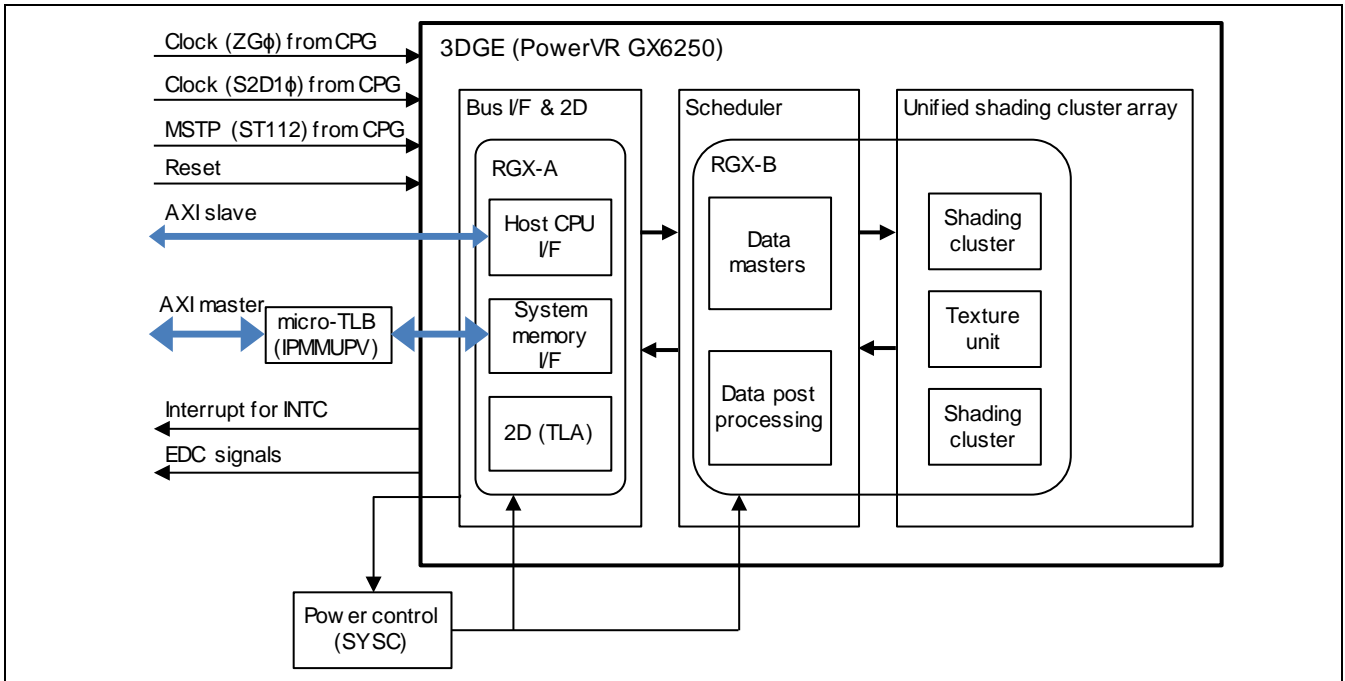


Figure 28.2 Block Diagram of PowerVR GX6250

The PowerVR GE7800 has a single shading cluster. Figure 28.3 shows the block diagram of the PowerVR GE7800 and the corresponding power-domains (RGX-A and RGX-B).

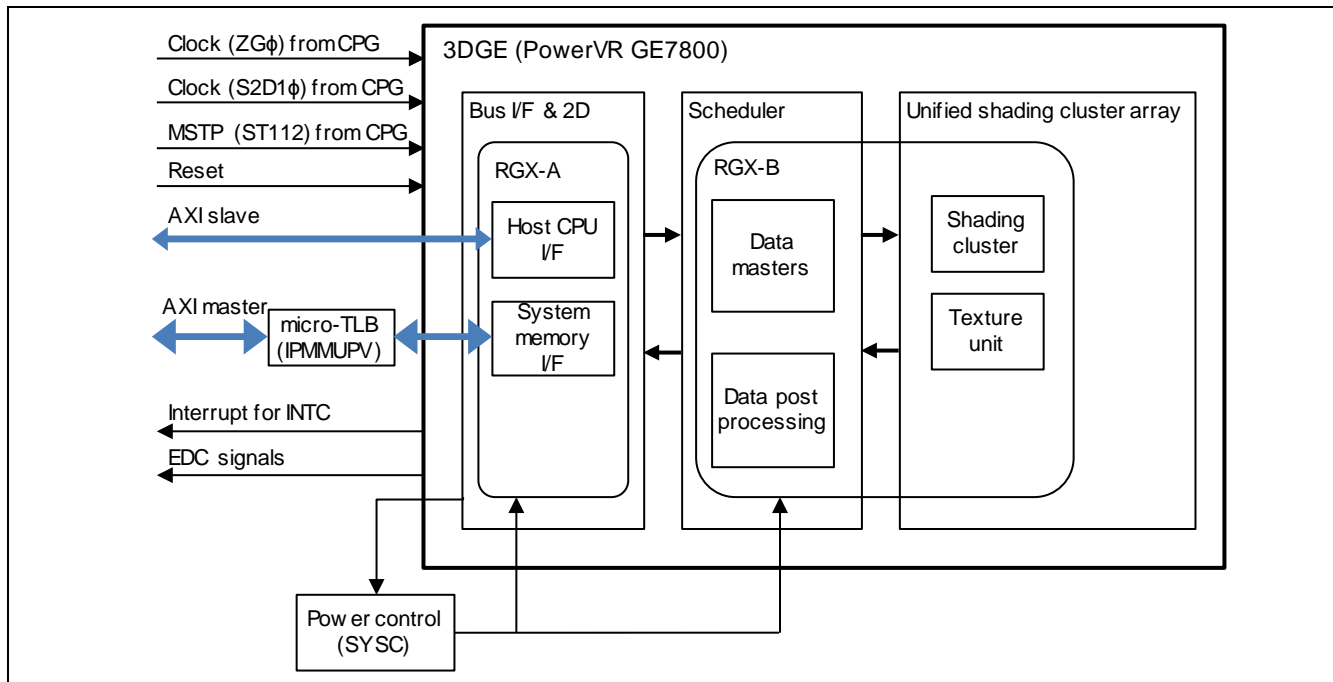


Figure 28.3 Block Diagram of PowerVR GE7800

Figure 28.4 shows the block diagram of the RZ/G2E PowerVR GE8300. Figure 28.4 shows the block diagram of the RZ/G2E PowerVR GE8300 and the corresponding power-domains (RGX-A and RGX-B).

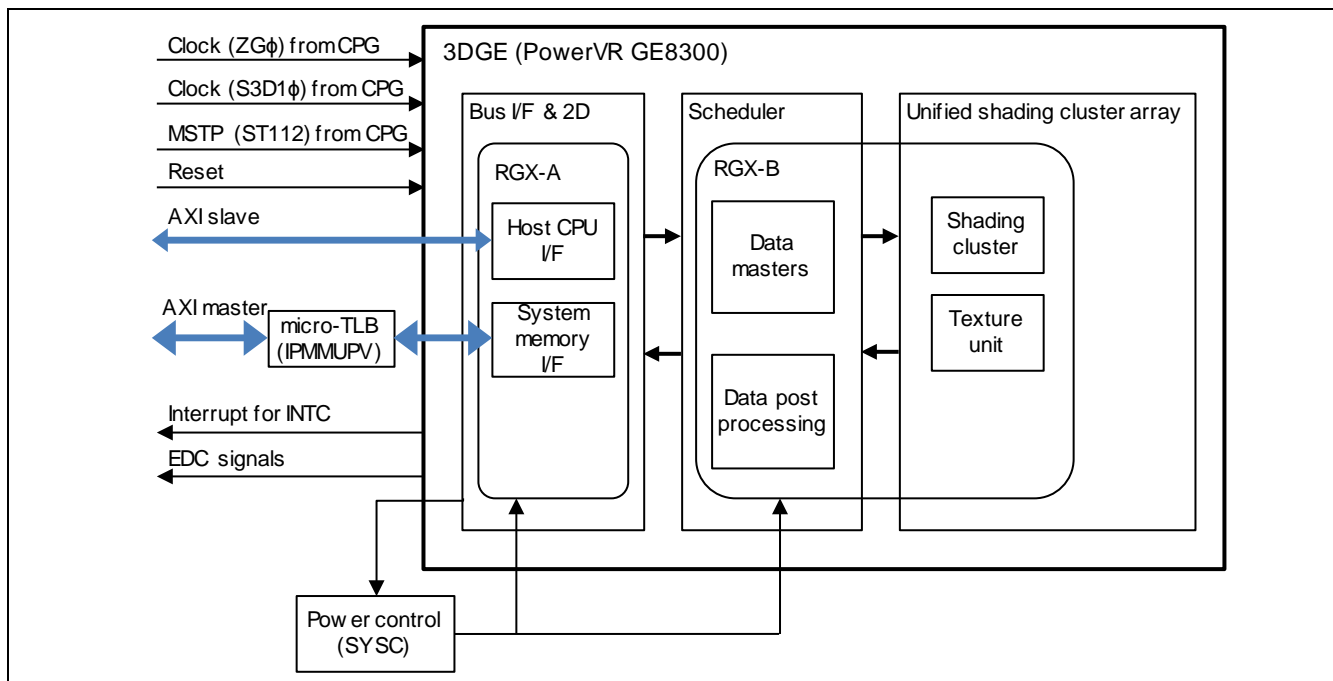


Figure 28.4 Block Diagram of RZ/G2E PowerVR GE8300

28.1.3 Register Configuration

Base address of registers is allocated to H'00_FD00_0000. The address range is 512 KB (H'00_FD00_0000 - H'00_FD03_FFFF).

28.1.4 Connected Module

| Module name | Connected module name | Function of connected module |
|-------------|-----------------------|------------------------------|
| 3DGE | SYSC | Power supply |
| | CPG | Clock supply |
| | RESET | Reset control |
| | INTC | Interrupt request control |
| | AXI-bus | AXI bus |

28.2 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

28.2.1 Initial Setting

28.2.1.1 Power and Clock Supply

The power and clock need to be supplied before PowerVR GX6650 is used. Figure 28.5 shows the example of initial setting. Each power-domain can be configured separately. In this example, all power-domains are on at the same time.

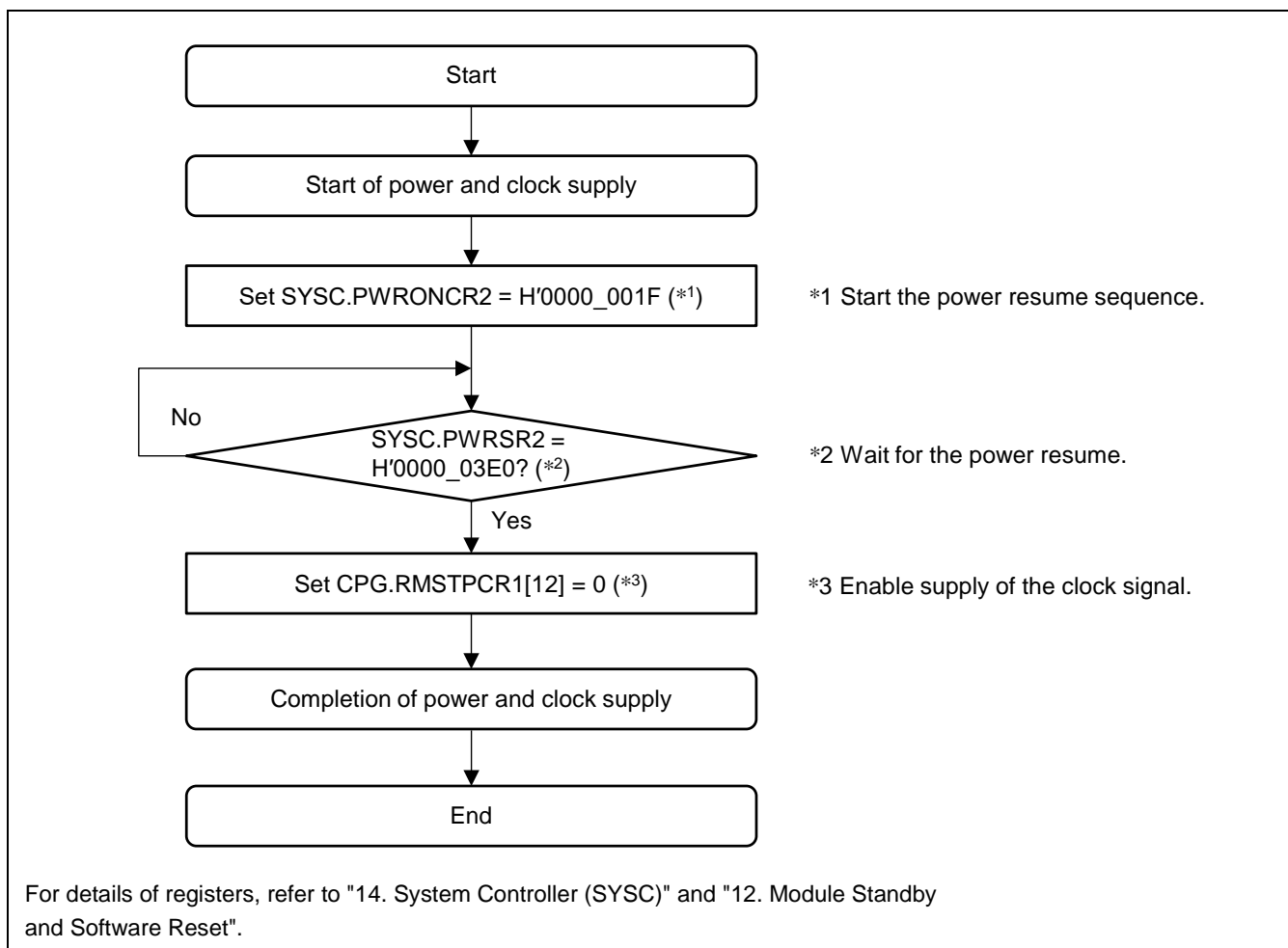


Figure 28.5 GX6650 Power and Clock Supply Flowchart

The power and clock need to be supplied before PowerVR GX6250 is used. Figure 28.6 shows the example of initial setting. Each power-domain can be configured separately. In this example, all power-domains are on at the same time.

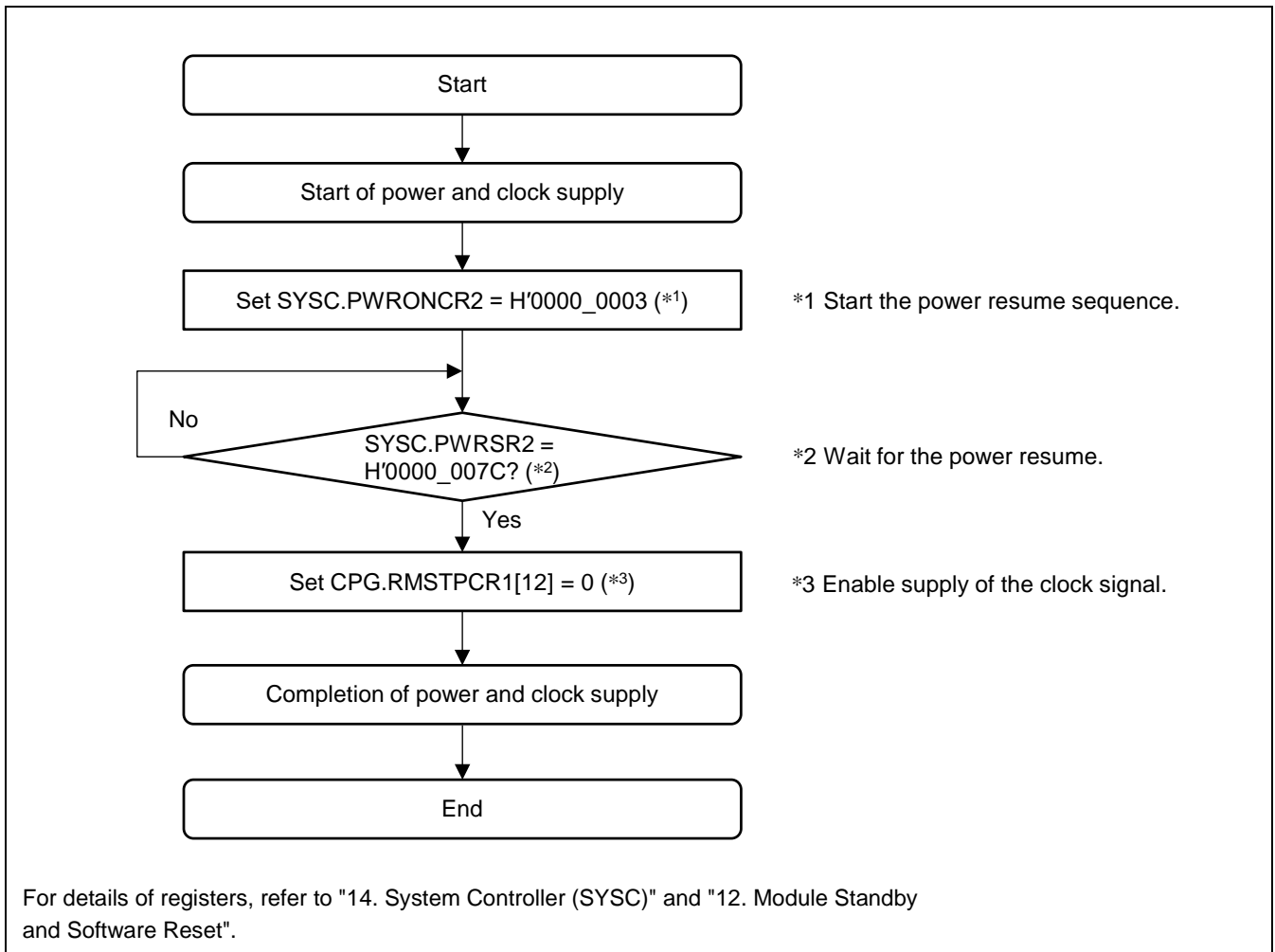


Figure 28.6 GX6250 Power and Clock Supply Flowchart

The power and clock need to be supplied before PowerVR GE7800 is used. Figure 28.7 shows the example of initial setting. Each power-domain can be configured separately. In this example, all power-domains are on at the same time.

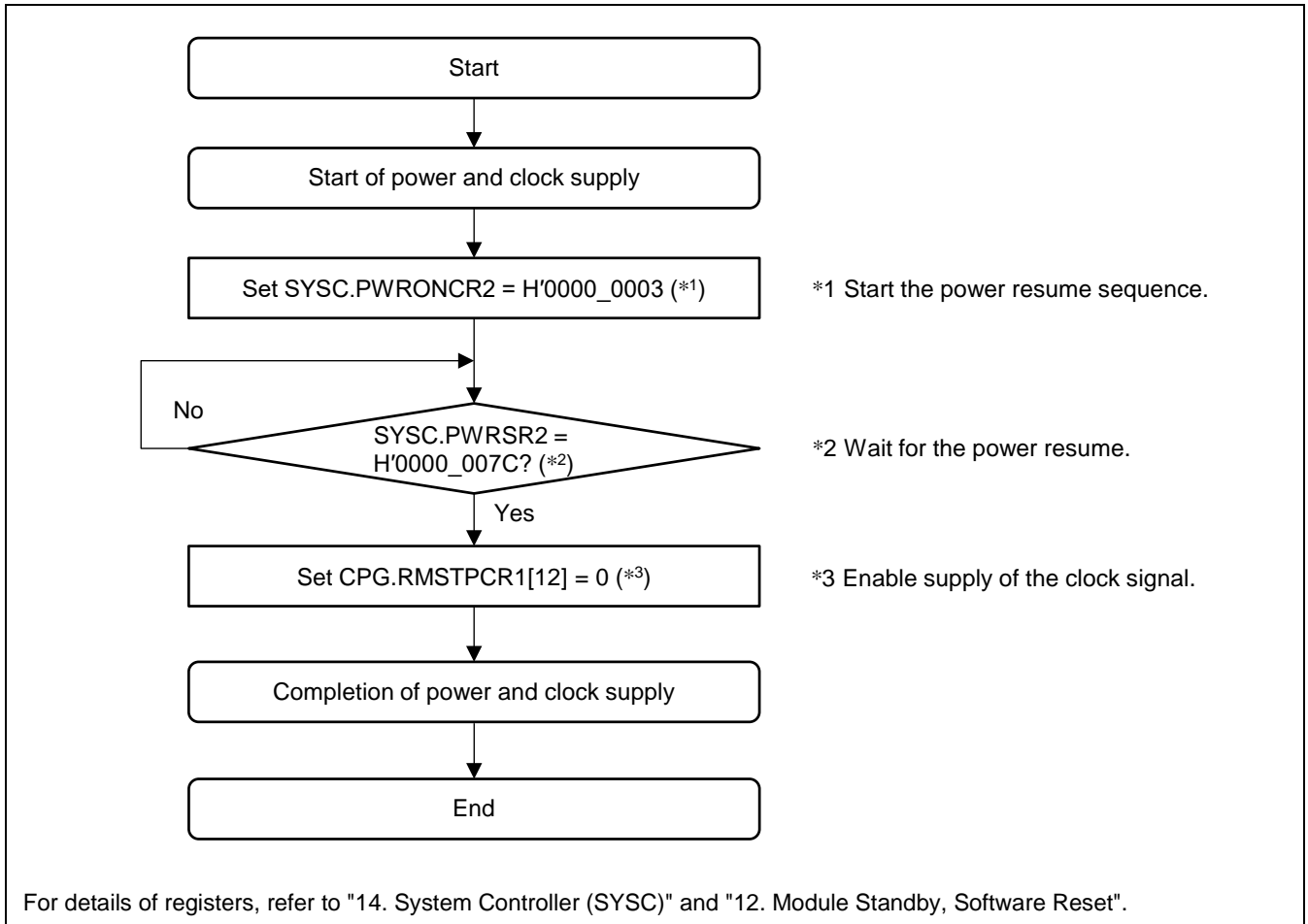


Figure 28.7 GE7800 Power and Clock Supply Flowchart

The power and clock need to be supplied before RZ/G2E PowerVR GE8300 is used. Figure 28.8 shows the example of initial setting. Each power-domain can be configured separately. In this example, all power-domains are on at the same time.

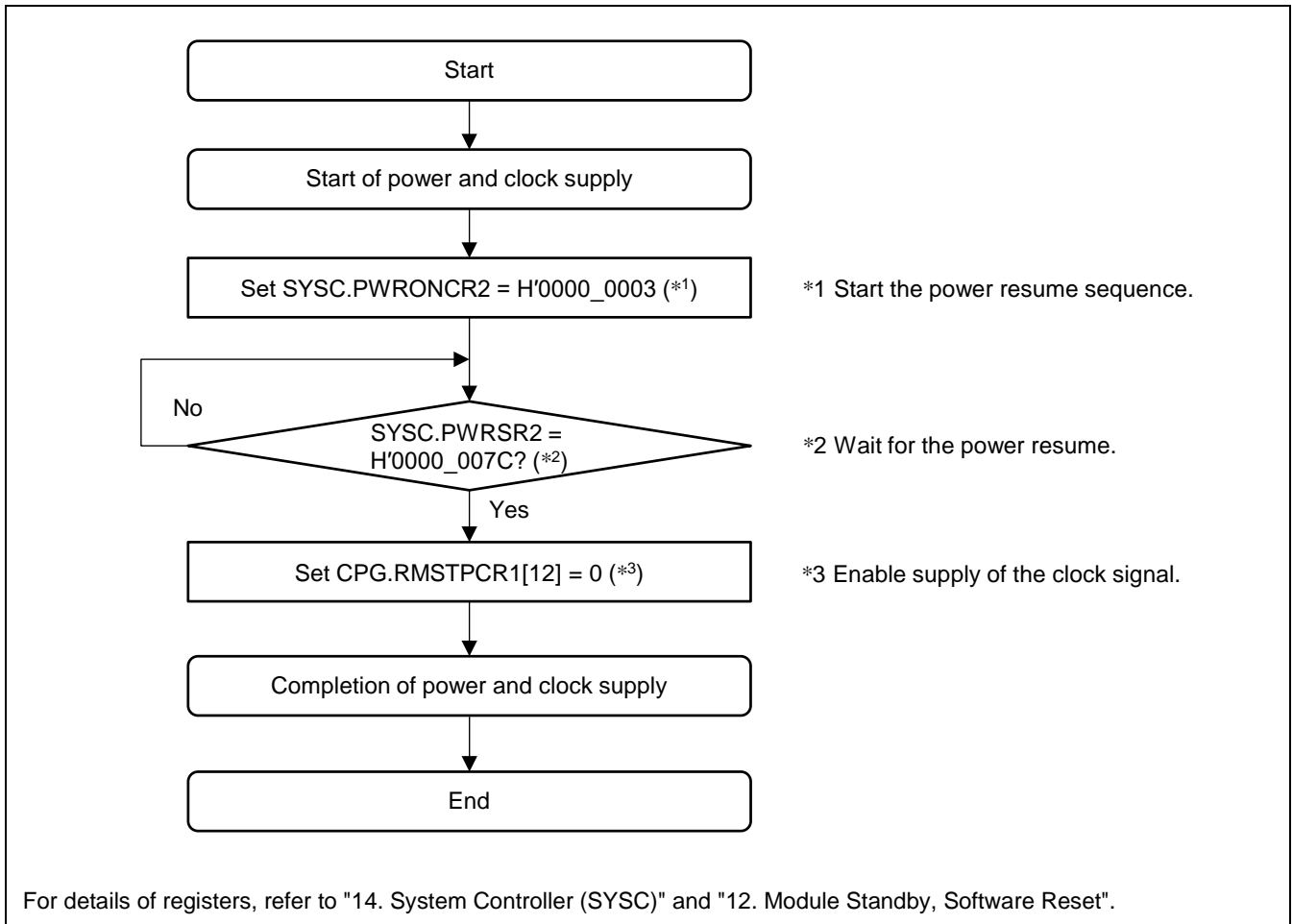


Figure 28.8 RZ/G2E GE8300 Power and Clock Supply Flowchart

29. CSI2

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

29.1 Overview

The CSI2 is an MIPI CSI-2 (Camera Serial Interface 2) receiver module.

It is supported MIPI CSI-2 V1.1 and MIPI D-PHY V1.1.

It includes the PHY and LINK modules. The PHY module incorporates a DPHY

- The RZ/G2H has two CSI2 set: one 4-lane CSI2, one 2-lane CSI2.
- The RZ/G2M V1.3 has two CSI2 sets: one 4-lane CSI2, one 2-lane CSI2.
- The RZ/G2M V3.0 has two CSI2 sets: one 4-lane CSI2, one 2-lane CSI2.
- The RZ/G2N has two CSI2 set: one 4-lane CSI2, one 2-lane CSI2.
- The RZ/G2E has one CSI2 set: one 2-lane CSI2.

The received signals are input to the PHY modules; the LINK modules receive the byte data output from the PHY modules; and the PHY modules extract the image data (embedded data included), data type, and channel information before outputting the data to the following VIN module. Here, the LINK modules generate the vertical sync signals (VD), field signals (FLD), horizontal sync signals (HD, EBD_HD), data enable signals (PE), byte enable signals (PEB) for each channel, and outputs them to the VIN module.

The LINK modules have the APB interface but the PHY modules have no bus interfaces. The LINK registers are used to control the PHY modules and monitor the state of the PHY modules.

Note: CSI2 module name of RZ/G2E is CSI40(CSI4LNK0), but the module is 2-lane CSI2.

29.1.1 Features

The CSI2 module has the following features.

- (1) From 80-Mbps to 1.5-Gbps transfer rate of MIPI CSI-2*1 (refer to section 29.3.1)
- (2) ECC 1-bit error correction and 2-bit or more error detection of a packet header (refer to section 29.3.2)
- (3) CRC error detection of a payload data part (refer to section 29.3.3)
- (4) Generation of VD (vertical sync), HD (horizontal sync), and FLD (field) signals (refer to section 29.3.4)
- (5) Four-channel output (refer to section 29.3.5)
- (6) Interrupts (refer to section 29.3.6)
- (7) Lane swapping (refer to section 29.3.7)
- (8) Initial setting of PHY through register setting (refer to section 29.3.8)
- (9) PHY control and monitoring through register setting (refer to section 29.3.9)
- (10) Interrupt generation by ULP status, and reception error status (refer to section 29.3.10)
- (11) Support data type (refer to section 30 Video Input Module (VIN))

Notes: 1. Up to 1.1-Gbps transfer rate of MIPI CSI-2 in RZ/G2E.

29.1.2 Block Diagram

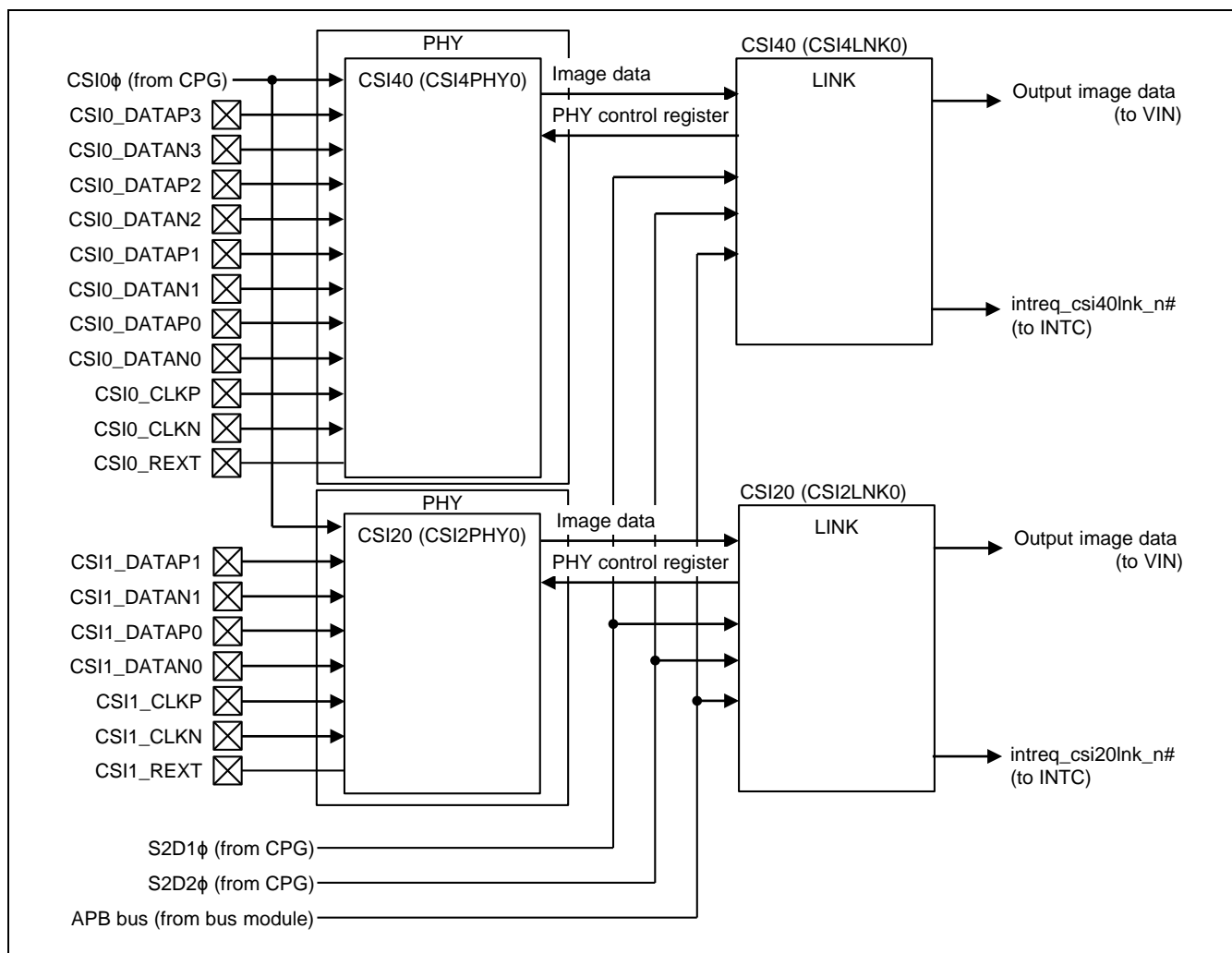


Figure 29.1 CSI2 Block Diagram(RZ/G2H)

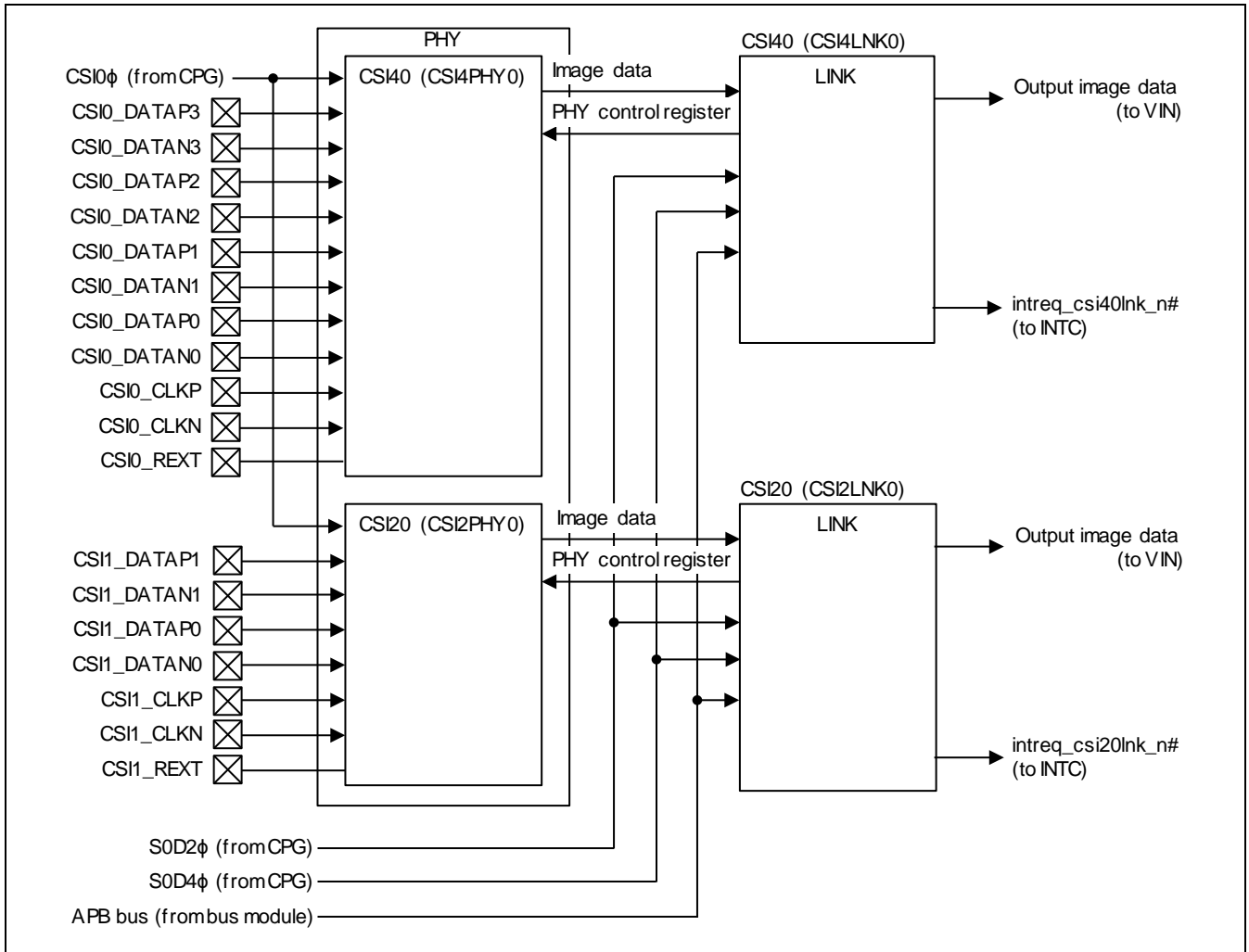


Figure 29.2 CSI2 Block Diagram (RZ/G2M V1.3, RZ/G2M V3.0)

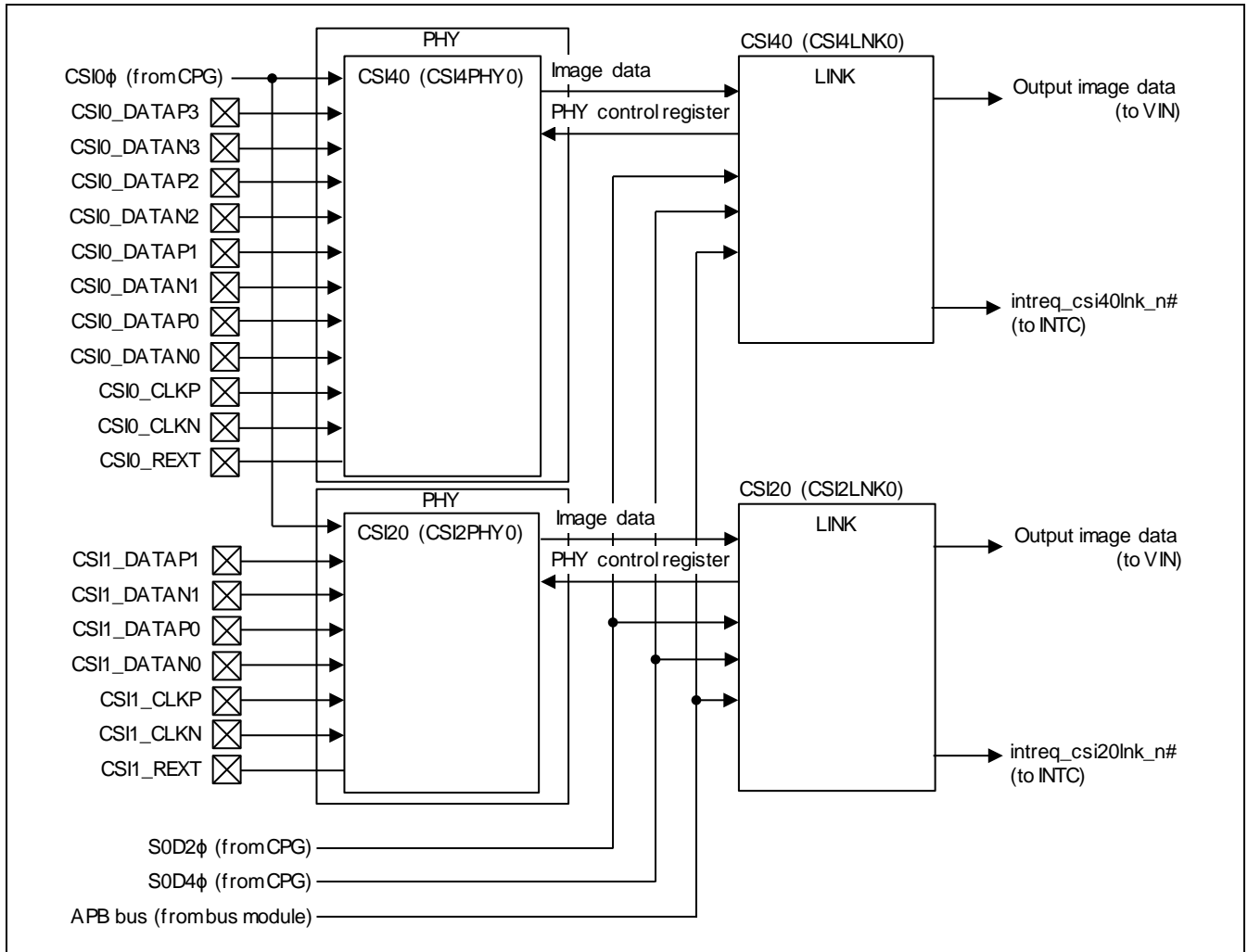


Figure 29.3 CSI2 Block Diagram(RZ/G2N)

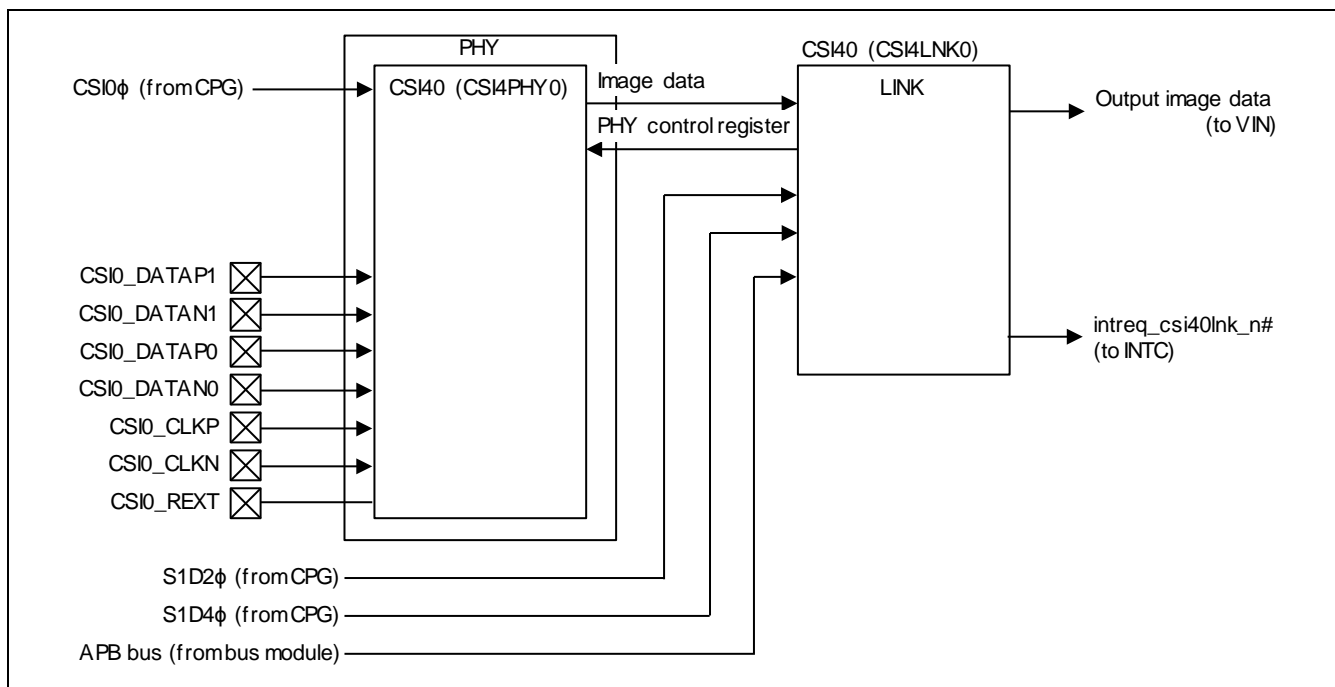


Figure 29.4 CSI2 Block Diagram(RZ/G2E)

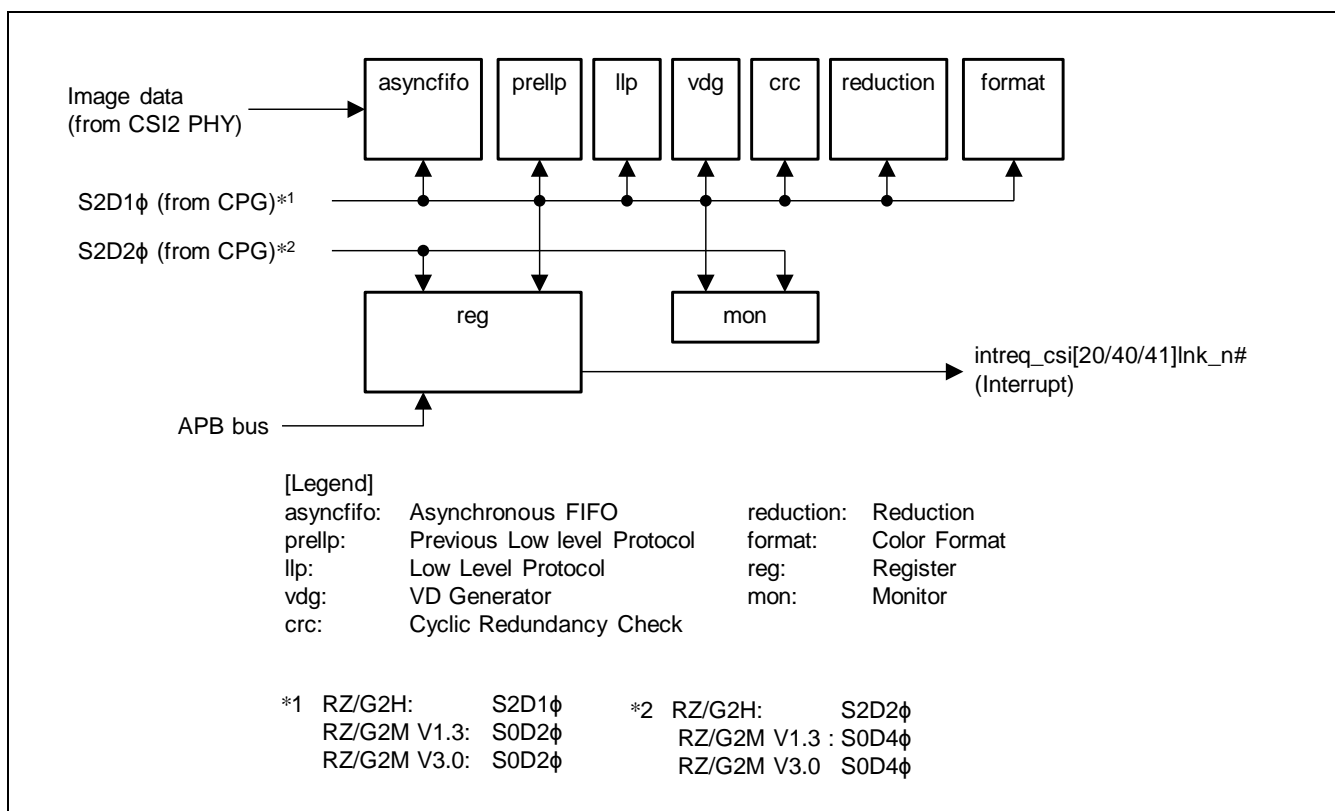


Figure 29.5 CSI2 LINK Block Diagram(RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E)

29.1.3 External Pins

Table 29.1 External Pins

| Pin Name | Abbreviation | I/O | Function | Second Generation RZ/G Series Products | | | | |
|--|--------------|-------|--|---|----------------------------|--------|--------|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| CSI40 lane 0 positive data pin | CSI0_DATAP0 | Input | Differential positive receive data input on CSI2 set 0 lane 0 | √ | √ | √ | √ | |
| CSI40 lane 0 negative data pin | CSI0_DATAN0 | Input | Differential negative receive data input on CSI2 set 0 lane 0 | √ | √ | √ | √ | |
| CSI40 lane 1 positive data pin | CSI0_DATAP1 | Input | Differential positive receive data input on CSI2 set 0 lane 1 | √ | √ | √ | √ | |
| CSI40 lane 1 negative data pin | CSI0_DATAN1 | Input | Differential negative receive data input on CSI2 set 0 lane 1 | √ | √ | √ | √ | |
| CSI40 lane 2 positive data pin | CSI0_DATAP2 | Input | Differential positive receive data input on CSI2 set 0 lane 2 | √ | √ | √ | — | |
| CSI40 lane 2 negative data pin | CSI0_DATAN2 | Input | Differential negative receive data input on CSI2 set 0 lane 2 | √ | √ | √ | — | |
| CSI40 lane 3 positive data pin | CSI0_DATAP3 | Input | Differential positive receive data input on CSI2 set 0 lane 3 | √ | √ | √ | — | |
| CSI40 lane 3 negative data pin | CSI0_DATAN3 | Input | Differential negative receive data input on CSI2 set 0 lane 3 | √ | √ | √ | — | |
| CSI40 clock lane positive data pin | CSI0_CLKP | Input | Differential positive reception input on CSI2 set 0 clock lane | √ | √ | √ | √ | |
| CSI40 clock lane negative data pin | CSI0_CLKN | Input | Differential negative reception input on CSI2 set 0 clock lane | √ | √ | √ | √ | |
| CSI20 lane 0 positive data pin | CSI1_DATAP0 | Input | Differential positive receive data input on CSI2 set 1 lane 0 | √ | √ | √ | — | |
| CSI20 lane 0 negative data pin | CSI1_DATAN0 | Input | Differential negative receive data input on CSI2 set 1 lane 0 | √ | √ | √ | — | |
| CSI20 lane 1 positive data pin | CSI1_DATAP1 | Input | Differential positive receive data input on CSI2 set 1 lane 1 | √ | √ | √ | — | |
| CSI20 lane 1 negative data pin | CSI1_DATAN1 | Input | Differential negative receive data input on CSI2 set 1 lane 1 | √ | √ | √ | — | |
| CSI20 clock lane positive data pin | CSI1_CLKP | Input | Differential positive reception input on CSI2 set 1 clock lane | √ | √ | √ | — | |
| CSI20 clock lane negative data pin | CSI1_CLKN | Input | Differential negative reception input on CSI2 set 1 clock lane | √ | √ | √ | — | |
| CSI40 external resistor connection pin | CSI0_REXT | I/O | Internal reference current generation for CSI2 set 0 | √ | √ | √ | √ | |
| CSI20 external resistor connection pin | CSI1_REXT | I/O | Internal reference current generation for CSI2 set 1 | √ | √ | √ | — | |

29.1.4 Register Configuration

The following lists the base address of each CSI2 module.

Second Generation RZ/G Series Products

| Module Name | Address | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|------------------|-------------|--------|----------------------------|--------|--------|
| CSI20 (CSI2LNK0) | H'FEA8_0000 | √ | √ | √ | — |
| CSI40 (CSI4LNK0) | H'FEAA_0000 | √ | √ | √ | √ |

Table 29.2 Register Configuration

| Register Name | Abbreviation | R/W | Offset Address | Initial Value | Access Size*3 | Second Generation RZ/G Series Products | | | | |
|---|--------------|-------------|----------------|---------------|---------------|--|----------------------------|--------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| Control Timing Select Register | TREF | R/W | H'0000 | H'0000_0001 | 32 | √ | √ | √ | √ | |
| Software Reset Register | SRST | R/W | H'0004 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PHY Operation Control Register | PHYCNT | R/W | H'0008 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Checksum Control Register | CHKSUM | R/W | H'000C | H'0000_0003 | 32 | √ | √ | √ | √ | |
| Channel Data Type Select Register | VCDT | R/W | H'0010 | *4 | 32 | √ | √ | √ | √ | |
| Channel Data Type Select Register 2 | VCDT2 | R/W | H'0014 | *4 | 32 | √ | √ | √ | — | |
| Frame Data Type Select Register | FRDT | R/W | H'0018 | H'0001_0000 | 32 | √ | √ | √ | √ | |
| Field Detection Control Register | FLD | R/W | H'001C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Automatic Standby Control Register | ASTBY | R/W | H'0020 | H'0000_3F21 | 32 | √ | √ | √ | √ | |
| Long Data Type Setting Register 0 | LNGDT0 | R/W | H'0028 | H'FFFF_0000 | 32 | √ | √ | √ | √ | |
| Long Data Type Setting Register 1 | LNGDT1 | R/W | H'002C | H'FFFF_FFFF | 32 | √ | √ | √ | √ | |
| Interrupt Enable Register | INTEN | R/W | H'0030 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Interrupt Source Mask Register | INTCLOSE | R/W | H'0034 | H'0004_0000 | 32 | √ | √ | √ | √ | |
| Interrupt Status Monitor Register | INTSTATE | R/(W) *1 | H'0038 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Interrupt Error Status Monitor Register | INTERRSTATE | R/(W) *2 | H'003C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Short Packet Data Register | SHPDAT | R | H'0040 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Short Packet Count Register | SHPCNT | R | H'0044 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| LINK Operation Control Register | LINKCNT | R/W | H'0048 | H'8000_0000 | 32 | √ | √ | √ | √ | |
| Lane Swap Register | LSWAP | R/W | H'004C | H'0000_00E4 | 32 | √ | √ | √ | √ | |
| PHY Test Interface Write Register | PHTW | R/W | H'0050 | H'0000_0000 | 32 | √ | — | √ | √ | |
| PHY Test Interface Read Register | PHTR | R | H'0054 | H'0000_0000 | 32 | √ | — | √ | √ | |
| PHY Test Interface Clear Register | PHTC | R/W | H'0058 | H'0000_0001 | 32 | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Register Name | Abbreviation | R/W | Offset Address | Initial Value | Access Size*3 | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|------------------------------------|--------------|-----|----------------|---------------|---------------|--------|----------------------------|--------|--------|
| PHY Frequency Control Register | PHYPLL | R/W | H'0068 | H'0000_0000 | 32 | √ | √ | √ | — |
| PHY ESC Error Monitor Register | PHEERM | R | H'0074 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PHY Clock Lane Monitor Register | PHCLM | R | H'0078 | H'0000_000C | 32 | √ | √ | √ | √ |
| PHY Data Lane Monitor Register | PHDLM | R | H'007C | H'0000_F000 | 32 | √ | √ | √ | √ |
| Packet Header 0 Monitor Register 0 | PH0M0 | R | H'00F0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header 0 Monitor Register 1 | PH0M1 | R | H'00F4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header 1 Monitor Register 0 | PH1M0 | R | H'00F8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header 1 Monitor Register 1 | PH1M1 | R | H'00FC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header 2 Monitor Register 0 | PH2M0 | R | H'0100 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header 2 Monitor Register 1 | PH2M1 | R | H'0104 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header 3 Monitor Register 0 | PH3M0 | R | H'0108 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header 3 Monitor Register 1 | PH3M1 | R | H'010C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header R Monitor Register 0 | PHRM0 | R | H'0110 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header R Monitor Register 1 | PHRM1 | R | H'0114 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header R Monitor Register 2 | PHRM2 | R | H'0118 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header C Monitor Register 0 | PHCM0 | R | H'0120 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header C Monitor Register 1 | PHCM1 | R | H'0124 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CRC Monitor Register 0 | CRCM0 | R | H'0128 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CRC Monitor Register 1 | CRCM1 | R | H'012C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SOT Error Count Register | SERRCNT | R | H'0140 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SOTSYNC Error Count Register | SSERRCNT | R | H'0144 | H'0000_0000 | 32 | √ | √ | √ | √ |
| ECC_CRCT Count Register | ECCCM | R | H'0148 | H'0000_0000 | 32 | √ | √ | √ | √ |
| ECC_ERR Count Register | ECECM | R | H'014C | H'0000_0000 | 32 | √ | √ | √ | √ |
| CRC_ERR Count Register | CRCECM | R | H'0150 | H'0000_0000 | 32 | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Register Name | Abbreviation | R/W | Offset Address | Initial Value | Access Size*3 | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|----------------------------------|--------------|-----|----------------|---------------|---------------|--------|----------------------------|--------|--------|
| Line Register | LCNT | R | H'0160 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Line Register 2 | LCNT2 | R | H'0164 | H'0000_0000 | 32 | √ | √ | √ | — |
| Line Monitor Register | LCNTM | R | H'0168 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Line Monitor Register 2 | LCNTM2 | R | H'016C | H'0000_0000 | 32 | √ | √ | √ | — |
| Frame Count Monitor Register | FCNTM | R | H'0170 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Frame Count Monitor Register 2 | FCNTM2 | R | H'0174 | H'0000_0000 | 32 | √ | √ | √ | — |
| PHY Data IN Monitor Register | PHYDIM | R | H'0180 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PHY Input Monitor Register | PHYIM | R | H'0184 | H'0000_0000 | 32 | √ | √ | √ | √ |
| VIN Data Monitor Register | VINDM | R | H'018C | H'0000_0000 | 32 | √ | √ | √ | √ |
| VIN Signal Monitor Register 1 | VINSM1 | R | H'0190 | H'0000_0000 | 32 | √ | √ | √ | √ |
| VIN Signal Monitor Register 2 | VINSM2 | R | H'0194 | H'0000_0000 | 32 | √ | √ | √ | — |
| VIN Signal Monitor Register 3 | VINSM3 | R | H'0198 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PHY Output Monitor Register | PHYOM | R | H'019C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header Monitor Register 1 | PHM1 | R | H'01C0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header Monitor Register 2 | PHM2 | R | H'01C4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header Monitor Register 3 | PHM3 | R | H'01C8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header Monitor Register 4 | PHM4 | R | H'01CC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header Monitor Register 5 | PHM5 | R | H'01D0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header Monitor Register 6 | PHM6 | R | H'01D4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header Monitor Register 7 | PHM7 | R | H'01D8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Header Monitor Register 8 | PHM8 | R | H'01DC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Period Monitor Register 1 | PPM1 | R | H'01E0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Period Monitor Register 2 | PPM2 | R | H'01E4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Period Monitor Register 3 | PPM3 | R | H'01E8 | H'0000_0000 | 32 | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Register Name | Abbreviation | R/W | Offset Address | Initial Value | Access Size* ³ | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---|--------------|-----|----------------|---------------|---------------------------|--------|----------------------------|--------|--------|
| Packet Period Monitor Register 4 | PPM4 | R | H'01EC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Period Monitor Register 5 | PPM5 | R | H'01F0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Period Monitor Register 6 | PPM6 | R | H'01F4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Period Monitor Register 7 | PPM7 | R | H'01F8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Packet Period Monitor Register 8 | PPM8 | R | H'01FC | H'0000_0000 | 32 | √ | √ | √ | √ |
| CSI0CLK Frequency Configuration Preset Register | CSI0CLKFCPR | R/W | H'0260 | H'0000_0000 | 32 | √ | — | √ | — |

- Notes: 1. Only 1 can be written to clear the flag; bits 27, 4, and 3 are read-only bits and any value cannot be written to.
 2. Only 1 can be written to clear the flag.
 3. No supported except for 32 bits Access Size.
 4. Initial value will differ for each product. For detail, refer to Register Description.

29.1.5 Connected Module

Table 29.3 Connected Module

| Connected Modules | Function of the Connected Modules |
|-------------------|--|
| VIN | Output of image data |
| CPG | Output and control of the clock signals, execution of software reset |
| INTC*1 | Interrupt controller |

Notes: 1. Interrupt channel correspondence is as follow:

For details about interrupt channel, refer to chapter 12A Interrupt Controller (INTC-AP).

| CSI2 ch | Interrupt ch |
|------------------|--------------|
| CSI20 (CSI2LNK0) | CSI2.ch0 |
| CSI40 (CSI4LNK0) | CSI2.ch2 |

29.2 Register Description

[Legend]

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

R: Read-only. The write value should always be 0.

—/WB: Write-only. The read value is undefined.

29.2.1 Control Timing Select Register (TREF)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | TREF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | TREF | B'1 | R/W | Register value reflection timing select 1: Immediately (Don't change from initial value) |

29.2.2 Software Reset Register (SRST)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SRST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | SRST | B'0 | R/W | Software reset control 0: Normal 1: Reset This bit resets the control circuit in the CSI2 block. To reset the circuit, set this bit to 1. Be sure to set this bit to 0 to cancel a reset. A software reset does not initialize the CSI2 registers. |

Note: The settings of this register are applied the instant the register is set (immediate reflection).

29.2.3 PHY Operation Control Register (PHYCNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|---------------|--------------|--------------|---------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SHUTD OWNZ | RSTZ |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | ENABL ECLK | ENABL E_3 | ENABL E_2 | ENABL E_1 | ENABL E_0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 18 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 17 | SHUTDOWNZ | B'0 | R/W | Shuts down the PHY when this bit is 0. |
| 16 | RSTZ | B'0 | R/W | Initializes the PHY when this bit is 0. |
| 15 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | ENABLECLK | B'0 | R/W | Clock lane operation enable 0: Disables operation. 1: Enables operation. When any of the ENABLE_3 to ENABLE_0 bits are set to 1, be sure to set the ENABLECLK bit to 1. |
| 3 | ENABLE_3 | B'0 | R/W | Data lane 3 operation enable [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: Disables operation. 1: Enables operation. Note: When 2-lane PHY is connected, this setting has no effect. |
| 2 | ENABLE_2 | B'0 | R/W | Data lane 2 operation enable [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: Disables operation. 1: Enables operation. Note: When 2-lane PHY is connected, this setting has no effect. |
| 1 | ENABLE_1 | B'0 | R/W | Data lane 1 operation enable 0: Disables operation. 1: Enables operation. |
| 0 | ENABLE_0 | B'0 | R/W | Data lane 0 operation enable [RZ/G2M V1.3, RZ/G2M V3.0] 0: Disables operation. 1: Enables operation. Data lane 0 operation enable [RZ/G2H, RZ/G2N] 0: Disables operation. 1: Enables operation. This bit must set to 1. |

Note: The settings of this register are applied the instant the register is set (immediate reflection). Set this register while data input is suspended (a sensor is halted); never change the settings of this register during operation.

Note: ENABLE_* signal don't must be set 3 lanes. For example, ENABLE_0, 1, 2 is set to 1'b1 and ENABLE_3 is set to 1'b0.

29.2.4 Checksum Control Register (CHKSUM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ECC_EN | CRC_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | ECC_EN | B'1 | R/W | ECC process control 0: Disabled 1: Enabled This bit controls ECC process of the packet header (PH) for Low Level Protocol (LLP). Set this bit to 1 to use ECC. When set to 0, data is always processed assuming it has no errors and thus no errors are corrected. |
| 0 | CRC_EN | B'1 | R/W | CRC checksum process control 0: Disabled 1: Enabled This bit controls checksum process of the long packet data for Low Level Protocol (LLP). Set this bit to 1 to use CRC checksum. When set to 0, data is always processed assuming it has no errors. |

Note: The settings of this register are applied the instant the register is set (immediate reflection). Set this register while data input is suspended (a sensor is halted); never change the settings of this register during operation.

29.2.5 Channel Data Type Select Register (VCDT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----------|----|----|----|----|----|---------|-----|------------|---------|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | VCDT2_EN | — | — | — | — | — | SEL_VC2 | — | SEL_DT2_ON | SEL_DT2 | | | | | | |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| R/W: | R/W | R | R | R | R | R | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| | | | | | | | | | | | | | | | | |
|----------------|---------|----|----|----|----|----|--------|-----|-----------|--------|-----|-----|-----|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | VCDT_EN | — | — | — | — | — | SEL_VC | — | SEL_DT_ON | SEL_DT | | | | | | |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| R/W: | R/W | R | R | R | R | R | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 | VCDT2_EN | B'1 | R/W | Channel 1 enable 0: Disabled 1: Enabled This bit enables or disables channel 1. When set to 0, VD[1] and FLD[1] are fixed to low and TAG = 1 is not generated. |
| 30 to 26 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25, 24 | SEL_VC2 | B'01 | R/W | Channel 1 select channel [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] H'0 to H'3 These bits specify the virtual channels for channel 1 to capture. Channels can be selected from among four channels from H0 to H3. Channel 1 is output when the value of this register matches the received virtual channel. Channel 1 select channel [RZ/G2E] H'0 to H'1 These bits specify the virtual channels for channel 1 to capture. Channels can be selected from among four channels from H0 to H1. Channel 1 is output when the value of this register matches the received virtual channel. |
| 23 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 22 | SEL_DT2_ON | B'0 | R/W | Channel 1 select data type enable 0: Disabled 1: Enabled This bit enables or disables data type match function using SEL_DT2. When set to 0, channel 1 is output even when the setting of SEL_DT2 does not match the received data type as long as it matches the virtual channel. When set to 1, channel 1 is output only when the setting of SEL_DT2 matches the received data type and the virtual channel. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 21 to 16 | SEL_DT2 | H'1E | R/W | Channel 1 select data type H'00 to H'3F These bits specify the data type for channel 1 to capture. The CSI2 LINK module outputs the received payload data as it is regardless of the data type when SEL_DT2_ON is set to 0 whereas there are some restrictions on the corresponding data type (such as color format) for the following VIN module. |
| 15 | VCDT_EN | B'1 | R/W | Channel 0 enable 0: Disabled 1: Enabled This bit enables or disables channel 0. When set to 0, VD[0] and FLD[0] are fixed to low and TAG = 0 is not generated. |
| 14 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9, 8 | SEL_VC | B'00 | R/W | Channel 0 select channel [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] H'0 to H'3 These bits specify the virtual channels for channel 0 to capture. Channels can be selected from among four channels from H'0 to H'3. Channel 0 is output when the value of this register matches the received virtual channel. Channel 0 select channel [RZ/G2E] H'0 to H'1 These bits specify the virtual channels for channel 0 to capture. Channels can be selected from among four channels from H'0 to H'1. Channel 0 is output when the value of this register matches the received virtual channel. |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 | SEL_DT_ON | B'0 | R/W | Channel 0 select data type enable 0: Disabled 1: Enabled This bit enables or disables data type match function using SEL_DT. When set to 0, channel 0 is output even when the setting of SEL_DT does not match the received data type as long as it matches the virtual channel. When set to 1, channel 0 is output only when the setting of SEL_DT matches the received data type and the virtual channel. |
| 5 to 0 | SEL_DT | H'1E | R/W | Channel 0 select data type [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] H'00 to H'3F These bits specify the data type for channel 0 to capture. The CSI2 LINK module outputs the received payload data as it is regardless of the data type when SEL_DT_ON is set to 0 whereas there are some restrictions on the corresponding data type (such as color format) for the following VIN module. |

Notes: 1. This register should be set during initial sequence.

2. In the table above, "virtual channel" refers to a channel through which data flows from the camera and "channel" refers to an output channel of the CSI2.

29.2.6 Channel Data Type Select Register 2 (VCDT2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----------|----|----|----|----|----|---------|-----|------------|---------|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | VCDT4_EN | — | — | — | — | — | SEL_VC4 | — | SEL_DT4_ON | SEL_DT4 | | | | | | |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| R/W: | R/W | R | R | R | R | R | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| | | | | | | | | | | | | | | | | |
|----------------|----------|----|----|----|----|----|---------|-----|------------|---------|-----|-----|-----|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | VCDT3_EN | — | — | — | — | — | SEL_VC3 | — | SEL_DT3_ON | SEL_DT3 | | | | | | |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| R/W: | R/W | R | R | R | R | R | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 | VCDT4_EN | B'1 | R/W | Channel 3 enable 0: Disabled 1: Enabled This bit enables or disables channel 3. When set to 0, VD[3] and FLD[3] are fixed to low and TAG = 3 is not generated. |
| 30 to 26 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25, 24 | SEL_VC4 | H'3 | R/W | Channel 3 select channel H'0 to H'3 These bits specify the virtual channels for channel 3 to capture. Channels can be selected from among four channels from H'0 to H'3. Channel 3 is output when the value of this register matches the received virtual channel. |
| 23 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 22 | SEL_DT4_ON | B'0 | R/W | Channel 3 select data type enable 0: Disabled 1: Enabled This bit enables or disables data type match function using SEL_DT4. When set to 0, channel 3 is output even when the setting of SEL_DT4 does not match the received data type as long as it matches the virtual channel. When set to 1, channel 3 is output only when the setting of SEL_DT4 matches the received data type and the virtual channel. |
| 21 to 16 | SEL_DT4 | H'1E | R/W | Channel 3 select data type H'00 to H'3F These bits specify the data type for channel 3 to capture. The CSI2 LINK module outputs the received payload data as it is regardless of the data type when SEL_DT4_ON is set to 0 whereas there are some restrictions on the corresponding data type (such as color format) for the following VIN module. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 15 | VCDT3_EN | B'1 | R/W | Channel 2 enable 0: Disabled 1: Enabled This bit enables or disables channel 2. When set to 0, VD[2] and FLD[2] are fixed to low and TAG = 2 is not generated. |
| 14 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9, 8 | SEL_VC3 | H'2 | R/W | Channel 2 select channel H'0 to H'3 These bits specify the virtual channels for channel 2 to capture. Channels can be selected from among four channels from H'0 to H'3. Channel 2 is output when the value of this register matches the received virtual channel. |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 | SEL_DT3_ON | B'0 | R/W | Channel 2 select data type enable 0: Disabled 1: Enabled This bit enables or disables data type match function using SEL_DT3. When set to 0, channel 2 is output even when the setting of SEL_DT3 does not match the received data type as long as it matches the virtual channel. When set to 1, channel 2 is output only when the setting of SEL_DT3 matches the received data type and the virtual channel. |
| 5 to 0 | SEL_DT3 | H'1E | R/W | Channel 2 select data type H'00 to H'3F These bits specify the data type for channel 2 to capture. The CSI2 LINK module outputs the received payload data as it is regardless of the data type when SEL_DT3_ON is set to 0 whereas there are some restrictions on the corresponding data type (such as color format) for the following VIN module. |

- Notes:
1. This register should be set during initial sequence.
 2. In the table above, “virtual channel” refers to a channel through which data flows from the camera and “channel” refers to an output channel of the CSI2.

29.2.7 Frame Data Type Select Register (FRDT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|-------|-----|-----|-----|-----|-----|----|----|-------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | DT_FS | | | | | | — | — | DT_FE | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31, 30 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 24 | DT_FS | H'00 | R/W | Frame start data type These bits specify the frame start data type. When the value of these bits matches the received data type, the data is determined as a frame start. |
| 23, 22 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 to 16 | DT_FE | H'01 | R/W | Frame end data type These bits specify the frame end data type. When the value of these bits matches the received data type, the data is determined as a frame end. |
| 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

29.2.8 Field Detection Control Register (FLD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-------------|---------|---------|---------|---------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | FLD_NUM | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | FLD_DET_SEL | FLD_EN4 | FLD_EN3 | FLD_EN2 | FLD_EN1 | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 16 | FLD_NUM | All 0 | R/W | Even field number setting These bits specify the value for detecting the even field of the interlaced image. For detecting the even field, the WC value (16 bits) of the Frame Start packet is referred to. When the WC value matches the setting of these bits, the field is detected as the even field. During the frame period, the CSIR_FLD signal stays High. |
| 15 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5, 4 | FLD_DET_SEL | B'00 | R/W | Even field detection condition select b'00: The field is detected as the even field when FLD_NUM matches WC. b'01: The field is detected as the even field when FLD_NUM[0] matches WC[0]. b'10: The field is detected as the even field when FLD_NUM[0] matches WC[8]. b'11: Setting is prohibited. |
| 3 | FLD_EN4 | B'0 | R/W | Channel 3 even field detection control This bit turns on or off even field detection of channel 3. To use the detection feature, set 1 to this bit. It should be set to 1'b0 when input image is progressive sequence. |
| 2 | FLD_EN3 | B'0 | R/W | Channel 2 even field detection control This bit turns on or off even field detection of channel 2. To use the detection feature, set 1 to this bit. It should be set to 1'b0 when input image is progressive sequence. |
| 1 | FLD_EN2 | B'0 | R/W | Channel 1 even field detection control This bit turns on or off even field detection of channel 1. To use the detection feature, set 1 to this bit. It should be set to 1'b0 when input image is progressive sequence. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | FLD_EN | B'0 | R/W | Channel 0 even field detection control This bit turns on or off even field detection of channel 0. To use the detection feature, set 1 to this bit. It should be set to 1'b0 when input image is progressive sequence. |

Note: This register should be set during initial sequence.

29.2.9 Automatic Standby Control Register (ASTBY)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

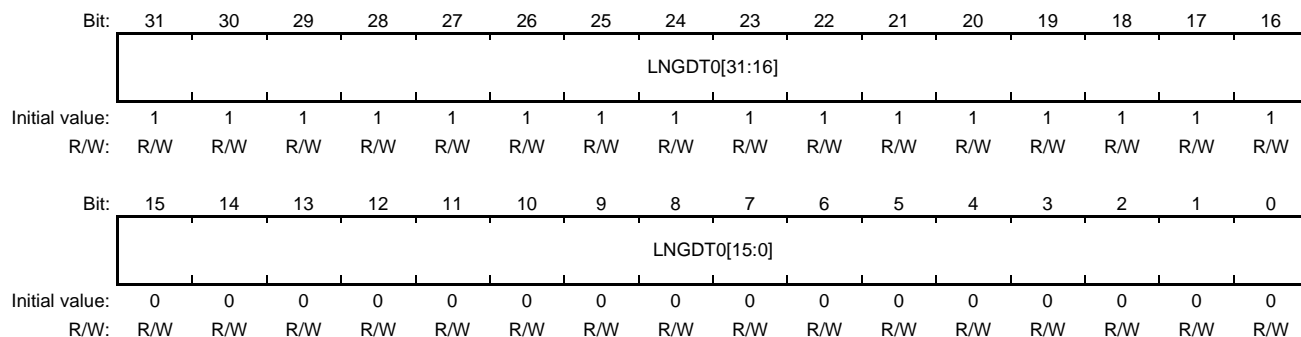
| | | | | | | | | | | | | | | | | |
|----------------|----|----|--------------|-----|-----|-----|-----|-----|----|-------------|-----------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | VD_MSK_CYCLE | | | | | — | — | VD_MSK_K_EN | AUTO_STANDBY_EN | | | | | |
| Initial value: | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|---|
| 31 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 to 8 | VD_MSK_CYCLE | H'3F | R/W | VD masking cycle These bits specify the period to mask VD when a frame start error (FS is received and another FS is received before FE is received) occurs. When a frame start error occurs, the VD signal does not go low (Vsync is not generated); VD masking function generates the period for which VD = low. VD masking period = 4 * VD_MSK_CYCLE * input clock. (about input clock, refer to 29.3.14) [Max value] VD masking period should be smaller than period from Short Packet (Frame start) of EOT to Long Packet of SOT. [Min Value] 5 or more (VD masking period is 20 * input clock). |
| 7, 6 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------------|---------------|-----|--|
| 5 | VD_MSK_EN | B'1 | R/W | <p>VD masking enable</p> <p>0: Disables masking VD. 1: Enables masking VD.</p> <p>This bit enables or disables VD masking function when a frame start error occurs.</p> <p>When this bit is set to enable the function, VD are masked for the period specified with the VD_MSK_CYCLE bits.</p> |
| 4 to 0 | AUTO_STANDBY_EN | H'01 | R/W | <p>Automatic standby control</p> <p>0: Does not initialize VD or FLD. 1: Initializes VD and FLD</p> <p>[0]: ECC 2-bit error [1]: ECC 1-bit error on receiving FS [2]: ECC 1-bit error on receiving FE [3]: ECC 1-bit error on receiving data other than FS or FE [4]: CRC error</p> <p>These bits control initialization of VD and FLD when an error occurs. To initialize the signals, set 1 to these bits.</p> |

29.2.10 Long Data Type Setting Register 0 (LNGDT0)

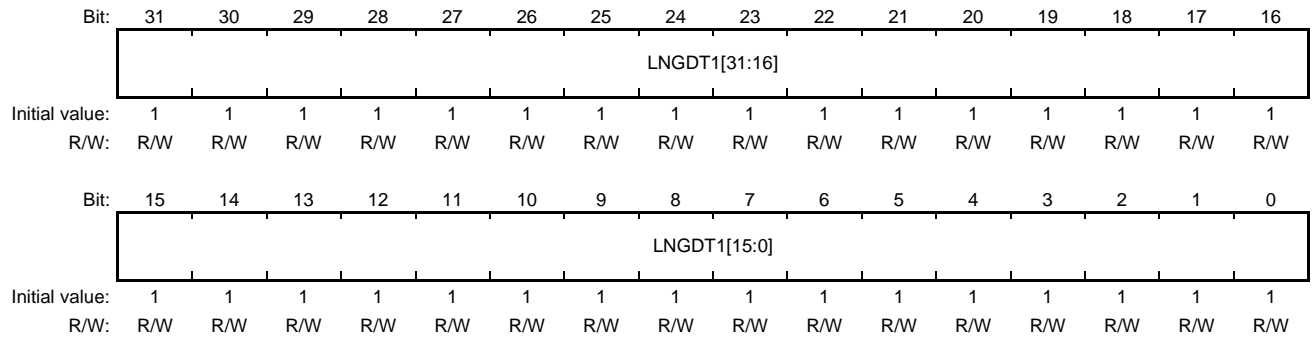
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | LNGDT0 | H'FFFF_0000 | R/W | <p>Long data type setting</p> <p>0: Short packet</p> <p>1: Long packet</p> <p>This register is combined with LNGDT1 register as in LNGDT = {LNGDT1, LNGDT0}.</p> <p>When LNGDT[received DT[5:0]] is 1, data is determined as a long packet and it is determined as a short packet when 0.</p> <p>According to the MIPI standard, DT = H'00 to H'0F is defined as a short packet and DT = H'10 to H'3F is defined as a long packet; to satisfy these conditions, the initial values of LNGDT are 0 in bits 0 to 15 and 1 in bits 16 to 63.</p> |

29.2.11 Long Data Type Setting Register 1 (LNGDT1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | LNGDT1 | H'FFFF_FFFF | R/W | Long data type setting 1 0: Short packet 1: Long packet These bits set the long packet or short packet as the data type (assigns H'3F to H'20). |

29.2.12 Interrupt Enable Register (INTEN)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----|------------|-----|-----|-----|-----|-----|----------|-----|-----|----------|-----|-----|-----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | IEN[29:16] | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | IEN[15:10] | | | | | | — | — | IEN[7:6] | | — | IEN[4:2] | | | — | IEN[0] |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R | R/W | R/W | R/W | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 31, 30 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 | IEN[29] | B'0 | R/W | Interrupt 29 (INT_EBD_CH0) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 28 | IEN[28] | B'0 | R/W | Interrupt 28 (INT_LESS_THAN_WC) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 27 | IEN[27] | B'0 | R/W | Interrupt 27 (INT_AFIFO_OF) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 26 | IEN[26] | B'0 | R/W | Interrupt 26 (INT_VD4_START) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 25 | IEN[25] | B'0 | R/W | Interrupt 25 (INT_VD4_END) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 24 | IEN[24] | B'0 | R/W | Interrupt 24 (INT_VD3_START) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 23 | IEN[23] | B'0 | R/W | Interrupt 23 (INT_VD3_END) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 22 | IEN[22] | B'0 | R/W | Interrupt 22 (INT_VD2_START) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 21 | IEN[21] | B'0 | R/W | Interrupt 21 (INT_VD2_END) enable 0: Disables the interrupt. 1: Enables the interrupt. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 20 | IEN[20] | B'0 | R/W | Interrupt 20 (INT_VD_START) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 19 | IEN[19] | B'0 | R/W | Interrupt 19 (INT_VD_END) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 18 | IEN[18] | B'0 | R/W | Interrupt 18 (INT_SHP_STB) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 17 | IEN[17] | B'0 | R/W | Interrupt 17 (INT_FSFE) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 16 | IEN[16] | B'0 | R/W | Interrupt 16 (INT_LNP_STB) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 15 | IEN[15] | B'0 | R/W | Interrupt 15 (INT_CRC_ERR) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 14 | IEN[14] | B'0 | R/W | Interrupt 14 (INT_HD_WC_ZERO) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 13 | IEN[13] | B'0 | R/W | Interrupt 13 (INT_FRM_SEQ_ERR1) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 12 | IEN[12] | B'0 | R/W | Interrupt 12 (INT_FRM_SEQ_ERR0) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 11 | IEN[11] | B'0 | R/W | Interrupt 11 (INT_ECC_ERR) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 10 | IEN[10] | B'0 | R/W | Interrupt 10 (INT_ECC_CRCT_ERR) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 9 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 8 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 7 | IEN[7] | B'0 | R/W | Interrupt 7 (INT_ULPS_START) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 6 | IEN[6] | B'0 | R/W | Interrupt 6 (INT_ULPS_END) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 5 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 4 | IEN[4] | B'0 | R/W | Interrupt 4 (INT_ERRSOTHS) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 3 | IEN[3] | B'0 | R/W | Interrupt 3 (INT_ERRSOTSYNCHS) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 2 | IEN[2] | B'0 | R/W | Interrupt 2 (INT_ERRESC) enable 0: Disables the interrupt. 1: Enables the interrupt. |
| 1 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 0 | IEN[0] | B'0 | R/W | Interrupt 0 (INT_ERRCONTROL) enable 0: Disables the interrupt. 1: Enables the interrupt. |

- Notes:
1. The settings of this register are applied the instant the register is set (immediate reflection).
 2. Do not change the settings in this register while the CSI2 is operating.

29.2.13 Interrupt Source Mask Register (INTCLOSE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | | |
|----------------|------------|-----|------------|-----|-----|-----|-----|-----|----------|-----|-----|----------|-----|-----|--------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | ICL[29:16] | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | ICL[15:10] | | | | | | — | — | ICL[7:6] | | — | ICL[4:2] | | — | ICL[0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R | R/W | R/W | R/W | R | R/W | |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 31, 30 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 | ICL[29] | B'0 | R/W | Interrupt 29 (INT_EBD_CH0) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 28 | ICL[28] | B'0 | R/W | Interrupt 28 (INT_LESS_THAN_WC) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 27 | ICL[27] | B'0 | R/W | Interrupt 27 (INT_AFIFO_OF) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 26 | ICL[26] | B'0 | R/W | Interrupt 26 (INT_VD4_START) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 25 | ICL[25] | B'0 | R/W | Interrupt 25 (INT_VD4_END) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 24 | ICL[24] | B'0 | R/W | Interrupt 24 (INT_VD3_START) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 23 | ICL[23] | B'0 | R/W | Interrupt 23 (INT_VD3_END) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 22 | ICL[22] | B'0 | R/W | Interrupt 22 (INT_VD2_START) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 21 | ICL[21] | B'0 | R/W | Interrupt 21 (INT_VD2_END) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 20 | ICL[20] | B'0 | R/W | Interrupt 20 (INT_VD_START) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 19 | ICL[19] | B'0 | R/W | Interrupt 19 (INT_VD_END) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 18 | ICL[18] | B'1 | R/W | Interrupt 18 (INT_SHP_STB) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 17 | ICL[17] | B'0 | R/W | Interrupt 17 (INT_FSFE) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 16 | ICL[16] | B'0 | R/W | Interrupt 16 (INT_LNP_STB) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 15 | ICL[15] | B'0 | R/W | Interrupt 15 (INT_CRC_ERR) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 14 | ICL[14] | B'0 | R/W | Interrupt 14 (INT_HD_WC_ZERO) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 13 | ICL[13] | B'0 | R/W | Interrupt 13 (INT_FRM_SEQ_ERR1) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 12 | ICL[12] | B'0 | R/W | Interrupt 12 (INT_FRM_SEQ_ERR0) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 11 | ICL[11] | B'0 | R/W | Interrupt 11 (INT_ECC_ERR) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 10 | ICL[10] | B'0 | R/W | Interrupt 10 (INT_ECC_CRCT_ERR) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 9 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 8 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 7 | ICL[7] | B'0 | R/W | Interrupt 7 (INT_ULPS_START) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 6 | ICL[6] | B'0 | R/W | Interrupt 6 (INT_ULPS_END) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 5 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 4 | ICL[4] | B'0 | R/W | Interrupt 4 (INT_ERRSOTHS) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 3 | ICL[3] | B'0 | R/W | Interrupt 3 (INT_ERRSOTSYNCHS) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 2 | ICL[2] | B'0 | R/W | Interrupt 2 (INT_ERRESC) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |
| 1 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 0 | ICL[0] | B'0 | R/W | Interrupt 0 (INT_ERRCONTROL) source mask 0: Enables the interrupt source. 1: Masks the interrupt source. |

Note: The settings of this register are applied the instant the register is set (immediate reflection).

29.2.14 Interrupt Status Monitor Register (INTSTATE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|-------|-------------|-------|-------|-------|-------|-------|----------|-------|-------|----------|-------|-------|-------|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | IST [29:16] | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/WC1 | R/WC1 | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | IST[15:10] | | | | | | — | — | IST[7:6] | | — | IST[4:2] | | | — | IST[0] |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R | R | R/WC1 | R/WC1 | R | R | R | R/WC1 | R | R/WC1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-------|--|
| 31, 30 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 | IST[29] | B'0 | R/WC1 | Interrupt 29 (INT_EBD_CH0) monitor When this bit is written to, the interrupt is cleared. 1: Clears the interrupt. When read, the status is monitored. 0: Normal 1: The interrupt has been generated. Interrupt 29 is a reception end interrupt of embedded data from output Ch0. It is generated when the embedded data is received. |
| 28 | IST[28] | B'0 | R/WC1 | Interrupt 28 (INT_LESS_THAN_WC) monitor When this bit is written to, the interrupt is cleared. 1: Clears the interrupt. When read, the status is monitored. 0: Normal 1: The interrupt has been generated. Interrupt 28 is an error interrupt that is generated when the length of payload data of a long packet is less than the WC value. |
| 27 | IST[27] | B'0 | R | Interrupt 27 (INT_AFIFO_OF) monitor (read-only) 0: Normal 1: The interrupt has been generated. Interrupt 27 is an overflow interrupt of the asynchronous FIFO, in which HS data from PHY is stored. IST[27] does not have the corresponding register (DFF); the OR of the lanes of the interrupt error status monitoring register (IEST[11:8]) generates the interrupt. Therefore, the interrupt is cleared using the interrupt error status monitoring register. Check the relevant registers such as the PHY data lane monitoring register since synchronization between lanes may not be taken or data may not be received in a specific lane. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 26 | IST[26] | B'0 | R/WC1 | <p>Interrupt 26 (INT_VD4_START) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 26 is generated by the start of VD4 output from the CSI2 (a frame start interrupt). It is issued in synchronization with the rising edge of the CSIR_VD[3] signal.</p> |
| 25 | IST[25] | B'0 | R/WC1 | <p>Interrupt 25 (INT_VD4_END) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 25 is generated by the end of VD4 output from the CSI2 (a frame end interrupt). It is issued in synchronization with the falling edge of the CSIR_VD[3] signal.</p> |
| 24 | IST[24] | B'0 | R/WC1 | <p>Interrupt 24 (INT_VD3_START) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 24 is generated by the start of V output from the CSI2 (a frame start interrupt). It is issued in synchronization with the rising edge of the CSIR_VD[2] signal.</p> |
| 23 | IST[23] | B'0 | R/WC1 | <p>Interrupt 23 (INT_VD3_END) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 23 is generated by the end of V output from the CSI2 (a frame end interrupt). It is issued in synchronization with the falling edge of the CSIR_VD[2] signal.</p> |
| 22 | IST[22] | B'0 | R/WC1 | <p>Interrupt 22 (INT_VD2_START) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 22 is generated by the start of VD2 output from the CSI2 (a frame start interrupt). It is issued in synchronization with the rising edge of the CSIR_VD[1] signal.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 21 | IST[21] | B'0 | R/WC1 | <p>Interrupt 21 (INT_VD2_END) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 21 is generated by the end of VD2 output from the CSI2 (a frame end interrupt). It is issued in synchronization with the falling edge of the CSIR_VD[1] signal.</p> |
| 20 | IST[20] | B'0 | R/WC1 | <p>Interrupt 20 (INT_VD_START) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 20 is generated by the start of VD output from the CSI2 (a frame start interrupt). It is issued in synchronization with the rising edge of the CSIR_VD[0] signal.</p> |
| 19 | IST[19] | B'0 | R/WC1 | <p>Interrupt 19 (INT_VD_END) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 19 is generated by the end of VD output from the CSI2 (a frame end interrupt). It is issued in synchronization with the falling edge of the CSIR_VD[0] signal.</p> |
| 18 | IST[18] | B'0 | R/WC1 | <p>Interrupt 18 (INT_SHP_STB) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 18 is a short packet reception interrupt. It is issued upon reception of short packets such as Frame Start and Frame End packets.</p> |
| 17 | IST[17] | B'0 | R/WC1 | <p>Interrupt 17 (INT_FSFE) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 17 is a frame packet reception interrupt. It is issued upon reception of short packets of Frame Start and Frame End packets.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 16 | IST[16] | B'0 | R/WC1 | <p>Interrupt 16 (INT_LNP_STB) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 16 is a long packet reception interrupt. It is issued upon reception of long packets such as YUV data and Embedded Data.</p> |
| 15 | IST[15] | B'0 | R/WC1 | <p>Interrupt 15 (INT_CRC_ERR) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 15 is a CRC error interrupt. It is issued if there is a data error in a long packet. To perform the CRC check, set the CRC_EN bit in the CHKSUM register to 1.</p> |
| 14 | IST[14] | B'0 | R/WC1 | <p>Interrupt 14 (INT_HD_WC_ZERO) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 14 is a WC (word count) zero interrupt. It is issued when the WC value in the packet header of a long packet is zero.</p> |
| 13 | IST[13] | B'0 | R/WC1 | <p>Interrupt 13 (INT_FRM_SEQ_ERR1) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 13 is a frame sequence error 1 interrupt. It is issued when an illegal Frame End packet is received. Specifically, it is issued in the following cases (assuming that the same channel is used):</p> <ul style="list-style-type: none"> - A Frame End packet is received before a Frame Start packet is received. - Frame End packets are continuously received. |
| 12 | IST[12] | B'0 | R/WC1 | <p>Interrupt 12 (INT_FRM_SEQ_ERR0) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 12 is a frame sequence error 0 interrupt. It is issued when an illegal Frame Start packet is received. Specifically, it is issued in the following case (assuming that the same channel is used):</p> <ul style="list-style-type: none"> - Frame Start packets are continuously received. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 11 | IST[11] | B'0 | R/WC1 | <p>Interrupt 11 (INT_ECC_ERR) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 11 is an ECC error interrupt. It is issued when two or more bits of errors are found in a packet header. To perform the ECC processing, set the ECC_EN bit in the CHKSUM register to 1.</p> |
| 10 | IST[10] | B'0 | R/WC1 | <p>Interrupt 10 (INT_ECC_CRCT_ERR) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 10 is an ECC 1-bit correction interrupt. It is issued when 1-bit correction is performed for a packet header during ECC processing. To perform the ECC processing, set the ECC_EN bit in the CHKSUM register to 1.</p> |
| 9 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 8 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 7 | IST[7] | B'0 | R/WC1 | <p>Interrupt 7 (INT_ULPS_START) monitor [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 7 is an ultra-low power data transfer start interrupt. This bit is set to high when any of the lanes shifts to the ULP state. For the lane state, see the description of the PHY data lane monitoring register (PHDLM).</p> <hr/> <p>Interrupt 7 (INT_ULPS_START) monitor [RZ/G2H, RZ/G2N]</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 7 is an ultra-low power data transfer start interrupt. This bit is set to high when any of the lanes shifts to the ULP state. For the lane state, see the description of the PHY data lane monitoring register (PHDLM).</p> <p>It is need to write to 1'b1 after initial sequence of PHY.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 6 | IST[6] | B'0 | R/WC1 | <p>Interrupt 6 (INT_ULPS_END) monitor [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 6 is an ultra-low power data transfer end interrupt. This bit is set to high when no lanes are in the ULP state. For the lane state, see the description of the PHY data lane monitoring register (PHDLM).</p> <hr/> <p>Interrupt 6 (INT_ULPS_END) monitor [RZ/G2H, RZ/G2N]</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 6 is an ultra-low power data transfer end interrupt. This bit is set to high when no lanes are in the ULP state. For the lane state, see the description of the PHY data lane monitoring register (PHDLM).</p> <p>It is need to write to 1'b1 after initial sequence of PHY.</p> |
| 5 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 4 | IST[4] | B'0 | R | <p>Interrupt 4 (INT_ERRSOTHS) monitor (read-only)</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Interrupt 4 is a synchronized SOT (start of transfer) error interrupt during HS reception. This bit is set to high when an error is detected in any of the lanes. For the lane state, see the description of the interrupt error status monitoring register (INTERRSTATE).</p> <p>IST[4] does not have the corresponding register (DFF); the OR of the lanes of the interrupt error status monitoring register (IEST[7:4]) generates the interrupt.</p> <p>Since the ERRSOTHS_[3-0] input from the PHY are pulse signals, the state are not retained. Therefore, the interrupt error status register is provided as the interrupt status register for each lane. Therefore, the interrupt is cleared using the interrupt error status monitoring register.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 3 | IST[3] | B'0 | R | <p>Interrupt 3 (INT_ERRSOTSYNCHS) monitor (read-only)</p> <p>0: Normal 1: The interrupt has been generated.</p> <p>Interrupt 3 is a non-synchronizable SOT (start of transfer) error interrupt during HS reception. This bit is set to high when an error is detected in any of the lanes. For the lane state, see the description of the interrupt error status monitoring register (INTERRSTATE).</p> <p>IST[4] does not have the corresponding register (DFF); the OR of the lanes of the interrupt error status monitoring register (IEST[7:4]) generates the interrupt.</p> <p>Since the ERRSOTSYNCHS_[3-0] input from the PHY are pulse signals, the state are not retained. Therefore, the interrupt error status register is prepared as the interrupt status register for each lane. Therefore, the interrupt is cleared using the interrupt error status monitoring register.</p> |
| 2 | IST[2] | B'0 | R/WC1 | <p>Interrupt 2 (INT_ERRESC) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt. When read, the status is monitored. 0: Normal 1: The interrupt has been generated.</p> <p>Interrupt 2 is an escape mode entry error interrupt. It is issued when an unrecognizable escape entry command is received. This bit is set to high when an error is detected in any of the lanes. For the lane state, see the description of the PHY ESC error monitoring register (PHEERM). Even after the interrupt is cleared, the PHEERM register retains the high-level state until a stop state is received.</p> |
| 1 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 0 | IST[0] | B'0 | R/WC1 | <p>Interrupt 0 (INT_ERRCONTROL) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt. When read, the status is monitored. 0: Normal 1: The interrupt has been generated.</p> <p>Interrupt 0 is a PHY control error interrupt. It is issued when a transition is made to an incorrect line state. This bit is set to high when an error is detected in any of the lanes. For the lane state, see the description of the PHY ESC error monitoring register (PHEERM). Even after the interrupt is cleared, the PHEERM register retains the high-level state until a stop state is received.</p> |

29.2.15 Interrupt Error Status Monitor Register (INTERRSTATE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | IEST/—* | | | IEST/—* | | | IEST | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/WC1, R* | R/WC1, R* | R/WC1, R* | R/WC1, R* | R/WC1, R* | R/WC1, R* | R/WC1, R* | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 |

Note: * For reserved bit (Bit Name: —). Refer to the following table.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | IEST[15] | B'0 | R/WC1 | Error interrupt 15 (INT_FIFO_OF_3) monitor [RZ/G2H, RZ/G2N, RZ/G2E] When this bit is written to, the interrupt is cleared. 1: Clears the interrupt. When read, the status is monitored. 0: Normal 1: The interrupt has been generated. Error interrupt 11 is an overflow interrupt of the asynchronous FIFO, in which HS data from PHY is stored. When ICL27 (interrupt source mask register) is set to 1, INT_FIFO_OF_3 is cleared to 0. |
| | — | B'0 | R | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] This bit is always read as 0. The write value should always be 0. |
| 14 | IEST[14] | B'0 | R/WC1 | Error interrupt 14 (INT_FIFO_OF_2) monitor [RZ/G2H, RZ/G2N, RZ/G2E] When this bit is written to, the interrupt is cleared. 1: Clears the interrupt. When read, the status is monitored. 0: Normal 1: The interrupt has been generated. Error interrupt 11 is an overflow interrupt of the asynchronous FIFO, in which HS data from PHY is stored. When ICL27 (interrupt source mask register) is set to 1, INT_FIFO_OF_2 is cleared to 0. |
| | — | B'0 | R | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 13 | IEST[13] | B'0 | R/WC1 | <p>Error interrupt 13 (INT_FIFO_OF_1) monitor [RZ/G2H, RZ/G2N, RZ/G2E]</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated. Error interrupt 11 is an overflow interrupt of the asynchronous FIFO, in which HS data from PHY is stored.</p> <p>When ICL27 (interrupt source mask register) is set to 1, INT_FIFO_OF_1 is cleared to 0.</p> |
| | — | B'0 | R | <p>Reserved [RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 12 | IEST[12] | B'0 | R/WC1 | <p>Error interrupt 12 (INT_FIFO_OF_0) monitor [RZ/G2H, RZ/G2N, RZ/G2E]</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated. Error interrupt 11 is an overflow interrupt of the asynchronous FIFO, in which HS data from PHY is stored.</p> <p>When ICL27 (interrupt source mask register) is set to 1, INT_FIFO_OF_0 is cleared to 0.</p> |
| | — | B'0 | R | <p>Reserved [RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 11 | IEST[11] | B'0 | R/WC1 | <p>Error interrupt 11 (INT_AFIFO_OF_3) monitor [RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Note: When 2-lane PHY is connected, the interrupt is not generated.</p> <p>Error interrupt 11 is an overflow interrupt of the asynchronous FIFO, in which HS data from PHY is stored.</p> <p>When ICL27 (interrupt source mask register) is set to 1, INT_AFIFO_OF_3 is cleared to 0.</p> |
| | — | B'0 | R | <p>Reserved [RZ/G2H, RZ/G2N, RZ/G2E]</p> <p>This bit is always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 10 | IEST[10] | B'0 | R/WC1 | <p>Error interrupt 10 (INT_AFIFO_OF_2) monitor [RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Note: When 2-lane PHY is connected, the interrupt is not generated.</p> <p>Error interrupt 10 is an overflow interrupt of the asynchronous FIFO, in which HS data from PHY is stored.</p> <p>When ICL27 (interrupt source mask register) is set to 1, INT_AFIFO_OF_2 is cleared to 0.</p> |
| — | — | B'0 | R | <p>Reserved [RZ/G2H, RZ/G2N, RZ/G2E]</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 9 | IEST[9] | B'0 | R/WC1 | <p>Error interrupt 9 (INT_AFIFO_OF_1) monitor [RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Error interrupt 9 is an overflow interrupt of the asynchronous FIFO, in which HS data from PHY is stored.</p> <p>When ICL27 (interrupt source mask register) is set to 1, INT_AFIFO_OF_1 is cleared to 0.</p> |
| — | — | B'0 | R | <p>Reserved [RZ/G2H, RZ/G2N, RZ/G2E]</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 8 | IEST[8] | B'0 | R/WC1 | <p>Error interrupt 8 (INT_AFIFO_OF_0) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Error interrupt 8 is an overflow interrupt of the asynchronous FIFO, in which HS data from PHY is stored.</p> <p>When ICL27 (interrupt source mask register) is set to 1, INT_AFIFO_OF_0 is cleared to 0.</p> |
| 7 | IEST[7] | B'0 | R/WC1 | <p>Error interrupt 7 (INT_ERRSOTHS_3) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Note: When 2-lane PHY is connected, the interrupt is not generated.</p> <p>Error interrupt 7 is a synchronized SOT (start of transfer) error interrupt during HS reception. This bit is set to high when an error is detected in lane 3. When ICL04 (interrupt source mask register) is set to 1, INT_ERRSOTHS_3 is cleared to 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 6 | IEST[6] | B'0 | R/WC1 | <p>Error interrupt 6 (INT_ERRSOTHS_2) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Note: When 2-lane PHY is connected, the interrupt is not generated.</p> <p>Error interrupt 6 is a synchronized SOT (start of transfer) error interrupt during HS reception. This bit is set to high when an error is detected in lane 2. When ICL04 (interrupt source mask register) is set to 1, INT_ERRSOTHS_2 is cleared to 0.</p> |
| 5 | IEST[5] | B'0 | R/WC1 | <p>Error interrupt 5 (INT_ERRSOTHS_1) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Error interrupt 5 is a synchronized SOT (start of transfer) error interrupt during HS reception. This bit is set to high when an error is detected in lane 1. When ICL04 (interrupt source mask register) is set to 1, INT_ERRSOTHS_1 is cleared to 0.</p> |
| 4 | IEST[4] | B'0 | R/WC1 | <p>Error interrupt 4 (INT_ERRSOTHS_0) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Error interrupt 4 is a synchronized SOT (start of transfer) error interrupt during HS reception. This bit is set to high when an error is detected in lane 0. When ICL04 (interrupt source mask register) is set to 1, INT_ERRSOTHS_0 is cleared to 0.</p> |
| 3 | IEST[3] | B'0 | R/WC1 | <p>Error interrupt 3 (INT_ERRSOTSYNCHS_3) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Note: When 2-lane PHY is connected, the interrupt is not generated.</p> <p>Error interrupt 3 is a non-synchronizable SOT (start of transfer) error interrupt during HS reception. This bit is set to high when an error is detected in lane 3. When ICL03 (interrupt source mask register) is set to 1, INT_ERRSOTSYNCHS_3 is cleared to 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 2 | IEST[2] | B'0 | R/WC1 | <p>Error interrupt 2 (INT_ERRSOTSYNCHS_2) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Note: When 2-lane PHY is connected, the interrupt is not generated.</p> <p>Error interrupt 2 is a non-synchronizable SOT (start of transfer) error interrupt during HS reception. This bit is set to high when an error is detected in lane 2. When ICL03 (interrupt source mask register) is set to 1, INT_ERRSOTSYNCHS_2 is cleared to 0.</p> |
| 1 | IEST[1] | B'0 | R/WC1 | <p>Error interrupt 1 (INT_ERRSOTSYNCHS_1) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Error interrupt 1 is a non-synchronizable SOT (start of transfer) error interrupt during HS reception. This bit is set to high when an error is detected in lane 1. When ICL03 (interrupt source mask register) is set to 1, INT_ERRSOTSYNCHS_1 is cleared to 0.</p> |
| 0 | IEST[0] | B'0 | R/WC1 | <p>Error interrupt 0 (INT_ERRSOTSYNCHS_0) monitor</p> <p>When this bit is written to, the interrupt is cleared.</p> <p>1: Clears the interrupt.</p> <p>When read, the status is monitored.</p> <p>0: Normal</p> <p>1: The interrupt has been generated.</p> <p>Error interrupt 0 is a non-synchronizable SOT (start of transfer) error interrupt during HS reception. This bit is set to high when an error is detected in lane 0. When ICL03 (interrupt source mask register) is set to 1, INT_ERRSOTSYNCHS_0 is cleared to 0.</p> |

29.2.16 Short Packet Data Register (SHPDAT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----|----|----|----|----|------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | ECC | | | | | | | | DATA[15:8] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DATA[7:0] | | | | | | | VC | | DT | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|-------------------------------|
| 31 to 24 | ECC | All 0 | R | ECC data received |
| 23 to 8 | DATA | All 0 | R | Short packet data received |
| 7, 6 | VC | B'00 | R | Virtual channel data received |
| 5 to 0 | DT | H'00 | R | Data type received |

SHPDAT is a register to read the received short packet data. The value of this register is stored in the FIFO and the number of packets stored can be checked by reading the short packet count register (SHPCNT). Short packets with the data types H'00 (frame start), H'01 (frame end), and H'08 (Generic Short Packet Code 1, DSI End of Transmission) are not stored.

29.2.17 Short Packet Count Register (SHPCNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|-----|----|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | OVF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | NUM | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | OVF | B'0 | R | FIFO overflow bit FIFO has overflowed. The received data may have been lost. |
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | NUM | H'0 | R | Stored short packet count The number of short packets stored in the FIFO can be read. Up to nine packets can be stored. |

29.2.18 LINK Operation Control Register (LINKCNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|----|----|----|----|----|------------------|--------------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MONITOR_EN | — | — | — | — | — | REG_MONI_PACT_EN | ICLK_NONSTOP | — | — | — | — | — | — | — | — |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|---|
| 31 | MONITOR_EN | B'1 | R/W | Monitoring function control 0: Disable 1: Enable This bit updates the monitor state to be read by the registers. |
| 30 to 26 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25 | REG_MONI_PACT_EN | B'0 | R/W | Monitoring packet counter enable 0: Disable 1: Enable This bit controls the packet counter for monitoring. |
| 24 | ICLK_NONSTOP | B'0 | R/W | Stopping clock function control 0: Enable 1: Disable This bit should be set to 1 during the initialization sequence. |
| 23 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

29.2.19 Lane Swap Register (LSWAP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------|-----|-------|-----|-------|-----|-------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | L3SEL | | L2SEL | | L1SEL | | L0SEL | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7, 6 | L3SEL | H'3 | R/W | LINK lane 3 select bits H'0: Uses PHY lane 0 H'1: Uses PHY lane 1 H'2: Uses PHY lane 2 H'3: Uses PHY lane 3 Note: When 2-lane PHY is connected, this setting has no effect. These bits select a PHY lane to be assigned to LINK lane 3. These bits should be set before the start of PHY operation and should not be modified during operation. |
| 5, 4 | L2SEL | H'2 | R/W | LINK lane 2 select bits H'0: Uses PHY lane 0 H'1: Uses PHY lane 1 H'2: Uses PHY lane 2 H'3: Uses PHY lane 3 Note: When 2-lane PHY is connected, this setting has no effect. These bits select a PHY lane to be assigned to LINK lane 2. These bits should be set before the start of PHY operation and should not be modified during operation. |
| 3, 2 | L1SEL | H'1 | R/W | LINK lane 1 select bits H'0: Uses PHY lane 0 H'1: Uses PHY lane 1 H'2: Uses PHY lane 2 H'3: Uses PHY lane 3 These bits select a PHY lane to be assigned to LINK lane 1. These bits should be set before the start of PHY operation and should not be modified during operation. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 1, 0 | L0SEL | H'0 | R/W | LINK lane 0 select bits H'0: Uses PHY lane 0 H'1: Uses PHY lane 1 H'2: Uses PHY lane 2 H'3: Uses PHY lane 3 These bits select a PHY lane to be assigned to LINK lane 0. These bits should be set before the start of PHY operation and should not be modified during operation. |

[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]

When modifying this register, be sure to set ENABLE_[3-0] appropriately. For example, when L1SEL = H'3 and L0SEL = H'2 are set to perform two-lane transfer using PHY lanes 3 and 2, it is necessary to set:

ENABLE_3 = 1, ENABLE_2 = 1, ENABLE_1 = 0, ENABLE_0 = 0, and ENABLECLK = 1.

Note: The settings of this register are applied the instant the register is set (immediate reflection).

[RZ/G2H, RZ/G2N]

When modifying this register, must be set ENABLE_0 to 1.

29.2.20 PHY Test Interface Write Register (PHTW)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | √ |

| | | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|------|--------------|-----|-----|-----|-----|-----|-----|-----|--|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | | |
| | — | — | — | — | — | — | — | DWEN | TESTDIN_DATA | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | |
| | — | — | — | — | — | — | — | CWEN | TESTDIN_CODE | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|---|
| 31 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | DWEN | B'0 | R/W | TESTDIN_DATA enable 0: Disable 1: Enable About setting, Refer to section 29.3.9 PHY Control and Monitoring through Register Setting. This bit is auto cleared when value of TESTDIN_DATA sets to PHY. If DWEN is set to 1'b1, it should be read PHTW to confirm that DWEN is auto cleared to 1'b0. |
| 23 to 16 | TESTDIN_DATA | All 0 | R/W | For the phy control and set values, refer to section 29.3.9 PHY Control and Monitoring through Register Setting. |
| 15 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | CWEN | B'0 | R/W | TESTDIN_CODE enable 0: Disable 1: Enable About setting, Refer to section 29.3.9 PHY Control and Monitoring through Register Setting. This bit is auto cleared when value of TESTDIN_CODE sets to PHY. If CWEN is set to 1'b1, it should be read PHTW to confirm that CWEN is auto cleared to 1'b0. |
| 7 to 0 | TESTDIN_CODE | All 0 | R/W | For the phy control and set values, refer to section 29.3.9 PHY Control and Monitoring through Register Setting. |

29.2.21 PHY Test Interface Read Register (PHTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | √ |

| | | | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----------|----|----|----|----|----|----|----|--|--|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | | | |
| | — | — | — | — | — | — | — | — | TESTDOUT | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | TESTDOUT | All 0 | R | Those bits are TESTDIN_DATA loopback value from CSI2 PHY. If those bits are read, it is necessary to confirm DWEN and CWEN in PHTW is 1'b0. [RZ/G2E] If Table 29.11 is applied to PHTW, TESTDIN_DATA[6:0] is loopback to TESTDOUT[6:0] (TESTDOUT[7] is fixed to 1'b0). If Other than the above setting is applied to PHTW, TESTDOUT_DATA[5:0] is loopback to TESTDOUT[5:0] (TESTDOUT[7:6] is fixed 2'b00). |
| 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

29.2.22 PHY Test Interface Clear Register (PHTC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | TEST CLR |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | TESTCLR | B'1 | R/W | This bit sets TESTCLR of the PHY. |

Note: The settings of this register are applied the instant the register is set (immediate reflection).

29.2.23 PHY Frequency Control Register (PHYPLL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|------------------|-------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | HSFREQRANGE / —* | HSFREQRANGE | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W, R* | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

Note: * For reserved bit (Bit Name: —). Refer to the following table.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|--|
| 31 to 23 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 22 | HSFREQRANGE[6] | B'0 | R/W | PHY high speed range select bits [RZ/G2H, RZ/G2N] These bits set the HSFREQRANGE pin of the PHY. These bits should be set according to the transfer rate. For the received frequencies and set values, see Table 29.9, HSFREQRANGE Bit Set Values. |
| | — | B'0 | R | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] This bit is always read as 0. The write value should always be 0. |
| 21 to 16 | HSFREQRANGE[5:0] | H'00 | R/W | PHY high speed range select bits [RZ/G2H, RZ/G2N] These bits set the HSFREQRANGE pin of the PHY. These bits should be set according to the transfer rate. For the received frequencies and set values, see Table 29.9, HSFREQRANGE Bit Set Values. PHY high speed range select bits [RZ/G2M V1.3, RZ/G2M V3.0] These bits set the HSFREQRANGE pin of the PHY. These bits should be set according to the transfer rate. For the received frequencies and set values, see Table 29.10, HSFREQRANGE Bit Set Values. |
| 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

Note: The settings of this register are applied the instant the register is set (immediate reflection).

29.2.24 PHY ESC Error Monitor Register (PHEERM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----------|----------|----------|----------|----|----|----|----|---------------|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | ERRESC_3 | ERRESC_2 | ERRESC_1 | ERRESC_0 | — | — | — | — | ERRCONTR_OL_3 | ERRCONTR_OL_2 | ERRCONTR_OL_1 | ERRCONTR_OL_0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 | ERRESC_3 | B'0 | R | Lane 3 escape mode error bit Note: When 2-lane PHY is connected, the error is not generated. If an unrecognizable escape entry command is received, the pin is set to high and the state is retained until a transition is made to the stop state. |
| 10 | ERRESC_2 | B'0 | R | Lane 2 escape mode error bit Note: When 2-lane PHY is connected, the error is not generated. If an unrecognizable escape entry command is received, the pin is set to high and the state is retained until a transition is made to the stop state. |
| 9 | ERRESC_1 | B'0 | R | Lane 1 escape mode error bit If an unrecognizable escape entry command is received, the pin is set to high and the state is retained until a transition is made to the stop state. |
| 8 | ERRESC_0 | B'0 | R | Lane 0 escape mode error bit If an unrecognizable escape entry command is received, the pin is set to high and the state is retained until a transition is made to the stop state. |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 5 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 4 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 3 | ERRCONTR_OL_3 | B'0 | R | Lane 3 status control error bit Note: When 2-lane PHY is connected, the error is not generated. If an incorrect line state is entered during entry to HS or escape mode, the pin is set to high and the state is retained until a transition is made to the stop state. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|------------------|---------------|-----|---|
| 2 | ERRCONTR OL_2 | B'0 | R | Lane 2 status control error bit Note: When 2-lane PHY is connected, the error is not generated. If an incorrect line state is entered during entry to HS or escape mode, the pin is set to high and the state is retained until a transition is made to the stop state. |
| 1 | ERRCONTR OL_1 | B'0 | R | Lane 1 status control error bit If an incorrect line state is entered during entry to HS or escape mode, the pin is set to high and the state is retained until a transition is made to the stop state. |
| 0 | ERRCONTR OL_0 | B'0 | R | Lane 0 status control error bit If an incorrect line state is entered during entry to HS or escape mode, the pin is set to high and the state is retained until a transition is made to the stop state. |

29.2.25 PHY Clock Lane Monitor Register (PHCLM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----------------------|------------------|-------------------|------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | ULPSACTIVENO TCLK | RXULPS CLKNOT | RXCLKACTIVEH S | STOPST ATECLK |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------------|---------------|-----|---|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 | ULPSACTIVE NOTCLK | B'1 | R | This bit indicates the ULP state on the clock lane. Negative logic After the completion of ULP, the pin is set to high without waiting for Twakeup. |
| 2 | RXULPSCLK NOT | B'1 | R | This bit indicates the ULP state on the clock lane. Negative logic After the completion of ULP, the pin is set to high without waiting for Twakeup. |
| 1 | RXCLKACTI VEHS | B'0 | R | This bit indicates that the clock lane is receiving the HS DRR clock. |
| 0 | STOPSTATE CLK | B'0 | R | This bit indicates the stop state on the clock lane. |

29.2.26 PHY Data Lane Monitor Register (PHDLM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------------------|---------------------|---------------------|---------------------|--------------------|--------------------|--------------------|--------------------|----|----|----|----|---------------------------|---------------------------|---------------------------|---------------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | ULPSACTIVENO T_3 | ULPSACTIVENO T_2 | ULPSACTIVENO T_1 | ULPSACTIVENO T_0 | RXULPSESC ESC_3 | RXULPSESC ESC_2 | RXULPSESC ESC_1 | RXULPSESC ESC_0 | — | — | — | — | STOPSTAT ATEDAT A_3 | STOPSTAT ATEDAT A_2 | STOPSTAT ATEDAT A_1 | STOPSTAT ATEDAT A_0 |
| Initial value: | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | ULPSACTIVE NOT_3 | B'1 | R | This bit indicates the ULP state on lane 3. Negative logic Note: When 2-lane PHY is connected, this bit is fixed to 0. After the completion of ULP, the pin is set to high without waiting for Twakeup. |
| 14 | ULPSACTIVE NOT_2 | B'1 | R | This bit indicates the ULP state on lane 2. Negative logic Note: When 2-lane PHY is connected, this bit is fixed to 0. After the completion of ULP, the pin is set to high without waiting for Twakeup. |
| 13 | ULPSACTIVE NOT_1 | B'1 | R | This bit indicates the ULP state on lane 1. Negative logic After the completion of ULP, the pin is set to high without waiting for Twakeup. |
| 12 | ULPSACTIVE NOT_0 | B'1 | R | This bit indicates the ULP state on lane 0. Negative logic After the completion of ULP, the pin is set to high without waiting for Twakeup. |
| 11 | RXULPSESC _3 | B'0 | R | This bit indicates the ULP state on lane 3. Note: When 2-lane PHY is connected, this bit is fixed to 0. After the completion of ULP, the pin is set to low without waiting for Twakeup. |
| 10 | RXULPSESC _2 | B'0 | R | This bit indicates the ULP state on lane 2. Note: When 2-lane PHY is connected, this bit is fixed to 0. After the completion of ULP, the pin is set to low without waiting for Twakeup. |
| 9 | RXULPSESC _1 | B'0 | R | This bit indicates the ULP state on lane 1. After the completion of ULP, the pin is set to low without waiting for Twakeup. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|---------------------|---------------|-----|--|
| 8 | RXULPSESC _0 | B'0 | R | This bit indicates the ULP state on lane 0. After the completion of ULP, the pin is set to low without waiting for Twakeup. |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 5 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 4 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 3 | STOPSTATE DATA_3 | B'0 | R | This bit indicates the stop state on lane 3. Note: When 2-lane PHY is connected, this bit is fixed to 0. |
| 2 | STOPSTATE DATA_2 | B'0 | R | This bit indicates the stop state on lane 2. Note: When 2-lane PHY is connected, this bit is fixed to 0. |
| 1 | STOPSTATE DATA_1 | B'0 | R | This bit indicates the stop state on lane 1. |
| 0 | STOPSTATE DATA_0 | B'0 | R | This bit indicates the stop state on lane 0. |

29.2.27 Packet Header 0 Monitor Register 0 (PH0M0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|----|----|----|----|----|----|----|----------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | WC[15:8] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | WC[7:0] | | | | | | | | VC | | DT | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 8 | WC | All 0 | R | Word count data received |
| 7, 6 | VC | B'00 | R | Virtual channel data received |
| 5 to 0 | DT | H'00 | R | Data type data received |

PH0M0 is a register to monitor the received packet header.

29.2.28 Packet Header 0 Monitor Register 1 (PH0M1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PH_CNT | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | PH_CNT | All 0 | R | PH0M0 register consecutive reception count These bits monitor the number of times of consecutive receptions for the PH0M0 register. |

29.2.29 Packet Header 1 Monitor Register 0 (PH1M0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|----|----|----|----|----|----|----|----------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | WC[15:8] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | WC[7:0] | | | | | | | | VC | | DT | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 8 | WC | All 0 | R | Word count data received |
| 7, 6 | VC | B'00 | R | Virtual channel data received |
| 5 to 0 | DT | H'00 | R | Data type data received |

PH1M0 is a register to monitor the received packet header (previous data of PH0M0).

29.2.30 Packet Header 1 Monitor Register 1 (PH1M1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PH_CNT | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | PH_CNT | All 0 | R | PH1M0 register consecutive reception count These bits monitor the number of times of consecutive receptions for the PH1M0 register. |

29.2.31 Packet Header 2 Monitor Register 0 (PH2M0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|----|----|----|----|----|----|----|----------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | WC[15:8] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | WC[7:0] | | | | | | | | VC | | DT | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 8 | WC | All 0 | R | Word count data received |
| 7, 6 | VC | B'00 | R | Virtual channel data received |
| 5 to 0 | DT | H'00 | R | Data type data received |

PH2M0 is a register to monitor the received packet header (previous data of PH1M0).

29.2.32 Packet Header 2 Monitor Register 1 (PH2M1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PH_CNT | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | PH_CNT | All 0 | R | PH2M0 register consecutive reception count These bits monitor the number of times of consecutive receptions for the PH2M0 register. |

29.2.33 Packet Header 3 Monitor Register 0 (PH3M0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|----|----|----|----|----|----|----|----------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | WC[15:8] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | WC[7:0] | | | | | | | | VC | | DT | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 8 | WC | All 0 | R | Word count data received |
| 7, 6 | VC | B'00 | R | Virtual channel data received |
| 5 to 0 | DT | H'00 | R | Data type data received |

P M0 is a register to monitor the received packet header (previous data of PH2M0)

29.2.34 Packet Header 3 Monitor Register 1 (PH3M1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PH_CNT | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | PH_CNT | All 0 | R | PH3M0 register consecutive reception count These bits monitor the number of times of consecutive receptions for the PH3M0 register. |

29.2.35 Packet Header R Monitor Register 0 (PHRM0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|----|----|----|----|----|----|----|----------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | ECC | | | | | | | | WC[15:8] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | WC[7:0] | | | | | | | VC | | DT | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|-------------------------------|
| 31 to 24 | ECC | All 0 | R | ECC data received |
| 23 to 8 | WC | All 0 | R | Word count data received |
| 7, 6 | VC | B'00 | R | Virtual channel data received |
| 5 to 0 | DT | H'00 | R | Data type data received |

PHRM0 is a register to monitor the currently received packet header.

29.2.36 Packet Header R Monitor Register 1 (PHRM1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|----|----|----|----|----|----|----|----------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | ECC | | | | | | | | WC[15:8] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | WC[7:0] | | | | | | | | VC | | DT | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|-------------------------------|
| 31 to 24 | ECC | All 0 | R | ECC data received |
| 23 to 8 | WC | All 0 | R | Word count data received |
| 7, 6 | VC | B'00 | R | Virtual channel data received |
| 5 to 0 | DT | H'00 | R | Data type data received |

PHRM1 is a register to monitor the received packet header (previous data of PHRM0).

29.2.37 Packet Header R Monitor Register 2 (PHRM2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|----|----|----|----|----|----|----|----------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | ECC | | | | | | | | WC[15:8] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | WC[7:0] | | | | | | | | VC | | DT | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|-------------------------------|
| 31 to 24 | ECC | All 0 | R | ECC data received |
| 23 to 8 | WC | All 0 | R | Word count data received |
| 7, 6 | VC | B'00 | R | Virtual channel data received |
| 5 to 0 | DT | H'00 | R | Data type data received |

PHRM2 is a register to monitor the received packet header (previous data of PHRM1).

29.2.38 Packet Header C Monitor Register 0 (PHCM0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|----|----|----|----|----|----|----|----------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CAL_PARITY | | | | | | | | WC[15:8] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | WC[7:0] | | | | | | | | VC | | DT | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 24 | CAL_PARITY | All 0 | R | Parity calculated from the header |
| 23 to 8 | WC | All 0 | R | Received word count data after ECC correction |
| 7, 6 | VC | B'00 | R | Received virtual channel data after ECC correction |
| 5 to 0 | DT | H'00 | R | Received data type data after ECC correction |

PHCM0 is a register to monitor the currently received ECC-corrected packet header and calculated parity.

29.2.39 Packet Header C Monitor Register 1 (PHCM1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|----|----|----|----|----|----|----|----------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CAL_PARITY | | | | | | | | WC[15:8] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | WC[7:0] | | | | | | | | VC | | DT | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 24 | CAL_PARITY | All 0 | R | Parity calculated from the header |
| 23 to 8 | WC | All 0 | R | Received word count data after ECC correction |
| 7, 6 | VC | B'00 | R | Received virtual channel data after ECC correction |
| 5 to 0 | DT | H'00 | R | Received data type data after ECC correction |

PHCM1 is a register to monitor the currently received ECC-corrected packet header and calculated parity (previous data of PHCM0).

29.2.40 CRC Monitor Register 0 (CRCM0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CRC | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CAL_CRC | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | CRC | All 0 | R | Received CRC |
| 15 to 0 | CAL_CRC | All 0 | R | CRC result calculated from the received data |

CRCM0 is a register to monitor the CRC currently received and CRC calculated from the received data.

29.2.41 CRC Monitor Register 1 (CRCM1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CRC | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CAL_CRC | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | CRC | All 0 | R | Received CRC |
| 15 to 0 | CAL_CRC | All 0 | R | CRC result calculated from the received data |

CRCM1 is a register to monitor the CRC currently received and CRC calculated from the received data (previous data of CRCM0).

29.2.42 SOT Error Count Register (SERRCNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | ERRSOTHS_COUNT | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | ERRSOTHS_COUNT | H'00 | R | Synchronized SOT error count These bits monitor the number of synchronized SOT errors. The count is the sum of the errors on each lane. |

29.2.43 SOTSYNC Error Count Register (SSERRCNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|--------------|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | ERRSOTSYNCHS | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|---|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | ERRSOTSYNCHS | H'0 | R | Non-synchronizable SOT error count These bits monitor the number of non-synchronizable SOT errors. The count is the sum of the errors on each lane. After this error is input, no synchronization signal is generated. If the asynchronous FIFO for storing data from the PHY overflows here, the internal registers shift to the abnormal state. In this case, the state can only be restored by resetting the CSI2 control circuit using the software reset register (within the CSI2 module) or inputting a software reset from an external module after resetting the PHY and stopping the normal lane data from the PHY. |

29.2.44 ECC_CRCT Count Register (ECCCM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----------------|----|----|----|----|----|----|----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | ECC_CRCT_COUNT | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | ECC_CRCT_COUNT | H'00 | R | ECC 1-bit correction count These bits monitor the number of ECC 1-bit corrections. |

29.2.45 ECC_ERR Count Register (ECECM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | ECC_ERR_COUNT | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | ECC_ERR_COUNT | H'00 | R | ECC error (two bits or more) count These bits monitor the number of ECC errors (of two bits or more). |

29.2.46 CRC_ERR Count Register (CRCECM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | CRC_ERR_COUNT | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | CRC_ERR_COUNT | H'00 | R | CRC error count These bits monitor the number of CRC errors. |

29.2.47 Line Register (LCNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | LINE_CNT2 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | LINE_CNT1 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 16 | LINE_CNT2 | All 0 | R | Channel 1 reception line These bits monitor the line currently received on channel 1. |
| 15 to 0 | LINE_CNT1 | All 0 | R | Channel 0 reception line These bits monitor the line currently received on channel 0. |

29.2.48 Line Register 2 (LCNT2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | LINE_CNT4 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | LINE_CNT3 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 16 | LINE_CNT4 | All 0 | R | Channel 3 reception line These bits monitor the line currently received on channel 3. |
| 15 to 0 | LINE_CNT3 | All 0 | R | Channel 2 reception line These bits monitor the line currently received on channel 2. |

29.2.49 Line Monitor Register (LCNTM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MONI_LINECNT2 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MONI_LINECNT1 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 to 16 | MONI_LINECNT2 | All 0 | R | Channel 1 received line count These bits monitor the number of lines of the most recently received frame on channel 1. |
| 15 to 0 | MONI_LINECNT1 | All 0 | R | Channel 0 received line count These bits monitor the number of lines of the most recently received frame on channel 0. |

29.2.50 Line Monitor Register 2 (LCNTM2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|---------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MONI_LINECNT4 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MONI_LINECNT3 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 to 16 | MONI_LINECNT4 | All 0 | R | Channel 3 received line count These bits monitor the number of lines of the most recently received frame on channel 3. |
| 15 to 0 | MONI_LINECNT3 | All 0 | R | Channel 2 received line count These bits monitor the number of lines of the most recently received frame on channel 2. |

29.2.51 Frame Count Monitor Register (FCNTM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MONI_FCOUNTER2 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MONI_FCOUNTER1 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 16 | MONI_FCOUNTER2 | All 0 | R | Channel 1 received frame count These bits monitor the number of frames which have been received so far on channel 1. |
| 15 to 0 | MONI_FCOUNTER1 | All 0 | R | Channel 0 received frame count These bits monitor the number of frames which have been received so far on channel 0. |

29.2.52 Frame Count Monitor Register 2 (FCNTM2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MONI_FCOUNTER4 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MONI_FCOUNTER3 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 16 | MONI_FCOUNTER4 | All 0 | R | Channel 3 received frame count These bits monitor the number of frames which have been received so far on channel 3. |
| 15 to 0 | MONI_FCOUNTER3 | All 0 | R | Channel 2 received frame count These bits monitor the number of frames which have been received so far on channel 2. |

29.2.53 PHY Data IN Monitor Register (PHYDIM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|----|----|----|----|----|----|----|------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RXDATAHS_3 | | | | | | | | RXDATAHS_2 | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RXDATAHS_1 | | | | | | | | RXDATAHS_0 | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 24 | RXDATAHS_ 3 | H'00 | R | Byte data received on data lane 3 Note: When 2-lane PHY is connected, these bits are fixed to 0. |
| 23 to 16 | RXDATAHS_ 2 | H'00 | R | Byte data received on data lane 2 Note: When 2-lane PHY is connected, these bits are fixed to 0. |
| 15 to 8 | RXDATAHS_ 1 | H'00 | R | Byte data received on data lane1 |
| 7 to 0 | RXDATAHS_ 0 | H'00 | R | Byte data received on data lane 0 |

PHYDIM is a register to monitor byte data input from the PHY.

29.2.54 PHY Input Monitor Register (PHYIM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----------------|----|----|----|----------------|----|----|----|----------------|-------------|-------------|-------------|----------------|--------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RXCLK_CNT | — | — | — | — | — | — | — | RXVALIDHS_3 | RXVALIDHS_2 | RXVALIDHS_1 | RXVALIDHS_0 | RXACTIVEHS_3 | RXACTIVEHS_2 | RXACTIVEHS_1 | RXACTIVEHS_0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RXSYNCHS_3_CNT | | | | RXSYNCHS_2_CNT | | | | RXSYNCHS_1_CNT | | | | RXSYNCHS_0_CNT | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 | RXCLK_CNT | B'0 | R | RXCLK counter 0 This bit monitors signals toggled at the rising edge of RXCLK. |
| 30 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 | RXVALIDHS_3 | B'0 | R | Data lane 3 HS valid signal Note: When 2-lane PHY is connected, this bit is fixed to 0. This bit monitors the valid signal on lane 3 input from the PHY. |
| 22 | RXVALIDHS_2 | B'0 | R | Data lane 2 HS valid signal Note: When 2-lane PHY is connected, this bit is fixed to 0. This bit monitors the valid signal on lane 2 input from the PHY. |
| 21 | RXVALIDHS_1 | B'0 | R | Data lane 1 HS valid signal This bit monitors the valid signal on lane 1 input from the PHY. |
| 20 | RXVALIDHS_0 | B'0 | R | Data lane 0 HS valid signal This bit monitors the valid signal on lane 0 input from the PHY. |
| 19 | RXACTIVEHS_S_3 | B'0 | R | Data lane 3 HS active signal Note: When 2-lane PHY is connected, this bit is fixed to 0. This bit monitors the active signal on lane 3 input from the PHY. |
| 18 | RXACTIVEHS_S_2 | B'0 | R | Data lane 2 HS active signal Note: When 2-lane PHY is connected, this bit is fixed to 0. This bit monitors the active signal on lane 2 input from the PHY. |
| 17 | RXACTIVEHS_S_1 | B'0 | R | Data lane 1 HS active signal This bit monitors the active signal on lane 1 input from the PHY. |
| 16 | RXACTIVEHS_S_0 | B'0 | R | Data lane 0 HS active signal This bit monitors the active signal on lane 0 input from the PHY. |
| 15 to 12 | RXSYNCHS_3_CNT | H'0 | R | Data lane 3 HS synchronization signal count Note: When 2-lane PHY is connected, these bits are fixed to 0. This bit monitors the count of RSYNCHS_3 input from the PHY. |
| 11 to 8 | RXSYNCHS_2_CNT | H'0 | R | Data lane 2 HS synchronization signal count Note: When 2-lane PHY is connected, these bits are fixed to 0. This bit monitors the count of RSYNCHS_2 input from the PHY. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------|---------------|-----|---|
| 7 to 4 | RXSYNCHS_1_CNT | H'0 | R | Data lane 1 HS synchronization signal count This bit monitors the count of RSYNCHS_1 input from the PHY. |
| 3 to 0 | RXSYNCHS_0_CNT | H'0 | R | Data lane 0 HS synchronization signal count This bit monitors the count of RSYNCHS_0 input from the PHY. |

29.2.55 VIN Data Monitor Register (VINDM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CSIR_DAT[31:16] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CSIR_DAT[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 31 to 0 | CSIR_DAT [31:0] | All 0 | R | Data output from CSI2 These bits monitor the CSIR_DAT signal output from the CSI2. |

29.2.56 VIN Signal Monitor Register 1 (VINSM1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------------|----|----|----|--------------|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CSIR_VD2_CNT | | | | CSIR_HD2_CNT | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CSIR_VD1_CNT | | | | CSIR_HD1_CNT | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 28 | CSIR_VD2_CNT | H'0 | R | Count of VD2 output from CSI2 These bits count and monitor the vertical synchronization signals for channel 1 output from CSI2. |
| 27 to 16 | CSIR_HD2_CNT | H'000 | R | Count of HD2 output from CSI2 These bits count and monitor the HD signals for channel 1 output from the CSI2. |
| 15 to 12 | CSIR_VD1_CNT | H'0 | R | Count of VD1 output from CSI2 These bits count and monitor the vertical synchronization signals for channel 0 output from CSI2. |
| 11 to 0 | CSIR_HD1_CNT | H'000 | R | Count of HD1 output from CSI2 These bits count and monitor the HD signals for channel 0 output from CSI2. |

29.2.57 VIN Signal Monitor Register 2 (VINSM2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|--------------|----|----|----|--------------|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CSIR_VD4_CNT | | | | CSIR_HD4_CNT | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CSIR_VD3_CNT | | | | CSIR_HD3_CNT | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 28 | CSIR_VD4_CNT | H'0 | R | Count of VD4 output from CSI2 These bits count and monitor the vertical synchronization signals for channel 3 output from CSI2. |
| 27 to 16 | CSIR_HD4_CNT | H'000 | R | Count of HD4 output from CSI2 These bits count and monitor the HD signals for channel 3 output from CSI2. |
| 15 to 12 | CSIR_VD3_CNT | H'0 | R | Count of VD3 output from CSI2 These bits count and monitor the vertical synchronization signals for channel 2 output from CSI2. |
| 11 to 0 | CSIR_HD3_CNT | H'000 | R | Count of HD3 output from CSI2 These bits count and monitor the HD signals for channel 2 output from CSI2. |

29.2.58 VIN Signal Monitor Register 3 (VINSM3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|-----------|----------|----------|----|----|----|----------|----|----|----|----|----|----|---------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CSIR_ERRE | CSIR_ERRC | CSIR_TAG | CSIR_FLD | | | | CSIR_PEB | | | | — | — | — | CSIR_PE | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | CSIR_ERRE | B'0 | R | ECC error output from CSI2 This bit monitors the ECC error signal output from CSI2. |
| 14 | CSIR_ERRC | B'0 | R | CRC error output from CSI2 This bit monitors the CRC error signal output from CSI2. |
| 13, 12 | CSIR_TAG | B'00 | R | TAG output from CSI2 These bits monitor the channel information output from CSI2. |
| 11 to 8 | CSIR_FLD | H'0 | R | FLD output from CSI2 These bits monitor the field information output from CSI2. |
| 7 to 4 | CSIR_PEB | H'0 | R | PEB output from CSI2 These bits monitor the byte enable signal output from CSI2. |
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | CSIR_PE | B'0 | R | PE output from CSI2 This bit monitors the packet enable signal output from CSI2. |

29.2.59 PHY Output Monitor Register (PHYOM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|---------------|-----------------|-----------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | ENABL ECLK | ENABL E_3/—* | ENABL E_2/—* | ENABL E_1 | ENABL E_0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

Note: * For reserved bit (Bit Name: —). Refer to the following table.

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | ENABLECLK | B'0 | R | Lane enable signal to the clock lane |
| 3 | ENABLE_3 | B'0 | R | Lane enable signal to data lane 3 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| | — | B'0 | R | Reserved [RZ/G2E] This bit is always read as 0. The write value should always be 0. |
| 2 | ENABLE_2 | B'0 | R | Lane enable signal to data lane 2 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| | — | B'0 | R | Reserved [RZ/G2E] This bit is always read as 0. The write value should always be 0. |
| 1 | ENABLE_1 | B'0 | R | Lane enable signal to data lane 1 |
| 0 | ENABLE_0 | B'0 | R | Lane enable signal to data lane 0 |

PHYOM is a register to monitor the lane enable signal output from the CSI2 to the PHY.

29.2.60 Packet Header Monitor Registers 1 to 8 (PHM1 to PHM8)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------|----|----|----|----|----|----|----|----------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | ECC | | | | | | | | WC[15:8] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | WC[7:0] | | | | | | | VC | | DT | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|-------------------------------|
| 31 to 24 | ECC | All 0 | R | ECC data received |
| 23 to 8 | WC | All 0 | R | Word count data received |
| 7, 6 | VC | B'00 | R | Virtual channel data received |
| 5 to 0 | DT | H'00 | R | Data type data received |

PHM1 to PHM8 are registers to monitor the received packet header.

Data is stored in PHM1 first, and the data is shifted to PHM2 when the received packet header changes. After data is stored in PHM8, data is stored in PHM1 again.

29.2.61 Packet Period Monitor Registers 1 to 8 (PPM1 to PPM8)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | PACKET_PERIOD_MON[31:16] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PACKET_PERIOD_MON[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------------------|---------------|-----|--|
| 31 to 0 | PACKET_PE RIOD_MON [31:0] | All 0 | R | <p>Packet period</p> <p>A quarter number of cycles during which the packet header is not modified are read with respect to PHM1 to PHM8.</p> <p>The PPM contents correspond to the PHM contents. Specifically, a quarter number of cycles are stored in PPM1 between when data is stored in PHM1 and when data is shifted to PHM2.</p> |

29.2.62 CSI0CLK Frequency Configuration Preset Register (CSI0CLK)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-----------------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | CSI0CLKFREQRANGE[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------------|---------------|-----|--|
| 31 to 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 to 16 | CSI0CLKFREQRANGE[5:0] | H'00 | R/W | CSI0 ϕ frequency configuration It should be set to H'20. |
| 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

29.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

29.3.1 Transfer Rate

This module supports a transfer rate of from 80 Mbps to 1.5 Gbps*¹.

1.5 Gbps transmission is shown in Figure 29.6.

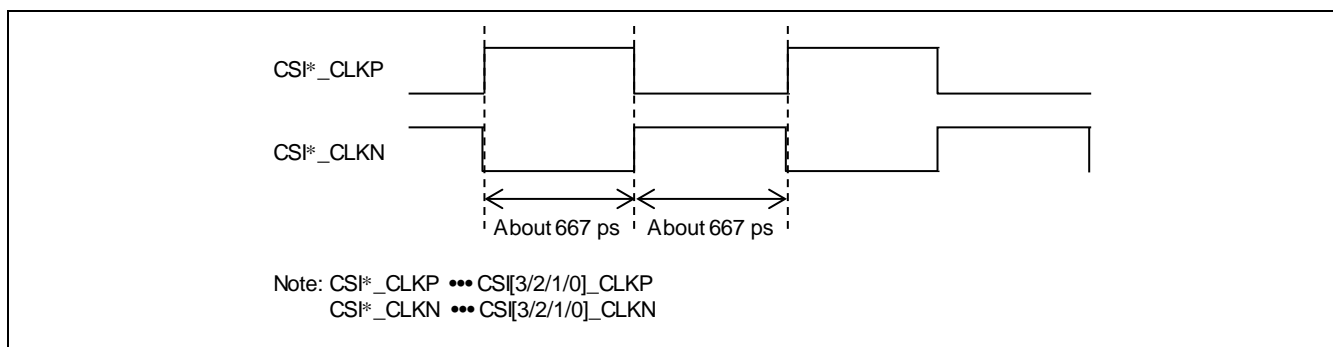


Figure 29.6 CSI2 Transfer Rate

Notes: 1. Up to 1.1Gbps transfer rate of MIPI CSI-2 in RZ/G2E.

29.3.2 ECC 1-Bit Error Correction and 2-Bit or More Error Detection of a Packet Header

An interrupt can be generated upon ECC 1-bit error correction and 2-bit or more error detection.

| Interrupt Source | Register Bit Name | Bit |
|----------------------------|-------------------|------|
| ECC 2-bit or more error | ECC_ERR | [11] |
| ECC 1-bit error correction | ECC_CRCT_ERR | [10] |

The ECC_EN bit in the checksum control register (CHKSUM) can enable or disable the function.

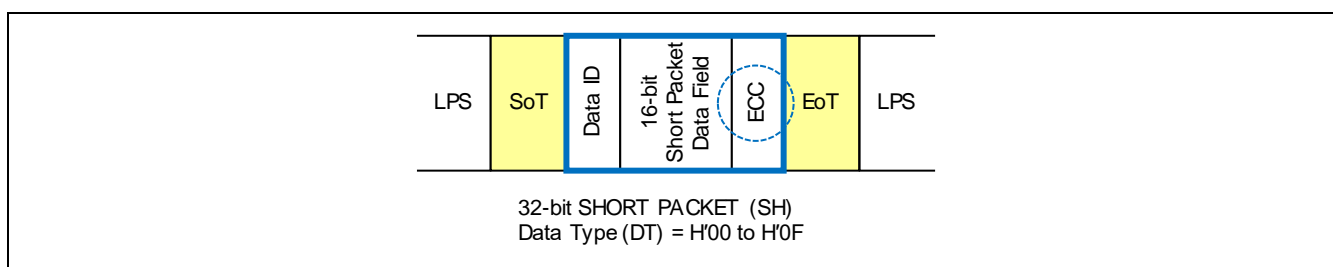


Figure 29.7 MIPI CSI-2 Short Packet

29.3.3 CRC Error Detection of a Payload Data Part

An interrupt can be generated upon CRC error detection.

| Interrupt Source | Register Bit Name | Bit |
|------------------|-------------------|------|
| CRC error | CRC_ERR | [15] |

The CRC_EN bit in the checksum control register (CHKSUM) can enable or disable the function.

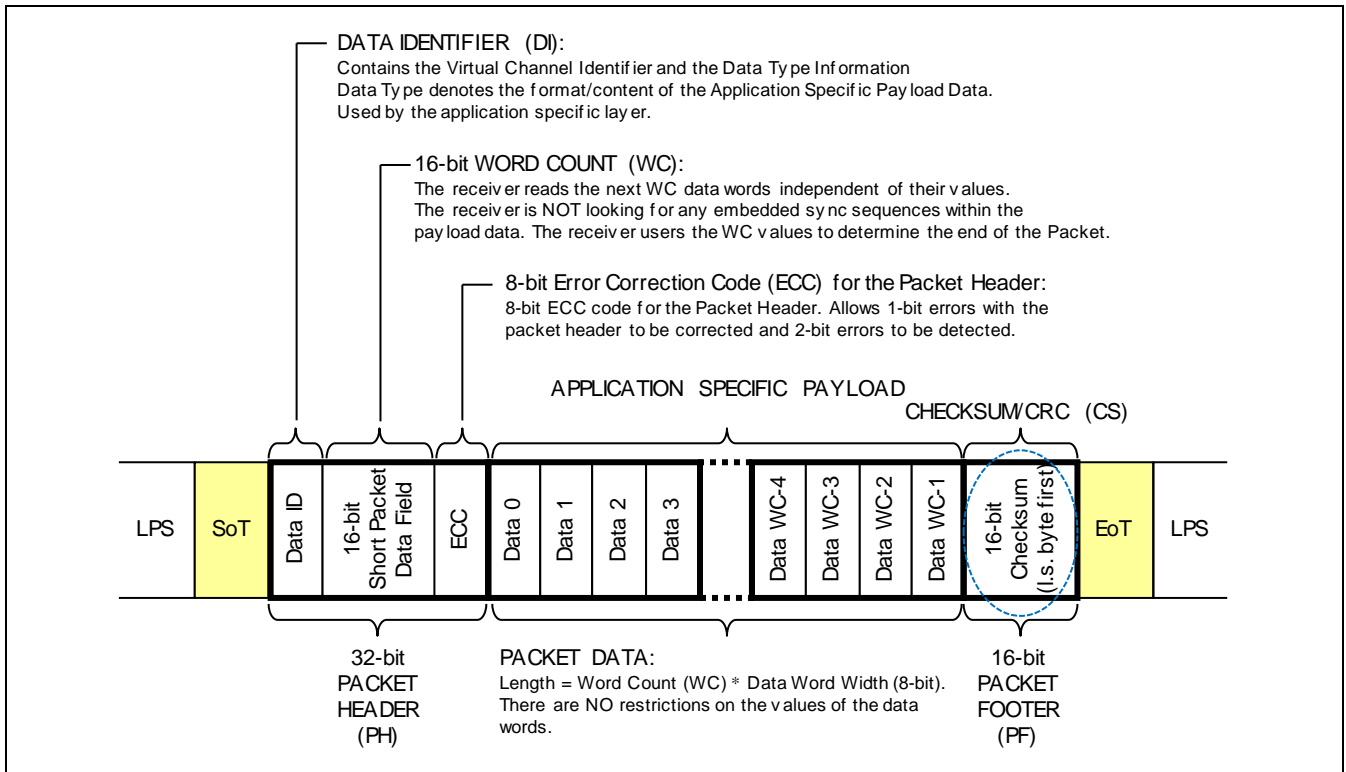


Figure 29.1 MIPI CSI-2 Long Packet

29.3.4 Generation of VD (vertical sync), HD (horizontal sync), and FLD (field) Signals

(1) VD

When a frame start short packet is received, VD is driven high. When a frame end short packet is received, VD is driven low. (This applies to the channel specified with the VC value in the packet header.)

(2) HD

When the sync signal of a long packet (SOT) is received, HD is driven high. When the number of the received long packets reaches the WC value in the packet header, HD is driven low. The frame start short packet of the reception channel should be received in advance. When Embedded is received, this applies to EBD_HD.

(3) FLD

When a frame start short packet is received, if the field value of the short packet data satisfies the condition specified with the field detection control register (FLD), FLD is driven high. When a frame end short packet is received, FLD is driven low. (This applies to the channel specified with the VC value in the packet header.)

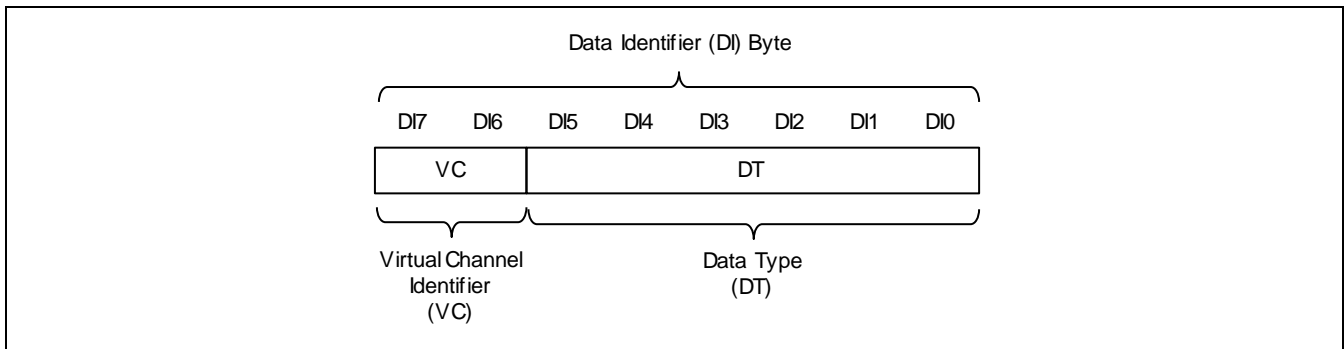


Figure 29.9 MIPI CSI-2 DI

Table 29.4 MIPI CSI-2 Short Packet Data Type Codes

| data type | Description |
|--------------|----------------------------|
| H'00 | Frame Start Code |
| H'01 | Frame End Code |
| H'02 | Line Start Code (Optional) |
| H'03 | Line End Code (Optional) |
| H'04 to H'07 | Reserved |

29.3.5 Four-Channel Output

As shown in Figure 29.10 and Figure 29.11 interleaved transmission of up to four virtual channels is possible with the RZ/G Gen2 CSI2*1. Therefore, the VD and FLD signals for each channel are output to the following VIN module. (There are four VD and four FLD signals.) The TAG signal, which indicates the reception channels, is also output to the VIN module. The received virtual channels and the channels to be output to the VIN module can be changed by setting the SEL_VC*(* = null, 2 to 4) bits in the relevant registers (channel data type select register VCDT and channel data type select register 2 VCDT2).

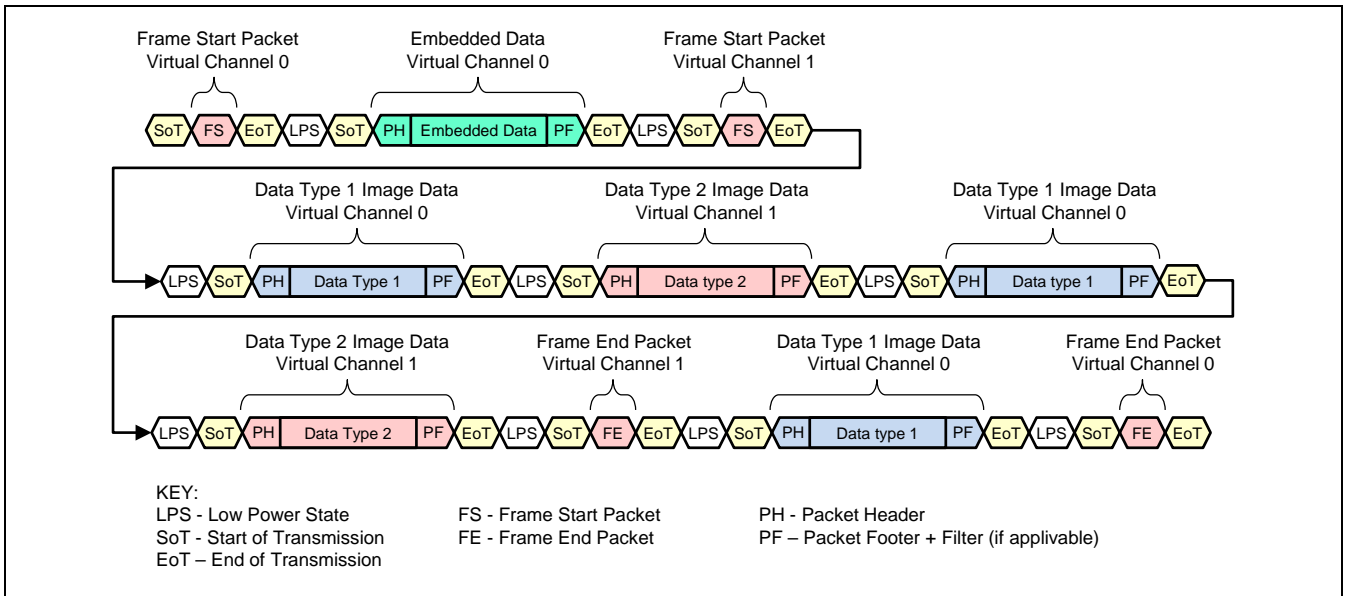


Figure 29.20 MIPI CSI-2 Interleaved Transmission 1

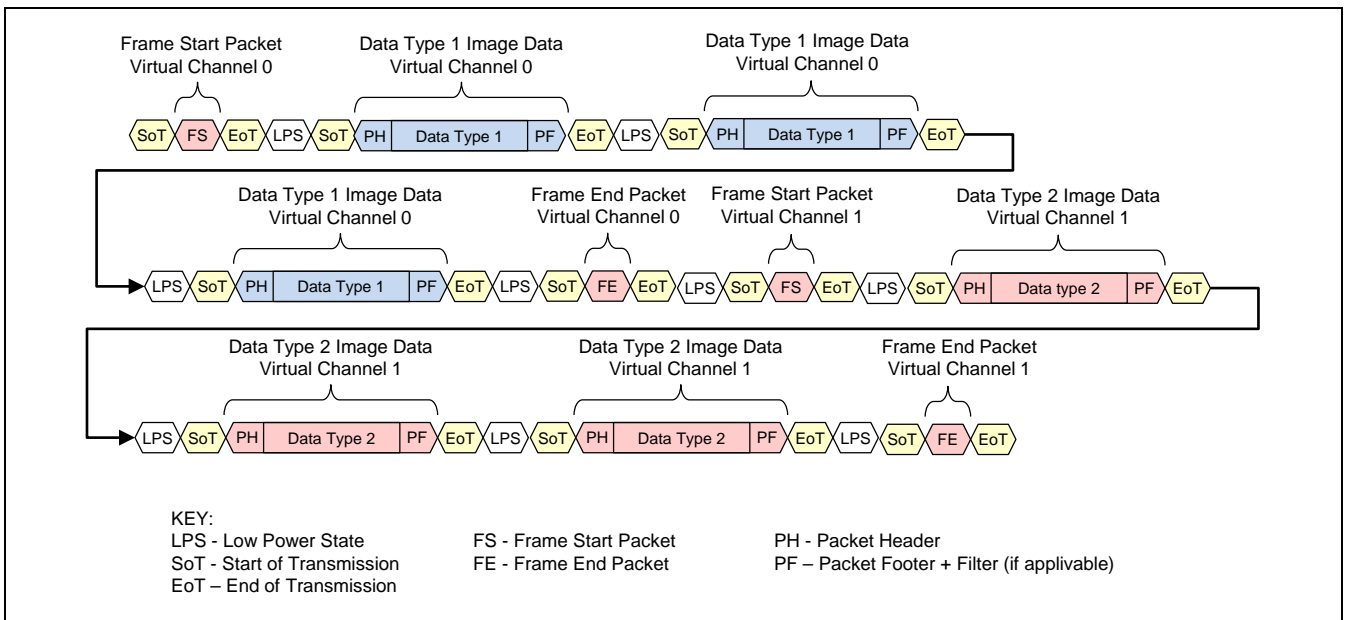


Figure 29.11 MIPI CSI-2 Interleaved Transmission 2

Note: *1. Interleaved transmission of up to two virtual channels (virtual channel 0 and 1) is possible with the RZ/G2E CSI2.

29.3.6 Interrupts

Table 29.5 Interrupt Registers

| Function | Register Name |
|---------------------------------|---------------|
| Monitors interrupt state. | INTSTATE |
| Monitors interrupt error state. | INTERRSTATE |
| Enables interrupts. | INTEN |
| Masks interrupt sources. | INTCLOSE |

As shown in Figure 29.12, the FIFO_OF_[3/2/1/0] (overflow of synchronous FIFO), AFIFO_OF_[3/2/1/0] (overflow of asynchronous FIFO), ERRSOTHS_[3/2/1/0] (synchronized SOT (start of transfer) error) and ERRSOTSYNCHS_[3/2/1/0] (non-synchronizable SOT (start of transfer) error) have the corresponding interrupt error status monitoring register bits for each channel. To clear the status, write 1 to the corresponding INTERRSTATE register bit.

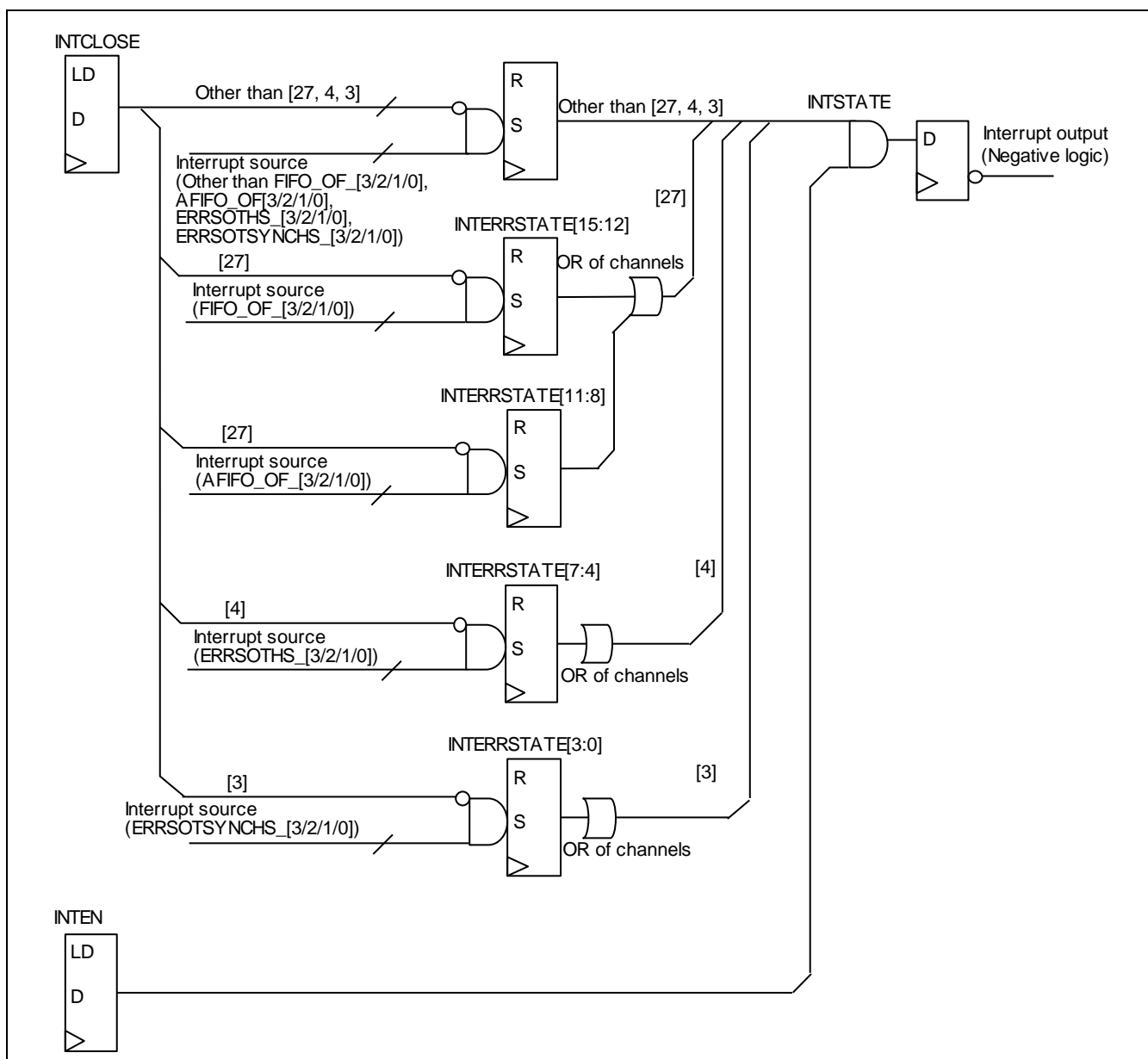


Figure 29.12 Interrupt Structure

Table 29.6 Interrupt Bit Assignments

| Interrupt Source | Register Bit Name | Bit |
|---|-------------------|------|
| Reception of embedded data from output channel 0 | EBD_CH0 | [29] |
| Long packet payload data count less than WC value | LESS_THAN_WC | [28] |
| Overflow of asynchronous FIFO for storing HS data | AFIFO_OF | [27] |
| Start of frame on output channel 3 (rising edge of CSIR_VD[3] signal) | VD4_START | [26] |
| End of frame on output channel 3 (falling edge of CSIR_VD[3] signal) | VD4_END | [25] |
| Start of frame on output channel 2 (rising edge of CSIR_VD[2] signal) | VD3_START | [24] |
| End of frame on output channel 2 (falling edge of CSIR_VD[2] signal) | VD3_END | [23] |
| Start of frame on output channel 1 (rising edge of CSIR_VD[1] signal) | VD2_START | [22] |
| End of frame on output channel 1 (falling edge of CSIR_VD[1] signal) | VD2_END | [21] |
| Start of frame on output channel 0 (rising edge of CSIR_VD[0] signal) | VD_START | [20] |
| End of frame on output channel 0 (falling edge of CSIR_VD[0] signal) | VD_END | [19] |
| Reception of a short packet (reception of short packets such as Frame Start and Frame End) | SHP | [18] |
| Reception of a frame packet (reception of short packets of Frame Start and Frame End) | FSFE | [17] |
| Reception of a long packet | LNP | [16] |
| CRC error | CRC_ERR | [15] |
| Reception of WC = 0 | HD_WC_ZERO | [14] |
| Reception of an illegal frame end packet (Frame End is received without receiving Frame Start.) | FRM_SEQ_ERR1 | [13] |
| Reception of an illegal frame start packet (After reception of Frame Start, Frame Start is received again without receiving Frame End) | FRM_SEQ_ERR0 | [12] |
| ECC 2-bit or more error | ECC_ERR | [11] |
| ECC 1-bit correction | ECC_CRCT_ERR | [10] |
| — | — | [9] |
| — | — | [8] |
| Start of ultra-low power state | ULPS_START | [7] |
| End of ultra-low power state | ULPS_END | [6] |
| — | — | [5] |
| Synchronized SOT (start of transfer) error during HS reception | ERRSOTHS | [4] |
| Non-synchronizable SOT (start of transfer) error during HS reception | ERRSOTSYNCHS | [3] |
| Escape mode entry error | ERRESC | [2] |
| — | — | [1] |
| PHY control error | ERRCONTROL | [0] |

Table 29.7 Bit Assignments in Error Status Monitor Register

| Interrupt Source | Register Bit Name | Bit |
|--|-------------------|------|
| Overflow of synchronous FIFO for storing HS data on lane 3*1 | FIFO_OF_3 | [15] |
| Overflow of synchronous FIFO for storing HS data on lane 2*1 | FIFO_OF_2 | [14] |
| Overflow of synchronous FIFO for storing HS data on lane 1*1 | FIFO_OF_1 | [13] |
| Overflow of synchronous FIFO for storing HS data on lane 0*1 | FIFO_OF_0 | [12] |
| Overflow of asynchronous FIFO for storing HS data on lane 3 | AFIFO_OF_3 | [11] |
| Overflow of asynchronous FIFO for storing HS data on lane 2 | AFIFO_OF_2 | [10] |
| Overflow of asynchronous FIFO for storing HS data on lane 1 | AFIFO_OF_1 | [9] |
| Overflow of asynchronous FIFO for storing HS data on lane 0 | AFIFO_OF_0 | [8] |
| Synchronized SOT (start of transfer) error during HS reception on lane 3 | ERRSOTHS_3 | [7] |
| Synchronized SOT (start of transfer) error during HS reception on lane 2 | ERRSOTHS_2 | [6] |
| Synchronized SOT (start of transfer) error during HS reception on lane 1 | ERRSOTHS_1 | [5] |
| Synchronized SOT (start of transfer) error during HS reception on lane 0 | ERRSOTHS_0 | [4] |
| Non-synchronizable SOT (start of transfer) error during HS reception on lane 3 | ERRSOTSYNCHS_3 | [3] |
| Non-synchronizable SOT (start of transfer) error during HS reception on lane 2 | ERRSOTSYNCHS_2 | [2] |
| Non-synchronizable SOT (start of transfer) error during HS reception on lane 1 | ERRSOTSYNCHS_1 | [1] |
| Non-synchronizable SOT (start of transfer) error during HS reception on lane 0 | ERRSOTSYNCHS_0 | [0] |

Note: *1. ,RZ/G2H, RZ/G2N, RZ/G2E only

Table 29.8 MIPI CSI-2 PHY/ Annex A.1 Signal Description

| Error Signals | Description |
|----------------------|--|
| ERRSOTHS | Start-of-Transmission (SoT) Error. If the High-Speed Sot leader sequence is corrupted, but in such a way that proper synchronization can still be achieved, this active high signal is asserted for one cycle of RXBYTECLKHS. This is considered to be a "soft error" in the leader sequence and confidence in the payload data is reduced. |
| ERRSOTSYNCHS | Start-of-Transmission Synchronization Error. If the High-Speed Sot leader sequence is corrupted in a way that proper synchronization cannot be expected, this active high signal is asserted for one cycle of RXBYTECLKHS. |
| ERRESC | Escape Entry Error. If an unrecognized escape entry command is received, this active high signal is asserted and remains asserted until the next change in line state. |
| ERRCONTROL | Control Error. This active high signal is asserted when an incorrect line state sequence is detected. For example, if a turn-around request or escape mode request is immediately followed by a Stop state instead of the required Bridge state, this signal is asserted and remains asserted until the next change in line state. |

29.3.7 Lane Swapping

The LINK can modify lane assignment for the data received from the PHY by setting the relevant register (see section 29.2.19). This capability is provided for flexible pin assignment to compensate for the MIPI CSI-2 has no connector specification.

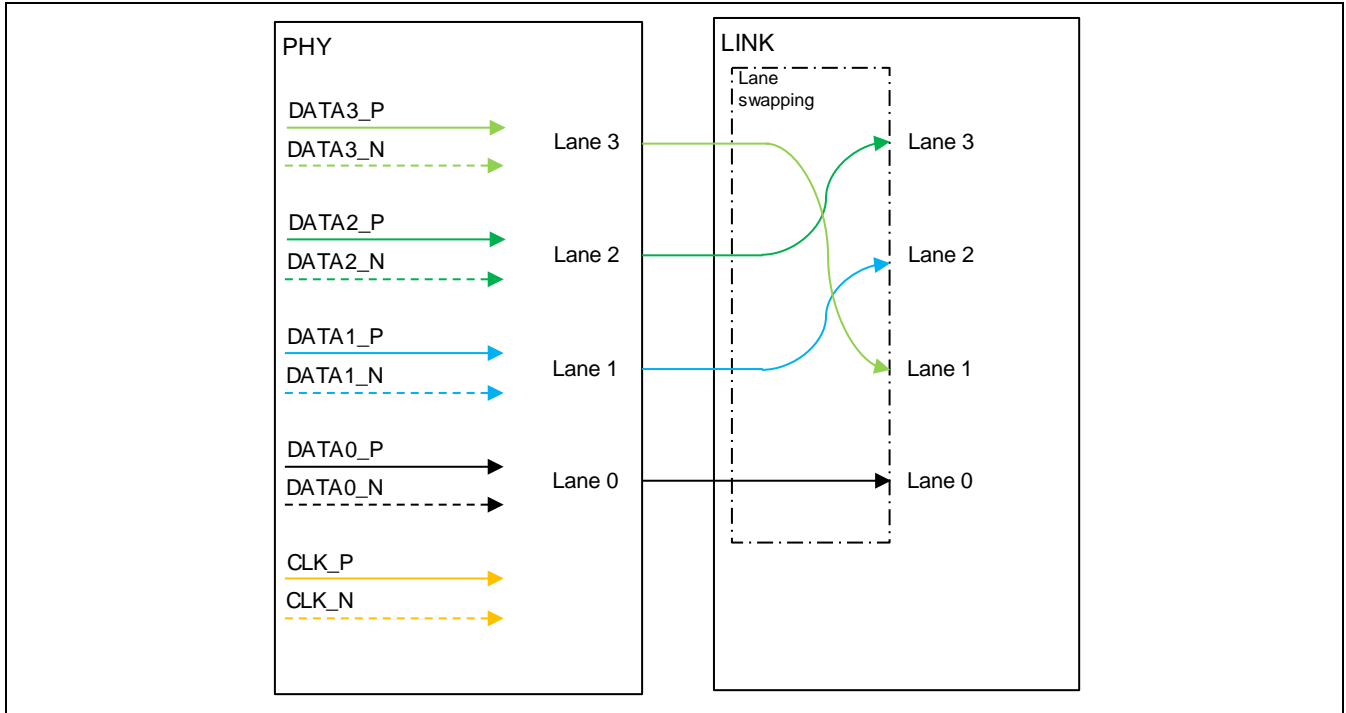


Figure 29.13 Example of Lane Swapping

29.3.8 Initial Setting of PHY

It is possible to run VIN initial sequence either before or after CSI2 initial sequence (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E).

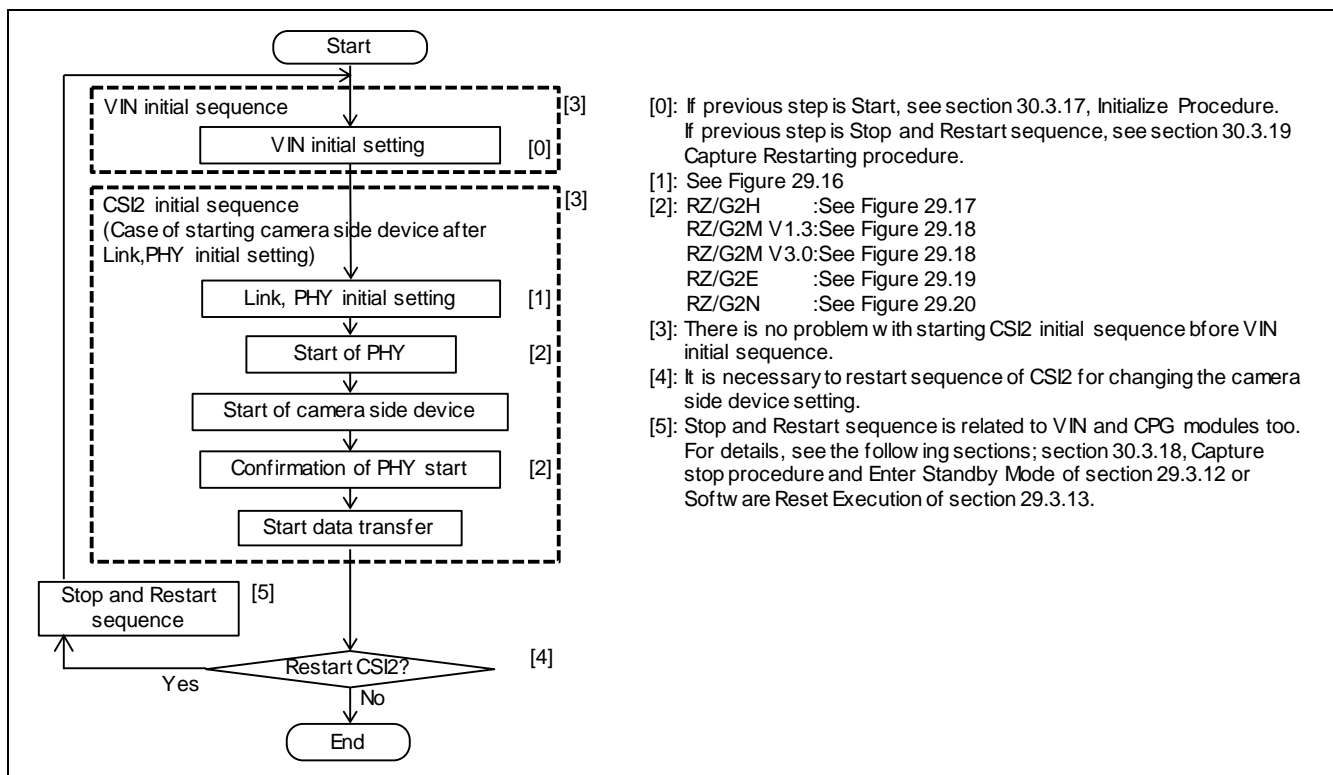


Figure 29.14 Example of Video Module Initial Setting Procedure 1

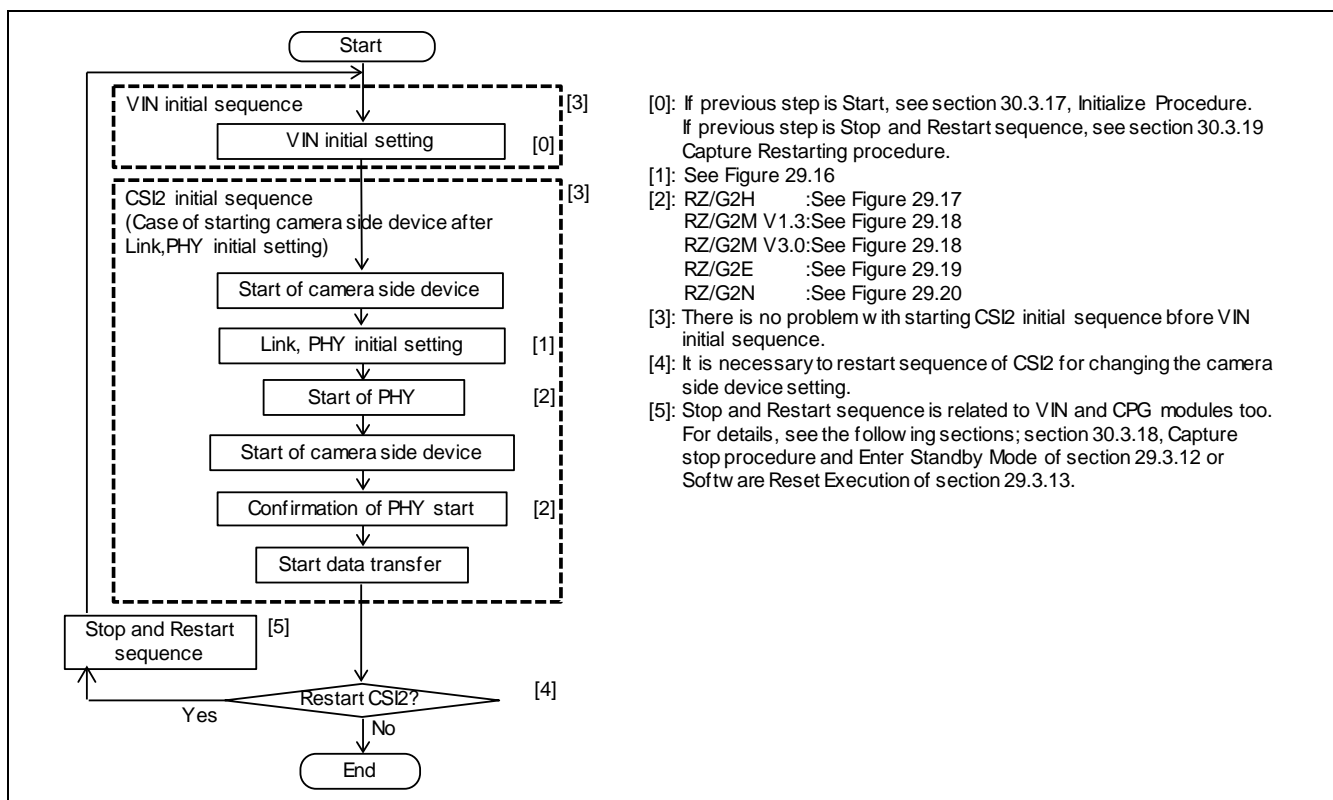


Figure 29.15 Example of Video Module Initial Setting Procedure 2

To initially set the PHY IP, the TESTCLR bit in the PHY test interface clear register must be cleared to 0.

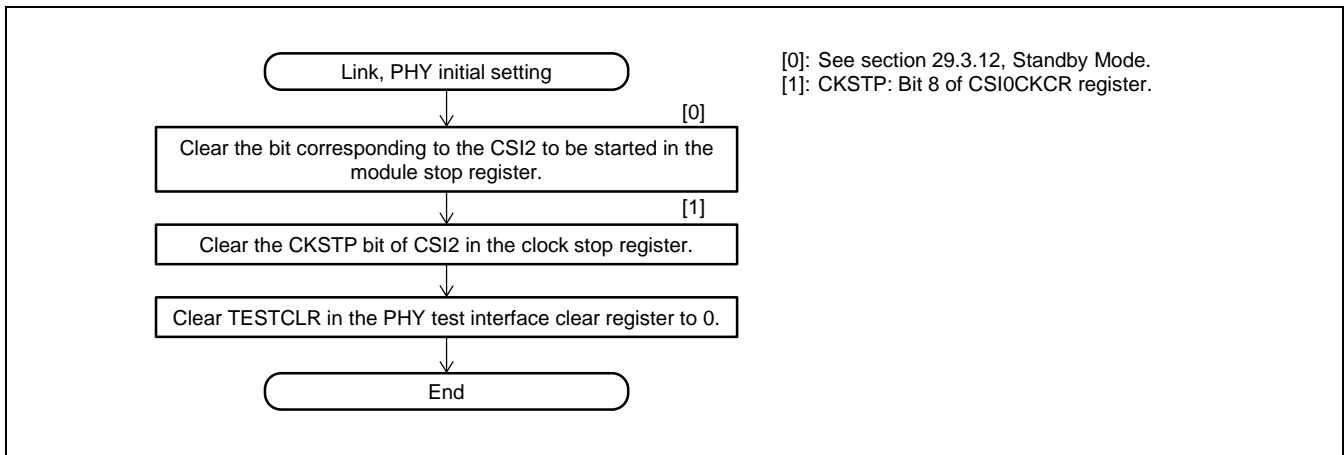


Figure 29.16 Example of PHY Initial Setting Procedure

29.3.9 PHY Control and Monitoring through Register Setting

The PHY IP can be controlled by setting the relevant LINK registers. For register functions, see section 29.2.3, 29.2.8, 29.2.18, 29.2.20*¹, 29.2.23*³ and 29.2.62*².

The PHY IP status can be monitored through the relevant LINK registers. For register functions, see section 29.2.25 and 29.2.26.

- Notes:
1. Except for RZ/G2M V1.3, RZ/G2M V3.0.
 2. Except for RZ/G2M V1.3, RZ/G2M V3.0, and RZ/G2E.
 3. Except for RZ/G2E.

(1) Case 1 Stable data transfer starting procedure

1. Start the CSI-2 module of the camera without starting data transfer.
2. Start the CSI2 PHY module of the RZ/G Gen2.
3. Start data transfer from the camera.

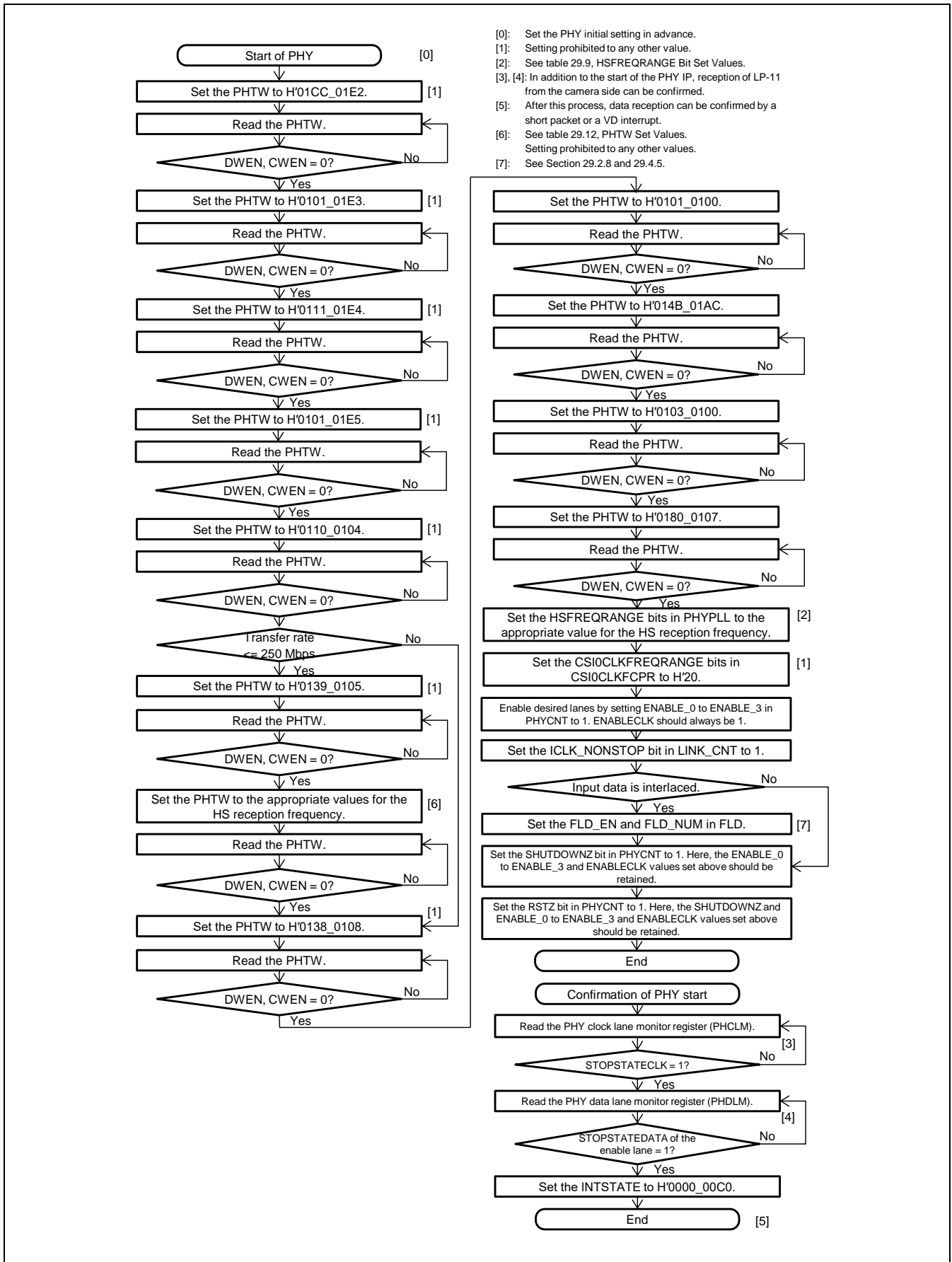


Figure 29.17 Example of PHY Starting Procedure in RZ/G2H (Stable)

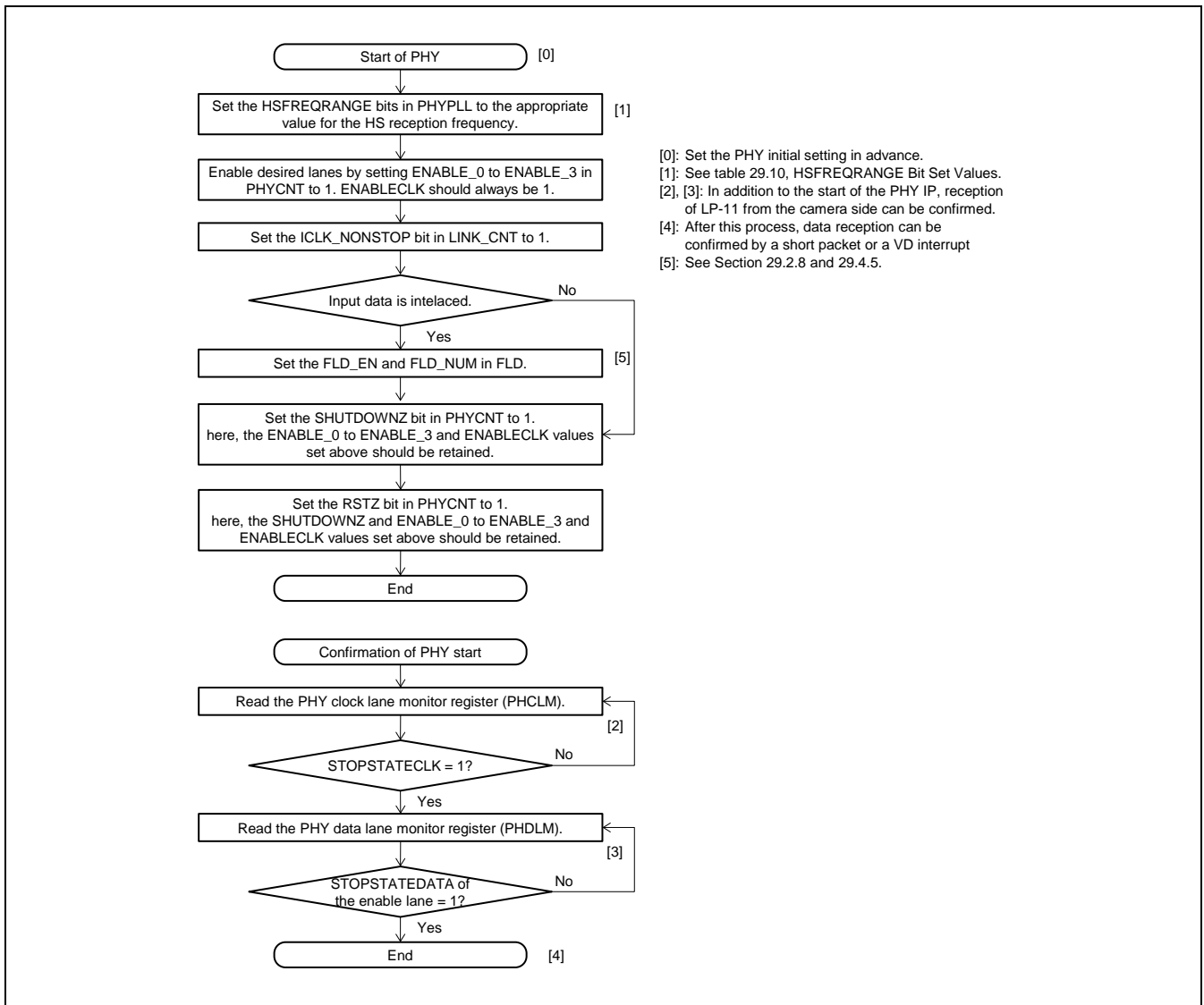


Figure 29.18 Example of PHY Starting Procedure RZ/G2M V1.3, RZ/G2M V3.0 (Stable)

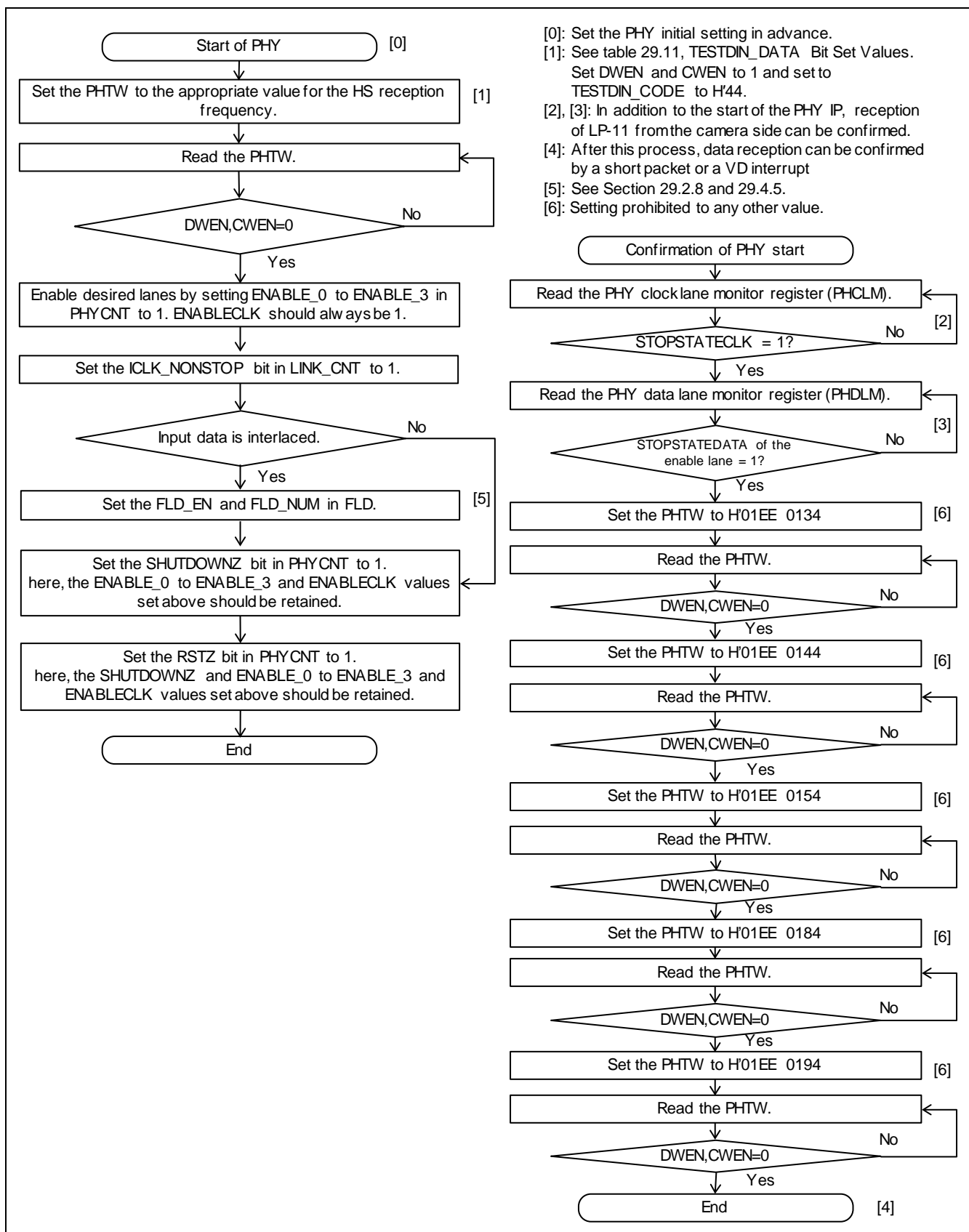


Figure 29.19 Example of PHY Starting Procedure in RZ/G2E (Stable)

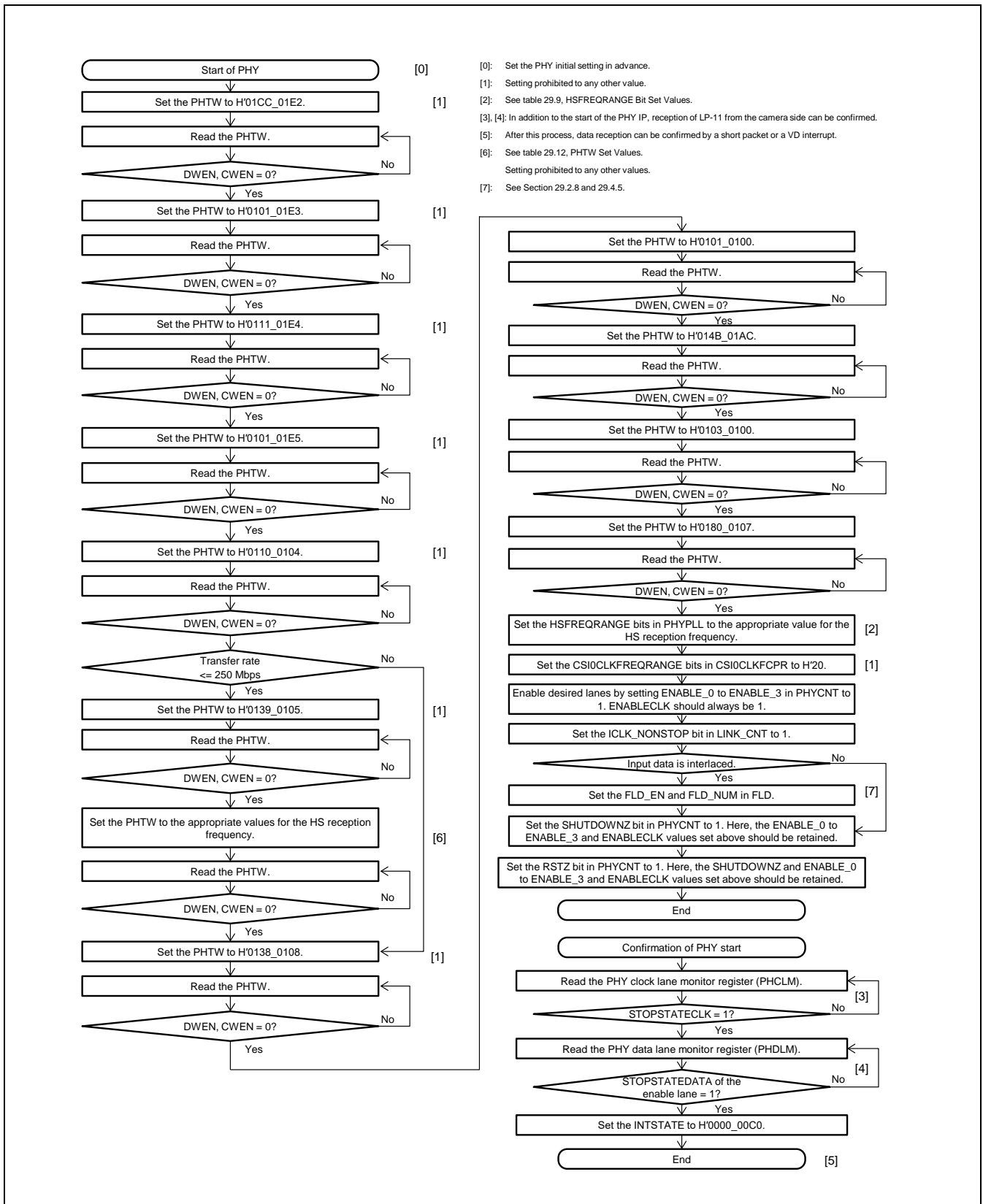


Figure 29.20 Example of PHY Starting Procedure in RZ/G2N (Stable)

It is necessary to input LP-11 to assert STOPSTATECLK and STOPSTATEDATA.
 An example is shown Figure 29.21 and 29.21.2.

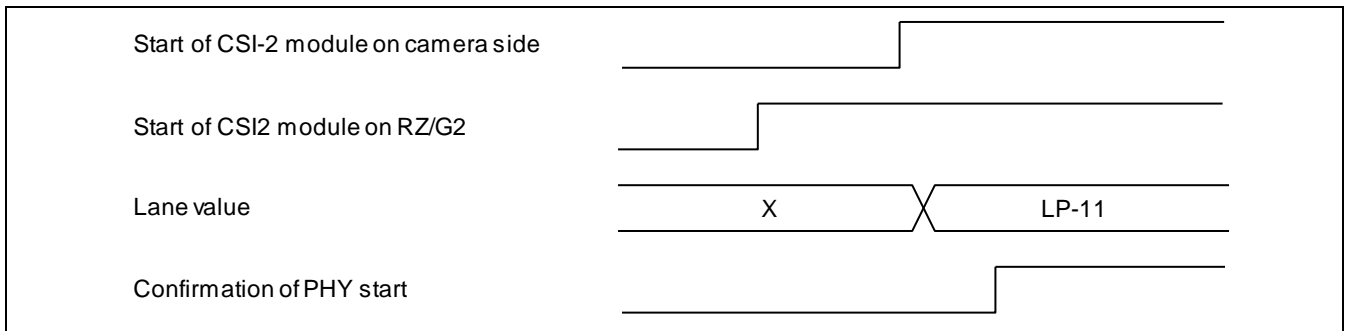


Figure 29.21 PHY Starting Order 1 (Stable)

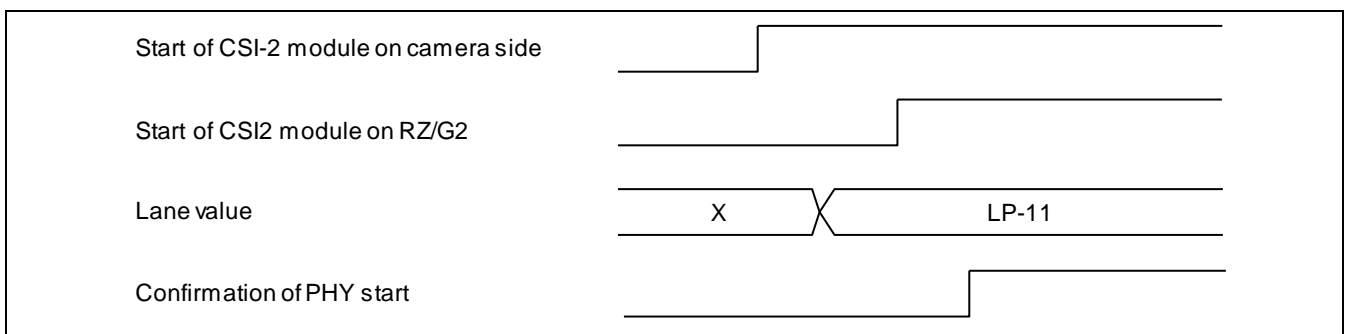


Figure 29.22 PHY Starting Order 2 (Stable)

Table 29.9 HSFREQRANGE Bit Set Values (RZ/G2H, RZ/G2N)

| Range (Mbps) | Default Bit Rate (Mbps) | HSFREQRANGE [6:0] | Range (Mbps) | Default Bit Rate (Mbps) | HSFREQRANGE [6:0] |
|-------------------|-------------------------|-------------------|---------------------|-------------------------|-------------------|
| 80 - 97.125 | 80 | B'000_0000 | 463.125 - 538.125 | 500 | B'010_0110 |
| 80 - 107.625 | 90 | B'001_0000 | 498.75 - 590.625 | 550 | B'011_0111 |
| 83.125 - 118.125 | 100 | B'010_0000 | 558.125 - 643.125 | 600 | B'000_0111 |
| 92.625 - 128.625 | 110 | B'011_0000 | 605.625 - 695.625 | 650 | B'001_1000 |
| 102.12 - 139.125 | 120 | B'000_0001 | 653.125 - 748.125 | 700 | B'010_1000 |
| 111.62 - 149.625 | 130 | B'001_0001 | 700.625 - 800.625 | 750 | B'011_1001 |
| 121.12 - 160.125 | 140 | B'010_0001 | 748.125 - 853.125 | 800 | B'000_1001 |
| 130.625 - 170.625 | 150 | B'011_0001 | 795.625 - 905.625 | 850 | B'001_1001 |
| 140.125 - 181.125 | 160 | B'000_0010 | 843.125 - 958.125 | 900 | B'010_1001 |
| 149.625 - 191.625 | 170 | B'001_0010 | 890.625 - 1010.625 | 950 | B'011_1010 |
| 159.125 - 202.125 | 180 | B'010_0010 | 938.125 - 1063.125 | 1000 | B'000_1010 |
| 168.625 - 212.625 | 190 | B'011_0010 | 985.625 - 1115.625 | 1050 | B'001_1010 |
| 182.875 - 228.375 | 205 | B'000_0011 | 1033.125 - 1168.125 | 1100 | B'010_1010 |
| 197.125 - 224.125 | 220 | B'001_0011 | 1080.625 - 1220.625 | 1150 | B'011_1011 |
| 211.375 - 259.875 | 235 | B'010_0011 | 1128.125 - 1273.125 | 1200 | B'000_1011 |
| 225.625 - 275.625 | 250 | B'011_0011 | 1175.625 - 1325.625 | 1250 | B'001_1011 |
| 249.375 - 301.875 | 275 | B'000_0100 | 1211.25 - 1378.125 | 1300 | B'010_1011 |
| 273.125 - 328.125 | 300 | B'001_0100 | 1270.625 - 1430.625 | 1350 | B'011_1100 |
| 296.875 - 354.375 | 325 | B'010_0101 | 1318.125 - 1483.125 | 1400 | B'000_1100 |
| 320.625 - 380.625 | 350 | B'011_0101 | 1365.625 - 1500.00 | 1450 | B'001_1100 |
| 368.125 - 433.125 | 400 | B'000_0101 | 1413.125 - 1500.000 | 1500 | B'010_1100 |
| 415.125 - 485.125 | 450 | B'001_0110 | | | |

Note: Must set the values above in HSFREQRANGE.

Table 29.10 HSFREQRANGE Bit Set Values (RZ/G2M V1.3, RZ/G2M V3.0)

| Range (Mbps) | Default Bit Rate (Mbps) | HSFREQRANGE [5:0] | Range (Mbps) | Default Bit Rate (Mbps) | HSFREQRANGE [5:0] |
|-------------------|-------------------------|-------------------|-------------------|-------------------------|-------------------|
| 80 - 110.25 | 80 | B'00_0000 | 451.25 - 551.25 | 500 | B'01_0110 |
| 80 - 120.75 | 90 | B'01_0000 | 498.75 - 603.75 | 550 | B'00_0111 |
| 80 - 131.25 | 100 | B'10_0000 | 546.25 - 656.25 | 600 | B'01_0111 |
| 80.75 - 141.75 | 110 | B'11_0000 | 593.75 - 708.75 | 650 | B'00_1000 |
| 90.25 - 152.25 | 120 | B'00_0001 | 641.25 - 761.25 | 700 | B'01_1000 |
| 99.75 - 162.75 | 130 | B'01_0001 | 688.75 - 813.75 | 750 | B'00_1001 |
| 109.25 - 173.25 | 140 | B'10_0001 | 736.25 - 866.25 | 800 | B'01_1001 |
| 118.75 - 183.75 | 150 | B'11_0001 | 783.75 - 918.75 | 850 | B'10_1001 |
| 128.25 - 194.25 | 160 | B'00_0010 | 831.25 - 971.25 | 900 | B'11_1001 |
| 137.75 - 204.75 | 170 | B'01_0010 | 878.75 - 1023.75 | 950 | B'00_1010 |
| 147.25 - 215.25 | 180 | B'10_0010 | 926.25 - 1076.25 | 1000 | B'01_1010 |
| 156.75 - 225.75 | 190 | B'11_0010 | 973.75 - 1128.75 | 1050 | B'10_1010 |
| 171 - 241.5 | 205 | B'00_0011 | 1021.25 - 1181.25 | 1100 | B'11_1010 |
| 185.25 - 257.25 | 220 | B'01_0011 | 1068.75 - 1233.75 | 1150 | B'00_1011 |
| 199.5 - 273 | 235 | B'10_0011 | 1116.25 - 1286.25 | 1200 | B'01_1011 |
| 237.5 - 275.625 | 250 | B'11_0011 | 1163.75 - 1338.75 | 1250 | B'10_1011 |
| 249.375 - 301.875 | 275 | B'00_0100 | 1211.25 - 1391.25 | 1300 | B'11_1011 |
| 273.125 - 328.125 | 300 | B'01_0100 | 1258.75 - 1443.75 | 1350 | B'00_1100 |
| 296.875 - 354.375 | 325 | B'00_0101 | 1306.25 - 1496.25 | 1400 | B'01_1100 |
| 320.625 - 393.75 | 350 | B'01_0101 | 1353.75 - 1500.00 | 1450 | B'10_1100 |
| 356.25 - 446.25 | 400 | B'10_0101 | 1401.25 - 1500.00 | 1500 | B'11_1100 |
| 403.75 - 498.75 | 450 | B'00_0110 | | | |

Note: Must set the values above in HSFREQRANGE.

Table 29.11 TESTDIN_DATA Bit Set Values (RZ/G2E)

| Range (Mbps) | TESTDIN_DATA [7:0] | Range (Mbps) | TESTDIN_DATA [7:0] |
|--------------|--------------------|--------------|--------------------|
| 80 - 89 | B'0000_0000 | 650 - 699 | B'0011_0000 |
| 90 - 99 | B'0010_0000 | 700 - 749 | B'0001_0010 |
| 100 - 109 | B'0100_0000 | 750 - 799 | B'0011_0010 |
| 110 - 129 | B'0000_0010 | 800 - 849 | B'0101_0010 |
| 130 - 139 | B'0010_0010 | 850 - 899 | B'0111_0010 |
| 140 - 149 | B'0100_0010 | 900 - 949 | B'0001_0100 |
| 150 - 169 | B'0000_0100 | 950 - 999 | B'0011_0100 |
| 170 - 179 | B'0010_0100 | 1000 - 1049 | B'0101_0100 |
| 180 - 199 | B'0100_0100 | 1050 - 1099 | B'0111_0100 |
| 200 - 219 | B'0000_0110 | 1100 - 1125 | B'0001_0110 |
| 220 - 239 | B'0010_0110 | | |
| 240 - 249 | B'0100_0110 | | |
| 250 - 269 | B'0000_1000 | | |
| 270 - 299 | B'0010_1000 | | |
| 300 - 329 | B'0000_1010 | | |
| 330 - 359 | B'0010_1010 | | |
| 360 - 399 | B'0100_1010 | | |
| 400 - 449 | B'0000_1100 | | |
| 450 - 499 | B'0010_1100 | | |
| 500 - 549 | B'0000_1110 | | |
| 550 - 599 | B'0010_1110 | | |
| 600 - 649 | B'0001_0000 | | |

Note: Must set the values above in TESTDIN_DATA.

Table 29.12 PHTW Set Values (RZ/G2H, RZ/G2N)

| Default Bit Rate (Mbps) | PHTW [31:0] |
|--------------------------------|------------------------|
| 80 | H'0186_01F1 |
| 90 | H'0186_01F1 |
| 100 | H'0187_01F1 |
| 110 | H'0187_01F1 |
| 120 | H'0188_01F1 |
| 130 | H'0188_01F1 |
| 140 | H'0189_01F1 |
| 150 | H'0189_01F1 |
| 160 | H'018A_01F1 |
| 170 | H'018A_01F1 |
| 180 | H'018B_01F1 |
| 190 | H'018B_01F1 |
| 205 | H'018C_01F1 |
| 220 | H'018D_01F1 |
| 235 | H'018E_01F1 |
| 250 | H'018E_01F1 |

Table 29.13 Causes for Abnormal Reception

| Symptom | Condition | State | Possible Causes | |
|---------------------|--|--|-----------------|---|
| PHY will not start. | Data is not transferred from the camera. | STOPSTATE = low | Internal | The CSI2 (CSI40 or 20) channel started is different from the output channel on the camera side. CFG_CLK has been stopped. Module standby mode Software reset state (of CPG output) RSTZ = 0 SHUTDOWNZ = 0 ENABLE = 0, ENABLECLK = 0 on reception lane |
| | | | External | No lane connected CSI2 on the camera side has not been started. CSI2 on the camera side is not outputting LP-11. |
| | Data is transferred from the camera. | STOPSTATE is always low. | External | Camera side starts earlier and continues outputting data without stopping the clock lane (LP-11 has never been received.). |
| Reception failure | Data is transferred from the camera. | ERRSOTHS or ERRSOTSYNCHS is generated. | Internal | Different phases between the clock and data. Measures: Adjust the phase physically. Reduce the transfer rate. HSFREQRANGE value does not correspond to the transfer rate. Measures: Change the HSFREQRANGE set values. |
| | | | Internal | No interrupts are generated though ERRSOTHS or ERRSOTSYNCHS is not generated. Measures: Adjust the voltage level physically. Reduce the transfer rate. |
| | | | Internal | ECC error, CRC error, or asynchronous FIFO overflow occurs. Measures: Change the connections. Perform lane swapping. |
| | | | Internal | Clock lane is placed in ULP state. Measures: Change the connections. |
| Reception failure | Data is transferred from the camera. | No output or abnormal output though no error occurs. | External | Incorrect transmission protocol Measures: Transfer data in the order: Frame Start → Payload data → Frame End. Transfer at 1.5 Gbps or faster (4-lane transfer only) |
| | | | Internal | Incorrect LINK register setting (Same values in SEL_VC*, VCDT*_EN = 0 (* = null, 2 to 4)) |

29.3.10 Interrupt Generation by ULP Status, and Reception Error Status

Table 29.14 MIPI CSI-2 Escape Entry Code

| Escape Mode Action | Command Type | Entry Command Pattern (first bit transmitted to last bit transmitted) |
|-----------------------|--------------|--|
| Ultra-Low Power State | mode | B'0001 1110 |
| Undefined-1 | mode | B'1001 1111 |
| Undefined-2 | mode | B'1101 1110 |

- The ULP (ultra-low power) status start and end interrupts can be generated.

| Interrupt Source | Register Bit Name | Bit |
|--------------------------------|-------------------|-----|
| Start of ultra-low power state | ULPS_START | [7] |
| End of ultra-low power state | ULPS_END | [6] |

The state of the lanes can be monitored through the PHY clock lane monitoring register (PHCLM) and PHY data lane monitoring register (PHDLM).

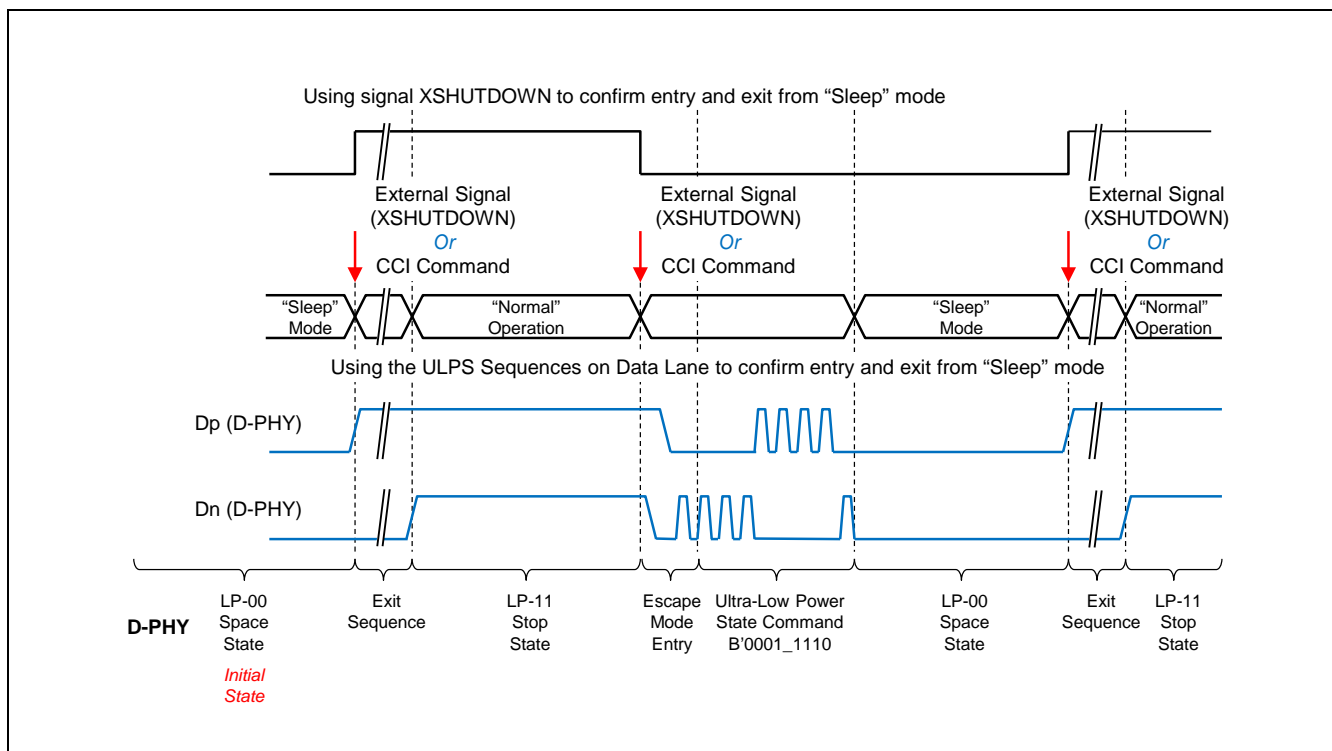


Figure 29.23 MIPI CSI-2 ULPS

- The reception error status interrupt can be generated.

| Interrupt Source | Register Bit Name | Bit |
|--|-------------------|-----|
| Synchronized SOT (start of transfer) error during HS reception | ERRSOTHS | [4] |
| Non-synchronizable SOT (start of transfer) error during HS reception | ERRSOTSYNCHS | [3] |
| Escape mode entry error | ERRESC | [2] |
| — | — | [1] |
| PHY control error | ERRCONTROL | [0] |

The error lanes related to ERRSOTHS and ERRSOTSYNCHS can be monitored through the interrupt error status monitoring register (INTERRSTATE) and the other error lanes can be monitored through the PHY ESC error monitoring register (PHEERM).

29.3.11 Monitoring Function

(1) Monitoring function

For monitoring of the packet header, see section 29.2.27 to 29.2.39 and 29.2.60 to 29.2.61; for monitoring of CRC reception data, see section 29.2.40 and 29.2.41; and for monitoring of the module internal signals used for debugging, see section 29.2.43 to 29.2.59.

29.3.12 Standby Mode

The CSI2 module can gate the clock signals to reduce power consumption. Module standby mode can be set by controlling the module stop register 7 (RMSTPCR7 and SMSTPCR7) of the clock pulse generator (CPG) module. CSI2 channels CSI40, CSI20 correspond to bits 16, 15, and 14 in the RMSTPCR7 and SMSTPCR7 registers, respectively. To release any of the CSI2 channels from standby mode, enable the corresponding the bits in the RMSTPCR7 and SMSTPCR7 registers. Enabling the bits allows all types of access to the channel.

To set any of the CSI2 channels to standby mode, execute the following procedure using the register corresponding to the channel.

[Enter Standby Mode]

1. Set RSTZ and SHUTDOWNZ to 0 in the PHY operation control register (PHYCNT).
2. Set TESTCLR to 1 in PHY Test Interface Clear register (PHTC).
3. Set the corresponding bit in the SRCR7 register of the Clock Pulse Generator (CPG) module (about SRCR7, refer to section 29.3.13).
4. Wait 100us.
5. Disable the corresponding bits in the RMSTPCR7 and SMSTPCR7 registers of the Clock Pulse Generator (CPG) module.

To set any of the CSI2 channels to operating mode from standby mode, execute the following procedure using the register corresponding to the channel.

[Exit Standby Mode]

1. Enable the corresponding bits in the RMSTPCR7 and SMSTPCR7 registers of the Clock Pulse Generator (CPG) module.
2. Set the corresponding bit in the SRSTCLR7 register of the Clock Pulse Generator (CPG) module (about SRSTCLR7, refer to section 29.3.13).
3. Return to Figure 29.16 [1] after step 2 of Exit Standby Mode.

29.3.13 Software Reset

A software reset can be set by controlling the software reset register 7 (SRCR7) and software reset clear register 7 (SRSTCLR7) of the clock pulse generator (CPG) module. CSI2 channels CSI40, CSI20 correspond to bits 16, 15, and 14 in the SRCR7 and SRSTCLR7 registers, respectively. To apply a software reset to any of the CSI2 channels, set the corresponding bit in the SRCR7 register. Enabling the bit initializes all the registers of the corresponding CSI2 channel. To cancel a software reset of any of the CSI2 channels, set the corresponding bit in the SRSTCLR7 register.

VIN module should be reset or stopped capture before reset of CSI2.

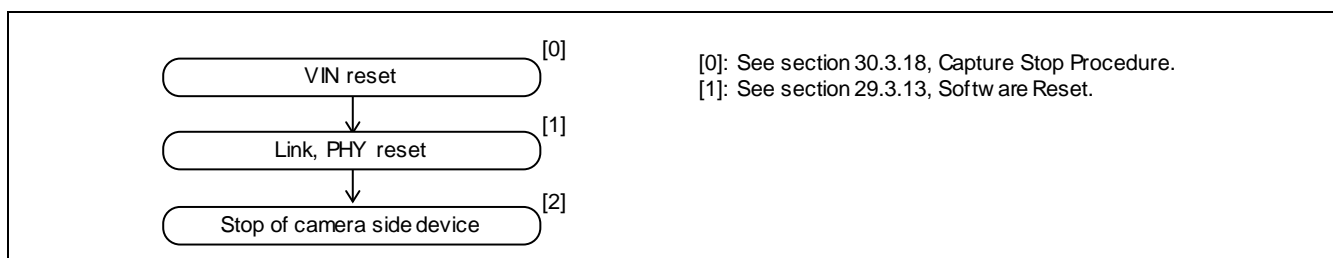


Figure 29.24 Example of Video Module Reset Procedure

To apply a software reset to any of the CSI2 channels, execute the following procedure using the register corresponding to the channel.

[Software Reset Execution]

1. Set the corresponding bit in the SRCR7 register of the Clock Pulse Generator (CPG) module.

[Software Reset Cancellation]

To cancel a software reset of any of the CSI2 channels and start the channel, execute the following procedure using the register corresponding to the channel.

1. Set the corresponding bit in the SRSTCLR7 register of the Clock Pulse Generator (CPG) module.
2. Return to Figure 29.16 [1] after step 1 of Software Reset Cancellation.

29.3.14 Input Clock Signals

(1) CSI2 LINK module

CPG clock input signals:

RZ/G2H: S2D1 ϕ (x_sck) and S2D2 ϕ (x_bck)

RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: S0D2 ϕ (x_sck) and S0D4 ϕ (x_bck)

RZ/G2E: S1D2 ϕ (x_sck) and S1D4 ϕ (x_bck)

The CSI2 module pin name is enclosed in ().

(2) CSI2 PHY module

CPG clock input signals: CSI0 ϕ (CFG_CLK)

The CSI2 module pin name is enclosed in ().

29.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

29.4.1 MIPI CSI-2 Transfer Rate

As described in section 29.3.1, a transfer rate must not exceed 1.5 Gbps*¹. If data is received at a rate exceeding 1.5 Gbps*¹, the following VIN cannot receive data, resulting in loss of the received data.

Notes: 1. Up to 1.1-Gbps transfer rate of MIPI CSI-2 in RZ/G2E.

29.4.2 PHY Operation Control Register (PHYCNT)

When setting any of the ENABLE_3 to ENABLE_0 bits to 1, be sure to set the ENABLECLK bit to 1. All the ENABLE_3 to ENABLE_0 and ENABLECLK bits are initially zero.

29.4.3 Channel Data Type Select Register VCDT and Channel Data Type Select Register VCDT2

The SEL_VC to SEL_VC4 bits must not be set to the same value. (A single virtual channel cannot be output to the multiple output channels.) If the same value is actually required, set to 0 the VCDT_EN to VCDT4_EN bits of the channels that are not to be output.

29.4.4 Synchronization Error during HS (Hi-Speed) Reception

If normal HS reception fails, the PHY outputs a synchronization error to the LINK. When a synchronization error occurs, the PHY may not output HS reception data to the LINK. If this occurs only in the specific lane, the lane synchronization function prevents the FIFO in the LINK from being read thus causing an overflow. The synchronization error and overflow can be monitored through the interrupt error status monitoring register (INTERRSTATE) by setting the interrupt source mask register (INTCLOSE) to 0.

After an overflow occurs, the FIFO should be initialized by software reset. About software reset, refer to section 29.3.13 Software Reset.

29.4.5 FLD Signal

Since the FLD signal is not generated in the initial state, appropriately set the field detection control register (FLD) in order to generate the FLD signal.

It is shown about Interlaced Video Example at Figure 29.25.

In this case (frame number of even field is H'0002), set FLD_NUM to same value as frame number of even field (H'0002) in field detection control register (FLD) and set FLE_EN* to 1 in field detection control register (FLD).

Note: Frame number is included in FS and FE. Frame number increments by 1 every FS packet with the same virtual channel and is periodically reset to one e.g. 1, 2, 1,2,1,2.

If frame number of even field is H'0001, set FLD_NUM to H'0001.

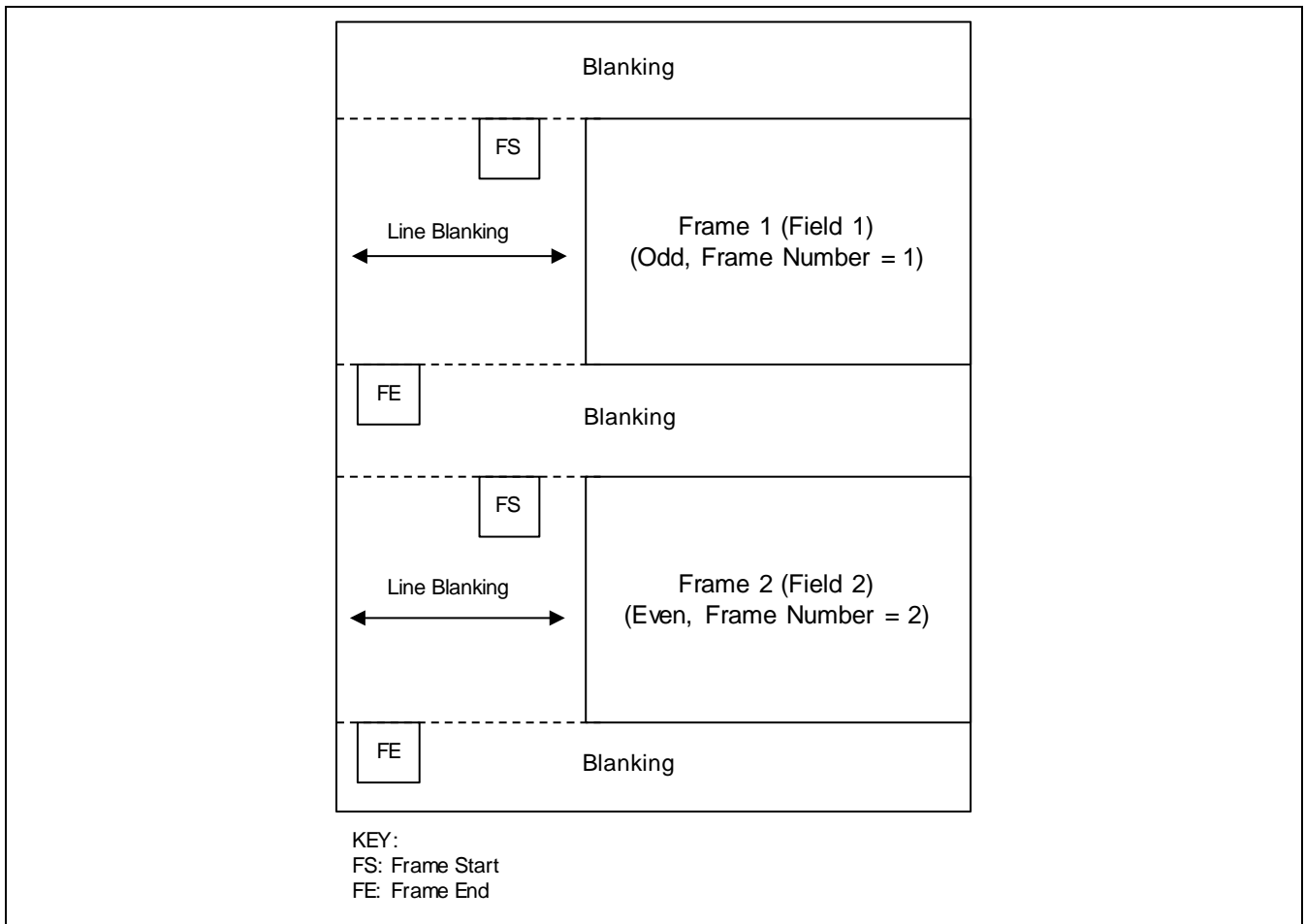


Figure 29.25 Interlaced Video Example

29.4.6 INT_LESS_THAN_WC

When debugging INT_LESS_THAN_WC, only use one of virtual channel.

29.4.7 Transfer Rate of Interleaved Transmission

When some virtual channels are used, the transfer rate input to PHY should be same each virtual channel.

In order to maintain the frame rate, the transfer rate should be a total value of each virtual channel.

29.4.8 Lane Setting

CSI2 supports to set number of lane as below.

Table 29.15 CSI2 supports number of lanes

| Product | Support lane setting | |
|-------------|----------------------|-------|
| | CSI40 | CSI20 |
| RZ/G2H | 4/2/1 | 2/1 |
| RZ/G2M V1.3 | 4 | 2/1 |
| RZ/G2M V3.0 | 4/2/1 | 2/1 |
| RZ/G2N | 4/2/1 | 2/1 |
| RZ/G2E | 2/1 | — |

29.4.9 V-blanking Period

In order to receive correctly, the minimum v-blanking period is as follows.

It requires 3 lines or more v-blanking.

Note: v-blanking = period from Frame End packet to Frame Start packet.

The minimum and maximum value of VD_MSK_CYCLE does not mean that supported minimum and maximum v-blanking period.

29.4.10 Notes on data

Input data not conformant to MIPI CSI-2 standard might cause CSI2 module to hang or unexpected operation.

30. Video Input Module (VIN)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

30.1 Overview

The video input module (hereinafter abbreviated as VIN) is a video capture module that stores in external memory YCbCr-422 data through the ITU-R BT.601, ITU-R BT.656, or ITU-R BT.709 interface and RGB data through the ITU-R BT.601 or ITU-R BT.709 interface.

The module has up to eight video channels that can independently control the capture of data into a capture area of up to 4096×4096 pixels. It can also provide vertical and horizontal scaling of the data by up to three and two times, respectively.

For captured video data, the VIN provides a color space conversion function from YCbCr-422 to RGB, a format conversion function from RGB to ARGB.

As the VIN internally generates a field signal, it can capture progressive data.

Input of data through digital pins (only present for video channels 4 and 5) and through the MIPI Alliance CSI-2 interfaces is possible.

Notes: 1. For detailed specifications of the MIPI CSI-2 interface, see section 29.

2. When the ITU-R BT.709 interface is in use, the maximum frequency of the video clock of Digital pin is 100 MHz.
3. The upper limits on size clipping depends on the use case.
4. The number of video input channels for each product is as follows:

| | |
|--------------|------------|
| RZ/G2H: | 8 channels |
| RZ/G2M V1.3: | 8 channels |
| RZ/G2M V3.0: | 8 channels |
| RZ/G2N: | 8 channels |
| RZ/G2E: | 2 channels |

30.1.1 Features

(1) Video Channels 0 and 4

The following input interfaces can be selected for these channels.

Table 30.1 Input Interface for Video Channels 0 and 4 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, Channel 4 for RZ/G2E

| | | | Second Generation RZ/G Series Products | | | | |
|------------------------|------------|----------------------------------|---|----------------------------|--------|--------|--|
| Interface (CSI2) | Data Width | Data Type | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Interface (MIPI CSI-2) | — | YCbCr-422 8-bit data*1 | √ | √ | √ | √ | |
| Interface (MIPI CSI-2) | — | YCbCr-422 10-bit data*1*2 | √ | √ | √ | √ | |
| Interface (MIPI CSI-2) | — | RGB-888 data*1 | √ | √ | √ | √ | |
| Interface (MIPI CSI-2) | — | 8-bit user defined data (RAW8)*1 | √ | √ | √ | √ | |

Notes: 1. Inputs are in the formats which conform to the standard for the MIPI CSI-2 interface.

2. Disable the XY scaling settings when using the YCbCr-422 10-bit format (only the 100% scaling is enabled).

Table 30.2 Input Interface for Video Channel 4 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E

| | | | Second Generation RZ/G Series Products | | | | |
|-------------------------------------|---------------|--|---|----------------------------|--------|--------|--|
| Interface (Digital Pins) | Data Width | Data Type | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| ITU-R BT.601/BT.709*1*3 | 8/10/12 bits | YCbCr-422 data (UYVY format) | √ | √ | √ | √ | |
| ITU-R BT.601/BT.709*3/ BT.1358*2 | 16/20/24 bits | YCbCr-422 data (8/10/12 bits (Y) + 8/10/12 bits (CbCr) format) | √ | √ | √ | √ | |
| ITU-R BT.656*1 | 8/10/12 bits | YCbCr-422 data (UYVY format) | √ | √ | √ | √ | |
| ITU-R BT.601/BT.709*3 | 18 bits | RGB-666 data | √ | √ | √ | √ | |
| ITU-R BT.601/BT.709*3 | 24 bits | RGB-888 data | √ | √ | √ | √ | |
| RAW | 12 bits | RAW12 | √ | √ | √ | √ | |

Notes: 1. Disable the XY scaling settings when using the 10/12-bit YCbCr-422 format in the ITU-R BT.601, BT.709, or BT.656 interface (only the 100% scaling is enabled).

2. Disable the XY scaling settings when using the 20/24-bit YCbCr-422 format in the ITU-R BT.601, BT.709, or BT.1358 interface (only the 100% scaling is enabled).

3. When the ITU-R BT.709 interface is in use, the maximum frequency of the video clock of Digital pin is 100 MHz

(2) Video Channels 1 and 5

The following input interfaces can be selected for these channels.

Table 30.3 Input Interface for Video Channels 1 and 5 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, Channel 5 for RZ/G2E

| | | | Second Generation RZ/G Series Products | | | | |
|-------------------------|-------------------|----------------------------------|---|--------------------|--------------------|---------------|---------------|
| Interface (CSI2) | Data Width | Data Type | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Interface (MIPI CSI-2) | — | YCbCr-422 8-bit data*1 | √ | √ | √ | √ | √ |
| Interface (MIPI CSI-2) | — | YCbCr-422 10-bit data*1*2 | √ | √ | √ | √ | √ |
| Interface (MIPI CSI-2) | — | RGB-888 data*1 | √ | √ | √ | √ | √ |
| Interface (MIPI CSI-2) | — | 8-bit user defined data (RAW8)*1 | √ | √ | √ | √ | √ |
| Interface (MIPI CSI-2) | — | 8-bit embedded data*1 | — | — | — | — | √ |

Notes: 1. Inputs are in the formats which conform to the standard for the MIPI CSI-2 interface.

2. Disable the XY scaling settings when using the YCbCr-422 10-bit format (only the 100% scaling is enabled).

Table 30.4 Input Interface for Video Channel 5 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E

| | | | Second Generation RZ/G Series Products | | | | |
|-------------------------------------|-------------------|--|---|--------------------|--------------------|---------------|---------------|
| Interface (Digital Pins) | Data Width | Data Type | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| ITU-R BT.601/BT.709*1*3 | 8/10/12 bits | YCbCr-422 data (UYVY format) | √ | √ | √ | √ | √ |
| ITU-R BT.601/BT.709*3/ BT.1358*2 | 16 bits | YCbCr-422 data (8 bits (Y) + 8 bits (CbCr) format) | √ | √ | √ | √ | √ |
| ITU-R BT.656*1 | 8/10/12 bits | YCbCr-422 data (UYVY format) | √ | √ | √ | √ | √ |
| RAW | 12 bits | RAW12 | √ | √ | √ | √ | √ |

Notes: 1. Disable the XY scaling settings when using the 10/12-bit YCbCr-422 format in the ITU-R BT.601, BT.709, or BT.656 interface (only the 100% scaling is enabled).

2. Disable the XY scaling settings when using the 20/24-bit YCbCr-422 format in the ITU-R BT.601, BT.709, or BT.1358 interface (only the 100% scaling is enabled).

3. When the ITU-R BT.709 interface is in use, the maximum frequency of the video clock of Digital pin is 100 MHz.

(3) Video Channels 2 and 6 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N)

The following input interfaces can be selected for these channels.

Table 30.5 Input Interface for Video Channels 2 and 6 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N

| | | | Second Generation RZ/G Series Products | | | | |
|-------------------------|-------------------|----------------------------------|---|------------------------------------|---------------|---------------|--|
| Interface (CSI2) | Data Width | Data Type | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Interface (MIPI CSI-2) | — | YCbCr-422 8-bit data*1 | √ | √ | √ | — | |
| Interface (MIPI CSI-2) | — | YCbCr-422 10-bit data*1 | √ | √ | √ | — | |
| Interface (MIPI CSI-2) | — | RGB-888 data*1 | √ | √ | √ | — | |
| Interface (MIPI CSI-2) | — | 8-bit user defined data (RAW8)*1 | √ | √ | √ | — | |

Note: 1. Inputs are in the formats which conform to the standard for the MIPI CSI-2 interface.

(4) Video Channels 3 and 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N)

The following input interfaces can be selected for these channels.

Table 30.6 Input Interface for Video Channels 3 and 7 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N

| | | | Second Generation RZ/G Series Products | | | | |
|-------------------------|-------------------|-------------------------|---|------------------------------------|---------------|---------------|--|
| Interface (CSI2) | Data Width | Data Type | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Interface (MIPI CSI-2) | — | YCbCr-422 8-bit data*1 | √ | √ | √ | — | |
| Interface (MIPI CSI-2) | — | YCbCr-422 10-bit data*1 | √ | √ | √ | — | |
| Interface (MIPI CSI-2) | — | RGB-888 data*1 | √ | √ | √ | — | |
| Interface (MIPI CSI-2) | — | 8-bit embedded data*1 | √ | √ | √ | — | |

Note: 1. Inputs are in the formats which conform to the standard for the MIPI CSI-2 interface.

(5) Internal Sync Signal Generation

For video data capturing through the ITU-R BT.601 or ITU-R BT.709 interface, the field signal can be internally generated even if the input sync signals stops (In video data capture through the ITU-R BT.656 interface, only the field signal is generated).

(6) Capture Mode

The following four modes can be selected to capture the interlace images. In addition, single frame capture or continuous frame capture mode can be selected.

Triple-buffering control is provided in accordance with the captured field image and frame image to coordinate with the video capture mode of the display module.

- Odd-field capture mode
- Even-/odd-field capture mode
- Even-field capture mode
- Full interlace capture mode

(7) Vertical and Horizontal Scaling

- Video channels 0, 1, 4, and 5 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N only)
Factors for scaling up or down can be set to values in the range from one sixteenth to three in the vertical direction and from one sixteenth to two in the horizontal direction.
- Video channels 4 and 5 (RZ/G2E only)
Factors for scaling up or down can be set to values in the range from one sixteenth to three in the vertical direction and from one sixteenth to two in the horizontal direction.
- Video channels 2, 3, 6, and 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N only)
These channels do not support scaling up or down in either direction.

Notes: 1. When scaling-up is used, horizontal size that can be input is up to 2048 pixel.
2. When scaling-down is used, horizontal size that can be output is up to 2048 pixel.

(8) Size Clipping

- Video channels 0, 1, 4, and 5 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N only)
The VIN module has two clipping circuits, which independently handle images with up to 4096×4096 pixels. Any capture size within this limit can be specified before or after scaling.
- Video channels 4 and 5 (RZ/G2E only)
The VIN module has two clipping circuits, which independently handle images with up to 4096×4096 pixels. Any capture size within this limit can be specified before or after scaling.
- Video channels 2, 3, 6, and 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N only)
The VIN module has one clipping circuit, which independently handles images with up to 4096×4096 pixels. However, it does not support scaling.

Note: The upper limits on size clipping depend on the use case.

(9) Color Space Conversion

Color space conversion can be performed from YC to RGB or from RGB to YC. Desired conversion coefficients can be specified through registers to adjust colors.

(10) Lookup Table (LUT) Density Conversion

The lookup table (LUT) density can be converted from 10 bits to 8 bits for each pixel according to the color space conversion result.

Note: When the density conversion function is not used, the upper 8 bits of data are output.

(11) Image Data Format Conversion

To the density-converted YCbCr444 or RGB888 image data, the following data format conversions are available:

- YCbCr image data
 - Y/Cb/Cr 8-bit multiplex conversion
 - YCbCr444 → YC separation (separated into Y and CbCr components.)
 - YCbCr444 → Y component extraction
- RGB image data
 - RGB-888 → 32 bits/pixel conversion
 - RGB-888 → RGB-565 (16 bits/pixel) conversion
 - RGB-888 → ARGB-1555 (16 bits/pixel) conversion
 - RGB-888 → RGB-888 (32 bits/pixel) conversion

Note: The lookup table is common to the YCbCr and RGB formats.

(12) Memory Output Data Format

Format-converted image data can be transferred to the memory. The following shows the available formats of the data stored.

- YCbCr image data
 - Y/Cb/Cr422, 8-bit multiplexed
 - Y/Cb/Cr, 10-bit multiplexed
 - Y/Cb/Cr, 12-bit multiplexed
 - YC separation, YCbCr422, Y, 8-bit Cb/Cr, 8-bit multiplexed
 - YC separation, YCbCr420, Y, 8-bit Cb/Cr, 8-bit multiplexed ^{*1*2}
 - YC separation, YCbCr422, Y, 10-bit Cb/Cr, 8-bit multiplexed
 - YC separation, YCbCr422, Y, 12-bit Cb/Cr, 8-bit multiplexed
 - YC separation, YCbCr422, Y, 10-bit Cb/Cr, 10-bit multiplexed
 - YC separation, YCbCr422, Y, 12-bit Cb/Cr, 12-bit multiplexed
 - YC separation, Y data, 8-bit
 - YC separation, Y data, 10-bit
 - YC separation, Y data, 12-bit
- RGB image data
 - RGB-565 (16 bits/pixel)
 - ARGB-1555 (16 bits/pixel)
 - RGB-888 (32 bits/pixel)
 - ARGB-8888 (32 bits/pixel)

Notes: 1. Video channels 0, 1, 4, and 5 only support these formats.

2. When outputs are in the YCbCr420 format, disable any settings for XY scaling (scaling up and down are not available in this case).

30.1.2 Block Diagram

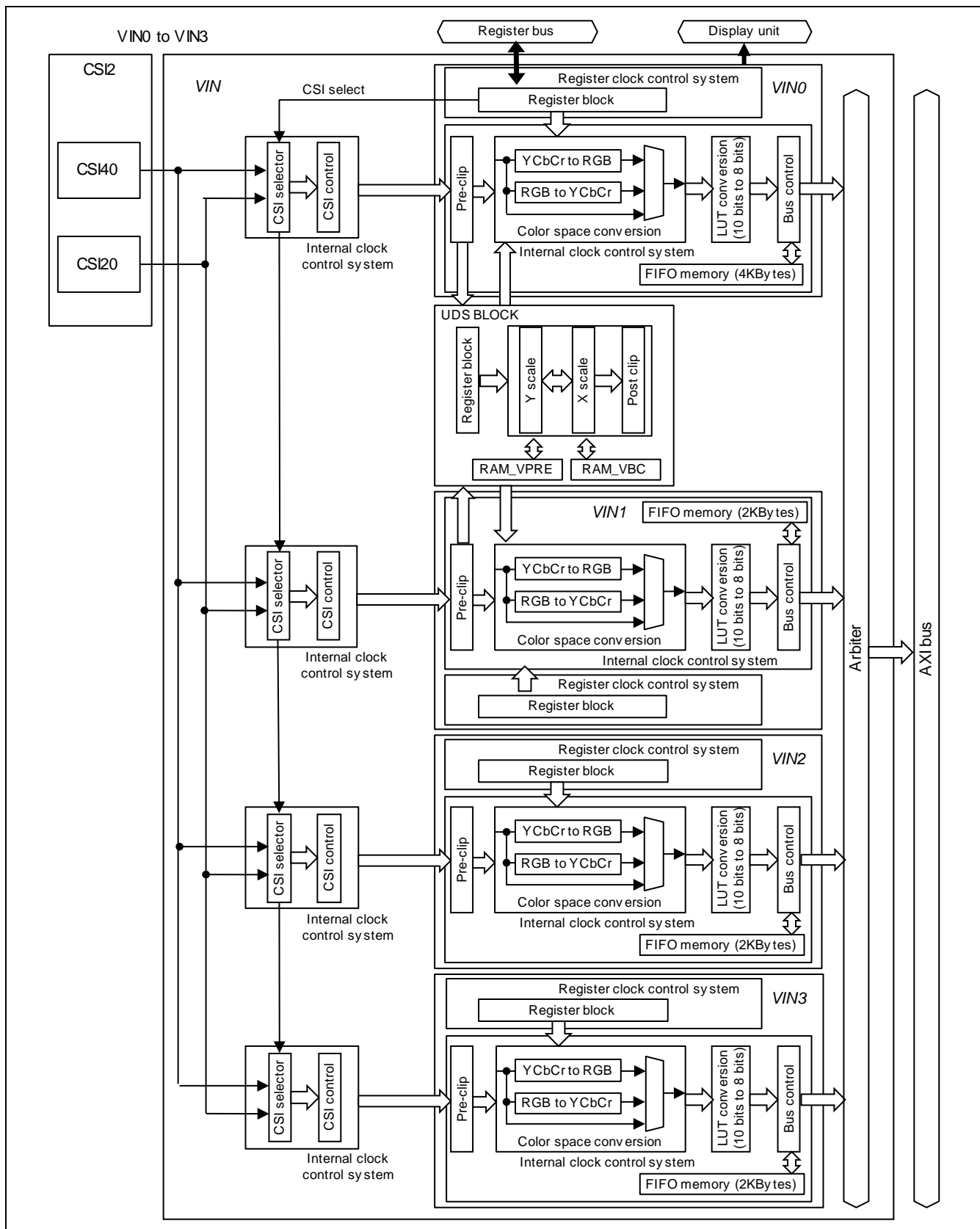


Figure 30.1 Functional Block Diagram of Video Input Modules 0 to 3 (VIN0 to VIN3)
(RZ/G2M V1.3, RZ/G2M V3.0)

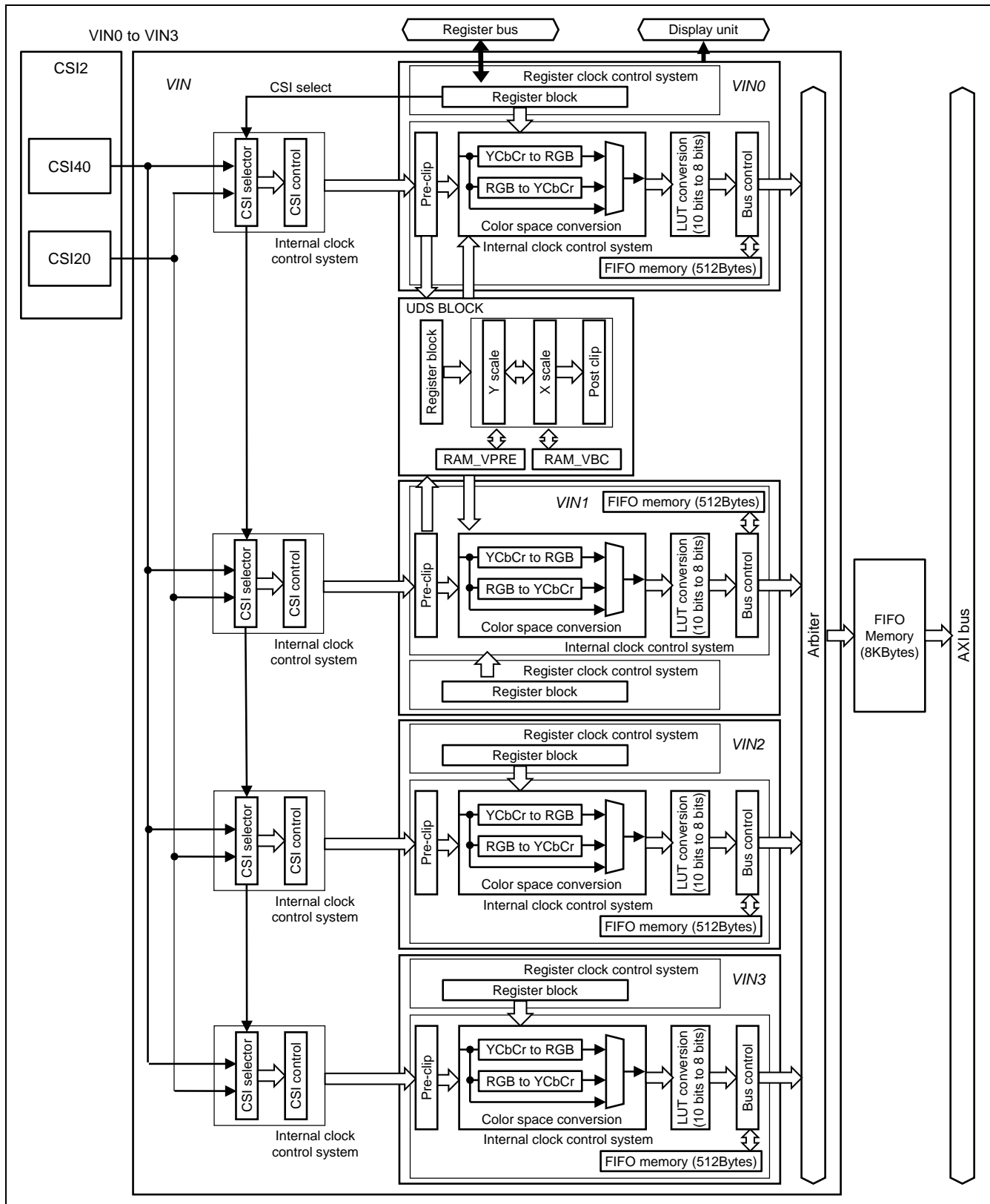


Figure 30.2 Functional Block Diagram of Video Input Modules 0 to 3 (VIN0 to VIN3) (RZ/G2N)

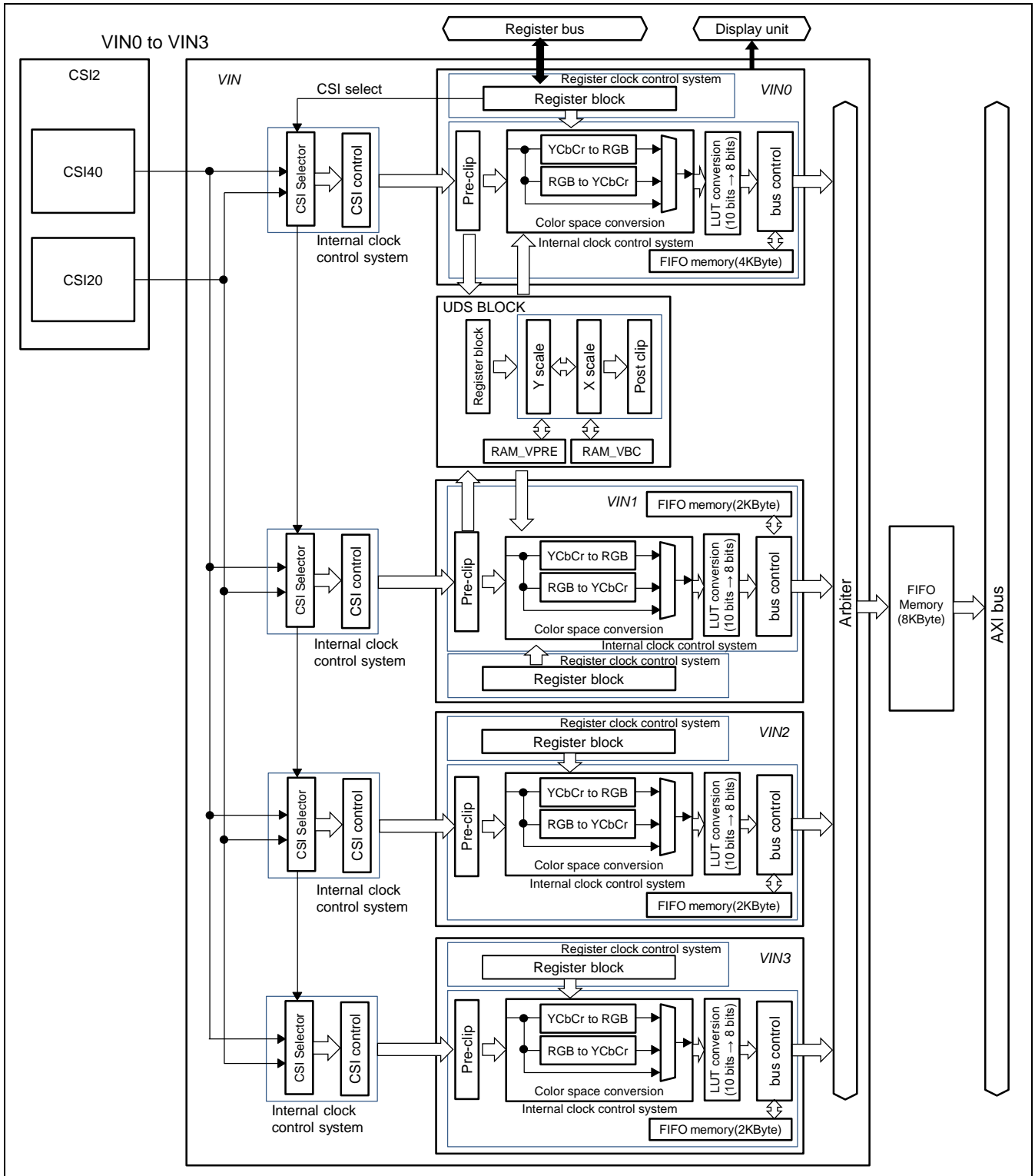


Figure 30.3 Functional Block Diagram of Video Input Modules 0 to 3 (VIN0 to VIN3) (RZ/G2H)

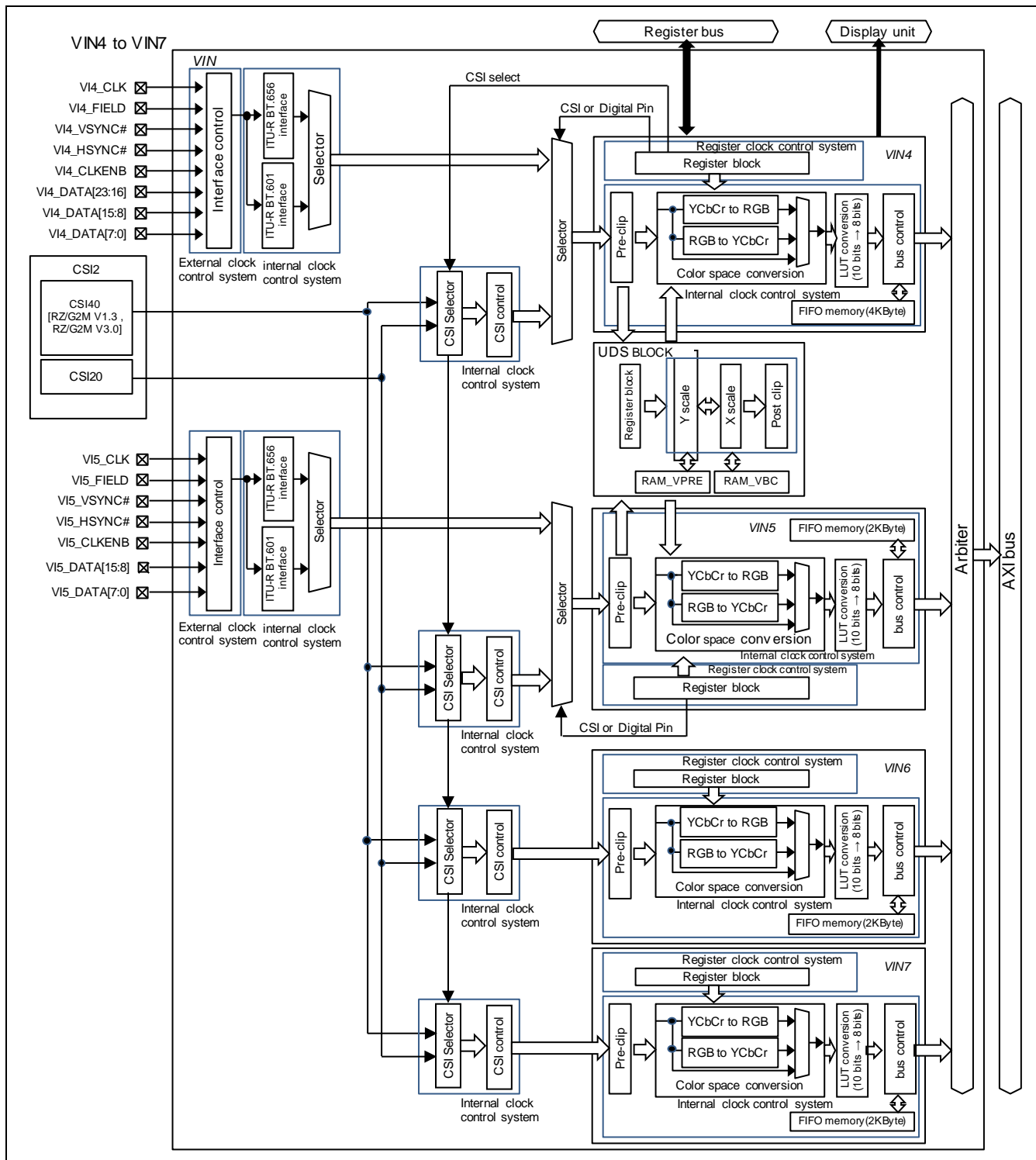


Figure 30.4 Functional Block Diagram of Video Input Modules 4 to 7 (VIN4 to VIN7) (RZ/G2M V1.3, RZ/G2M V3.0)

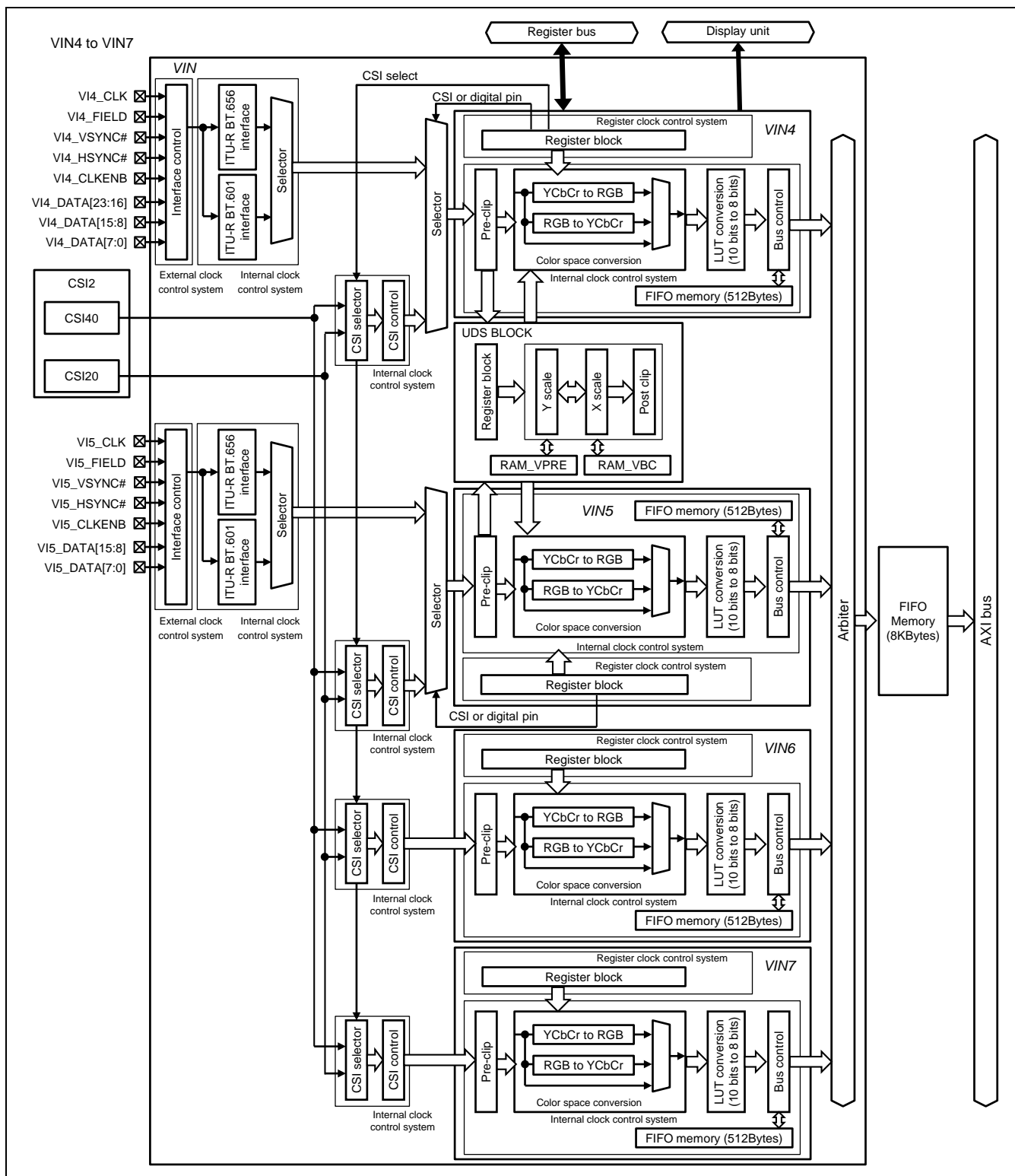


Figure 30.5 Functional Block Diagram of Video Input Modules 4 to 7 (VIN4 to VIN7) (RZ/G2N)

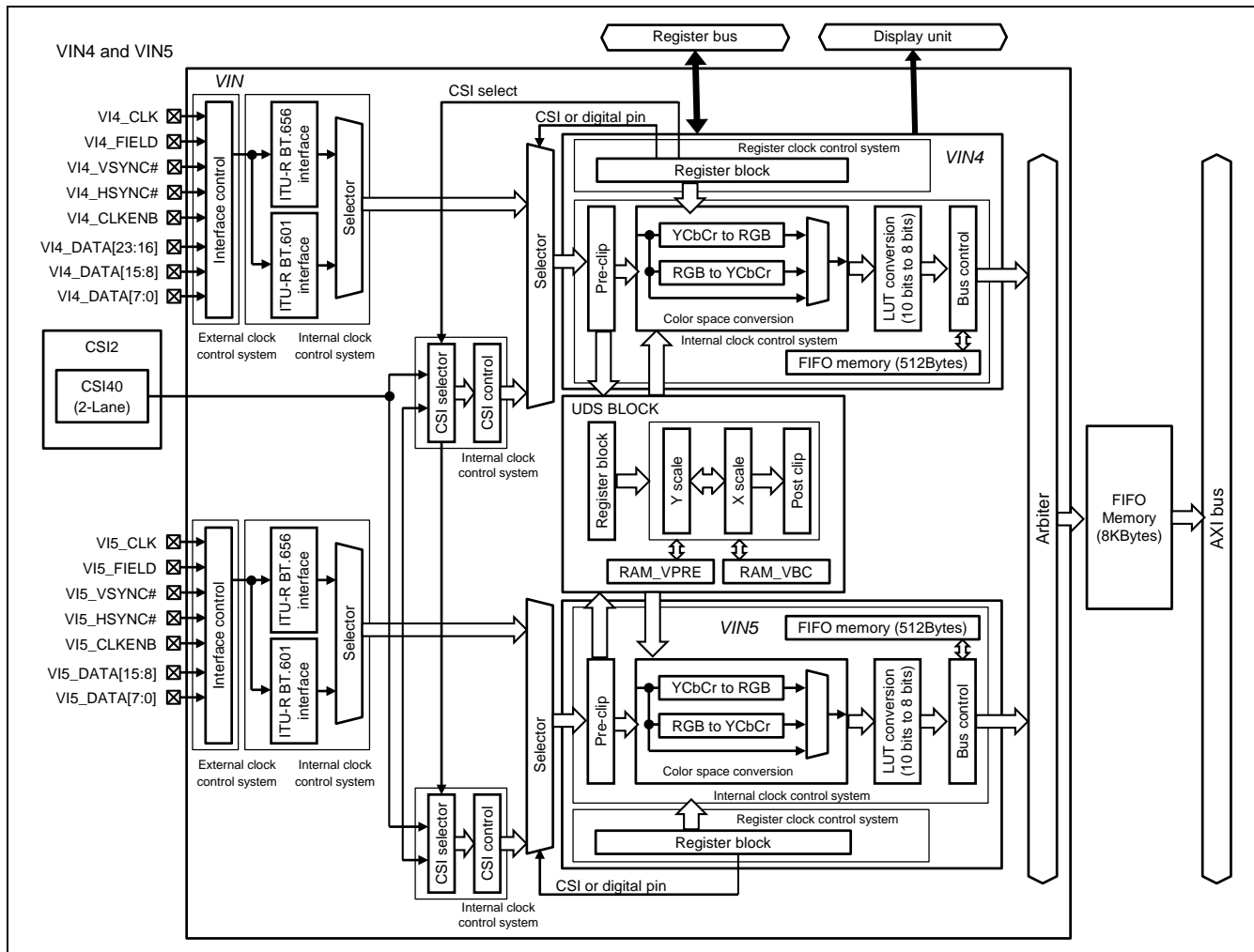


Figure 30.6 Functional Block Diagram of Video Input Modules 4 and 5 (VIN4 and VIN5) (RZ/G2E)

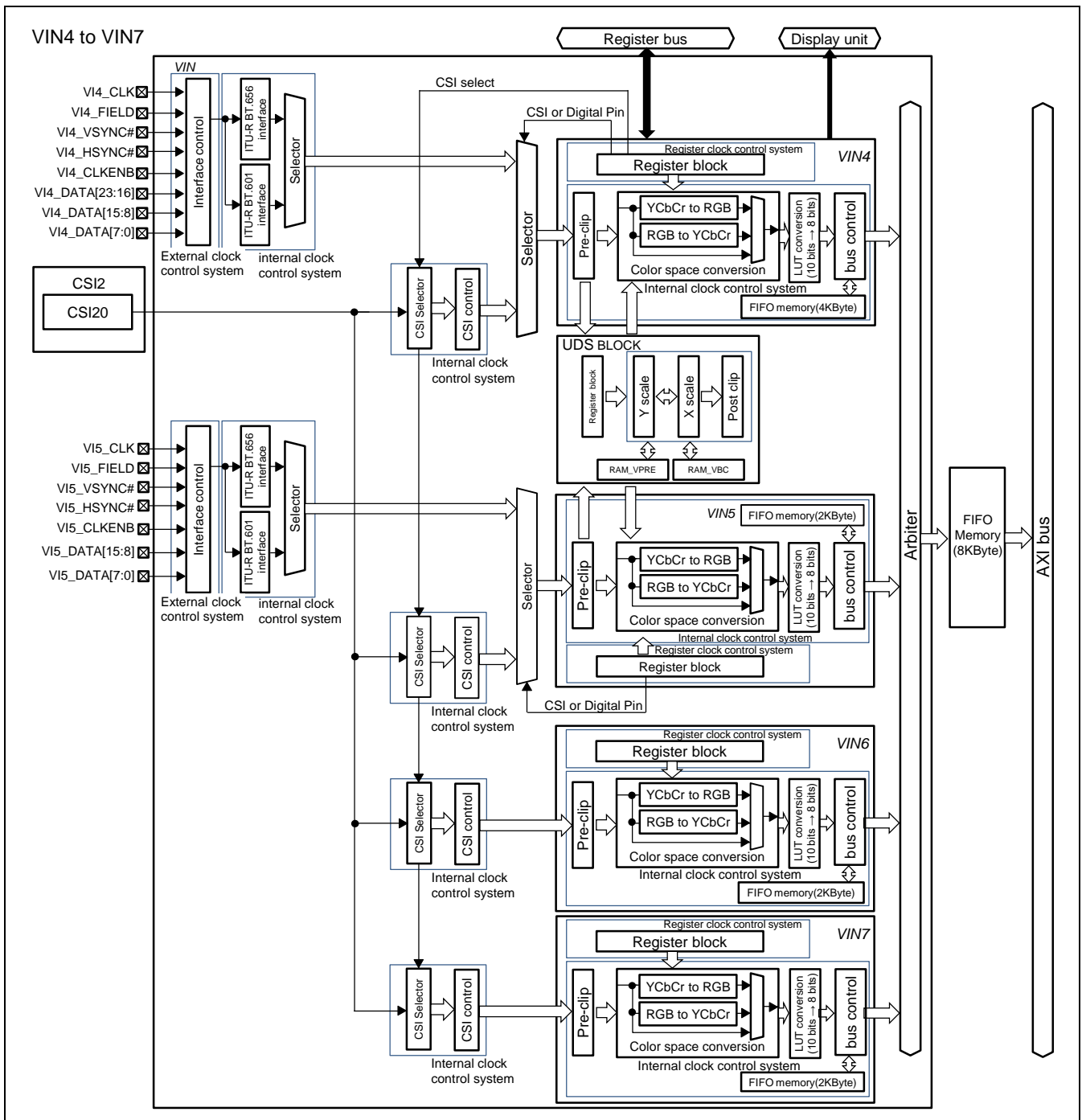


Figure 30.7 Functional Block Diagram of Video Input Modules 4 to 7 (VIN4 to VIN7) (RZ/G2H)

30.1.3 External Pins

Table 30.7 Pin Configuration

| | | | | Second Generation RZ/G Series Products | | | | |
|---------------------------------------|-----------------------------|-------|--|---|-------------|-------------|--------|--------|
| Pin Name | Function | I/O | Description | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| VIN4 video clock (clock) | VI4_CLK | Input | External video clock in the ITU-R BT.601, ITU-R BT.656, ITU-R BT.1358, or ITU-R BT.709 interface | √ | √ | √ | √ | √ |
| VIN4 field signal (control) | VI4_FIELD | Input | Field signal in the ITU-R BT.601 or ITU-R BT.709 interface Fix this pin high or low when the pin is not used. | √ | √ | √ | √ | √ |
| VIN4 vertical sync signal (control) | VI4_VSYNC# | Input | Vertical sync signal in the ITU-R BT.601, ITU-R BT.1358, or ITU-R BT.709 interface Fix this pin high or low when the pin is not used. | √ | √ | √ | √ | √ |
| VIN4 horizontal sync signal (control) | VI4_HSYNC# | Input | Horizontal sync signal in the ITU-R BT.601, ITU-R BT.1358, or ITU-R BT.709 interface Fix this pin high or low when the pin is not used. | √ | √ | √ | √ | √ |
| VIN4 data enable (control) | VI4_CLKENB | Input | Data enable signal in the ITU-R BT.601, ITU-R BT.1358, or ITU-R BT.709 interface Fix this pin high or low when the pin is not used. If the signal is not present in the ITU-R BT.601, ITU-R BT.1358, or ITU-R BT.709 interface, connect the pin to a horizontal sync signal. | √ | √ | √ | √ | √ |
| VIN4 video data (video data) | VI4_DATA23 to VI4_DATA 0 | Input | Data signals in the ITU-R BT.601, ITU-R BT.656, ITU-R BT.1358, or ITU-R BT.709 interface Fix these pins high or low respectively when these pins are not used in whole or in part. | √ | √ | √ | √ | √ |
| VIN5 video clock (clock) | VI5_CLK | Input | External video clock in the ITU-R BT.601, ITU-R BT.656, ITU-R BT.1358, or ITU-R BT.709 interface | √ | √ | √ | √ | √ |
| VIN5 field signal (control) | VI5_FIELD | Input | Field signal in the ITU-R BT.601 or ITU-R BT.709 interface Fix this pin high or low when the pin is not used. | √ | √ | √ | √ | √ |
| VIN5 vertical sync signal (control) | VI5_VSYNC# | Input | Vertical sync signal in the ITU-R BT.601, ITU-R BT.1358, or ITU-R BT.709 interface Fix this pin high or low when the pin is not used. | √ | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Pin Name | Function | I/O | Description | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---------------------------------------|--------------------------|-------|--|--------|-------------|-------------|--------|--------|
| VIN5 horizontal sync signal (control) | VI5_HSYNC# | Input | Horizontal sync signal in the ITU-R BT.601, ITU-R BT.1358, or ITU-R BT.709 interface Fix this pin high or low when the pin is not used. | √ | √ | √ | √ | √ |
| VIN5 data enable (control) | VI5_CLKENB | Input | Data enable signal in the ITU-R BT.601, ITU-R BT.1358, or ITU-R BT.709 interface Fix this pin high or low when the pin is not used. If the signal is not present in the ITU-R BT.601, ITU-R BT.1358, or ITU-R BT.709 interface, connect the pin to a horizontal sync signal. | √ | √ | √ | √ | √ |
| VIN5 video data (video data) | VI5_DATA15 to VI5_DATA 0 | Input | Data signals in the ITU-R BT.601, ITU-R BT.656, ITU-R BT.1358, or ITU-R BT.709 interface Fix these pins high or low respectively when these pins are not used in whole or in part. | √ | √ | √ | √ | √ |

Table 30.8 Channel 4 Data Pin Connections

| Input data format | RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E: VI4_DATA[23:16] | | | | | | | | RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E: VI4_DATA[15:8] | | | | | | | | RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E: VI4_DATA[7:0] | | | | | | | | | | | |
|---|---|---|---|---|-------------------|---|---|---|--|---|---|---|------------------|--------------------------|-------------------------|---|---|---|---|---|-----|---|---|---|---|---|---|---|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | |
| ITU-R BT.601/BT.709/BT.656 8-bit YCbCr-422 (VnDMR2/YDS = 0) | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | Y/Cb/Cr video data[7:0] | | | | | | | | | | | |
| ITU-R BT.601/BT.709/BT.656 8-bit YCbCr-422 (VnDMR2/YDS = 1) | * | * | * | * | * | * | * | * | Y/Cb/Cr video data[7:0] | | | | | | | | * | * | * | * | * | * | * | * | * | * | * | * |
| ITU-R BT.601/BT.709/BT.656 10-bit YCbCr-422 | * | * | * | * | * | * | * | * | * | * | * | * | * | * | Y/Cb/Cr video data[9:0] | | | | | | | | | | | | | |
| ITU-R BT.601/BT.709/BT.656 12-bit YCbCr-422 | * | * | * | * | * | * | * | * | * | * | * | * | * | Y/Cb/Cr video data[11:0] | | | | | | | | | | | | | | |
| RAW12 | * | * | * | * | * | * | * | * | * | * | * | * | RAW12 data[11:0] | | | | | | | | | | | | | | | |
| ITU-R BT.601/BT.709/BT.1358 16-bit YCbCr-422 | * | * | * | * | * | * | * | * | Y video data[7:0] | | | | | | | | Cb/Cr video data[7:0] | | | | | | | | | | | |
| ITU-R BT.601/BT.709/BT.1358 20-bit YCbCr-422 | * | * | * | * | Y video data[9:0] | | | | | | | | | | | | Cb/Cr video data[9:0] | | | | | | | | | | | |
| ITU-R BT.601/BT.709/BT.1358 24-bit YCbCr-422 | Y video data[11:0] | | | | | | | | | | | | | | | | Cb/Cr video data[11:0] | | | | | | | | | | | |
| ITU-R BT.601/BT.709 RGB-666 | R video data[5:0] | | | | * * | | | | G video data[5:0] | | | | * * | | | | B video data[5:0] | | | | * * | | | | | | | |
| ITU-R BT.601/BT.709 24-bit RGB-888 | R video data[7:0] | | | | | | | | G video data[7:0] | | | | | | | | B video data[7:0] | | | | | | | | | | | |

Note: * Fix the pins at a high or low level.

Table 30.9 Channel 5 Data Pin Connections (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E)

| Input data format | RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E: — | | | | | | | | RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E: VI5_DATA[15:8] | | | | | | | | RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E: VI5_DATA[7:0] | | | | | | | | |
|--|---|---|---|---|---|---|---|---|--|---|---|---|---|---|--------------------------|-------------------------|---|---|---|---|---|---|---|---|--|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| ITU-R BT.601/BT.709/BT.656 8-bit YCbCr-422 (VnDMR2/YDS = 0) | — | — | — | — | — | — | — | — | * | * | * | * | * | * | * | * | Y/Cb/Cr video data[7:0] | | | | | | | | |
| ITU-R BT.601/BT.709/BT.656 8-bit YCbCr-422 (VnDMR2/YDS = 1) | — | — | — | — | — | — | — | — | Y/Cb/Cr video data[7:0] | | | | | | | | * | * | * | * | * | * | * | * | |
| ITU-R BT.601/BT.709/BT.656 10-bit YCbCr-422 | — | — | — | — | — | — | — | — | * | * | * | * | * | * | * | Y/Cb/Cr video data[9:0] | | | | | | | | | |
| ITU-R BT.601/BT.709/BT.656 12-bit YCbCr-422 | — | — | — | — | — | — | — | — | * | * | * | * | * | * | Y/Cb/Cr video data[11:0] | | | | | | | | | | |
| RAW12 | — | — | — | — | — | — | — | — | * | * | * | * | * | * | RAW12 data[11:0] | | | | | | | | | | |
| ITU-R BT.601/BT.709/BT.1358 16-bit YCbCr-422 | — | — | — | — | — | — | — | — | Y video data[7:0] | | | | | | | Cb/Cr video data[7:0] | | | | | | | | | |

Note: * Fix the pins at a high or low level.
 — These pins are not present on the given devices.

30.1.4 Register Configuration

Table 30.10 through Table 30.13 show the VIN register configuration for each channel.

Notes: Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined. The access size is no supported except for 32 bits.

* For the internal update mode, refer to the description of the VUP bit in the main control register (VnMC).

Δ: The internal update mode supports VnMC.CLP[1:0], VnMC.DPINE, VnMC.FOC, VnMC.LUTE bit only.

(1) Channels 0 and 4

Table 30.10 VIN0 and VIN4 Registers

Channels 0 and 4 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), Channels 4 (RZ/G2E)

| | | | | | | | Second Generation RZ/G Series Products | | | | |
|---------------------------------------|---------------|------------|----------------|----------------------|--------------------|--------------------------------------|---|--------------------|--------------------|---------------|---------------|
| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Video 0 main control register | V0MC | R/W | H'E6EF_0000 | H'0000_0000 | 32 | Δ | √ | √ | √ | — | — |
| Video 0 module status register | V0MS | R | H'E6EF_0004 | H'0000_0018 | 32 | — | √ | √ | √ | — | — |
| Video 0 frame capture register | V0FC | R/W | H'E6EF_0008 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 0 start line pre-clip register | V0SLPrC | R/W | H'E6EF_000C | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 end line pre-clip register | V0ELPrC | R/W | H'E6EF_0010 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 start pixel pre-clip register | V0SPPrC | R/W | H'E6EF_0014 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 end pixel pre-clip register | V0EPPrC | R/W | H'E6EF_0018 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 CSI2 interface mode register | V0CSI_IFMD | R/W | H'E6EF_0020 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 image stride register | V0IS | R/W | H'E6EF_002C | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 memory base 1 register | V0MB1 | R/W | H'E6EF_0030 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 memory base 2 register | V0MB2 | R/W | H'E6EF_0034 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 memory base 3 register | V0MB3 | R/W | H'E6EF_0038 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 line count register | V0LC | R | H'E6EF_003C | H'0000_0000 | 32 | — | √ | √ | √ | — | — |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---|-------------------|-----|-------------|---------------|-------------|-------------------------------|--------|-------------|-------------|--------|--------|
| Video 0 interrupt enable register | VOIE | R/W | H'E6EF_0040 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 0 interrupt status register | VOINTS | R/W | H'E6EF_0044 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 0 scanline interrupt register | VOSI | R/W | H'E6EF_0048 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 data mode register | V0DMR | R/W | H'E6EF_0058 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 data mode register 2 | V0DMR2 | R/W | H'E6EF_005C | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 0 UV address offset register | V0UVAOF | R/W | H'E6EF_0060 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 color space change coefficient 1 register | V0CSCC1 | R/W | H'E6EF_0064 | H'0129_1080 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 color space change coefficient 2 register | V0CSCC2 | R/W | H'E6EF_0068 | H'0198_00D0 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 color space change coefficient 3 register | V0CSCC3 | R/W | H'E6EF_006C | H'0064_0204 | 32 | Supported | √ | √ | √ | — | — |
| Video 0 scaling control register | V0UDS_CTRL | R/W | H'E6EF_0080 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 0 scaling factor register | V0UDS_SCALE | R/W | H'E6EF_0084 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 0 passband register | V0UDS_PASS_BWIDTH | R/W | H'E6EF_0090 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 0 UDS output size clipping register | V0UDS_CLIP_SIZE | R/W | H'E6EF_00A4 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 0 lookup table pointer | V0LUTP | R/W | H'E6EF_0100 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 0 lookup table data register | V0LUTD | R/W | H'E6EF_0104 | H'00xx_xxxx | 32 | — | √ | √ | √ | — | — |
| Video 0 RGB→Y calculation setting register 1 | V0YCCR1 | R/W | H'E6EF_0228 | H'0000_0107 | 32 | Supported | √ | √ | √ | — | — |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | Second Generation RZ/G Series Products | | | | |
|---|----------|-----|-------------|---------------|-------------|-------------------------------|--|-------------|-------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Video 0 RGB→Y calculation setting register 2 | V0YCCR2 | R/W | H'E6EF_022C | H'0064_0204 | 32 | Supported | √ | √ | √ | — | |
| Video 0 RGB→Y calculation setting register 3 | V0YCCR3 | R/W | H'E6EF_0230 | H'0A00_0010 | 32 | Supported | √ | √ | √ | — | |
| Video 0 RGB→Cb calculation setting register 1 | V0CBCCR1 | R/W | H'E6EF_0234 | H'0000_1F68 | 32 | Supported | √ | √ | √ | — | |
| Video 0 RGB→Cb calculation setting register 2 | V0CBCCR2 | R/W | H'E6EF_0238 | H'01C2_1ED6 | 32 | Supported | √ | √ | √ | — | |
| Video 0 RGB→Cb calculation setting register 3 | V0CBCCR3 | R/W | H'E6EF_023C | H'0A00_0080 | 32 | Supported | √ | √ | √ | — | |
| Video 0 RGB→Cr calculation setting register 1 | V0CRCCR1 | R/W | H'E6EF_0240 | H'0000_01C2 | 32 | Supported | √ | √ | √ | — | |
| Video 0 RGB→Cr calculation setting register 2 | V0CRCCR2 | R/W | H'E6EF_0244 | H'1FB7_1E87 | 32 | Supported | √ | √ | √ | — | |
| Video 0 RGB→Cr calculation setting register 3 | V0CRCCR3 | R/W | H'E6EF_0248 | H'0A00_0080 | 32 | Supported | √ | √ | √ | — | |
| Video 0 YC→RGB calculation setting register 1 | V0CSCE1 | R/W | H'E6EF0300 | H'0000_129F | 32 | Supported | √ | √ | √ | — | |
| Video 0 YC→RGB calculation setting register 2 | V0CSCE2 | R/W | H'E6EF_0304 | H'0100_0800 | 32 | Supported | √ | √ | √ | — | |
| Video 0 YC→RGB calculation setting register 3 | V0CSCE3 | R/W | H'E6EF_0308 | H'1989_0D02 | 32 | Supported | √ | √ | √ | — | |
| Video 0 YC→RGB calculation setting register 4 | V0CSCE4 | R/W | H'E6EF_030C | H'0645_2045 | 32 | Supported | √ | √ | √ | — | |
| Video 0 field toggle minimum register | V0FTMIN | R/W | H'E6EF_0324 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 0 field toggle maximum register | V0FTMAX | R/W | H'E6EF_0328 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 0 hsync toggle minimum register | V0HSTMIN | R/W | H'E6EF_032C | H'0000_0000 | 32 | — | √ | — | √ | — | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | Second Generation RZ/G Series Products | | | | |
|--|----------|-----|-------------|---------------|-------------|-------------------------------|--|-------------|-------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Video 0 hsync toggle maximum register | V0HSTMAX | R/W | H'E6EF_0330 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 0 vsync toggle minimum register | V0VSTMIN | R/W | H'E6EF_0334 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 0 vsync toggle maximum register | V0VSTMAX | R/W | H'E6EF_0338 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 0 clkenb enable minimum Register | V0DEMIN | R/W | H'E6EF_0344 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 0 clkenb enable maximum register | V0DEMAX | R/W | H'E6EF_0348 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 0 frame counter register | V0FRCT | R | H'E6EF_0350 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 0 CRC code register 1 | V0CRC1 | R | H'E6EF_0354 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 0 CRC code register 2 | V0CRC2 | R | H'E6EF_0358 | H'0000_0000 | 32 | — | √ | — | √ | — | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---------------------------------------|------------|-----|-------------|---------------|-------------|-------------------------------|--------|-------------|-------------|--------|--------|
| Video 4 main control register | V4MC | R/W | H'E6EF_4000 | H'0000_0000 | 32 | Δ | √ | √ | √ | √ | √ |
| Video 4 module status register | V4MS | R | H'E6EF_4004 | H'0000_0018 | 32 | — | √ | √ | √ | √ | √ |
| Video 4 frame capture register | V4FC | R/W | H'E6EF_4008 | H'0000_0000 | 32 | — | √ | √ | √ | √ | √ |
| Video 4 start line pre-clip register | V4SLPrC | R/W | H'E6EF_400C | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | √ |
| Video 4 end line pre-clip register | V4ELPrC | R/W | H'E6EF_4010 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | √ |
| Video 4 start pixel pre-clip register | V4SPPrC | R/W | H'E6EF_4014 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | √ |
| Video 4 end pixel pre-clip register | V4EPPrC | R/W | H'E6EF_4018 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | √ |
| Video 4 CSI2 interface mode register | V4CSI_IFMD | R/W | H'E6EF_4020 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | √ |
| Video 4 image stride register | V4IS | R/W | H'E6EF_402C | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | √ |
| Video 4 memory base 1 register | V4MB1 | R/W | H'E6EF_4030 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | √ |
| Video 4 memory base 2 register | V4MB2 | R/W | H'E6EF_4034 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | √ |
| Video 4 memory base 3 register | V4MB3 | R/W | H'E6EF_4038 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | √ |
| Video 4 line count register | V4LC | R | H'E6EF_403C | H'0000_0000 | 32 | — | √ | √ | √ | √ | √ |
| Video 4 interrupt enable register | V4IE | R/W | H'E6EF_4040 | H'0000_0000 | 32 | — | √ | √ | √ | √ | √ |
| Video 4 interrupt status register | V4INTS | R/W | H'E6EF_4044 | H'0000_0000 | 32 | — | √ | √ | √ | √ | √ |
| Video 4 scanline interrupt register | V4SI | R/W | H'E6EF_4048 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | √ |
| Video 4 data mode register | V4DMR | R/W | H'E6EF_4058 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | √ |
| Video 4 data mode register 2 | V4DMR2 | R/W | H'E6EF_405C | H'0000_0000 | 32 | — | √ | √ | √ | √ | √ |
| Video 4 UV address offset register | V4UVAOF | R/W | H'E6EF_4060 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---|-------------------|-----|-------------|---------------|-------------|-------------------------------|--------|-------------|-------------|--------|--------|
| Video 4 color space change coefficient 1 register | V4CSCC1 | R/W | H'E6EF_4064 | H'0129_1080 | 32 | Supported | √ | √ | √ | √ | |
| Video 4 color space change coefficient 2 register | V4CSCC2 | R/W | H'E6EF_4068 | H'0198_00D0 | 32 | Supported | √ | √ | √ | √ | |
| Video 4 color space change coefficient 3 register | V4CSCC3 | R/W | H'E6EF_406C | H'0064_0204 | 32 | Supported | √ | √ | √ | √ | |
| Video 4 scaling control register | V4UDS_CTRL | R/W | H'E6EF_4080 | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 4 scaling factor register | V4UDS_SCALE | R/W | H'E6EF_4084 | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 4 passband register | V4UDS_PASS_BWIDTH | R/W | H'E6EF_4090 | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 4 UDS output size clipping register | V4UDS_CLIP_SIZE | R/W | H'E6EF_40A4 | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 4 lookup table pointer | V4LUTP | R/W | H'E6EF_4100 | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 4 lookup table data register | V4LUTD | R/W | H'E6EF_4104 | H'00xx_xxxx | 32 | — | √ | √ | √ | √ | |
| Video 4 RGB→Y calculation setting register 1 | V4YCCR1 | R/W | H'E6EF_4228 | H'0000_0107 | 32 | Supported | √ | √ | √ | √ | |
| Video 4 RGB→Y calculation setting register 2 | V4YCCR2 | R/W | H'E6EF_422C | H'0064_0204 | 32 | Supported | √ | √ | √ | √ | |
| Video 4 RGB→Y calculation setting register 3 | V4YCCR3 | R/W | H'E6EF_4230 | H'0A00_0010 | 32 | Supported | √ | √ | √ | √ | |
| Video 4 RGB→Cb calculation setting register 1 | V4CBCCR1 | R/W | H'E6EF_4234 | H'0000_1F68 | 32 | Supported | √ | √ | √ | √ | |
| Video 4 RGB→Cb calculation setting register 2 | V4CBCCR2 | R/W | H'E6EF_4238 | H'01C2_1ED6 | 32 | Supported | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---|-----------|-----|-------------|---------------|-------------|-------------------------------|--------|-------------|-------------|--------|--------|
| Video 4 RGB→Cb calculation setting register 3 | V4CBCCR3 | R/W | H'E6EF_423C | H'0A00_0080 | 32 | Supported | √ | √ | √ | √ | |
| Video 4 RGB→Cr calculation setting register 1 | V4CRCC R1 | R/W | H'E6EF_4240 | H'0000_01C2 | 32 | Supported | √ | √ | √ | √ | |
| Video 4 RGB→Cr calculation setting register 2 | V4CRCC R2 | R/W | H'E6EF_4244 | H'1FB7_1E87 | 32 | Supported | √ | √ | √ | √ | |
| Video 4 RGB→Cr calculation setting register 3 | V4CRCC R3 | R/W | H'E6EF_4248 | H'0A00_0080 | 32 | Supported | √ | √ | √ | √ | |
| Video 4 YC→RGB calculation setting register 1 | V4CSCE1 | R/W | H'E6EF_4300 | H'0000_129F | 32 | Supported | √ | √ | √ | √ | |
| Video 4 YC→RGB calculation setting register 2 | V4CSCE2 | R/W | H'E6EF_4304 | H'0100_0800 | 32 | Supported | √ | √ | √ | √ | |
| Video 4 YC→RGB calculation setting register 3 | V4CSCE3 | R/W | H'E6EF_4308 | H'1989_0D02 | 32 | Supported | √ | √ | √ | √ | |
| Video 4 YC→RGB calculation setting register 4 | V4CSCE4 | R/W | H'E6EF_430C | H'0645_2045 | 32 | Supported | √ | √ | √ | √ | |
| Video 4 field toggle minimum register | V4FTMIN | R/W | H'E6EF_4324 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 4 field toggle maximum register | V4FTMAX | R/W | H'E6EF_4328 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 4 hsync toggle minimum register | V4HSTMI N | R/W | H'E6EF_432C | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 4 hsync toggle maximum register | V4HSTMA X | R/W | H'E6EF_4330 | H'0000_0000 | 32 | — | √ | — | √ | √ | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--|--------------|-----|-------------|---------------|-------------|-------------------------------|--------|-------------|-------------|--------|--------|
| Video 4 vsync toggle minimum register | V4VSTMI N | R/W | H'E6EF_4334 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 4 vsync toggle maximum register | V4VSTMA X | R/W | H'E6EF_4338 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 4 clkenb enable minimum Register | V4DEMIN | R/W | H'E6EF_4344 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 4 clkenb enable maximum register | V4DEMAX | R/W | H'E6EF_4348 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 4 digital pin toggle register | V4DPTGL | R/W | H'E6EF_434C | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 4 frame counter register | V4FRCT | R | H'E6EF_4350 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 4 CRC code register 1 | V4CRC1 | R | H'E6EF_4354 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 4 CRC code register 2 | V4CRC2 | R | H'E6EF_4358 | H'0000_0000 | 32 | — | √ | — | √ | √ | |

(2) Channels 1 and 5

Table 30.11 VIN1 and VIN5 Registers

Channels 1 and 5 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), Channels 5 (RZ/G2E)

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | Second Generation RZ/G Series Products | | | | |
|---------------------------------------|---------|-----|-------------|---------------|-------------|-------------------------------|--|-------------|-------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Video 1 main control register | V1MC | R/W | H'E6EF_1000 | H'0000_0000 | 32 | Δ | √ | √ | √ | — | |
| Video 1 module status register | V1MS | R | H'E6EF_1004 | H'0000_0018 | 32 | — | √ | √ | √ | — | |
| Video 1 frame capture register | V1FC | R/W | H'E6EF_1008 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 1 start line pre-clip register | V1SLPrC | R/W | H'E6EF_100C | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 1 end line pre-clip register | V1ELPrC | R/W | H'E6EF_1010 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 1 start pixel pre-clip register | V1SPPrC | R/W | H'E6EF_1014 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 1 end pixel pre-clip register | V1EPPrC | R/W | H'E6EF_1018 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 1 image stride register | V1IS | R/W | H'E6EF_102C | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 1 memory base 1 register | V1MB1 | R/W | H'E6EF_1030 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 1 memory base 2 register | V1MB2 | R/W | H'E6EF_1034 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 1 memory base 3 register | V1MB3 | R/W | H'E6EF_1038 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 1 line count register | V1LC | R | H'E6EF_103C | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 1 interrupt enable register | V1IE | R/W | H'E6EF_1040 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 1 interrupt status register | V1INTS | R/W | H'E6EF_1044 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 1 scanline interrupt register | V1SI | R/W | H'E6EF_1048 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 1 data mode register | V1DMR | R/W | H'E6EF_1058 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 1 data mode register 2 | V1DMR2 | R/W | H'E6EF_105C | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 1 UV address offset register | V1UVAOF | R/W | H'E6EF_1060 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---|-----------------------|-----|-------------|---------------|-------------|-------------------------------|--------|-------------|-------------|--------|--------|
| Video 1 color space change coefficient 1 register | V1CSCC1 | R/W | H'E6EF_1064 | H'0129_1080 | 32 | Supported | √ | √ | √ | — | |
| Video 1 color space change coefficient 2 register | V1CSCC2 | R/W | H'E6EF_1068 | H'0198_00D0 | 32 | Supported | √ | √ | √ | — | |
| Video 1 color space change coefficient 3 register | V1CSCC3 | R/W | H'E6EF_106C | H'0064_0204 | 32 | Supported | √ | √ | √ | — | |
| Video 1 scaling control register | V1UDS_CT RL | R/W | H'E6EF_1080 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 1 scaling factor register | V1UDS_SC ALE | R/W | H'E6EF_1084 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 1 passband register | V1UDS_PA SS_BWIDTH | R/W | H'E6EF_1090 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 1 UDS output size clipping register | V1UDS_CLI P_SIZE | R/W | H'E6EF_10A4 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 1 lookup table pointer | V1LUTP | R/W | H'E6EF_1100 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 1 lookup table data register | V1LUTD | R/W | H'E6EF_1104 | H'00xx_xxxx | 32 | — | √ | √ | √ | — | |
| Video 1 RGB→Y calculation setting register 1 | V1YCCR1 | R/W | H'E6EF_1228 | H'0000_0107 | 32 | Supported | √ | √ | √ | — | |
| Video 1 RGB→Y calculation setting register 2 | V1YCCR2 | R/W | H'E6EF_122C | H'0064_0204 | 32 | Supported | √ | √ | √ | — | |
| Video 1 RGB→Y calculation setting register 3 | V1YCCR3 | R/W | H'E6EF_1230 | H'0A00_0010 | 32 | Supported | √ | √ | √ | — | |
| Video 1 RGB→Cb calculation setting register 1 | V1CBCCR1 | R/W | H'E6EF_1234 | H'0000_1F68 | 32 | Supported | √ | √ | √ | — | |
| Video 1 RGB→Cb calculation setting register 2 | V1CBCCR2 | R/W | H'E6EF_1238 | H'01C2_1ED6 | 32 | Supported | √ | √ | √ | — | |
| Video 1 RGB→Cb calculation setting register 3 | V1CBCCR3 | R/W | H'E6EF_123C | H'0A00_0080 | 32 | Supported | √ | √ | √ | — | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | Second Generation RZ/G Series Products | | | | |
|---|----------|-----|-------------|---------------|-------------|-------------------------------|--|-------------|-------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Video 1 RGB→Cr calculation setting register 1 | V1CRCCR1 | R/W | H'E6EF_1240 | H'0000_01C2 | 32 | Supported | √ | √ | √ | — | |
| Video 1 RGB→Cr calculation setting register 2 | V1CRCCR2 | R/W | H'E6EF_1244 | H'1FB7_1E87 | 32 | Supported | √ | √ | √ | — | |
| Video 1 RGB→Cr calculation setting register 3 | V1CRCCR3 | R/W | H'E6EF_1248 | H'0A00_0080 | 32 | Supported | √ | √ | √ | — | |
| Video 1 YC→RGB calculation setting register 1 | V1CSCE1 | R/W | H'E6EF_1300 | H'0000_129F | 32 | Supported | √ | √ | √ | — | |
| Video 1 YC→RGB calculation setting register 2 | V1CSCE2 | R/W | H'E6EF_1304 | H'0100_0800 | 32 | Supported | √ | √ | √ | — | |
| Video 1 YC→RGB calculation setting register 3 | V1CSCE3 | R/W | H'E6EF_1308 | H'1989_0D02 | 32 | Supported | √ | √ | √ | — | |
| Video 1 YC→RGB calculation setting register 4 | V1CSCE4 | R/W | H'E6EF_130C | H'0645_2045 | 32 | Supported | √ | √ | √ | — | |
| Video 1 field toggle minimum register | V1FTMIN | R/W | H'E6EF_1324 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 1 field toggle maximum register | V1FTMAX | R/W | H'E6EF_1328 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 1 hsync toggle minimum register | V1HSTMIN | R/W | H'E6EF_132C | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 1 hsync toggle maximum register | V1HSTMAX | R/W | H'E6EF_1330 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 1 vsync toggle minimum register | V1VSTMIN | R/W | H'E6EF_1334 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 1 vsync toggle maximum register | V1VSTMAX | R/W | H'E6EF_1338 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 1 clkenb enable minimum Register | V1DEMIN | R/W | H'E6EF_1344 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 1 clkenb enable maximum register | V1DEMAX | R/W | H'E6EF_1348 | H'0000_0000 | 32 | — | √ | — | √ | — | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------------------------------|--------|-----|-------------|---------------|-------------|-------------------------------|--------|-------------|-------------|--------|--------|
| Video 1 frame counter register | V1FRCT | R | H'E6EF_1350 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 1 CRC code register 1 | V1CRC1 | R | H'E6EF_1354 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 1 CRC code register 2 | V1CRC2 | R | H'E6EF_1358 | H'0000_0000 | 32 | — | √ | — | √ | — | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | Second Generation RZ/G Series Products | | | | |
|---|---------|-----|-------------|---------------|-------------|-------------------------------|--|-------------|-------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Video 5 main control register | V5MC | R/W | H'E6EF_5000 | H'0000_0000 | 32 | Δ | √ | √ | √ | √ | |
| Video 5 module status register | V5MS | R | H'E6EF_5004 | H'0000_0018 | 32 | — | √ | √ | √ | √ | |
| Video 5 frame capture register | V5FC | R/W | H'E6EF_5008 | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 5 start line pre-clip register | V5SLPrC | R/W | H'E6EF_500C | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 end line pre-clip register | V5ELPrC | R/W | H'E6EF_5010 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 start pixel pre-clip register | V5SPPrC | R/W | H'E6EF_5014 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 end pixel pre-clip register | V5EPPrC | R/W | H'E6EF_5018 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 image stride register | V5IS | R/W | H'E6EF_502C | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 memory base 1 register | V5MB1 | R/W | H'E6EF_5030 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 memory base 2 register | V5MB2 | R/W | H'E6EF_5034 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 memory base 3 register | V5MB3 | R/W | H'E6EF_5038 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 line count register | V5LC | R | H'E6EF_503C | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 5 interrupt enable register | V5IE | R/W | H'E6EF_5040 | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 5 interrupt status register | V5INTS | R/W | H'E6EF_5044 | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 5 scanline interrupt register | V5SI | R/W | H'E6EF_5048 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 data mode register | V5DMR | R/W | H'E6EF_5058 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 data mode register 2 | V5DMR2 | R/W | H'E6EF_505C | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 5 UV address offset register | V5UVAOF | R/W | H'E6EF_5060 | H'0000_0000 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 color space change coefficient 1 register | V5SCCC1 | R/W | H'E6EF_5064 | H'0129_1080 | 32 | Supported | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | Second Generation RZ/G Series Products | | | | |
|---|-----------------------|-----|-------------|---------------|-------------|-------------------------------|--|-------------|-------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Video 5 color space change coefficient 2 register | V5SCCC2 | R/W | H'E6EF_5068 | H'0198_00D0 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 color space change coefficient 3 register | V5SCCC3 | R/W | H'E6EF_506C | H'0064_0204 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 scaling control register | V5UDS_CT RL | R/W | H'E6EF_5080 | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 5 scaling factor register | V5UDS_SC ALE | R/W | H'E6EF_5084 | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 5 passband register | V5UDS_PAS S_BWIDTH | R/W | H'E6EF_5090 | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 5 UDS output size clipping register | V5UDS_CLI P_SIZE | R/W | H'E6EF_50A4 | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 5 lookup table pointer | V5LUTP | R/W | H'E6EF_5100 | H'0000_0000 | 32 | — | √ | √ | √ | √ | |
| Video 5 lookup table data register | V5LUTD | R/W | H'E6EF_5104 | H'00xx_xxxx | 32 | — | √ | √ | √ | √ | |
| Video 5 RGB→Y calculation setting register 1 | V5YCCR1 | R/W | H'E6EF_5228 | H'0000_0107 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 RGB→Y calculation setting register 2 | V5YCCR2 | R/W | H'E6EF_522C | H'0064_0204 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 RGB→Y calculation setting register 3 | V5YCCR3 | R/W | H'E6EF_5230 | H'0A00_0010 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 RGB→Cb calculation setting register 1 | V5CBCCR1 | R/W | H'E6EF_5234 | H'0000_1F68 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 RGB→Cb calculation setting register 2 | V5CBCCR2 | R/W | H'E6EF_5238 | H'01C2_1ED6 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 RGB→Cb calculation setting register 3 | V5CBCCR3 | R/W | H'E6EF_523C | H'0A00_0080 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 RGB→Cr calculation setting register 1 | V5CRCCR1 | R/W | H'E6EF_5240 | H'0000_01C2 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 RGB→Cr calculation setting register 2 | V5CRCCR2 | R/W | H'E6EF_5244 | H'1FB7_1E87 | 32 | Supported | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | Second Generation RZ/G Series Products | | | | |
|---|----------|-----|-------------|---------------|-------------|-------------------------------|--|-------------|-------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Video 5 RGB→Cr calculation setting register 3 | V5CRCCR3 | R/W | H'E6EF_5248 | H'0A00_0080 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 YC→RGB calculation setting register 1 | V5CSCE1 | R/W | H'E6EF_5300 | H'0000_129F | 32 | Supported | √ | √ | √ | √ | |
| Video 5 YC→RGB calculation setting register 2 | V5CSCE2 | R/W | H'E6EF_5304 | H'0100_0800 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 YC→RGB calculation setting register 3 | V5CSCE3 | R/W | H'E6EF_5308 | H'1989_0D02 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 YC→RGB calculation setting register 4 | V5CSCE4 | R/W | H'E6EF_530C | H'0645_2045 | 32 | Supported | √ | √ | √ | √ | |
| Video 5 field toggle minimum register | V5FTMIN | R/W | H'E6EF_5324 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 5 field toggle maximum register | V5FTMAX | R/W | H'E6EF_5328 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 5 hsync toggle minimum register | V5HSTMIN | R/W | H'E6EF_532C | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 5 hsync toggle maximum register | V5HSTMAX | R/W | H'E6EF_5330 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 5 vsync toggle minimum register | V5VSTMIN | R/W | H'E6EF_5334 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 5 vsync toggle maximum register | V5VSTMAX | R/W | H'E6EF_5338 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 5 clkenb enable minimum Register | V5DEMIN | R/W | H'E6EF_5344 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 5 clkenb enable maximum register | V5DEMAX | R/W | H'E6EF_5348 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 5 digital pin toggle register | V5DPTGL | R/W | H'E6EF_534C | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 5 frame counter register | V5FRCT | R | H'E6EF_5350 | H'0000_0000 | 32 | — | √ | — | √ | √ | |
| Video 5 CRC code register 1 | V5CRC1 | R | H'E6EF_5354 | H'0000_0000 | 32 | — | √ | — | √ | √ | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|-----------------------------|--------|-----|-------------|---------------|-------------|-------------------------------|--------|-------------|-------------|--------|--------|
| Video 5 CRC code register 2 | V5CRC2 | R | H'E6EF_5358 | H'0000_0000 | 32 | — | √ | — | √ | √ | |

(3) Channels 2 and 6

Table 30.12 VIN2 and VIN6 Registers

Channels 2 and 6 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N)

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | Second Generation RZ/G Series Products | | | | |
|---------------------------------------|---------|-----|-------------|---------------|-------------|-------------------------------|--|-------------|-------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Video 2 main control register | V2MC | R/W | H'E6EF_2000 | H'0000_0000 | 32 | Δ | √ | √ | √ | — | |
| Video 2 module status register | V2MS | R | H'E6EF_2004 | H'0000_0018 | 32 | — | √ | √ | √ | — | |
| Video 2 frame capture register | V2FC | R/W | H'E6EF_2008 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 2 start line pre-clip register | V2SLPrC | R/W | H'E6EF_200C | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 2 end line pre-clip register | V2ELPrC | R/W | H'E6EF_2010 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 2 start pixel pre-clip register | V2SPPrC | R/W | H'E6EF_2014 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 2 end pixel pre-clip register | V2EPPrC | R/W | H'E6EF_2018 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 2 image stride register | V2IS | R/W | H'E6EF_202C | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 2 memory base 1 register | V2MB1 | R/W | H'E6EF_2030 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 2 memory base 2 register | V2MB2 | R/W | H'E6EF_2034 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 2 memory base 3 register | V2MB3 | R/W | H'E6EF_2038 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 2 line count register | V2LC | R | H'E6EF_203C | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 2 interrupt enable register | V2IE | R/W | H'E6EF_2040 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 2 interrupt status register | V2INTS | R/W | H'E6EF_2044 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 2 scanline interrupt register | V2SI | R/W | H'E6EF_2048 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 2 data mode register | V2DMR | R/W | H'E6EF_2058 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 2 data mode register 2 | V2DMR2 | R/W | H'E6EF_205C | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 2 UV address offset register | V2UVAOF | R/W | H'E6EF_2060 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | Second Generation RZ/G Series Products | | | | |
|---|-----------|-----|-------------|---------------|-------------|-------------------------------|--|-------------|-------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Video 2 color space change coefficient 1 register | V2CSCC1 | R/W | H'E6EF_2064 | H'0129_1080 | 32 | Supported | √ | √ | √ | — | |
| Video 2 color space change coefficient 2 register | V2CSCC2 | R/W | H'E6EF_2068 | H'0198_00D0 | 32 | Supported | √ | √ | √ | — | |
| Video 2 color space change coefficient 3 register | V2CSCC3 | R/W | H'E6EF_206C | H'0064_0204 | 32 | Supported | √ | √ | √ | — | |
| Video 2 lookup table pointer | V2LUTP | R/W | H'E6EF_2100 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 2 lookup table data register | V2LUTD | R/W | H'E6EF_2104 | H'00xx_xxxx | 32 | — | √ | √ | √ | — | |
| Video 2 RGB→Y calculation setting register 1 | V2YCCR1 | R/W | H'E6EF_2228 | H'0000_0107 | 32 | Supported | √ | √ | √ | — | |
| Video 2 RGB→Y calculation setting register 2 | V2YCCR2 | R/W | H'E6EF_222C | H'0064_0204 | 32 | Supported | √ | √ | √ | — | |
| Video 2 RGB→Y calculation setting register 3 | V2YCCR3 | R/W | H'E6EF_2230 | H'0A00_0010 | 32 | Supported | √ | √ | √ | — | |
| Video 2 RGB→Cb calculation setting register 1 | V2CBCCR1 | R/W | H'E6EF_2234 | H'0000_1F68 | 32 | Supported | √ | √ | √ | — | |
| Video 2 RGB→Cb calculation setting register 2 | V2CBCCR2 | R/W | H'E6EF_2238 | H'01C2_1ED6 | 32 | Supported | √ | √ | √ | — | |
| Video 2 RGB→Cb calculation setting register 3 | V2CBCCR3 | R/W | H'E6EF_223C | H'0A00_0080 | 32 | Supported | √ | √ | √ | — | |
| Video 2 RGB→Cr calculation setting register 1 | V2CRCCR 1 | R/W | H'E6EF_2240 | H'0000_01C2 | 32 | Supported | √ | √ | √ | — | |
| Video 2 RGB→Cr calculation setting register 2 | V2CRCCR 2 | R/W | H'E6EF_2244 | H'1FB7_1E87 | 32 | Supported | √ | √ | √ | — | |
| Video 2 RGB→Cr calculation setting register 3 | V2CRCCR 3 | R/W | H'E6EF_2248 | H'0A00_0080 | 32 | Supported | √ | √ | √ | — | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---|----------|-----|-------------|---------------|-------------|-------------------------------|--------|-------------|-------------|--------|--------|
| Video 2 YC→RGB calculation setting register 1 | V2CSCE1 | R/W | H'E6EF_2300 | H'0000_129F | 32 | Supported | √ | √ | √ | — | |
| Video 2 YC→RGB calculation setting register 2 | V2CSCE2 | R/W | H'E6EF_2304 | H'0100_0800 | 32 | Supported | √ | √ | √ | — | |
| Video 2 YC→RGB calculation setting register 3 | V2CSCE3 | R/W | H'E6EF_2308 | H'1989_0D02 | 32 | Supported | √ | √ | √ | — | |
| Video 2 YC→RGB calculation setting register 4 | V2CSCE4 | R/W | H'E6EF_230C | H'0645_2045 | 32 | Supported | √ | √ | √ | — | |
| Video 2 field toggle minimum register | V2FTMIN | R/W | H'E6EF_2324 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 2 field toggle maximum register | V2FTMAX | R/W | H'E6EF_2328 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 2 hsync toggle minimum register | V2HSTMIN | R/W | H'E6EF_232C | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 2 hsync toggle maximum register | V2HSTMAX | R/W | H'E6EF_2330 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 2 vsync toggle minimum register | V2VSTMIN | R/W | H'E6EF_2334 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 2 vsync toggle maximum register | V2VSTMAX | R/W | H'E6EF_2338 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 2 clkenb enable minimum Register | V2DEMIN | R/W | H'E6EF_2344 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 2 clkenb enable maximum register | V2DEMAX | R/W | H'E6EF_2348 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 2 frame counter register | V2FRCT | R | H'E6EF_2350 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 2 CRC code register 1 | V2CRC1 | R | H'E6EF_2354 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 2 CRC code register 2 | V2CRC2 | R | H'E6EF_2358 | H'0000_0000 | 32 | — | √ | — | √ | — | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---|---------|-----|-------------|---------------|-------------|-------------------------------|--------|-------------|-------------|--------|--------|
| Video 6 main control register | V6MC | R/W | H'E6EF_6000 | H'0000_0000 | 32 | Δ | √ | √ | √ | — | — |
| Video 6 module status register | V6MS | R | H'E6EF_6004 | H'0000_0018 | 32 | — | √ | √ | √ | — | — |
| Video 6 frame capture register | V6FC | R/W | H'E6EF_6008 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 6 start line pre-clip register | V6SLPrC | R/W | H'E6EF_600C | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 6 end line pre-clip register | V6ELPrC | R/W | H'E6EF_6010 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 6 start pixel pre-clip register | V6SPPrC | R/W | H'E6EF_6014 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 6 end pixel pre-clip register | V6EPPrC | R/W | H'E6EF_6018 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 6 image stride register | V6IS | R/W | H'E6EF_602C | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 6 memory base 1 register | V6MB1 | R/W | H'E6EF_6030 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 6 memory base 2 register | V6MB2 | R/W | H'E6EF_6034 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 6 memory base 3 register | V6MB3 | R/W | H'E6EF_6038 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 6 line count register | V6LC | R | H'E6EF_603C | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 6 interrupt enable register | V6IE | R/W | H'E6EF_6040 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 6 interrupt status register | V6INTS | R/W | H'E6EF_6044 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 6 scanline interrupt register | V6SI | R/W | H'E6EF_6048 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 6 data mode register | V6DMR | R/W | H'E6EF_6058 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 6 data mode register 2 | V6DMR2 | R/W | H'E6EF_605C | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 6 UV address offset register | V6UVAOF | R/W | H'E6EF_6060 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 6 color space change coefficient 1 register | V6SCCC1 | R/W | H'E6EF_6064 | H'0129_1080 | 32 | Supported | √ | √ | √ | — | — |
| Video 6 color space change coefficient 2 register | V6SCCC2 | R/W | H'E6EF_6068 | H'0198_00D0 | 32 | Supported | √ | √ | √ | — | — |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | Second Generation RZ/G Series Products | | | | |
|---|----------|-----|-------------|---------------|-------------|-------------------------------|--|-------------|-------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Video 6 color space change coefficient 3 register | V6CSCC3 | R/W | H'E6EF_606C | H'0064_0204 | 32 | Supported | √ | √ | √ | — | |
| Video 6 lookup table pointer | V6LUTP | R/W | H'E6EF_6100 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 6 lookup table data register | V6LUTD | R/W | H'E6EF_6104 | H'00xx_xxxx | 32 | — | √ | √ | √ | — | |
| Video 6 RGB→Y calculation setting register 1 | V6YCCR1 | R/W | H'E6EF_6228 | H'0000_0107 | 32 | Supported | √ | √ | √ | — | |
| Video 6 RGB→Y calculation setting register 2 | V6YCCR2 | R/W | H'E6EF_622C | H'0064_0204 | 32 | Supported | √ | √ | √ | — | |
| Video 6 RGB→Y calculation setting register 3 | V6YCCR3 | R/W | H'E6EF_6230 | H'0A00_0010 | 32 | Supported | √ | √ | √ | — | |
| Video 6 RGB→Cb calculation setting register 1 | V6CBCCR1 | R/W | H'E6EF_6234 | H'0000_1F68 | 32 | Supported | √ | √ | √ | — | |
| Video 6 RGB→Cb calculation setting register 2 | V6CBCCR2 | R/W | H'E6EF_6238 | H'01C2_1ED6 | 32 | Supported | √ | √ | √ | — | |
| Video 6 RGB→Cb calculation setting register 3 | V6CBCCR3 | R/W | H'E6EF_623C | H'0A00_0080 | 32 | Supported | √ | √ | √ | — | |
| Video 6 RGB→Cr calculation setting register 1 | V6CRCCR1 | R/W | H'E6EF_6240 | H'0000_01C2 | 32 | Supported | √ | √ | √ | — | |
| Video 6 RGB→Cr calculation setting register 2 | V6CRCCR2 | R/W | H'E6EF_6244 | H'1FB7_1E87 | 32 | Supported | √ | √ | √ | — | |
| Video 6 RGB→Cr calculation setting register 3 | V6CRCCR3 | R/W | H'E6EF_6248 | H'0A00_0080 | 32 | Supported | √ | √ | √ | — | |
| Video 6 YC→RGB calculation setting register 1 | V6CSCE1 | R/W | H'E6EF_6300 | H'0000_129F | 32 | Supported | √ | √ | √ | — | |
| Video 6 YC→RGB calculation setting register 2 | V6CSCE2 | R/W | H'E6EF_6304 | H'0100_0800 | 32 | Supported | √ | √ | √ | — | |
| Video 6 YC→RGB calculation setting register 3 | V6CSCE3 | R/W | H'E6EF_6308 | H'1989_0D02 | 32 | Supported | √ | √ | √ | — | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---|----------|-----|-------------|---------------|-------------|-------------------------------|--------|-------------|-------------|--------|--------|
| Video 6 YC→RGB calculation setting register 4 | V6CSCE4 | R/W | H'E6EF_630C | H'0645_2045 | 32 | Supported | √ | √ | √ | — | — |
| Video 6 field toggle minimum register | V6FTMIN | R/W | H'E6EF_6324 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 6 field toggle maximum register | V6FTMAX | R/W | H'E6EF_6328 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 6 hsync toggle minimum register | V6HSTMIN | R/W | H'E6EF_632C | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 6 hsync toggle maximum register | V6HSTMAX | R/W | H'E6EF_6330 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 6 vsync toggle minimum register | V6VSTMIN | R/W | H'E6EF_6334 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 6 vsync toggle maximum register | V6VSTMAX | R/W | H'E6EF_6338 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 6 clkenb enable minimum Register | V6DEMIN | R/W | H'E6EF_6344 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 6 clkenb enable maximum register | V6DEMAX | R/W | H'E6EF_6348 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 6 frame counter register | V6FRCT | R | H'E6EF_6350 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 6 CRC code register 1 | V6CRC1 | R | H'E6EF_6354 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 6 CRC code register 2 | V6CRC2 | R | H'E6EF_6358 | H'0000_0000 | 32 | — | √ | — | √ | — | — |

(4) Channels 3 and 7

Table 30.13 VIN3 and VIN7 Registers

Channels 3 and 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N)

Second Generation
RZ/G Series Products

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | Second Generation RZ/G Series Products | | | | |
|---------------------------------------|---------|-----|-------------|---------------|-------------|-------------------------------|--|-------------|-------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Video 3 main control register | V3MC | R/W | H'E6EF_3000 | H'0000_0000 | 32 | Δ | √ | √ | √ | — | |
| Video 3 module status register | V3MS | R | H'E6EF_3004 | H'0000_0018 | 32 | — | √ | √ | √ | — | |
| Video 3 frame capture register | V3FC | R/W | H'E6EF_3008 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 3 start line pre-clip register | V3SLPrC | R/W | H'E6EF_300C | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 3 end line pre-clip register | V3ELPrC | R/W | H'E6EF_3010 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 3 start pixel pre-clip register | V3SPPrC | R/W | H'E6EF_3014 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 3 end pixel pre-clip register | V3EPPrC | R/W | H'E6EF_3018 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 3 image stride register | V3IS | R/W | H'E6EF_302C | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 3 memory base 1 register | V3MB1 | R/W | H'E6EF_3030 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 3 memory base 2 register | V3MB2 | R/W | H'E6EF_3034 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 3 memory base 3 register | V3MB3 | R/W | H'E6EF_3038 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 3 line count register | V3LC | R | H'E6EF_303C | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 3 interrupt enable register | V3IE | R/W | H'E6EF_3040 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 3 interrupt status register | V3INTS | R/W | H'E6EF_3044 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 3 scanline interrupt register | V3SI | R/W | H'E6EF_3048 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 3 data mode register | V3DMR | R/W | H'E6EF_3058 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |
| Video 3 data mode register 2 | V3DMR2 | R/W | H'E6EF_305C | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 3 UV address offset register | V3UVAOF | R/W | H'E6EF_3060 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | Second Generation RZ/G Series Products | | | | |
|---|----------|-----|-------------|---------------|-------------|-------------------------------|--|-------------|-------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Video 3 color space change coefficient 1 register | V3CSCC1 | R/W | H'E6EF_3064 | H'0129_1080 | 32 | Supported | √ | √ | √ | — | |
| Video 3 color space change coefficient 2 register | V3CSCC2 | R/W | H'E6EF_3068 | H'0198_00D0 | 32 | Supported | √ | √ | √ | — | |
| Video 3 color space change coefficient 3 register | V3CSCC3 | R/W | H'E6EF_306C | H'0064_0204 | 32 | Supported | √ | √ | √ | — | |
| Video 3 lookup table pointer | V3LUTP | R/W | H'E6EF_3100 | H'0000_0000 | 32 | — | √ | √ | √ | — | |
| Video 3 lookup table data register | V3LUTD | R/W | H'E6EF_3104 | H'00xx_xxxx | 32 | — | √ | √ | √ | — | |
| Video 3 RGB→Y calculation setting register 1 | V3YCCR1 | R/W | H'E6EF_3228 | H'0000_0107 | 32 | Supported | √ | √ | √ | — | |
| Video 3 RGB→Y calculation setting register 2 | V3YCCR2 | R/W | H'E6EF_322C | H'0064_0204 | 32 | Supported | √ | √ | √ | — | |
| Video 3 RGB→Y calculation setting register 3 | V3YCCR3 | R/W | H'E6EF_3230 | H'0A00_0010 | 32 | Supported | √ | √ | √ | — | |
| Video 3 RGB→Cb calculation setting register 1 | V3CBCCR1 | R/W | H'E6EF_3234 | H'0000_1F68 | 32 | Supported | √ | √ | √ | — | |
| Video 3 RGB→Cb calculation setting register 2 | V3CBCCR2 | R/W | H'E6EF_3238 | H'01C2_1ED6 | 32 | Supported | √ | √ | √ | — | |
| Video 3 RGB→Cb calculation setting register 3 | V3CBCCR3 | R/W | H'E6EF_323C | H'0A00_0080 | 32 | Supported | √ | √ | √ | — | |
| Video 3 RGB→Cr calculation setting register 1 | V3CRCCR1 | R/W | H'E6EF_3240 | H'0000_01C2 | 32 | Supported | √ | √ | √ | — | |
| Video 3 RGB→Cr calculation setting register 2 | V3CRCCR2 | R/W | H'E6EF_3244 | H'1FB7_1E87 | 32 | Supported | √ | √ | √ | — | |
| Video 3 RGB→Cr calculation setting register 3 | V3CRCCR3 | R/W | H'E6EF_3248 | H'0A00_0080 | 32 | Supported | √ | √ | √ | — | |
| Video 3 YC→RGB calculation setting register 1 | V3CSCE1 | R/W | H'E6EF_3300 | H'0000_129F | 32 | Supported | √ | √ | √ | — | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | Second Generation RZ/G Series Products | | | | |
|---|----------|-----|-------------|---------------|-------------|-------------------------------|--|-------------|-------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Video 3 YC→RGB calculation setting register 2 | V3CSCE2 | R/W | H'E6EF_3304 | H'0100_0800 | 32 | Supported | √ | √ | √ | — | |
| Video 3 YC→RGB calculation setting register 3 | V3CSCE3 | R/W | H'E6EF_3308 | H'1989_0D02 | 32 | Supported | √ | √ | √ | — | |
| Video 3 YC→RGB calculation setting register 4 | V3CSCE4 | R/W | H'E6EF_330C | H'0645_2045 | 32 | Supported | √ | √ | √ | — | |
| Video 3 field toggle minimum register | V3FTMIN | R/W | H'E6EF_3324 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 3 field toggle maximum register | V3FTMAX | R/W | H'E6EF_3328 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 3 hsync toggle minimum register | V3HSTMIN | R/W | H'E6EF_332C | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 3 hsync toggle maximum register | V3HSTMAX | R/W | H'E6EF_3330 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 3 vsync toggle minimum register | V3VSTMIN | R/W | H'E6EF_3334 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 3 vsync toggle maximum register | V3VSTMAX | R/W | H'E6EF_3338 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 3 clkenb enable minimum Register | V3DEMIN | R/W | H'E6EF_3344 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 3 clkenb enable maximum register | V3DEMAX | R/W | H'E6EF_3348 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 3 frame counter register | V3FRCT | R | H'E6EF_3350 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 3 CRC code register 1 | V3CRC1 | R | H'E6EF_3354 | H'0000_0000 | 32 | — | √ | — | √ | — | |
| Video 3 CRC code register 2 | V3CRC2 | R | H'E6EF_3358 | H'0000_0000 | 32 | — | √ | — | √ | — | |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---|---------|-----|-------------|---------------|-------------|-------------------------------|--------|-------------|-------------|--------|--------|
| Video 7 main control register | V7MC | R/W | H'E6EF_7000 | H'0000_0000 | 32 | Δ | √ | √ | √ | — | — |
| Video 7 module status register | V7MS | R | H'E6EF_7004 | H'0000_0018 | 32 | — | √ | √ | √ | — | — |
| Video 7 frame capture register | V7FC | R/W | H'E6EF_7008 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 7 start line pre-clip register | V7SLPrC | R/W | H'E6EF_700C | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 end line pre-clip register | V7ELPrC | R/W | H'E6EF_7010 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 start pixel pre-clip register | V7SPPrC | R/W | H'E6EF_7014 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 end pixel pre-clip register | V7EPPrC | R/W | H'E6EF_7018 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 image stride register | V7IS | R/W | H'E6EF_702C | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 memory base 1 register | V7MB1 | R/W | H'E6EF_7030 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 memory base 2 register | V7MB2 | R/W | H'E6EF_7034 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 memory base 3 register | V7MB3 | R/W | H'E6EF_7038 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 line count register | V7LC | R | H'E6EF_703C | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 7 interrupt enable register | V7IE | R/W | H'E6EF_7040 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 7 interrupt status register | V7INTS | R/W | H'E6EF_7044 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 7 scanline interrupt register | V7SI | R/W | H'E6EF_7048 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 data mode register | V7DMR | R/W | H'E6EF_7058 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 data mode register 2 | V7DMR2 | R/W | H'E6EF_705C | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 7 UV address offset register | V7UVAOF | R/W | H'E6EF_7060 | H'0000_0000 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 color space change coefficient 1 register | V7CSCC1 | R/W | H'E6EF_7064 | H'0129_1080 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 color space change coefficient 2 register | V7CSCC2 | R/W | H'E6EF_7068 | H'0198_00D0 | 32 | Supported | √ | √ | √ | — | — |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---|----------|-----|-------------|---------------|-------------|-------------------------------|--------|-------------|-------------|--------|--------|
| Video 7 color space change coefficient 3 register | V7CSCC3 | R/W | H'E6EF_706C | H'0064_0204 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 lookup table pointer | V7LUTP | R/W | H'E6EF_7100 | H'0000_0000 | 32 | — | √ | √ | √ | — | — |
| Video 7 lookup table data register | V7LUTD | R/W | H'E6EF_7104 | H'00xx_xxxx | 32 | — | √ | √ | √ | — | — |
| Video 7 RGB→Y calculation setting register 1 | V7YCCR1 | R/W | H'E6EF_7228 | H'0000_0107 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 RGB→Y calculation setting register 2 | V7YCCR2 | R/W | H'E6EF_722C | H'0064_0204 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 RGB→Y calculation setting register 3 | V7YCCR3 | R/W | H'E6EF_7230 | H'0A00_0010 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 RGB→Cb calculation setting register 1 | V7CBCCR1 | R/W | H'E6EF_7234 | H'0000_1F68 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 RGB→Cb calculation setting register 2 | V7CBCCR2 | R/W | H'E6EF_7238 | H'01C2_1ED6 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 RGB→Cb calculation setting register 3 | V7CBCCR3 | R/W | H'E6EF_723C | H'0A00_0080 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 RGB→Cr calculation setting register 1 | V7CRCCR1 | R/W | H'E6EF_7240 | H'0000_01C2 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 RGB→Cr calculation setting register 2 | V7CRCCR2 | R/W | H'E6EF_7244 | H'1FB7_1E87 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 RGB→Cr calculation setting register 3 | V7CRCCR3 | R/W | H'E6EF_7248 | H'0A00_0080 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 YC→RGB calculation setting register 1 | V7CSCE1 | R/W | H'E6EF_7300 | H'0000_129F | 32 | Supported | √ | √ | √ | — | — |
| Video 7 YC→RGB calculation setting register 2 | V7CSCE2 | R/W | H'E6EF_7304 | H'0100_0800 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 YC→RGB calculation setting register 3 | V7CSCE3 | R/W | H'E6EF_7308 | H'1989_0D02 | 32 | Supported | √ | √ | √ | — | — |

**Second Generation
RZ/G Series Products**

| Name | Symbol | R/W | Address | Initial Value | Access Size | Internal Update Mode Support* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---|----------|-----|-------------|---------------|-------------|-------------------------------|--------|-------------|-------------|--------|--------|
| Video 7 YC→RGB calculation setting register 4 | V7CSCE4 | R/W | H'E6EF_730C | H'0645_2045 | 32 | Supported | √ | √ | √ | — | — |
| Video 7 field toggle minimum register | V7FTMIN | R/W | H'E6EF_7324 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 7 field toggle maximum register | V7FTMAX | R/W | H'E6EF_7328 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 7 hsync toggle minimum register | V7HSTMIN | R/W | H'E6EF_732C | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 7 hsync toggle maximum register | V7HSTMAX | R/W | H'E6EF_7330 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 7 vsync toggle minimum register | V7VSTMIN | R/W | H'E6EF_7334 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 7 vsync toggle maximum register | V7VSTMAX | R/W | H'E6EF_7338 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 7 clkenb enable minimum Register | V7DEMIN | R/W | H'E6EF_7344 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 7 clkenb enable maximum register | V7DEMAX | R/W | H'E6EF_7348 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 7 frame counter register | V7FRCT | R | H'E6EF_7350 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 7 CRC code register 1 | V7CRC1 | R | H'E6EF_7354 | H'0000_0000 | 32 | — | √ | — | √ | — | — |
| Video 7 CRC code register 2 | V7CRC2 | R | H'E6EF_7358 | H'0000_0000 | 32 | — | √ | — | √ | — | — |

30.1.5 Connected Module

Table 30.14 Connected Modules

| Connected Modules | Function of the Connected Modules |
|--------------------------|--|
| CSI2 | Input of image data |
| CPG | Input reset and clock stop signal |
| INTC | Interrupt controller |

30.2 Register Description

[Legend]

—: Reserved. The write value should always be 0.

Initial value: Register value after a reset

R/W: Readable/writable. The written value can be read.

R: Read-only. The write value should always be 0.

Vn: Video channel n

30.2.1 Video n Main Control Register (VnMC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:

n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

| | | | | | | | | | | | | | | | | |
|----------------|---------|-----|------------|-------|------|-----|----|----|----|-----|------|---------|----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 30 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | CLP[1:0] | DPINE | SCLE | — | — | — | — | FOC | LUTE | YCAL | INF[2:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DC[1:0] | | EXINF[1:0] | | — | VUP | — | — | — | EN | EC | IM[1:0] | | — | BPS | ME |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R/W | R | R | R | R/W | R/W | R/W | R/W | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | |
|----------|--|---|---------------|---|-----|-----------|------------------|---------------|------|-------------|--|---------------|------|--|--|------|-------------|---|------|-------------|-------------|
| 31 to 30 | — | B'0 | R/W | Reserved This bit is always read as 0. The write value should always be 0. | | | | | | | | | | | | | | | | | |
| 29, 28 | CLP[1:0] | B'00 | R/W | <p>Pixel Data Clipping</p> <p>When the input image data is in the YCbCr format, these bits specify the data clip value for clipping the YCbCr-RGB color conversion input data to the nominal range prescribed in the ITU-R BT.601 standard.</p> <table border="1"> <thead> <tr> <th>CLP</th> <th>Luminance</th> <th>Color Difference</th> <th>Initial value</th> </tr> </thead> <tbody> <tr> <td>b'00</td> <td>No clipping</td> <td>16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240.</td> <td rowspan="4">Initial value</td> </tr> <tr> <td>b'01</td> <td>16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240.</td> <td>16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240.</td> </tr> <tr> <td>b'10</td> <td>No clipping</td> <td>16 or a smaller value is clipped to 128. 240 or a greater value is clipped to 128.</td> </tr> <tr> <td>b'11</td> <td>No clipping</td> <td>No clipping</td> </tr> </tbody> </table> <p>Note: These bits support the internal update mode.</p> | CLP | Luminance | Color Difference | Initial value | b'00 | No clipping | 16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240. | Initial value | b'01 | 16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240. | 16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240. | b'10 | No clipping | 16 or a smaller value is clipped to 128. 240 or a greater value is clipped to 128. | b'11 | No clipping | No clipping |
| CLP | Luminance | Color Difference | Initial value | | | | | | | | | | | | | | | | | | |
| b'00 | No clipping | 16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240. | Initial value | | | | | | | | | | | | | | | | | | |
| b'01 | 16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240. | 16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240. | | | | | | | | | | | | | | | | | | | |
| b'10 | No clipping | 16 or a smaller value is clipped to 128. 240 or a greater value is clipped to 128. | | | | | | | | | | | | | | | | | | | |
| b'11 | No clipping | No clipping | | | | | | | | | | | | | | | | | | | |
| 27 | DPINE | B'0 | R/W | <p>This bit is used to select the source of the field signal.</p> <p>0: MIPI CSI-2 interfaces 1: Digital pins</p> <p>Note: This bit should be set to 0 in other than the below video channels. video channels 4 and 5 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> | | | | | | | | | | | | | | | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 30 | SCLE | B'0 | R/W | <p>This bit is used to enable or disable scaling by the UDS.</p> <p>0: Disables scaling by the UDS. 1: Enables scaling by the UDS.</p> <p>Notes: 1. This bit is only present for video channels 0, 1, 4, and 5. 2. Do not set this bit to 1 for both video channels 0 and 1 at the same time. 3. Do not set this bit to 1 for both video channels 4 and 5 at the same time.</p> |
| 25 to 22 | — | H'0 | R | <p>Reserved</p> <p>This bits are always read as 0. The write value should always be 0.</p> |
| 21 | FOC | B'0 | R/W | <p>Field Order Control</p> <p>This bit controls the field order for full interlace capturing.</p> <p>0: Top field = Odd field (field 1) 1: Top field = Even field (field 2)</p> <p>Note: This bit supports the internal update mode.</p> |
| 20 | LUTE | B'0 | R/W | <p>Lookup Table Enable</p> <p>This bit enables lookup table conversion from 10 bits to 8 bits.</p> <p>0: LUT is not used. 1: LUT is used.</p> <p>Notes: 1. To perform LUT conversion, the conversion table should be set with VnLUTP and VnLUTD. 2. This bit supports the internal update mode.</p> |
| 19 | YCAL | B'0 | R/W | <p>YCbCr-422 Input Data Alignment</p> <p>This bit controls data alignment for YCbCr-422 input.</p> <p>0: When the multiplexed CbCr interface is set, capturing is performed with Y in the upper bits and CbCr in the lower bits. 1: When the multiplexed CbCr interface is set, capturing is performed with CbCr in the upper bits and Y in the lower bits.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 18 to 16 | INF[2:0] | B'000 | R/W | <p>Input Interface Format</p> <p>These bits specify the image format input to the VIN.</p> <p>[Digital Pins]</p> <p>b'000: ITU-R BT.656 8-bit YCbCr-422 input*¹</p> <p>b'001: ITU-R BT.601/BT.709 8-bit YCbCr-422 input*¹</p> <p>b'010: ITU-R BT.656 10/12-bit YCbCr-422 input*¹</p> <p>b'011: ITU-R BT.601/BT.709 10/12-bit YCbCr-422 or RAW12 input*¹</p> <p>b'100: Prohibit</p> <p>b'101: ITU-R BT.601/BT.1358 16-bit YCbCr-422 input*¹</p> <p>b'110: ITU-R BT.601/BT.709 24-bit RGB-888 input*¹</p> <p>b'111: ITU-R BT.601/BT.709 18-bit RGB-666 input*¹</p> <p>[MIPI CSI-2]</p> <p>b'000: Prohibit</p> <p>b'001: YCbCr-422 8-bit data input*¹</p> <p>b'010: Prohibit</p> <p>b'011: YCbCr-422 10-bit data input*¹</p> <p>b'100: 8-bit user defined data (RAW8) or 8-bit embedded input*¹</p> <p>b'101: Prohibit</p> <p>b'110: RGB-888 data input*¹</p> <p>b'111: Prohibit</p> |
| 15, 14 | DC[1:0] | B'00 | R/W | <p>Dithering Mode Control</p> <p>These bits select the dithering mode for conversion from RGB888 to RGB565/ARGB1565.</p> <p>b'00: Dithering with cumulative addition</p> <p>b'01: Ordered dithering</p> <p>b'10: Setting prohibited</p> <p>b'11: Setting prohibited</p> |
| 13, 12 | EXINF[1:0] | B'00 | R/W | <p>Extension Interface Select</p> <p>b'00: Data extension is not performed</p> <p>b'01: Combined with the INF setting, 8-bit data extension is performed.</p> <p>b'10: Combined with the INF setting, 10-bit data extension is performed.</p> <p>b'11: Combined with the INF setting, 12-bit data extension is performed.</p> |
| 11 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 10 | VUP | B'0 | R/W | <p>VIN Register Update Control</p> <p>This bit specifies the internal register update timing after writing register. See the list of registers for applicable registers.</p> <p>0: The register contents are updated immediately after register writing.</p> <p>1: The register contents are updated after a valid field is detected in the ITU-R BT.601 data or the field (F) bit changes in the ITU-R BT.656 data.</p> |
| 9 to 7 | — | B'0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 6 | EN | B'0 | R/W | <p>Endian Type</p> <p>This bit specifies the endian type for data to be output to external memory.</p> <p>0: Image data is packed and allocated in little endian.</p> <p>1: Image data is packed and allocated in big endian.</p> <p>Note: When allocating the YCbCr422 (UYVY format) data in big endian, be sure to set the BPSM bit in VnDMR to 1.</p> |
| 5 | EC | B'0 | R/W | <p>Error Correction Control</p> <p>This bit specifies whether error correction with the parity bit is performed on the ITU-R BT.656 input.</p> <p>0: Error correction is not performed on the ITU-R BT.656 input.</p> <p>1: Error correction with the parity bit is performed on the ITU-R BT.656 input.</p> <p>Error correction must not be performed in the following cases:</p> <ul style="list-style-type: none"> • When data is captured in the ITU-R BT.601 interface • When input data does not meet the standard of the ITU-R BT.656 parity bit |
| 4, 3 | IM[1:0] | B'00 | R/W | <p>Interlace Mode</p> <p>These bits specify the capture mode. Do not modify this setting during capture operation.</p> <p>b'00: Odd-field (field 1) capture mode Handles only odd fields as frames and stores them in external memory.</p> <p>b'01: Odd-/even-field capture mode Handles odd and even fields as separate frames and stores them in external memory. This mode is available only in continuous frame capture mode.</p> <p>b'10: Even-field (field 2) capture mode Handles only even fields as frames and stores them in external memory.</p> <p>b'11: Full interlace mode Handles combinations of odd and even fields as single frames and stores them in external memory.</p> |
| 2 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 1 | BPS | B'0 | R/W | <p>Color Space Conversion Bypass Mode</p> <p>0: The input YCbCr data is converted into the RGB color space and RGB data is converted into the YCbCr color space*2.</p> <p>1: Color space conversion is not performed.</p> <p>Note: YCbCr→RGB or RGB→YCbCr conversion is performed with the coefficients specified by the YC-RGB conversion coefficient register or RGB-YC coefficient register.</p> |
| 0 | ME | B'0 | R/W | <p>Module Enable</p> <p>This is the enable bit for the VIN. Set this bit before setting the Frame Capture (VnFC) register.</p> <p>0: The module operation is stopped.</p> <p>1: The module operation is enabled. *3</p> |

- Notes:
1. Table 30.15 shows the combinations of interfaces which can be set by the input interface format (the INF bits in VnMC) and extension interface select (the EXINF bit in VnMC). Do not make the other settings.
 2. Table 30.16 shows the image data which can be converted according to the color space conversion bypass mode settings (set by the BPS bit in VnMC).
 3. To stop capturing operation, only set the ME bit to 0. Do not change the other bit settings.

Table 30.15 Capture Interface Settings

| Interface | | VnMC/EXINF | VnMC/INF |
|---|---------|------------|----------|
| ITU-R BT.656 (multiplexed YCbCr422) | 8 bits | 00 | 000 |
| | 10 bits | 00 | 010 |
| | 12 bits | 11 | 010 |
| ITU-R BT.601/BT.709 (multiplexed YCbCr422) | 8 bits | 00 | 001 |
| | 10 bits | 00 | 011 |
| | 12 bits | 11 | 011 |
| ITU-R BT.601/BT.709/BT.1358 (non-multiplexed Y/multiplexed CbCr) | 16 bits | 00 | 101 |
| | 20 bits | 10 | 101 |
| | 24 bits | 11 | 101 |
| ITU-R BT.601/BT.709 (RGB666) | 18 bits | 00 | 111 |
| ITU-R BT.601/BT.709 (RGB888) | 24 bits | 00 | 110 |
| MIPI CSI-2 YCbCr-422 8-bit | — | 00 | 001 |
| MIPI CSI-2 YCbCr-422 10-bit | — | 00 | 011 |
| MIPI CSI-2 RGB888 | — | 00 | 110 |
| MIPI CSI-2 8-bit user defined data (RAW8) or 8-bit embedded | — | 00 | 100 |
| RAW12 (Digital Pins Interface Only) | 12 bits | 11 | 011 |

Table 30.16 Captured Data Formats

| Input Data Format | VnMC/INF | VnMC/BPS | VnMC/IM | Captured Data Format |
|---|----------|----------|---------|----------------------|
| ITU-R BT.656/BT.601/BT.709/BT.1358 YCbCr | 000/001 | 0 | — | RGB format |
| | 010/011 | 1 | — | YCbCr format |
| | 100/101 | | | |
| ITU-R BT.601/BT.709 RGB | 110/111 | 0 | — | YCbCr format |
| | | 1 | — | RGB format |
| 8-bit user defined data (RAW8) or 8-bit embedded input (MIPI CSI-2 IF Only) | 100 | 1 | 01 | User defined format |
| RAW12 (Digital Pins Interface Only) | 011 | 1 | 01 | RAW12 format |

30.2.2 Video n Module Status Register (VnMS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:

n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 30 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | FBS[1:0] | FS | AV | CA | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 7 to 5 | — | B'000 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4, 3 | FBS[1:0] | B'11 | R | Frame Buffer Status These bits show the frame buffer status. b'00: The latest valid frame buffer has the base address defined by the memory base 1 register. b'01: The latest valid frame buffer has the base address defined by the memory base 2 register. b'10: The latest valid frame buffer has the base address defined by the memory base 3 register. b'11: There is no valid frame buffer. Note: When video capture is operating, this bit should be read after the FIS bit in VnINTS is set to 1. |
| 2 | FS | B'0 | R | Field Status This bit shows the type of the current capture field. 0: The current field is an odd field (field 1). 1: The current field is an even field (field 2). Note: When video capture is operating, this bit should be read after the FIS bit in VnINTS is set to 1. |
| 1 | AV | B'0 | R | Active Video Status This bit shows whether the current field is in the active video area defined by the pre-clipping register. 0: The current field is not in the active video area. 1: The current field is in the active video area. Note: This bit will be 0 if no input data is captured. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | CA | B'0 | R | <p>Video Capture Active Status</p> <p>This bit shows the current video capture operation status. This bit is updated by the captured field signal.</p> <p>0: Video capture is not operating.</p> <p>1: Video capture is operating.</p> <p>Note: In field capture mode, this bit is set to 1 even for the field that does not capture data.</p> |

30.2.3 Video n Frame Capture Register (VnFC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:

n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 30 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CC | SC |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | CC | B'0 | R/W | <p>Continuous Frame Capture Mode</p> <p>This bit specifies the continuous frame capture mode. In this mode, the first capture frame is written into the memory address specified by the memory base 1 (VnMB1) register. After that, the capture operation is repeated in the order of MB2, MB3, MB1, MB2, and such.</p> <p>Writing 0 into this bit during continuous capture operation will immediately terminate the capture operation if the current frame is completed or it has not been captured. After the capture operation resumed by writing 1 into this bit, the memory address that first capture frame is written is depending on the value indicated by the frame buffer status (FBS[1:0]) bits in the module status (VnMS) register.</p> <p>0: The continuous frame capture mode is not set. 1: The continuous frame capture mode is set.</p> <p>Note: When the capture is started with the SC bit set to 1 before setting this bit to 1, the first capture frame is written into the memory address specified by the memory base 2 (VnMB2) register.</p> |
| 0 | SC | B'0 | R/W | <p>Single Frame Capture Mode</p> <p>This bit specifies the single frame capture mode. In this mode, the capture frame is written into the memory address specified by the memory base 1 (VnMB1) register.</p> <p>Immediately after this bit is set to 1, the frame buffer status (FBS) bits in MS are initialized and the SC bit is also cleared to 0.</p> <p>0: The single frame capture mode is not set. 1: The single frame capture mode is set.</p> <p>Note: Do not set this bit to 1 when the interlace mode (IM) bits in the main control register (VnMC) are set to 01 (odd-/even-field capture mode).</p> |

Note: Do not specify the single frame capture mode and continuous frame capture mode at the same time.

30.2.4 Video n Start Line Pre-Clip Register (VnSLPrC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 30 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | SLPrC[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 0 | SLPrC[11:0] | All 0 | R/W | Start Line Pre-Clip These bits specify the (pre-clipping start line – 1) value in line units. This value is used before scaling. Specify a value in the range from 0 to 4094 so that the number of lines after pre-clipping will be 2 or more. (The value of 0 indicates the first valid line.) Note: In case of using Scaler, set the value of SLPrC within input lines. |

30.2.5 Video n End Line Pre-Clip Register (VnELPrC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

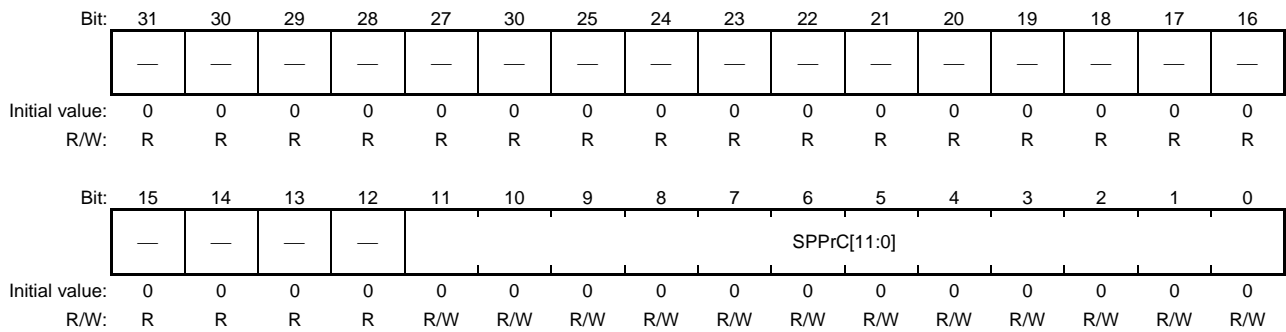
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 30 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | ELPrC[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 0 | ELPrC[11:0] | All 0 | R/W | End Line Pre-Clip These bits specify the (pre-clipping end line – 1) value in line units. This value is used before scaling. Specify a value in the range from 1 to 4095 so that the number of lines after pre-clipping will be 2 or more. Note: 1. When using vertical or horizontal scaling, specify a value in the range from 3 to 4095 so that the number of lines after pre-clipping will be 4 or more. 2. In case of using Scaler, set the value of ELPrC within input lines. 3. If the interlace images are input, the value here should be such that ELPrC - SLPrC is half of the value of the number of the original line. |

30.2.6 Video n Start Pixel Pre-Clip Register (VnSPPrC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 0 | SPPrC[11:0] | All 0 | R/W | Start Pixel Pre-Clip These bits specify the (pre-clipping start pixel – 1) value in pixel units. This value is used before scaling. Specify a value in the range from 0 to 4090 so that the number of pixels after pre-clipping will be an even number than 6. Notes: 1. When SPPrC is set to H'0, the first valid pixel is specified. 2. Specify an even number. 3. The capacity of the internal buffer is limited, so if the horizontal scaling up function is in use, the value here should be such that EPPrC – SPPrC is no greater than 2048 pixels. 4. In case of using Scaler, set the value of SPPrC within input pixels. 5. When RAW12 format are input, these bits specify the ((pre-clipping start pixel – 1) / 2) value in pixel units. |

30.2.7 Video n End Pixel Pre-Clip Register (VnEPPrC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 30 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | EPPrC[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 0 | EPPrC[11:0] | All 0 | R/W | End Pixel Pre-Clip These bits specify the (pre-clipping end pixel – 1) value in pixel units. This value is used before scaling. Specify a value in the range from 5 to 4095 so that the number of pixels after pre-clipping will be an even number than 6. Notes: 1. Set this bit so that the (EPPrC – SPPrC) value is an odd number. 2. The capacity of the internal buffer is limited, so if the horizontal scaling up function is in use, the value here should be such that EPPrC – SPPrC is no greater than 2048 pixels. 3. In case of using Scaler, set the value of EPPrC within input pixels. 4. When RAW12 format are input, these bits specify the ((pre-clipping end pixel / 2) – 1) value in pixel units. 5. When RAW12 format are input, it is necessary that the pre-clipping end pixel is an even number. |

30.2.8 Video n CSI2 Interface Mode Register (VnCSI_IFMD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:

n = 0 and 4 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 (RZ/G2E)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-----|------------|------|----|----|----|----|----|----------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 30 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | DES1/ — | DES0 | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CSI_CHSEL[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 28 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 30 | DES1/— | B'0 | R/W | Data Extension Select (CSI20 Input Data) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] This bit is used to select how data are expanded to 12 bits within the VIN module. 0: Empty bits in the input data are repeatedly expanded from the highest-order bit. 1: Empty bits will be padded with zeros. This bit must be set to 1 when the YCbCr-422 interface is in use. Reserved [RZ/G2E] These bits are always read as 0. The write value should always be 0. |
| 25 | DES0 | B'0 | R/W | Data Extension Select (CSI40 Input Data) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] This bit is used to select how data are expanded to 12 bits within the VIN module. 0: Empty bits in the input data are repeatedly expanded from the highest-order bit. 1: Empty bits will be padded with zeros. This bit must be set to 1 when the YCbCr-422 interface is in use. Data Extension Select (CSI40(2-Lane) Input Data) [RZ/G2E] This bit is used to select how data are expanded to 12 bits within the VIN module. 0: Empty bits in the input data are repeatedly expanded from the highest-order bit. 1: Empty bits will be padded with zeros. This bit must be set to 1 when the YCbCr-422 interface is in use. |
| 24 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------|---------------|-----|--|
| 3 to 0 | CSI_CHSEL[3:0] | H'0 | R/W | <p>These bits are used to select the CSI2 channels to which each of the video channels is connected.</p> <p>Available selections are given in Table 30.17.</p> <p>Note: x101, x110, x111: prohibit. [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] x100, x101, x110, x111: prohibit. [RZ/G2E] x011 (Video Channel 4), x101, x110, x111: prohibit. [RZ/G2H]</p> |

Table 30.17 CSI2 Channel Select Settings**[RZ/G2H]**

| VIN0 | VIN1 | VIN2 | VIN3 | V0CSI_IFMD/ CSI_CHSEL[2:0] | V0CSI_IFMD/ CSI_CHSEL[3] |
|-----------|-----------|-----------|-----------|-------------------------------|-----------------------------|
| CSI40/VC0 | CSI20/VC0 | CSI20/VC1 | CSI40/VC1 | 000 | 0 |
| | | | CSI40/EMB | 000 | 1 |
| CSI20/VC0 | CSI40/VC1 | CSI40/VC0 | CSI20/VC1 | 001 | 0 |
| | | | CSI20/EMB | 001 | 1 |
| CSI40/VC1 | CSI40/VC0 | CSI20/VC0 | CSI20/VC1 | 010 | 0 |
| | | | CSI20/EMB | 010 | 1 |
| CSI40/VC0 | CSI40/VC1 | CSI40/VC2 | CSI40/VC3 | 011 | 0 |
| | | | CSI40/EMB | 011 | 1 |
| CSI20/VC0 | CSI20/VC1 | CSI20/VC2 | CSI20/VC3 | 100 | 0 |
| | | | CSI20/EMB | 100 | 1 |

| VIN4 | VIN5 | VIN6 | VIN7 | V4CSI_IFMD/ CSI_CHSEL[2:0] | V4CSI_IFMD/ CSI_CHSEL[3] |
|------------|------------|------------|------------|-------------------------------|-----------------------------|
| No operate | CSI20/VC0 | CSI20/VC1 | No operate | 000 | 0 |
| | | | No operate | 000 | 1 |
| CSI20/VC0 | No operate | No operate | CSI20/VC1 | 001 | 0 |
| | | | CSI20/EMB | 001 | 1 |
| No operate | No operate | CSI20/VC0 | CSI20/VC1 | 010 | 0 |
| | | | CSI20/EMB | 010 | 1 |
| CSI20/VC0 | CSI20/VC1 | CSI20/VC2 | CSI20/VC3 | 100 | 0 |
| | | | CSI20/EMB | 100 | 1 |

Note: VC*: CSI2 Virtual Channel

EMB: CSI2 Embedded of Virtual Channel 0

[RZ/G2M V1.3, RZ/G2M V3.0]

| VIN0 | VIN1 | VIN2 | VIN3 | V0CSI_IFMD/ CSI_CHSEL[2:0] | V0CSI_IFMD/ CSI_CHSEL[3] |
|------------|------------|------------|------------|-------------------------------|-----------------------------|
| CSI40/VC0 | CSI20/VC0 | No operate | CSI40/VC1 | 000 | 0 |
| | | | CSI40/EMB | 000 | 1 |
| CSI20/VC0 | No operate | CSI40/VC0 | CSI20/VC1 | 001 | 0 |
| | | | CSI20/EMB | 001 | 1 |
| No operate | CSI40/VC0 | CSI20/VC0 | No operate | 010 | 0 |
| | | | No operate | 010 | 1 |
| CSI40/VC0 | CSI40/VC1 | CSI40/VC2 | CSI40/VC3 | 011 | 0 |
| | | | CSI40/EMB | 011 | 1 |
| CSI20/VC0 | CSI20/VC1 | CSI20/VC2 | CSI20/VC3 | 100 | 0 |
| | | | CSI20/EMB | 100 | 1 |

| VIN4 | VIN5 | VIN6 | VIN7 | V4CSI_IFMD/ CSI_CHSEL[2:0] | V4CSI_IFMD/ CSI_CHSEL[3] |
|------------|------------|------------|------------|-------------------------------|-----------------------------|
| CSI40/VC0 | CSI20/VC0 | No operate | CSI40/VC1 | 000 | 0 |
| | | | CSI40/EMB | 000 | 1 |
| CSI20/VC0 | No operate | CSI40/VC0 | CSI20/VC1 | 001 | 0 |
| | | | CSI20/EMB | 001 | 1 |
| No operate | CSI40/VC0 | CSI20/VC0 | No operate | 010 | 0 |
| | | | No operate | 010 | 1 |
| CSI40/VC0 | CSI40/VC1 | CSI40/VC2 | CSI40/VC3 | 011 | 0 |
| | | | CSI40/EMB | 011 | 1 |
| CSI20/VC0 | CSI20/VC1 | CSI20/VC2 | CSI20/VC3 | 100 | 0 |
| | | | CSI20/EMB | 100 | 1 |

Note: VC*: CSI2 Virtual Channel

EMB: CSI2 Embedded of Virtual Channel 0

No operate: Set 0 to VnMC/ME and VnFC/CC and VnFC/SC bit.

[RZ/G2N]

| VIN0 | VIN1 | VIN2 | VIN3 | V0CSI_IFMD/ CSI_CHSEL[2:0] | V0CSI_IFMD/ CSI_CHSEL[3] |
|-----------|-----------|-----------|-----------|-------------------------------|-----------------------------|
| CSI40/VC0 | CSI20/VC0 | CSI20/VC1 | CSI40/VC1 | 000 | 0 |
| | | | CSI40/EMB | 000 | 1 |
| CSI20/VC0 | CSI40/VC1 | CSI40/VC0 | CSI20/VC1 | 001 | 0 |
| | | | CSI20/EMB | 001 | 1 |
| CSI40/VC1 | CSI40/VC0 | CSI20/VC0 | CSI20/VC1 | 010 | 0 |
| | | | CSI20/EMB | 010 | 1 |
| CSI40/VC0 | CSI40/VC1 | CSI40/VC2 | CSI40/VC3 | 011 | 0 |
| | | | CSI40/EMB | 011 | 1 |
| CSI20/VC0 | CSI20/VC1 | CSI20/VC2 | CSI20/VC3 | 100 | 0 |
| | | | CSI20/EMB | 100 | 1 |

| VIN4 | VIN5 | VIN6 | VIN7 | V4CSI_IFMD/ CSI_CHSEL[2:0] | V4CSI_IFMD/ CSI_CHSEL[3] |
|-----------|-----------|-----------|-----------|-------------------------------|-----------------------------|
| CSI40/VC0 | CSI20/VC0 | CSI20/VC1 | CSI40/VC1 | 000 | 0 |
| | | | CSI40/EMB | 000 | 1 |
| CSI20/VC0 | CSI40/VC1 | CSI40/VC0 | CSI20/VC1 | 001 | 0 |
| | | | CSI20/EMB | 001 | 1 |
| CSI40/VC1 | CSI40/VC0 | CSI20/VC0 | CSI20/VC1 | 010 | 0 |
| | | | CSI20/EMB | 010 | 1 |
| CSI40/VC0 | CSI40/VC1 | CSI40/VC2 | CSI40/VC3 | 011 | 0 |
| | | | CSI40/EMB | 011 | 1 |
| CSI20/VC0 | CSI20/VC1 | CSI20/VC2 | CSI20/VC3 | 100 | 0 |
| | | | CSI20/EMB | 100 | 1 |

Note: VC*: CSI2 Virtual Channel
 EMB: CSI2 Embedded of Virtual Channel 0

[RZ/G2E]

| VIN4 | VIN5 | V4CSI_IFMD/ CSI_CHSEL[2:0] | V4CSI_IFMD/ CSI_CHSEL[3] |
|------------|-----------|-------------------------------|-----------------------------|
| CSI40/VC0 | CSI40/EMB | 000 | 0/1 |
| No operate | CSI40/VC1 | 001 | 0/1 |
| CSI40/VC1 | CSI40/VC0 | 010 | 0/1 |
| CSI40/VC0 | CSI40/VC1 | 011 | 0/1 |

Note: VC*: CSI2 Virtual Channel
 EMB: CSI2 Embedded of Virtual Channel 0
 No operate: Set 0 to VnMC/ME and VnFC/CC and VnFC/SC bit.
 CSI2 module name of RZ/G2E is CSI40(CSI4LNK0), but the module is 2-Lane CSI2.
 CSI2 module name of RZ/G2E is not supported 4-Lane.

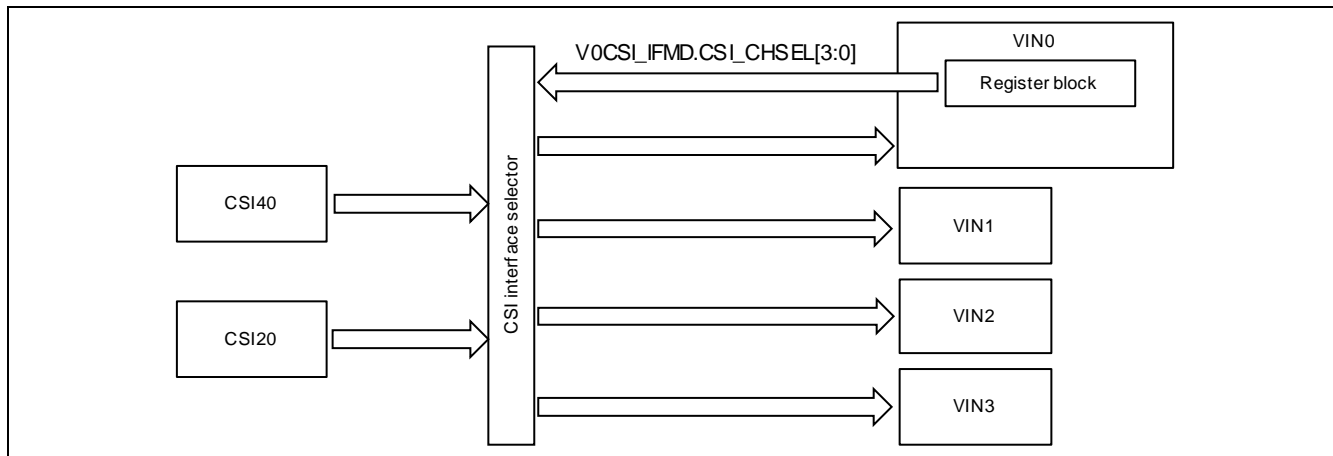


Figure 30.8 CSI2-VIN0/1/2/3 Connection [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

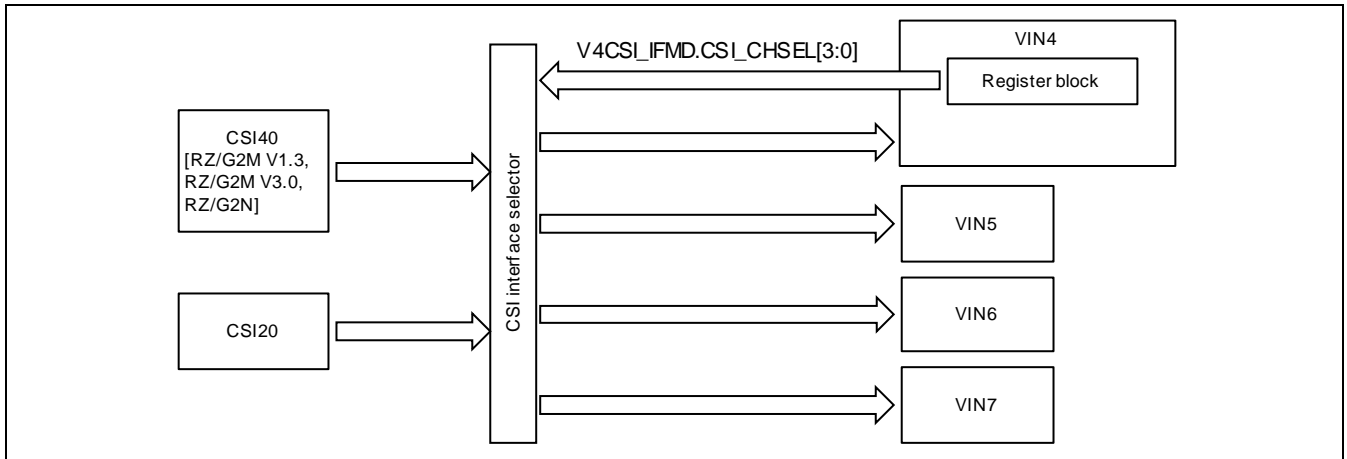


Figure 30.9 CSI2-VIN4/5/6/7 Connection [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

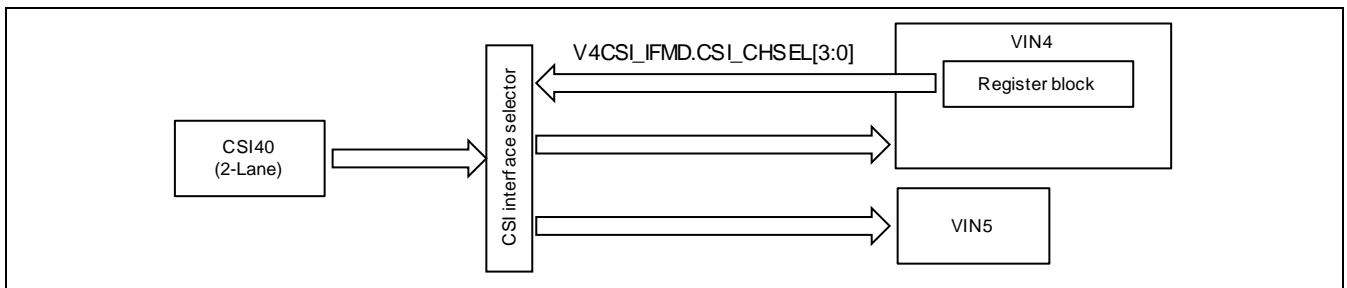


Figure 30.10 CSI2-VIN4/5 Connection [RZ/G2E]

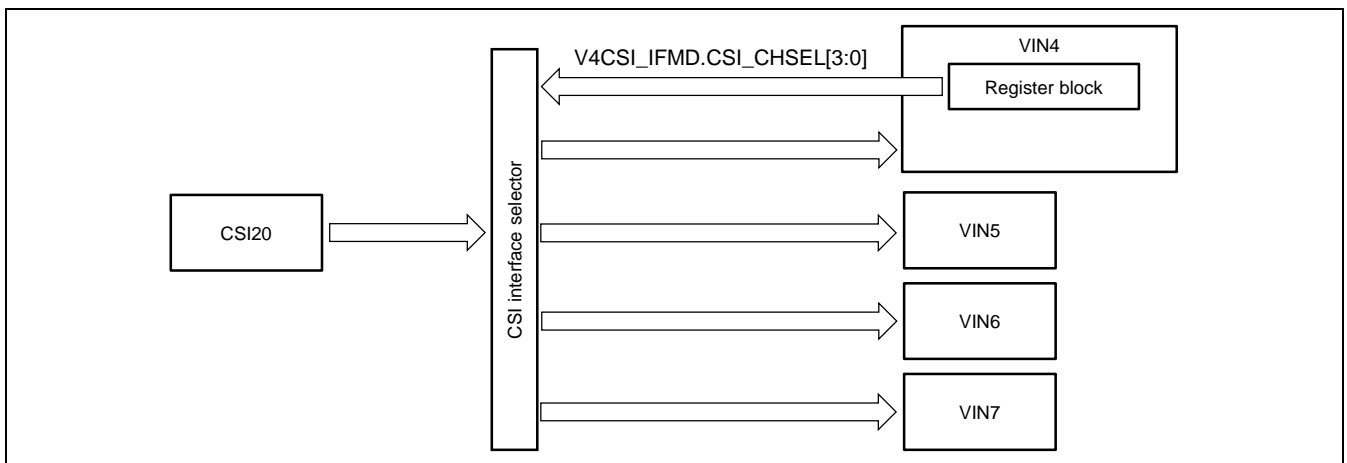


Figure 30.11 CSI2-VIN4/5/6/7 Connection [RZ/G2H]

30.2.9 Video n Image Stride Register (VnIS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

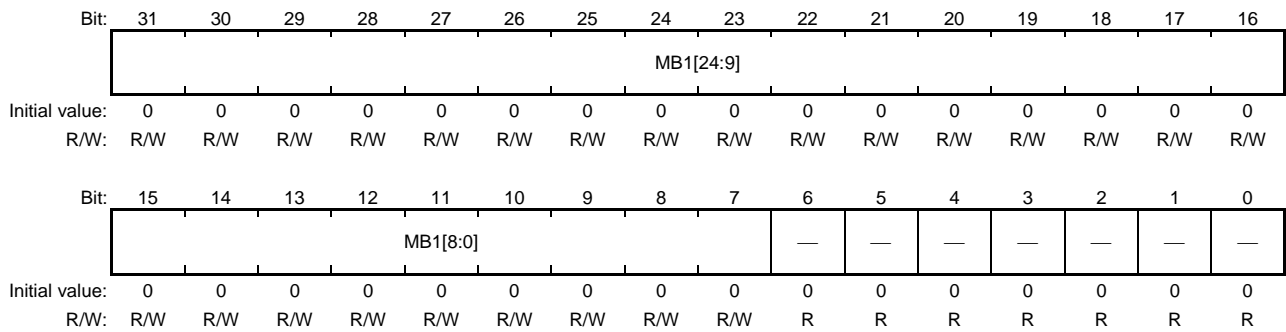
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|---------|-----|-----|-----|-----|-----|-----|-----|-----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | IS[8:0] | | | | | | | | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--|----------------------|-----------------|-----|--|---------------|----------------------|--------------|-------------------|----|---|------------------------------|----|---|--------------------|----|---|--------------------|----|---|---------------------------------|-----|---|---------------------------------|-----|---|--|----|---|--|----|---|----------------------------------|-----|-----------------|----------------------------------|-----|-----------------|--------|----|---|----------|----|---|--------|----|---|----------|----|---|-------|----|---|
| 31 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 12 to 4 | IS[8:0] | All 0 | R/W | Image Stride These bits specify the image stride in 16-pixel units. <ul style="list-style-type: none"> When the UDS is in use: Specify a value no less than the post-clipping width (CL_HSIZE). When the UDS is not in use: Specify a value no less than the pre-clipping width (EPPrC - SPPrC). <p>Note: The below table describes the stride register settings for a line stride alignment to 128 bytes. This is the recommended setting and ideal for the bus performance. The minimum requirement for line stride alignment is 32 bytes. Using line stride alignments less than 128 bytes will influence the bus performance of the system.</p> <table border="1" data-bbox="710 896 1428 1814"> <thead> <tr> <th>Output format</th> <th>Setting Unit (Pixel)</th> <th>Byte / Pixel</th> </tr> </thead> <tbody> <tr> <td>YCbCr-422, 8 bits</td> <td>64</td> <td>2</td> </tr> <tr> <td>RAW8 and 8-bit Embedded Data</td> <td>64</td> <td>2</td> </tr> <tr> <td>YCbCr-422, 10 bits</td> <td>32</td> <td>4</td> </tr> <tr> <td>YCbCr-422, 12 bits</td> <td>32</td> <td>4</td> </tr> <tr> <td>YC Separation YCbCr-422, 8 bits</td> <td>128</td> <td>1</td> </tr> <tr> <td>YC Separation YCbCr-420, 8 bits</td> <td>128</td> <td>1</td> </tr> <tr> <td>YC Separation YCbCr-422, Y(10 bits) / C(10 bits)</td> <td>64</td> <td>2</td> </tr> <tr> <td>YC Separation YCbCr-422, Y(12 bits) / C(12 bits)</td> <td>64</td> <td>2</td> </tr> <tr> <td>YC Separation YCbCr-422, 10 bits</td> <td>128</td> <td>Y: 2 CbCr: 1</td> </tr> <tr> <td>YC Separation YCbCr-422, 12 bits</td> <td>128</td> <td>Y: 2 CbCr: 1</td> </tr> <tr> <td>RGB565</td> <td>64</td> <td>2</td> </tr> <tr> <td>ARGB1555</td> <td>64</td> <td>2</td> </tr> <tr> <td>RGB888</td> <td>32</td> <td>4</td> </tr> <tr> <td>ARGB8888</td> <td>32</td> <td>4</td> </tr> <tr> <td>RAW12</td> <td>32</td> <td>4</td> </tr> </tbody> </table> | Output format | Setting Unit (Pixel) | Byte / Pixel | YCbCr-422, 8 bits | 64 | 2 | RAW8 and 8-bit Embedded Data | 64 | 2 | YCbCr-422, 10 bits | 32 | 4 | YCbCr-422, 12 bits | 32 | 4 | YC Separation YCbCr-422, 8 bits | 128 | 1 | YC Separation YCbCr-420, 8 bits | 128 | 1 | YC Separation YCbCr-422, Y(10 bits) / C(10 bits) | 64 | 2 | YC Separation YCbCr-422, Y(12 bits) / C(12 bits) | 64 | 2 | YC Separation YCbCr-422, 10 bits | 128 | Y: 2 CbCr: 1 | YC Separation YCbCr-422, 12 bits | 128 | Y: 2 CbCr: 1 | RGB565 | 64 | 2 | ARGB1555 | 64 | 2 | RGB888 | 32 | 4 | ARGB8888 | 32 | 4 | RAW12 | 32 | 4 |
| Output format | Setting Unit (Pixel) | Byte / Pixel | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| YCbCr-422, 8 bits | 64 | 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| RAW8 and 8-bit Embedded Data | 64 | 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| YCbCr-422, 10 bits | 32 | 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| YCbCr-422, 12 bits | 32 | 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| YC Separation YCbCr-422, 8 bits | 128 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| YC Separation YCbCr-420, 8 bits | 128 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| YC Separation YCbCr-422, Y(10 bits) / C(10 bits) | 64 | 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| YC Separation YCbCr-422, Y(12 bits) / C(12 bits) | 64 | 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| YC Separation YCbCr-422, 10 bits | 128 | Y: 2 CbCr: 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| YC Separation YCbCr-422, 12 bits | 128 | Y: 2 CbCr: 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| RGB565 | 64 | 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ARGB1555 | 64 | 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| RGB888 | 32 | 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ARGB8888 | 32 | 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| RAW12 | 32 | 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 to 0 | — | H'0 | R | Reserved bits that indicate the lower-order four bits of the image stride. These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

30.2.10 Video n Memory Base 1 Register (VnMB1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

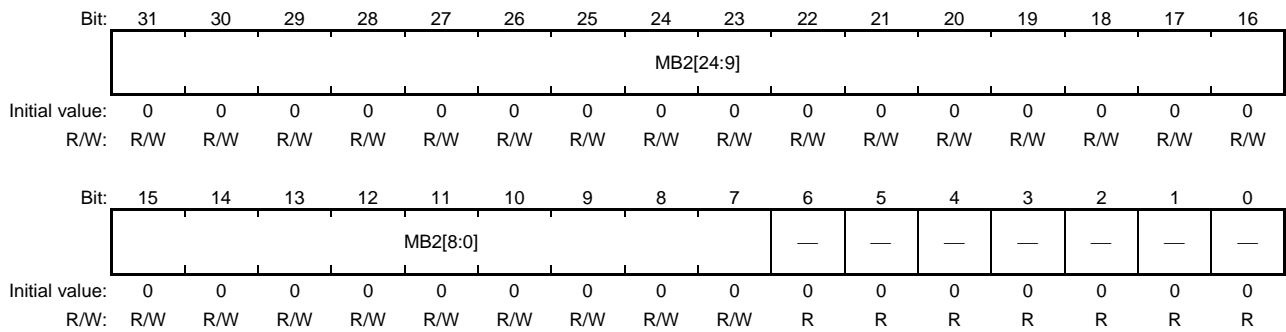


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 7 | MB1[24:0] | All 0 | R/W | <p>Memory Base Address 1</p> <p>These bits specify the transfer start address in frame buffer 1. Specify a value for physical address bits [31:7] in units of 128 bytes.</p> <p>If the module is in continuous frame capture mode, this value is used as the MB1 address in the following capture sequence: MB1 → MB2 → MB3 → MB1 → MB2 → MB3.</p> <p>In single frame capture mode, this value is used as the capture address.</p> <p>Specify a memory address taking into account the image size so that the image data does not exceed an area boundary on the address map.</p> |
| 6 to 0 | — | All 0 | R | <p>Reserved bits that indicate the lower-order seven bits of memory base address 1 (a multiple of 128 bytes).</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

30.2.11 Video n Memory Base 2 Register (VnMB2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

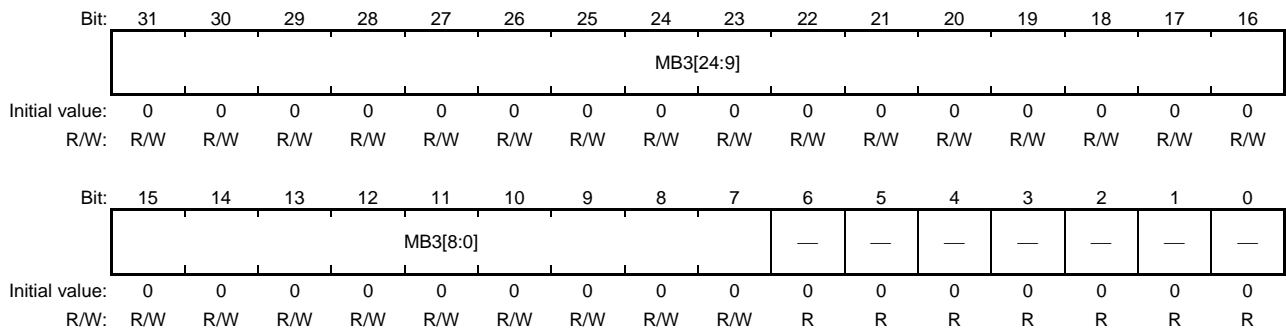


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 7 | MB2[24:0] | All 0 | R/W | <p>Memory Base Address 2</p> <p>These bits specify the transfer start address in frame buffer 2. Specify a value for physical address bits [31:7] in units of 128 bytes.</p> <p>If the module is in continuous frame capture mode, this value is used as the MB2 address in the following capture sequence: MB1 → MB2 → MB3 → MB1 → MB2 → MB3.</p> <p>Specify a memory address taking into account the image size so that the image data does not exceed an area boundary on the address map.</p> |
| 6 to 0 | — | All 0 | R | <p>Reserved bits that indicate the lower-order seven bits of memory base address 2 (a multiple of 128 bytes).</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

30.2.12 Video n Memory Base 3 Register (VnMB3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 7 | MB3[24:0] | All 0 | R/W | <p>Memory Base Address 3</p> <p>These bits specify the transfer start address in frame buffer 3. Specify a value for physical address bits [31:7] in units of 128 bytes. If the module is in continuous frame capture mode, this value is used as the MB3 address in the following capture sequence: MB1 → MB2 → MB3 → MB1 → MB2 → MB3.</p> <p>Specify a memory address taking into account the image size so that the image data does not exceed an area boundary on the address map.</p> |
| 6 to 0 | — | All 0 | R | <p>Reserved bits that indicate the lower-order seven bits of memory base address 3 (a multiple of 128 bytes).</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

30.2.13 Video n Line Count Register (VnLC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----------|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | LC[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 0 | LC[11:0] | All 0 | R | Line Count These bits show the line position in the current capture field. |

30.2.14 Video n Interrupt Enable Register (VnIE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:

n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

| | | | | | | | | | | | | | | | | |
|----------------|------|----|----|----|----|----|----|----|----|----|--------|---------|---------|--------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | FIE2 | — | — | — | — | — | — | — | — | — | —/EYEE | —/EUUEE | —/EVVEE | —/SEEE | VFE | VRE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R | R | R/W*1 | R/W*1 | R/W*1 | R/W*1 | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | FIE | CEE | SIE | EFE | FOE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

Notes: 1. This bit is reserved in RZ/G2M V1.3, RZ/G2M V3.0.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | FIE2 | B'0 | R/W | Field Interrupt Enable 2 This bit enables or disables INTC output for field interrupts. Interrupt signals by this enable bit is asserted irrespective of whether capture is taking place. 0: Field interrupts are disabled. 1: Field interrupts are enabled. |
| 30 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 22 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 21 | —/EYEE | B'0 | R | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] This bit is always read as 0. The write value should always be 0. |
| | | | R/W | Reserved. Keep initial value. [RZ/G2H, RZ/G2N, RZ/G2E] |
| 20 | —/EUUEE | B'0 | R | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] This bit is always read as 0. The write value should always be 0. |
| | | | R/W | Reserved. Keep initial value. [RZ/G2H, RZ/G2N, RZ/G2E] |
| 19 | —/EVVEE | B'0 | R | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] This bit is always read as 0. The write value should always be 0. |
| | | | R/W | Reserved. Keep initial value. [RZ/G2H, RZ/G2N, RZ/G2E] |
| 18 | —/SEEE | B'0 | R | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] This bit is always read as 0. The write value should always be 0. |
| | | | R/W | Reserved. Keep initial value. [RZ/G2H, RZ/G2N, RZ/G2E] |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 17 | VFE | B'0 | R/W | <p>VSYNC Falling Edge Detect Interrupt Enable</p> <p>This bit enables or disables VSYNC falling edge detect interrupts. Interrupt signals by this enable bit is asserted irrespective of whether capture is taking place.</p> <p>0: VSYNC falling edge detect interrupts are disabled. 1: VSYNC falling edge detect interrupts are enabled.</p> |
| 16 | VRE | B'0 | R/W | <p>VSYNC Rising Edge Detect Interrupt Enable</p> <p>This bit enables or disables VSYNC rising edge detect interrupts. Interrupt signals by this enable bit is asserted irrespective of whether capture is taking place.</p> <p>0: VSYNC rising edge detect interrupts are disabled. 1: VSYNC rising edge detect interrupts are enabled.</p> |
| 15 to 5 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 4 | FIE | B'0 | R/W | <p>Field Interrupt Enable</p> <p>This bit enables or disables field-switching interrupts. This interrupt enable setting is valid when the CA bit in VnMS is 1.</p> <p>0: Field-switching interrupts are disabled. 1: Field-switching interrupts are enabled.</p> |
| 3 | CEE | B'0 | R/W | <p>Correction Error Interrupt Enable</p> <p>This bit enables or disables interrupts due to error correction in the timing reference code (SAV/EAV) described in the ITU-R BT.656 specification. This interrupt enable setting is valid when the CA bit in VnMS is 1.</p> <p>0: ITU-R BT.656 timing reference code error interrupts are disabled. 1: ITU-R BT.656 timing reference code error interrupts are enabled.</p> |
| 2 | SIE | B'0 | R/W | <p>Scanline Interrupt Enable</p> <p>This bit enables or disables scanline interrupts. This interrupt enable setting is valid when the CA bit in VnMS is 1.</p> <p>0: Scanline interrupts are disabled. 1: Scanline interrupts are enabled.</p> |
| 1 | EFE | B'0 | R/W | <p>End of Frame Interrupt Enable</p> <p>This bit enables or disables end of frame interrupts. This interrupt enable setting is valid when the CA bit in VnMS is 1.</p> <p>0: End of frame interrupts are disabled. 1: End of frame interrupts are enabled.</p> |
| 0 | FOE | B'0 | R/W | <p>FIFO Overflow Interrupt Enable</p> <p>This bit enables or disables FIFO overflow interrupts. This interrupt enable setting is valid when the CA bit in VnMS is 1.</p> <p>0: FIFO overflow interrupts are disabled. 1: FIFO overflow interrupts are enabled.</p> |

30.2.15 Video n Interrupt Status Register (VnINTS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:

n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

| | | | | | | | | | | | | | | | | |
|----------------|------|----|----|----|----|----|----|----|----|----|--------|--------|--------|-------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | FIS2 | — | — | — | — | — | — | — | — | — | —/EYER | —/EUER | —/EVER | —/SER | VFS | VRS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R | R | R/W*1 | R/W*1 | R/W*1 | R/W*1 | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | FIS | CES | SIS | EFS | FOS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

Notes: 1. This bit is reserved in RZ/G2M V1.3, RZ/G2M V3.0.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | FIS2 | B'0 | R/W | Field Interrupt Status 2 This bit shows that the field has changed. This bit is set to 1 when a valid field is detected in the ITU-R BT.601 interface or the F bit defined in ITU-R BT.656 changes. After being set to 1, this bit is cleared to 0 by writing 1. Note: This bit is set to 1 irrespective of whether or not capture is taking place. Be sure to clear this bit to 0 before using it. |
| 30 to 24 | — | All 0 | R | Reserved These bits are always read as 0 or 1. The write value should always be 0. |
| 23 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 22 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 21 | —/EYER | B'0 | R | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] This bit is always read as 0. The write value should always be 0. |
| | | | R/W | Reserved. Keep initial value. [RZ/G2H, RZ/G2N, RZ/G2E] |
| 20 | —/EUER | B'0 | R | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] This bit is always read as 0. The write value should always be 0. |
| | | | R/W | Reserved. Keep initial value. [RZ/G2H, RZ/G2N, RZ/G2E] |
| 19 | —/EVER | B'0 | R | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] This bit is always read as 0. The write value should always be 0. |
| | | | R/W | Reserved. Keep initial value. [RZ/G2H, RZ/G2N, RZ/G2E] |
| 18 | —/SER | B'0 | R | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] This bit is always read as 0. The write value should always be 0. |
| | | | R/W | Reserved. Keep initial value. [RZ/G2H, RZ/G2N, RZ/G2E] |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 17 | VFS | B'0 | R/W | <p>VSYNC Falling Edge Detect Interrupt Status</p> <p>This bit shows that a VSYNC falling edge has been detected in the ITU-R BT.601 input. After being set to 1, this bit is cleared to 0 by writing 1.</p> <p>Note: This bit is set to 1 irrespective of whether or not capture is taking place. Be sure to clear this bit to 0 before using it.</p> |
| 16 | VRS | B'0 | R/W | <p>VSYNC Rising Edge Detect Interrupt Status</p> <p>This bit shows that a VSYNC rising edge has been detected in the ITU-R BT.601 input. After being set to 1, this bit is cleared to 0 by writing 1.</p> <p>Note: This bit is set to 1 irrespective of whether or not capture is taking place. Be sure to clear this bit to 0 before using it.</p> |
| 15 to 5 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 4 | FIS | B'0 | R/W | <p>Field Interrupt Status</p> <p>This bit shows that a field has been captured in the active capture operation.</p> <p>This bit is set to 1 when a valid field is detected in the ITU-R BT.601 interface or the F bit defined in ITU-R BT.656 changes. After being set to 1, this bit is cleared to 0 by writing 1.</p> |
| 3 | CES | B'0 | R/W | <p>Correction Error Interrupt Status</p> <p>This bit shows that the timing reference code in the active capture operation has an error involving at least two bits. After being set to 1, this bit is cleared to 0 by writing 1.</p> <p>This bit is set to 1 if the EC bit in VnMC is enabled and the timing reference code has an error involving at least two bits. If a 1-bit error occurs when the EC bit is enabled, this bit is not set to 1.</p> |
| 2 | SIS | B'0 | R/W | <p>Scanline Interrupt Status</p> <p>This bit shows that the number of lines specified by VnSI has been reached in the active capture operation. After being set to 1, this bit is cleared to 0 by writing 1.</p> <p>This bit is set to 1 at the next line start timing after the value in the VnLC register matches the VnSI register setting. This timing is shown in Figure 30.12, Scanline Interrupt Status Generation Timing.</p> |
| 1 | EFS | B'0 | R/W | <p>End of Frame Interrupt Status</p> <p>This bit shows that the last frame has been reached in the active capture operation. This bit is set to 1 at the end of even field (field 2). After being set to 1, this bit is cleared to 0 by writing 1.</p> |
| 0 | FOS | B'0 | R/W | <p>FIFO Overflow Interrupt Status</p> <p>This bit shows that the FIFO has overflowed in the active capture operation. After being set to 1, this bit is cleared to 0 by writing 1.</p> <p>If the FIFO overflows, the FIFO data is overwritten by the pixel data captured after the overflow, and the resultant data is sent to the frame buffer.</p> |

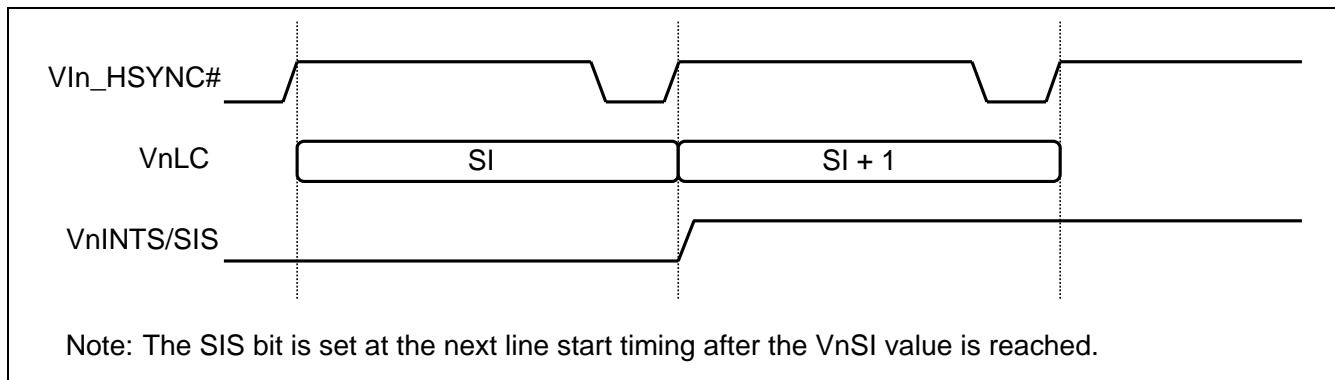


Figure 30.12 Scanline Interrupt Status Generation Timing

30.2.16 Video n Scanline Interrupt Register (VnSI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

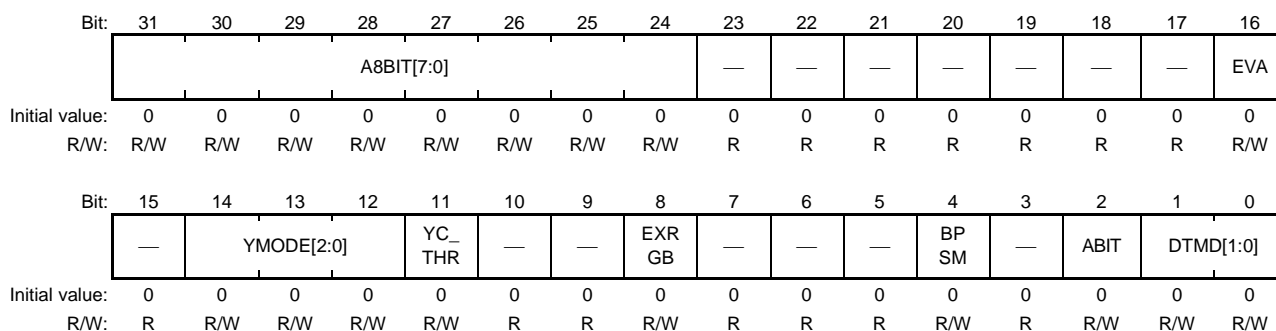
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | SI[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 0 | SI[11:0] | All 0 | R/W | Scanline Interrupt Setting These bits specify a value to be compared with the VnLC register value in each field while the SIE bit in the VnIE register is set to 1. When this value matches the VnLC register value, an interrupt signal is asserted. Note: When these bits are set to H'000, the scanline interrupt status bit (SIS) in the VnINTS is always 0. |

30.2.17 Video n Data Mode Register (VnDMR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 24 | A8BIT[7:0] | All 0 | R/W | Alpha 8 These bits set the alpha value for the ARGB8888 format output. |
| 23 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | EVA | B'0 | R/W | Even Field Address Offset This bit specifies the address offset of the even field (field 2) in memory in odd-/even-field capture mode. 0: Data are stored from the base address in external memory. 1: Data are stored from the base address plus the memory width in external memory. |
| 15 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 14 to 12 | YMODE[2:0] | B'000 | R/W | YC Data Transfer Mode These bits specify the transfer method of Y/CbCr when the data conversion mode (DTMD) is 10 (YC separation). b'000: Both Y and CbCr data are transferred to memory b'001: Only Y data is transferred to memory as 8-bit data b'010: 10-bit Y data is converted to 16-bit data and both Y and CbCr data are transferred to memory. b'011: 10-bit Y data is converted to 16-bit data and only Y data is transferred to memory. b'100: 12-bit Y data is converted to 16-bit data and both Y and CbCr data are transferred to memory. b'101: 12-bit Y data is converted to 16-bit data and only Y data is transferred to memory. b'110: Setting prohibited b'111: Setting prohibited |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 11 | YC_THR | B'0 | R/W | <p>YC Data Through Mode</p> <p>Y and CbCr data are transferred to memory as 10-bit or 12-bit data according to the input format by setting this bit.</p> <p>0: Y and CbCr data are transferred to memory according to YMODE[2:0] bits.</p> <p>1: Y and CbCr data are transferred to memory as 10-bit or 12-bit data according to the input format.</p> |
| 10 to 9 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 8 | EXRGB | B'0 | R/W | <p>Extension RGB Conversion Mode</p> <p>0: RGB data extension processing is not performed.</p> <p>1: Data is extended to 32-bit RGB conversion when DTMD[1:0] is set to 00 or 01 as the data conversion mode.</p> |
| 7 to 5 | — | B'000 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 4 | BPSM | B'0 | R/W | <p>Output Data Byte Swap Mode</p> <p>0: Bytes are not swapped in output data.</p> <p>1: Bytes are swapped in output data.</p> <p>Note: When YCbCr-422 data is output in big endian, data is transferred in the YUYV format in most cases. To transfer data in the UYVY format, set this bit to 1.</p> |
| 3 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 2 | ABIT | B'0 | R/W | <p>Alpha Bit</p> <p>This bit specifies the alpha value for data in ARGB-1555 output mode.</p> <p>0: The alpha value is set to 0.</p> <p>1: The alpha value is set to 1.</p> |
| 1, 0 | DTMD[1:0] | B'00 | R/W | <p>Data Conversion Mode</p> <p>These bits set the format for storing RGB888 or YCbCr444 data after LUT conversion, in the external memory. *1</p> <p>b'00: Data is not converted.</p> <p>b'01: RGB is converted to ARGB before output.</p> <p>b'10: YC is separated before output. *2</p> <p>b'11: YC is separated before output (YCbCr420 format) *3.</p> |

- Notes:
1. The data conversion modes that can be set are shown in Table 30.18. Do not set any other mode that is not listed in the table. RGB and YCbCr data after LUT conversion should be set as shown in Table 30.18.
 2. Do not set for any other data format except for YCbCr422; set YC separation only for YCbCr data format.
 3. Do not make this setting for video channels 2, 3, 6 and 7.

Table 30.18 Data Conversion Settings

• RGB Data Conversion Modes

| Format of Data Stored in Memory | VnDMR/ YMODE[2:0] | VnDMR/ EXRGB | VnDMR/ YC_THR | VnDMR/ DTMD[1:0] | Remarks |
|----------------------------------|----------------------|-----------------|------------------|---------------------|---|
| RGB-565 (16 bits/pixel) format | 000 | 0 | 0 | 00 | |
| RGB-888 (32 bits/pixel) format | 000 | 1 | 0 | 00 | |
| ARGB-1555 (16 bits/pixel) format | 000 | 0 | 0 | 01 | The alpha bit is set with the ABIT bit in VnDMR. |
| ARGB-8888 (32 bits/pixel) format | 000 | 1 | 0 | 01 | The alpha bit is set with the A8BIT bit in VnDMR. |

• YCbCr Data Conversion Modes

| Format of Data Stored in Memory | VnDMR/ YMODE[2:0] | VnDMR/ EXRGB | VnDMR/ YC_THR | VnDMR/ DTMD[1:0] | Remarks |
|---|----------------------|-----------------|------------------|---------------------|---|
| YCbCr-422 (8 bits) transfer | 000 | 0 | 0 | 00 | Set the BPSM bit in VnDMR to 1 to transfer in UYVY format in big endian. Make these settings if the setting of the INF[2:0] bits in VnMC is B'100. |
| YCbCr-422 (10 or 12 bits) transfer* | 000 | 0 | 1 | 00 | Data selection of 10-bit or 12-bit is according to the input format. |
| Y (8 bits)/CbCr separation transfer | 000 | 0 | 0 | 10/11 | CbCr transfer destination is determined by the VnUVAOF register setting. Setting the DTMD[1:0] bits to B'11 is only allowed for video channels 0, 1, 4, and 5. |
| Y (10 or 12 bits)/CbCr separation transfer* | 000 | 0 | 1 | 10 | Data selection of 10-bit or 12-bit is according to the input format. |
| Y (8 bits) transfer | 001 | 0 | 0 | 10 | |
| Y (16 bits)/CbCr separation transfer | 010/100 | 0 | 0 | 10 | CbCr transfer destination is determined by the VnUVAOF register setting. |
| Y (16 bits) transfer | 011/101 | 0 | 0 | 10 | |

Note: If any combination of values not listed above is specified, correct operation is not guaranteed.

• 8-bit user defined data Mode

| Format of Data Stored in Memory | VnDMR/ YMODE[2:0] | VnDMR/ EXRGB | VnDMR/ YC_THR | VnDMR/ DTMD[1:0] | Remarks |
|---------------------------------|----------------------|-----------------|------------------|---------------------|--------------------|
| RAW8 or Embedded | 000 | 0 | 0 | 00 | MIPI CSI-2 IF Only |

- 12bit RAW data Mode

| Format of Data Stored in Memory | VnDMR/ YMODE[2:0] | VnDMR/ EXRGB | VnDMR/ YC_THR | VnDMR/ DTMD[1:0] | Remarks |
|--|------------------------------|-------------------------|--------------------------|-----------------------------|-----------------------------|
| RAW12 | 000 | 0 | 1 | 00 | Digital Pins Interface Only |

30.2.18 Video n Data Mode Register 2 (VnDMR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:

n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

| | | | | | | | | | | | | | | | | |
|----------------|----------|-----|-----|-----|-----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | FPS | VPS | HPS | CES | DES | — | — | — | CHS | YDS | — | — | — | — | FT EV | FT EH |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | VLV[3:0] | | | | HLV[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | FPS | B'0 | R/W | Field Signal Polarity Select This bit specifies the polarity of the input field signal in the ITU-R BT.601 interface. 0: 0/1 = Odd field (field 1)/Even field (field 2) 1: 0/1 = Even field (field 2)/Odd field (field 1) |
| 30 | VPS | B'0 | R/W | Vsync Signal Polarity Select This bit specifies the polarity of the input vertical sync signal in the ITU-R BT.601 interface. 0: Active low 1: Active high |
| 29 | HPS | B'0 | R/W | Hsync Signal Polarity Select This bit specifies the polarity of the input horizontal sync signal in the ITU-R BT.601 interface. 0: Active low 1: Active high |
| 28 | CES | B'0 | R/W | Clock Enable Signal Polarity Select This bit specifies the polarity of the input clock enable signal in the ITU-R BT.601. 0: Active high 1: Active low |
| 27 | DES | B'0 | R/W | Data Extension Select This bit is used to select how data are expanded to 12 bits within the VIN module. 0: Empty bits in the input data are repeatedly expanded from the highest-order bit. 1: Empty bits will be padded with zeros. This bit must be set to 1 when the YCbCr-422 interface is in use. |
| 30 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

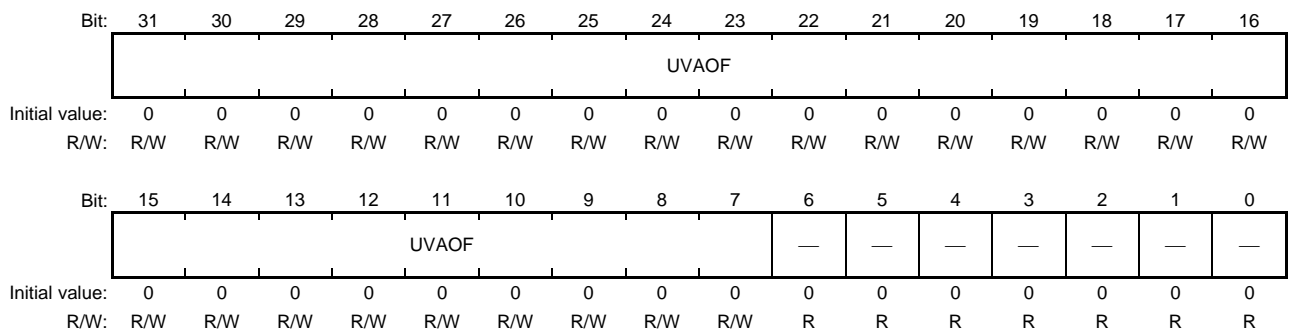
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 23 | CHS | B'0 | R/W | <p>Clock Enable Hsync Select</p> <p>The HSYNC signal (VIn_HSYNC#) input from the pin is internally used as the clock enable signal.</p> <p>0: Clock enable signal (VIn_CLKENB) input from the pin is internally used as the clock enable signal.</p> <p>1: HSYNC signal (VIn_HSYNC#) input from the pin is internally used as the clock enable signal.</p> <p>Note: When using ITU-R BT.601, BT.709, BT.1358 interface, and the VIn_CLKENB pin is unused, the CHS bit must be set to 1.</p> |
| 22 | YDS | B'0 | R/W | <p>YCbCr422 8-bit Data Input Pin Select</p> <p>This bit specifies the YCbCr422 8-bit data input pins.</p> <p>0: VIn_DATA[7:0] pins</p> <p>1: VIn_DATA[15:8] pins</p> <p>Note: This bit can only be set when the YCbCr-422 interface is in use.</p> |
| 21 to 18 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 17 | FTEV | B'0 | R/W | <p>VSYNC Field Toggle Mode Enable</p> <p>The VSYNC field toggle mode changes the capture field signal level according to the count of input VSYNC signal assertion. As the VIN controls capture operation only when the input field signal level changes, select the VSYNC field toggle mode when capturing progressive images or MIPI CSI-2 input data.</p> <p>0: The field toggle function according to the VSYNC count is disabled.</p> <p>1: The field toggle function according to the VSYNC count is enabled. The period before a toggle should be specified in the VLV bits.</p> <p>Note: Do not set both FTEH and FTEV at the same time.</p> |
| 16 | FTEH | B'0 | R/W | <p>HSYNC Field Toggle Counter Enable</p> <p>0: The field toggle function according to the capture active line is disabled.</p> <p>1: The field toggle function according to the capture active line is enabled. The period before a toggle should be specified in the HLV bits.</p> <p>Note: Do not set both FTEH and FTEV at the same time.</p> |
| 15 to 12 | VLV[3:0] | H'0 | R/W | <p>VSYNC Field Toggle Mode Transition Period</p> <p>These bits specify the count of vertical sync signal input before the VSYNC field toggle mode is entered. After a transition to the VSYNC field toggle mode, the capture field signal is toggled every time VSYNC is input.</p> <p>When a change in the input field signal is detected, the toggle mode is canceled.</p> <p>H'0: The field signal is toggled at every VSYNC input.</p> <p>H'1: Toggle mode is entered after VSYNC is input once.</p> <p>H'2: Toggle mode is entered after VSYNC is input two times.</p> <p>H'3: Toggle mode is entered after VSYNC is input three times.</p> <p>:</p> <p>H'E: Toggle mode is entered after VSYNC is input 14 times</p> <p>H'F: Toggle mode is entered after VSYNC is input 15 times.</p> <p>Note: If the field signal changes while the transition period is counted, the counter is initialized.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 11 to 0 | HLV[11:0] | H'000 | R/W | <p>HSYNC Field Toggle Count Value</p> <p>The HSYNC field toggle counter counts the capture active lines. If the external field signal does not change before the prespecified counter value is reached, the capture field signal is toggled.</p> <p>H'000: The field signal is toggled for every valid line. H'001: The field signal is toggled for every two valid lines. H'002: The field signal is toggled for every three valid lines. : H'FFF: The field signal is toggled for every 4096 valid lines.</p> <p>Note: For the period before a toggle, specify a value greater than one VSYNC period.</p> |

30.2.19 Video n UV Address Offset Register (VnUVAOF)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|--|
| 31 to 7 | UVAOF[24:0] | All 0 | R/W | <p>UV Data Address Offset</p> <p>These bits specify the transfer offset address for the YC separation YCbCr-422 UV and YCbCr-420 UV data.</p> <p>Specify bits 31 to 7 of the physical address in 128-byte units.</p> <p>Note: The specified address should be equal to or greater than the Y transfer size. Otherwise, the overwriting of Y data occurs.</p> |
| 6 to 0 | — | All 0 | R | <p>Reserved bits that indicate the 128-byte boundary of the UVAOF value.</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

30.2.20 YC-RGB Conversion Coefficient Registers

YC→RGB color space conversion is performed with the following formula. Each of the coefficients can be set through the registers. Here, if 8-bit data format is set, the coefficient registers set with the YMUL, CSUB, and YSUB bits in the VnCSCE1 register, RCRMUL and GCRMUL bits in the VnCSCE2 register and GCBMUL and BCBMUL bits in the VnCSCE3 register are used to convert the color space. When 10-bit/12-bit data format is set, the YMUL2 bit in the VnCSCE1 register, CSUB2 and YSUB2 bits in the VnCSCE2 register, RCRMUL2 and GCRMUL2 bits in the VnCSCE3 register, GCBMUL2 and BCBMUL2 bits in the VnCSCE4 register are used in color space conversion.

$$\begin{aligned}
 R &= \begin{pmatrix} VnCSCE1/ \\ YMUL[9:0] \\ or \\ VnCSCE1/ \\ YMUL2[13:0] \end{pmatrix} \times (Y - \begin{pmatrix} VnCSCE1/ \\ YSUB[7:0] \\ or \\ VnCSCE1/ \\ YSUB2[11:0] \end{pmatrix}) + \begin{pmatrix} VnCSCE2/ \\ RCRMUL[9:0] \\ or \\ VnCSCE3/ \\ RCRMUL2[13:0] \end{pmatrix} \times (Cr - \begin{pmatrix} VnCSCE1/ \\ CSUB[7:0] \\ or \\ VnCSCE1/ \\ CSUB2[11:0] \end{pmatrix}) \\
 G &= \begin{pmatrix} VnCSCE1/ \\ YMUL[9:0] \\ or \\ VnCSCE1/ \\ YMUL2[13:0] \end{pmatrix} \times (Y - \begin{pmatrix} VnCSCE1/ \\ YSUB[7:0] \\ or \\ VnCSCE1/ \\ YSUB2[11:0] \end{pmatrix}) - \begin{pmatrix} VnCSCE2/ \\ GCRMUL[9:0] \\ or \\ VnCSCE3/ \\ GCRMUL2[13:0] \end{pmatrix} \times (Cr - \begin{pmatrix} VnCSCE1/ \\ CSUB[7:0] \\ or \\ VnCSCE1/ \\ CSUB2[11:0] \end{pmatrix}) - \begin{pmatrix} VnCSCE3/ \\ GCBMUL[9:0] \\ or \\ VnCSCE4/ \\ GCBMUL2[13:0] \end{pmatrix} \times (Cb - \begin{pmatrix} VnCSCE1/ \\ CSUB[7:0] \\ or \\ VnCSCE1/ \\ CSUB2[11:0] \end{pmatrix}) \\
 B &= \begin{pmatrix} VnCSCE1/ \\ YMUL[9:0] \\ or \\ VnCSCE1/ \\ YMUL2[13:0] \end{pmatrix} \times (Y - \begin{pmatrix} VnCSCE1/ \\ YSUB[7:0] \\ or \\ VnCSCE1/ \\ YSUB2[11:0] \end{pmatrix}) + \begin{pmatrix} VnCSCE3/ \\ BCBMUL[9:0] \\ or \\ VnCSCE4/ \\ BCBMUL2[13:0] \end{pmatrix} \times (Cb - \begin{pmatrix} VnCSCE1/ \\ CSUB[7:0] \\ or \\ VnCSCE1/ \\ CSUB2[11:0] \end{pmatrix})
 \end{aligned}$$

Note: Set the coefficients for ITU-R BT.601 (8 bits) as the initial value.

$$R = 1.164 \times (Y - 16) + 1.596 \times (Cr - 128)$$

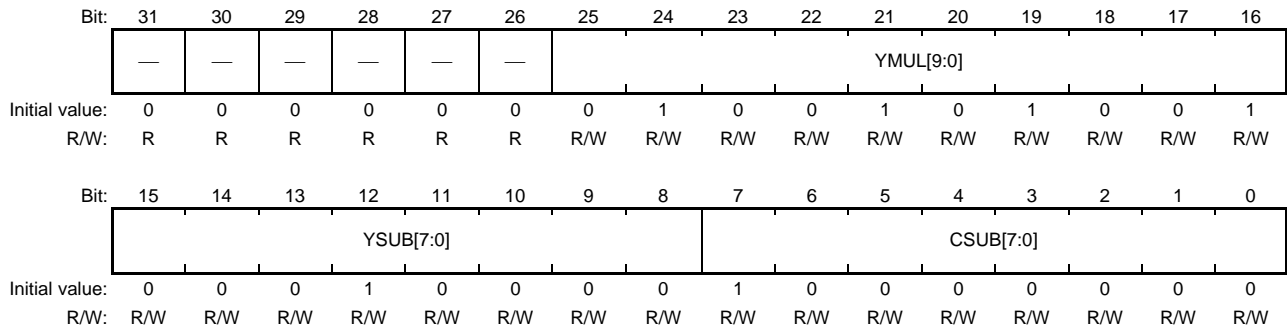
$$G = 1.164 \times (Y - 16) - 0.813 \times (Cr - 128) - 0.392 \times (Cb - 128)$$

$$B = 1.164 \times (Y - 16) + 2.017 \times (Cb - 128)$$

30.2.20.1 Video n Color Space Change Coefficient 1 Register (VnCSCC1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

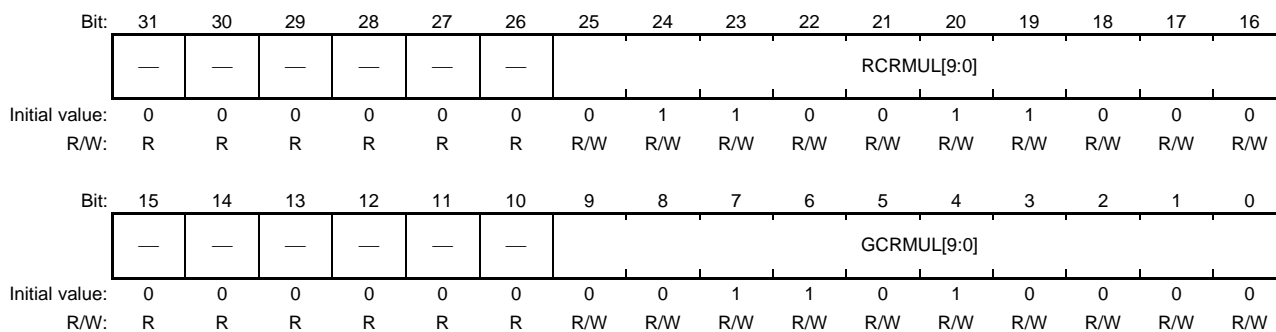


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25 to 16 | YMUL[9:0] | H'129 | R/W | Y Data Multiplication Coefficient These bits specify the multiplication coefficient for Y data in YCbCr-444→RGB-888 color space conversion. (Initial value: 1.164) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256. |
| 15 to 8 | YSUB[7:0] | H'10 | R/W | Y Data Subtraction Coefficient These bits specify the subtraction coefficient for Y data in YCbCr-444→RGB-888 color space conversion. (Initial value: 16) Specify an unsigned 8-bit integer. |
| 7 to 0 | CSUB[7:0] | H'80 | R/W | CbCr Data Subtraction Coefficient These bits specify the subtraction coefficient for Cb and Cr data in YCbCr-444→RGB-888 color space conversion. (Initial value: 128) Specify an unsigned 8-bit integer. |

30.2.20.2 Video n Color Space Change Coefficient 2 Register (VnCSCC2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

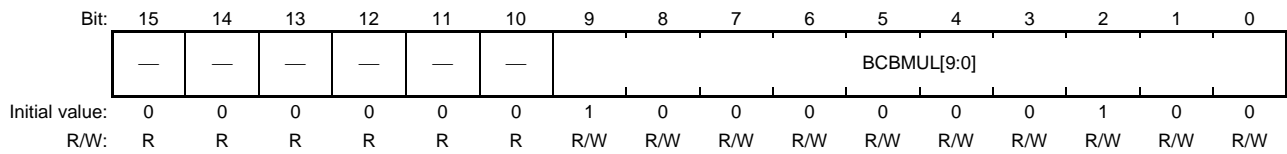
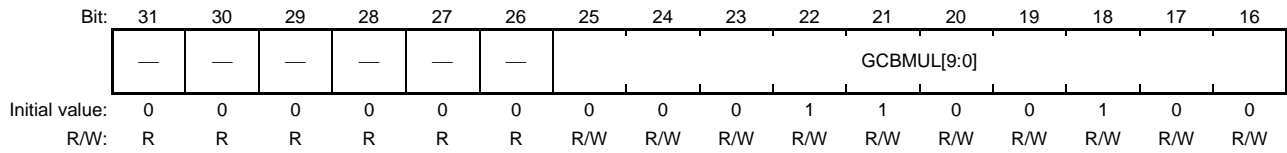


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25 to 16 | RCRMUL [9:0] | H'198 | R/W | Cr Multiplication Coefficient for R Data Calculation These bits specify the Cr multiplication coefficient for the R data calculation equation in YCbCr-444 → RGB-888 color space conversion. (Initial value: 1.596) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256. |
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 to 0 | GCRMUL [9:0] | H'0D0 | R/W | Cr Multiplication Coefficient for G Data Calculation These bits specify the Cr multiplication coefficient for the G data calculation equation in YCbCr-444 → RGB-888 color space conversion. (Initial value: 0.813) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256. |

30.2.20.3 Video n Color Space Change Coefficient 3 Register (VnCSCC3)

| | | | |
|-------------|-------------------------------|-------------|-------------|
| RZ/G2H √ | RZ/G2M V1.3, RZ/G2M V3.0 √ | RZ/G2N √ | RZ/G2E √ |
|-------------|-------------------------------|-------------|-------------|

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25 to 16 | GCBMUL [9:0] | H'064 | R/W | Cb Multiplication Coefficient for G Data Calculation These bits specify the Cb multiplication coefficient for the G data calculation equation in YCbCr-444 → RGB-888 color space conversion. (Initial value: 0.392) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256. |
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 to 0 | BCBMUL [9:0] | H'204 | R/W | Cb Multiplication Coefficient for B Data Calculation These bits specify the Cb multiplication coefficient for the B data calculation equation in YCbCr-444 → RGB-888 color space conversion. (Initial value: 2.017) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256. |

30.2.20.4 Video n YC → RGB Calculation Setting Extension Register 1 (VnCSCE1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

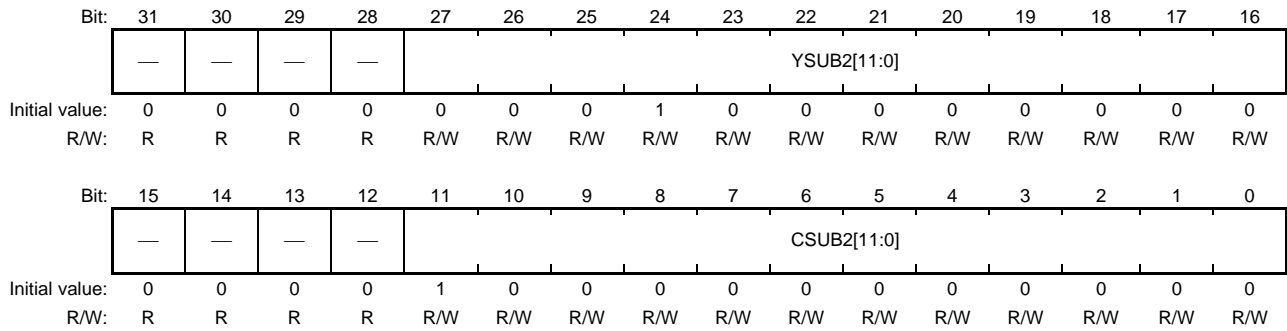
| | | | | | | | | | | | | | | | | |
|----------------|----|----|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | YMUL2[13:0] | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 to 0 | YMUL2[13:0] | H'129F | R/W | Y Multiplication Coefficient 2 for RGB Calculation These bits specify the multiplication coefficient for Y data in YCbCr-444 → RGB-101010/RGB121212 color space conversion. (Initial value: 1.164) Specify an unsigned 12-bit integer obtained by multiplying the desired coefficient value by 4096. |

30.2.20.5 Video n YC → RGB Calculation Setting Extension Register 2 (VnCSCE2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 28 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 to 16 | YSUB[11:0] | H'100 | R/W | Y Subtraction Coefficient 2 for RGB Calculation These bits specify the subtraction coefficient for Y data in YCbCr-444 → RGB-101010/RGB121212 color space conversion. (Initial value: 256) Specify an unsigned 12-bit integer. |
| 15 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 0 | CSUB[11:0] | H'800 | R/W | CbCr Subtraction Coefficient 2 for RGB Calculation These bits specify the subtraction coefficient for Cb and Cr data in YCbCr-444 → RGB-101010/RGB121212 color space conversion. (Initial value: 2048) Specify an unsigned 12-bit integer. |

30.2.20.6 Video n YC → RGB Calculation Setting Extension Register 3 (VnCSRZ/G2E)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:

n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

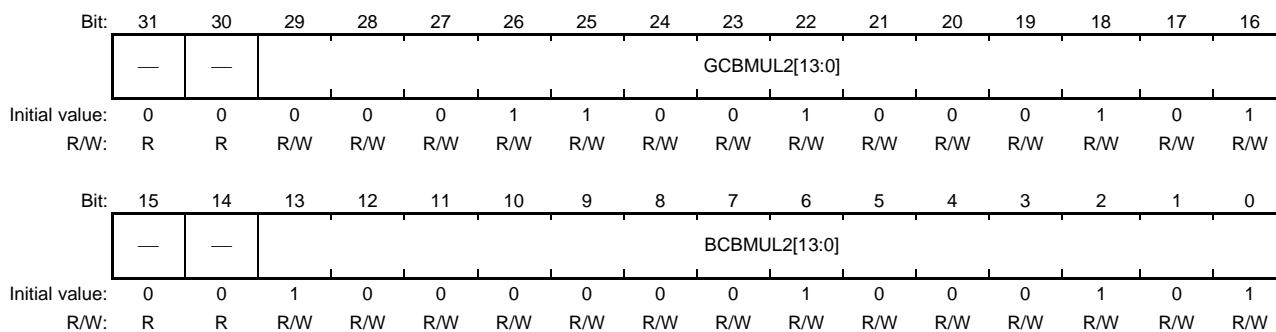
| | | | | | | | | | | | | | | | | |
|----------------|----|----|---------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | RCRMUL2[13:0] | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | GCRMUL2[13:0] | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31, 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 16 | RCRMUL2[13:0] | H'1989 | R/W | Cr Multiplication Coefficient 2 for R Calculation These bits specify the Cr multiplication coefficient for the R data calculation equation in YCbCr-444 → RGB-101010/RGB121212 color space conversion. (Initial value: 1.596) Specify an unsigned 14-bit integer obtained by multiplying the desired coefficient value by 4096. |
| 15, 14 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 to 0 | GCRMUL2[13:0] | H'0D02 | R/W | Cr Multiplication Coefficient 2 for G Calculation These bits specify the Cr multiplication coefficient for the G data calculation equation in YCbCr-444 → RGB-101010/RGB121212 color space conversion. (Initial value: 0.813) Specify an unsigned 14-bit integer obtained by multiplying the desired coefficient value by 4096. |

30.2.20.7 Video n YC → RGB Calculation Setting Extension Register 4 (VnCSCE4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31, 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 16 | GCBMUL2[13:0] | H'0645 | R/W | Cb Multiplication Coefficient 2 for G Calculation These bits specify the Cb multiplication coefficient for the G data calculation equation in YCbCr-444 → RGB-101010/RGB121212 color space conversion. (Initial value: 0.392) Specify an unsigned 14-bit integer obtained by multiplying the desired coefficient value by 4096. |
| 15, 14 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 to 0 | BCBMUL2[13:0] | H'2045 | R/W | Cb Multiplication Coefficient 2 for B Calculation These bits specify the Cb multiplication coefficient for the B data calculation equation in YCbCr-444 → RGB-101010/RGB121212 color space conversion. (Initial value: 2.017) Specify an unsigned 14-bit integer obtained by multiplying the desired coefficient value by 4096. |

30.2.21 UDS Control Registers

Note: The following operators, notations are defined for explaining the functions of UDS.

$\langle x \rangle$: Discard decimal places of value x

30.2.21.1 Video n Scaling Control Register (VnUDS_CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0, 1, 4, and 5 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

| | | | | | | | | | | | | | | | | |
|----------------|----|-----|----|-------|----|----|----|----|----|----|----|-----|----|--------|-------|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | AMD | — | BLADV | — | — | — | — | — | — | — | BC | — | NE_RCR | NE_GY | NE_BCB |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R | R/W | R | R | R | R | R | R | R | R/W | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 30 | AMD | B'0 | R/W | Pixel Count at Scale-Up Specifies the number of pixels generated through scale-up in the UDS. This bit setting is ignored for scale-down. 0: Pixel count after scale-up is $1 + \langle n - 1 \rangle \times \text{scale-up factor}$ 1: Pixel count after scale-up is $\langle n \rangle \times \text{scale-up factor}$ Note: n: Number of pixels input to UDS |
| 29 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 28 | BLADV | B'0 | R/W | Bilinear or Nearest Neighbor Interpolation Characteristic Control Controls the characteristics of bilinear or nearest neighbor interpolation. Setting this bit to 1 improves the aliasing characteristics at $\times 1/2$ to $1/8$ scaling (VnUDS_SCALE.VMANT/HMANT = 2 to 8). Note that the apparent resolution around edges may deteriorate. In multi-tap mode, this control is not available; to use this control, set the BC bit to 0. |
| 27 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 | BC | B'0 | R/W | Pixel Component Interpolation Method at Scale-Up/Down Specifies the method for interpolating pixel components at scale-up/-down. 0: Bilinear or nearest neighbor interpolation method is used 1: Interpolation method equivalent to 4 to 17 taps in accordance with the scaling factor is used (multi-tap mode) |

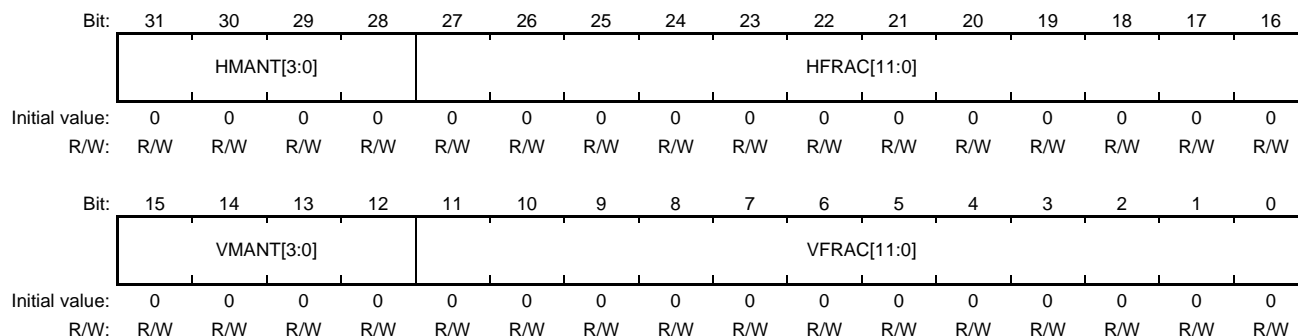
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 19 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 18 | NE_RCR | B'0 | R/W | R/Cr Interpolation Method When Bilinear/Nearest Neighbor Interpolation is Selected Specifies the interpolation method of the R/Cr component when bilinear/nearest neighbor interpolation is selected (BC = 0). 0: Bilinear method 1: Nearest neighbor method* |
| 17 | NE_GY | B'0 | R/W | G/Y Interpolation Method When Bilinear/Nearest Neighbor Interpolation is Selected Specifies the interpolation method of the G/Y component when bilinear/nearest neighbor interpolation is selected (BC = 0). 0: Bilinear method 1: Nearest neighbor method* |
| 16 | NE_BCB | B'0 | R/W | B/Cb Interpolation Method When Bilinear/Nearest Neighbor Interpolation is Selected Specifies the interpolation method of the B/Cb component when bilinear/nearest neighbor interpolation is selected (BC = 0). 0: Bilinear method 1: Nearest neighbor method* |
| 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

Note: * This method can be used only when the scale-up/-down factor is 1/1 to 1/4.

30.2.21.2 Video n Scaling Factor Register (VnUDS_SCALE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0, 1, 4, and 5 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 28 | HMANT[3:0] | H'0 | R/W | <p>Multiplier (Integral Part) of Horizontal Scaling Factor</p> <p>These bits specify the integral part of the horizontal scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size.</p> <p>A value from H'0 to H'F can be specified. Select a value within the range shown in Table 30.19.</p> |
| 27 to 16 | HFRAC[11:0] | H'000 | R/W | <p>Multiplier (Fractional Part) of Horizontal Scaling Factor</p> <p>These bits specify the fractional part of the horizontal scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size.</p> <p>A value from H'800 to H'FFF can be specified when an image is upscaled (the HMANT value is 0) in the horizontal direction. A value from H'000 to H'FFF can be specified when an image is downscaled (the HMANT value is not 0) in the horizontal direction. Select a value within the range shown in Table 30.19.</p> |
| 15 to 12 | VMANT[3:0] | H'0 | R/W | <p>Multiplier (Integral Part) of Vertical Scaling Factor</p> <p>These bits specify the integral part of the vertical scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size.</p> <p>A value from H'0 to H'F can be specified. Select a value within the range shown in Table 30.20.</p> |
| 11 to 0 | VFRAC[11:0] | H'000 | R/W | <p>Multiplier (Fractional Part) of Vertical Scaling Factor</p> <p>These bits specify the fractional part of the vertical scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size.</p> <p>A value from H'556 to H'FFF can be specified when an image is upscaled (the VMANT value is 0) in the vertical direction. A value from H'000 to H'FFF can be specified when an image is downscaled (the VMANT value is not 0) in the vertical direction. Select a value within the range shown in Table 30.20.</p> |

The HMANT and HFRAC bits set the scale-up/-down factor for an image in the horizontal direction, and the VMANT and VFRAC bits set the scale-up/-down factor for an image in the vertical direction. The UDS operation switches between horizontal scale-up and horizontal scale-down according to the HMANT and HFRAC bit settings, as shown in Table 30.19. (Setting a value outside the ranges of Table 30.19 in UDS operation is prohibited). Table 30.36 and 30.20 show the settings in the horizontal and vertical directions.

Note that settings for scaling in the horizontal direction and settings for scaling in the vertical direction can be made independently. Therefore, a setting for scale-up in the horizontal direction and scale-down in the vertical direction is possible. In such a case, because the formula (described later) for obtaining the image size after scale-up/-down is different between scale-up and scale-down, a formula matching the scale-up or scale-down operation should be selected independently for the horizontal direction and vertical direction.

Table 30.19 Switching of Horizontal Scale-Up/Down Operation According to HMANT and HFRAC Bit Settings

| HMANT | HFRAC | UDS Operation |
|------------|----------------|-------------------------------|
| H'0 | H'800 to H'FFF | Scale-up |
| H'1 | H'000 | Same size (no scale-up/-down) |
| | H'001 to H'FFF | Scale-down |
| H'2 to H'F | H'000 to H'FFF | |

Table 30.20 Switching of Vertical Scale-Up/Down Operation According to VMANT and VFRAC Bit Settings

| VMANT | VFRAC | UDS Operation |
|------------|----------------|-------------------------------|
| H'0 | H'556 to H'FFF | Scale-up |
| H'1 | H'000 | Same size (no scale-up/-down) |
| | H'001 to H'FFF | Scale-down |
| H'2 to H'F | H'000 to H'FFF | |

Described here is the method for calculating the size of the upscaled/downscaled image that was obtained based on this register setting. First, define the variables necessary for calculating the horizontal size of the upscaled/downscaled image, as shown below.

$$hscale = \frac{4096}{4096 \times m_h + f_h}$$

m_h is the value of VnUDS_SCALE.HMANT, and f_h is the value of VnUDS_SCALE.HFRAC. This formula expresses the estimate of the scale-up/-down factor processed by the UDS. If the horizontal size of the image before scale-up/-down is set as $hsize_{org}$, the horizontal size of the image after scale-up/-down can be roughly obtained through $hsize_{org} \times hscale$.

Similarly, define the variables necessary for calculating the vertical size of the upscaled/downscaled image.

$$vscale = \frac{4096}{4096 \times m_v + f_v}$$

When setting m_v as the value of VnUDS_SCALE.VMANT, f_v as the value of VnUDS_SCALE.VFRAC, and $vsize_{org}$ as the vertical size of the image before scale-up/-down, the vertical size of the image after scale-up/-down can be roughly obtained through $vsize_{org} \times vscale$.

When the UDS performs scale-down with the settings of Table 30.19, using the variables defined so far, the horizontal size of the downscaled image $hsizedown_{scaled}$ and the vertical size of the downscaled image $vsize_{down_{scaled}}$ become as follows:

$$hsize_{down_scaled} = \left\langle 1 + \left[\left\langle 1 + \frac{\langle hsize_{org} - 1 \rangle}{m_h'} \right\rangle - 1 \right] \times m_h' \times hscale \right\rangle$$

$$vsize_{down_scaled} = \left\langle 1 + \left[\left\langle 1 + \frac{\langle vsize_{org} - 1 \rangle}{m_v'} \right\rangle - 1 \right] \times m_v' \times vscale \right\rangle$$

Since the division of hscale and vscale has to be executed at the end of the above formulas, respectively, formulas considering the order of operation become as follows:

$$hsize_{down_scaled} = \left\langle 1 + \left[\left[\left\langle 1 + \frac{\langle hsize_{org} - 1 \rangle}{m_h'} \right\rangle - 1 \right] \times m_h' \times 4096 \right] / (4096 \times m_h + f_h) \right\rangle$$

$$vsize_{down_scaled} = \left\langle 1 + \left[\left[\left\langle 1 + \frac{\langle vsize_{org} - 1 \rangle}{m_v'} \right\rangle - 1 \right] \times m_v' \times 4096 \right] / (4096 \times m_v + f_v) \right\rangle$$

The value of m_h' or m_v' , which is shown in Table 30.21, changes according to the setting of VnUDS_SCALE.HMANT or VnUDS_SCALE.VMANT.

Table 30.21 m_h' or m_v' Setting

| VnUDS_SCALE.HMANT Setting (VnUDS_SCALE.VMANT Setting) | m_h' (m_v') |
|--|----------------------|
| 1 to 3 | 1 |
| 4 to 7 | 2 |
| 8 to 15 | 4 |

When the UDS performs scale-up with the settings of Table 30.19 and VnUDS_CTRL.AMD is 0, the horizontal size of the upscaled image $hsize_{up_scaled}$ and the vertical size of the upscaled image $vsize_{up_scaled}$ become as follows:

$$hsize_{up_scaled} = \langle 1 + (hsize_{org} - 1) \times hscale \rangle$$

$$vsize_{up_scaled} = \langle 1 + (vsize_{org} - 1) \times vscale \rangle$$

Similar to the scale-down case, formulas considering the division of hscale and vscale become as follows:

$$hsize_{up_scaled} = \langle 1 + ((hsize_{org} - 1) \times 4096) / (4096 \times m_h + f_h) \rangle$$

$$vsize_{up_scaled} = \langle 1 + ((vsize_{org} - 1) \times 4096) / (4096 \times m_v + f_v) \rangle$$

When $VnUDS_CTRL.AMD$ is 1, the horizontal size of the upscaled image $hsize_{up_scaled}$ and the vertical size of the upscaled image $vsize_{up_scaled}$ become as follows:

$$hsize_{up_scaled} = \langle hsize_{org} \times hscale \rangle$$

$$vsize_{up_scaled} = \langle vsize_{org} \times vscale \rangle$$

After considering the division of $hscale$ and $vscale$, the formulas become as follows:

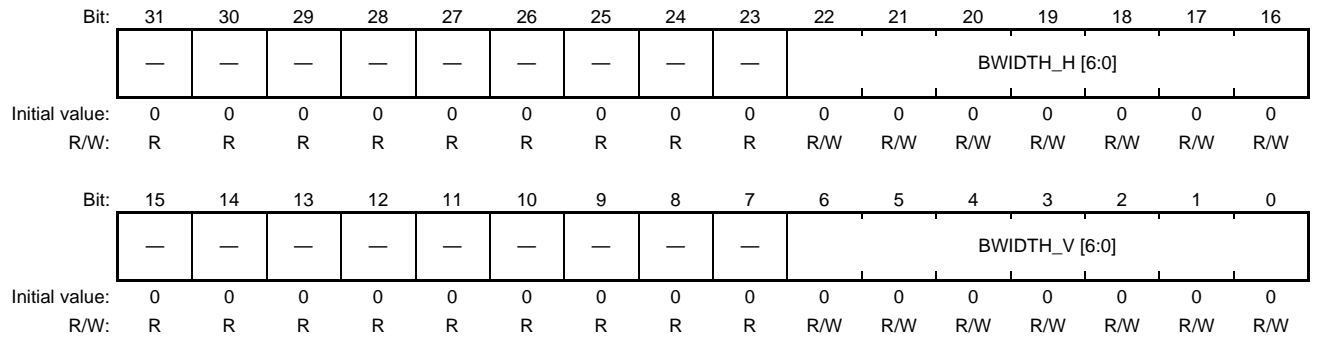
$$hsize_{up_scaled} = \langle (hsize_{org} \times 4096) / (4096 \times m_h + f_h) \rangle$$

$$vsize_{up_scaled} = \langle (vsize_{org} \times 4096) / (4096 \times m_v + f_v) \rangle$$

30.2.21.3 Video n Passband Register (VnUDS_PASS_BWIDTH)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0, 1, 4, and 5 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 23 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 22 to 16 | BWIDTH_H [6:0] | H'00 | R/W | Horizontal Signal Passband at Image Scale-Up/Down Set these bits following the method described later. |
| 15 to 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 to 0 | BWIDTH_V [6:0] | All 0 | R/W | Vertical Signal Passband at Image Scale-Up/Down Set these bits following the method described later. |

The method for setting the passband in the horizontal direction of an image is described. When the VnUDS_SCALE.HMANT bits for horizontal scale-up/-down factor setting are not 0, set the BWIDTH_H bits according to the following formula. When the VnUDS_SCALE.HMANT bits are 0, set 64 in the BWIDTH_H bits.

$$BWIDTH\ H = \left\langle 64 \times \frac{4096 \times m_h'}{4096 \times m_h + f_h} \right\rangle \quad (VnUDS_SCALE.HMANT \neq 0)$$

$$BWIDTH\ H = 64 \quad (VnUDS_SCALE.HMANT = 0)$$

m_h is the value of VnUDS_SCALE.HMANT, and f_h is the value of VnUDS_SCALE.HFRAC. For the m_h' value, see Table 30.21. The method for setting the passband in the vertical direction of an image is similar to that for the horizontal direction described earlier. Since only the correspondence relationship of the registers is changed as shown below, replace the variables in the previous explanation as shown below when reading.

BWIDTH_H → BWIDTH_V

m_h' → m_v'

m_h → m_v

f_h → f_v

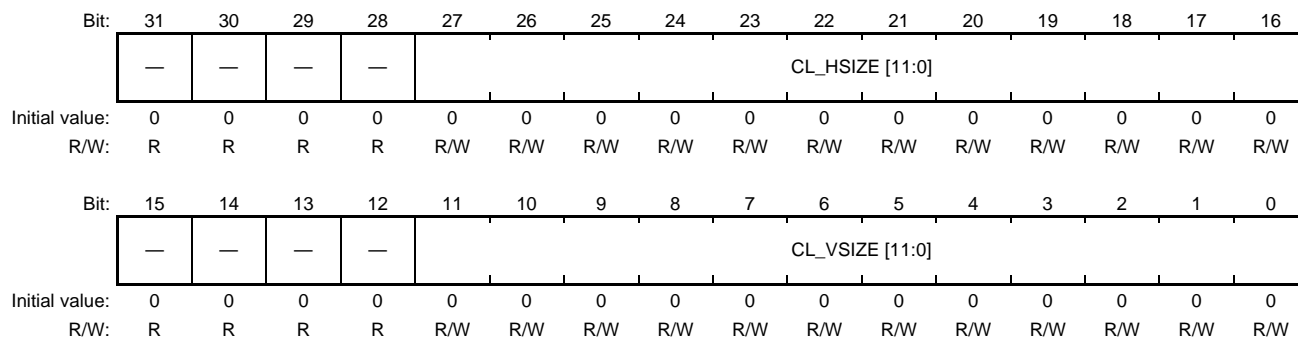
VnUDS_SCALE.HMANT → VnUDS_SCALE.VMANT

VnUDS_SCALE.HFRAC → VnUDS_SCALE.VFRAC

30.2.21.4 Video n UDS Output Size Clipping Register (VnUDS_CLIP_SIZE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0, 1, 4, and 5 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|---|
| 31 to 28 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 to 16 | CL_HSIZE [11:0] | All 0 | R/W | Clipping Size of Horizontal Pixel Count after Scale-Up/-Down The horizontal width of an image output from the scaling filter is adjusted (clipped or padded) to match the pixel count set in the CL_HSIZE bits. The setting range is 4 to 2,048 in a scale-down operation (see Table 30.19) and 4 to 2,048 in a scale-up operation. These bits always have to be set when using the UDS, regardless of the scale-up, scale-down, or no-scaling setting by the VnUDS_SCALE register. |
| 15 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 0 | CL_VSIZE [11:0] | All 0 | R/W | Clipping Size of Vertical Pixel Count after Scale-Up/-Down The vertical width of an image output from the scaling filter is adjusted (clipped or padded) to match the pixel count set in the CL_VSIZE bits. The setting range is 4 to 2,048 in a scale-down operation (see Table 30.20) and 4 to 2,048 in a scale-up operation. These bits always have to be set when using the UDS, regardless of the scale-up, scale-down, or no-scaling setting by the VnUDS_SCALE register. |

Figure 30.13 shows the configuration of the UDS. The UDS consists of a scaling filter and clipping circuit, such as the configuration shown in Figure 30.13. The scaling filter and clipping circuit are independent of each other.

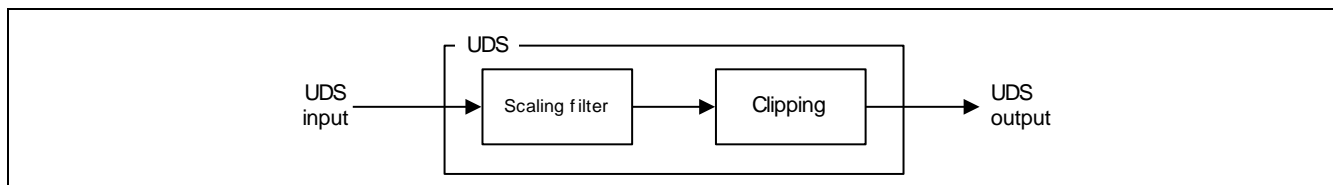


Figure 30.13 UDS Configuration

The size of the image actually output by the scaling filter (refer to section 30.2.21.2 for calculation) is determined from the size of the image input to the scaling filter and the VnUDS_SCALE setting; that is, two types of image are output by the UDS according to the relationship between the size of the image actually output from the scaling filter ($hsize_{scaled}$ and $vsize_{scaled}$) and this register setting as shown in Figure 30.14.

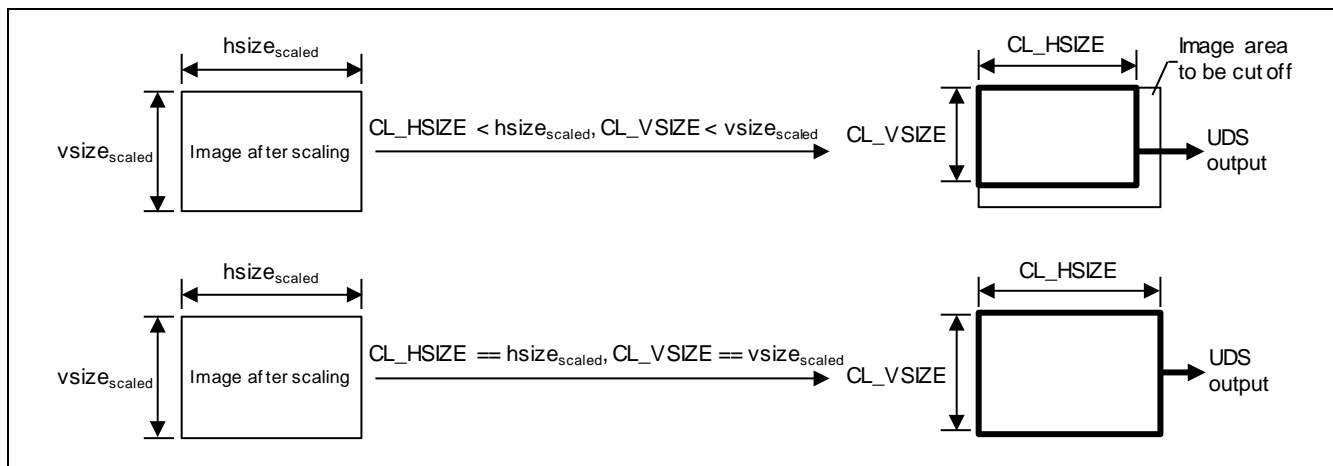


Figure 30.14 UDS Output Image for Each CL_HSIZE/VSIZE Setting

When the settings of the CL_HSIZE and CL_VSIZE bits are smaller than the horizontal and vertical pixel counts ($hsize_{scaled}$ and $vsize_{scaled}$) actually output by the scaling filter, the upscaled/downscaled image is clipped to become the UDS output image.

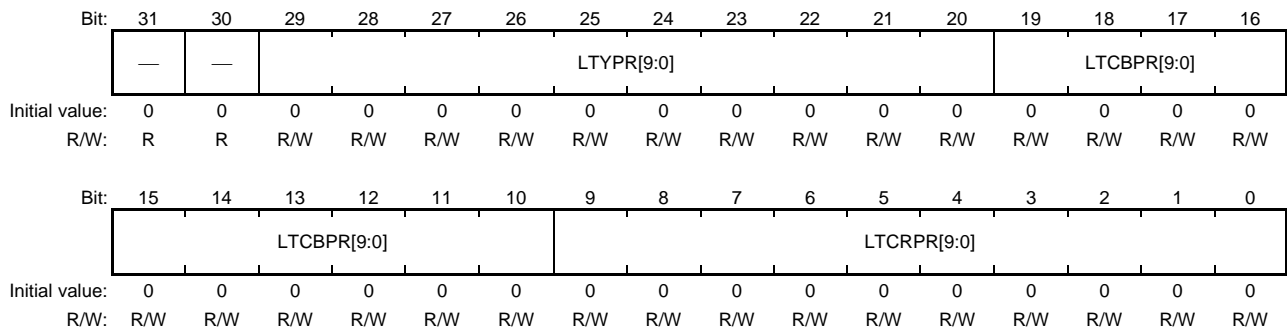
When the settings of the CL_HSIZE and CL_VSIZE bits are equal to the horizontal and vertical pixel counts ($hsize_{scaled}$ and $vsize_{scaled}$) actually output by the scaling filter, the image actually output by the scaling filter becomes the UDS output image without change.

Do not set the CL_HSIZE and CL_VSIZE bits to values that satisfy $CL_HSIZE > hsize_{scaled}$ or $CL_VSIZE > vsize_{scaled}$.

30.2.22 Video n Lookup Table Pointer (VnLUTP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



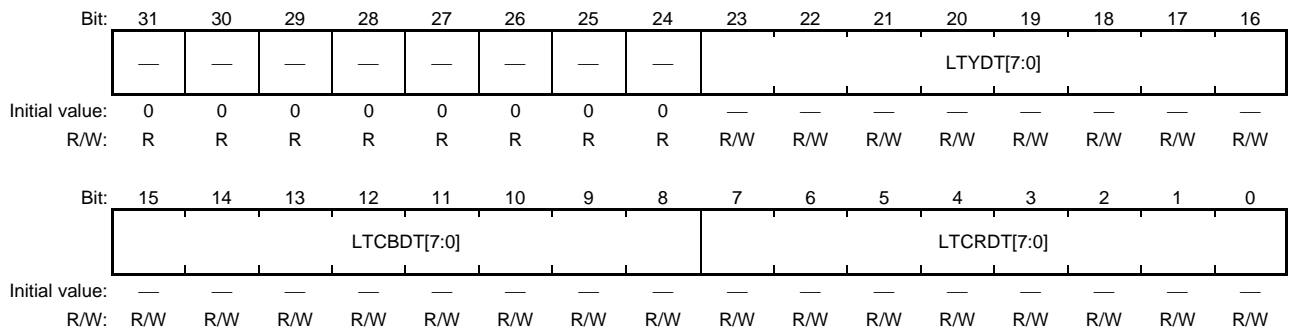
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31, 30 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 20 | LTYPR[9:0] | All 0 | R/W | Lookup Table Y Pointer These bits set the LUT pointer to Y and R data after color space conversion. |
| 19 to 10 | LTCBPR[9:0] | All 0 | R/W | Lookup Table Cb Pointer These bits set the LUT pointer to Cb and G data after color space conversion. |
| 9 to 0 | LTCRPR[9:0] | All 0 | R/W | Lookup Table Cr Pointer These bits set the LUT pointer to Cr and G data after color space conversion. |

Note: Set the LUT pointer to the upper 10 bits of the 12-bit data for the color space conversion result.
 The access pointer to the LUT will be automatically incremented by writing to the VnLUTD register.
 There will be no automatic increment at read access.

30.2.23 Video n Lookup Table Data Register (VnLUTD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | LTYDT[7:0] | — | R/W | Lookup Table Y Data These bits set the LUT conversion data for Y and R data after color space conversion. |
| 15 to 8 | LTCBDT[7:0] | — | R/W | Lookup Table Cb Data These bits set the LUT conversion data for Cb and G data after color space conversion. |
| 7 to 0 | LTCRDT[7:0] | — | R/W | Lookup Table Cr Data These bits set the LUT conversion data for Cr and B data after color space conversion. |

Note: The 8-bit data after LUT conversion is subjected to upper shift to carry out capture control as 12-bit data. As for reading from the VnLUTD register, the data read the second time, including dummy read, after the VnLUTP register has been set is valid.

30.2.24 RGB-YC Conversion Coefficient Registers

Color space conversion from RGB to YC is done with the following formula. Each of the coefficients can be set with the registers.

$$\begin{aligned}
 Y &= YCLRP \times R + YCLGP \times G + YCLBP \times B + YCLAP \\
 Cb &= CBCLRP \times R + CBCLGP \times G + CBCLBP \times B + CBCLAP \\
 Cr &= YCLRP \times R + YCLGP \times G + YCLBP \times B + YCLAP
 \end{aligned}$$

Note: Set the coefficients for ITU-R BT.601 (8 bits) as the initial value.

$$\begin{aligned}
 Y &= 0.257 \times R + 0.504 \times G + 0.098 \times B + 16 \\
 Cb &= -0.148 \times R - 0.291 \times G + 0.439 \times B + 128 \\
 Cr &= 0.439 \times R - 0.368 \times G - 0.071 \times B + 128
 \end{aligned}$$

30.2.24.1 Video n RGB → Y Calculation Setting Register 1 (VnYCCR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

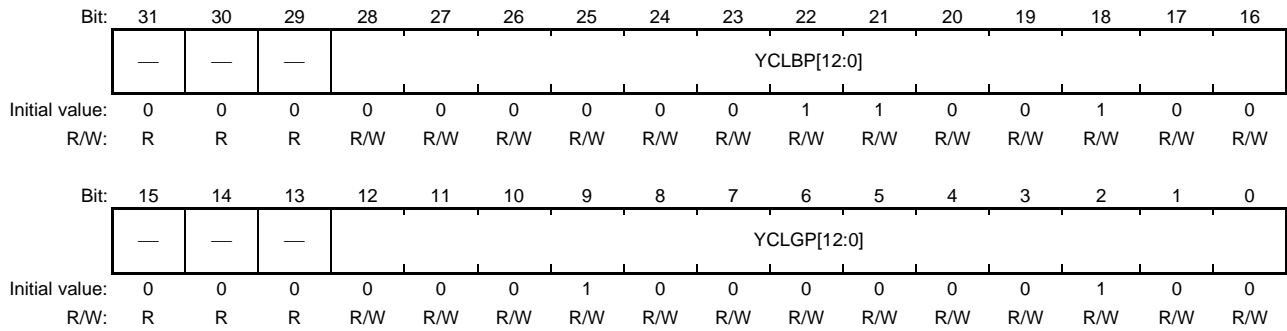
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | YCLRP[12:0] | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 0 | YCLRP[12:0] | H'0107 | R/W | R Multiplication Coefficient for Y Calculation These bits specify the R multiplication coefficient for the Y data calculation equation in RGB-888 → YCbCr-444 color space conversion. (Initial value: 303) The MSB is a sign bit. |

30.2.24.2 Video n RGB → Y Calculation Setting Register 2 (VnYCCR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | YCLBP[12:0] | H'0064 | R/W | B Multiplication Coefficient for Y Calculation These bits specify the B multiplication coefficient for the Y data calculation equation in RGB-888→YCbCr-444 color space conversion. (Initial value: 100) The MSB is a sign bit. |
| 15 to 13 | — | B'000 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 0 | YCLGP[12:0] | H'0204 | R/W | G Multiplication Coefficient for Y Calculation These bits specify the G multiplication coefficient for the Y data calculation equation in RGB-888→YCbCr-444 color space conversion. (Initial value: 516) The MSB is a sign bit. |

30.2.24.3 Video n RGB → Y Calculation Setting Register 3 (VnYCCR3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:

n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

| | | | | | | | | | | | | | | | | |
|----------------|--------|----|----|-------------|-------------|-----|-----|--------|-----|-----|-----|-----|-----|-----|-----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | YEXPEN | — | — | YCLSFT[4:0] | | | | YCLHEN | — | — | — | — | — | — | — | YCLCEN |
| Initial value: | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | | | | YCLAP[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 | YEXPEN | B'0 | R/W | Y Calculation Sign Extension Enable This bit controls the sign extension for Y data calculation in RGB888 → YCbCr444 color space conversion. 0: Disables YC conversion sign bit. 1: Enables YC conversion sign bit. |
| 30, 29 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 24 | YCLSFT[4:0] | H'0A | R/W | Y Calculation Shift Down Volume These bits set the amount of down shift for Y calculation in RGB888 → YCbCr444 color space conversion (Initial value: 10) Unit: Bit shift count |
| 23 | YCLHEN | B'0 | R/W | Y Calculation Shift Down Result Round-Off Enable This bit enables round-off process for Y data calculation in RGB888 → YCbCr444 color space conversion 0: Round down to down shift process 1: Round-off to down shift process is enabled. |
| 22 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | YCLCEN | B'0 | R/W | Y Calculation Data Clip Enable This bit enables data clipping process for Y data calculation in RGB888 → YCbCr444 color space conversion. 0: Data clipping process is disabled. 1: Data clipping process is enabled. |
| 15 to 12 | — | H'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 0 | YCLAP[11:0] | H'010 | R/W | Y Calculation Data Normalized Additional Value These bits set the Y data addition constant for RGB888 → YCbCr444 color space conversion. (Initial value: 16) |

30.2.24.4 Video n RGB → Cb Calculation Setting Register 1 (VnCBCCR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

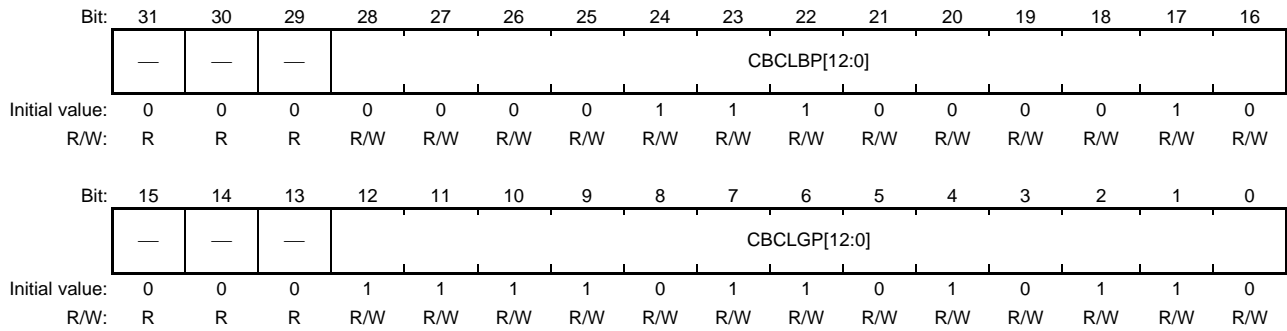
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|--------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | CBCLRP[12:0] | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 0 | CBCLRP[12:0] | H'1F68 | R/W | R Multiplication Coefficient for Cb Calculation These bits specify the R multiplication coefficient for the Cb data calculation equation in RGB-888 → YCbCr-444 color space conversion. (Initial value: -152) The MSB is a sign bit. |

30.2.24.5 Video n RGB → Cb Calculation Setting Register 2 (VnCBCCR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 29 | — | B'000 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | CBCLBP[12:0] | H'01C2 | R/W | B Multiplication Coefficient for Cb Calculation These bits specify the B multiplication coefficient for the Cb data calculation equation in RGB-888 → YCbCr-444 color space conversion. (Initial value: 450) The MSB is a sign bit. |
| 15 to 13 | — | B'000 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 0 | CBCLGP[12:0] | H'1ED6 | R/W | G Multiplication Coefficient for Cb Calculation These bits specify the B multiplication coefficient for the Cb data calculation equation in RGB-888 → YCbCr-444 color space conversion. (Initial value: -298) The MSB is a sign bit. |

30.2.24.6 Video n RGB → Cb Calculation Setting Register 3 (VnCBCCR3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:

n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

| | | | | | | | | | | | | | | | | |
|----------------|-------------|----|----|--------------|--------------|-----|-----|-------------|-----|-----|-----|-----|-----|-----|-----|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CBEXP EN | — | — | CBCLSFT[4:0] | | | | CBCLH EN | — | — | — | — | — | — | — | CBCLC EN |
| Initial value: | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | CBCLAP[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 | CBEXPEN | B'0 | R/W | Cb Calculation Sign Extension Enable This bit controls the sign extension for Cb data calculation in RGB888 → YCbCr444 color space conversion. 0: Disables YC conversion sign bit. 1: Enables YC conversion sign bit. |
| 30, 29 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 24 | CBCLSFT[4:0] | H'0A | R/W | Cb Calculation Shift Down Volume These bits set the amount of down shift for Cb calculation in RGB888 → YCbCr444 color space conversion (Initial value: 10) Unit: Bit shift count |
| 23 | CBCLHEN | B'0 | R/W | Cb Calculation Shift Down Result Round-Off Enable This bit enables round-off process for Cb data calculation in RGB888 → YCbCr444 color space conversion 0: Round down to down shift process 1: Round-off to down shift process is enabled. |
| 22 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | CBCLCEN | B'0 | R/W | Cb Calculation Data Clip Enable This bit enables data clipping process for Cb data calculation in RGB888 → YCbCr444 color space conversion. 0: Data clipping process is disabled. 1: Data clipping process is enabled. |
| 15 to 12 | — | H'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 0 | CBCLAP[11:0] | H'080 | R/W | Cb Calculation Data Normalized Additional Value These bits set the Cb data addition constant for RGB888 → YCbCr444 color space conversion. (Initial value: 128) |

30.2.24.7 Video n RGB → Cr Calculation Setting Register 1 (VnCRCCR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

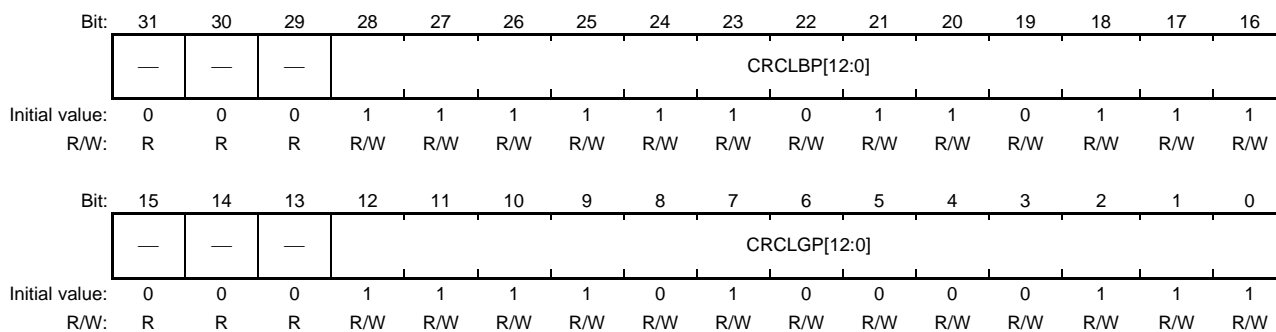
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|--------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | CRCLRP[12:0] | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|---|
| 31 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 0 | CRCLRP[12:0] | H'01C2 | R/W | R Multiplication Coefficient for Cr Calculation These bits specify the R multiplication coefficient for the Cr data calculation equation in RGB-888 → YCbCr-444 color space conversion. (Initial value: 450) The MSB is a sign bit. |

30.2.24.8 Video n RGB → Cr Calculation Setting Register 2 (VnCRCCR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:
 n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 29 | — | B'000 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | CRCLBP[12:0] | H'1FB7 | R/W | B Multiplication Coefficient for Cr Calculation These bits specify the B multiplication coefficient for the Cr data calculation equation in RGB-888 → YCbCr-444 color space conversion. (Initial value: -73) The MSB is a sign bit. |
| 15 to 13 | — | B'000 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 0 | CRCLGP[12:0] | H'1E87 | R/W | G Multiplication Coefficient for Cr Calculation These bits specify the B multiplication coefficient for the Cr data calculation equation in RGB-888 → YCbCr-444 color space conversion. (Initial value: -377) The MSB is a sign bit. |

30.2.24.9 Video n RGB → Cr Calculation Setting Register 3 (VnCRCCR3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: Availability of channels:

n = 0 to 7 (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N), n = 4 to 5 (RZ/G2E)

| | | | | | | | | | | | | | | | | | |
|----------------|-------------|----|----|--------------|--------------|-----|-----|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | CREXP EN | — | — | CRCLSFT[4:0] | | | | CRCLH EN | — | — | — | — | — | — | — | — | CRCLC EN |
| Initial value: | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R/W | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | CRCLAP[11:0] | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 | CREXPEN | B'0 | R/W | Cr Calculation Sign Extension Enable This bit controls the sign extension for Cr data calculation in RGB888 → YCbCr444 color space conversion. 0: Disables YC conversion sign bit. 1: Enables YC conversion sign bit. |
| 30, 29 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 24 | CRCLSFT[4:0] | H'0A | R/W | Cr Calculation Shift Down Volume These bits set the amount of down shift for Cr calculation in RGB888 → YCbCr444 color space conversion (Initial value: 10) Unit: Bit shift count |
| 23 | CRCLHEN | B'0 | R/W | Cr Calculation Shift Down Result Round-Off Enable This bit enables round-off process for Cr data calculation in RGB888 → YCbCr444 color space conversion 0: Round down to down shift process 1: Round-off to down shift process is enabled. |
| 22 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | CRCLCEN | B'0 | R/W | Cr Calculation Data Clip Enable This bit enables data clipping process for Cr data calculation in RGB888 → YCbCr444 color space conversion. 0: Data clipping process is disabled. 1: Data clipping process is enabled. |
| 15 to 12 | — | H'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 0 | CRCLAP[11:0] | H'080 | R/W | Cr Calculation Data Normalized Additional Value These bits set the Cr data addition constant for RGB888 → YCbCr444 color space conversion. (Initial value: 128) |

30.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

30.3.1 Input Interface

The VIN captures video data in the ITU-R BT.601, ITU-R BT.656, ITU-R BT.1358, or ITU-R BT.709 interface and stores it in external memory. The module has four input-interface channels. The interface and data format can be set as desired for each video channel.

The following tables show the interface and data format supported in each product and its channels.

Note: When the ITU-R BT.709 interface is in use, the maximum frequency of the video clock of Digital pin is 100 MHz.

Table 30.22 Video Channels and Supported Interfaces (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N Channels 0 to 3)

| | | | | Channel 0 | Channel 1 | Channel 2 | Channel 3 |
|------------------|--------------|--------------------------------|-----------------------------|-----------|-----------|-----------|-----------|
| Input interface | MIPI CSI-2 | YCbCr-422 | 8 bits | Supported | Supported | Supported | Supported |
| | | | 10 bits | Supported | Supported | Supported | Supported |
| | | RGB-888 | 24 bits | Supported | Supported | Supported | Supported |
| | | 8-bit user defined data (RAW8) | 8 bits | Supported | Supported | Supported | — |
| | | Embedded | 8 bits | — | — | — | Supported |
| Output interface | RGB output | RGB-565 | 16 bits | Supported | Supported | Supported | Supported |
| | | ARGB-1555 | 16 bits | Supported | Supported | Supported | Supported |
| | | RGB-888 | 32 bits | Supported | Supported | Supported | Supported |
| | | ARGB-8888 | 32 bits | Supported | Supported | Supported | Supported |
| | YCbCr output | YCbCr-422 multiplexed | 8 bits | Supported | Supported | Supported | Supported |
| | | | 10 bits | Supported | Supported | Supported | Supported |
| | | YCbCr-420 separated Y/CbCr | Y: 8 bits CbCr: 8 bits | Supported | Supported | — | — |
| | | YCbCr-422 separated Y/CbCr | Y: 8 bits CbCr: 8 bits | Supported | Supported | Supported | Supported |
| | | | Y: 10 bits CbCr: 8 bits | Supported | Supported | Supported | Supported |
| | | | Y: 10 bits CbCr: 10 bits | Supported | Supported | Supported | Supported |
| | | YCbCr-422 separated Only Y | 8 bits | Supported | Supported | Supported | Supported |
| | | | 10 bits | Supported | Supported | Supported | Supported |

- Notes:
1. When capturing progressive images or MIPI CSI-2 input data, be sure to enable the internal field signal generation function.
 2. Dithering is performed for RGB-888→RGB-565 conversion.
 3. RGB data is converted and transferred to memory from the video module after 8-bit precision conversion.
 4. For details of output interfaces, see section 30.3.10, Output Data Format.

Table 30.23 Video Channels and Supported Interfaces (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N Channels 4 to 7)

| | | | | Channel 4 | Channel 5 | Channel 6 | Channel 7 |
|--------------------------------|-------------------------------|----------------------------|-----------------------------|-------------------------|-------------------------|-----------|-----------|
| Input interface | ITU-R BT.656 | YCbCr-422 | 8 bits | Supported | Supported | — | — |
| | | | 10 bits | Supported | Supported | — | — |
| | | | 12 bits | Supported | Supported | — | — |
| | ITU-R BT.1358 | YCbCr-422 | 16 bits | Supported | Supported | — | — |
| | | | 20 bits | Supported | — | — | — |
| | | | 24 bits | Supported | — | — | — |
| | ITU-R BT.601/ ITU-R BT.709 | YCbCr-422 | 8 bits | Supported | Supported | — | — |
| | | | 10 bits | Supported | Supported | — | — |
| | | | 12 bits | Supported | Supported | — | — |
| | | RGB-666 | 18 bits | Supported | — | — | — |
| | | RGB-888 | 24 bits | Supported | — | — | — |
| | RAW input | RAW12 | 12 bits | Supported* ⁵ | Supported* ⁵ | — | — |
| | MIPI CSI-2 | YCbCr-422 | 8 bits | Supported | Supported | Supported | Supported |
| | | | 10 bits | Supported | Supported | Supported | Supported |
| | | RGB-888 | 24 bits | Supported | Supported | Supported | Supported |
| 8-bit user defined data (RAW8) | | 8 bits | Supported | Supported | Supported | — | |
| Embedded | | 8 bits | — | — | — | Supported | |
| Output interface | RGB output | RGB-565 | 16 bits | Supported | Supported | Supported | Supported |
| | | ARGB-1555 | 16 bits | Supported | Supported | Supported | Supported |
| | | RGB-888 | 32 bits | Supported | Supported | Supported | Supported |
| | | ARGB-8888 | 32 bits | Supported | Supported | Supported | Supported |
| | YCbCr output | YCbCr-422 multiplexed | 8 bits | Supported | Supported | Supported | Supported |
| | | | 10 bits | Supported | Supported | Supported | Supported |
| | | | 12 bits | Supported* ⁵ | Supported* ⁵ | — | — |
| | | YCbCr-420 separated Y/CbCr | Y: 8 bits CbCr: 8 bits | Supported | Supported | — | — |
| | | YCbCr-422 separated Y/CbCr | Y: 8 bits CbCr: 8 bits | Supported | Supported | Supported | Supported |
| | | | Y: 10 bits CbCr: 8 bits | Supported | Supported | Supported | Supported |
| | | | Y: 10 bits CbCr: 10 bits | Supported | Supported | Supported | Supported |
| | | | Y: 12 bits CbCr: 8 bits | Supported* ⁵ | Supported* ⁵ | — | — |
| | | | Y: 12 bits CbCr: 12 bits | Supported* ⁵ | Supported* ⁵ | — | — |
| | | YCbCr-422 separated Only Y | 8 bits | Supported | Supported | Supported | Supported |
| | 10 bits | | Supported | Supported | Supported | Supported | |
| | 12 bits | | Supported* ⁵ | Supported* ⁵ | — | — | |
| | RAW output | RAW12 | 12 bits | Supported* ⁵ | Supported* ⁵ | — | — |

- Notes:
1. When capturing progressive images or MIPI CSI-2 input data, be sure to enable the internal field signal generation function.
 2. Dithering is performed for RGB-888→RGB-565 conversion.
 3. RGB data is converted and transferred to memory from the video module after 8-bit precision conversion.
 4. For details of output interfaces, see section 30.3.10, Output Data Format.
 5. Y 12bit and CbCr 12bit and RAW12 are supported only when using the digital pin.

Table 30.24 Video Channels and Supported Interfaces (RZ/G2E Channels 4 to 5)

| | | | Channel 4 | Channel 5 | |
|--------------------------------|-------------------------------|----------------------------|-----------------------------|-------------------------|-------------------------|
| Input interface | ITU-R BT.656 | YCbCr-422 | 8 bits | Supported | Supported |
| | | | 10 bits | Supported | Supported |
| | | | 12 bits | Supported | Supported |
| | ITU-R BT.1358 | YCbCr-422 | 16 bits | Supported | Supported |
| | | | 20 bits | Supported | — |
| | | | 24 bits | Supported | — |
| | ITU-R BT.601/ ITU-R BT.709 | YCbCr-422 | 8 bits | Supported | Supported |
| | | | 10 bits | Supported | Supported |
| | | | 12 bits | Supported | Supported |
| | | | RGB-666 | 18 bits | Supported |
| | | RGB-888 | 24 bits | Supported | — |
| | RAW input | RAW12 | 12 bits | Supported* ⁵ | Supported* ⁵ |
| | MIPI CSI-2 | YCbCr-422 | 8 bits | Supported | Supported |
| | | | 10 bits | Supported | Supported |
| | | | RGB-888 | 24 bits | Supported |
| 8-bit user defined data (RAW8) | | | 8 bits | Supported | Supported |
| Embedded | | | 8 bits | — | Supported |
| Output interface | RGB output | RGB-565 | 16 bits | Supported | Supported |
| | | ARGB-1555 | 16 bits | Supported | Supported |
| | | RGB-888 | 32 bits | Supported | Supported |
| | | ARGB-8888 | 32 bits | Supported | Supported |
| | YCbCr output | YCbCr-422 multiplexed | 8 bits | Supported | Supported |
| | | | 10 bits | Supported | Supported |
| | | | 12 bits | Supported* ⁵ | Supported* ⁵ |
| | | YCbCr-420 separated Y/CbCr | Y: 8 bits CbCr: 8 bits | Supported | Supported |
| | | YCbCr-422 separated Y/CbCr | Y: 8 bits CbCr: 8 bits | Supported | Supported |
| | | | Y: 10 bits CbCr: 8 bits | Supported | Supported |
| | | | Y: 10 bits CbCr: 10 bits | Supported | Supported |
| | | | Y: 12 bits CbCr: 8 bits | Supported* ⁵ | Supported |
| | | | Y: 12 bits CbCr: 12 bits | Supported* ⁵ | Supported* ⁵ |
| | | YCbCr-422 separated Only Y | 8 bits | Supported | Supported |
| | | | 10 bits | Supported | Supported |
| 12 bits | Supported | | Supported | | |
| RAW output | RAW12 | 12 bits | Supported* ⁵ | Supported* ⁵ | |

Notes: 1. When capturing progressive images or MIPI CSI-2 input data, be sure to enable the internal field signal generation function.

2. Dithering is performed for RGB-888→RGB-565 conversion.

3. RGB data is converted and transferred to memory from the video module after 8-bit precision conversion.

4. For details of output interfaces, see section 30.3.10, Output Data Format.

5. Y 12bit and CbCr 12bit and RAW12 are supported only when using the digital pin.

(1) Multiplexed YCbCr-422 Data Format in ITU-R BT.601 Interface

Data in the multiplexed YCbCr-422 format is a UYVY format (U0Y0V0Y1 format) YCbCr data. The Y, Cb, Cr image data is captured at the rising edges of the input video clock signal.

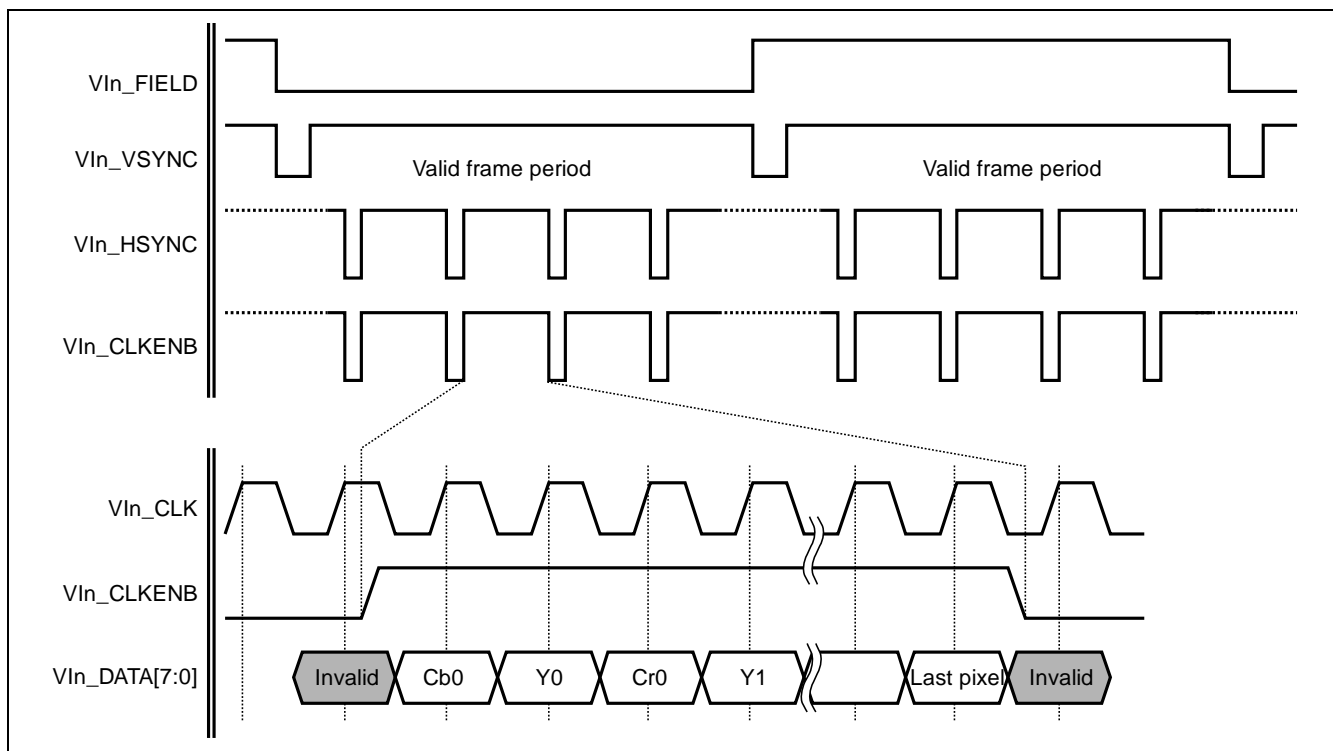


Figure 30.15 Multiplexed YCbCr-422 Data Format

(2) CbCr Multiplexed YCbCr-422 Data Format in ITU-R BT.601/1358 Interface

Data in the CbCr multiplexed YCbCr-422 format is YCbCr data with only the CbCr data being multiplexed. The Y and CbCr image data are captured at the rising edges of the input video clock signal.

Setting the YCAL bit in VnMC to 1 enables the capturing of image data with CbCr data in the upper bits and Y data in the lower bits.

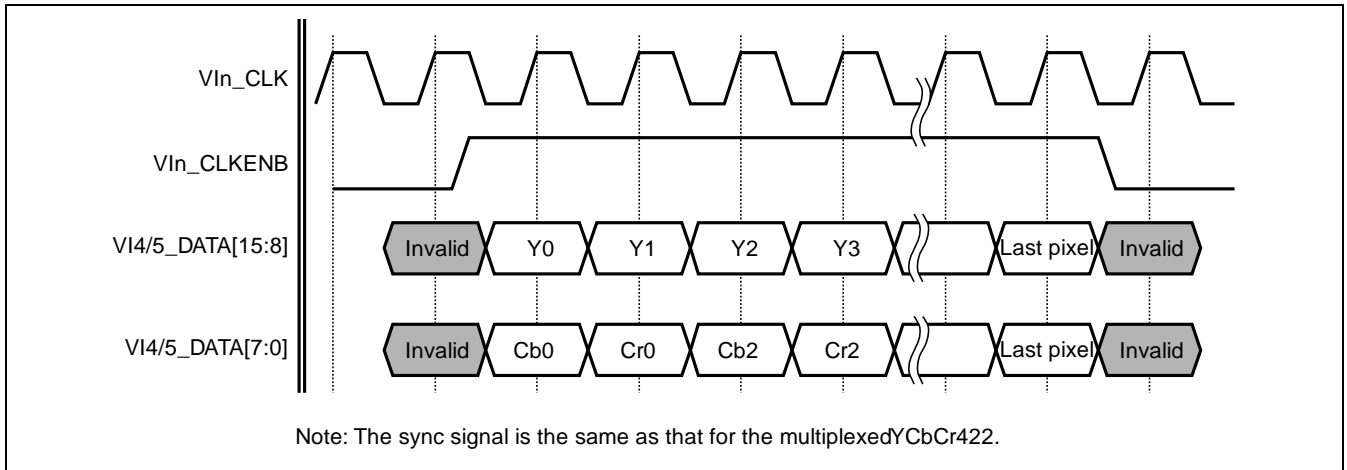


Figure 30.16 CbCr Multiplexed YCbCr-422 Data Format (Y Data in Upper Bits)

(3) RGB-666 Data Format in ITU-R BT.601 Interface

This data format can be used for the color conversion space in the RGB=4:4:4 format defined in the ITU-R BT.601 standard. The R, G, B image data are captured at the rising edges of the input video clock signal.

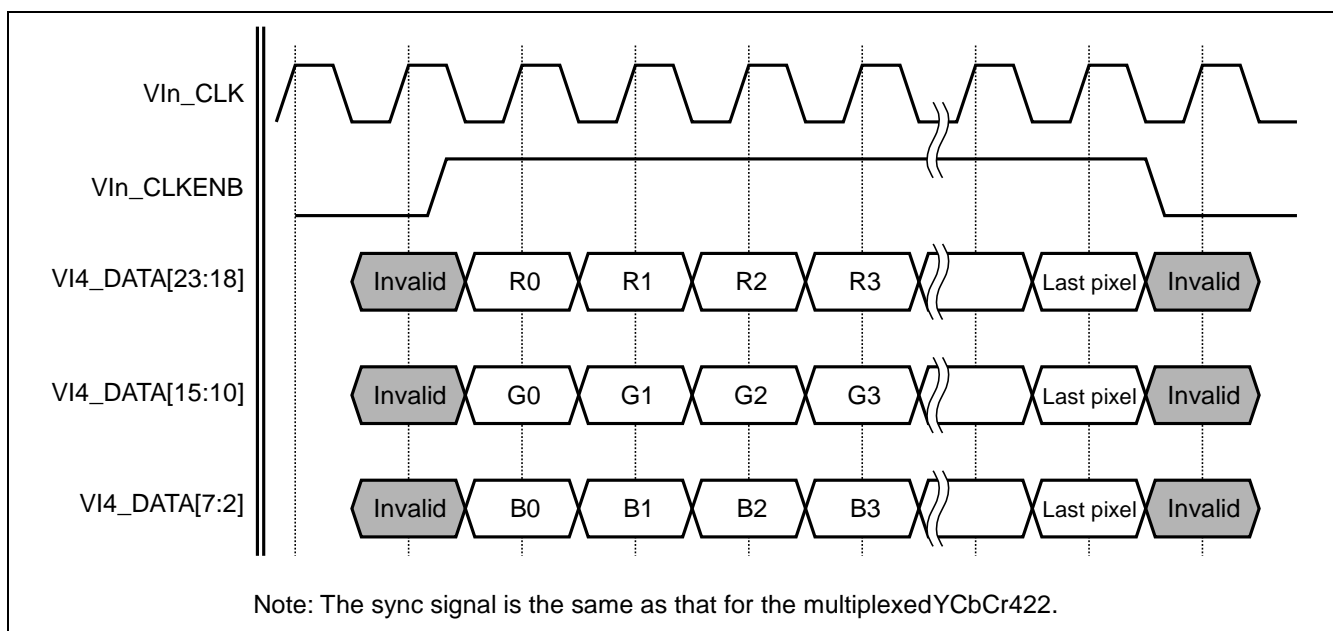


Figure 30.17 RGB-666 Data Format

(4) RGB-888 Data Format in ITU-R BT.601 Interface

This data format can be used for the color conversion space in the RGB=4:4:4 format defined in the ITU-R BT.601. The R, G, B image data are captured at the rising edges of the input video clock signal.

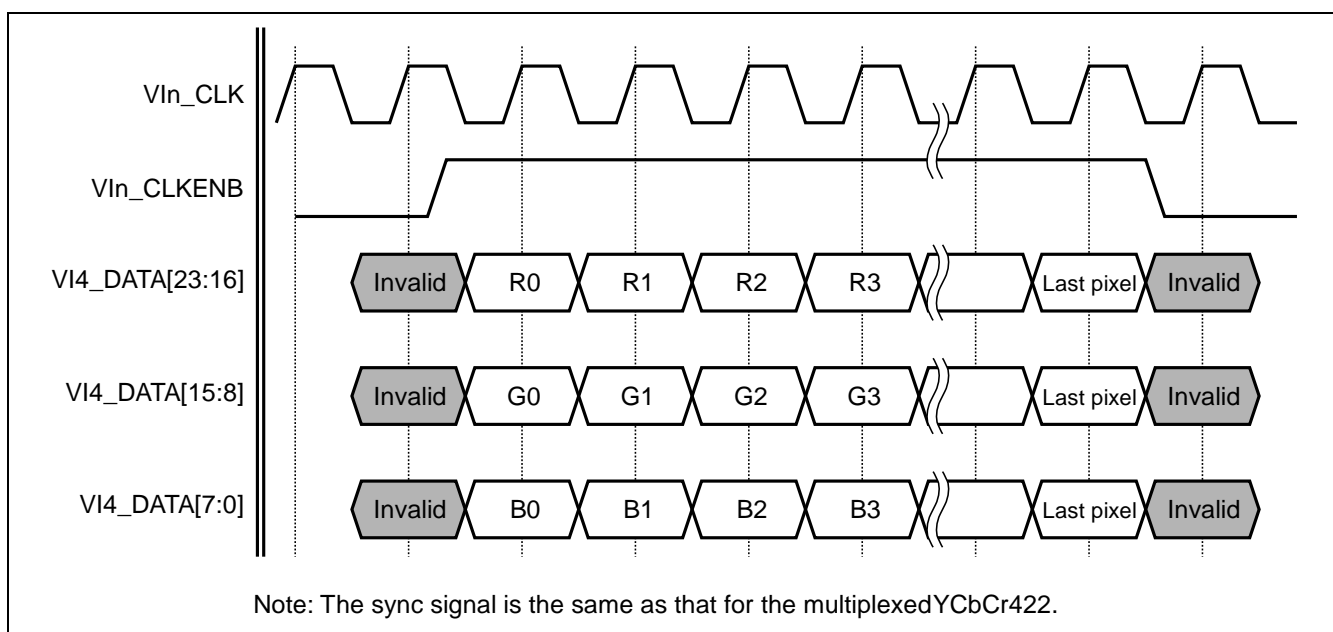


Figure 30.18 RGB-888 Data Format

(5) Multiplexed YCbCr-422 Data Format in ITU-R BT.656 Interface

In the ITU-R BT.656 interface, active data between the start of active video (SAV) and the end of active video (EAV) indicated with the timing reference code is captured.

Table 30.25 Video Timing Reference Code

| VINn Image Data (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N) | | | First Word | Second Word | Third Word | Fourth Word (XY) | |
|---|-------------|-------------|------------|-------------|------------|------------------|-----------------------------------|
| 12 bits | 10 bits | 8 bits | | | | | |
| VIn_DATA[11] | VIn_DATA[9] | VIn_DATA[7] | 1 | 0 | 0 | 1 | |
| VIn_DATA[10] | VIn_DATA[8] | VIn_DATA[6] | 1 | 0 | 0 | F | 0: Field 1 / 1: Field 2 |
| VIn_DATA[9] | VIn_DATA[7] | VIn_DATA[5] | 1 | 0 | 0 | V | 0: elsewhere 1: Field Blanking |
| VIn_DATA[8] | VIn_DATA[6] | VIn_DATA[4] | 1 | 0 | 0 | H | 0: SAV 1: EAV |
| VIn_DATA[7] | VIn_DATA[5] | VIn_DATA[3] | 1 | 0 | 0 | P3 | Protection bit 3 |
| VIn_DATA[6] | VIn_DATA[4] | VIn_DATA[2] | 1 | 0 | 0 | P2 | Protection bit 2 |
| VIn_DATA[5] | VIn_DATA[3] | VIn_DATA[1] | 1 | 0 | 0 | P1 | Protection bit 1 |
| VIn_DATA[4] | VIn_DATA[2] | VIn_DATA[0] | 1 | 0 | 0 | P0 | Protection bit 0 |
| VIn_DATA[3] | VIn_DATA[1] | — | 1 | 0 | 0 | 0 | |
| VIn_DATA[2] | VIn_DATA[0] | — | 1 | 0 | 0 | 0 | |
| VIn_DATA[1] | — | — | 1 | 0 | 0 | 0 | |
| VIn_DATA[0] | — | — | 1 | 0 | 0 | 0 | |

Active data is available with the multiplexed YCbCr-422 data format in the UYVY (U0Y0V0Y1) format. In this interface, timing reference and Y, Cb, Cr image data are captured at the rising edges of the input clock signal.

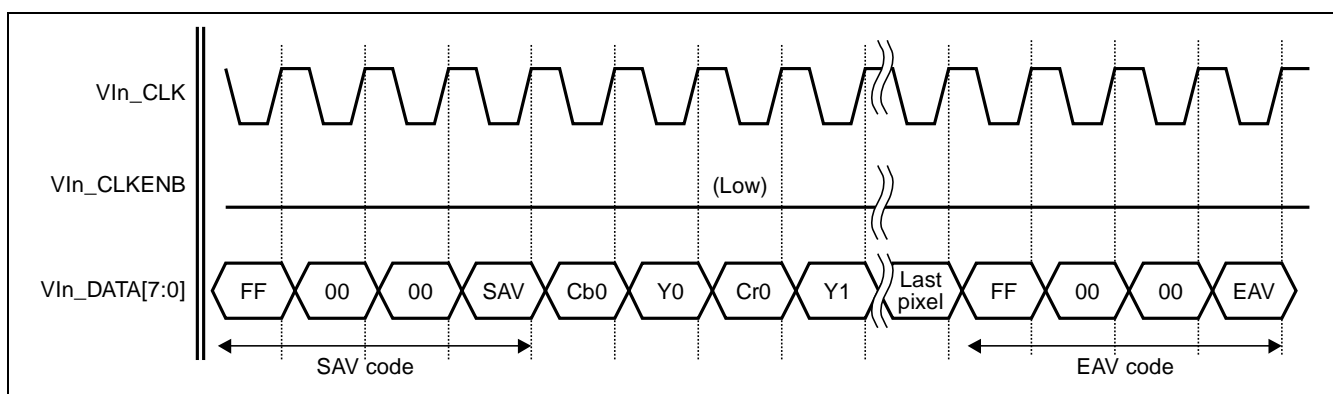


Figure 30.19 Multiplexed YCbCr-422 Data Format

Capturing is controlled based on field information. Therefore, the field signal and timing reference codes should be correct.

(6) RAW12 Data Format in ITU-R BT.601 Interface

The Pn image data is captured at the rising edges of the input video clock signal.

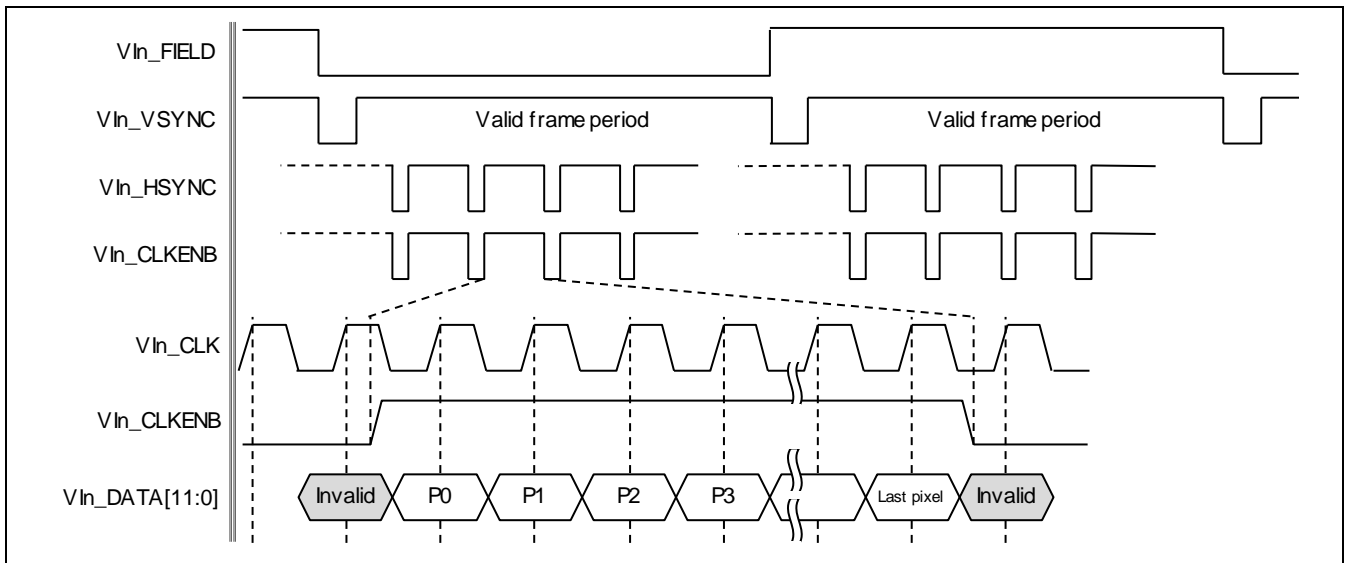


Figure 30.20 RAW12 Data Format

30.3.2 Error Correction

At the time of capturing with ITU-R BT.656, the VIN allows errors to be corrected with the timing reference code (SAV/EAV). The timing reference code (SAV/EAV) of ITU-R BT.656 has four protection bits, which can be used to correct only one-bit errors in the interface.

If the VIN cannot correct errors while the CEE bit in the interrupt enable register (VnIE) has been set to 1, an interrupt signal is generated as soon as the CES bit in the interrupt status (VnINTS) register is set. Note that no interrupt signal is generated if errors can be corrected.

30.3.3 Capture Mode

With the VIN, either of the single frame capture mode or continuous frame capture mode can be selected.

Specifying the capture field in IM bits of the main control (VnMC) register and then setting the SC bit in the frame capture (VnFC) register to 1 provides single frame capture mode. In this mode, the current frame is captured when the SC bit write timing (current scanline position) is smaller than the value of the start line pre-clip (VnSLPrC) register, or the next frame is captured in other cases. The capture data is transferred to the memory address that is set in the memory base 1 (VnMB1) register. For using single frame capture mode, necessary procedure of register setting is as followings.

1. Set 1 to the module enable (ME) bit in the main control register (VnMC).
2. Set 1 to the single frame capture mode (SC) bit in the frame capture register (VnFC).

Note: When using this setting, don't set 2'b01 to the interlace mode (IM) bits in the main control register (VnMC).

Specifying the capture field in IM bits of VnMC and then setting the CC bit of VnFC to 1 provides continuous frame capture mode, in which capture data is sequentially transferred to the addresses that are set in MB1 to MB3. In this case, the latest captured frame ID is shown in the FBS bits in the module status (VnMS) register. For using continuous frame capture mode, necessary procedure of register setting is as followings.

1. Set 1 to the module enable (ME) bit in the main control register (VnMC).
2. Set 1 to the continuous frame capture mode (CC) bit in the frame capture register (VnFC).

Note: Don't set 1 to the single frame capture mode (SC) bit and the continuous frame capture mode (CC) bit at the same time.

When the IM bits in VnMC are set to the full interlace mode, data in the capture start field is stored in every other line in the memory as the odd field (field 1) data and then data in the next field is stored as the even field (field 2) between the written lines so that the top and bottom field lines alternate with each other for interlace composition. The capture start field (top field) can be changed through the FOC bit in VnMC.

The following is a schematic diagram of capturing in full interlace mode when the odd field (field 1) is selected as the top field, the memory width is set to H'200, and VnMB1 is set to H'0000.

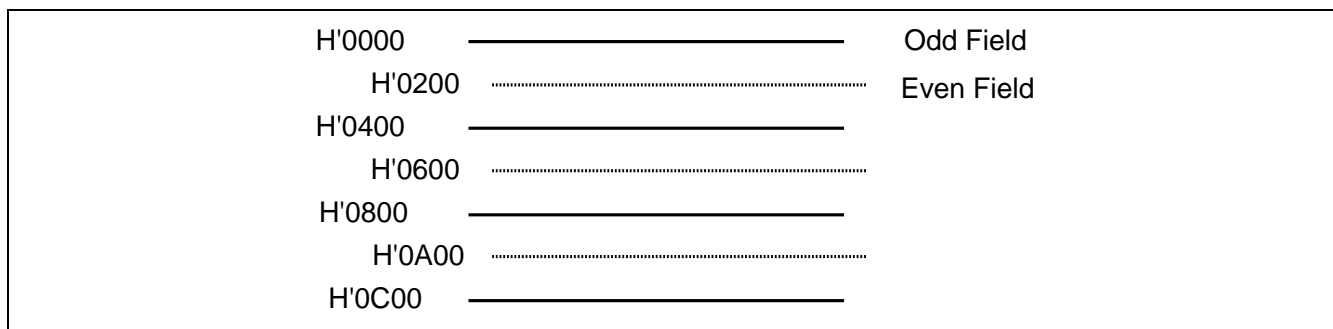


Figure 30.21 Example of Capturing Full Interlace

The VIN starts capture operation after detecting a frame signal switching in the ITU-R BT.601 or 656 interface. When capturing progressive data, in which the field signal does not change, use the internal field signal generation function to toggle the internal field signal. For details, see section 30.2.18.

30.3.4 Size Clipping

Image data that has been captured is pre-clipped according to the settings of the following registers: start line pre-clip (VnSLPrC), end line pre-clip (VnELPrC), start pixel pre-clip (VnSPPrC), and end pixel pre-clip (VnEPPrC).

For post-clipping after horizontal or vertical scaling, see section 30.2.21.5.

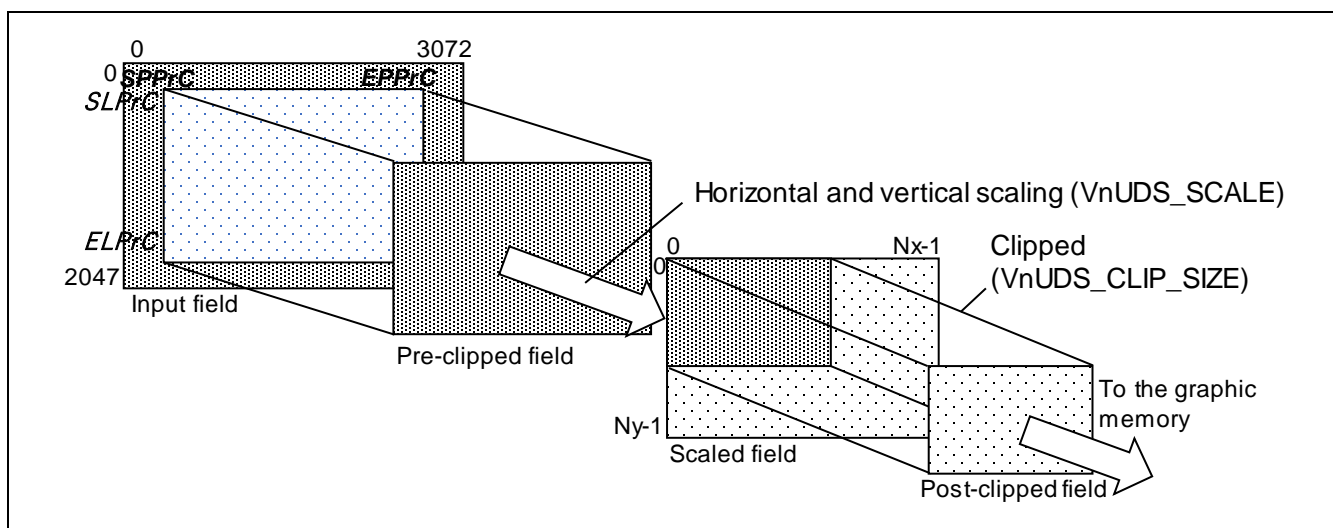


Figure 30.22 Example of Clipping

For all the post-clipped lines, the lengths of individual lines written into the memory are defined by the image stride (VnIS) register. The setting can be larger than the post-clipped frame width, but cannot be smaller than the width. The VnIS register must be filled with a value larger than the horizontal post-clipping width. The input field in the above figure shows an effective image area from the video decoder; the VIN does not allow anything exceeding the image area to be captured.

Note: Each of the following registers specifies a distance from the starting point in the effective image area: start line pre-clip (VnSLPrC), end line pre-clip (VnELPrC), start pixel pre-clip (VnSPPrC), and end pixel pre-clip (VnEPPrC). Specifically, in ITU-R BT.601, the distance is from the starting point of VIn_CLKENB (data enable); in ITU-R BT.656, the distance is from the SAV (start of active video) signal.

30.3.5 Vertical Scaling

For details on scaling in the vertical direction, see section 30.2.21.2.

30.3.6 Horizontal Scaling

For details on scaling in the horizontal direction, see section 30.2.21.2.

30.3.7 Color Conversion Function

(1) YC-RGB Color Conversion

When the data input format is YCbCr, set the BPS bit in the VnMC register to 0 to convert YCbCr data to RGB data. If an 8-bit data format is used for the capture interface, the color conversion is carried out according to the matrix coefficients set in the VnCSCC1, VnCSCC2, and VnCSCC3 registers. Otherwise, it is carried out according to the matrix coefficients set in the VnCSCE1, VnCSCE2, VnCSCE3, and VnCSCE4 registers. All of the input YCbCr data is extended to 12-bit data before carrying out color conversion.

Here, if the BPS bit in VnMC is set to 1, the data is stored in memory as it is in YCbCr format.

$$\begin{aligned}
 R &= \begin{pmatrix} VnCSCE1/ \\ YMUL[9:0] \\ or \\ VnCSCE1/ \\ YMUL2[13:0] \end{pmatrix} \times (Y - \begin{pmatrix} VnCSCE1/ \\ YSUB[7:0] \\ or \\ VnCSCE1/ \\ YSUB2[11:0] \end{pmatrix}) + \begin{pmatrix} VnCSCE3/ \\ RCRMUL[9:0] \\ or \\ VnCSCE3/ \\ RCRMUL2[13:0] \end{pmatrix} \times (Cr - \begin{pmatrix} VnCSCE1/ \\ CSUB[7:0] \\ or \\ VnCSCE1/ \\ CSUB2[11:0] \end{pmatrix}) \\
 G &= \begin{pmatrix} VnCSCE1/ \\ YMUL[9:0] \\ or \\ VnCSCE1/ \\ YMUL2[13:0] \end{pmatrix} \times (Y - \begin{pmatrix} VnCSCE1/ \\ YSUB[7:0] \\ or \\ VnCSCE1/ \\ YSUB2[11:0] \end{pmatrix}) - \begin{pmatrix} VnCSCE3/ \\ GCRMUL[9:0] \\ or \\ VnCSCE3/ \\ GCRMUL2[13:0] \end{pmatrix} \times (Cr - \begin{pmatrix} VnCSCE1/ \\ CSUB[7:0] \\ or \\ VnCSCE1/ \\ CSUB2[11:0] \end{pmatrix}) - \begin{pmatrix} VnCSCE4/ \\ GCBMUL[9:0] \\ or \\ VnCSCE4/ \\ GCBMUL2[13:0] \end{pmatrix} \times (Cb - \begin{pmatrix} VnCSCE1/ \\ CSUB[7:0] \\ or \\ VnCSCE1/ \\ CSUB2[11:0] \end{pmatrix}) \\
 B &= \begin{pmatrix} VnCSCE1/ \\ YMUL[9:0] \\ or \\ VnCSCE1/ \\ YMUL2[13:0] \end{pmatrix} \times (Y - \begin{pmatrix} VnCSCE1/ \\ YSUB[7:0] \\ or \\ VnCSCE1/ \\ YSUB2[11:0] \end{pmatrix}) + \begin{pmatrix} VnCSCE4/ \\ BCBMUL[9:0] \\ or \\ VnCSCE4/ \\ BCBMUL2[13:0] \end{pmatrix} \times (Cb - \begin{pmatrix} VnCSCE1/ \\ CSUB[7:0] \\ or \\ VnCSCE1/ \\ CSUB2[11:0] \end{pmatrix})
 \end{aligned}$$

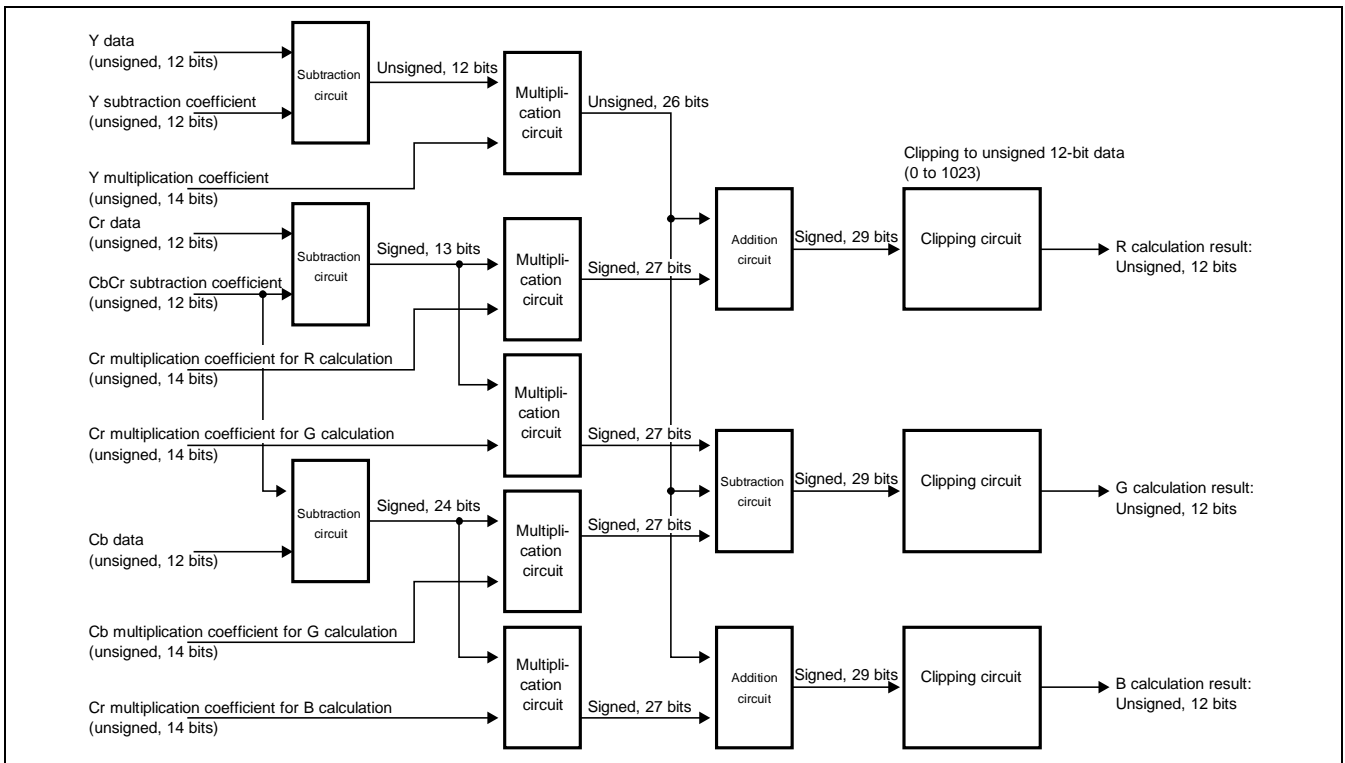


Figure 30.23 YCbCr→RGB Calculation Circuit Configuration

Examples of color space that can be set in YC-RGB conversion coefficient registers are shown below. See the register descriptions for details of the coefficients.

Table 30.26 Examples of YC-RGB Conversion Coefficient Register Settings

| YC-RGB Conversion Coefficient | YMUL | YSUB | CSUB | RCRMUL | GCRMUL | GCBMUL | BCBMUL |
|---|-------|------|------|--------|--------|--------|--------|
| ITU-R BT.601 (initial value) 16 ≤ Y ≤ 235, 16 ≤ Cb, Cr ≤ 240 | 1.164 | 16 | 128 | 1.596 | 0.813 | 0.392 | 2.017 |
| Luminance expansion example 1 ≤ Y ≤ 254, 16 ≤ Cb, Cr ≤ 240 | 1.008 | 1 | 128 | 1.596 | 0.392 | 0.813 | 2.017 |

Specify an integer in each addition coefficient bit field. For each multiplication coefficient, specify a value obtained by multiplying the desired coefficient value by 256.

Example: When the desired multiplication coefficient is 1.164

$$1.164 \times 256 = 297 \text{ (set value: B'0100101001)}$$

- Notes:
1. In order to process the data outside the range prescribed by the ITU-R BT.601 standard, set the CLP[1:0] bits in VnMC to 11. The data is clipped to the specified value before color space conversion.
 2. The YC-RGB color conversion data is unconditionally rounded to the $0 \leq R, G, B \leq 255$ range.

(2) RGB-YC Color Conversion

When the data input format is RGB, set the BPS bit in VnMC to 0 for color conversion of RGB data into YCbCr data format. Color conversion from RGB to YCbCr is done according to the matrix coefficients set in RGB-YC conversion coefficient registers (VnYCCR1-VnYCCR3/VnCBCCR1-VnCBCCR3/VnCRCCR1-VnCRCCR3). All of the input RGB data is extended to 12-bit data before color conversion.

If 1 is set to the BPS bit in VnMC, the data will be stored in the memory as it is in RGB format.

$$\begin{aligned}
 Y &= ((YCLRP \times R + YCLGP \times G + YCLGP \times B) \times 2^{YCLSFT}) + YCALP \\
 Cb &= ((CBCLRP \times R + CBCLGP \times G + CBCLGP \times B) \times 2^{CBCLSFT}) + CBCALP \\
 Cr &= ((CRCLRP \times R + CRCLGP \times G + CRCLGP \times B) \times 2^{CRCLSFT}) + CRCALP
 \end{aligned}$$

The following data rounding functions of RGB-YCbCr color conversion function can be independently set to each pixel. The circuit configuration of Y data is given below.

Table 30.27 Data Rounding Functions of RGB-YC Color Conversion Function

| Function | Symbol | Description |
|---|-------------|--|
| Sign extension enable | YEXPEN | Enables/disables signed bits of the matrix multiplication result. |
| Multiplication result shift down amount | YCLSFT[4:0] | Amount of shift down in the matrix multiplication result |
| Rounding off enable | YCLHEN | Enables/disables rounding off to the shift down amount. |
| Clipping enable | YCLCEN | Enables/disables data rounding process between 0 to 1023 of the output data. |

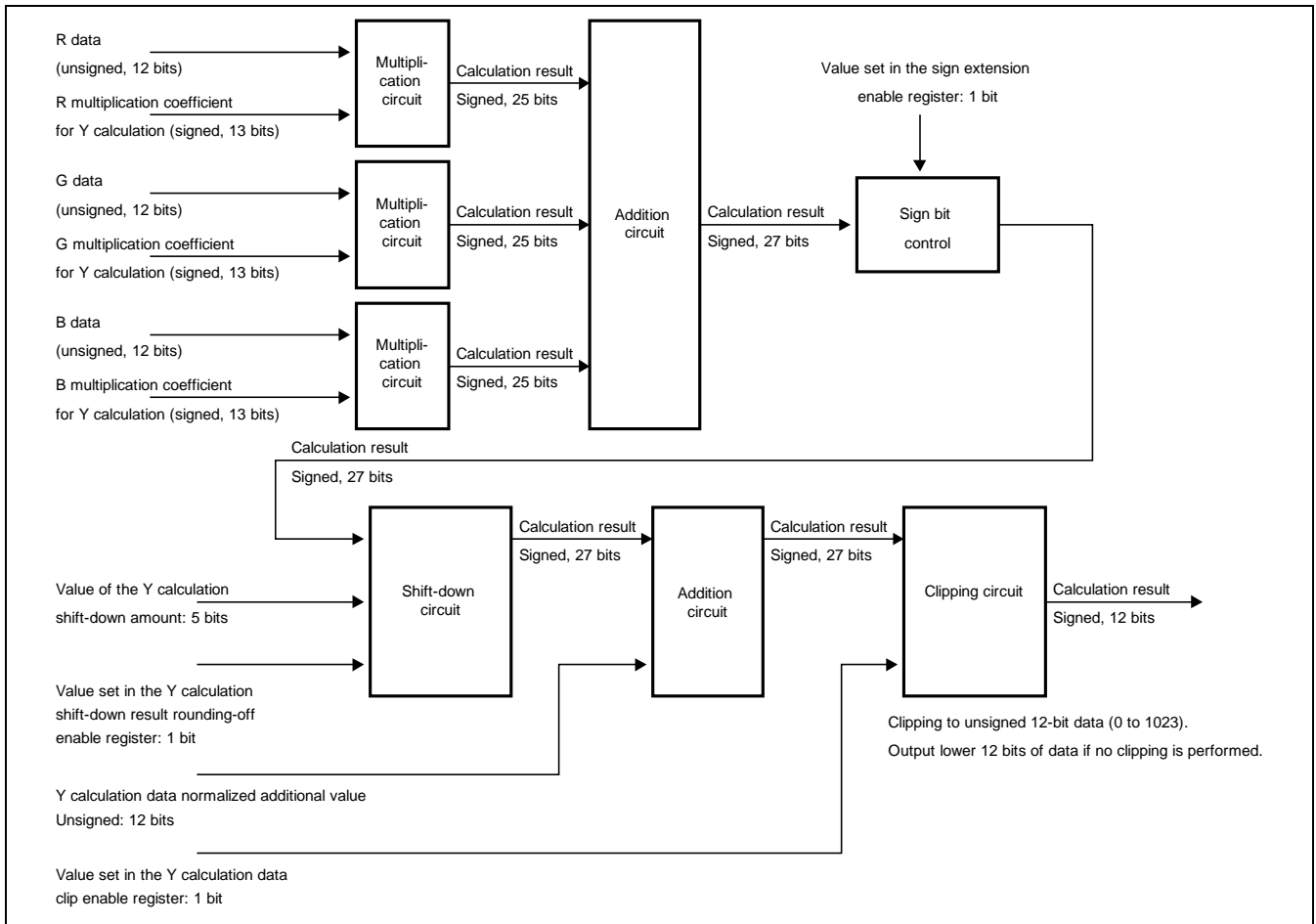


Figure 30.24 Y Data Circuit Configuration of RGB → YCbCr Color Conversion Function

30.3.8 Image Data Format Conversion Functions

(1) Lookup Table (LUT) Density Conversion Function

Set the LUTE bit in VnMC to 1 to enable table conversion of each pixel data of Y, Cb, Cr and R, G, B data after color conversion.

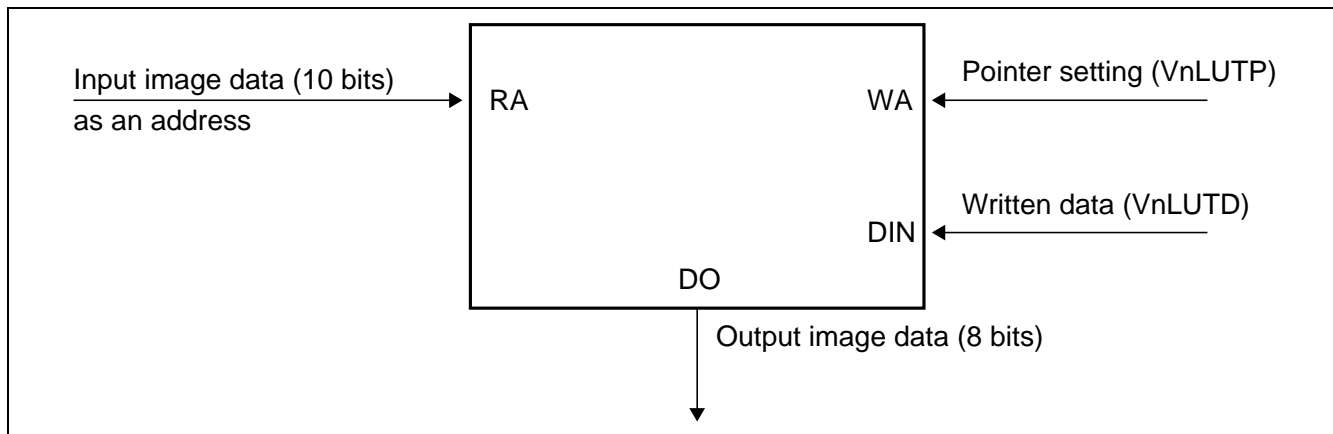


Figure 30.25 Lookup Table

Pointer Access:

The pointer is incremented every time a write access to the VnLUTD register is made.

The incremented data is read when the VnLUTP register is read. Since data is not incremented at read access, the pointer should be set just before reading.

The procedure given below must be followed to access the lookup table.

The lookup table cannot be accessed from the CPU during the conversion process.

(For write/read access to the LUT from the CPU, the LUTE bit in VnMC should be 0.)

Write access:

1. Write the access destination address in VnLUTP.
2. Write the data in VnLUTD.
Thereby the data will be reflected in the lookup table.
Also the pointer will automatically increase by 1.
3. Writing to VnLUTD will consecutively update the lookup table.

Read Access:

1. Write the access destination address in VnLUTP.
2. Read VnLUTD. This is dummy read; thus the read data needs to be discarded.
3. Read VnLUTD. This reading allows the address data written in VnLUTP to be acquired.

[Notes on Using the Lookup Table]

1. Set the lookup table only after capture operation has stopped.
2. Rewrite all before using the lookup table.
3. Data set to the lookup table should be within the range compatible to the input bit width.

(2) YCbCr444→YC Separation Function

When transferring YCbCr data to memory, Y data and CbCr data can be separated and transferred to different address spaces.

To perform YCbCr separation transfer, set the DTMD[1:0] bits in the VnDMR register to B'10. In this case, the Y data will be transferred to the address set in the memory base address register and CbCr data will be transferred to the address obtained by adding the value set in the VnUVAOF register to the memory base address register.

If the YMODE[2:0] bits in the VnDMR register are set to B'001 or B'011 or B'101, only Y data will be transferred to memory and CbCr data is not transferred to memory.

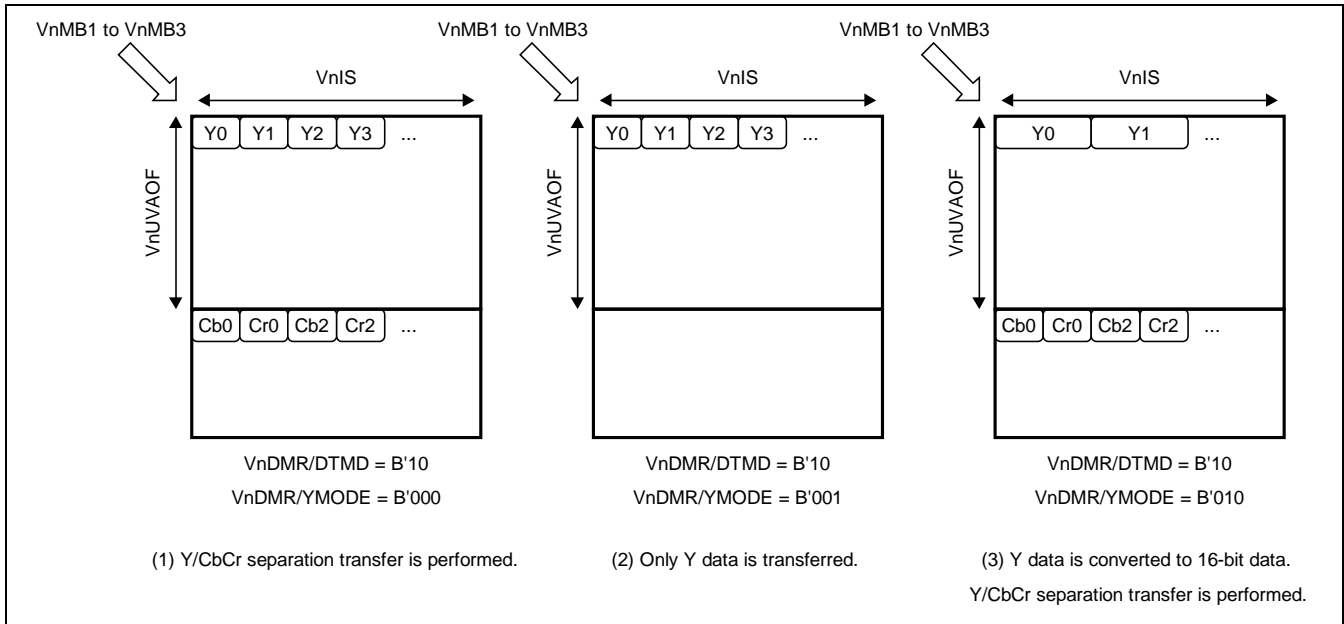


Figure 30.26 Y/Cb/Cr Separation in Big Endian

(3) Dithering Function

Dithering is performed when the internal RGB-888 format after color space conversion is converted to the RGB-565 or RGB-1555 format. The dithering mode can be selected with the DC[1:0] bits in VnMC.

(a) Dithering with Cumulative Addition

Set the DC[1:0] bits in VnMC to B'00 to perform dithering using the cumulative addition method in horizontal pixel units.

$$R_n[7:3] \leq (R_x[7:0] + R_{n-1}[2:0]) \gg 3$$

$$G_n[7:2] \leq (G_x[7:0] + G_{n-1}[1:0]) \gg 2$$

$$B_n[7:3] \leq (B_x[7:0] + B_{n-1}[2:0]) \gg 3$$

Where:

(R_n, G_n, B_n) = Output RGB-565 pixel

(R_x, G_x, B_x) = Input RGB-888 pixel

$(R_{n-1}, G_{n-1}, B_{n-1})$ = Pseudo random error (LSB of the cumulative error)

(b) Ordered Dithering

Set the DC[1:0] bits in VnMC to B'01 to perform dithering using the ordered dithering method.

$$R_n[7:3] \leq (R_x[7:0] + D42_{xy}) \gg 3$$

$$G_n[7:2] \leq (G_x[7:0] + D22_{xy}) \gg 2$$

$$B_n[7:3] \leq (B_x[7:0] + D42_{xy}) \gg 3$$

Where:

(R_n , G_n , B_n) = Output RGB-565 pixel

(R_x , G_x , B_x) = Input RGB-888 pixel

($D22_{xy}$, $D42_{xy}$) = Input x, y coordinate dithering matrix result

$$D22 = \begin{bmatrix} 0 & 3 \\ 2 & 1 \end{bmatrix}, D42 = \begin{bmatrix} 0 & 3 & 4 & 7 \\ 2 & 1 & 6 & 5 \end{bmatrix}$$

30.3.9 Internal Field Signal Generation

As the video input module controls capture of data in interlaced mode, correct capture control is not achieved if the external field signal level does not change.

Through the internal field signal generation function, the VIN can control the capture field signal even when the input field signal does not change, such as during progressive data capturing. The following settings can be made for the internal field generation function through the FTEV and FTEH bits in the data mode register 2 (VnDMR2).

- VSYNC field toggle mode (FTEV = 1 in VnDMR2)**
 When this setting is made, the VSYNC field toggle mode is entered for capture field signal control if the input field signal does not change for the VSYNC cycles specified by the VLV bits in VnDMR2. The toggle mode is canceled when a change in the external field signal level is detected (the capture operation is controlled according to the input field signal).
- HSYNC field toggle counter (FTEH = 1 in VnDMR2)**
 This counter counts the capture active lines. If the external field signal does not change until the count reaches the HLV setting in VnDMR2, the capture field signal is controlled.

Notes: 1. Do not set both the FTEV and FTEH bits in VnDMR2 at the same time.
 2. Immediately after cancellation of the toggle mode, capture control is skipped for one VSYNC cycle in some cases depending on the input field signal state.

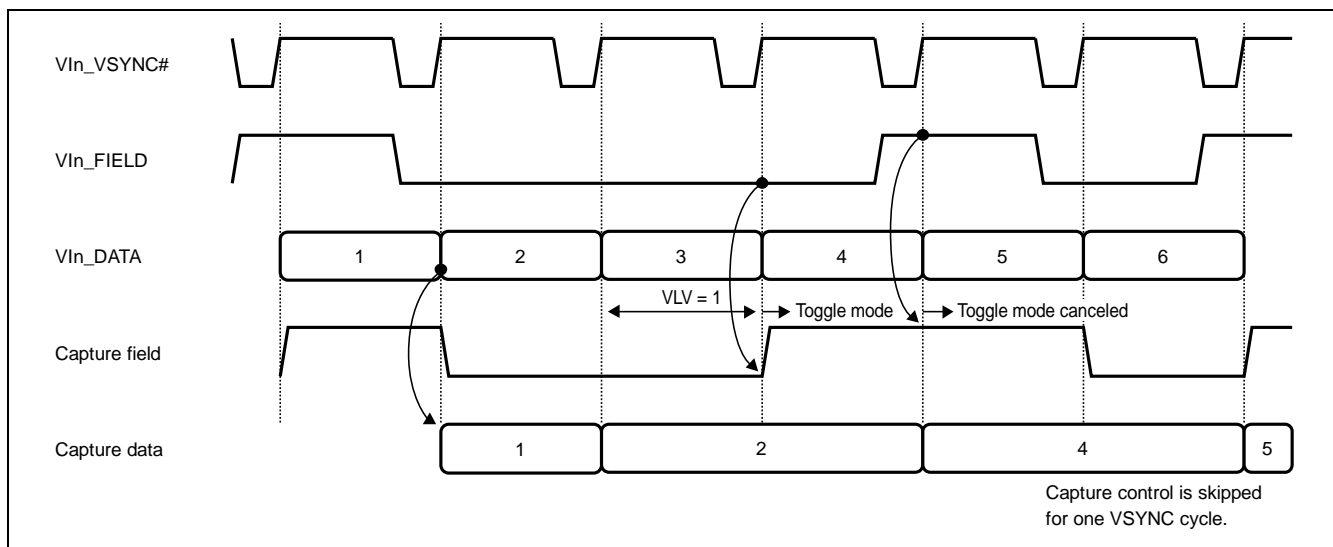


Figure 30.27 Overview and Notes of VSYNC Field Toggle Mode

30.3.10 Output Data Format

The VIN can output image data in the following formats. The figures in this section assume that data is stored in unified memory in little endian.

(1) YC: YCbCr-422, 8 bits

The 8-bit YUV image data in the YC (YCbCr) = 4:2:2 format is shown below. YC data can be switched between the UYVY format and YUYV format through the BPSM bit in VnDMR.

- BPSM = 0 in VnDMR: UYVY format

8-bit YCbCr-422 data (UYVY format)

| | | | | |
|--------------------|---------|----|----------|----|
| D63 to D48 | 63 | 56 | 55 | 48 |
| Image data 3 and 4 | Y3[7:0] | | Cr2[7:0] | |
| D47 to D32 | 47 | 40 | 39 | 32 |
| Image data 3 and 4 | Y2[7:0] | | Cb2[7:0] | |
| D31 to D16 | 31 | 24 | 23 | 16 |
| Image data 1 and 2 | Y1[7:0] | | Cr0[7:0] | |
| D15 to D0 | 15 | 8 | 7 | 0 |
| Image data 1 and 2 | Y0[7:0] | | Cb0[7:0] | |

- BPSM = 1 in VnDMR: YUYV format

8-bit YCbCr-422 data (YUYV format)

| | | | | |
|--------------------|----------|----|---------|----|
| D63 to D48 | 63 | 56 | 55 | 48 |
| Image data 3 and 4 | Cr2[7:0] | | Y3[7:0] | |
| D47 to D32 | 47 | 40 | 39 | 32 |
| Image data 3 and 4 | Cb2[7:0] | | Y2[7:0] | |
| D31 to D16 | 31 | 24 | 23 | 16 |
| Image data 1 and 2 | Cr0[7:0] | | Y1[7:0] | |
| D15 to D0 | 15 | 8 | 7 | 0 |
| Image data 1 and 2 | Cb0[7:0] | | Y0[7:0] | |

(2) YC: YCbCr-422, 10 bits

The 10-bit YUV image data in the YC (YCbCr) = 4:2:2 format is shown below.

Notes: 1. This format needs to set "1" to VnDMR/YC_THR.

2. A word address is output for YCbCr data according to the VnIS register setting.

10-bit YCbCr-422 data (UYVY format)

| | | | | | | | | | | | | |
|--------------|-----|---|---|---|---|-----|-----|----------|--|--|--|-----|
| D127 to D112 | 127 | | | | | 122 | 121 | | | | | 112 |
| Image data 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Y3[9:0] | | | | |
| D111 to D96 | 111 | | | | | 106 | 105 | | | | | 96 |
| Image data 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Cr2[9:0] | | | | |
| D95 to D80 | 95 | | | | | 90 | 89 | | | | | 80 |
| Image data 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Y2[9:0] | | | | |
| D79 to D64 | 79 | | | | | 74 | 73 | | | | | 64 |
| Image data 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Cb2[9:0] | | | | |
| D63 to D48 | 63 | | | | | 58 | 57 | | | | | 48 |
| Image data 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Y1[9:0] | | | | |
| D47 to D32 | 47 | | | | | 42 | 41 | | | | | 32 |
| Image data 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Cr0[9:0] | | | | |
| D31 to D16 | 31 | | | | | 30 | 25 | | | | | 16 |
| Image data 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Y0[9:0] | | | | |
| D15 to D0 | 15 | | | | | 10 | 9 | | | | | 0 |
| Image data 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Cb0[9:0] | | | | |

(3) YC: YCbCr-422, 12 bits

The 12-bit YUV image data in the YC (YCbCr) = 4:2:2 format is shown below.

Notes: 1. This format needs to set "1" to VnDMR/YC_THR.

2. A word address is output for YCbCr data according to the VnIS register setting.

12-bit YCbCr-422 data (UYVY format)

| | | | | | | |
|--------------|-----|---|-----|-----|-----------|-----|
| D127 to D112 | 127 | | 124 | 123 | | 112 |
| Image data 8 | 0 | 0 | 0 | 0 | Y3[11:0] | |
| D111 to D96 | 111 | | 108 | 107 | | 96 |
| Image data 7 | 0 | 0 | 0 | 0 | Cr2[11:0] | |
| D95 to D80 | 95 | | 92 | 91 | | 80 |
| Image data 6 | 0 | 0 | 0 | 0 | Y2[11:0] | |
| D79 to D64 | 79 | | 76 | 75 | | 64 |
| Image data 5 | 0 | 0 | 0 | 0 | Cb2[11:0] | |
| D63 to D48 | 63 | | 60 | 59 | | 48 |
| Image data 4 | 0 | 0 | 0 | 0 | Y1[11:0] | |
| D47 to D32 | 47 | | 44 | 43 | | 32 |
| Image data 3 | 0 | 0 | 0 | 0 | Cr0[11:0] | |
| D31 to D16 | 31 | | 28 | 27 | | 16 |
| Image data 2 | 0 | 0 | 0 | 0 | Y0[11:0] | |
| D15 to D0 | 15 | | 12 | 11 | | 0 |
| Image data 1 | 0 | 0 | 0 | 0 | Cb0[11:0] | |

(4) YC: YC Separation YCbCr-422, 8 bits

This is 8-bit YUV image data in the YC separated YC (YCbCr) = 4:2:2 format. UV data is supported by the NV16 format only.

If the YMODE[2:0] bits in the data mode register (VnDMR) are set to B'000 or B'001, setting the DTMD[1:0] bits in VnDMR to B'10 changes the format to 4:2:2 and transfers the UV data to an address which is specified by the addition of the value set in the UV address offset register (VnUVAOF) to the Memory Base. If the YMODE[2:0] bits in VnDMR are set to B'001, only Y data can be transferred and UV data cannot be transferred.

| | | | | |
|--------------------|----------|----|----------|----|
| Y data | | | | |
| D31 to D16 | 31 | 24 | 23 | 16 |
| Image data 3 and 4 | Y3[7:0] | | Y2[7:0] | |
| D15 to D0 | 15 | 8 | 7 | 0 |
| Image data 1 and 2 | Y1[7:0] | | Y0[7:0] | |
| Cb, Cr data | | | | |
| D31 to D16 | 31 | 24 | 23 | 16 |
| Image data 3 and 4 | Cr2[7:0] | | Cb2[7:0] | |
| D15 to D0 | 15 | 8 | 7 | 0 |
| Image data 1 and 2 | Cr0[7:0] | | Cb0[7:0] | |

(5) YC: YC Separation YCbCr-420, 8 bits

This is 8-bit YUV image data in the YC separated YC (YCbCr) = 4:2:0 format. UV data is supported by the NV12 format only.

If the YMODE[2:0] bits in the data mode register (VnDMR) are set to B'000, setting the DTMD[1:0] bits in VnDMR to B'11 changes the format to 4:2:0 and transfers the UV data to an address which is specified by the addition of the value set in the UV address offset register (VnUVAOF) to the Memory Base.

| | | | | |
|--------------------|-----------|----|-----------|----|
| Y data | | | | |
| D31 to D16 | 31 | 24 | 23 | 16 |
| Image data 3 and 4 | Y03[7:0] | | Y02[7:0] | |
| D15 to D0 | 15 | 8 | 7 | 0 |
| Image data 1 and 2 | Y01[7:0] | | Y00[7:0] | |
| Cb, Cr data | | | | |
| D31 to D16 | 31 | 24 | 23 | 16 |
| Image data 3 and 4 | Cr02[7:0] | | Cb02[7:0] | |
| D15 to D0 | 15 | 8 | 7 | 0 |
| Image data 1 and 2 | Cr00[7:0] | | Cb00[7:0] | |

(6) YC: YC Separation YCbCr-422, Y(10 bits) / C(10 bits)

This is 10-bit YUV image data in the YC separated YC (YCbCr) = 4:2:2 format. UV data is supported by the NV12 format only.

If the YMODE[2:0] bits in the data mode register (VnDMR) are set to B'000 or B'001, setting the DTMD[1:0] bits in VnDMR to B'10 changes the format to 4:2:2 and transfers the UV data to an address which is specified by the addition of the value set in the UV address offset register (VnUVAOF) to the Memory Base. If the YMODE[2:0] bits in VnDMR are set to B'001, only Y data can be transferred and UV data cannot be transferred.

Notes: 1. This format needs to set "1" to VnDMR/YC_THR.

2. A word address is output for YCbCr data according to the VnIS register setting.

Y data (10 bits)

| | | | | | | | |
|--------------|----|----|----|----|---|---|---------|
| D63 to D48 | 63 | 58 | 57 | 48 | | | |
| Image data 4 | 0 | 0 | 0 | 0 | 0 | 0 | Y3[9:0] |
| D47 to D32 | 47 | 42 | 41 | 32 | | | |
| Image data 3 | 0 | 0 | 0 | 0 | 0 | 0 | Y2[9:0] |
| D31 to D16 | 31 | 30 | 25 | 16 | | | |
| Image data 2 | 0 | 0 | 0 | 0 | 0 | 0 | Y1[9:0] |
| D15 to D0 | 15 | 10 | 9 | 0 | | | |
| Image data 1 | 0 | 0 | 0 | 0 | 0 | 0 | Y0[9:0] |

Cb, Cr data (10 bits)

| | | | | | | | |
|--------------|----|----|----|----|---|---|----------|
| D63 to D48 | 63 | 58 | 57 | 48 | | | |
| Image data 4 | 0 | 0 | 0 | 0 | 0 | 0 | Cr2[9:0] |
| D47 to D32 | 47 | 42 | 41 | 32 | | | |
| Image data 3 | 0 | 0 | 0 | 0 | 0 | 0 | Cb2[9:0] |
| D31 to D16 | 31 | 30 | 25 | 16 | | | |
| Image data 2 | 0 | 0 | 0 | 0 | 0 | 0 | Cr0[9:0] |
| D15 to D0 | 15 | 10 | 9 | 0 | | | |
| Image data 1 | 0 | 0 | 0 | 0 | 0 | 0 | Cb0[9:0] |

(7) YC: YC Separation YCbCr-422, Y(12 bits) / C(12 bits)

This is 12-bit YUV image data in the YC separated YC (YCbCr) = 4:2:2 format. UV data is supported by the NV12 format only.

If the YMODE[2:0] bits in the data mode register (VnDMR) are set to B'000 or B'001, setting the DTMD[1:0] bits in VnDMR to B'10 changes the format to 4:2:2 and transfers the UV data to an address which is specified by the addition of the value set in the UV address offset register (VnUVAOF) to the Memory Base. If the YMODE[2:0] bits in VnDMR are set to B'001, only Y data can be transferred and UV data cannot be transferred.

- Notes: 1. This format needs to set "1" to VnDMR/YC_THR.
 2. A word address is output for YCbCr data according to the VnIS register setting.

Y data (12 bits)

| | | | | |
|--------------|----|----|----|----------|
| D63 to D48 | 63 | 60 | 59 | 48 |
| Image data 4 | 0 | 0 | 0 | Y3[11:0] |
| D47 to D32 | 47 | 44 | 43 | 32 |
| Image data 3 | 0 | 0 | 0 | Y2[11:0] |
| D31 to D16 | 31 | 28 | 27 | 16 |
| Image data 2 | 0 | 0 | 0 | Y1[11:0] |
| D15 to D0 | 15 | 12 | 11 | 0 |
| Image data 1 | 0 | 0 | 0 | Y0[11:0] |

Cb, Cr data (12 bits)

| | | | | |
|--------------|----|----|----|-----------|
| D63 to D48 | 63 | 60 | 59 | 48 |
| Image data 4 | 0 | 0 | 0 | Cr2[11:0] |
| D47 to D32 | 47 | 44 | 43 | 32 |
| Image data 3 | 0 | 0 | 0 | Cb2[11:0] |
| D31 to D16 | 31 | 28 | 27 | 16 |
| Image data 2 | 0 | 0 | 0 | Cr0[11:0] |
| D15 to D0 | 15 | 12 | 11 | 0 |
| Image data 1 | 0 | 0 | 0 | Cb0[11:0] |

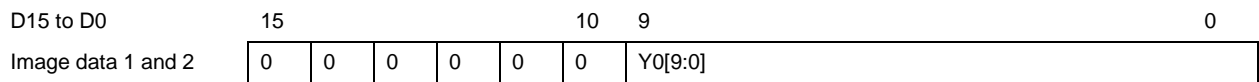
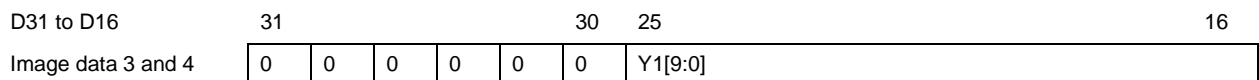
(8) YC: YC Separation YCbCr-422, 10 bits

This is 10-bit YUV image data in the YC separated YC (YCbCr) = 4:2:2 format. UV data is supported by the NV16 format only.

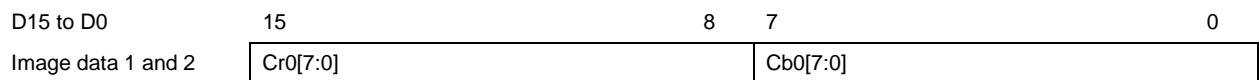
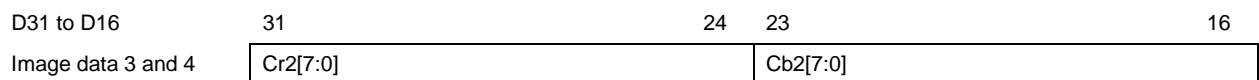
If the YMODE[2:0] bits in the data mode register (VnDMR) are set to B'010 or B'011, setting the DTMD[1:0] bits in VnDMR to B'10 converts 10-bit Y data to 16-bit and changes the format to 4:2:2. UV data is transferred to the address obtained by adding the value set in the UV address offset register (VnUVAOF) to Memory Base. If the YMODE[2:0] bits in VnDMR are set to B'011, only Y data can be transferred and UV data cannot be transferred.

- Notes: 1. A word address is output for Y data according to the VnIS register setting.
2. A byte address is output for CbCr data according to the VnIS register setting.

Y data



Cb, Cr data



(9) YC: YC Separation YCbCr-422, 12 bits

This is 12-bit YUV image data in the YC separated YC (YCbCr) = 4:2:2 format. UV data is supported by the NV16 format only.

If the YMODE[2:0] bits in the data mode register (VnDMR) are set to B'100 or B'101, setting the DTMD[1:0] bits in VnDMR to B'10 converts 12-bit Y data to 16-bit and changes the format to 4:2:2. UV data is transferred to the address obtained by adding the value set in the UV address offset register (VnUVAOF) to Memory Base. If the YMODE[2:0] bits in VnDMR are set to B'101, only Y data can be transferred and UV data cannot be transferred.

- Notes: 1. A word address is output for Y data according to the VnIS register setting.
2. A byte address is output for CbCr data according to the VnIS register setting.

Y data

| | | | | |
|--------------------|----|----|----|----------|
| D31 to D16 | 31 | 28 | 27 | 16 |
| Image data 3 and 4 | 0 | 0 | 0 | Y1[11:0] |

| | | | | |
|--------------------|----|----|----|----------|
| D15 to D0 | 15 | 12 | 11 | 0 |
| Image data 1 and 2 | 0 | 0 | 0 | Y0[11:0] |

Cb, Cr data

| | | | | |
|--------------------|----------|----|----------|----|
| D31 to D16 | 31 | 24 | 23 | 16 |
| Image data 3 and 4 | Cr2[7:0] | | Cb2[7:0] | |

| | | | | |
|--------------------|----------|---|----------|---|
| D15 to D0 | 15 | 8 | 7 | 0 |
| Image data 1 and 2 | Cr0[7:0] | | Cb0[7:0] | |

(10) 16 Bits/Pixel: RGB-565

The RGB levels are expressed through 5 bits for R, 6 bits for G, and 5 bits for B.

16 bits/pixel data (RGB data) format

| | | | | | | |
|------------|--------|----|--------|---|--------|---|
| D15 to D0 | 15 | 11 | 10 | 5 | 4 | 0 |
| Image data | R[4:0] | | G[5:0] | | B[4:0] | |

(11) 16 Bits/Pixel: ARGB-1555

The ARGB levels are expressed through 1 bit for A, 5 bits for R, 5 bits for G, and 5 bits for B. For data conversion to ARGB-1555, the lowest bit of the G data in RGB-565 data is truncated, and the A value specified through the register is added.

Set the DTMD[1:0] bits in the data mode register (VnDMR) to B'01 to specify conversion to ARGB-1555, and specify the A value in the ABIT bit in VnDMR.

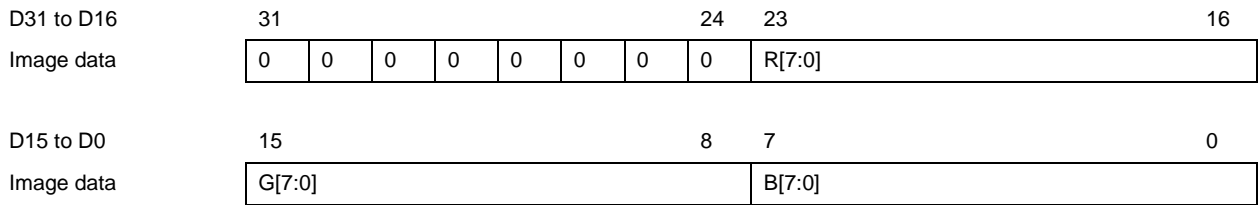
16 bits/pixel data (ARGB data) format

| | | | | | | | |
|------------|----|--------|----|--------|---|--------|---|
| D15 to D0 | 15 | 14 | 10 | 9 | 5 | 4 | 0 |
| Image data | A | R[4:0] | | G[4:0] | | B[4:0] | |

(12) 32 Bits/Pixel: RGB-888

The RGB levels are expressed through 8 bits for R, 8 bits for G, and 8 bits for B. Bits 31 to 24 are fixed to 0.

32 bits/pixel data (RGB data) format



(13) 32 Bits/Pixel: ARGB-8888

The ARGB levels are expressed through 8 bits for A, 8 bits for R, 8 bits for G, and 8 bits for B. The A value specified by the A8BIT[7:0] bits in VnDMR is set in bits 31 to 24.

32 bits/pixel data (ARGB data) format



(14) RAW: 12 bits

The 12-bit RAW image data format is shown below.

- Notes: 1. This format needs to set "1" to VnDMR/YC_THR.
 2. A word address is output for RAW data according to the VnIS register setting.

12-bit RAW data format

| | | | | | | | | | |
|--------------|-----|---|---|-----|----------|--|--|--|-----|
| D127 to D112 | 127 | | | 124 | 123 | | | | 112 |
| Image data 8 | 0 | 0 | 0 | 0 | P7[11:0] | | | | |
| D111 to D96 | 111 | | | 108 | 107 | | | | 96 |
| Image data 7 | 0 | 0 | 0 | 0 | P6[11:0] | | | | |
| D95 to D80 | 95 | | | 92 | 91 | | | | 80 |
| Image data 6 | 0 | 0 | 0 | 0 | P5[11:0] | | | | |
| D79 to D64 | 79 | | | 76 | 75 | | | | 64 |
| Image data 5 | 0 | 0 | 0 | 0 | P4[11:0] | | | | |
| D63 to D48 | 63 | | | 60 | 59 | | | | 48 |
| Image data 4 | 0 | 0 | 0 | 0 | P3[11:0] | | | | |
| D47 to D32 | 47 | | | 44 | 43 | | | | 32 |
| Image data 3 | 0 | 0 | 0 | 0 | P2[11:0] | | | | |
| D31 to D16 | 31 | | | 28 | 27 | | | | 16 |
| Image data 2 | 0 | 0 | 0 | 0 | P1[11:0] | | | | |
| D15 to D0 | 15 | | | 12 | 11 | | | | 0 |
| Image data 1 | 0 | 0 | 0 | 0 | P0[11:0] | | | | |

(15) RAW: 8 bits and 8-bit Embedded Data

The 8-bit RAW image data and 8-bit embedded data format are shown below.

Notes: 1. These data are transferred by 4-Byte unit.

2. A byte address is output for RAW data according to the VnIS register setting.

8-bit RAW data format

| | | | | |
|--------------------|---------|----|---------|----|
| D63 to D48 | 63 | 56 | 55 | 48 |
| Image data 7 and 8 | P7[7:0] | | P6[7:0] | |
| D47 to D32 | 47 | 40 | 39 | 32 |
| Image data 5 and 6 | P5[7:0] | | P4[7:0] | |
| D31 to D16 | 31 | 24 | 23 | 16 |
| Image data 3 and 4 | P3[7:0] | | P2[7:0] | |
| D15 to D0 | 15 | 8 | 7 | 0 |
| Image data 1 and 2 | P1[7:0] | | P0[7:0] | |

30.3.11 Endian Conversion

The VIN stores captured data in memory in little endian with the initial settings. Set the EN bit in the main control register (VnMC) to 1 to convert data into big endian before storing in memory.

Endian conversion in word units is controlled through the EN bit in VnMC, and swapping in byte units is controlled through the BPSM bit in the data mode register (VnDMR). For conversion to big endian, specify the BPSM bit as shown in the following table according to the data format specified through the DTMD bits in VnDMR and BPS bit in VnMC.

Table 30.28 Endian Conversion Unit

| Data Format | BPS in VnMC | DTMD[1:0] in VnDMR | EXRGB in VnDMR | BPSM in VnDMR | Endian Conversion Unit |
|-------------|-------------|--------------------|----------------|---------------|------------------------|
| RGB-565 | 0 | 00 | 0 | 0 | Word units |
| RGB-888 | 0 | 00 | 1 | 0 | Longword units |
| YCbCr-422 | 1 | 00 | 0 | 1 | Byte units |
| ARGB-1555 | 0 | 01 | 0 | 0 | Word units |
| ARGB-8888 | 0 | 01 | 1 | 0 | Longword units |
| YC | 1 | 10 | 0 | 0 | Byte units |

The following figures show endian conversions in 1 byte, 2 byte, and 4 byte units.

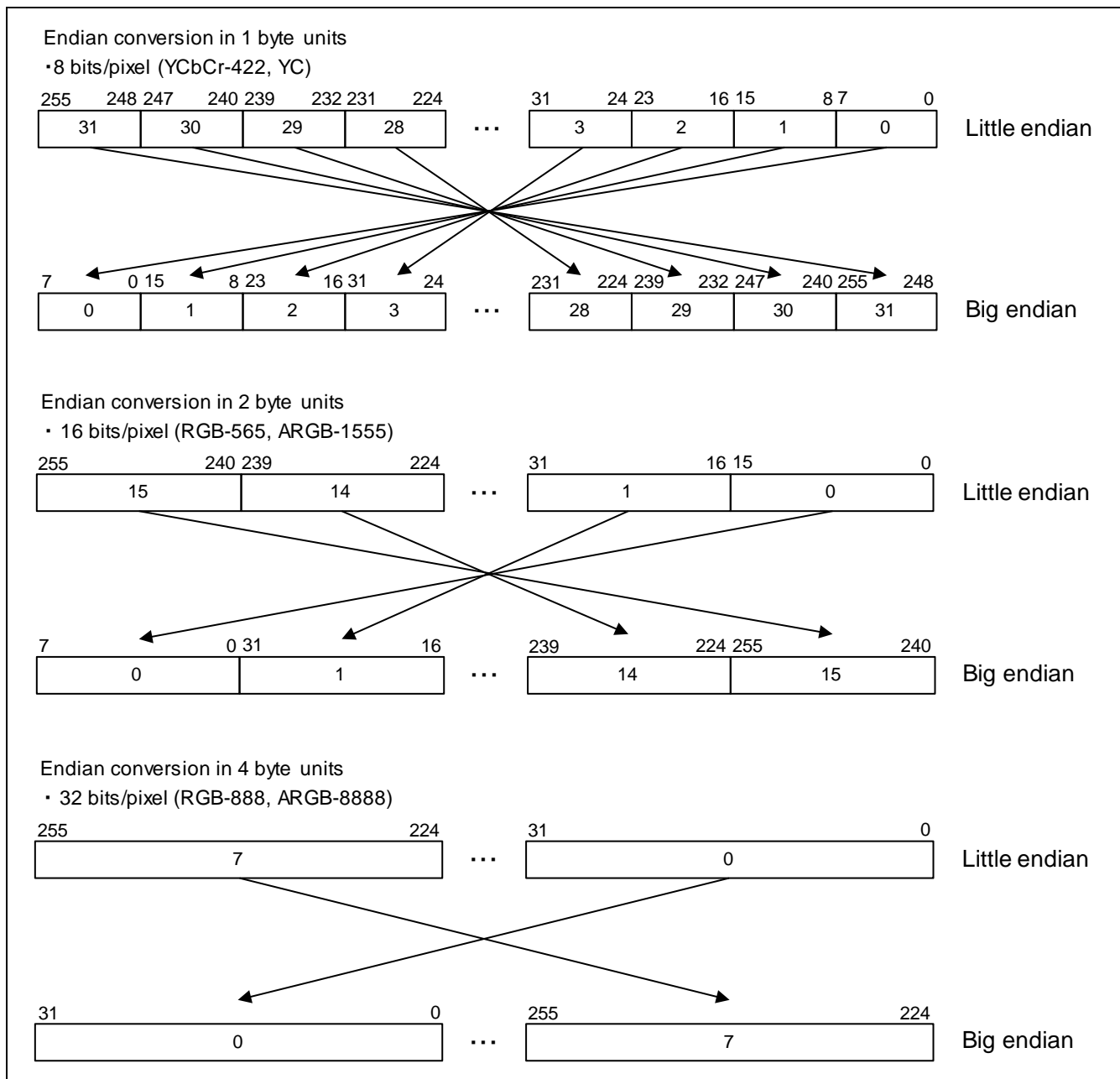


Figure 30.28 Data Alignment Conversion from Little Endian to Big Endian

30.3.12 Module Standby Mode

The LSI supports module standby mode in which clock supply to the VIN is stopped. The video input module should not be accessed during module standby mode.

30.3.13 Transition to Module Standby Mode

1. Clear the module enable (ME) bit in the main control register (VnMC) and the continuous frame capture (CC) bit and the single frame capture (SC) bit in the frame capture register (VnFC) to 0 to stop the video input module.
2. Confirm that the capture active (CA) bit in the module status register (VnMS) is cleared to 0*.
3. Stop the clock supply.

Note: * The camera device should be kept operation until the CA bit in VnMS set to 0.

30.3.14 Cancellation of Module Standby Mode and Restarting of Video Input Module

1. Start the clock supply.
2. Set the module enable (ME) bit in the main control register (VnMC) to 1 to start the video input module.
3. Set the continuous frame capture (CC) bit or the single frame capture (SC) bit in the frame capture register (VnFC) to 1.

30.3.15 Interrupts

The Video Input Module is capable of generating 24 interrupt(RZ/G2H, RZ/G2N).

The Video Input Module is capable of generating 8 interrupt(RZ/G2M V1.3, RZ/G2M V3.0).

The Video Input Module is capable of generating 6 interrupt(RZ/G2E).

For details, refer to section 19.

Table 30.29 Interrupt assignment

| | | Second Generation RZ/G Series Products | | | | |
|-----------------------|---|---|------------------------------------|---------------|---------------|--|
| Interrupt Name | Cause of interrupt | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| VIN.ch0 | FIS2 or VFS or VRS or FIS or CES or SIS or EFS or FOS bit in V0INTS | √ | √ | √ | — | |
| VIN.ch1 | FIS2 or VFS or VRS or FIS or CES or SIS or EFS or FOS bit in V1INTS | √ | √ | √ | — | |
| VIN.ch2 | FIS2 or VFS or VRS or FIS or CES or SIS or EFS or FOS bit in V2INTS | √ | √ | √ | — | |
| VIN.ch3 | FIS2 or VFS or VRS or FIS or CES or SIS or EFS or FOS bit in V3INTS | √ | √ | √ | — | |
| VIN.ch4 | FIS2 or VFS or VRS or FIS or CES or SIS or EFS or FOS bit in V4INTS | √ | √ | √ | √ | |
| VIN.ch5 | FIS2 or VFS or VRS or FIS or CES or SIS or EFS or FOS bit in V5INTS | √ | √ | √ | √ | |
| VIN.ch6 | FIS2 or VFS or VRS or FIS or CES or SIS or EFS or FOS bit in V6INTS | √ | √ | √ | — | |
| VIN.ch7 | FIS2 or VFS or VRS or FIS or CES or SIS or EFS or FOS bit in V7INTS | √ | √ | √ | — | |

30.3.16 Input and Output of YCbCr-422 Format

The example of input and output of YCbCr-422 format is as followings.

The example is case which the image is input from digital pins.

(1) 10 bits (Valid data is 8bit)

(a) Input: UYVY format and Register Settings

10-bit YCbCr-422 data (UYVY format)

VIn_DATA15 to

| VIn_DATA0 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|----|----|----|----|----|----|----------|---|---|---|---|---|---|---|---|---|
| Y1 | * | * | * | * | * | * | Y1[7:0] | | | | | | * | * | | |
| Cr0 | * | * | * | * | * | * | Cr0[7:0] | | | | | | * | * | | |
| Y0 | * | * | * | * | * | * | Y0[7:0] | | | | | | * | * | | |
| Cb0 | * | * | * | * | * | * | Cb0[7:0] | | | | | | * | * | | |

Note: * Fix the pins at a high or low level.

| Interface | VnMC/EXINF | VnMC/INF | VnMC/BPS |
|----------------------|------------|----------|----------|
| Multiplexed YCbCr422 | 10 bits | 00 | 011 |
| | | | 1 |

(b) Output: UYVY, 8 bits, format and Register Settings

8-bit YCbCr-422 data (UYVY format)

| | | | | |
|--------------------|---------|----|----------|----|
| D31 to D16 | 31 | 24 | 23 | 16 |
| Image data 1 and 2 | Y1[7:0] | | Cr0[7:0] | |
| D15 to D0 | 15 | 8 | 7 | 0 |
| Image data 1 and 2 | Y0[7:0] | | Cb0[7:0] | |

| Format of Data Stored in Memory | VnDMR/ YMODE[2:0] | VnDMR/ EXRGB | VnDMR/ YC_THR | VnDMR/ BPSM | VnDMR/ DTMD[1:0] |
|---------------------------------|-------------------|--------------|---------------|-------------|------------------|
| YCbCr-422 (8 bits) transfer | 000 | 0 | 0 | 0 | 00 |

(c) Output: UYVY, 10 bits, format and Register Settings

| | | | | | | | | | |
|--------------|----|----|----|----|---|---|----------|---|---|
| D63 to D48 | 63 | 58 | 57 | 48 | | | | | |
| Image data 4 | 0 | 0 | 0 | 0 | 0 | 0 | Y1[7:0] | * | * |
| D47 to D32 | 47 | 42 | 41 | 32 | | | | | |
| Image data 3 | 0 | 0 | 0 | 0 | 0 | 0 | Cr0[7:0] | * | * |
| D31 to D16 | 31 | 30 | 25 | 16 | | | | | |
| Image data 2 | 0 | 0 | 0 | 0 | 0 | 0 | Y0[7:0] | * | * |
| D15 to D0 | 15 | 10 | 9 | 0 | | | | | |
| Image data 1 | 0 | 0 | 0 | 0 | 0 | 0 | Cb0[7:0] | * | * |

Note: * Invalid Data.

| Format of Data Stored in Memory | VnDMR/ YMODE[2:0] | VnDMR/ EXRGB | VnDMR/ YC_THR | VnDMR/ DTMD[1:0] |
|--|------------------------------|-------------------------|--------------------------|-----------------------------|
| YCbCr-422 (10 or 12 bits) transfer* | 000 | 0 | 1 | 00 |

(d) Output: YC Separation, 8 bits format and Register Settings

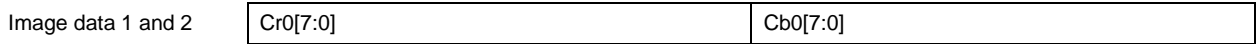
Y data

D15 to D0 15 8 7 0



Cb, Cr data

D15 to D0 15 8 7 0

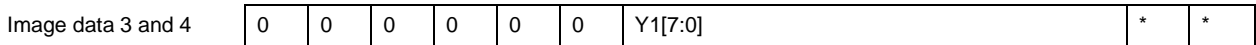


| Format of Data Stored in Memory | VnDMR/ YMODE[2:0] | VnDMR/ EXRGB | VnDMR/ YC_THR | VnDMR/ DTMD[1:0] |
|-------------------------------------|----------------------|-----------------|------------------|---------------------|
| Y (8 bits)/CbCr separation transfer | 000 | 0 | 0 | 10 |

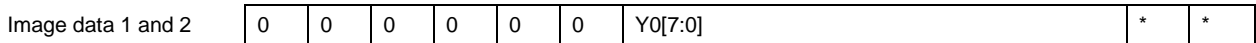
(e) Output: YC Separation, 10 bits format and Register Settings

Y data

D31 to D16 31 30 25 16

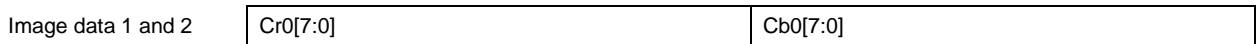


D15 to D0 15 10 9 0



Cb, Cr data

D15 to D0 15 8 7 0



Note: * Invalid Data.

| Format of Data Stored in Memory | VnDMR/ YMODE[2:0] | VnDMR/ EXRGB | VnDMR/ YC_THR | VnDMR/ DTMD[1:0] |
|--------------------------------------|----------------------|-----------------|------------------|---------------------|
| Y (16 bits)/CbCr separation transfer | 010 | 0 | 0 | 10 |

(f) Output: YC Separation Y(10 bits) / C(10 bits) format and Register Settings

Y data (10 bits)

| | | | | | | | | | | | | | | | | |
|--------------|----|---|---|---|---|---|----|----|---------|--|--|--|--|--|---|----|
| D31 to D16 | 31 | | | | | | 30 | 25 | | | | | | | | 16 |
| Image data 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Y1[7:0] | | | | | | * | * |

| | | | | | | | | | | | | | | | | |
|--------------|----|---|---|---|---|---|----|---|---------|--|--|--|--|--|---|---|
| D15 to D0 | 15 | | | | | | 10 | 9 | | | | | | | | 0 |
| Image data 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Y0[7:0] | | | | | | * | * |

Cb, Cr data (10 bits)

| | | | | | | | | | | | | | | | | |
|--------------|----|---|---|---|---|---|----|----|----------|--|--|--|--|--|---|----|
| D31 to D16 | 31 | | | | | | 30 | 25 | | | | | | | | 16 |
| Image data 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Cr0[7:0] | | | | | | * | * |

| | | | | | | | | | | | | | | | | |
|--------------|----|---|---|---|---|---|----|---|----------|--|--|--|--|--|---|---|
| D15 to D0 | 15 | | | | | | 10 | 9 | | | | | | | | 0 |
| Image data 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Cb0[7:0] | | | | | | * | * |

Note: *Invalid Data.

| Format of Data Stored in Memory | VnDMR/ YMODE[2:0] | VnDMR/ EXRGB | VnDMR/ YC_THR | VnDMR/ DTMD[1:0] |
|---|-------------------|--------------|---------------|------------------|
| Y (10 or 12 bits)/CbCr separation transfer* | 000 | 0 | 1 | 10 |

(2) 12 bits (Valid data is 8bit)

(a) Input: UYVY format and Register Settings

12-bit YCbCr-422 data (UYVY format)

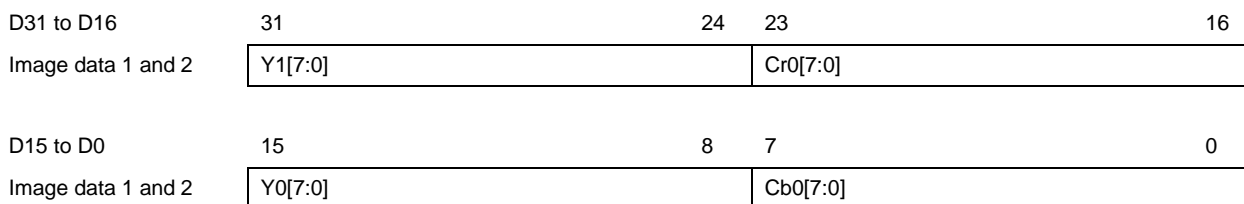
| | | | | | | | | | | | | | | | | |
|-------------------------|----|----|----|----|----------|----|---|---|---|---|---|---|---|---|---|---|
| VIn_DATA15 to VIn_DATA0 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Y1 | * | * | * | * | Y1[7:0] | | | | | | * | * | * | * | | |
| Cr0 | * | * | * | * | Cr0[7:0] | | | | | | * | * | * | * | | |
| Y0 | * | * | * | * | Y0[7:0] | | | | | | * | * | * | * | | |
| Cb0 | * | * | * | * | Cb0[7:0] | | | | | | * | * | * | * | | |

Note: * Fix the pins at a high or low level.

| Interface | VnMC/EXINF | VnMC/INF | VnMC/BPS | |
|----------------------|------------|----------|----------|---|
| Multiplexed YCbCr422 | 12 bits | 11 | 011 | 1 |

(b) Output: UYVY, 8 bits, format and Register Settings

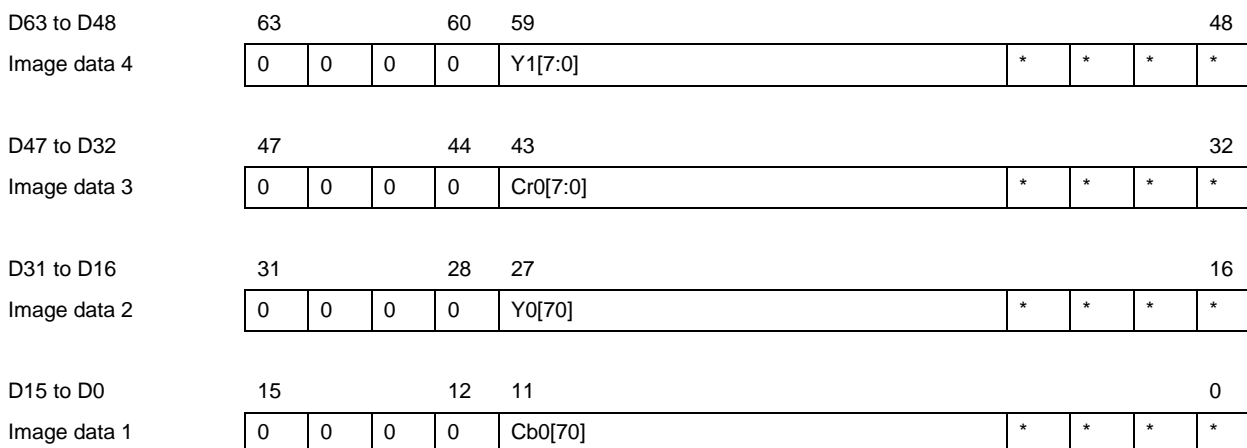
8-bit YCbCr-422 data (UYVY format)



| Format of Data Stored in Memory | VnDMR/ YMODE[2:0] | VnDMR/ EXRGB | VnDMR/ YC_THR | VnDMR/ BPSM | VnDMR/ DTMD[1:0] |
|---------------------------------|-------------------|--------------|---------------|-------------|------------------|
| YCbCr-422 (8 bits) transfer | 000 | 0 | 0 | 0 | 00 |

(c) Output: UYVY, 12 bits, format and Register Settings

12-bit YCbCr-422 data (UYVY format)



Note: * Invalid Data.

| Format of Data Stored in Memory | VnDMR/ YMODE[2:0] | VnDMR/ EXRGB | VnDMR/ YC_THR | VnDMR/ DTMD[1:0] |
|-------------------------------------|-------------------|--------------|---------------|------------------|
| YCbCr-422 (10 or 12 bits) transfer* | 000 | 0 | 1 | 00 |

(d) Output: YC Separation, 8 bits format and Register Settings

Y data

| | | | | |
|--------------------|---------|---|---------|---|
| D15 to D0 | 15 | 8 | 7 | 0 |
| Image data 1 and 2 | Y1[7:0] | | Y0[7:0] | |

Cb, Cr data

| | | | | |
|--------------------|----------|---|----------|---|
| D15 to D0 | 15 | 8 | 7 | 0 |
| Image data 1 and 2 | Cr0[7:0] | | Cb0[7:0] | |

| Format of Data Stored in Memory | VnDMR/ YMODE[2:0] | VnDMR/ EXRGB | VnDMR/ YC_THR | VnDMR/ DTMD[1:0] |
|-------------------------------------|-------------------|--------------|---------------|------------------|
| Y (8 bits)/CbCr separation transfer | 000 | 0 | 0 | 10 |

(e) Output: YC Separation, 12 bits format and Register Settings

Y data

| | | | | | | | | | |
|--------------------|----|----|----|----|---------|---|---|---|---|
| D31 to D16 | 31 | 28 | 27 | 16 | | | | | |
| Image data 3 and 4 | 0 | 0 | 0 | 0 | Y1[7:0] | * | * | * | * |

| | | | | | | | | | |
|--------------------|----|----|----|---|---------|---|---|---|---|
| D15 to D0 | 15 | 12 | 11 | 0 | | | | | |
| Image data 3 and 4 | 0 | 0 | 0 | 0 | Y0[7:0] | * | * | * | * |

Cb, Cr data

| | | | | |
|--------------------|----------|---|----------|---|
| D15 to D0 | 15 | 8 | 7 | 0 |
| Image data 1 and 2 | Cr0[7:0] | | Cb0[7:0] | |

Note: * Invalid Data.

| Format of Data Stored in Memory | VnDMR/ YMODE[2:0] | VnDMR/ EXRGB | VnDMR/ YC_THR | VnDMR/ DTMD[1:0] |
|--------------------------------------|-------------------|--------------|---------------|------------------|
| Y (16 bits)/CbCr separation transfer | 100 | 0 | 0 | 10 |

(f) Output: YC Separation Y(12 bits) / C(12 bits) format and Register Settings

Y data (12 bits)

| | | | | | | | | | | | | |
|--------------|----|---|---|----|---------|--|--|----|---|---|----|---|
| D31 to D16 | 31 | | | 28 | 27 | | | 20 | | | 16 | |
| Image data 2 | 0 | 0 | 0 | 0 | Y1[7:0] | | | | * | * | * | * |

| | | | | | | | | | | | | |
|--------------|----|---|---|----|---------|--|--|--|---|---|---|---|
| D15 to D0 | 15 | | | 12 | 11 | | | | | | 0 | |
| Image data 1 | 0 | 0 | 0 | 0 | Y0[7:0] | | | | * | * | * | * |

Cb, Cr data (12 bits)

| | | | | | | | | | | | | |
|--------------|----|---|---|----|----------|--|--|--|---|---|----|---|
| D31 to D16 | 31 | | | 28 | 27 | | | | | | 16 | |
| Image data 2 | 0 | 0 | 0 | 0 | Cr0[7:0] | | | | * | * | * | * |

| | | | | | | | | | | | | |
|--------------|----|---|---|----|----------|--|--|--|---|---|---|---|
| D15 to D0 | 15 | | | 12 | 11 | | | | | | 0 | |
| Image data 1 | 0 | 0 | 0 | 0 | Cb0[7:0] | | | | * | * | * | * |

Note: * Invalid Data.

| Format of Data Stored in Memory | VnDMR/ YMODE[2:0] | VnDMR/ EXRGB | VnDMR/ YC_THR | VnDMR/ DTMD[1:0] | Remarks |
|---|----------------------|-----------------|------------------|---------------------|--|
| Y (10 or 12 bits)/CbCr separation transfer* | 000 | 0 | 1 | 10 | Data selection of 10-bit or 12-bit is according to the input format. |

30.3.17 Initialize Procedure

The Initialization procedure of VIN is as followings.

(1) Case of inputting images from CSI-2 module

It is possible to run VIN initial procedure either before or after CSI2 initial procedure.

[VIN initial procedure]

1. Set VIN registers.
2. Set the VSYNC field toggle mode enable (FTEV) bit in the Video n data mode register 2 (VnDMR2) to 1. The VSYNC field toggle mode transition period (VLV[3:0]) bit in the VnDMR2 shall be set to 4'h0.
3. Set the module enable (ME) bit in the main control register (VnMC) to 1 to start the video input module.
4. Set the continuous frame capture (CC) bit or the single frame capture (SC) bit in the frame capture register (VnFC) to 1.

[CSI2 initial procedure]

1. Set the Link and PHY of CSI-2 module registers*¹.
2. Start camera device.
3. Confirm PHY of CSI-2 module starts.

(2) Case of inputting images from Digital Pins

1. Set pin function controller (PFC) registers*².
2. Start camera device.
3. Set SRTm bit to 1 in the software reset register n (SRCRn) of the clock pulse generator (CPG) module*⁴.
4. Set SRTCLRm to 1 in software reset clearing register n (SRSTCLRn) of clock pulse generator (CPG) module*⁴.
5. Set VIN registers.
6. Set the module enable (ME) bit in the main control register (VnMC) to 1 to start the video input module.
7. Set the continuous frame capture (CC) bit or the single frame capture (SC) bit in the frame capture register (VnFC) to 1.

(3) Case of inputting images from CSI-2 module and Digital Pins

It is possible to run VIN initial procedure either before or after CSI2 initial procedure.

[VIN initial procedure]

1. Set pin function controller (PFC) registers*².
2. Start camera side device of Digital Pins side.
3. Set SRTm bit to 1 in the software reset register n (SRCRn) of the clock pulse generator (CPG) module*⁴.
4. Set SRTCLRm to 1 in software reset clearing register n (SRSTCLRn) of clock pulse generator (CPG) module*⁴.
5. Set VIN registers.
6. Set the VSYNC field toggle mode enable (FTEV) bit in the Video n data mode register 2 (VnDMR2) to 1. The VSYNC field toggle mode transition period (VLV[3:0]) bit in the VnDMR2 shall be set to 4'h0*³.
7. Set the module enable (ME) bit in the main control register (VnMC) to 1 to start the video input module.
8. Set the continuous frame capture (CC) bit or the single frame capture (SC) bit in the frame capture register (VnFC) to 1.

[CSI2 initial procedure]

1. Set the Link and PHY of CSI-2 module registers*¹.
2. Start camera device of MIPI-CSI-2 side.
3. Confirm PHY of CSI-2 module starts.

Notes: 1. For detail, see section 29.
2. For detail, see section 8.
3. This setting is necessary to the channel which input images from CSI29-2 module.
4. For detail, see section 12.

30.3.18 Capture Stop Procedure

The capture stop procedure of VIN is as followings.

(1) Case of inputting images from CSI-2 module

1. Clear the module enable (ME) bit in the main control register (VnMC) and the continuous frame capture (CC) bit and the single frame capture (SC) bit in the frame capture register (VnFC) to 0 to stop the video input module.
2. Confirm that the capture active (CA) bit in the module status register (VnMS) is cleared to 0*³.
3. Set MSTPm bit to 1 in the module stop control register n (RMSTPCRn) of the clock pulse generator (CPG) module*¹.
4. Confirm that MSTPSTm bit to 1 in the module stop status register n (MSTPSRn) of the clock pulse generator (CPG) module*¹.
5. Set SRTm bit to 1 in the software reset register n (SRCRn) of the clock pulse generator (CPG) module*¹.
6. Reset the Link and PHY of CSI-2 module*².
7. Stop camera device.

(2) Case of inputting images from Digital Pins

1. Clear the module enable (ME) bit in the main control register (VnMC) and the continuous frame capture (CC) bit and the single frame capture (SC) bit in the frame capture register (VnFC) to 0 to stop the video input module.
2. Confirm that the capture active (CA) bit in the module status register (VnMS) is cleared to 0*³.

(3) Case of inputting images from CSI-2 module and Digital Pins

1. Clear the module enable (ME) bit in the main control register (VnMC) and the continuous frame capture (CC) bit and the single frame capture (SC) bit in the frame capture register (VnFC) to 0 to stop the video input module.
2. Confirm that the capture active (CA) bit in the module status register (VnMS) is cleared to 0*³.
3. Set MSTPm bit to 1 in the module stop control register n (RMSTPCRn) of the clock pulse generator (CPG) module*¹.
4. Confirm that MSTPSTm bit to 1 in the module stop status register n (MSTPSRn) of the clock pulse generator (CPG) module*¹.
5. Set SRTm bit to 1 in the software reset register n (SRCRn) of the clock pulse generator (CPG) module*¹.
6. Reset the Link and PHY of CSI-2 module*².
7. Stop camera device of MIPI-CSI-2 side.

Notes: 1. For detail, see section 12.

2. For detail, see section 29.

3. The camera device should be kept operation until the CA bit in VnMS set to 0.

30.3.19 Capture Restarting procedure

The capture restarting procedure of VIN is as followings.

(1) Case of inputting images from CSI-2 module

It is possible to run VIN initial procedure either before or after CSI2 initial procedure.

[VIN initial procedure]

1. Set SRTCLR_m to 1 in software reset clearing register n (SRSTCLR_n) of clock pulse generator (CPG) module*¹.
2. Set MSTP_m bit to 0 in the module stop control register n (RMSTPCR_n) of the clock pulse generator (CPG) module*¹.
3. Confirm that MSTPST_m bit to 0 in the module stop status register n (MSTPSR_n) of the clock pulse generator (CPG) module*¹.
4. Set VIN registers.
5. Set the VSYNC field toggle mode enable (FTEV) bit in the Video n data mode register 2 (VnDMR2) to 1. The VSYNC field toggle mode transition period (VLV[3:0]) bit in the VnDMR2 shall be set to 4'h0*³.
6. Set the module enable (ME) bit in the main control register (VnMC) to 1 to start the video input module.
7. Set the continuous frame capture (CC) bit or the single frame capture (SC) bit in the frame capture register (VnFC) to 1.

[CSI2 initial procedure]

1. Set the Link and PHY of CSI-2 module registers*².
2. Start camera device.
3. Confirm PHY of CSI-2 module starts.

(2) Case of inputting images from Digital Pins

1. Set the module enable (ME) bit in the main control register (VnMC) to 1 to start the video input module.
2. Set the continuous frame capture (CC) bit or the single frame capture (SC) bit in the frame capture register (VnFC) to 1.

(3) Case of inputting images from CSI-2 module and Digital Pins

It is possible to run VIN initial procedure either before or after CSI2 initial procedure.

[VIN initial procedure]

1. Set SRTCLR_m to 1 in software reset clearing register n (SRSTCLR_n) of clock pulse generator (CPG) module*¹.
2. Set MSTP_m bit to 0 in the module stop control register n (RMSTPCR_n) of the clock pulse generator (CPG) module*¹.
3. Confirm that MSTPST_m bit to 1 in the module stop status register n (MSTPSR_n) of the clock pulse generator (CPG) module*¹.
4. Set VIN registers.
5. Set the VSYNC field toggle mode enable (FTEV) bit in the Video n data mode register 2 (VnDMR2) to 1. The VSYNC field toggle mode transition period (VLV[3:0]) bit in the VnDMR2 shall be set to 4'h0*³.
6. Set the module enable (ME) bit in the main control register (VnMC) to 1 to start the video input module.
7. Set the continuous frame capture (CC) bit or the single frame capture (SC) bit in the frame capture register (VnFC) to 1.

[CSI2 initial procedure]

1. Set the Link and PHY of CSI-2 module registers*².
2. Start camera device of MIPI-CSI-2 side.
3. Confirm PHY of CSI-2 module starts.

- Notes:
1. For detail, see section 12.
 2. For detail, see section 29.
 3. This setting is necessary to the channel which input images from CSI-2 module.

30.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

30.4.1 Specification

The following shows the specification.

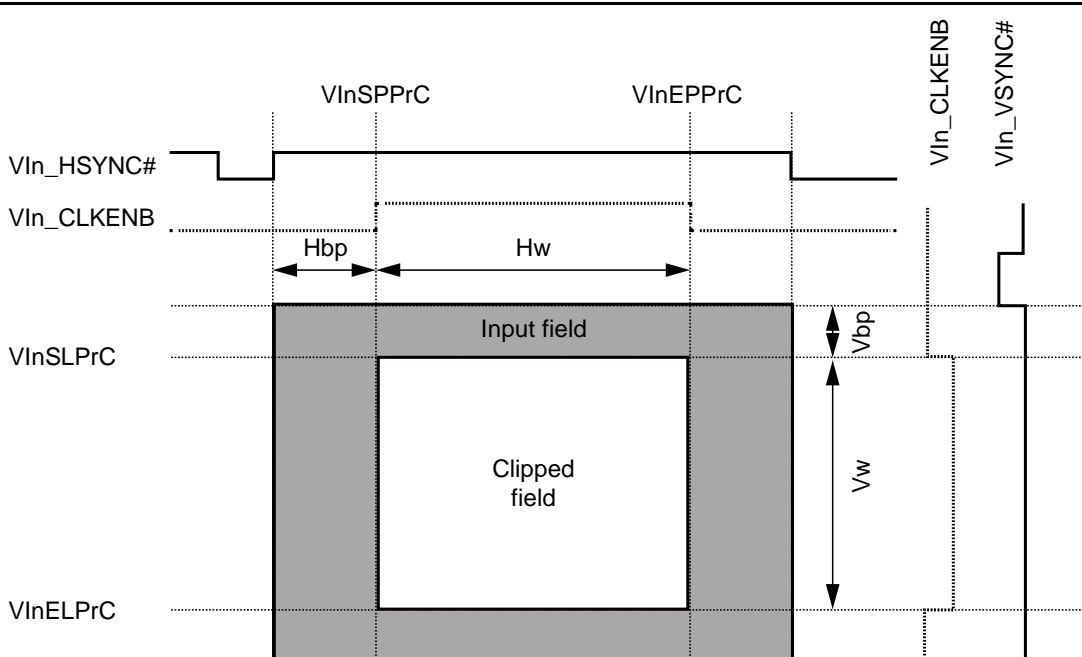
Table 30.30 Specification

| Item | Description |
|---|--|
| Limits on input video clock of Digital pin | The upper limit on frequency of the input video clock varies with the clock mode, capture interface and scaling ratio. The upper limits on frequency of the video clock for this module as a stand-alone unit are shown below. Usage of other modules may make reaching the upper limits impossible. Fully evaluate performance before deciding on the settings for the input video clock, particularly if you are using magnification. Upper limits on video-clock frequency is 100 MHz. |
| Limitation on size clipping | The upper limits on size clipping depends on the use case. |
| ITU-R BT.709 interface | When the ITU-R BT.709 interface is in use, the maximum frequency of the video clock of Digital pin is 100 MHz. |
| Capture in progressive mode | Use the internal field signal generation function in combination with an input interface in progressive mode. For details, see section 30.2.18. |
| Input from MIPI CSI-2 interface | In the input of progressive image through the MIPI CSI-2 interface, use this together with the VSYNC field toggle mode function. For detail, see section 30.2.18. |
| Limitation on register update | If a register is updated during capture, data captured immediately after the register update cannot be guaranteed. To update a register which supports the internal update mode as shown in Table 30.10 to Table 30.13, VIN Registers, specify an internal update for VIN registers by using the MC/VUP bit. To update other registers, stop capture operation and then update them. |
| Field capture mode image quality | Images of odd-numbered, odd/even-numbered and even-numbered fields, captured by VnMC/IM interlace mode bit settings, contain every other line from the input interlace images. Therefore, note that the horizontal resolution for video display is in units of fields. |
| Full interlace mode image quality | In full interlace composition mode, horizontal stripe noise (such as combing noise) is generated in composite images as fields based on different timelines are combined in memory due to the interlacing method. |
| Limitation on horizontal scaling | When scaling-up is used, horizontal size that can be input is up to 2048 pixel. When scaling-down is used, horizontal size that can be output is up to 2048 pixel. |
| Interrupt event timing | Interrupt event asserted by this module indicates the time when an interrupt event occurs in VIN, not the time when the transfer of capture data to memory is completed. |
| Pixel post-clip setting | Even if the output format is YCbCr-422, pixel post-clip is performed according VnUDS_CLIP_SIZE setting. For that reason, if the CL_HSIZE clipping size is an odd number, the Cr data of the rightmost two pixels is not output to memory. Therefore, if the output format is YCbCr-422, set the CL_HSIZE clipping size to even number. Note that the clipping size must not be set to a value larger than the size of the output image. |
| Coefficient settings for color space conversion | Specify appropriate values in the color space change coefficient 1 to 3 registers (VnCSCC1 to VnCSCC3) to keep the RGB image data within the range $0 \leq R', G', B' \leq 255$. After calculation for color space conversion, minus pixel data is normalized to 0 and pixel data more than or equal to 255 is normalized to 255. |
| Scaling up | When horizontal and vertical scaling up is specified, the amount of memory transfer becomes greater with an increase in traffic. Note the amount of traffic on the entire system when using the scaling-up function because the overall transfer efficiency of the system might degrade due to the increased use of the internal buses. |

| Item | Description |
|---|--|
| YC Restrictions on YC separation function | Set the offset register that stores UV data (VnUVAOF) such that the storage areas of Y and UV data are not the same. If the same storage area is set for both data, Y data will be overwritten. |
| RGB-888→RGB-565 conversion function | In the dithering process by cumulative addition, if the same color is captured as in blue back image, periodic noise may be generated by the cumulative addition process (carried by addition). In this case, set the DC[1:0] bits in VnMC to the ordered dithering. |
| Video display operation | This module performs capture control based on the external input synchronization signal. Note that it is not synchronous to the timing of frame update set in the Display Unit and thus video display in sync with the capture frame is disabled. |

Item Description

The maximum size that can be captured when the HSYNC signal is connected to the VIn_CLKENB pin (including the case where the CHS bit of the VnDMR2 register is set to 1).



Hbp: Horizontal back porch period
 Hw: Horizontal valid pixels
 Vbp: Vertical back porch period
 Vw: Vertical valid lines

Capture area when the HSYNC signal is connected to the VIn_CLKENB pin

When the ITU-R BT.601/709/1358 interface is in use, so the HSYNC signal is connected to the VIn_CLKENB pin (including the case where the CHS bit of the VnDMR2 register is set to 1), the VIN also captures data during the horizontal and vertical back porch periods (Hbp and Vbp). Thus, capturing only the valid data is required by setting the clipping registers as shown below.

- VnSLPrC (start line pre-clip register) = vertical back porch period (Vbp)
- VnELPrC (end line pre-clip register) = vertical back porch period (Vbp) + number of valid lines (Vw) – 1
- VnSPPrC (start pixel pre-clip register) = horizontal back porch period (Hbp)
- VnEPPrC (end pixel pre-clip register) = horizontal back porch period (Hbp) + number of valid pixels (Hw) – 1

The maximum number of lines and pixels that can be captured at this time is as shown below. When this size is exceeded, use the VIN with the data enable signal connected to the VIn_CLKENB pin.

- The maximum number of lines that can be captured = 4096 (the maximum value of the clipping register) – vertical back porch period (Vbp)
- The maximum number of pixels that can be captured = 4096 (the maximum value of the clipping register) – horizontal back porch period (Hbp)

Note: The horizontal back porch period unit shall be the followings.

| Capture data | Horizontal back porch period unit |
|----------------|-----------------------------------|
| YCbCr-422 data | 2 pixels |
| RGB data | 2 pixels |
| RAW data | 4 pixels |

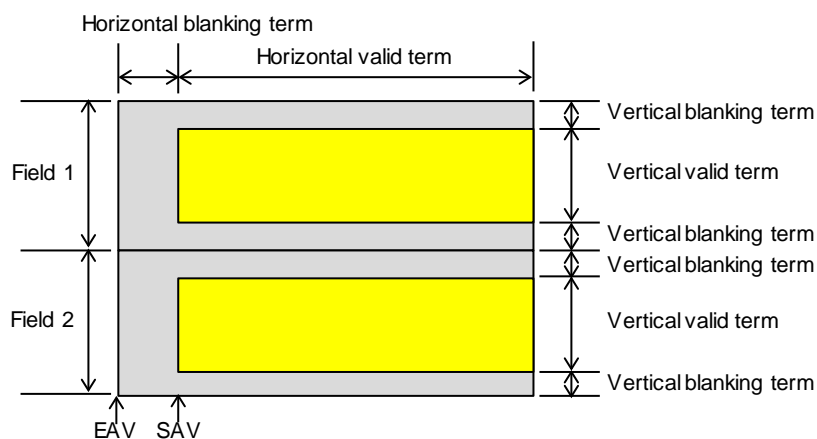
| Item | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--|--|--------|---------|---------|---------|---------|---------|---------|---------|--|---------------|---|---|---|---|---|---|----------------|---|---|---|---|---|---|----------------|---|---|---|---|---|---|----------------|---|---|---|---|---|---|----------------|---|---|---|---|---|---|----------------|---|---|---|---|---|---|----------------|---|---|---|---|---|---|--|--|--------|---------|---------|---------|---------|---------|------------------------------------|---------------|-----|-----|-----|-----|---|---|----------------|-----|-----|-----|-----|---|---|----------------|-----|-----|-----|-----|---|---|----------------|-------|-------|-------|-------|---|---|----------------|---|---|---|---|---|---|----------------|---|---|---|---|---|---|----------------|---|---|---|---|---|
| Limits on input format | When the two channels use the digital pins, there are limits on input format. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | <p style="text-align: center;">Video Channels 5 for , RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N</p> <table border="1"> <thead> <tr> <th></th> <th></th> <th>8 bits</th> <th>10 bits</th> <th>12 bits</th> <th>16 bits</th> <th>20 bits</th> <th>24 bits</th> </tr> </thead> <tbody> <tr> <td rowspan="7">Video Channels 4 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N</td> <td>8 bits</td> <td>√</td> <td>√</td> <td>√</td> <td>√</td> <td>—</td> <td>—</td> </tr> <tr> <td>10 bits</td> <td>√</td> <td>√</td> <td>√</td> <td>√</td> <td>—</td> <td>—</td> </tr> <tr> <td>12 bits</td> <td>√</td> <td>√</td> <td>√</td> <td>√</td> <td>—</td> <td>—</td> </tr> <tr> <td>16 bits</td> <td>√</td> <td>√</td> <td>√</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>18 bits</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>20 bits</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>24 bits</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> </tbody> </table> <p style="text-align: center;">Video Channels 5 for RZ/G2E</p> <table border="1"> <thead> <tr> <th></th> <th></th> <th>8 bits</th> <th>10 bits</th> <th>12 bits</th> <th>16 bits</th> <th>20 bits</th> <th>24 bits</th> </tr> </thead> <tbody> <tr> <td rowspan="7">Video Channels 4 for RZ/G2E</td> <td>8 bits</td> <td>√*1</td> <td>√*1</td> <td>√*1</td> <td>√*1</td> <td>—</td> <td>—</td> </tr> <tr> <td>10 bits</td> <td>√*1</td> <td>√*1</td> <td>√*1</td> <td>√*1</td> <td>—</td> <td>—</td> </tr> <tr> <td>12 bits</td> <td>√*1</td> <td>√*1</td> <td>√*1</td> <td>√*1</td> <td>—</td> <td>—</td> </tr> <tr> <td>16 bits</td> <td>√*1*2</td> <td>√*1*2</td> <td>√*1*2</td> <td>√*1*2</td> <td>—</td> <td>—</td> </tr> <tr> <td>18 bits</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>20 bits</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>24 bits</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> </tr> </tbody> </table> | | | 8 bits | 10 bits | 12 bits | 16 bits | 20 bits | 24 bits | Video Channels 4 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N | 8 bits | √ | √ | √ | √ | — | — | 10 bits | √ | √ | √ | √ | — | — | 12 bits | √ | √ | √ | √ | — | — | 16 bits | √ | √ | √ | — | — | — | 18 bits | — | — | — | — | — | — | 20 bits | — | — | — | — | — | — | 24 bits | — | — | — | — | — | — | | | 8 bits | 10 bits | 12 bits | 16 bits | 20 bits | 24 bits | Video Channels 4 for RZ/G2E | 8 bits | √*1 | √*1 | √*1 | √*1 | — | — | 10 bits | √*1 | √*1 | √*1 | √*1 | — | — | 12 bits | √*1 | √*1 | √*1 | √*1 | — | — | 16 bits | √*1*2 | √*1*2 | √*1*2 | √*1*2 | — | — | 18 bits | — | — | — | — | — | — | 20 bits | — | — | — | — | — | — | 24 bits | — | — | — | — | — |
| | | 8 bits | 10 bits | 12 bits | 16 bits | 20 bits | 24 bits | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Video Channels 4 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N | 8 bits | √ | √ | √ | √ | — | — | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 10 bits | √ | √ | √ | √ | — | — | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 12 bits | √ | √ | √ | √ | — | — | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 16 bits | √ | √ | √ | — | — | — | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 18 bits | — | — | — | — | — | — | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 20 bits | — | — | — | — | — | — | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 24 bits | — | — | — | — | — | — | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | 8 bits | 10 bits | 12 bits | 16 bits | 20 bits | 24 bits | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Video Channels 4 for RZ/G2E | 8 bits | √*1 | √*1 | √*1 | √*1 | — | — | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 10 bits | √*1 | √*1 | √*1 | √*1 | — | — | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 12 bits | √*1 | √*1 | √*1 | √*1 | — | — | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 16 bits | √*1*2 | √*1*2 | √*1*2 | √*1*2 | — | — | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 18 bits | — | — | — | — | — | — | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 20 bits | — | — | — | — | — | — | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 24 bits | — | — | — | — | — | — | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

- Notes: 1. The CHS bit in V4DMR2 shall be set to 1.
2. The FTEV bit in V5DMR2 shall be set to 1.

| Item | Description | | | | | | | | | | | |
|---|--|-------------|--------|--------|---------------|-------------|-----------|---|-----------|-----------|---------------|---|
| Available CSI2 module | There are limits in available CSI2 module. | | | | | | | | | | | |
| | <table border="1"> <thead> <tr> <th></th> <th>4 Lane</th> <th>2 Lane</th> </tr> </thead> <tbody> <tr> <td>RZ/G2H</td> <td>1 modules*2</td> <td>1 modules</td> </tr> <tr> <td>RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N</td> <td>1 modules</td> <td>1 modules</td> </tr> <tr> <td>RZ/G2E</td> <td>—</td> <td>1 modules*1</td> </tr> </tbody> </table> | | 4 Lane | 2 Lane | RZ/G2H | 1 modules*2 | 1 modules | RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N | 1 modules | 1 modules | RZ/G2E | — |
| | 4 Lane | 2 Lane | | | | | | | | | | |
| RZ/G2H | 1 modules*2 | 1 modules | | | | | | | | | | |
| RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N | 1 modules | 1 modules | | | | | | | | | | |
| RZ/G2E | — | 1 modules*1 | | | | | | | | | | |

- Notes: 1. The input from Virtual Channel 2 and Virtual Channel 3 are not supported.
2. CSI 4 Lane module is available only in video channels 0 to 3.

| Item | Description | | | |
|------------------------|---|---|-----------------------------|-----------------------------|
| ITU-R BT.656 interface | VIN expects the sequence of the followings. | | | |
| | Field | Line Condition | From | To |
| | Field 1 | During Vertical Blanking term | EAV : (F, V, H) = (0, 1, 1) | SAV : (F, V, H) = (0, 1, 0) |
| | | Transition to Vertical Valid term from Vertical Blanking term | SAV : (F, V, H) = (0, 1, 0) | EVA : (F, V, H) = (0, 0, 1) |
| | | During Vertical Valid term | EAV : (F, V, H) = (0, 0, 1) | SAV : (F, V, H) = (0, 0, 0) |
| | | End of Vertical Valid term | SAV : (F, V, H) = (0, 0, 0) | EAV : (F, V, H) = (0, 1, 1) |
| | | During Vertical Blanking term (Minimum 3line) | EAV : (F, V, H) = (0, 1, 1) | SAV : (F, V, H) = (0, 1, 0) |
| | | Transition to Field 2 from Field 1 | SAV : (F, V, H) = (0, 1, 0) | EAV : (F, V, H) = (1, 1, 1) |
| | Field 2 | During Vertical Blanking term | EAV : (F, V, H) = (1, 1, 1) | SAV : (F, V, H) = (1, 1, 0) |
| | | Transition to Vertical Valid term from Vertical Blanking term | SAV : (F, V, H) = (1, 1, 0) | EAV : (F, V, H) = (1, 0, 1) |
| | | During Vertical Valid term | EAV : (F, V, H) = (1, 0, 1) | SAV : (F, V, H) = (1, 0, 0) |
| | | End of Vertical Valid term | SAV : (F, V, H) = (1, 0, 0) | EAV : (F, V, H) = (1, 1, 1) |
| | | During Vertical Blanking term (Minimum 3line) | EAV : (F, V, H) = (1, 1, 1) | SAV : (F, V, H) = (1, 1, 0) |
| | | Transition to Field 1 from Field 2 | SAV : (F, V, H) = (1, 1, 0) | EAV : (F, V, H) = (0, 1, 1) |



| | |
|-------------------------------|--|
| Clip size setting | Pre-Clip size of Odd and Even field is set to the common by VnSLPrC, VnELPrC, VnSPPrC and VnEPPrC. Post-Clip size of Odd and Even field is set to the common by VnUDS_CLIP_SIZE. |
| Limitation on interlace input | When the interlace image is input, be sure to supply the field signal to VIN. When the interlace image is input from the MIPI CSI-2 interface, it is necessary to create the field signal in CSI2 module. For the detail of the field signal creation in CSI2, see section 29. |

| Item | Description | | | | | | | | | | | | |
|--|--|--------------------|-------------------------|--------------------|----------------|----------|----------|----------|----------|----------|----------|----------|----------|
| The change of the video clock of Digital pin | <p>VIN doesn't support changing the video clock of Digital pin during VIN operation. After VIN operation start, if the video clock of Digital pin is changed, follow the following procedures. For detail of RMSTPCR8 and MSTPSR8 and SRSTCLR8, see section 12.</p> <p>[1] Set 1 to the MSTPn bit in RMSTPCR8 n = 807(VIN4), 806(VIN5) (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0)</p> <p>[2] Wait until the MSTPSTn bit in MSTPSR8 is set to 1 n = 807(VIN4), 806(VIN5) (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0)</p> <p>[3] Set 1 to the SRTn bit in SRCR8 n = 807(VIN4), 806(VIN5) (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0)</p> <p>[4] Change the video clock of Digital pin</p> <p>[5] Set 1 to the SRTCLRn bit in SRSTCLR8 n = 807(VIN4), 806(VIN5) (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0)</p> <p>[6] Set 0 to the MSTPn bit in RMSTPCR8n = 807(VIN4), 806(VIN5) RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0)</p> <p>[7] Wait until the MSTPSTn bit in MSTPSR8 is set to 0 n = 807(VIN4), 806(VIN5) (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0)</p> <p>[8] Configure the VIN register</p> | | | | | | | | | | | | |
| UDS scaler setting sequence | <p>[Initial setting ~ Capture start]</p> <p>[1] Set SCLE bit in VnMC to 1. Set other bits in VnMC to 0.</p> <p>[2] Set UDS registers.</p> <p>[3] Set VIN registers.</p> <p>[4] Set ME bit in VnMC to 1.</p> <p>[5] Set CC bit in VnFC to 1.</p> <p>[6] Capture start</p> <p>[Flow of capture stop and UDS registers disable]</p> <p>[1] Set VUP and ME bit in VnMC to 0. Hold other bits in VnMC to current value.</p> <p>[2] Set CC bit in VnFC to 0.</p> <p>[3] Wait until CA bit in VnMS is set to 0. (Maximum 2-vsyc)</p> <p>[4] Disable UDS registers.</p> <p>[Capture Restart in case of using UDS]</p> <p>[5] Change UDS registers.</p> <p>[6] Set ME bit VnMC to 1.</p> <p>[7] Set CC bit in VnFC to 1.</p> <p>[8] Capture restart. (After maximum 2-vsyc)</p> | | | | | | | | | | | | |
| Horizontal clipping specification | <p>The horizontal clipping size shall be set as the followings.</p> <table border="1"> <thead> <tr> <th>Capture data</th> <th>Pre-clipping start unit</th> <th>Clipping size unit</th> </tr> </thead> <tbody> <tr> <td>YCbCr-422 data</td> <td>2 pixels</td> <td>2 pixels</td> </tr> <tr> <td>RGB data</td> <td>2 pixels</td> <td>2 pixels</td> </tr> <tr> <td>RAW data</td> <td>4 pixels</td> <td>4 pixels</td> </tr> </tbody> </table> | Capture data | Pre-clipping start unit | Clipping size unit | YCbCr-422 data | 2 pixels | 2 pixels | RGB data | 2 pixels | 2 pixels | RAW data | 4 pixels | 4 pixels |
| Capture data | Pre-clipping start unit | Clipping size unit | | | | | | | | | | | |
| YCbCr-422 data | 2 pixels | 2 pixels | | | | | | | | | | | |
| RGB data | 2 pixels | 2 pixels | | | | | | | | | | | |
| RAW data | 4 pixels | 4 pixels | | | | | | | | | | | |
| Vertical clipping specification | When YCbCr420 format is output, the vertical clipping size shall be set to even lines. | | | | | | | | | | | | |

30.4.2 Limitations on Usage

The VIN does not operate correctly in some cases depending on the usage. The following shows cases that require attention.

Table 30.31 Limitations on Usage

| Item | Description |
|--|--|
| Limitation on vertical scaling | For vertical scaling and full interlace composition, the capture lines are inverted in some cases depending on the scaling ratio because the scaling processing is applied before interlace composition in memory. Be sure to evaluate the image quality before practical application. |
| Image Stride setting | When outputting the NV16 format and NV12 format, set value of 32pixel unit to the image stride (IS) bit in the image stride register (VnIS). |
| Limitation of when YUV420 format is output | When YUV420 format is output, upper limits on the horizontal clip size is 2048 pixel. |
| Bus QoS setting [RZ/G2M V1.3] | When vertical scaling up is used, the bus QoS shall be set as the below according output format. RGB-888 : more than or equal to 1.6Gbyte/s YCbCr-422 8bit : more than or equal to 800Mbyte/s |

31. Video Decoding Processor for inter-device video transfer (iVDP1C)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

31.1 Overview

The iVDP1C is a low-latency video decoder which provides high definition, high fidelity capabilities of H.264/AVC and JPEG standards. Require software or the library to RENESAS for operating this module.

Figure 31.1 shows iVDP1C and related module for RZ/G2M V1.3. iVDP1C connects to FCPCI.

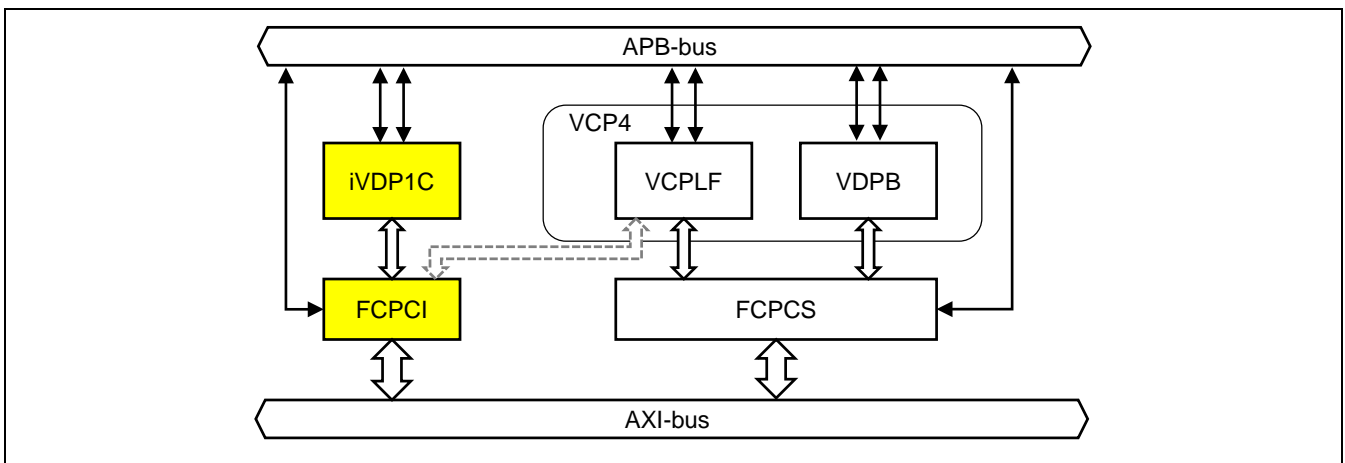


Figure 31.1 iVDP1C and related modules for RZ/G2M V1.3

Figure 31.2 and Figure 31.3 show iVDP1C and related modules (Figure 31.2 for RZ/G2H, and RZ/G2M V3.0, Figure 31.3 for RZ/G2N and RZ/G2E). iVDP1C connects FCPCS, and shares it with VCPLF and VDPB. All video codec modules, iVDP1C, VCPLF, and VDPB need to work exclusively. However, a pixel processing unit of all codec modules for RZ/G2H, RZ/G2M V3.0 and RZ/G2N have twice the performance as codec modules for RZ/G2M V1.3. So, the performance of iVDP1C for RZ/G2H, RZ/G2M V3.0, and RZ/G2N is same as iVDP1C for RZ/G2M V1.3. Note that a performance of variable length code processing unit for RZ/G2H, RZ/G2M V3.0, and RZ/G2N doesn't change from RZ/G2M V1.3. A performance beyond the RZ/G2M V1.3 cannot be realized with RZ/G2H, RZ/G2M V3.0, and RZ/G2N. For detail performance of each product, refer maximum pixel rate in section 31.1.1.

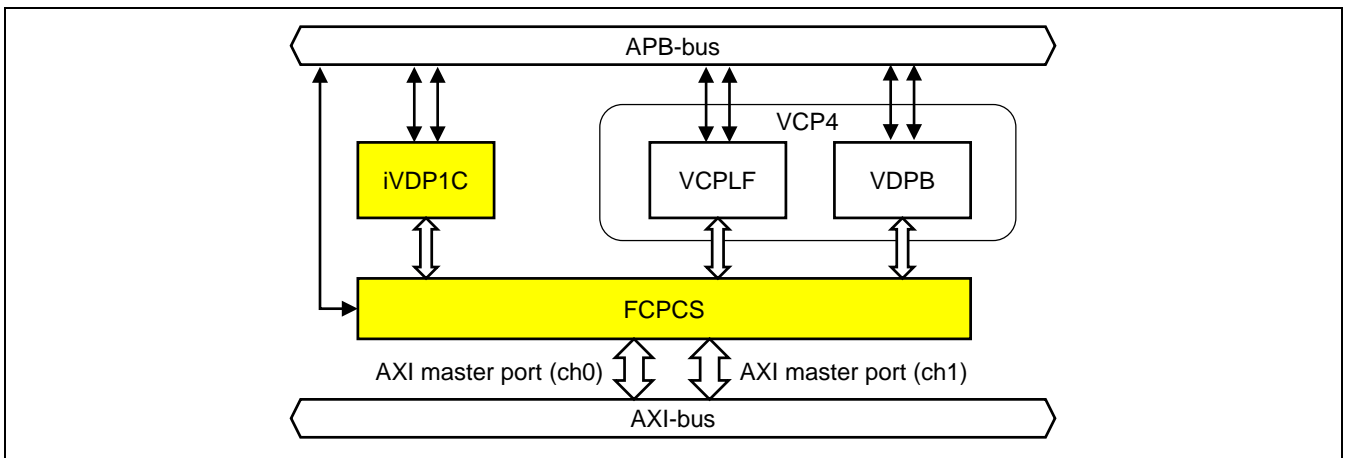


Figure 31.2 iVDP1C and related modules for RZ/G2H, and RZ/G2M V3.0

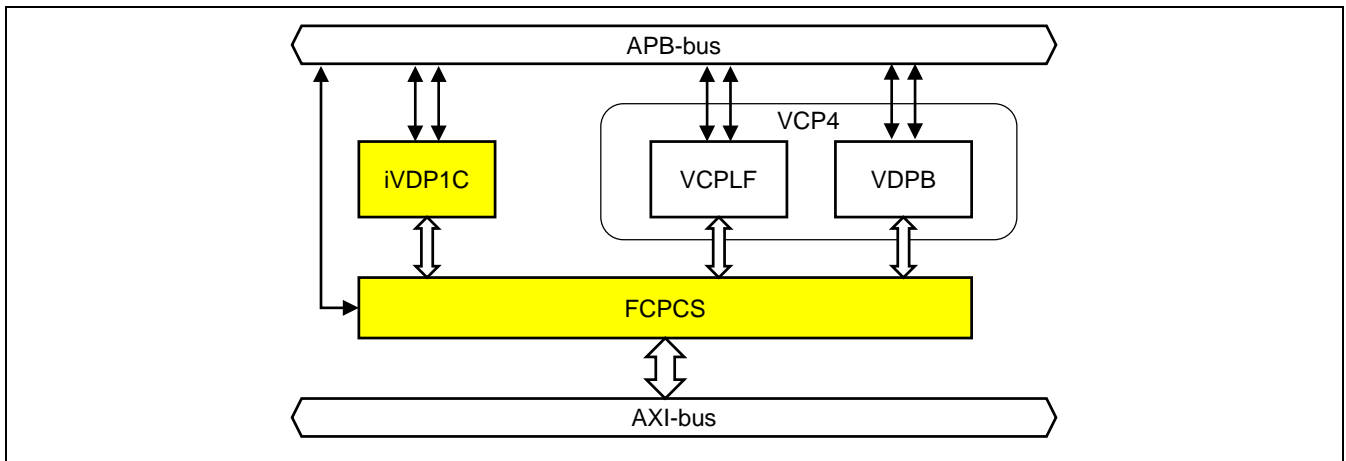


Figure 31.3 iVDP1C and related modules for RZ/G2N and RZ/G2E

The iVDP1C is connected memory bus via companion module FCPCI or FCPCS. For detail information, refer section 35. Note that lossless image compression for reference picture by FCPCI or FCPCS is supported if bit depth of image is 8 bits and color format is YCbCr 4:2:0.

Note that the VCPLF, multi-codec decoder and encoder module, for RZ/G2M V1.3 may connect to FCPCI. In this case, the iVDP1C isn't able to operate at all. For more information, refer to section 32.

31.1.1 Features

The iVDP1C has the following features.

- Support for H.264/AVC and JPEG with 8/10/12-bit depth
- Low-latency decoding
- Support 1920 × 1080 resolution
- Data handling on a picture-by-picture basis

Table 31.1 Main Function of the iVDP1C

| Item | Description | | | | |
|-----------------------------|---|---------------|---------------------------------|---------------|---------------|
| Decoding standard supported | <ul style="list-style-type: none"> H.264/AVC Baseline Profile@Level 5 (The arbitrary slice order, flexible macroblock order and redundant slice aren't supported), Constrained Baseline Profile@Level 5, High Profile@Level 5, Main Profile@Level 5, Constrained High Profile@Level 5, Progressive High Profile@Level 5, High 10 Profile@Level 5, High 10 Intra Profile @Level 5, High4:2:2 Profile@Level 5, High4:2:2 Intra Profile @Level 5, High4:4:4 Predictive Profile (*1), High4:4:4 Intra Profile (*1) MBAFF, PICAFF, B-Slice are supported in case of 8-bit depth and YCbCr 4:2:0 format. Note: 1. Only 12-bit depth samples is supported tool in this profile. H.264/MVC Stereo High Profile@Level 4.1 JPEG Baseline process (*2), Extended DCT-based processes (*3) Notes: 2. Supported function is limited as follows: DCT-based process, 8-bit depth samples, Sequential, Huffman coding (2 AC and 2 DC tables), 3 components, interleaved single scan, 4:2:0 and 4:2:2 color format. 3. Only 12-bit depth samples is supported tool in this profile. | | | | |
| Maximum pixel rate | <table border="1"> <tr> <td>RZ/G2H</td> <td>RZ/G2M V1.3, RZ/G2M V3.0</td> <td>RZ/G2N</td> </tr> </table> <p>1280 × 960p × 120 fps, or 1920 × 1080p × 60 fps</p> <table border="1"> <tr> <td>RZ/G2E</td> </tr> </table> <p>1280 × 960p × 30 fps (if VPC4 is used) 1280 × 960p × 60 fps, or 1920 × 1080p × 30 fps (if VPC4 is not used) This indicates the decoding performance including both display picture and non-display picture.</p> | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | | | |
| RZ/G2E | | | | | |
| Maximum bit rate | <table border="1"> <tr> <td>RZ/G2H</td> <td>RZ/G2M V1.3, RZ/G2M V3.0</td> <td>RZ/G2N</td> </tr> </table> <p>40 Mbps × 4 streams or 160 Mbps × 1 stream</p> <table border="1"> <tr> <td>RZ/G2E</td> </tr> </table> <p>10 Mbps × 4 streams or 40 Mbps × 1 stream (if VCP4 is used) 20 Mbps × 4 streams or 80 Mbps × 1 stream (if VCP4 is not used) Note that these streams assume constant bit rate.</p> | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | | | |
| RZ/G2E | | | | | |
| Picture size supported | <p>Minimum size: Horizontal 80 pixels × Vertical 80 lines Maximum size: Horizontal 1920 pixels × Vertical 1088 lines Horizontal size is a multiple of 8 pixels, and vertical size is a multiple of 8 lines. 320 × 180 and 854 × 480 (WVGA+) are supported.</p> | | | | |
| Color format | <p>YCbCr 4:2:0 semi-planar (Y and C plan) and YCbCr 4:2:2 semi-planar Cb/Cr and Cr/Cb interleaving order is supported.</p> | | | | |
| Bit stream format | <p>Video elementary stream as shown in Table 31.2. Function to remove the emulation prevention byte in the H.264 bit stream is supported. Function to remove the zero byte which is inserted by byte stuffing in the JPEG bit stream is supported.</p> | | | | |
| Picture structure | <p>Frame structure and field structure</p> | | | | |

Table 31.2 Data Handled by the iVDP1C

| Standard | Decoding Range |
|----------|---|
| H.264 | Slice header (other than the first slice header in picture) and subsequent data |
| JPEG | MCU data |

31.1.2 Connected Module

Table 31.3 shows modules connected to the iVDP1C.

Table 31.3 Connected Module

| Module | Connected Module | Description |
|--------|------------------|---|
| iVDP1C | FCPCI | RZ/G2M V1.3 only has this module. Reference data read cache, lossless data compression and decompression |
| | FCPCS | Reference data read cache, lossless data compression and decompression For RZ/G2M V1.3, iVDP1C doesn't connect to FCPCS. For RZ/G2H, RZ/G2N, RZ/G2M V3.0, and RZ/G2E, FCPCS is shared by VDPB, VCPLF, and iVDP1C. |

31.2 Usage Notes

31.2.1 Limitations on Software Reset, Module Standby and power-shutoff

Before executing following operations for iVDP1C, confirm that iVDP1C is not decoding a picture. If iVDP1C is decoding a picture, wait completion of decoding.

- Execute Software Reset by CPG
- Enter Module Standby (stop clock supply) by CPG
- Enter power-shutoff status (stop power supply) by SYSC

31.2.2 Note on input bitstream for decoding

In decoding process of iVDP1C, input bitstream not conformant to video codec standard might cause hang of iVDP1C.

32. Video Codec Processor (VCP4)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

32.1 Overview

The VCP4 is a multi-codec module set which provides encoding and decoding capabilities on the basis of multiple video coding schemes, e.g., H.265/HEVC, H.264/AVC. Require software or the library to RENESAS for operating this module.

The VCP4 consists of following two codec IP cores.

1. VDPB: H.265/HEVC decoder
2. VCPLF: Multi-codec decoder and encoder

Figure 32.1 and Figure 32.2 show VCP4 and related modules for RZ/G2M V1.3. There are two configurations when iVDP1C is used or not.

If iVDP1C is used, VCPLF can only connects to FCPCS. VDPB and VCPLF share FCPCS (Figure 32.1), and need to work exclusively.

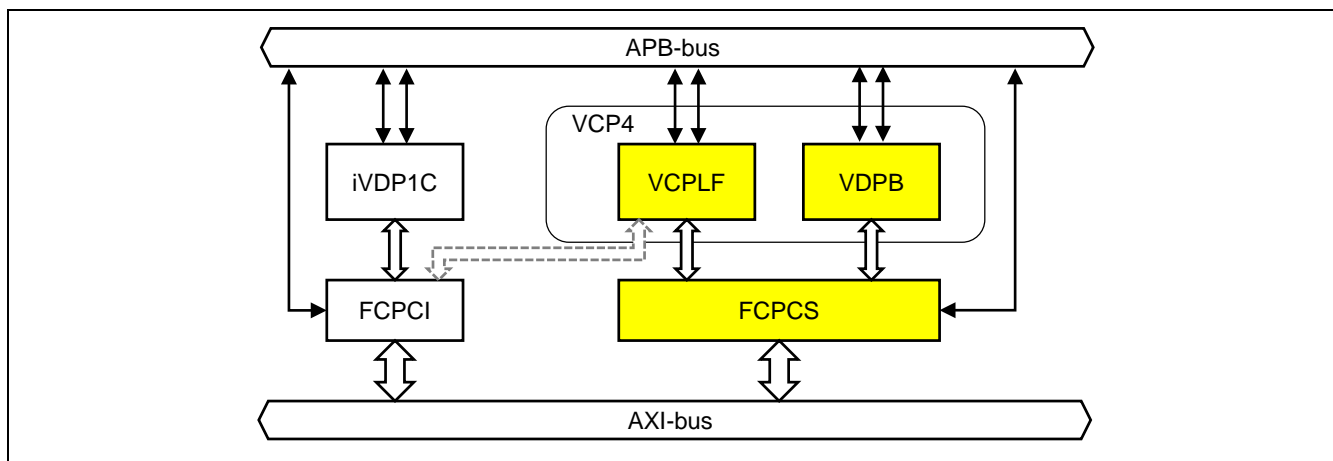


Figure 32.1 VCP4 and related modules for RZ/G2M V1.3, in Case that VCPLF Connects to FCPCS

On the other hand, if iVDP1C isn't used, VCPLF can connect to FCPCI. VDPB and VCPLF can operate simultaneously (Figure 32.2). Note that all related IPs shall stop the operation to switch the configuration.

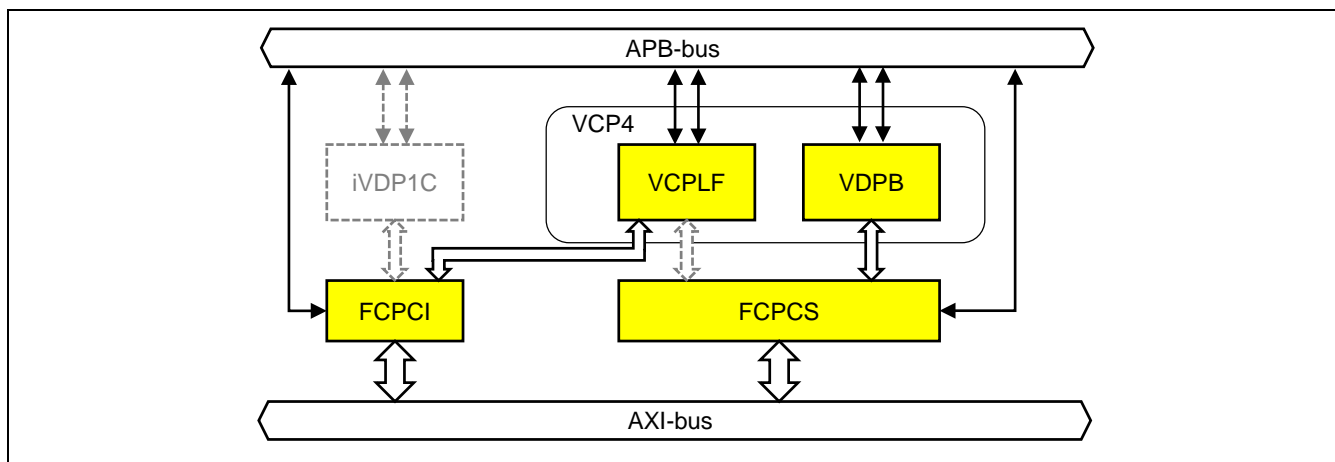


Figure 32.2 VCP4 and related modules for RZ/G2M V1.3, in Case that VCPLF Connects to FCPCI

Figure 32.3 and Figure 32.4 show VCP4 and related modules (Figure 32.3 for RZ/G2H and RZ/G2M V3.0, Figure 32.4 for RZ/G2N and RZ/G2E). All video codec modules, VDPB, VCPLF, and iVDP1C, share one FCPCS, and need to work exclusively. However, a pixel processing unit of all codec modules for RZ/G2H, RZ/G2M V3.0 and RZ/G2N have twice the performance as codec modules for RZ/G2M V1.3. So, the performance of VCP4 for RZ/G2H, RZ/G2M V3.0, and RZ/G2N is same as VCP4 for RZ/G2M V1.3. Note that a performance of variable length code processing unit for RZ/G2H, RZ/G2M V3.0 and RZ/G2N doesn't change from RZ/G2M V1.3. A performance beyond the RZ/G2M V1.3, such as H.265 3840 × 2160p × 60 fps decoding, cannot be realized with RZ/G2H, RZ/G2M V3.0 and RZ/G2N. For detail performance of each product, refer to summary of maximum pixel rate in section.32.1.1.

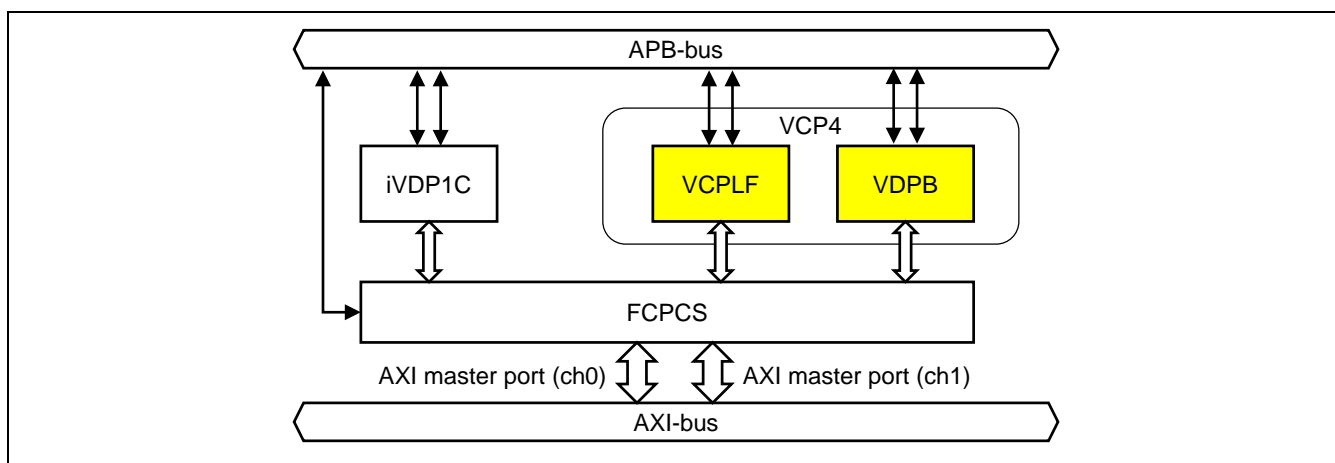


Figure 32.3 VCP4 and related modules for RZ/G2H and RZ/G2M V3.0

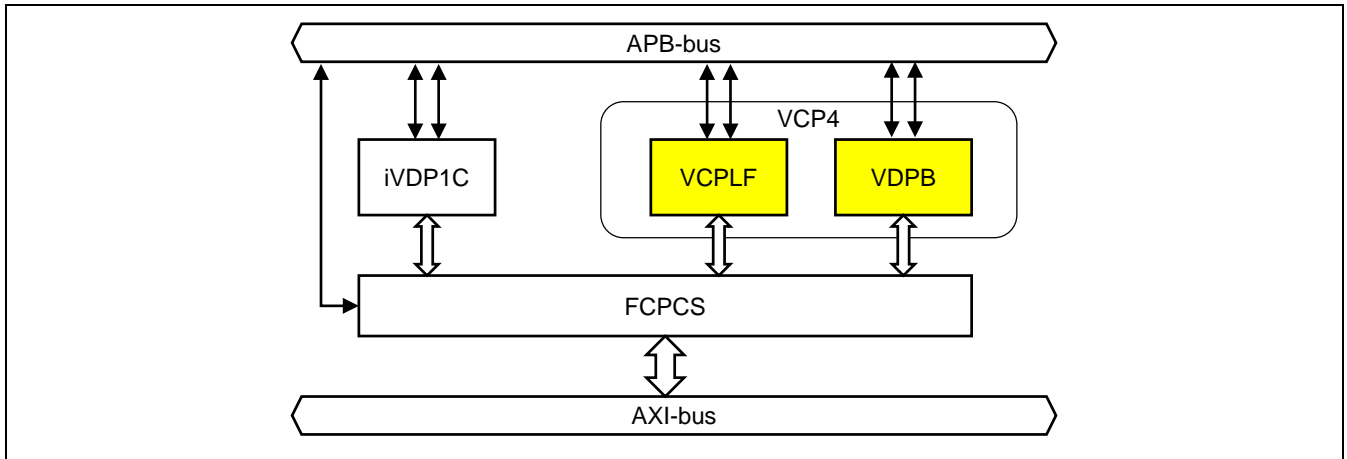


Figure 32.4 VCP4 and related modules for RZ/G2N and RZ/G2E

The VDPB and VCPLF are connected memory bus (AXI-bus) via companion modules, FCPCS or FCPCI. For detail information, refer to section 35.

The iVDP1C is a low-latency video decoder. For detail information, refer to section 31.

32.1.1 Features

(1) VDPB

The VDPB has the following features.

- Support H.265/HEVC MP (Main Profile) decoding
- Support 3840 x 2160 resolution
- Data handling on a picture-by-picture basis

Table 32.1 Main Function of the VDPB

| Item | Description |
|-----------------------------|--|
| Decoding standard supported | RZ/G2H RZ/G2M • H.265/HEVC Main Profile@Level 5 |
| | RZ/G2M V3.0 RZ/G2N • H.265/HEVC Main Profile@Level 5 |
| | RZ/G2E • H.265/HEVC Main Profile@Level 4.1 |
| Maximum pixel rate | • H.265/HEVC RZ/G2H RZ/G2M V1.3, RZ/G2M V3.0 RZ/G2N 1920 × 1080p × 120 fps, or 3840 × 2160p × 30 fps RZ/G2E 1920 × 1080p × 30 fps (if iVDP1C is used) 1920 × 1080p × 60 fps (if iVDP1C is not used) This indicates the decoding performance including both display picture and non-display picture |

| Item | Description |
|------------------------|--|
| Maximum bit rate | <ul style="list-style-type: none"> H.265/HEVC <p>RZ/G2H RZ/G2M V1.3, RZ/G2M V3.0 RZ/G2N</p> <p>30 Mbps × 4 streams, or 120 Mbps × 1 stream</p> <p>RZ/G2E</p> <p>15 Mbps × 2 streams, or 30 Mbps × 1 stream (if iVDP1C is used) 30 Mbps × 2 streams, or 60 Mbps × 1 stream (if iVDP1C is not used)</p> |
| Picture size supported | <p>Minimum:</p> <p>H.265/HEVC</p> <p>If CtbSizeY is 16, Horizontal 64 pixels × Vertical 64 lines If CtbSizeY is 32, Horizontal 96 pixels × Vertical 96 lines Otherwise (CtbSizeY is 64), Horizontal 192 pixels × Vertical 192 lines</p> <p>Maximum:</p> <p>Horizontal 3840 pixels × Vertical 2160 lines 2160 × 3840 is supported.</p> <p>RZ/G2E</p> <p>Note: When picture size is 3840 × 2160, frame rate is smaller than 30 fps. For example, if iVDP1C isn't used and codec is H.265/HEVC, frame rate is 15 fps.</p> <p>H.265/HEVC</p> <p>Horizontal size is a multiple of 2 pixels, and vertical size is a multiple of 2 lines.</p> |
| Color format | <p>YCbCr 4:2:0 semi-planar (Y and C plane)</p> <p>Cb/Cr and Cr/Cb interleaving order is supported.</p> |
| Bit stream format | <p>Video elementary stream as shown in Table 32.2.</p> <p>Function to remove the emulation prevention byte in the HEVC VCL NAL unit is supported.</p> |
| Picture structure | <p>Frame structure</p> |

Table 32.2 Data Handled by the VDPB

| Standard | Decoding Range |
|------------|--|
| H.265/HEVC | VCL NAL units in byte stream format data |

(2) VCPLF

The VCPLF has the following features.

- Support for multiple codecs
- Support 3840 × 2160 resolution (H.264 decode/encode only)
- Data handling on a picture-by-picture basis
 - Encode/decode data one picture (frame or field) at a time.
- High picture quality
 - Support the H.264 high-efficiency coding tools (CABAC, 8 × 8 frequency conversion, and quantization matrix).
 - High-efficiency motion vector detection by a combination of discrete search and trace search
 - Optimal-mode selection by Rate-Distortion (RD) cost evaluation
 - Picture quality control based on activity analysis results which match visual models

Table 32.3 Main Function of the VCPLF

| Item | Description | | | | | | | | |
|-----------------------------|--|---------------|---------------------------------|---------------|---------------|---------------|---------------------------------|---------------|---------------|
| Encoding standard supported | <ul style="list-style-type: none"> <li data-bbox="464 320 616 349">• H.264/AVC <table border="1" data-bbox="496 353 1137 394"> <tr> <td data-bbox="496 353 644 394">RZ/G2H</td> <td data-bbox="644 353 991 394">RZ/G2M V1.3, RZ/G2M V3.0</td> <td data-bbox="991 353 1137 394">RZ/G2N</td> </tr> </table> <p data-bbox="496 405 1385 633">High Profile@Level 5.1, Progressive High Profile@Level 5.1, Constrained High Profile@Level 5.1, Main Profile@Level 5.1, Constrained Baseline Profile@Level 5.1 Baseline Profile@Level 5.1 (The arbitrary slice order, flexible macroblock order and redundant slice aren't supported)</p> <table border="1" data-bbox="496 645 612 678"> <tr> <td data-bbox="496 645 612 678">RZ/G2E</td> </tr> </table> <p data-bbox="496 689 1385 913">High Profile@Level 4.2, Progressive High Profile@Level 4.2, Constrained High Profile@Level 4.2, Main Profile@Level 4.2, Constrained Baseline Profile@Level 4.2 Baseline Profile@Level 4.2 (The arbitrary slice order, flexible macroblock order and redundant slice aren't supported)</p> <li data-bbox="464 925 616 954">• H.264/MVC <table border="1" data-bbox="496 958 1137 999"> <tr> <td data-bbox="496 958 644 999">RZ/G2H</td> <td data-bbox="644 958 991 999">RZ/G2M V1.3, RZ/G2M V3.0</td> <td data-bbox="991 958 1137 999">RZ/G2N</td> </tr> </table> <p data-bbox="496 1010 820 1039">Stereo High Profile@Level 4.2</p> <table border="1" data-bbox="496 1050 612 1084"> <tr> <td data-bbox="496 1050 612 1084">RZ/G2E</td> </tr> </table> <p data-bbox="496 1095 799 1120">Stereo High Profile@Level 4</p> | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | | | | | | | |
| RZ/G2E | | | | | | | | | |
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | | | | | | | |
| RZ/G2E | | | | | | | | | |

| Item | Description |
|-----------------------------|---|
| Decoding standard supported | <ul style="list-style-type: none"> <li data-bbox="464 271 619 293">• H.264/AVC <div style="display: flex; justify-content: space-between; margin-top: 5px;"> RZ/G2H RZ/G2M V1.3, RZ/G2M V3.0 RZ/G2N </div> <p data-bbox="496 349 1385 577">High Profile@Level 5.1, Progressive High Profile@Level 5.1, Constrained High Profile@Level 5.1, Main Profile@Level 5.1, Constrained Baseline Profile@Level 5.1, Baseline Profile@Level 5.1 (The arbitrary slice order, flexible macroblock order and redundant slice aren't supported)</p> <div style="display: flex; justify-content: space-between; margin-top: 5px;"> RZ/G2E </div> <p data-bbox="496 633 1385 862">High Profile@Level 4.2, Progressive High Profile@Level 4.2, Constrained High Profile@Level 4.2, Main Profile@Level 4.2, Constrained Baseline Profile@Level 4.2, Baseline Profile@Level 4.2 (The arbitrary slice order, flexible macroblock order and redundant slice aren't supported)</p> <li data-bbox="464 880 619 902">• H.264/MVC <div style="display: flex; justify-content: space-between; margin-top: 5px;"> RZ/G2H RZ/G2M V1.3, RZ/G2M V3.0 RZ/G2N </div> <p data-bbox="496 954 820 976">Stereo High Profile@Level 4.2</p> <div style="display: flex; justify-content: space-between; margin-top: 5px;"> RZ/G2E </div> <p data-bbox="496 1032 799 1055">Stereo High Profile@Level 4</p> |

| Item | Description |
|--------------------|---|
| Maximum pixel rate | <p data-bbox="464 271 708 293">One of following items:</p> <p data-bbox="464 304 1098 338">RZ/G2H RZ/G2M V1.3, RZ/G2M V3.0 RZ/G2N</p> <ul data-bbox="464 349 836 371" style="list-style-type: none"> <li data-bbox="464 349 836 371">• (H.264 1920 × 1080p × 120 fps) <p data-bbox="464 394 580 416">RZ/G2E</p> <p data-bbox="464 439 660 461">If iVDP1C is used:</p> <ul data-bbox="464 472 820 495" style="list-style-type: none"> <li data-bbox="464 472 820 495">• (H.264 1920 × 1080p × 30 fps) <p data-bbox="464 506 820 528">Otherwise (If iVDP1C isn't used):</p> <ul data-bbox="464 551 820 573" style="list-style-type: none"> <li data-bbox="464 551 820 573">• (H.264 1920 × 1080p × 60 fps) <p data-bbox="464 595 1358 618">"1920 × 1080p × 120 fps" can be replaced as "3840 × 2160p × 30 fps" (H.264 only),</p> <p data-bbox="464 629 1198 651">"1920 × 1080p × 60 fps" can be replaced as "1280 × 720p × 120 fps"</p> <p data-bbox="464 663 1182 685">"1920 × 1080p × 30 fps" can be replaced as "1280 × 720p × 60 fps"</p> <p data-bbox="464 707 1374 757">In case of decoding, this indicates the processing performance including both display picture and non-display picture.</p> |
| Maximum bit rate | <p data-bbox="464 779 692 801">One of the followings:</p> <p data-bbox="464 813 1098 846">RZ/G2H RZ/G2M V1.3, RZ/G2M V3.0 RZ/G2N</p> <ul data-bbox="464 857 1129 880" style="list-style-type: none"> <li data-bbox="464 857 1129 880">• H.264 (40 Mbps × 4ch, 80 Mbps × 2ch, or 160 Mbps × 1ch) <p data-bbox="464 902 580 925">RZ/G2E</p> <p data-bbox="464 947 660 969">If iVDP1C is used:</p> <ul data-bbox="464 981 943 1003" style="list-style-type: none"> <li data-bbox="464 981 943 1003">• H.264 (20 Mbps × 2ch, or 40 Mbps × 1ch) <p data-bbox="464 1014 692 1037">If iVDP1C isn't used:</p> <ul data-bbox="464 1059 943 1081" style="list-style-type: none"> <li data-bbox="464 1059 943 1081">• H.264 (40 Mbps × 2ch, or 80 Mbps × 1ch) |

| Item | Description |
|------------------------------------|--|
| Picture size supported in decoding | <p>Minimum size: Horizontal 80 pixels x vertical 80 lines</p> <p>Maximum size: Horizontal 3840 pixels x vertical 2160 lines (H.264) (If horizontal size is larger than 1920, the VCPLF only supports a progressive sequence.)</p> <p>RZ/G2E</p> <p>Note: When picture size is 3840 x 2160, frame rate is smaller than 30 fps. For example, if iVDP1C isn't used and codec is H.264/AVC, frame rate is 15 fps.</p> <p>Horizontal size is a multiple of 2 pixels, and vertical size is a multiple of 2 lines. Additionally, 1080 x 1920, 1088 x 1920, and 2160 x 3840 (H.264 only) are supported.</p> |
| Picture size supported in encoding | <p>Minimum size: Horizontal 80 pixels x vertical 80 lines</p> <p>Maximum size: Horizontal 3840 pixels x vertical 2160 lines (H.264) Horizontal size is a multiple of 8 pixels, and vertical size is a multiple of 8 lines. Additionally, 320 x 180, 854 x 480 (WVGA+), 1080 x 1920, 1088 x 1920, and 2160 x 3840 (H.264 only) are supported.</p> |
| Color format | <p>YCbCr 4:2:0 planar (Y, Cb, and Cr plane) and semi-planar (Y and C plane) are supported as an input picture in encoding.</p> <p>YCbCr 4:2:0 semi-planar is supported as an output picture in decoding.</p> <p>Cb/Cr and Cr/Cb interleaving order is supported in YCbCr 4:2:0 semi-planar.</p> |
| Bit stream format | <p>Video elementary stream as shown in Table 32.4.</p> <p>Function to add or remove the emulation prevention byte in the H.264 bit stream is supported.</p> |
| Picture structure | Frame structure and field structure |

Table 32.4 Data Handled by the VCPLF

| Standard | Encoding Range | Decoding Range |
|----------|----------------------------------|---|
| H.264 | Slice header and subsequent data | Slice header (other than the first slice header in picture) and subsequent data |

(3) Summary of Maximum Pixel Rate

The following combinations of maximum pixel rate shall be applied to RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E.

RZ/G2H | **RZ/G2M V1.3, RZ/G2M V3.0** | **RZ/G2N**

If iVDP1C is used, the maximum pixel rate is one of following items:

- (H.264/H.265 1920 x 1080p x 120 fps)

Otherwise (if iVDP1C isn't used), the maximum pixel rate is one of following items:

- (H.265 1920 x 1080p x 120 fps) + (H.264 1920 x 1080p x 120 fps)

RZ/G2E

If iVDP1C is used, the maximum pixel rate is one of following items:

- (H.264/H.265 1920 × 1080p × 30 fps)

Otherwise (if iVDP1C isn't used), the maximum pixel rate is one of following items:

- (H.264/H.265 1920 × 1080p × 60 fps)

Notes:

“1920 × 1080p × 120 fps” can be replaced as “3840 × 2160p × 30 fps” (H.264/H.265).

“1920 × 1080p × 60 fps” can be replaced as “1280 × 720p × 120 fps”.

“1920 × 1080p × 30 fps” can be replaced as “1280 × 720p × 60 fps”.

32.1.2 Connected Module

Table 32.5 shows modules connected to the VCP4.

Table 32.5 Connected Module

| Module | Connected Module | Description |
|--------|------------------|--|
| VDPB | FCPCS | Reference data read cache, lossless data compression and decompression For RZ/G2M V1.3, FCPCS is shared between VDPB and VCPLF. For RZ/G2H, RZ/G2M V3.0, RZ/G2N and RZ/G2E, FCPCS is shared among VDPB, VCPLF, and iVDP1C. |
| VCPLF | FCPCS | Reference data read cache, lossless data compression and decompression For RZ/G2M V1.3, FCPCS is shared between VDPB and VCPLF. For RZ/G2H, RZ/G2M V3.0, RZ/G2N and RZ/G2E, FCPCS is shared among VDPB, VCPLF, and iVDP1C. |
| | FCPCI | RZ/G2M V1.3 only has this module. Reference data read cache, lossless data compression and decompression FCPCI is dedicated for iVDP1C or VCPLF. |

32.2 Usage Notes

32.2.1 Limitations on Software Reset, Module Standby and power-shutoff

Before executing following operations for VCP4, confirm that VCP4 is not encoding or decoding a picture. If VCP4 is encoding or decoding a picture, wait completion of encoding or decoding.

- Execute Software Reset by CPG
- Enter Module Standby (stop clock supply) by CPG
- Enter power-shutoff status (stop power supply) by SYSC

32.2.2 Note on input bitstream for decoding

In decoding process of VCP4, input bitstream not conformant to video codec standard might cause hang of VCP4.

33. Video Signal Processor (VSP2)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

33.1 Overview

The VSP2 is the successor IP of Renesas' VSP-IP series which supports image processing such as super resolution, sharpness, up/down scaling, 1D look-up tables, 3D look-up tables, histogram, image blending, rotation/flipping, interface to Display Unit (DU).

33.1.1 Features

This LSI incorporates different types of VSP2 modules (VSPI, VSPBS, VSPBC, VSPBD, VSPB, VSPD and VSPDL) as image processing systems. Table 33.1 shows the number of VSP for each product.

The VSPI supports image processing by memory to memory such as super resolution, sharpness, up/down scaling, look-up tables, histogram and rotation/flipping.

The VSPBC supports image processing by memory to memory such as look-up tables, 1D-histogram and image blending. [RZ/G2H].

The VSPBD supports image processing by memory to memory such as image blending [RZ/G2H].

The VSPB supports image processing by memory to memory such as look-up tables, 1D-histogram, image blending and vertical flipping, VSPB is composed by unifying VSPBC function with VSPBD.

The VSPD supports image processing such as image blending, interface to and output image data to Display Unit (DU) with or without writing back the image data to memory. The VSPD supports image processing by memory to memory also, but the VSPD cannot output image data to Display Unit (DU) in that case.

The VSPDL supports image processing such as image blending, and output image data to Display Unit (DU) with or without writing back the image data to memory. The VSPDL supports two interfaces with Display Unit (DU). The VSPDL supports image processing by memory to memory also, but the VSPDL cannot output image data to Display Unit (DU) in that case. [RZ/G2H, RZ/G2N]

Table 33.1 VSP2 configuration

| Module Name | Number of channels | | | |
|-------------|--------------------|----------------------------|--------|--------|
| | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| VSPI | 2 | 1 | 1 | 1 |
| VSPBC | 1 | 0 | 0 | 0 |
| VSPBD | 1 | 0 | 0 | 0 |
| VSPB (*) | 0 | 1 | 1 | 1 |
| VSPBS | 0 | 0 | 0 | 0 |
| VSPD | 1 | 3 | 1 | 2 |
| VSPDL | 1 | 0 | 1 | 0 |

Note: * INTC, module standby for VSPB in RZ/G2M V1.3, RZ/G2M V3.0 are same with VSPBD in RZ/G2H.
Each VSPs features are shown in from section 33.1.1.1 to section 33.1.1.7.

33.1.1.1 Features of VSPI

- Supports various data formats and conversion
 - Supports YCbCr444/422/420, RGB, α RGB, α plane.
 - Color space conversion and changes to the number of colors by dithering
 - Color keying
 - Supports combination between pixel alpha and global alpha.
 - Supports generating pre multiplied alpha.
- Video processing
 - Supporting 4K (3840 pixels \times 2160 lines) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 - Supporting Full HD (1920 pixels \times 1080 lines) [RZ/G2E]
 - Up and down scaling with arbitrary scaling ratio
 - Super resolution processing
 - Sharpness filter
 - 1D look up table
 - 3D look up table
 - Supports Hue, brightness, and saturation format for look up table or/and histogram.
 - 1D histogram
 - 2D histogram
 - Image rotation with unit of 90-degree, vertical and horizontal flipping
- Performance of VSPI (*¹)
 - 500 Mpix/s process rate (processing rate (*²)) \times 2 sets [RZ/G2H]
 - 500 Mpix/s process rate (processing rate (*²)) \times 1 set [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 - 125 Mpix/s process rate (processing rate (*²)) \times 1 set [RZ/G2E]

Notes: 1. This performance cannot be realized in some condition. Detail performance is shown in section 33.4.6.1.
2. When scale-down is applied to input image, processing rate is input rate. When scale-up is applied to input image, processing rate is output rate. When no-scaling is applied, processing rate is input rate.

33.1.1.2 Features of VSPBD [RZ/G2H]

- Supports various data formats and conversion
 - Supports YCbCr444/422/420, RGB, α RGB, α plane.
 - Color space conversion and changes to the number of colors by dithering
 - Color keying
 - Supports combination between pixel alpha and global alpha.
 - Supports generating pre multiplied alpha.
- Video processing.
 - Supporting 4K (3840 pixels \times 2160 lines)
 - Blending of five picture layers and raster operations (ROPs)
 - Vertical flipping
- Performance of VSPBD (*)
 - 500 Mpix/s process rate (output rate) \times 1 set

Note: * This performance cannot be realized in some condition. Detail performance is shown in section 33.4.7.2.

33.1.1.3 Features of VSPBC [RZ/G2H]

- Supports various data formats and conversion
 - Supports YCbCr444/422/420, RGB, α RGB, α plane.
 - Color space conversion and changes to the number of colors by dithering
 - Color keying
 - Supports combination between pixel alpha and global alpha.
 - Supports generating pre multiplied alpha.
- Video processing
 - Supporting 4K (3840 pixels \times 2160 lines)
 - Blending of five picture layers and raster operations (ROPs)
 - 1D look up table
 - 3D look up table
 - 1D histogram
 - Vertical flipping
- Performance of VSPBC (*)
 - 500 Mpix/s process rate (output rate) \times 1 set

Note: * This performance cannot be realized in some condition. Detail performance is shown in section 33.4.7.2.

33.1.1.4 Features of VSPB [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

- Supports various data formats and conversion
 - Supports YCbCr444/422/420, RGB, α RGB, α plane.
 - Color space conversion and changes to the number of colors by dithering
 - Color keying
 - Supports combination between pixel alpha and global alpha.
 - Supports generating pre multiplied alpha.
- Video processing
 - Supporting 4K (3840 pixels \times 2160 lines) [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 - Supporting Full HD (1920 pixels \times 1080 lines) [RZ/G2E]
 - Blending of five picture layers and raster operations (ROPs)
 - 1D look up table
 - 3D look up table
 - 1D histogram
 - Vertical flipping
- Performance of VSPB (*)
 - 500 Mpix/s process rate (output rate) \times 1 set [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 - 125 Mpix/s process rate (output rate) \times 1 set [RZ/G2E]

Note: * This performance cannot be realized in some condition. Detail performance is shown in section 33.4.7.2.

33.1.1.5 Features of VSPD

- Supports various data formats and conversion
 - Supports YCbCr444/422/420, RGB, α RGB, α plane.
 - Color space conversion and changes to the number of colors by dithering
 - Color keying
 - Supports combination between pixel alpha and global alpha.
 - Supports generating pre multiplied alpha.
- Video processing
 - Blending of five picture layers and raster operations (ROPs)
 - Vertical flipping in case of output to memory.
- Direct connection to display module
 - Supporting 4096 pixels in horizontal direction [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 - Supporting 2048 pixels in horizontal direction [RZ/G2E]
 - Writing back image data which is transferred to Display Unit (DU) to memory.

33.1.1.6 Features of VSPDL [RZ/G2H, RZ/G2N]

- Supports various data formats and conversion
 - Supports YCbCr444/422/420, RGB, α RGB, α plane.
 - Color space conversion and changes to the number of colors by dithering
 - Color keying
 - Supports combination between pixel alpha and global alpha.
 - Supports generating pre multiplied alpha.
- Video processing
 - Blending of five picture layers and raster operations (ROPs)
 - Vertical flipping in case of output to memory.
- Direct connection to display module
 - Supporting two display output interfaces
 - Supporting 2048 pixels in horizontal direction.
 - Writing back image data which is transferred to Display Unit (DU) to memory.

33.1.1.7 Common restrictions

When the data output from the VSP2 is written back to the same memory area where the input data for the VSP2 has been read, the VSP2 has the following restrictions.

1. The access order and format on the frame memory are the same between the input pixels and output pixels.
2. Specifying a larger output image size than the input image size, either vertically or horizontally or both vertically and horizontally, is prohibited.
3. When the YCbCr4:2:0 format is input, operation between color components is prohibited.

These restrictions are summarized in Table 33.2. Refer to the descriptions of the registers related to each restriction. In the table, RPF_m indicates the RPF_n that inputs the master layer, and WPF_{wb} indicates the WPF_n that writes back the output image to the source image area for the master layer.

Table 33.2 Restrictions on Use when Output Data is Written Back to Input Data Area

| No. | Restriction | Related Registers |
|----------------|--|--|
| Restriction 1 | The RPF _m input format and the WPF _{wb} output format should be the same. | VI6_RPFn_INFMT.RDFMT VI6_RPFn_INFMT.VIR |
| | The RPF _m source image storing address and the WPF _{wb} destination address should be the same. | VI6_RPFn_SRCM_ADDR_* |
| | The RPF _m source picture memory stride and the WPF _{wb} destination memory stride should be the same. | VI6_RPFn_SRCM_PSTRIDE VI6_RPFn_SRCM_ASTRIDE |
| | The RPF _m and WPF _{wb} data swapping settings should be the same. | VI6_WPFn_DSWAP |
| | The image rotation and flipping function cannot be used if WPF _{wb} is WPF0. | VI6_WPFn_OUTFMT.ROT |
| | Super resolution processing with double scale-up by the SRU is prohibited (super resolution processing of the same size is allowed). | VI6_SRU_CTRL0 |
| Restriction 2 | Scale-up and color filling by UDS is prohibited. | VI6_UDS_SCALE VI6_UDS_CLIP_SIZE |
| | The value of scaling filter horizontal and vertical phase registers should be zero in case of using UDS. | VI6_UDS_HPHASE VI6_UDS_VPHASE |
| | The RPF _m basic read size and extended read size should be the same. | VI6_RPFn_SRC_BSIZE VI6_RPFn_SRC_ESIZE |
| | Color space conversion is prohibited in RPF _m and WPF _{wb} . | VI6_RPFn_INFMT.CSC VI6_WPFn_OUTFMT.CSC |
| Restriction 3* | Use of the CLU is prohibited. Make sure that the CLU is not used in DPR routing. | VI6_DPR_CLU_ROUTE.RT |
| | NOP should be specified for IROP operation. | VI6_RPFn_ALPH_SEL.IROP |
| | Color keying is prohibited. | VI6_RPFn_CKEY_CTRL.CV |

Note: * When the input format is not YCbCr4:2:0, restriction 3 is not applied.

33.1.2 Block diagram

33.1.2.1 Block diagram of each VSP

Figure 33.1 to Figure 33.6 shows the configuration of VSPI, VSPBC, VSPBD, VSPB, VSPBS, VSPD and VSPDL sub modules.

All the registers and functions which doesn't appears in the each VSP2 unit's block diagram should be treated as un-implemented functions in each VSP2 unit. Do not set any registers related to the un-implemented function without any statement.

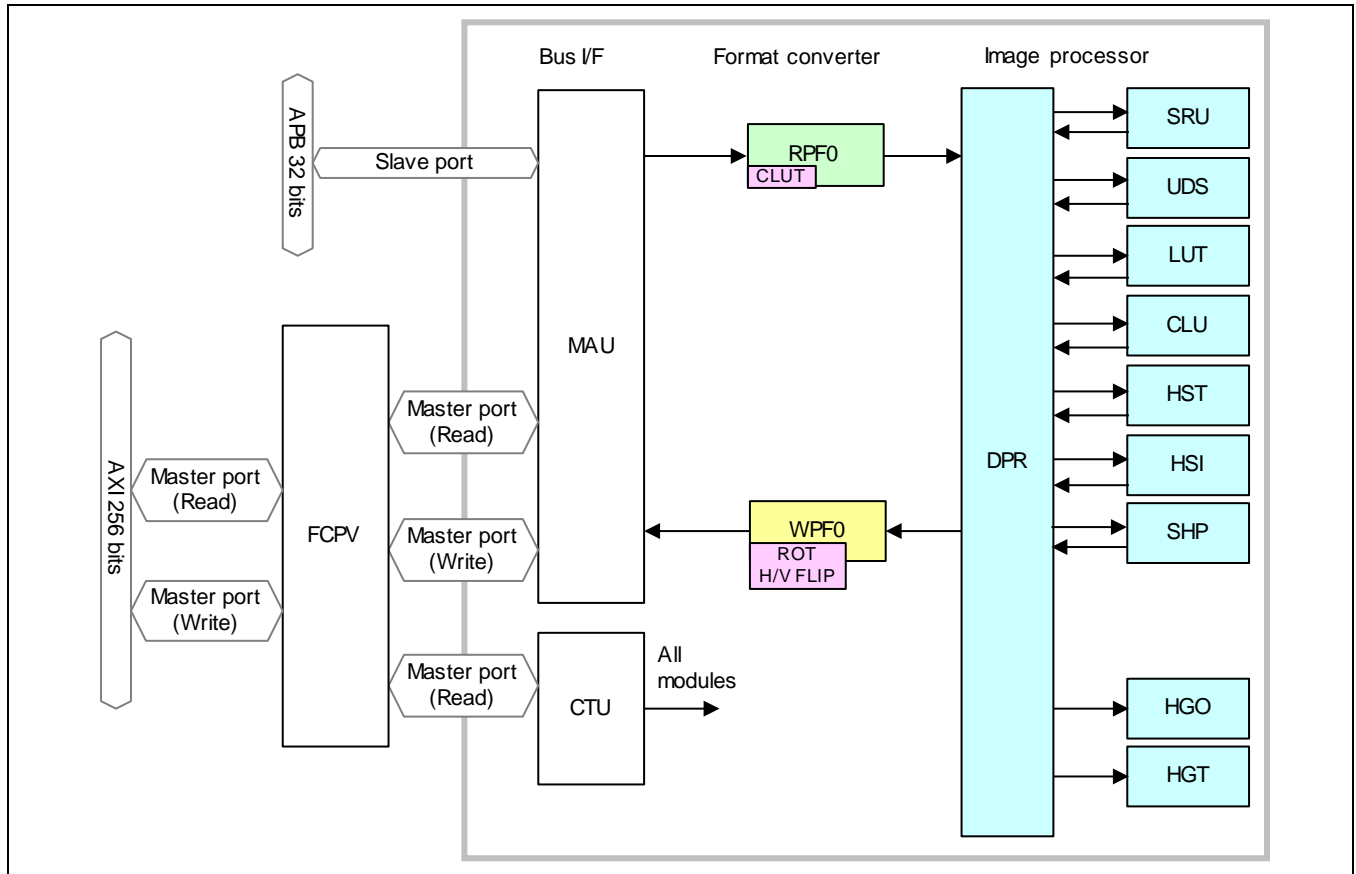


Figure 33.1 VSPI Module Configuration

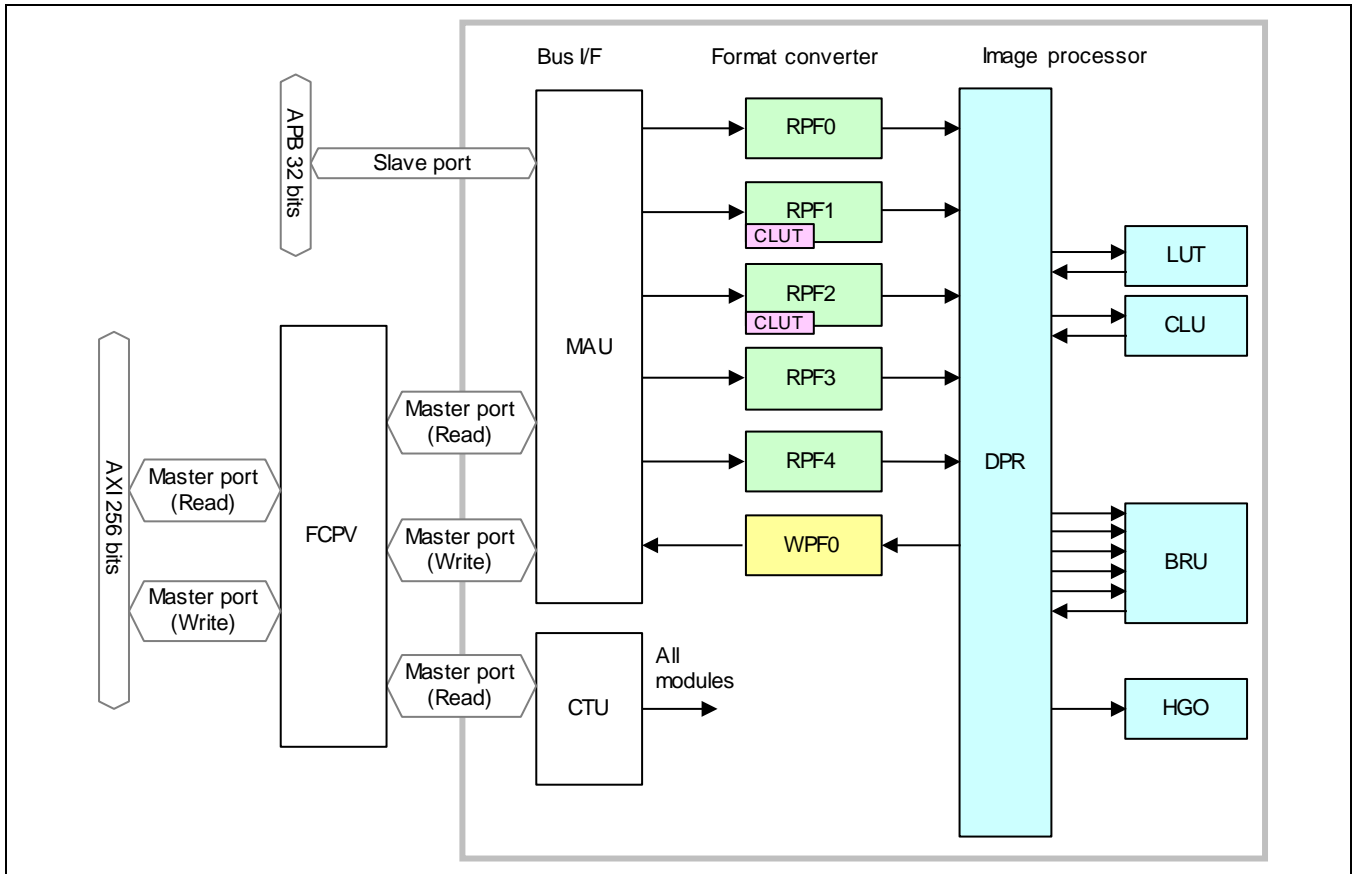


Figure 33.2 VSPBC Module Configuration [RZ/G2H]

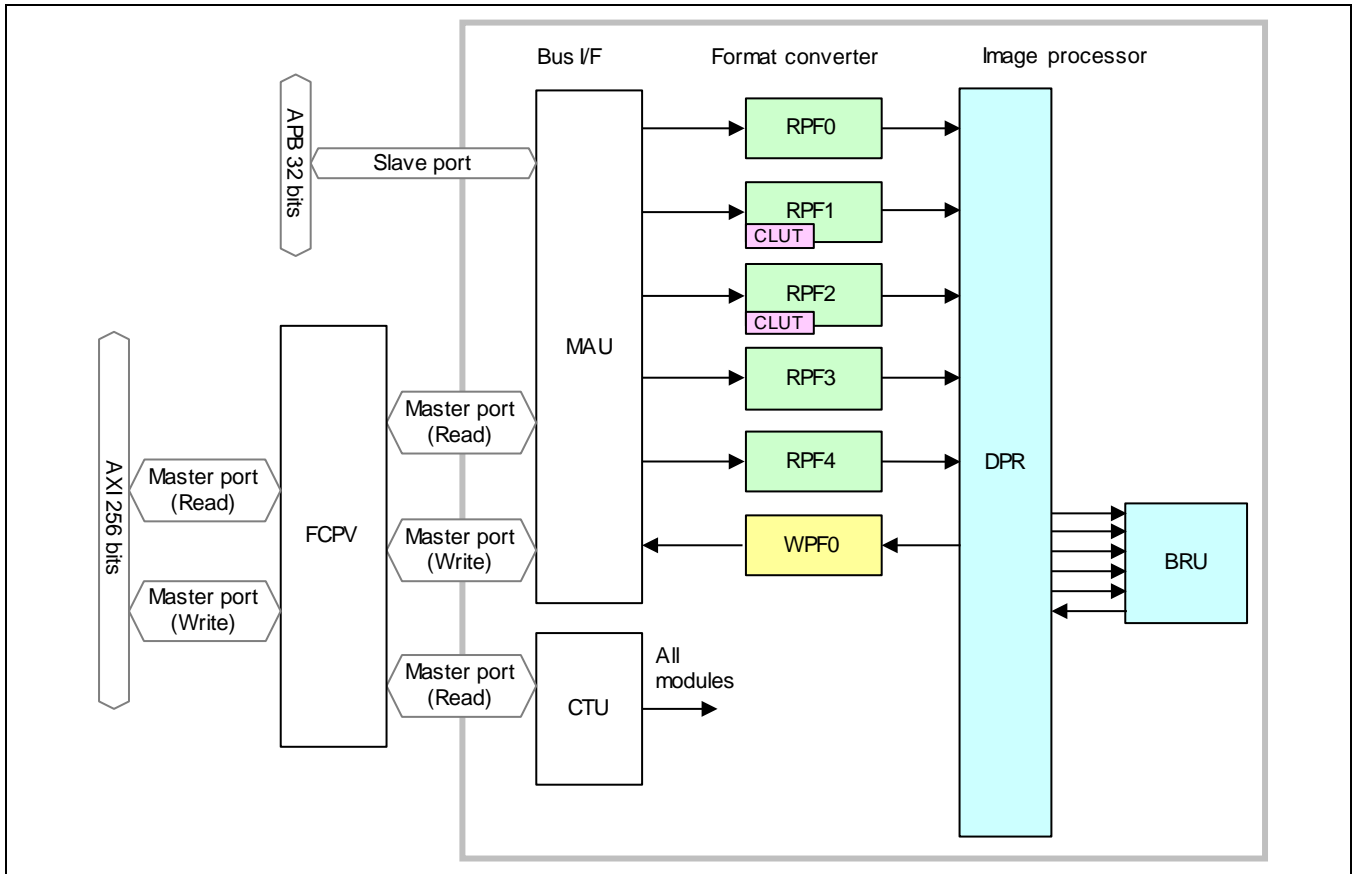


Figure 33.3 VSPBD Module Configuration [RZ/G2H]

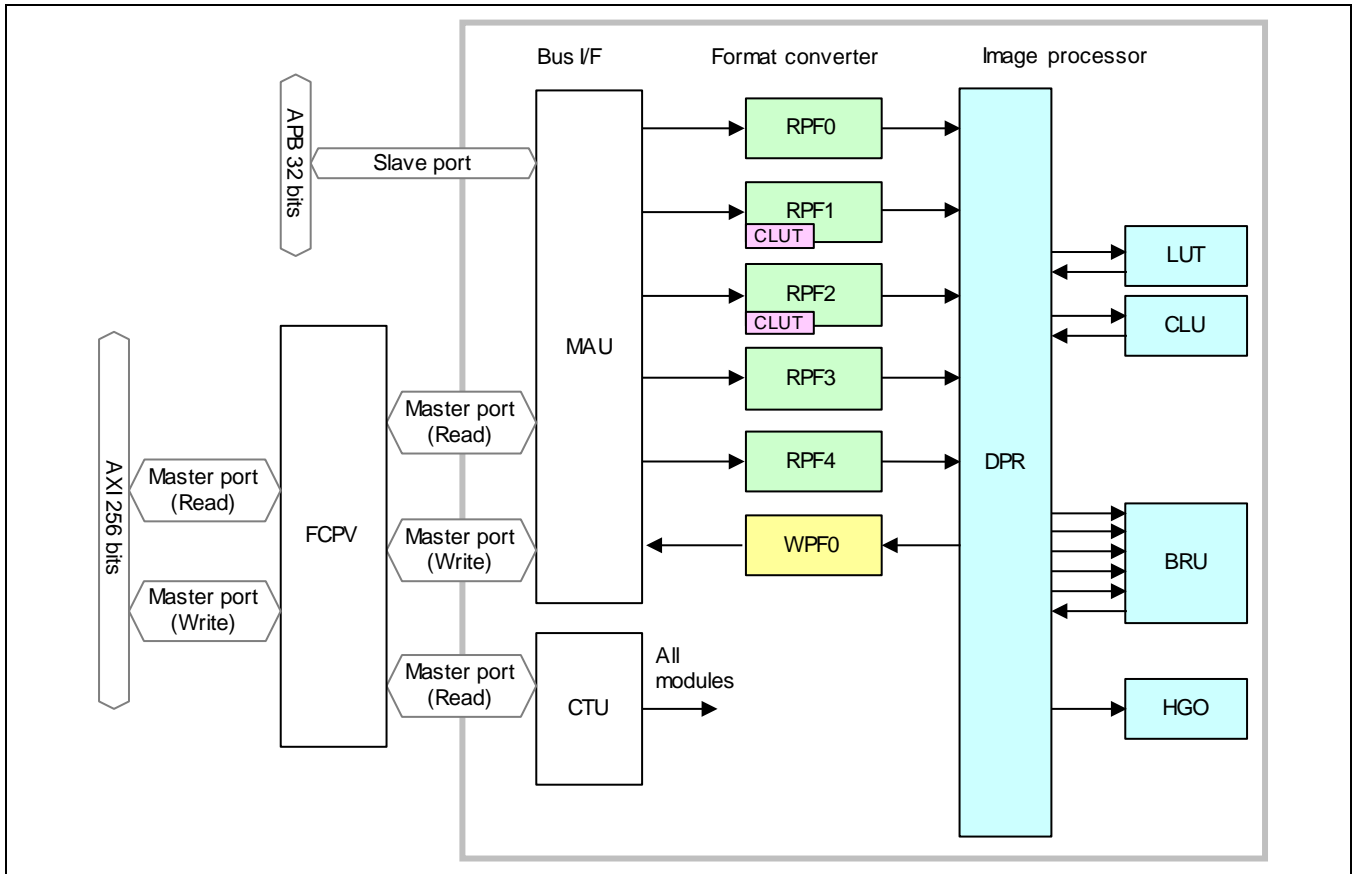


Figure 33.4 VSPB Module Configuration [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

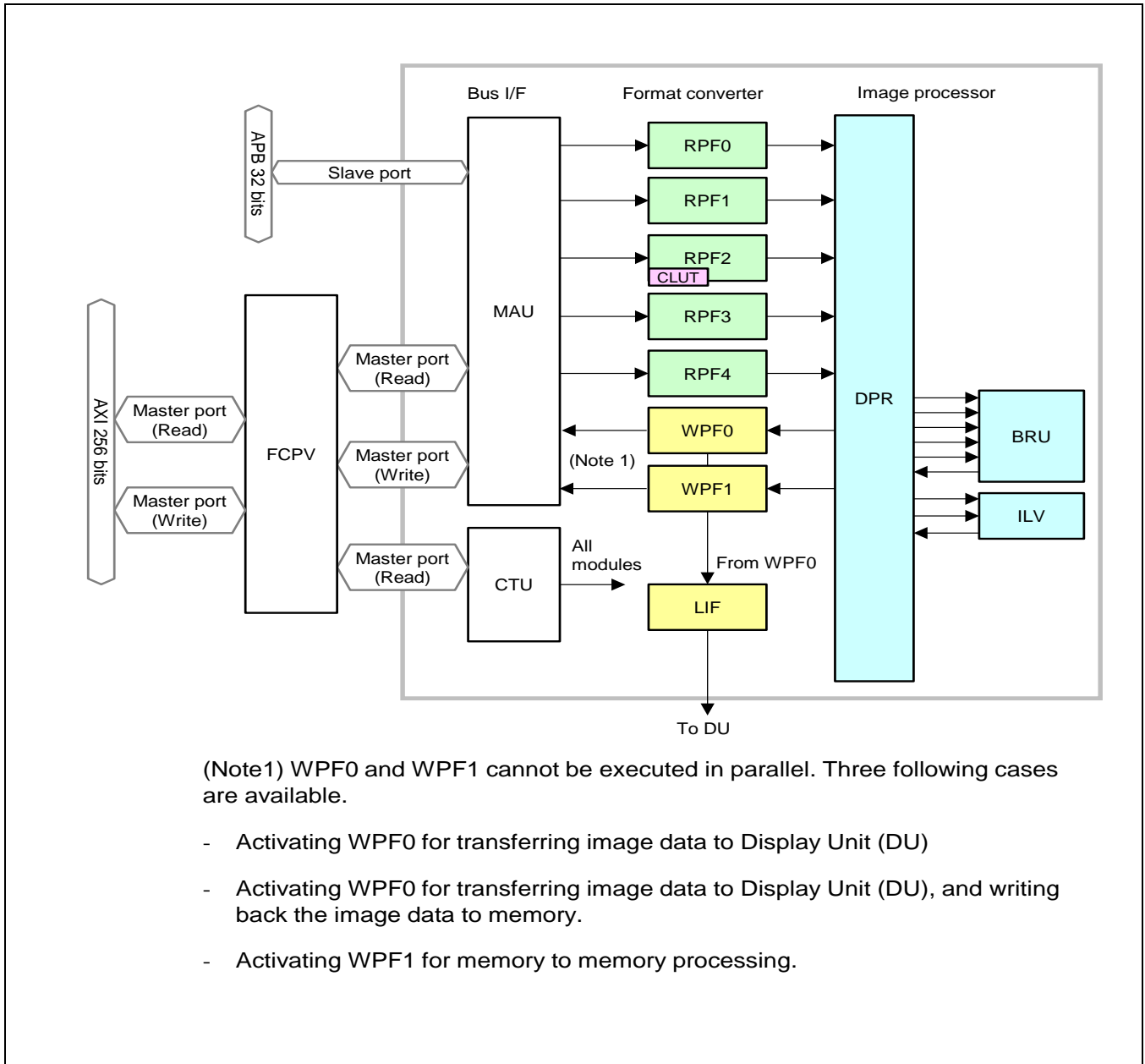


Figure 33.5 VSPD Module Configuration

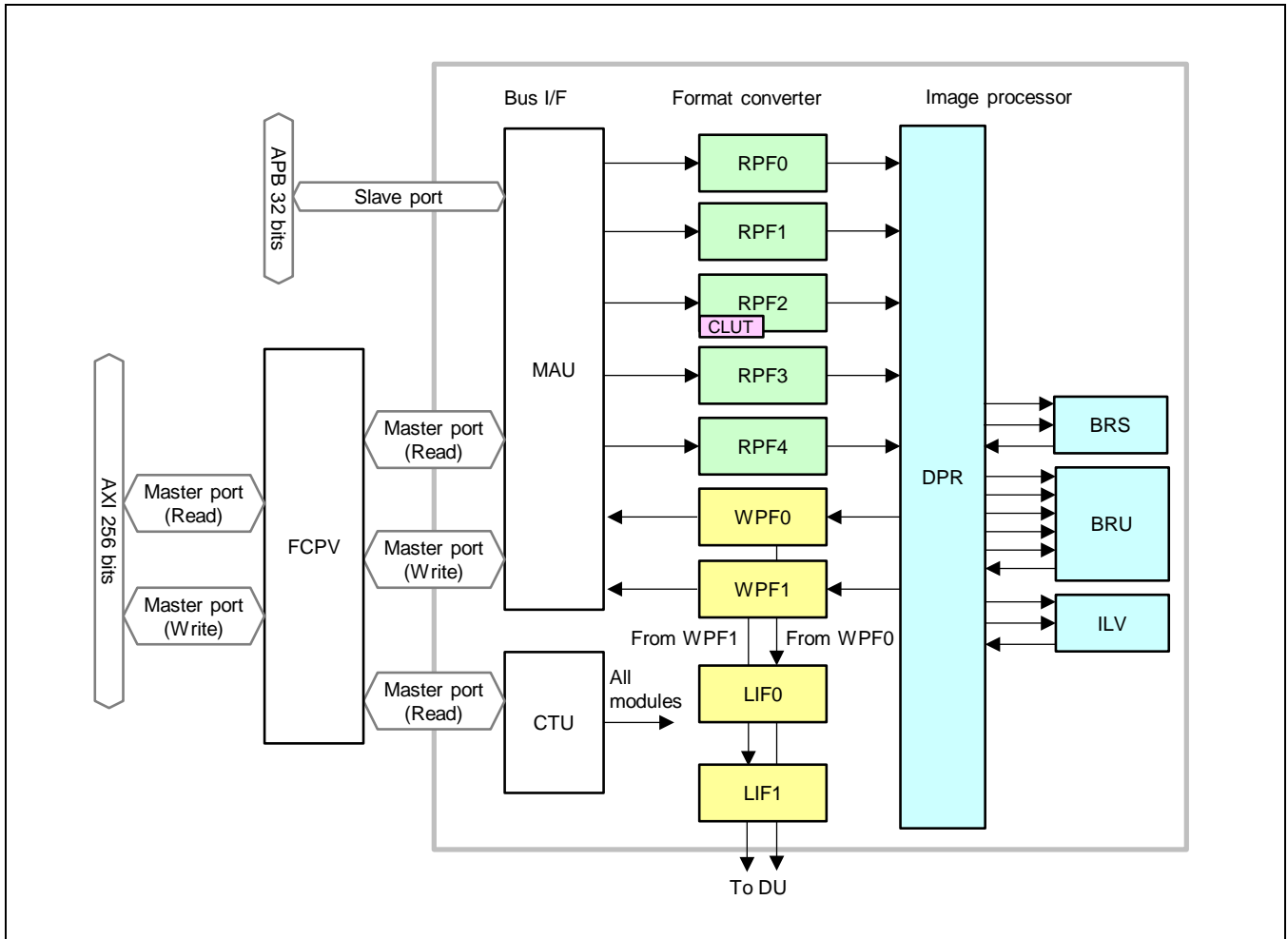


Figure 33.6 VSPDL Module Configuration [RZ/G2H, RZ/G2N]

Note: When neither WPF0 nor WPF1 is used to transfer image data to Display Unit (DU), WPF0 and/or WPF1 can be used as memory to memory processing.

Two following cases are available

- Activating WPF0 for transferring image data to Display Unit (DU) with or without writing back the image data to memory. Activating WPF1 for transferring image data to Display Unit (DU) with or without writing back the image data to memory.
- Activating WPF0 for memory to memory processing. Activating WPF1 for memory to memory processing.

The following gives an overview of each block function shown in Figure 33.1 to Figure 33.6.

The operation of these functions is determined by the register setting explained in section 33.2. Refer to section 33.2 for the relationship between the register setting and the functional behavior.

33.1.2.2 Memory Access Unit (MAU)

The VSP2 applies processing to the image data stored in the external memory and writes the resultant data back to the external memory. The data transfer between the external memory and VSP2 necessary for this operation is done by the MAU, which works as the bus master, according to the register settings. The MAU executes this data transfer between the external memory and VSP2.

33.1.2.3 Command Transfer Unit (CTU)

The VSP2 can directly read register parameters for image processing by display lists stored in external memory. The CTU module is a bus interface and controls display lists when the CTU reads display lists as a bus master.

33.1.2.4 Read Pixel Formatter (RPF)

The RPF reads image data from the external memory through the MAU, unpacks data according to the specified format, converts the color space, converts the number of colors, executes color keying, ROP operation, Multiply-alpha and OSD processing, and outputs the resultant data to the DPR. The RPF has an input format unpacking unit, an index color expanding unit (OSD-CLUT), a 1-bit mask generator, a raster operation unit (ROP unit), a color keying unit, a color space converter and multiply-alpha. Figure 33.7 shows the processing flow in the RPF.

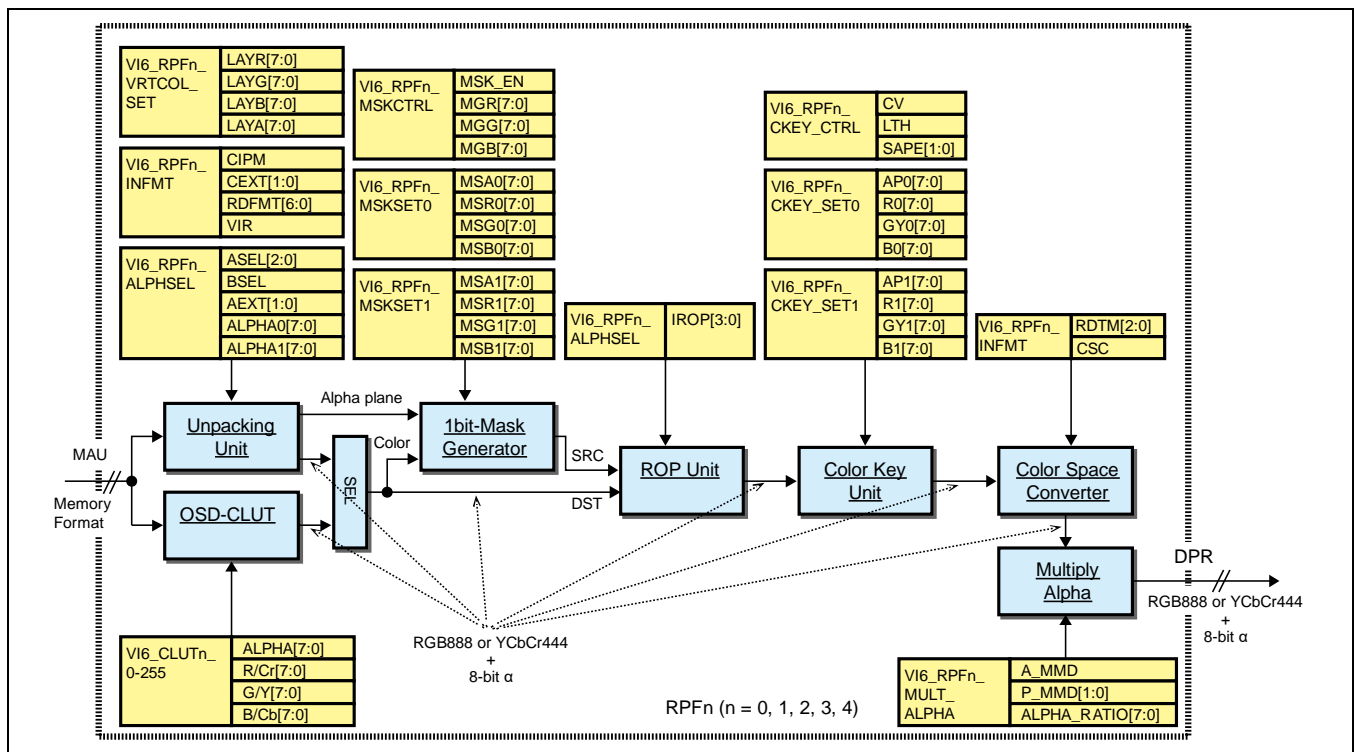


Figure 33.7 RPF Processing Flow

The input format unpacking unit and index color expanding unit (OSD-CLUT) expand the image data input from the MAU into the image format for internal processing (YCbCr444 or RGB888), and the 1-bit mask generator generates a 1-bit image mask from the image data expanded through the unpacking unit and OSD-CLUT. Alternately, a 1-bit image mask can be generated from the alpha plane that is different from the picture plane read through the MAU.

The raster operation unit (ROP unit) executes raster operation between the data from the 1-bit mask generator and the image data expanded from the input format, and the color keying unit applies color replacement and specifies the transparent color for the image data input from the ROP unit. The color space converter converts the color space (RGB-YCbCr) of the image data input from the color keying unit as necessary. The multiply-alpha unit multiplies pixel-alpha by specified alpha or/and multiplies image data by pixel-alpha or/and specified alpha.

The VSP2 provides maximum five RPF modules (RPF0 to RPF4). The implementation of OSD-CLUT depends on the configuration of each LSI.

33.1.2.5 Data Path Router (DPR)

The DPR controls the data paths among RPFs, function modules, and WPFs. The DPR selects one of the images input from RPFs, outputs it to a function module (BRU, BRS, SRU, UDS, LUT, CLU, HST, HSI, HGO, HGT, SHP or ILV), and selects one of WPFs as the destination where the image data processed in the function module will be output. Before output to the WPF, the output from each function module can be input to another function module, which enables multiple image processing functions to be executed continuously without involving the external memory.

33.1.2.6 Super Resolution Unit (SRU)

The SRU is a module connected to the DPR, which executes the super resolution processing.

33.1.2.7 Up Down Scalar (UDS)

The UDS is a module connected to the DPR, which upscales or downscales the image size. It can also upscale or downscale the alpha value.

33.1.2.8 Look Up Table (LUT)

This is a 1D-LUT that converts each of three color components by using a lookup table. The LUT is connected to the DPR and can be used for gamma correction, negative-positive conversion, posterization, and binarization through desired tone curve settings.

33.1.2.9 Blend ROP Unit (BRU)

The BRU is a module connected to the DPR, which executes the image blending processing and ROP operation. The BRU has five blend/ROP operation units (blend/ROP unit m, m = A to E), an ROP operation unit, a blend/ROP input switch (SEL) for selecting the input to these operation units, and a divider for normalization (div unit).

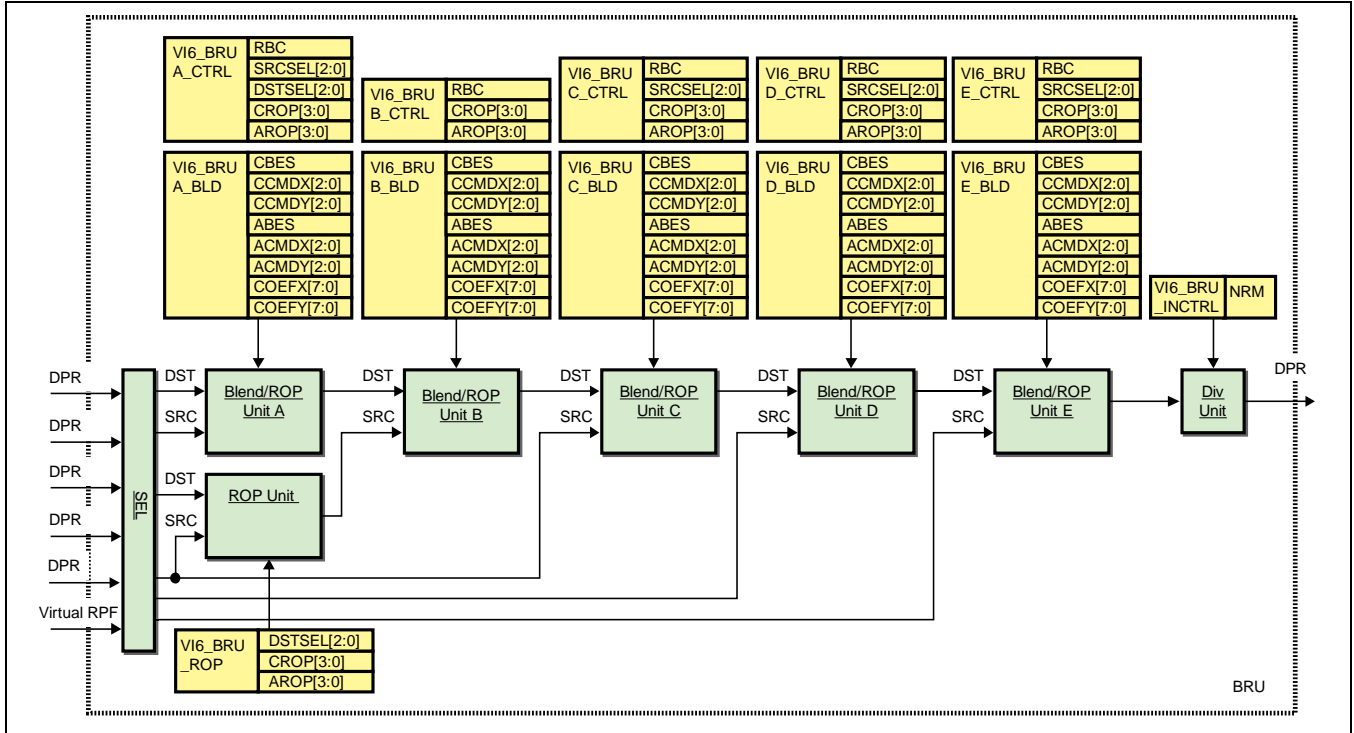


Figure 33.8 BRU Processing Flow

Each of the five blend-ROP operation units (blend/ROP unit m) receives the output from the SEL, blend/ROP Unit m, or ROP unit, and executes blending or raster operation (ROP) of images. The ROP unit receives the output from the SEL and executes 2-input raster operation (ROP2). By combining the ROP operation in the blend/ROP unit m and ROP2 operation in the ROP unit, 256-type 3-input raster operation (ROP3) can be implemented.

The divider for normalization (div unit) divides the pixel value by the α value.

33.1.2.10 Blend ROP Sub Unit (BRS)

The BRS is a module connected to the DPR, which executes the image blending processing and ROP operation. The BRS has two blend/ROP operation units (blend/ROP unit m, m = A to B), a blend/ROP input switch (SEL) for selecting the input to these operation units, and a divider for normalization (div unit).

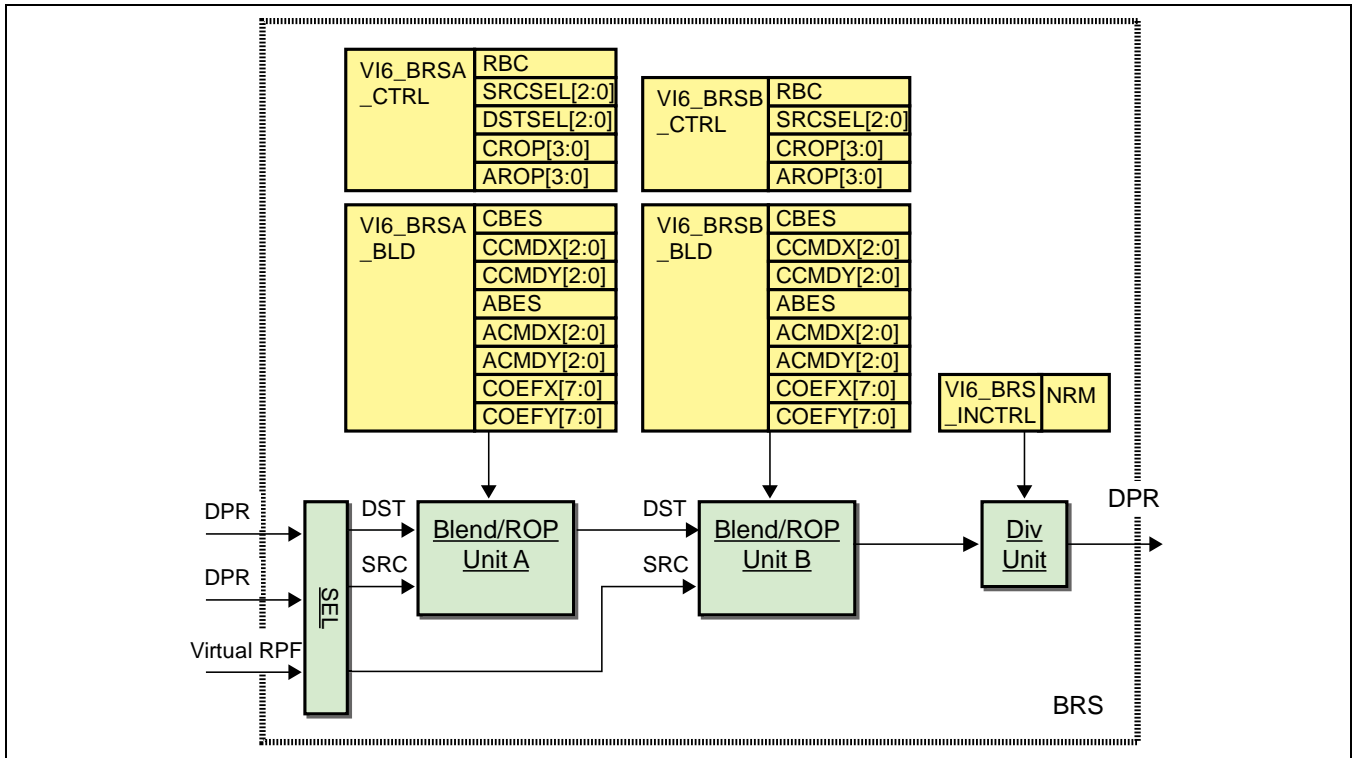


Figure 33.9 BRS Processing Flow

Each of the two blend-ROP operation units (blend/ROP unit m) receives the output from the SEL or blend/ROP Unit m, and executes blending or raster operation (ROP) of images.

The divider for normalization (div unit) divides the pixel value by the α value.

33.1.2.11 Write Pixel Formatter (WPF)

The WPF is an output module that receives 32bits image data (YCbCr444 or RGB888 + 8bit- α) from the DPR, converts the color space, number of colors, and format of the data, and outputs the results of VSP2 image processing to external memory through the MAU. The WPF is mainly configured from a color space converter and an output format converter (the packing unit).

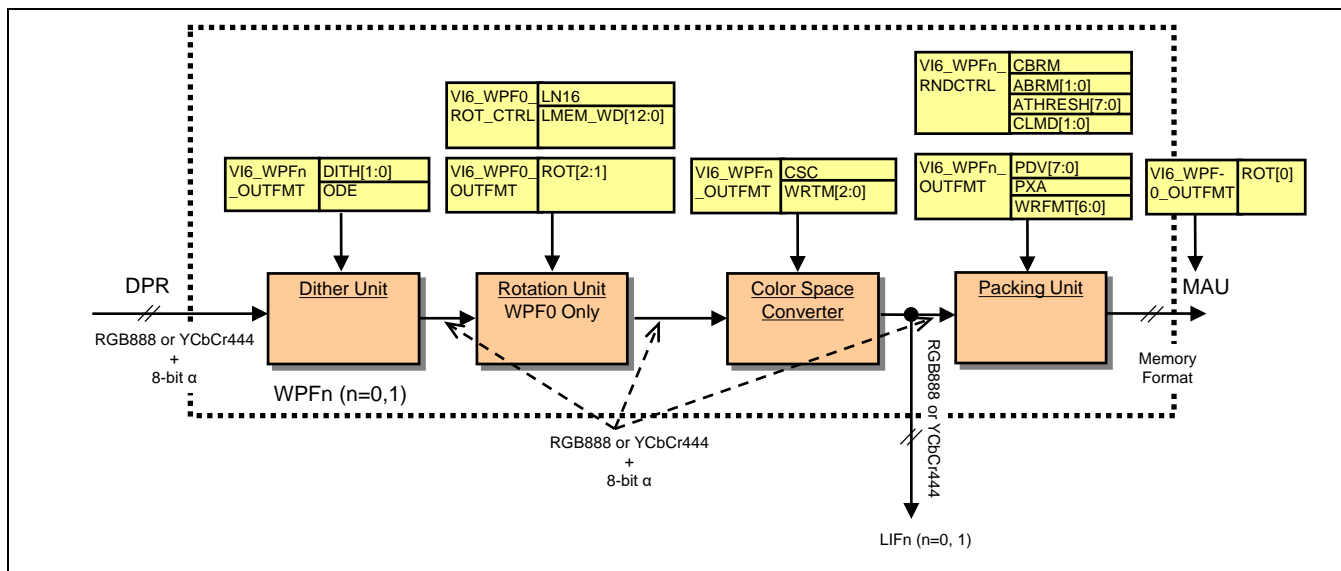


Figure 33.10 WPF Processing Flow

The color space converter converts the color space between RGB and YCbCr, and the packing unit converts the format into the picture plane storing format. The VSP2 provides two WPFs (WPF0 to WPF1). The WPF has the output to LIF module after the color space conversion for transferring the video data to the display module. Image data to LIF is 24bits (YCbCr444 or RGB888) format, and its' color space is after color space conversion.

33.1.2.12 Cubic Look Up table (CLU)

This is a three-dimensional LUT (3D-LUT) that converts the input three-color-component data into desired three color components by using a lookup table. The 1D-LUT can only control each of three color components separately, but the 3D-LUT can convert a specified color into a different color. The CLU is connected to the DPR and can be used for specific color adjustment, such as correction to memorable color.

33.1.2.13 Hue Saturation value Transform (HST)

The HST is a module connected to the DPR, which converts the RGB color space into the HSV color space. The HSV color space can represent a color with the hue, saturation, and value (brightness), and it makes color control through the LUT easier than in the RGB color space.

33.1.2.14 Hue Saturation value Inverse transform (HSI)

The HSI is a module connected to the DPR, which converts the HSV color space into the RGB color space. It executes conversion in the direction opposite to that of the HST.

33.1.2.15 Histogram Generator -One dimension (HGO)

The HGO is a module connected to the DPR and generates the one-dimensional histogram for the dynamic gamma correction. There is no output port of the image data.

33.1.2.16 Histogram Generator-Two dimension (HGT)

The HGT is a module connected to the DPR and generates the two-dimensional histogram for the dynamic color correction. There is no output port of the image data.

33.1.2.17 Display Unit InterFace (LIF)

The LIF module is used for transferring image data to the display module. The input port of the LIF_n (n = 0, 1) module is connected to WPF_n (n = 0, 1), and the output port of the LIF module is connected to DU (Display Unit). Data flow in LIF is shown in Figure 33.56.

33.1.2.18 SHarPness (SHP)

The SHP is a module connected to the DPR, which sharpen or un-sharpen the image.

33.1.2.19 InterLeaVer (ILV)

The ILV module is used for converting image data to the various 3D display formats. The ILV module has 2 input ports and 1 output port. Each port is connected to the DPR module.

33.1.2.20 Detail Function

Table 33.3 Detail Function of VSP2

| Image Data Transfer Function | | | |
|--|----------------|-------------------------------------|---|
| Bus interface | Protocol | AXI 256 bits through FCPV | |
| | Data alignment | Conversion method | Byte, Word, LW, or LLW data swapping |
| | | Channel | Data alignment can be specified separately for each input/output channel |
| Image memory | Input | Address setting | 1-byte units |
| | Output | Address setting | 1-byte units |
| | | Memory area | Images can be written to the same memory area where the master layer is stored. Note the restrictions shown in Table 33.2. |
| Tile transfer mode | | Supported by RPF0 to RPF4 | |
| YCbCr memory storage format | | Interleaved, planar, or semi-planar | |
| Display list/Extended Display List Transfer Function | | | |
| Bus interface | Protocol | AXI 256 bits through FCPV | |
| | Data alignment | Conversion method | Byte, Word, LW data swapping |
| | | | Data alignment can be specified separately for display list and each entry of extended display list |
| Data memory | Load | Address setting | 8-byte units |
| | Store | — | Not available |

| Read Pixel Formatter (RPF) | | | |
|-----------------------------------|------------------------------|--|--|
| Number of channels | | Five channels (RPF0 to RPF4) | |
| Image format | Input | RGB | RGB888, RGB565, RGB666, αRGB8888, αRGB4444, αRGB1555, α plane (8 bpp, 1 bpp) |
| | | YCbCr | YCbCr4:4:4/YCbCr4:2:2/YCbCr4:2:0 α plane (8 bpp, 1 bpp) |
| | Maximum size | 8190 × 8190 pixels The internal data path modules have separate restrictions on the maximum image size. For details, refer to section 33.4.3 | |
| | Minimum size | 1 × 1 pixel The internal data path modules have separate restrictions on the minimum image size. For details, refer to section 33.4.3. | |
| | Size setting unit | YCbCr420: 2-pixel units both horizontally and vertically. YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. In other formats, the size can be set in 1-pixel units. | |
| Color keying | Color replacement | Compared data | RGB888 or Y |
| | | Replaced data | αRGB8888 or αYCbCr |
| | | Comparison Mode | Matched color mode |
| | | Input source | RRPF0 to RPF4 |
| | Transparent color | Compared data | RGB888 or Y |
| | | Replaced data | α |
| | | Comparison Mode | Matched color mode, Luma threshold mode |
| | | Input source | RPF0 to RPF4 |
| OSD/CLUT | Format | Memory storage format | 8 bpp |
| | | Image format | α RGB32 or α YCbCr32 |
| | Input channel | RPF0 to RPF2 | |
| Raster operation | ROP2 (within input channels) | Operator | 16 types (OpenGL2.0 is supported) |
| | | Sources of operation | ROP2 operation between the 1-bpp α plane and RGB/YCbCr data in RPF0 to RPF4. Note that 1-bpp α is converted to αRGB888 or αYCbCr4:4:4. |
| Color space conversion | RGB to YCbCr | Conversion expression | RGB (0, 255) to BT601 (16, 235/240) RGB (0, 255) to BT709 (16, 235/240) RGB (0, 255) to BT601 (0, 255) RGB (16, 235) to BT709 (16, 235/240) |
| | | Target of conversion | RPF0 to RPF4 |
| | YCbCr to RGB | Conversion expression | BT.601 (16, 235/240) to RGB (0, 255) BT.709 (16, 235/240) to RGB (0, 255) BT.601 (0, 255) to RGB (0, 255) BT.709 (16, 235/240) to RGB (16, 235) |
| | | Target of conversion | RPF0 to RPF4 |

Read Pixel Formatter (RPF)

| | | | |
|-------------------------------|-------------------------------|---|--|
| Changing number of colors | Reducing RGB color depth | Target format | RGB666, RGB565, RGB555, RGB444, or RGB332 |
| | | Target of conversion | RPF0 to RPF4 |
| | Increasing RGB color depth | Padded with 0. Copied from the most significant bits. | |
| | YCbCr444 generation | CbCr copying or CbCr vertical copying and horizontal interpolation. | |
| Multiply-alpha function | Fade-alpha | | RPF0 to RPF4 Available for straight pixel and pre-multiplied pixel. |
| | Generate pre-multiplied alpha | | RPF0 to RPF4 |
| Virtual display | RPF0 to RPF4 | Color format | α RGB8888 or α YCbCr4:4:4 single-color |
| | | Display size | Same as the size of the input channel |
| | Virtual RPF | Color format | α RGB8888 or α YCbCr4:4:4 single-color |
| | | Display size | Maximum: 8190 × 8190 pixels Minimum: 4 × 4 pixels |
| α bit count conversion | Input | Bit increase | Padded with 0. Copied from the most significant bits. |

Write Pixel Formatter (WPF)

| | | | |
|--------------------|--------|-----------------------------|--|
| Number of channels | | Two channels (WPF0 to WPF1) | |
| Image format | Output | RGB | RGB332, RGB444, RGB565, RGB666, RGB888, α RGB8666, α RGB8888, α RGB4444, α RGB1555 |
| | | YCbCr | YCbCr4:4:4, YCbCr4:2:2, YCbCr4:2:0 |
| | | Maximum size | 8190 × 8190 pixels The internal data path modules have separate restrictions on the maximum image size. For details, refer to section 33.4.4 |
| | | Minimum size | 1 × 1 pixel The internal data path modules have separate restrictions on the minimum image size. For details, refer to section 33.4.4 |
| | | Size setting unit | YCbCr420: 2-pixel units both horizontally and vertically. YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. In other formats, the size can be set in 1-pixel units. |

Write Pixel Formatter (WPF)

| | | | |
|-------------------------------|--------------|--------------------------|--|
| Color space conversion | RGB to YCbCr | Conversion expression | RGB (0, 255) to BT.601 (16, 235/240) RGB (0, 255) to BT.709 (16, 235/240) RGB (0, 255) to BT.601 (0, 255) RGB (16, 235) to BT.709 (16, 235/240) |
| | | Target of conversion | WPF0 to WPF1 |
| | YCbCr to RGB | Conversion expression | BT.601 (16, 235/240) to RGB (0, 255) BT.709 (16, 235/240) to RGB (0, 255) BT.601 (0, 255) to RGB (0, 255) BT.709 (16, 235/240) to RGB (16, 235) |
| | | Target of conversion | WPF0 to WPF1 |
| Changing number of colors | Output | Reducing RGB color depth | Dithering, lower-order bit truncation, or rounding |
| | | YCbCr422/420 | CbCr skipping or CbCr vertical skipping and horizontal skipping |
| α bit count conversion | Output | Bit reduction | Truncation, rounding, or comparison with threshold (for 1 bpp) |

Image Compositing

| | | | |
|---------------------------------|--------------------------------|----------------------|---|
| α blending | Input α value selection | RGB | Pixel α , fixed α value, α plane, or 1-bit α converted from the color specified for pixels |
| | | YCbCr | α plane, fixed α value, or 1-bit α converted from the color specified for pixels |
| α blending expression | Plane A: Upper plane | | $x_A + y_B$, $x_A - y_B$ Coefficients x and y should be selected from the following. |
| | Plane B: Lower plane | | Fixed α value, (α for plane A), ($1 - \alpha$ for plane A), (α for plane B), ($1 - \alpha$ for plane B) |
| Output α value selection | RGB | | Fixed α value, x (α for plane A) + y (α for plane B), x (α for plane A) - y (α for plane B) Coefficients x and y should be selected from the following. Fixed α value, (α for plane A), ($1 - \alpha$ for plane A), (α for plane B), ($1 - \alpha$ for plane B) |
| Blending planes | Number of planes | | Five planes selected from RPF0 to RPF4 and video processing function output, and virtual RPF; six planes in total |
| | Order of planes | | The order of six planes selected from RPF0 to RPF4, virtual RPF, and video processing function output can be changed as desired. |
| α plane | Format | | 8 bpp or 1 bpp (α value can be specified through register) |
| | Input source | | RPF0 to RPF4 |
| Fixed α value | Format | | 8 bpp |
| | Input source | | RPF0 to RPF4, virtual RPF, or video processing function output |
| Raster operation | ROP2 (between input channels) | Operator | 16 types (OpenGL2.0 is supported) |
| | | Sources of operation | RPF0 to RPF4, virtual RPF, and video processing function output |
| | | Operation control | RGB/YCbCr and α are operated separately. |
| | ROP3 (between input channels) | Operator | 256-type ROP3 is available by combining ROP2 operations |
| | | Sources of operation | RPF0 to RPF4, virtual RPF, and video processing function output |
| | | Operation control | RGB/YCbCr and α are operated separately. |

Video Processing Functions

| | | | |
|-------------------------------------|---|------------|--|
| Super resolution | Processing method | | Processing within one frame |
| | Scaling ratio | | $\times 1$ (same size) or $\times 2$ |
| | Arbitrary scaling ratio | | Super resolution processing can be combined with the arbitrary ratio scaling function. |
| | α processing | | Fixed α output |
| Scalar with Arbitrary scaling ratio | Scaling factor | | $1/16 < \text{scaling factor} \leq 16$ |
| | Scaling ratio setting | Expression | $1 / [\text{Mant.Frac}]$ |
| | | Precision | Mant = 4 bits, Frac = 12 bits |
| | α processing | | Supported (for $1/4 < \text{scaling factor} \leq 16$) |
| Sharpness processing | Method | | Sharpness or blur is applied in horizontal direction |
| Rotation/flipping | Rotation function | | 90, 180 and 270 degree |
| | Flipping function | | Vertical flipping, horizontal flipping |
| | Combination of rotation and flipping | | Supported |
| | α processing | | Supported |
| Simultaneous processing | All video processing functions and color adjustment functions can be processed simultaneously (the order of function execution is limited). | | |

Color Adjustment Function

| | | | |
|----------------------------|-----------------------|--|---|
| 1D-LUT | LUT configuration | | Independent R/Y, G/Cb, and B/Cr. 256 entries each |
| 3D-LUT | Number of grid points | | $17 \times 17 \times 17$ |
| HSV color space conversion | RGB to HSV | | 8 bits each for H/S/V |
| | HSV to RGB | | 8 bits each for H/S/V |
| 1D Histogram | Number of bins | | 64 for each color component 256 for Y component or Max-RGB |
| | Number of hue areas | | 6 |
| 2D Histogram | Number of hue areas | | 6 |
| | Number of bins | | 32 for each hue area |

33.1.3 External Pins

VSP2 doesn't have external pins.

33.1.4 Register Configuration

(1) Base Address

Below are the base addresses of each VSP unit. VSPBC, VSPBD, VSPB, VSPBS and VSPIO~1 has 32Kbyte address space. VSPD0~2 has 28Kbyte address space.

| | |
|-----------|--|
| VSPBC: | H'FE92_0000 [RZ/G2H] |
| VSPBD: | H'FE96_0000 [RZ/G2H] |
| VSPB: | H'FE96_0000 [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] |
| VSPIO: | H'FE9A_0000 |
| VSP1: | H'FE9B_0000 [RZ/G2H] |
| VSPD0 (*) | H'FEA2_0000 |
| VSPD1 | H'FEA2_8000 |
| VSPD2 | H'FEA3_0000 [RZ/G2M V1.3, RZ/G2M V3.0] |

Note: * For RZ/G2H, RZ/G2N, instance of VSPD0 is VSPDL.

(2) Register/Table Address

Table 33.4 shows the VSP2 register configuration (represents in relative address).

Table 33.4 VSP2 Register Configuration

| Register Name | Abbr. | R/W | Offset Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | | |
|--------------------------------------|----------------|-----------|----------------|---------------|-------------|--|----------------------------|--------|--------|--|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| VSP2 Start Registers 0 | VI6_CMD0 | R/W *1 | H'0000 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| VSP2 Start Registers 1 | VI6_CMD1 | R/W *1 | H'0004 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Clock Control Register0 | VI6_CLK_CTRL0 | R/W | H'0010 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Clock Control Register1 | VI6_CLK_CTRL1 | R/W | H'0014 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Dynamic Clock Stop Control Register | VI6_CLK_DCSWT | R/W | H'0018 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Dynamic Clock Stop Disable Register0 | VI6_CLK_DCSTM0 | R/W | H'001C | H'0000_0000 | 32 | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Register Name | Abbr. | R/W | Offset Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | | |
|--------------------------------------|-------------------|-----|-------------------|---------------|----------------|---|----------------------------|--------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| Dynamic Clock Stop Disable Register1 | VI6_CLK_DCSM1 | R/W | H'0020 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Software Reset Register | VI6_SRESET | R/W | H'0028 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Module Reset Enable Register0 | VI6_MRESET_ENB0 | R/W | H'002C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Module Reset Enable Register1 | VI6_MRESET_ENB1 | R/W | H'0030 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Module Reset Issuing Register | VI6_MRESET | R/W | H'0034 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Operating Status Register | VI6_STATUS | R | H'0038 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPF0 Interrupt Enable Registers | VI6_WPF0_IRQ_ENB | R/W | H'0048 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPF0 Interrupt Status Registers | VI6_WPF0_IRQ_STA | R/W | H'004C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPF1 Interrupt Enable Registers | VI6_WPF1_IRQ_ENB | R/W | H'0054 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPF1 Interrupt Status Registers | VI6_WPF1_IRQ_STA | R/W | H'0058 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Display0 Interrupt Enable Register | VI6_DISP0_IRQ_ENB | R/W | H'0078 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Display0 Interrupt Status Register | VI6_DISP0_IRQ_STA | R/W | H'007C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Display1 Interrupt Enable Register | VI6_DISP1_IRQ_ENB | R/W | H'00B4 | H'0000_0000 | 32 | √ | — | √ | — | |

**Second Generation
RZ/G Series Products**

| Register Name | Abbr. | R/W | Offset Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | | |
|--|----------------------|-----|----------------------|---------------|----------------|---|----------------------------|--------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| Display1 Interrupt Status Register | VI6_DISP1_IRQ_STA | R/W | H'00B8 | H'0000_0000 | 32 | √ | — | √ | — | |
| Display List Control Register | VI6_DL_CTRL | R/W | H'0100 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Display List-0 Header Address Register | VI6_DL_HDR_ADDR0 | R/W | H'0104 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Display List-1 Header Address Register | VI6_DL_HDR_ADDR1 | R/W | H'0108 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Display List0 Data Swapping Register | VI6_DL_SWAP0 | R/W | H'0114 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Extended Display List0 Control Register | VI6_DL_EXT_CTRL0 | R/W | H'011C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Display List Body Size Register-0 | VI6_DL_BODY_SIZE0 | R/W | H'0120 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Display List-0 Header Reference Address Register | VI6_DL_HDR_REF_ADDR0 | R | H'0130 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Display List-1 Header Reference Address Register | VI6_DL_HDR_REF_ADDR1 | R | H'0134 | H'0000_0000 | 32 | √ | — | √ | — | |
| Extended Display List1 Control Register | VI6_DL_EXT_CTRL1 | R/W | H'0140 | H'0000_0000 | 32 | √ | — | √ | — | |
| Display List1 Data Swapping Register | VI6_DL_SWAP1 | R/W | H'014C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Basic Read Size Registers | VI6_RPFn_SRC_BSIZE | R/W | H'0300 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Register Name | Abbr. | R/W | Offset Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | | |
|---|-----------------------|-----|-------------------|---------------|----------------|---|----------------------------|--------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| RPFn Extended Read Size Registers | VI6_RPFn_SRC_ESIZE | R/W | H'0304 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Input Format Registers | VI6_RPFn_INFMT | R/W | H'0308 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Data Swapping Registers | VI6_RPFn_DSWAP | R/W | H'030C + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Display Location Registers | VI6_RPFn_LOC | R/W | H'0310 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn a Plane Selection Control Registers | VI6_RPFn_ALPH_SEL | R/W | H'0314 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Virtual Plane Color Information Registers | VI6_RPFn_VRTCOL_SET | R/W | H'0318 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Mask Control Registers | VI6_RPFn_MSKCTRL | R/W | H'031C + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn IROP-SRC Input Value Registers 0 | VI6_RPFn_MSKSET0 | R/W | H'0320 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn IROP-SRC Input Value Registers 1 | VI6_RPFn_MSKSET1 | R/W | H'0324 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Color Keying Control Registers | VI6_RPFn_CKEY_CTRL | R/W | H'0328 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Color Keying Color Setting Registers-0 | VI6_RPFn_CKEY_SET0 | R/W | H'032C + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Color Keying Color Setting Registers-1 | VI6_RPFn_CKEY_SET1 | R/W | H'0330 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Source Picture Memory Stride Setting Registers | VI6_RPFn_SRCM_PSTRIDE | R/W | H'0334 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Register Name | Abbr. | R/W | Offset Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | | |
|--|-----------------------|-----|-------------------|---------------|----------------|---|----------------------------|--------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| RPFn Source α Memory Stride Setting Registers | VI6_RPFn_SRCM_ASTRIDE | R/W | H'0338 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Source Y/RGB Address Registers | VI6_RPFn_SRCM_ADDR_Y | R/W | H'033C + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Source Chroma Address Registers 0 | VI6_RPFn_SRCM_ADDR_C0 | R/W | H'0340 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Source Chroma Address Registers 1 | VI6_RPFn_SRCM_ADDR_C1 | R/W | H'0344 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Source α Address Registers | VI6_RPFn_SRCM_ADDR_AI | R/W | H'0348 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPFn Bus Access Control Registers | VI6_RPFn_BAC | R/W | H'0350 + H'0100*n | H'0000_0000 | 32 | √ | — | — | — | |
| RPFn Multiple Alpha Control | VI6_RPFn_MULT_ALPH | R/W | H'036C + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPFn-Source-RPF Registers | VI6_WPFn_SRCRPF | R/W | H'1000 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPFn Horizontal Input Size Clipping Registers | VI6_WPFn_HSZCLIP | R/W | H'1004 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPFn Vertical Input Size Clipping Registers | VI6_WPFn_VSZCLIP | R/W | H'1008 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPFn Output Format Registers | VI6_WPFn_OUTFMT | R/W | H'100C + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPFn Data Swapping Registers | VI6_WPFn_DSWAP | R/W | H'1010 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPFn Rounding Control Registers | VI6_WPFn_RNDCTRL | R/W | H'1014 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Register Name | Abbr. | R/W | Offset Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | | |
|--|------------------------|-----|-------------------|---------------|----------------|---|----------------------------|--------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| WPF0 Rotation Control Registers | VI6_WPF0_ROT_CTRL | R/W | H'1018 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPFn Destination Y Plane Memory Stride Registers | VI6_WPFn_DSTM_STRIDE_Y | R/W | H'101C + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPFn Destination C Plane Memory Stride Registers | VI6_WPFn_DSTM_STRIDE_C | R/W | H'1020 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPFn Destination Y/RGB Address Registers | VI6_WPFn_DSTM_ADDR_Y | R/W | H'1024 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPFn Destination Chroma Address Registers 0 | VI6_WPFn_DSTM_ADDR_C0 | R/W | H'1028 + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPFn Destination Chroma Address Registers 1 | VI6_WPFn_DSTM_ADDR_C1 | R/W | H'102C + H'0100*n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPF0 LIF Write Back Control Registers | VI6_WPF0_WRBCK_CTRL | R/W | H'1034 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPF1 LIF Write Back Control Registers | VI6_WPF1_WRBCK_CTRL | R/W | H'1134 | H'0000_0000 | 32 | √ | — | √ | — | |
| RPF0 Routing Register | VI6_DPR_RPF0_ROUTE | R/W | H'2000 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPF1 Routing Register | VI6_DPR_RPF1_ROUTE | R/W | H'2004 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPF2 Routing Register | VI6_DPR_RPF2_ROUTE | R/W | H'2008 | H'0000_0000 | 32 | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Register Name | Abbr. | R/W | Offset Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | | |
|--|---------------------------|-----|-------------------|---------------|----------------|---|----------------------------|--------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| RPF3 Routing Register | VI6_DPR_RPF3_ ROUTE | R/W | H'200C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| RPF4 Routing Register | VI6_DPR_RPF4_ ROUTE | R/W | H'2010 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPF0 Timing Control Register | VI6_DPR_WPF0_ FPORCH | R/W | H'2014 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| WPF1 Timing Control Register | VI6_DPR_WPF1_ FPORCH | R/W | H'2018 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| SRU Routing Register | VI6_DPR_SRU_ ROUTE | R/W | H'2024 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| UDS Routing Register | VI6_DPR_UDS_ ROUTE | R/W | H'2028 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| LUT Routing Register | VI6_DPR_LUT_ ROUTE | R/W | H'203C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| CLU Routing Register | VI6_DPR_CLU_ ROUTE | R/W | H'2040 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HST Routing Register | VI6_DPR_HST_ ROUTE | R/W | H'2044 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HSI Routing Register | VI6_DPR_HSI_ ROUTE | R/W | H'2048 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| BRU Routing Register | VI6_DPR_BRU_ ROUTE | R/W | H'204C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| ILV BRS Routing Register | VI6_DPR_ILV_ BRS_ROUTE | R/W | H'2050 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Sampling Point Register | VI6_DPR_HGO_ SMPPT | R/W | H'2054 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Sampling Point Register | VI6_DPR_HGT_ SMPPT | R/W | H'2058 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| SHP Routing Register | VI6_DPR_SHP_ ROUTE | R/W | H'2060 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Super Resolution Control Register 0 | VI6_SRU_CTRL0 | R/W | H'2200 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Super Resolution Control Register 1 | VI6_SRU_CTRL1 | R/W | H'2204 | H'0000_0000 | 32 | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Register Name | Abbr. | R/W | Offset Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | | |
|---|-------------------------|-----|-------------------|---------------|----------------|---|----------------------------|--------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| Super Resolution Control Register 2 | VI6_SRU_CTRL2 | R/W | H'2208 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Scaling Control Registers | VI6_UDS_CTRL | R/W | H'2300 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Scaling Factor Registers | VI6_UDS_SCALE | R/W | H'2304 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Data Threshold Setting Registers | VI6_UDS_ALPTH | R/W | H'2308 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Data Replacing Value Setting Registers | VI6_UDS_ALPVAL | R/W | H'230C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Passband Registers | VI6_UDS_PASS_ BWIDTH | R/W | H'2310 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Scaling Filter Horizontal Phase Registers | VI6_UDS_HPHASE | R/W | H'2314 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| 2D IPC Setting Register | VI6_UDS_IPC | R/W | H'2318 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| UDS Horizontal Input Clipping Registers | VI6_UDS_HSZCLIP | R/W | H'231c | H'0000_0000 | 32 | √ | √ | √ | √ | |
| UDS Output Size Clipping Registers | VI6_UDS_CLIP_ SIZE | R/W | H'2324 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Color Fill Register | VI6_UDS_FILL_ COLOR | R/W | H'2328 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| LUT Control Register | VI6_LUT_CTRL | R/W | H'2800 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| CLU Control Register | VI6_CLU_CTRL | R/W | H'2900 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HST Control Register | VI6_HST_CTRL | R/W | H'2A00 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HSI Control Register | VI6_HSI_CTRL | R/W | H'2B00 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| BRU Input Control Register | VI6_BRU_INCTRL | R/W | H'2C00 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Size Register of BRU Input Virtual RPF | VI6_BRU_VIRRPF_ SIZE | R/W | H'2C04 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Display Location Register of BRU Input Virtual RPF | VI6_BRU_VIRRPF_ LOC | R/W | H'2C08 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Color Information Register of BRU Input Virtual RPF | VI6_BRU_VIRRPF_ COL | R/W | H'2C0C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| BRU Control Registers A | VI6_BRUA_CTRL | R/W | H'2C10 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| BRU Blend Control Registers A | VI6_BRUA_BLD | R/W | H'2C14 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| BRU Control Registers B | VI6_BRUB_CTRL | R/W | H'2C18 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| BRU Blend Control Registers B | VI6_BRUB_BLD | R/W | H'2C1C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| BRU Control Registers C | VI6_BRUC_CTRL | R/W | H'2C20 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| BRU Blend Control Registers C | VI6_BRUC_BLD | R/W | H'2C24 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| BRU Control Registers D | VI6_BRUD_CTRL | R/W | H'2C28 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| BRU Blend Control Registers D | VI6_BRUD_BLD | R/W | H'2C2C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| BRU Raster Operation Control Register | VI6_BRU_ROP | R/W | H'2C30 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| BRU Control Registers E | VI6_BRUE_CTRL | R/W | H'2C34 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| BRU Blend Control Registers E | VI6_BRUE_BLD | R/W | H'2C38 | H'0000_0000 | 32 | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Register Name | Abbr. | R/W | Offset Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | | |
|--|------------------------------------|-----|----------------|---------------|-------------|--|----------------------------|--------|--------|--|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| HGO Detection Window Offset Register | VI6_HGO_OFFSET | R/W | H'3000 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Detection Window Size Register | VI6_HGO_SIZE | R/W | H'3004 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Mode Register | VI6_HGO_MODE | R/W | H'3008 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO LB Detection Threshold Register | VI6_HGO_LB_TH | R/W | H'300C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Horizontal Position Register for LB Detection Zone-0 | VI6_HGO_LB0_H | R/W | H'3010 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Vertical Position Register for LB Detection Zone-0 | VI6_HGO_LB0_V | R/W | H'3014 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Horizontal Position Register for LB Detection Zone-1 | VI6_HGO_LB1_H | R/W | H'3018 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Vertical Position Register for LB Detection Zone-1 | VI6_HGO_LB1_V | R/W | H'301C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Horizontal Position Register for LB Detection Zone-2 | VI6_HGO_LB2_H | R/W | H'3020 | H'000_0000 | 32 | √ | √ | √ | √ | |
| HGO Vertical Position Register for LB Detection Zone-2 | VI6_HGO_LB2_V | R/W | H'3024 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Horizontal Position Register for LB Detection Zone-3 | VI6_HGO_LB3_H | R/W | H'3028 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Vertical Position Register for LB Detection Zone-3 | VI6_HGO_LB3_V | R/W | H'302C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Component-R Histogram n Register (n = 0 to 63) | VI6_HGO_R_HISTO_n (n = 0 to 63) | R | H'3030 + 4n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Component-R Min/Max Value Register | VI6_HGO_R_MAXMIN | R | H'3130 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Component-R Sum Register | VI6_HGO_R_SUM | R | H'3134 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Component-R LB Detection Result Register | VI6_HGO_R_LB_DET | R | H'3138 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Component-G Histogram n Register (n = 0 to 63) | VI6_HGO_G_HISTO_n (n = 0 to 63) | R | H'3140 + 4n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Component-G Min/Max Value Register | VI6_HGO_G_MAXMIN | R | H'3240 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Component-G Sum Register | VI6_HGO_G_SUM | R | H'3244 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Component-G LB Detection Result Register | VI6_HGO_G_LB_DET | R | H'3248 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Component-B Histogram n Register (n = 0 to 63) | VI6_HGO_B_HISTO_n (n = 0 to 63) | R | H'3250 + 4n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Component-B Min/Max Value Register | VI6_HGO_B_MAXMIN | R | H'3350 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Component-B Sum Register | VI6_HGO_B_SUM | R | H'3354 | H'0000_0000 | 32 | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Register Name | Abbr. | R/W | Offset Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | | |
|--|-----------------------|-----|-------------------|---------------|----------------|---|----------------------------|--------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| HGO Component-B LB Detection Result Register | VI6_HGO_B_LB_DET | R | H'3358 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Y/MaxRGB Extended Histogram Address Register | VI6_HGO_EXT_HIST_ADDR | R/W | H'335C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Y/MaxRGB Extended Histogram Data Register | VI6_HGO_EXT_HIST_DATA | R | H'3360 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGO Write Buffer Side | VI6_HGO_WBUFS | R/W | H'3364 | H'0000_0000 | 32 | — | √ | — | — | |
| HGO Read Buffer Side | VI6_HGO_RBUFS | R/W | H'3368 | H'0000_0000 | 32 | — | √ | — | — | |
| HGO Histogram Storing Address Register | VI6_HGO_HISTADD | R/W | H'336C | H'0000_0000 | 32 | √ | — | √ | √ | |
| HGO Histogram Swapping Register | VI6_HGO_HSWAP | R/W | H'3370 | H'0000_0000 | 32 | √ | — | √ | √ | |
| HGO Parameter Register Reset | VI6_HGO_REGRST | W | H'33FC | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Detection Window Offset Register | VI6_HGT_OFFSET | R/W | H'3400 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Detection Window Size Register | VI6_HGT_SIZE | R/W | H'3404 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Mode Register | VI6_HGT_MODE | R/W | H'3408 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Hue Area 0 Register | VI6_HGT_HUE_AREA_0 | R/W | H'340C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Hue Area 1 Register | VI6_HGT_HUE_AREA_1 | R/W | H'3410 | H'0000_0000 | 32 | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Register Name | Abbr. | R/W | Offset Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | | |
|--|------------------------|-----|-------------------|---------------|----------------|---|----------------------------|--------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| HGT Hue Area 2 Register | VI6_HGT_HUE_ AREA_2 | R/W | H'3414 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Hue Area 3 Register | VI6_HGT_HUE_ AREA_3 | R/W | H'3418 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Hue Area 4 Register | VI6_HGT_HUE_ AREA_4 | R/W | H'341C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Hue Area 5 Register | VI6_HGT_HUE_ AREA_5 | R/W | H'3420 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT LB Detection Threshold Register | VI6_HGT_LB_TH | R/W | H'3424 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Horizontal Position Register for LB Detection Zone-0 | VI6_HGT_LB0_H | R/W | H'3428 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Vertical Position Register for LB Detection Zone-0 | VI6_HGT_LB0_V | R/W | H'342C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Horizontal Position Register for LB Detection Zone-1 | VI6_HGT_LB1_H | R/W | H'3430 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Vertical Position Register for LB Detection Zone-1 | VI6_HGT_LB1_V | R/W | H'3434 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Horizontal Position Register for LB Detection Zone-2 | VI6_HGT_LB2_H | R/W | H'3438 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Vertical Position Register for LB Detection Zone-2 | VI6_HGT_LB2_V | R/W | H'343C | H'0000_0000 | 32 | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Register Name | Abbr. | R/W | Offset Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | | |
|--|---|-----|--------------------------|---------------|----------------|---|----------------------------|--------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| HGT Horizontal Position Register for LB Detection Zone-3 | VI6_HGT_LB3_H | R/W | H'3440 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Vertical Position Register for LB Detection Zone-3 | VI6_HGT_LB3_V | R/W | H'3444 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Histogram m-n Register (m = 0 to 5, n = 0 to 31) | VI6_HGT_HISTO_m_n (m = 0 to 5, n = 0 to 31) | R | H'3450 + 128m + 4n | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Max/Min Value Register | VI6_HGT_MAXMIN | R | H'3750 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Sum Register | VI6_HGT_SUM | R | H'3754 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT LB Detection Result Register | VI6_HGT_LB_DET | R | H'3758 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| HGT Write Buffer Side | VI6_HGT_WBUFS | R/W | H'3764 | H'0000_0000 | 32 | — | √ | — | — | |
| HGT Read Buffer Side | VI6_HGT_RBUFS | R/W | H'3768 | H'0000_0000 | 32 | — | √ | — | — | |
| HGT Histogram Storing Address Register | VI6_HGT_HISTADD | R/W | H'376C | H'0000_0000 | 32 | √ | — | √ | √ | |
| HGT Histogram Swapping Register | VI6_HGT_HSWAP | R/W | H'3770 | H'0000_0000 | 32 | √ | — | √ | √ | |
| HGT Parameter Register Reset | VI6_HGT_REGRST | W | H'37FC | H'0000_0000 | 32 | √ | √ | √ | √ | |
| ILV Control Register | VI6_ILV_CTRL | R/W | H'3800 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| BRS Input Control Register | VI6_BRS_INCTRL | R/W | H'3900 | H'0000_0000 | 32 | √ | — | √ | — | |
| Size Register of BRS Input Virtual RPF | VI6_BRS_VIRRPF_SIZE | R/W | H'3904 | H'0000_0000 | 32 | √ | — | √ | — | |
| Display Location Register of BRS Input Virtual RPF | VI6_BRS_VIRRPF_LOC | R/W | H'3908 | H'0000_0000 | 32 | √ | — | √ | — | |
| Color Information Register of BRS Input Virtual RPF | VI6_BRS_VIRRPF_COL | R/W | H'390C | H'0000_0000 | 32 | √ | — | √ | — | |
| BRS Control Registers A | VI6_BRSA_CTRL | R/W | H'3910 | H'0000_0000 | 32 | √ | — | √ | — | |
| BRS Blend Control Registers A | VI6_BRSA_BLD | R/W | H'3914 | H'0000_0000 | 32 | √ | — | √ | — | |
| BRS Control Registers B | VI6_BRSB_CTRL | R/W | H'3918 | H'0000_0000 | 32 | √ | — | √ | — | |
| BRS Blend Control Registers B | VI6_BRSB_BLD | R/W | H'391C | H'0000_0000 | 32 | √ | — | √ | — | |
| LIF1 Control Registers | VI6_LIF1_CTRL | R/W | H'3A00 | H'0000_0000 | 32 | √ | — | √ | — | |
| LIF1 Clock Stop Buffer Control Register | VI6_LIF1_CSBTH | R/W | H'3A04 | H'0000_0000 | 32 | √ | — | √ | — | |
| LIF0 Control Registers | VI6_LIF0_CTRL | R/W | H'3B00 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| LIF0 Clock Stop Buffer Control Register | VI6_LIF0_CSBTH | R/W | H'3B04 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| LIF0 Buffer Attribute Register | VI6_LIF0_LBA | R/W | H'3B0C | H'0000_0000 | 32 | — | √ | — | — | |
| SHP Control Register0 | VI6_SHP_CTRL0 | R/W | H'3E00 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| SHP Control Register1 | VI6_SHP_CTRL1 | R/W | H'3E04 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| SHP Control Register2 | VI6_SHP_CTRL2 | R/W | H'3E08 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| IP Version Register | VI6_IP_VERSION | R | H'3F00 | (*3) | 32 | √ | √ | √ | √ | |

**Second Generation
RZ/G Series Products**

| Register Name | Abbr. | R/W | Offset Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | | |
|----------------------------|--|-----|--|---------------|----------------|---|----------------------------|--------|--------|--|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| RPFn OSD-CLUT Table | VI6_CLUTn_ TBL0 to VI6_CLUTn_ TBL255 | R/W | H'4000 + H'0400*n to H'43FC + H'0400*n | Undefined | 32 | √ | √ | √ | √ | |
| LUT table | VI6_LUT_TBL_0 to VI6_LUT_TBL_ 255 | R/W | H'7000 to H'73FC | Undefined | 32 | √ | √ | √ | √ | |
| CLU table address register | VI6_CLU_ADDR | R/W | H'7400 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| CLU table data register | VI6_CLU_DATA | R/W | H'7404 | Undefined | 32 | √ | √ | √ | √ | |

Notes: 1. Read only bits or write only bits or read/write bits are mixed in the registers.

2. Initial value of IP version registers depends on product and Modules. See section 33.2.22.1 for more detail.

33.1.5 Connected Module

Table 33.5 shows modules connected to the VSP2. Table 33.6 shows channel number of DU which each VSPD/LIF is connected.

Table 33.5 Connected Module

| Module | Connected Module | Description |
|---|------------------|----------------|
| VSPI [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] | FCPVI | FCPV for VSPI |
| VSPBD [RZ/G2H] | FCPVB | FCPV for VSPBD |
| VSPBC [RZ/G2H] | FCPVB | FCPV for VSPBC |
| VSPB [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] | FCPVB | FCPV for VSPB |
| VSPD | FCPVD | FCPV for VSPD |
| | DU | Display Unit |
| VSPDL [RZ/G2H, RZ/G2N] | FCPVD | FCPV for VSPDL |
| | DU | Display Unit |

Table 33.6 Correspondence between ch number of VSPD/LIF and ch number of DU

| Product | VSPD ch | LIF ch | DU ch |
|----------------------------|---------|--------|-------|
| RZ/G2H | VSPD0 | LIF0 | DU0 |
| | | LIF1 | DU3 |
| | VSPD1 | LIF0 | DU1 |
| RZ/G2M V1.3 RZ/G2M V3.0 | VSPD0 | LIF0 | DU0 |
| | VSPD1 | LIF0 | DU1 |
| | VSPD2 | LIF0 | DU2 |
| RZ/G2N | VSPD0 | LIF0 | DU0 |
| | | LIF1 | DU3 |
| | VSPD1 | LIF0 | DU1 |
| RZ/G2E | VSPD0 | LIF0 | DU0 |
| | VSPD1 | LIF0 | DU1 |

Bus access of VSP2 is carried out through FCPV (Frame Compression IP for VSP) module in RZ/G Gen2 system. So, refer to also section 35 FCP User's Manual in the system.

FCP_CFG0.FCPVSEL described in section 35 FCP User's Manual is configuration register. Set following value for FCPVI, FCPVB and FCPVD respectively. Otherwise VSP may not work properly.

- Set 1 to FCP_CFG0.FCPVSEL to enable configuration of FCPVI.
- Set 0 to FCP_CFG0.FCPVSEL to enable configuration of FCPVB or FCPVD.

Frame compression for write is carried out in FCPV, but enable or disable frame compression should be set in VSP register (VI6_WPFn_OUTFMT.FCNL).

33.2 Register Description

33.2.1 Notational Conventions for Registers and Bit Fields

This document uses the following notational conventions for the VSP2 registers and bit fields.

1. The names of registers and bits are written in uppercase.
2. A bit or bit field in a register is indicated as [register name.bit name]. For example, the STRCMD bit in the VI6_CMD0 register is indicated as VI6_CMD0.STRCMD.
3. Lowercase "n" in a register name or a bit name indicates an integer and the range of value n is defined when necessary. For example, VI6_CMDn.STRCMD (n = 0, 1) indicates the STRCMD bit in two registers VI6_CMD0.STRCMD and VI6_CMD1.STRCMD. For RPFn, WPFn, and UDS, when the range of value n is not defined, RPFn (n = 0, 1, 2, 3, 4), WPFn (n = 0, 1), and UDS are assumed.
4. In each subsection for register description in section 33.2, when only a bit name is written without showing its register name, the bit is in the register described in that subsection.
5. A wildcard (*) indicates any characters in a name and represents all registers or bits that match the specified first part of a name. For example, when there are two registers VI6_RPF_SRC_BSIZE and VI6_RPF_SRC_ESIZE, VI6_RPF_SRC_* indicates both registers.

33.2.2 Register Classification

The VSP2 registers are arranged in the following order; the general control registers to display list control registers control operation of the entire VSP2, and the other registers control each image processing and specify parameters for the processing. The functions of the registers are described in this order starting from section 33.2.4.

1. General control registers
2. Display list control registers
3. RPF control registers (RPF0, RPF1, RPF2, RPF3 and RPF4)
4. WPF control registers (WPF0 and WPF1)
5. DPR control registers
6. SRU control registers
7. UDS control registers
8. LUT control register
9. CLU control register
10. HST control register
11. HSI control register
12. BRU control registers
13. HGO control registers
14. HGT control registers
15. ILV control register
16. BRS control registers
17. LIF control registers (LIF0 and LIF1)
18. SHP control registers
19. VSP internal registers

The VSP2 has five RPF channels and the register configuration is the same for all of RPF0 to RPF4. However, some bit fields have restrictions in certain RPFs. These restrictions are included in the description of the corresponding bit fields and registers. Likewise, the register configuration is the same for all Two WPF channels (WPF0 to WPF1), but some bit fields have restrictions in certain WPFs; the restrictions are included in the description of the corresponding bit fields and registers. For each register address, refer to section 33.1.4

33.2.3 Restrictions on Access to Registers and Lookup Tables

The VSP2 has control registers and lookup tables. All VSP2 registers are writable and readable by only 32 bits unit. To write partial bits in each register, read-modify-write is needed. When accessing the addresses where these registers and lookup tables are allocated, the following restrictions should be satisfied. If any restriction is violated, the VSP2 will not operate correctly.

1. For the read-only bits and reserved bits in all VSP2 registers, writing 1 is prohibited unless otherwise specified.
2. Addresses undefined in section 33.1.4 are reserved areas and write access is prohibited in these areas.
3. For all registers and lookup tables, except VI6_CMDn, VI6_SRESET, VI6_*IRQ* and two plane registers such as VI6_DL_HDR_ADDRn, VI6_DL_BODY_SIZE0, VI6_HGO_WBUFS and VI6_HGT_WBUFS, modifying register values during operation of the module is prohibited. Modify registers while the corresponding module is stopped. For the operating status of the target module, refer to section 33.3.2.
4. There are three types about General control registers (Section 33.2.4) and Display List Control Registers (Section 33.2.5) as below.
 Controlling WPF0 (ex. VI6_CMD0, VI6_SRESET.SRST0)
 Controlling WPF1 (ex. VI6_CMD1, VI6_SRESET.SRST1)
 Common setting of WPF0 and WPF1 (ex. VI6_DL_CTRL.AR_WAIT [15:0])
 VSPB, VSPBC, VSPBD, VSPBS, and VSPI, doesn't have WPF1.
 Therefore, don't write any values except default value to registers related to controlling WPF1 for VSPB, VSPBC, VSPBD, VSPBS, VSPI.

Table 33.7 Correspondence between Modules and Register Names

| Module Name | Register Name |
|-------------------|---------------|
| RPFn (n = 0 to 4) | VI6_RPFn_* |
| WPFn (n = 0, 1) | VI6_WPFn_* |
| DPR | VI6_DPR_* |
| SRU | VI6_SRU_* |
| SHP | VI6_SHP_* |
| UDS | VI6_UDS_* |
| LUT | VI6_LUT_CTRL |
| CLU | VI6_CLU_CTRL |
| HST | VI6_HST_CTRL |
| HSI | VI6_HSI_CTRL |
| BRU | VI6_BRU_* |
| HGO | VI6_HGO_* |
| HGT | VI6_HGT_* |
| ILV | VI6_ILV_* |
| BRS | VI6_BRS_* |
| LIFn (n = 0, 1) | VI6_LIFn_* |

33.2.4 General Control Registers

33.2.4.1 VSP2 Start Registers n (VI6_CMDn: n = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|---------|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | UPD HDR | — | — | — | STR CMD |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|----------|---------------|-----|---|
| 31 to 5, 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | UPDHDR | B'0 | R/W | Reserved state of updating DLH address. UPDHDR can be negated by software reset (VI6_SRESET.SRSTn). [Write] 0: NOP 1: Don't set 1 to this bit [Read] 0: Not reserved to update DLHeader address and VI6_HGO/HGT_WBUFS. 1: Reserved to update DLHeader address and VI6_HGO/HGT_WBUFS. Set 0 to this bit when STRCMD bit is set to 1. |
| 0 | STRCMD | B'0 | R/W | Start reservation of each WPF VSP2 has two output channels (WPFs). Two WPFs can be started at the same time. VI6_CMDn.STRCMD controls WPFn. Writing 1 to this bit starts WPFn in VSP2. Set this bit for activation only after all register settings in each output channel have been completed. If WPFn is idle, WPFn starts right after this bit is set to 1. If WPFn is active, writing 1 to this bit reserves starting WPFn operation. Wait to set 1 to VI6_CMDn.STRCMD until this bit is read as 0. VI6_CMDn.STRCMD can be negated by software reset (VI6_SRESET.SRSTn). [Write] 0: Start reservation of WPFn is canceled. 1: Start reservation of WPFn is set. [Read] 0: Starting VSP is not reserved. 1: Starting VSP is reserved. |

The basic concept of image processing operation started by activating the VSP2 is shown in Figure 33.11. The actual data input/output is executed by the MAU, which is the bus interface module, as described in section 33.1.2, but conceptually

the RPF works as the data entry point to the VSP2 and the WPF works as the data exit point. To process images through the VSP2, the RPF (entrance) and WPF (exit) should be connected and a data path from the RPF to the WPF should be formed.

To connect RPF_n to WPF_n, specify RPF_n as the source RPF for WPF_n in VI6_WPF_n_SRCRPF, which is a register for WPF_n (refer to section 33.2.7.1). This setting determines that RPF_n will be started when WPF_n is started through VI6_CMD_n.

A data path to execute desired image processing should then be formed between the RPF (entrance) and WPF (exit). To form a data path, connect the necessary function modules in the VSP2 between RPF and WPF. This function is provided by the DPR; specify the information for each module connection in data path routing registers VI6_DPR_*_ROUTE (refer to section 33.2.8).

After a data path is formed (RPF_n → WPF_n) as described above, starting output module WPF_n in the VSP2 through VI6_CMD_n starts all function modules connected to WPF_n and the desired image processing is executed. According to this design concept, starting a WPF module means starting the VSP2.

There are two types of data path configuration in the VSP2; one is "a single input module to a single output module" as shown in Figure 33.11 (A), and the other is "multiple input modules to a single output module" as shown in Figure 33.11 (B).

Figure 33.11 (A) shows an example of a configuration where modules with single input and single output are implemented through the DPR.

Figure 33.11 (B) shows another configuration example where the module with multiple input and single output is implemented through the DPR.

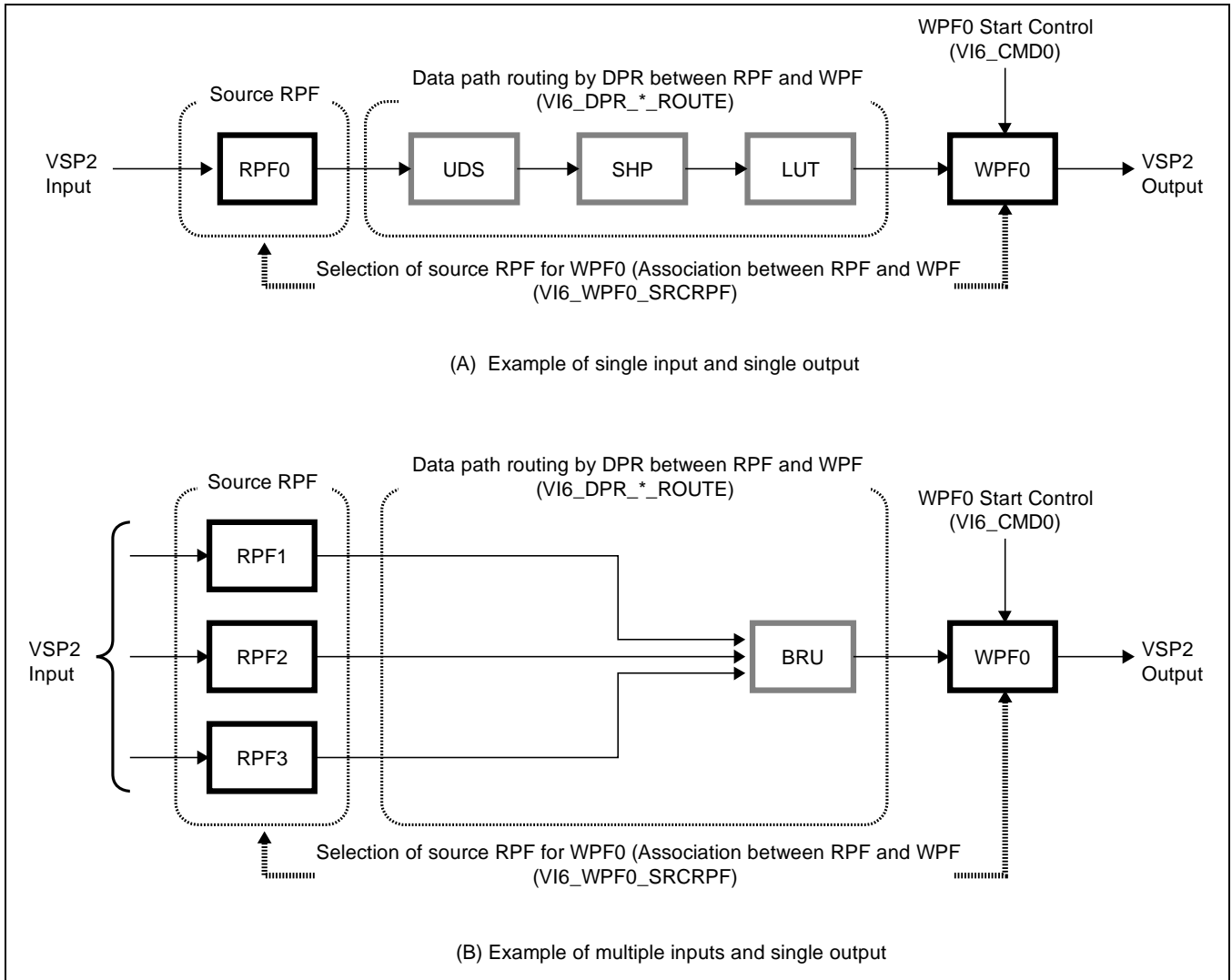


Figure 33.11 Basic Concept of VSP2 Startup

33.2.4.2 Clock Control Register 0 (VI6_CLK_CTRL0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|------|------|-----|-----|-----|----|----|----|------|-----|-----|-----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | GCS0 | — | — | — | — | — | — | — | — | — | — | — | GCS1 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | GCS2 | | | | — | — | — | GCS3 | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---|----------|---------------|-----|--|
| 31 to 29, 27 to 17, 15 to 12, 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | GCS0 | B'0 | R/W | Clock Control Setting0 |
| 16 | GCS1 | B'0 | R/W | Clock Control Setting1 |
| 11 to 8 | GCS2 | H'0 | R/W | Clock Control Setting2 |
| 4 to 0 | GCS3 | All 0 | R/W | Clock Control Setting3 |

VSP2 can stop its operating clock for reducing power consumption.

To enable clock stop function, set following registers:

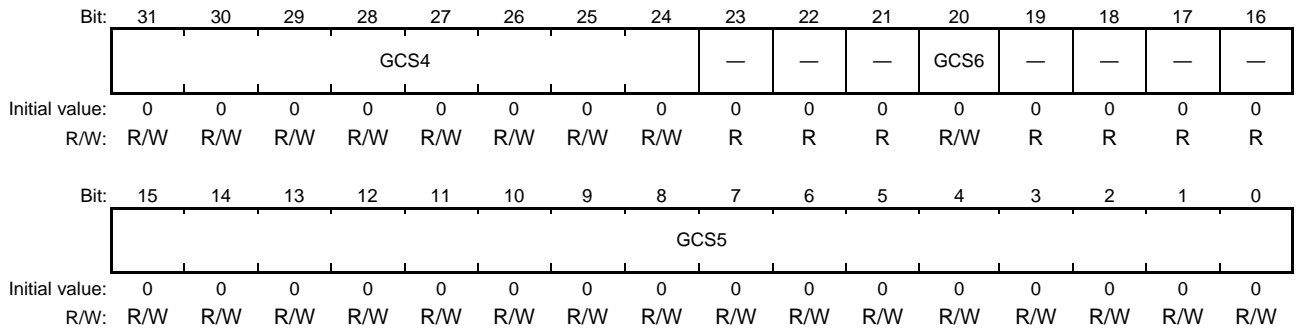
- VI6_CLK_CTRL0 = H'0000_0000
- VI6_CLK_CTRL1 = H'0000_0000
- VI6_CLK_DCSWT = H'0000_0808
- VI6_CLK_DCSTM0 = H'0000_0000
- VI6_CLK_DCSTM1 = H'0000_0000

To disable clock stop function, set following registers:

- VI6_CLK_CTRL0 = H'1001_0F1F
- VI6_CLK_CTRL1 = H'FF10_FFFF
- VI6_CLK_DCSWT = H'0033_0808
- VI6_CLK_DCSTM0 = H'1FFF_0F1F
- VI6_CLK_DCSTM1 = H'FF10_FFFF

33.2.4.3 Clock Control Register 1 (VI6_CLK_CTRL1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|----------|---------------|-----|--|
| 23 to 21, 19 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 31 to 24 | GCS4 | All 0 | R/W | Clock Control Setting4 |
| 20 | GCS6 | B'0 | R/W | Clock Control Setting6 |
| 15 to 0 | GCS5 | All 0 | R/W | Clock Control Setting5 |

See section 33.2.4.2 for detail.

33.2.4.4 Dynamic Clock Stop Control Register (VI6_CLK_DCSWT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

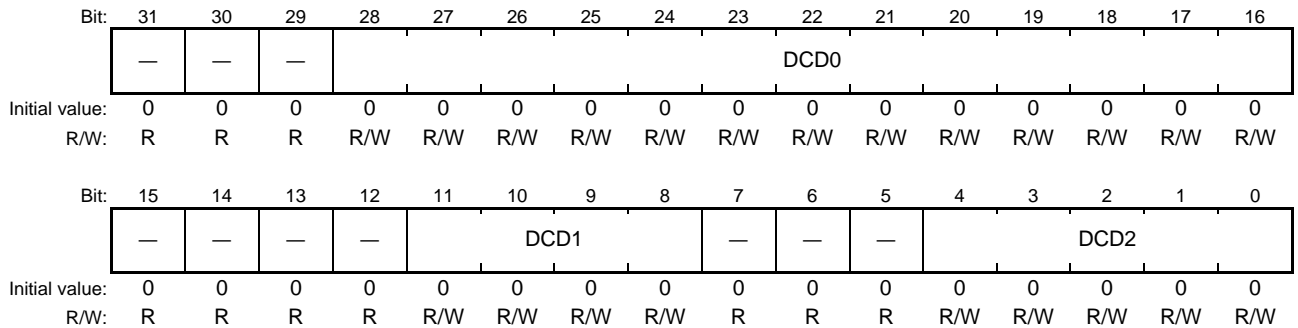
| | | | | | | | | | | | | | | | | |
|----------------|-------------|-----|-----|-----|-----|-----|-----|-----|-------------|-----|------------|-----|-----|------|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | DCC0 [1:0] | — | — | DCC1 | DCC2 | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CSTPW [7:0] | | | | | | | | CSTRW [7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|-------------|---------------|-----|--|
| 31 to 22, 19 to 18 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21, 20 | DCC0 | B'00 | R/W | Dynamic Clock Control Setting0 |
| 17 | DCC1 | B'0 | R/W | Dynamic Clock Control Setting1 |
| 16 | DCC2 | B'0 | R/W | Dynamic Clock Control Setting2 |
| 15 to 8 | CSTPW [7:0] | All 0 | R/W | Dynamic Clock Stop Control 1 Always specify 8. |
| 7 to 0 | CSTRW [7:0] | All 0 | R/W | Dynamic Clock Stop Control 2 Always specify 8. |

See section 33.2.4.2 for detail.

33.2.4.5 Dynamic Clock Stop Disable Register 0 (VI6_CLK_DCSM0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

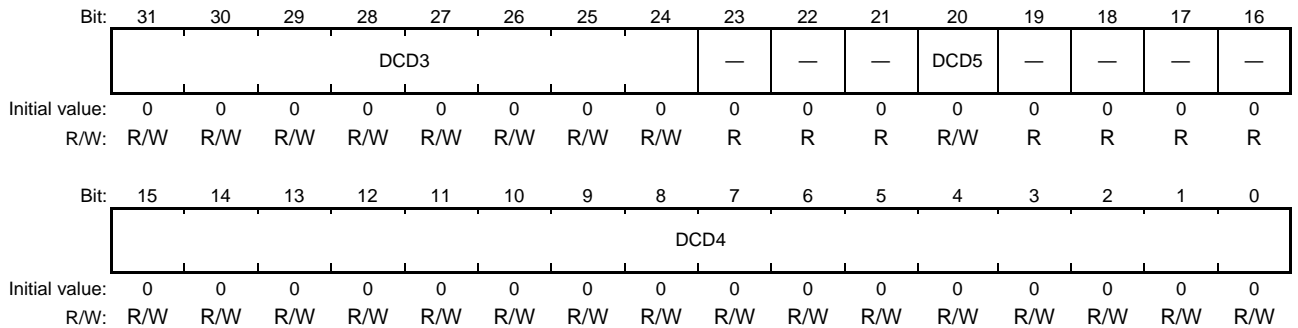


| Bit | Bit Name | Initial Value | R/W | Description |
|----------------------------|----------|---------------|-----|--|
| 31 to 29, 15 to 12, 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | DCD0 | All 0 | R/W | Dynamic Clock Stop Disable Setting0 |
| 11 to 8 | DCD1 | H'0 | R/W | Dynamic Clock Stop Disable Setting1 |
| 4 to 0 | DCD2 | All 0 | R/W | Dynamic Clock Stop Disable Setting2 |

See section 33.2.4.2 for detail.

33.2.4.6 Dynamic Clock Stop Disable Register 1 (VI6_CLK_DCSM1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|----------|---------------|-----|--|
| 23 to 21, 19 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 31 to 24 | DCD3 | All 0 | R/W | Dynamic Clock Stop Disable Setting3 |
| 20 | DCD5 | B'0 | R/W | Dynamic Clock Stop Disable Setting5 |
| 15 to 0 | DCD4 | All 0 | R/W | Dynamic Clock Stop Disable Setting4 |

See section 33.2.4.2 for detail.

33.2.4.7 Software Reset Register (VI6_SRESET)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----------|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SRST 1 | SRST 0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | SRST1 | B'0 | R/W | WPFn Software Reset (SRSTn, n = 0 to 1) |
| 0 | SRST0 | B'0 | R/W | Writing 1 to this bit aborts the current processing in WPFn (the partially-completed image undergoing processing is output). The period until this software reset processing is completed depends on the bus state. When this reset processing is completed, the VI6_WPFn_IRQ_STA.FRE interrupt source bit is set to 1; when the FRE interrupt is enabled, the FRE end interrupt is output to notify the end of the reset processing. This bit is always read as 0. 0: NOP 1: WPFn software reset (*) |

Note: * Applying a software reset to each WPF has the following restrictions.

- Notes:
1. A software reset can be applied to only one of WPF0 to WPF1 through single write access to VI6_SRESET.
 2. After a software reset is issued, no more software reset can be issued to another WPF until the issued software reset processing is completed
 3. The end of software reset processing is notified through the FRE bit in VI6_WPFn_IRQ_STA, but the software reset issued while WPF is stopped is ignored as NOP. As it takes a while until the reset is actually issued after the reset bit is set, the VSP2 may complete operation before the reset is actually issued. In this case, no interrupt is output for the software reset that is issued after the VSP2 completes operation.
 4. If a software reset is issued during downloading of a display list, the downloading processing is not aborted. After the end of downloading that is in progress when a software reset is issued, a frame end interrupt is output.
 5. When software reset is issued to VSP, issue software reset to also FCPV except VSPDL. See Figure 33.66 for VSPD software reset sequence. See section 35 FCP User's Manual for VSPI/VSPBD/VSPBC/VSPBS software reset sequence.

33.2.4.8 Module Reset Enable Register 0 (VI6_MRESET_ENB0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

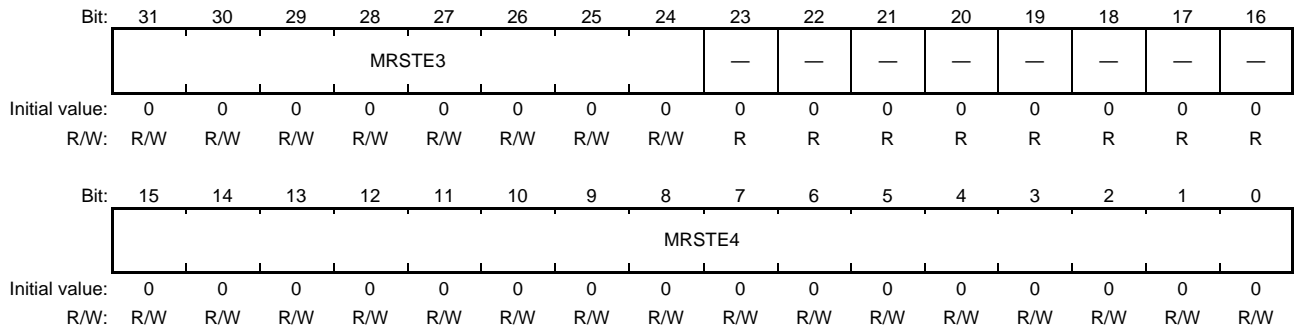
| | | | | | | | | | | | | | | | | |
|----------------|----|----|--------|-----|--------|-----|-----|-----|----|----|----|--------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | MRSTE0 | | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | MRSTE1 | | | | — | — | — | MRSTE2 | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------------------------|----------|---------------|-----|--|
| 31 to 30, 27 to 12, 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 28 | MRSTE0 | All 0 | R/W | Module Reset Enable 0 |
| 11 to 8 | MRSTE1 | H'0 | R/W | Module Reset Enable 1 |
| 4 to 0 | MRSTE2 | All 0 | R/W | Module Reset Enable 2 |

This register is for purpose of h/w debugging.

33.2.4.9 Module Reset Enable Register 1 (VI6_MRESET_ENB1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 23 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 31 to 24 | MRSTE3 | All 0 | R/W | Module Reset Enable 3 |
| 15 to 0 | MRSTE4 | All 0 | R/W | Module Reset Enable 4 |

This register is for purpose of h/w debugging.

33.2.4.10 Module Reset Issuing Register (VI6_MRESET)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | MRS T |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | MRST | B'0 | R/W | Module Reset Assertion This register is for purpose of h/w debugging. |

33.2.4.11 Operating Status Register (VI6_STATUS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|---------|---------|----|----|----------|----------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | FLD ST1 | FLD ST0 | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | SYS1_ACT | SYS0_ACT | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------------------------|----------|---------------|-----|--|
| 31 to 30, 27 to 10, 7 to 0 | — | All 0 | R | Reserved These bits are read as 0 or 1. Do not care about this value. Write access is prohibited. |
| 29 | FLDST1 | B'0 | R | Field status of previous frame of WPF1 [RZ/G2H, RZ/G2N]. 0: Previous finished frame is TOP field. 1: Previous finished frame is BOT field. This bit can be referred in case of AUTO-FLD or AUTO-DISP. This bit is changed at the timing of VI6_WPF1_IRQ_STA.FRE. |
| 28 | FLDST0 | B'0 | R | Field status of previous frame of WPF0. 0: Previous finished frame is TOP field. 1: Previous finished frame is BOT field. This bit can be referred in case of AUTO-FLD or AUTO-DISP. This bit is changed at the timing of VI6_WPF0_IRQ_STA.FRE. |
| 9 | SYS1_ACT | B'0 | R | WPFn Operating Status (SYSn_ACT, n = 0 to 1) |
| 8 | SYS0_ACT | B'0 | R | Each bit indicates the operating or stopped state of control channel n (WPFn). 0: WPFn is stopped. 1: WPFn is operating. |

33.2.4.12 WPFn Interrupt Enable Registers (VI6_WPFn_IRQ_ENB: n = 0, 1)

| | RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--|--------|--------------------------|--------|--------|
| | √ | √ | √ | √ |

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | UNDE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------------|----|----|----|----|----|----|---|---|---|---|---|---|---|---|------|------|
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | DFEE | FREE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

Each bit controls the interrupt enable of the corresponding interrupt source.

0: Interrupt Disabled

1: Interrupt Enabled

| Bit | Bit Name | Initial Value | R/W | Description |
|-------------------|----------|---------------|-----|--|
| 31 to 17, 15 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | UNDE | B'0 | R/W | Interrupt Enable for WPFn (n = 0 to 1) Underrun in case of DU connection. |
| 1 | DFEE | B'0 | R/W | Interrupt Enable for WPFn (n = 0 to 1) Display List Frame End |
| 0 | FREE | B'0 | R/W | Interrupt Enable for WPFn (n = 0 to 1) Frame End |

Each bit in VI6_WPFn_IRQ_STA is set to 1 when the corresponding interrupt source is generated.

VI6_WPFn_IRQ_ENB specifies whether to output an interrupt signal for the generated source. When an interrupt is disabled in this register, no interrupt signal is generated even when the corresponding bit in VI6_WPFn_IRQ_STA is set to 1. When an interrupt is enabled in this register, an interrupt signal is output when the corresponding bit in VI6_WPFn_IRQ_STA is set to 1.

33.2.4.13 WPFn Interrupt Status Registers (VI6_WPFn_IRQ_STA: n = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | UND |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | DFE | FRE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

The read value from each bit is the status of the interrupt source, and the write access to each bit controls the interrupt status.

[Read Access] Interrupt Status

0: No interrupt

1: Interrupt activated

[Write Access] Interrupt Clear

0: The interrupt status is cleared to 0

1: Hold the interrupt status value

| Bit | Bit Name | Initial Value | R/W | Description |
|-------------------|----------|---------------|-----|--|
| 31 to 17, 15 to 2 | — | All 0 | R | Reserved These bits are read as 0 or 1. Do not care about this value. The write value should always be 0. |
| 16 | UND | B'0 | R/W | Interrupt Status and Clear for WPFn (n = 0 to 1) Underrun in case of DU connection. This interrupt source bit is set to 1 when data underrun occurs in case of DU connection. Timing that this bit is set to 1 is at end of 1 frame. |
| 1 | DFE | B'0 | R/W | Interrupt Status and Clear for WPFn (n = 0 to 1) Display List Frame End This interrupt source bit is set to 1 when VSP2 completes one-frame processing while the current frame in enable value stored in the display list header is 1 (refer to section 33.3.3.2). When display lists are not used, this bit is not used. In this case, clear VI6_WPFn_IRQ_ENB.DFEE to 0 to mask the interrupt generation by this interrupt source. This bit can hold the most recent of two times of interrupt status at the maximum. When writing 0 to this bit, the oldest interrupt status is cleared. And the status bit becomes 0 after all interrupt statuses are cleared. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | FRE | B'0 | R/W | <p>Interrupt Status and Clear for WPFn (n = 0 to 1) Frame End</p> <p>This interrupt source bit is set to 1 when VSP2 completes one-frame processing. This bit is also set to 1 when one-frame processing using a display list is completed.</p> <p>The interrupt status is set to 1 by any of the following conditions.</p> <ul style="list-style-type: none"> a) Processing one frame is finished normally b) Software reset is issued during VSP is processing. c) One frame's data from DU is displayed while VSP couldn't transfer one frame data to DU. (in case of Linked with DU) <p>This bit can hold the most recent of two times of interrupt status at the maximum. When writing 0 to this bit, the oldest interrupt status is cleared. And the status bit becomes 0 after all interrupt statuses are cleared.</p> |

VI6_WPFn_IRQ_STA indicates the state of the interrupt sources generated in the VSP2. Whether to output a VSP2 interrupt when an interrupt source is generated and the corresponding bit is set to 1 is determined by the corresponding bit setting in VI6_WPFn_IRQ_ENB. While an interrupt is disabled in VI6_WPFn_IRQ_ENB, the VSP2 does not output an interrupt signal even when an interrupt source is generated, but the source flag in this register is set to 1.

Note that the interrupt source bits in this register cannot be cleared by write access using a display list.

33.2.4.14 Display-n Interrupt Enable Register (VI6_DISPn_IRQ_ENB: n = 0, 1)

[for n = 0]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[for n = 1]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----------|----|----|----------|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | DST E | — | — | MAE E | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R/W | R | R | R | R | R |

Each bit controls the interrupt enable of the corresponding interrupt source.

0: Interrupt Disabled

1: Interrupt Enabled

| Bit | Bit Name | Initial Value | R/W | Description |
|-------------------------------|----------|---------------|-----|--|
| 31 to 9, 7 to 6, 4 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | DSTE | B'0 | R/W | Interrupt Enable for Display Start |
| 5 | MAEE | B'0 | R/W | Interrupt Enable for Display Read Data End |

33.2.4.15 Display-n Interrupt Status Register (VI6_DISPn_IRQ_STA: n = 0, 1)

[for n = 0]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[for n = 1]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----|----|----|-----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | DST | — | — | MAE | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R/W | R | R | R | R | R |

The read value from each bit is the status of the interrupt source, and the write access to each bit controls the interrupt status.

[Read Access] Interrupt Status

0: No interrupt

1: Interrupt activated

[Write Access] Interrupt Clear

0: The interrupt status is cleared to 0

1: Hold the interrupt status value

| Bit | Bit Name | Initial Value | R/W | Description |
|-----------------------|----------|---------------|-----|---|
| 31 to 9, 7, 6, 4 to 0 | — | All 0 | R | Reserved These bits are always read as 0 or 1. Do not care about this value. The write value should always be 0. |
| 8 | DST | B'0 | R/W | Interrupt Status and Clear for Display Start This bit is set to 1 when LIF module transfers the first data to the display module at the beginning of each frame. The timing depends on the output buffer status of LIF module. |
| 5 | MAE | B'0 | R/W | Interrupt Status and Clear for Display Read Data End This bit is set to 1 when all RPFs transfer the last data of the frame to LIF module. The RPF module which is not used by LIF module does not affect this bit. |

33.2.5 Display List Control Registers

33.2.5.1 Display List Control Register (VI6_DL_CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

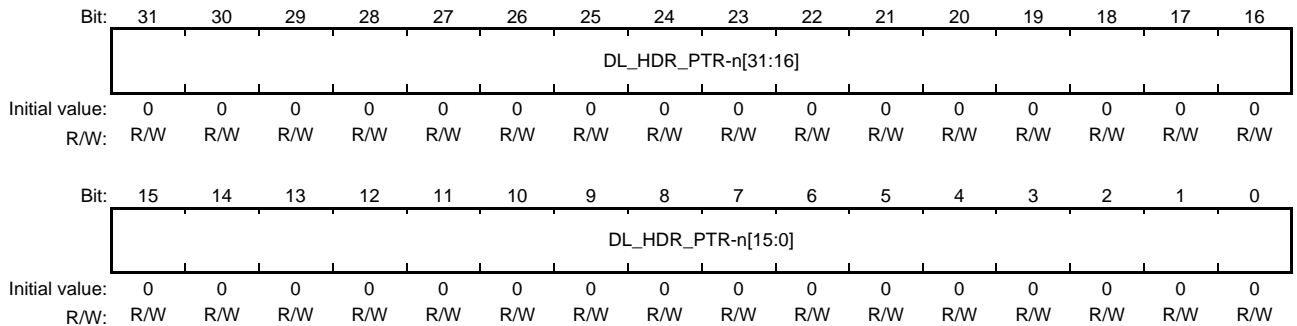
| | | | | | | | | | | | | | | | | |
|----------------|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|----------|----------|----------|-----|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | AR_WAIT [15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | DC2 | — | — | — | DC1 | — | — | — | DLE 1 | RLM 0 | CFM 0 | NH0 | DLE 0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------------------------|----------------|---------------|-----|--|
| 31 to 16 | AR_WAIT [15:0] | All 0 | R/W | Display List Control Setting Always specify 256. |
| 15 to 13, 11 to 9, 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | DC2 | B'0 | R/W | This bit doesn't affect anything to VSP (NOP) |
| 8 | DC1 | B'0 | R/W | This bit doesn't affect anything to VSP (NOP) |
| 4 | DLE1 | B'0 | R/W | Display List Enable/Disable for WPF1 Enables or disables the WPF1 display list function. When using display lists, note the restrictions in section 33.2.3. 0: The display list function is disabled 1: The display list function is enabled |
| 3 | RLM0 | B'0 | R/W | Loading two plane Registers Mode for WPF0. 0: Reserved status (VI6_CMD0.UPDHDR) is accepted by next frame auto start. Two plane registers are downloaded by next_frame_start also. 1: Reserved status (VI6_CMD0.UPDHDR) is not accepted by next frame auto start. pNextHeader in previous frame's DLH is used for loading DisplayList. Two plane registers are not downloaded by next_frame_start also. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | CFM0 | B'0 | R/W | <p>Continuous Frame Mode for Header-less Display List for WPF0</p> <p>This bit determines whether the next frame is automatically started or not. When the updated flag of the display list, VI6_DL_BODY_SIZE0.UPD0, is not updated, the display list of the next frame is not transferred and the same register values are used for the next frame. When the value of VI6_DL_BODY_SIZE0.UPD0 is updated, the new display list is transferred.</p> <p>0: Stopped at the end of every frame 1: The next frame is automatically started</p> |
| 1 | NH0 | B'0 | R/W | <p>Header-less Display List Mode</p> <p>This bit is used for specifying the header-less display list mode. In case of header-less mode, the number of the display lists is 1. The address of the display body is set in VI6_DL_HDR_ADDR0 register, and the body size is set in VI6_DL_BODY_SIZE0 register.</p> <p>When this bit is changed, make sure that VSP2 is stopped. And also make sure the following before starting VSP2.</p> <ul style="list-style-type: none"> - Header Address (VI6_DL_HDR_ADDR0) - Body Size (VI6_DL_BODY_SIZE0) in case of header-less mode <p>0: Use Display List Header (Normal DL Mode) 1: Don't use Display List Header (Header-less Mode)</p> <p>Notes: 1. Only WPF0 supports header-less display list. WPF1 work as the normal display list mode even if the WPF0 is set to header-less display list mode. 2. When DLE0 bit is 0, set 0 to NH0 bit.</p> |
| 0 | DLE0 | B'0 | R/W | <p>Display List Enable/Disable for WPF0</p> <p>Enables or disables the WPF0 display list function. When the display list function is enabled through this bit, all WPF processing channels work in display list mode.</p> <p>When using display lists, note the restrictions in section 33.2.3.</p> <p>0: The display list function is disabled 1: The display list function is enabled</p> |

33.2.5.2 Display List-n Header Address Register (VI6_DL_HDR_ADDRn: n = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------|---------------|-----|---|
| 31 to 0 | DL_HDR_PTR-n [31:0] | All 0 | R/W | <p>Display List-n Header Address</p> <p>These bits specify the address of the display list header to be read for display list-n in 16-byte units (the lower-order four bits are read-only). When WPFn is first started in display list mode, the display list header is loaded from the address specified in this register. After loading of the header is completed, the register value of the display list address is updated to the next header address stored in the loaded header to prepare for loading of the next display list header. After that, this header address updating is repeated.</p> <p>When VI6_DL_CTRL.NHn is 0, writing header address to this register reserves updating two plane registers for the next frame. Two plane registers are VI6_HGO_WBUFS, VI6_HGT_WBUFS and VI6_DL_HDR_ADDRn.</p> <p>Set VI6_HGO_WBUFS and VI6_HGT_WBUFS before setting this register.</p> <p>A value from H'0000_0000 to H'FFFF_FFF0 can be specified.</p> |

33.2.5.3 Display List-n Data Swapping Register (VI6_DL_SWAPn: n = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|----|----|----|----|----|----|----|----|----|----|----|----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | IND | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | LWS | WDS | BTS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 30 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 31 | IND | B'0 | R/W | Enabling independent swap setting per WPF. This bit is available only for VI6_DL_SWAP1. This bit is reserved for VI6_DL_SWAP0. 0: Display list swap for WPF1 is specified by LWS, WDS and BTS in VI6_DL_SWAP0. 1: Display list swap for WPF1 is specified by LWS, WDS and BTS in VI6_DL_SWAP1 |
| 2 | LWS | B'0 | R/W | Display List Data Swapping in long word Units The effect of this bit setting is defined in Table 33.8. 0: Data swapping in long word (32-bit) units is disabled 1: Data swapping in long word (32-bit) units is enabled |
| 1 | WDS | B'0 | R/W | Display List Data Swapping in Word Units The effect of this bit setting is defined in Table 33.8. 0: Data swapping in word (16-bit) units is disabled 1: Data swapping in word (16-bit) units is enabled |
| 0 | BTS | B'0 | R/W | Display List Data Swapping in Byte Units The effect of this bit setting is defined in Table 33.8. 0: Data swapping in byte (8-bit) units is disabled 1: Data swapping in byte (8-bit) units is enabled |

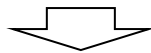
Table 33.8 shows the data order before and after swapping according to the long word, word, and byte swapping settings.

When data order in memory for each format is the same as Table 33.49, set 3'b111 to {LWS, WDS, BTS}. If data order is not the same as the definition, change data order within 8byte unit by these bits as shown in Table 33.8.

Table 33.8 Changing data order according to display list swap register

Data order in memory

| Byte address | 8n+0 | 8n+1 | 8n+2 | 8n+3 | 8n+4 | 8n+5 | 8n+6 | 8n+7 | *_LWS | *_WDS | *_BTS |
|--------------|------|------|------|------|------|------|------|------|-------|-------|-------|
| Data | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 1 | 1 | 1 |
| | 1 | 0 | 3 | 2 | 5 | 4 | 7 | 6 | 1 | 1 | 0 |
| | 2 | 3 | 0 | 1 | 6 | 7 | 4 | 5 | 1 | 0 | 1 |
| | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 1 | 0 | 0 |
| | 4 | 5 | 6 | 7 | 0 | 1 | 2 | 3 | 0 | 1 | 1 |
| | 5 | 4 | 7 | 6 | 1 | 0 | 3 | 2 | 0 | 1 | 0 |
| | 6 | 7 | 4 | 5 | 2 | 3 | 0 | 1 | 0 | 0 | 1 |
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 0 | 0 | 0 |



Data order defined in Table 33.49

| Byte address | 8n+0 | 8n+1 | 8n+2 | 8n+3 | 8n+4 | 8n+5 | 8n+6 | 8n+7 |
|--------------|------|------|------|------|------|------|------|------|
| Data | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |

33.2.5.4 Extended Display List-n Control Register (VI6_DL_EXT_CTRLn: n = 0, 1)

[for n = 0]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[for n = 1]

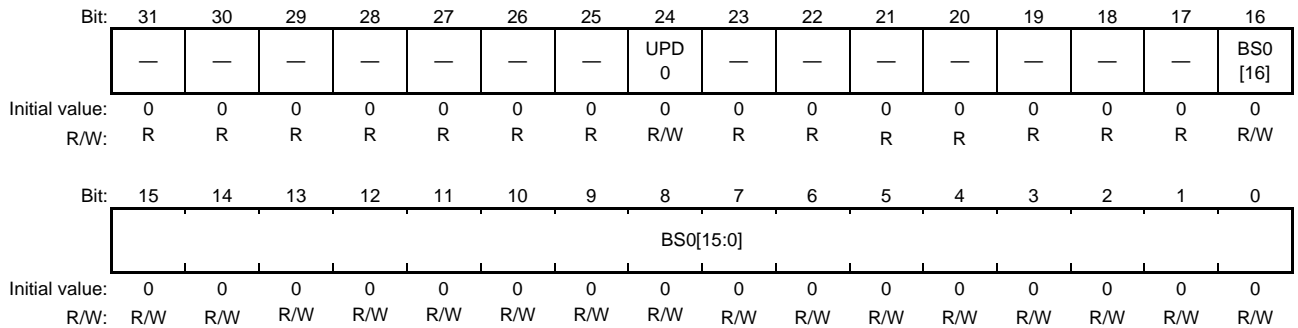
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|--------------|-----|-----|-----|-----|-----|----|-----------|-----------|-----|----|----|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | NWE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | POLINT [5:0] | | | | | — | — | DLP RI | EXP RI | — | — | — | — | EXT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------------------------|--------------|---------------|-----|--|
| 31 to 17, 15, 14, 7, 6, 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | NWE | B'0 | R/W | No Wait for Polling When this bit is set to 1, the polling condition for extended display lists is always assumed to be true. |
| 13 to 8 | POLINT [5:0] | All 0 | R/W | Extended Display List Command Control Always specify 2. |
| 5 | DLPRI | B'0 | R/W | Display List Control 0 Always specify 1. |
| 4 | EXPRI | B'0 | R/W | Display List Control 1 Always specify 0. |
| 0 | EXT | B'0 | R/W | Extended Display List for WPFn Enables or disables the extended display list function. When extended display lists are used, the display list header size is 96 bytes; when they are not used, the header size is 80 bytes. 0: No extended display lists are used 1: Extended display lists are used [Note] When using extended display lists, be sure to also use normal display list mode (VI6_DL_CTRL.DLEn); executing only extended display lists is not possible. When the header-less display list mode is activated, this bit should be set to 0. The extended display list cannot be used with the header-less display list mode. Extended Display List is available only for VSPD/LIF0 or VSPDL/LIFn (n = 0, 1) to realized AUTO-FLD or AUTO-DISP. |

33.2.5.5 Display List Body Size Register-n (VI6_DL_BODY_SIZE_n: n = 0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|-----------|---------------|-----|---|
| 31 to 25, 23 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | UPD0 | B'0 | R/W | Update Flag This bit controls the download of the display list at the next downloading timing in case that the header-less display list mode is used. Set 1 to this bit when display list need to be downloaded for next frame. This bit is automatically cleared to 0 after VI6_DL_HDR_ADDR _n and VI6_DL_BODY_SIZE _n .BS0 (n = 0) is downloaded in H/W side. When this bit is set to 1, the value of VI6_DL_HDR_ADDR _n and VI6_DL_BODY_SIZE _n .BS0 (n = 0) should not be changed. 0: Updating display list for the next frame is not reserved 1: Updating display list for the next frame is reserved |
| 16 to 0 | BS0[16:0] | All 0 | R/W | Header-less Display List Body Size (WPF _n , n = 0) These bits are used for specifying the body size of the display list in case of header-less display list mode. The unit of the size is byte. The value should be set in multiples of 8. |

See section 33.3.6.2 for detail.

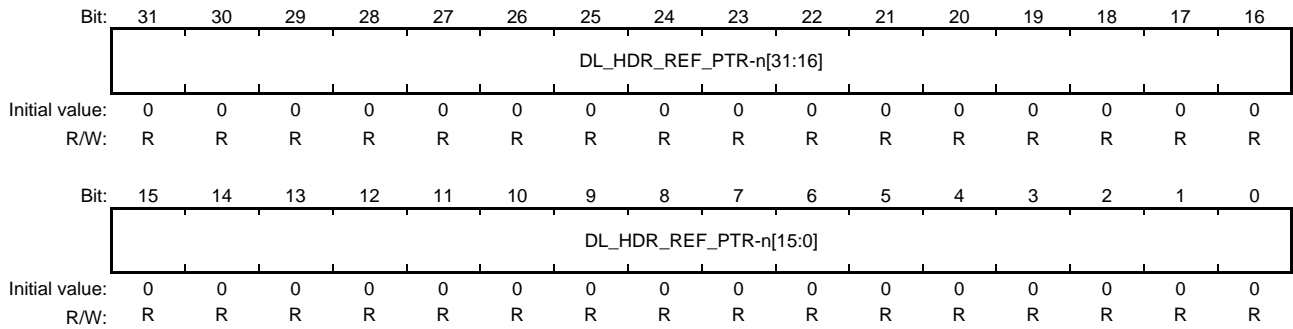
33.2.5.6 Display List-n Header Reference Address Register (VI6_DL_HDR_REF_ADDRn: n = 0, 1)

[for n = 0]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[for n = 1]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

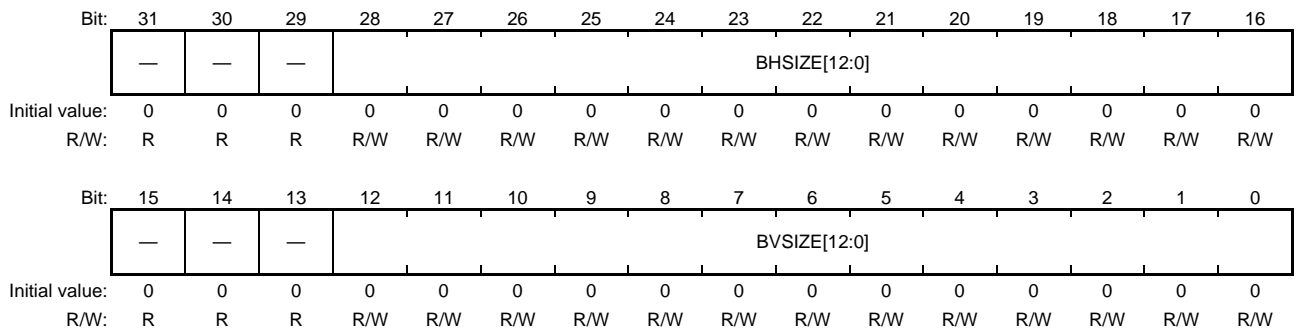


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------------|---------------|-----|--|
| 31 to 0 | DL_HDR_REF_PTR-n[31:0] | All 0 | R | Display List-n reference Header Address Following value is read out from the VI6_DL_HDR_REF_ADDRn for each period. (1) When H/W is reading display list from external memory, header address of the display list referred by VSP2-H/W is read out. (2) When H/W is not reading display list from external memory, the value of VI6_DL_HDR_ADDRn is read out. For details, refer to section 33.3.6.1. |

33.2.6 RPF Control Registers

33.2.6.1 RPFn Basic Read Size Registers (VI6_RPFn_SRC_BSIZE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|---------------|---------------|-----|---|
| 31 to 29, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | BHSIZE [12:0] | All 0 | R/W | Horizontal Size of RPF Basic Read Area These bits specify the horizontal size of the basic area to be read from the external RAM by the RPFn. When the input format is YCbCr4:2:2 or YCbCr4:2:0, specify the size in 2-pixel units. A value from 1 to 8190 can be specified. Specify a value equal to or smaller than the extended read size (VI6_RPFn_SRC_ESIZE.EHSIZE). |
| 12 to 0 | BVSIZE [12:0] | All 0 | R/W | Vertical Size of RPF Basic Read Area These bits specify the vertical size of the basic area to be read from the external RAM by the RPFn. When the input format is YCbCr4:2:0, specify the size in 2-pixel units. A value from 1 to 8190 can be specified. Specify a value equal to or smaller than the extended read size (VI6_RPFn_SRC_ESIZE.EVSIZE). |

Figure 33.12 shows the relationship between the basic read size and extended read size. The RPF reads data from the source memory area specified by the basic read size. The RPF repeats reading the basic read area in the horizontal and vertical directions up to the extended read size and sends the read data to the processing modules in the VSP2.

For basic read size reading, the reading start address, called the RPFn source image storing address, should be specified in VI6_RPFn_SRCM_ADDR_*. In the memory area where the basic read area image is stored, the distance (number of bytes) between addresses for lines n and n + 1 of two-dimensional image data, called the memory stride, should be specified in VI6_RPFn_SRCM_PSTRIDE.

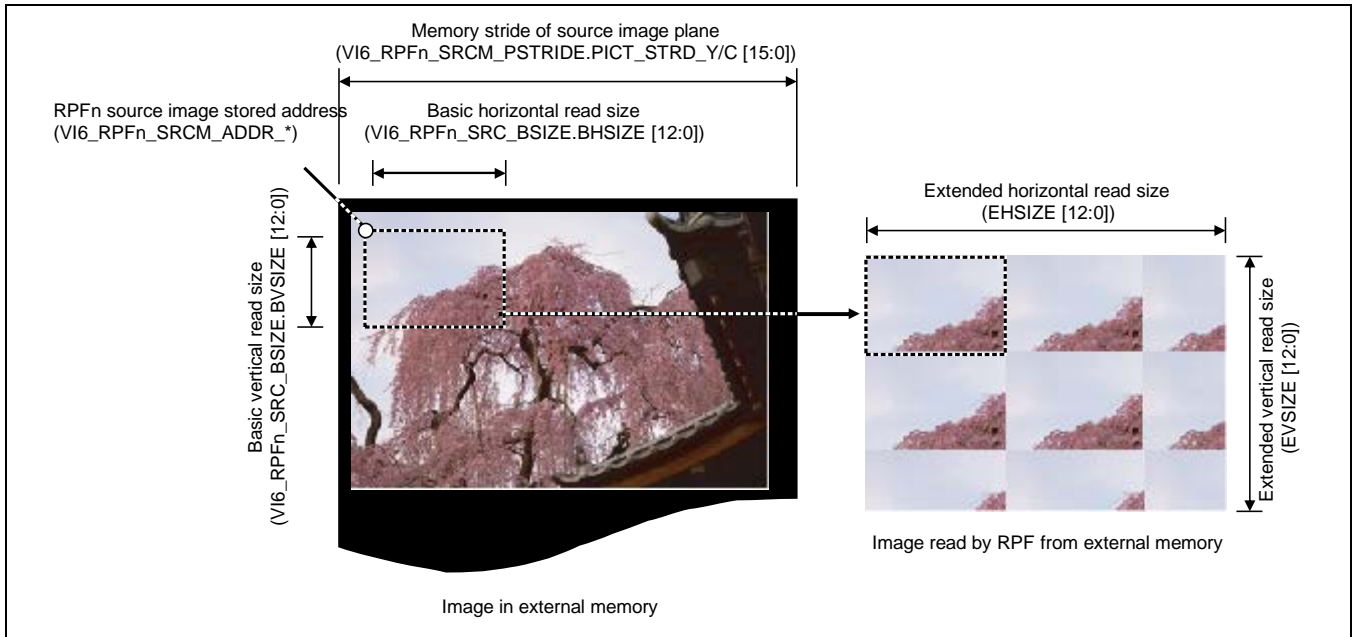


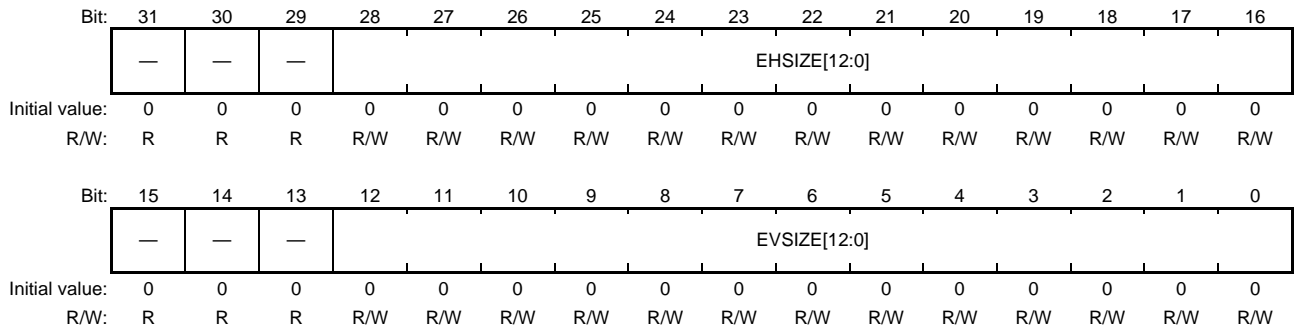
Figure 33.12 Relationship between Basic Read Size and Extended Read Size

Refer also to the following sections.

- Section 33.2.6.2, RPFn Extended Read Size Registers (VI6_RPFn_SRC_ESIZE)
- Section 33.2.6.13, RPFn Source Picture Memory Stride Setting Registers (VI6_RPFn_SRCM_PSTRIDE)
- Section 33.2.6.15, RPFn Source Y/RGB Address Registers (VI6_RPFn_SRCM_ADDR_Y)
- Section 33.2.6.16, RPFn Source Chroma Address Registers 0 (VI6_RPFn_SRCM_ADDR_C0)
- Section 33.2.6.17, RPFn Source Chroma Address Registers 1 (VI6_RPFn_SRCM_ADDR_C1)

33.2.6.2 RPFn Extended Read Size Registers (VI6_RPFn_SRC_ESIZE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|---------------|---------------|-----|--|
| 31 to 29, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | EHSIZE [12:0] | All 0 | R/W | RPF Extended Horizontal Read Size These bits specify the horizontal size of the extended read area to which the RPFn reads data from the external RAM. As shown in Figure 33.12, the basic read area image is repeatedly placed in the extended read area; in the EHSIZE bits, specify a value not smaller than the horizontal size of the basic read area. When the input format is YCbCr4:2:2 or YCbCr4:2:0, specify the size in 2-pixel units (an even value). A value from 1 to 8190 can be specified. |
| 12 to 0 | EVSIZE [12:0] | All 0 | R/W | RPF Extended Vertical Read Size These bits specify the vertical size of the extended read area to which the RPFn reads data from the external RAM. As shown in Figure 33.12, the basic read area image is repeatedly placed in the extended read area; in the EVSIZE bits, specify a value not smaller than the vertical size of the basic read area. When the input format is YCbCr4:2:0, specify the size in 2-pixel units (an even value). A value from 1 to 8190 can be specified. |

VI6_RPFn_SRC_ESIZE specifies the extended size for RPFn. The extended horizontal and vertical sizes should be equal to or greater than the basic sizes specified in VI6_RPFn_SRC_BSIZE. The RPF internal data processing described later and image processing described in section 33.2.8 and later sections are all applied to the image in the extended read size shown on the right side in Figure 33.12.

33.2.6.3 RPFn Input Format Registers (VI6_RPFn_INFMT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------|-----------|-----------|-----------|-----|-----|-----|------------|----|-----|-----|-----|-----|-----|-----|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | VIR | — | — | — | — | — | — | — | — | — | — | — | CIP M |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SPYCS | SPU VS | CEXT[1:0] | RDTM[2:0] | | CSC | — | RDFMT[6:0] | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----------------------|----------|---------------|-----|--|
| 31 to 29, 27 to 17, 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | VIR | B'0 | R/W | Virtual Input Enable Enables or disables the virtual input function of the RPFn. The image to be processed by the RPFn is usually read from the external memory by the MAU. Instead of this input, the virtual input function generates a single-color image within the RPFn and sends it to the modules in VSP2. When the virtual input function is enabled, the fixed value specified in VI6_RPFn_VRTCOL_SET is used as the input to the RPFn. While the virtual input function is enabled, data is not read from the external memory; that is, the α plane is not read and the IROP calculation thus cannot be executed. In this case, set VI6_RPFn_ALPH_SEL.ASEL to 4. Neither the color space conversion through CSC nor the color keying described in section 33.2.6.11 can be used. 0: RPFn uses general input. 1: RPFn uses virtual input. |
| 16 | CIPM | B'0 | R/W | Horizontal Chrominance Interpolation Method Setting Image data is processed in the YCbCr444 format inside VSP2 in case of YCbCr color space. When the chrominance format of the input image is YCbCr422 or YCbCr420, data is up-sampled as shown in Figure 33.13 for internal processing. This bit specifies the method of up-sampling for this purpose. 0: The nearest-neighbor method is used for horizontal chrominance interpolation. 1: The bilinear method is used for horizontal chrominance interpolation. |
| 15 | SPYCS | B'0 | R/W | RPF Input Mode Setting 1 When the input format is YUY2, set this bit to 1 and set the RDFMT bits to 71 (H'47). When the input format is YVYU, set this bit and the SPUVS bit to 1 and set the RDFMT bits to 71 (H'47). In other cases, set this bit to 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 14 | SPUVS | B'0 | R/W | <p>RPF Input Mode Setting 2</p> <p>When the input format is NV61, set this bit to 1 and set the RDFMT bits to 65 (H'41).</p> <p>When the input format is NV21, set this bit to 1 and set the RDFMT bits to 66 (H'42).</p> <p>When the input format is YVYU, set this bit and the SPYCS bit to 1 and set the RDFMT bits to 71 (H'47).</p> <p>In other cases, set this bit to 0.</p> |
| 13, 12 | CEXT [1:0] | B'00 | R/W | <p>Lower-Bit Color Data Extension Method Setting</p> <p>When an RGB input format where each color component is expressed in less than 8 bits are selected from Table 33.10 through the RDFMT bits, VSP2 internally extends each color component to 8 bits before using the data. These bits select this extension method.</p> <p>B'00: Lower-order bits of color data are extended with 0.</p> <p>B'01: Upper-order bits of color data are copied to the lower-order bits.</p> <p>B'10: Lower-order bits of color data are extended with 0. The maximum value is limited to H'FF.</p> <p>B'11: Setting prohibited</p> |
| 11 to 9 | RDTM [2:0] | B'000 | R/W | <p>CSC Conversion Expression Setting</p> <p>These bits select the expression used for color space conversion. The conversion direction is RGB → YCbCr when RGB is selected through the RDFMT bits; the direction is YCbCr → RGB when YCbCr is selected.</p> <p>B'000: BT.601 YCbCr [16,235/240] ↔ RGB [0,255]</p> <p>B'001: BT.601 YCbCr [0,255] ↔ RGB [0,255]</p> <p>B'010: BT.709 YCbCr [16,235/240] ↔ RGB [0,255]</p> <p>B'011: BT.709 YCbCr [16,235/240] ↔ RGB [16,235]</p> <p>B'100 to B'111: Setting prohibited</p> |
| 8 | CSC | B'0 | R/W | <p>Color Space Conversion Enable</p> <p>Enables or disables color space conversion between YCbCr and RGB to be executed in RPFn. The characteristics of color space conversion are determined by the RDTM bit setting. (*1)</p> <p>When using the virtual input (VIR = 1), specify 0.</p> <p>0: Color space conversion is disabled.</p> <p>1: Color space conversion is enabled.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------|---------------|-----|--|
| 6 to 0 | RDFMT [6:0] | All 0 | R/W | <p>RPF Input Image Format Setting</p> <p>These bits select the format of the image input from the external RAM to the RPFn. Select a value corresponding to the desired format from those shown in Table 33.10 and Table 33.11.</p> <p>When the virtual input function is used (VIR = 1), the color information for the virtual input should be specified in VI6_RPFn_VRTCOL_SET. If this color information is in the RGB format, set the RDFMT bits to 19. If the color information is in the YCbCr format, set these bits to 64.</p> <p>[Note 1] Number of input pixels</p> <p>When YCbCr4:2:2 is selected through the RDFMT bits, the horizontal size of the input image should be specified in 2-pixel units. When YCbCr4:2:0 is selected, the vertical and horizontal sizes should be specified in 2-pixel units. Observe these restrictions when specifying the image size in VI6_RPFn_SRC_BSIZE and VI6_RPFn_SRC_ESIZE.</p> <p>[Note 2] OSD-CLUT Setting</p> <p>When the RDFMT bits are set to H'3F, RGB color data should be stored in the OSD-CLUT. When these bits are set to H'7F, YCbCr color data should be stored in the OSD-CLUT. Note that while the target WPF is operating, the RPF is also operating and the OSD-CLUT cannot be read or written to.</p> |

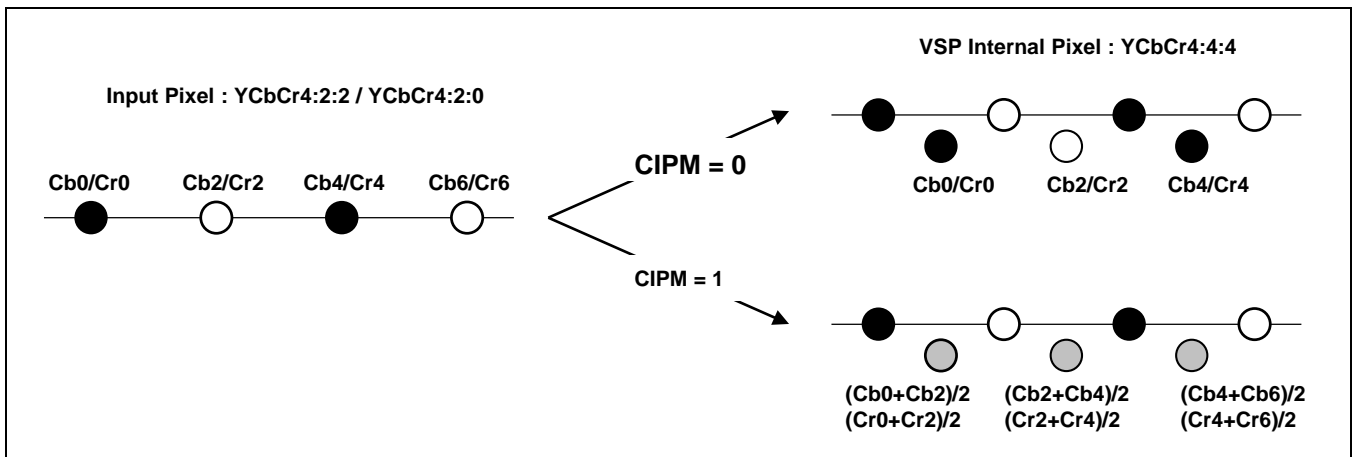


Figure 33.13 Chrominance Interpolation Methods Selectable through CIPM Setting

Note: * Note on color space settings

The color space for the image output from the RPF to VSP2 internal modules is determined by the combination of the color space for the image input to the RPF, which is selected through the VI6_RPFn_INFMT.RDFMT setting, and the enabled or disabled state of the color space conversion function, which is selected through the VI6_RPFn_INFMT.CSC setting (Table 33.9). For example, when the image input to the RPF is in the YCbCr format, the RPF outputs data to VSP2 internal modules in the YCbCr format if color space conversion is disabled through the CSC bit, and the RPF outputs data in the RGB format if color space conversion is enabled. When the image input to the RPF is in the RGB format, the relationship between the output format and the color space conversion setting is the opposite of the YCbCr case. For some VSP2 internal modules, the YCbCr format is recommended for image processing because of the characteristics of the processing, or the same color space needs to be specified between multiple RPF outputs. In these cases, set VI6_RPFn_INFMT.RDFMT and VI6_RPFn_INFMT.CSC appropriately so that the RPFs can output the required color space according to the color space conditions described above.

Table 33.9 RPFn Input Color Space and Output Color Space

| RPFn Input Color Space (VI6_RPFn_INFMT.RDFMT) | | Color Space Conversion Setting (VI6_RPFn_INFMT.CSC) | | RPFn Output Color Space |
|--|-----------------|---|-------|----------------------------|
| RGB | (H'00 to H'3F)* | Disabled | (0) * | RGB |
| | | Enabled | (1) * | YCbCr |
| YCbCr | (H'40 to H'7F)* | Disabled | (0) * | YCbCr |
| | | Enabled | (1) * | RGB |

Note: * Value specified in the register

A color space conversion function equivalent to that in the RPFn is also provided by the WPF. As shown in Table 33.9, the color space (YCbCr or RGB) output from the RPF becomes the input format for the WPF. Here, the color space of the output image obtained by the color space conversion function of the WPF must match the color space of the format specified through VI6_WPFn_OUTFMT.WRFMT.

Figure 33.14 shows the relationship between the input/output format and color space. The input color space for the RPF is determined when the input image format for the RPF is specified through the RDFMT bits. The color space for the image output from the RPF to subsequent VSP2 internal modules depends on the combination of the RPF input format and CSC (color space conversion function) enabled or disabled state in the RPF as shown in Table 33.9 and Figure 33.14. The user should first determine whether the image processing in the VSP2 is done in YCbCr or RGB, and then specify the RPF input format and CSC enabled or disabled state to obtain the desired color space. The color space of the RPF output image is also that of the WPF input image; the color space of the data output from the WPF to the outside of VSP2 depends on the combination of the WPF input color space and the enabled or disabled state of the CSC implemented in the WPF as shown in Figure 33.14. The color space of the WPF output image must match that of the WPF output format (determined by VI6_WPFn_OUTFMT.WRFMT). For example, in the flow shown in Figure 33.14, YCbCr should not be specified as the WPF output format regardless of the fact that the color space of the WPF output image is in RGB format.

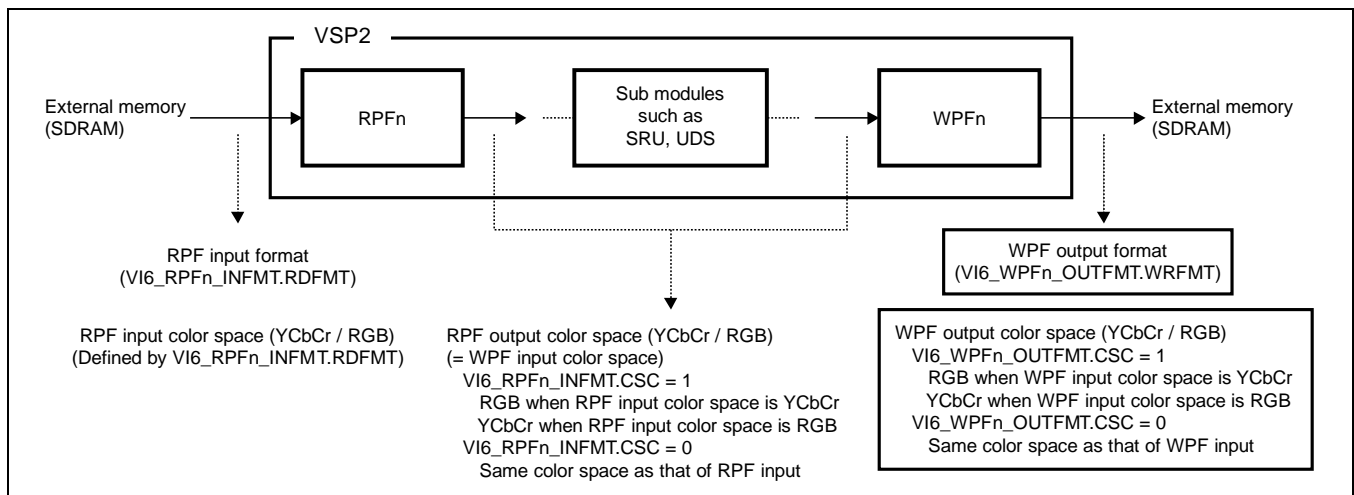


Figure 33.14 Relationship between Input/output Format and Color Space

Table 33.11 Packed YCbCr Formats for RPF Input

| RDFMT [6:0] | Packed YCbCr Input Format | Reference |
|--------------|--|----------------|
| H'40 | YCbCr4:4:4 semi-planar | Figure 33.15*6 |
| H'41 | YCbCr4:2:2 semi-planar (NV16, NV61*1) | |
| H'42 | YCbCr4:2:0 semi-planar (NV12, NV21*1) *7 | |
| H'43 to H'45 | Reserved | — |
| H'46 | YCbCr4:4:4 interleaved | Figure 33.16*6 |
| H'47 | YCbCr4:2:2 interleaved type 0 (UYVY, YUY2*2, YVYU*3) | |
| H'48 | YCbCr4:2:2 interleaved type 1 | |
| H'49 | YCbCr4:2:0 interleaved*8 | |
| H'4A | YCbCr4:4:4 planar | Figure 33.17*6 |
| H'4B | YCbCr4:2:2 planar (YV16) | |
| H'4C | YCbCr4:2:0 planar (YV12, I420) *7 | |
| H'4D to H'7E | Reserved | — |
| H'7F | YCBCR_CLUT_DATA*4 *5 | — |

- Notes:
1. When the input format is NV61 or NV21, set the SPUVS bit to 1.
 2. When the input format is YUY2, set the SPYCS bit to 1.
 3. When the input format is YVYU, set the SPUVS bit to 1 and SPYCS bit to 1.
 4. When the RDFMT [6:0] bits are set to H'7F, it is necessary to set up the OSD-CLUT in the YCbCr color space.
 5. The Monochrome format can be implemented through the OSD-CLUT.
 6. Figure 33.18 shows the definition of memory address for each pixel in Figure 33.15 to Figure 33.17.
 7. Each line of C plane is read twice, so byte/pixel of YCbCr420 is 2 byte/pixel (same as YCbCr422).
 8. Each line of plane is read twice, so byte/pixel of YCbCr420ITL is 3 byte/pixel (same as YCbCr444ITL).

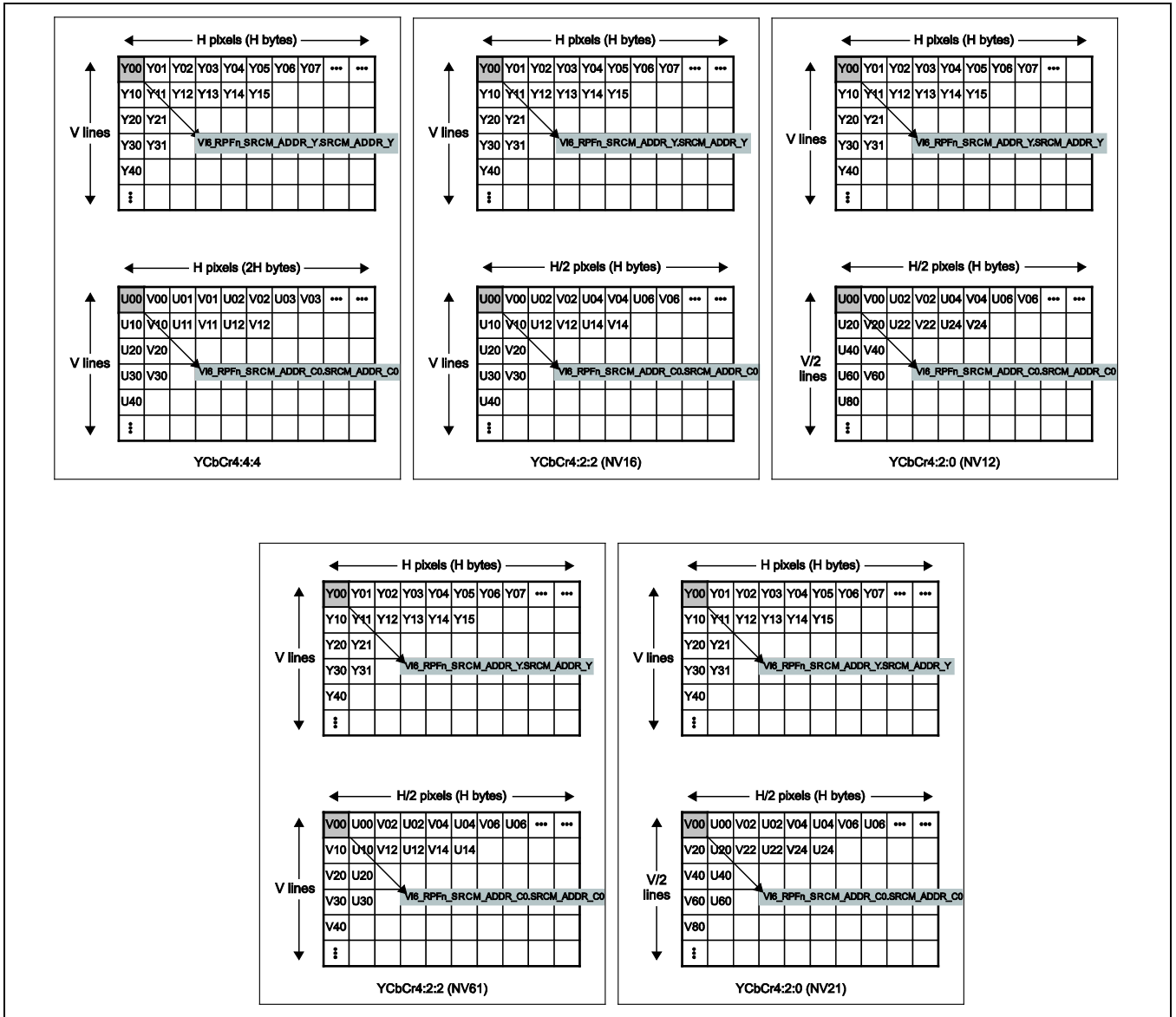


Figure 33.15 YCbCr Semi-Planar Formats

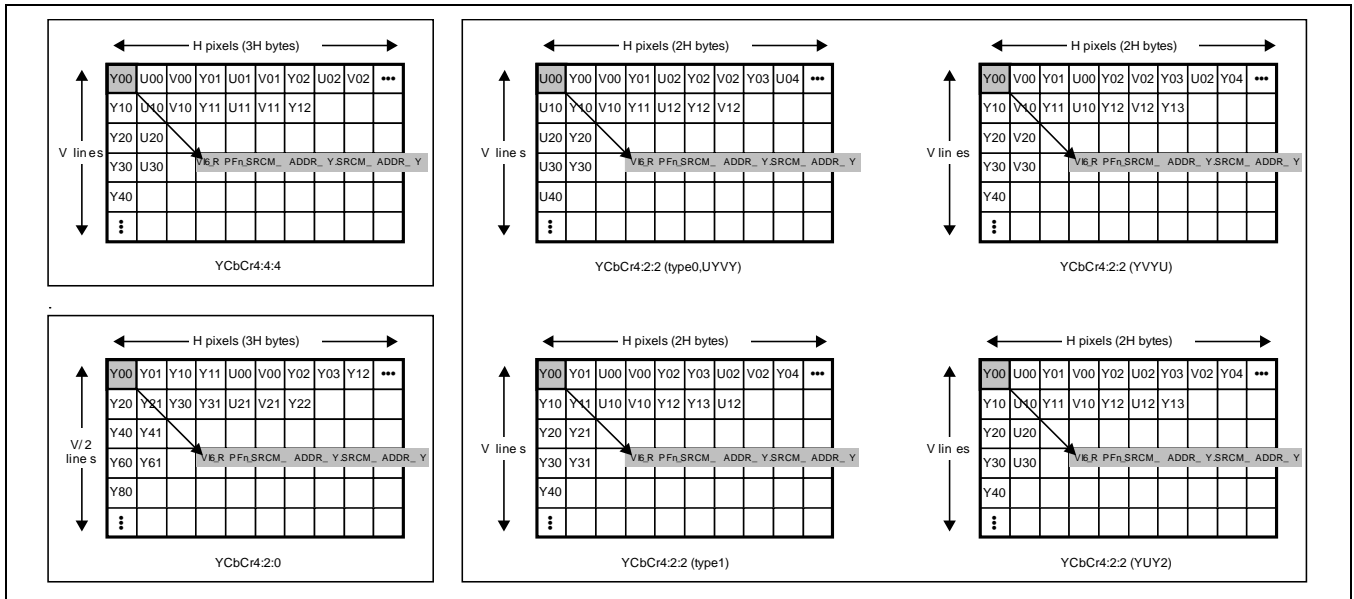


Figure 33.16 YCbCr Interleaved Formats

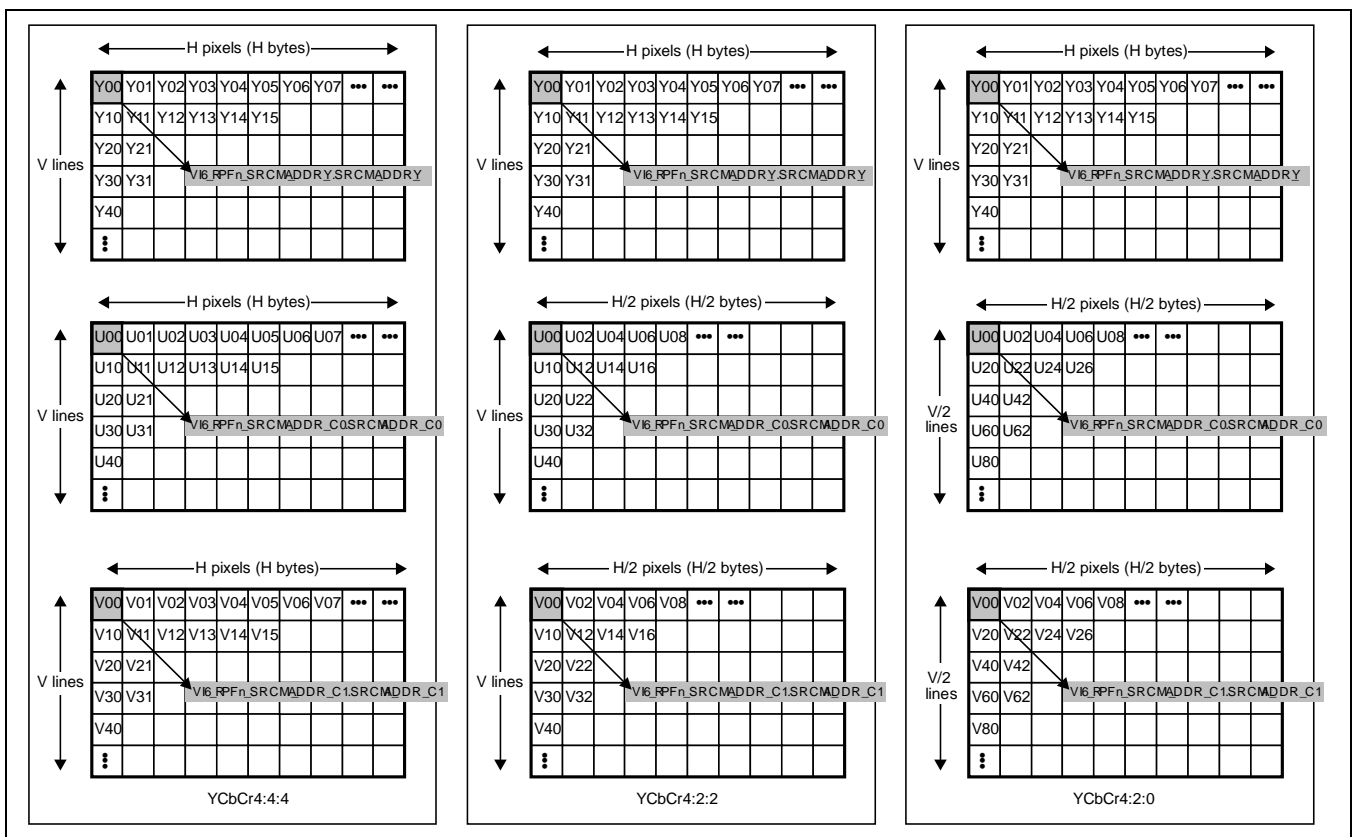


Figure 33.17 YCbCr Planar Formats

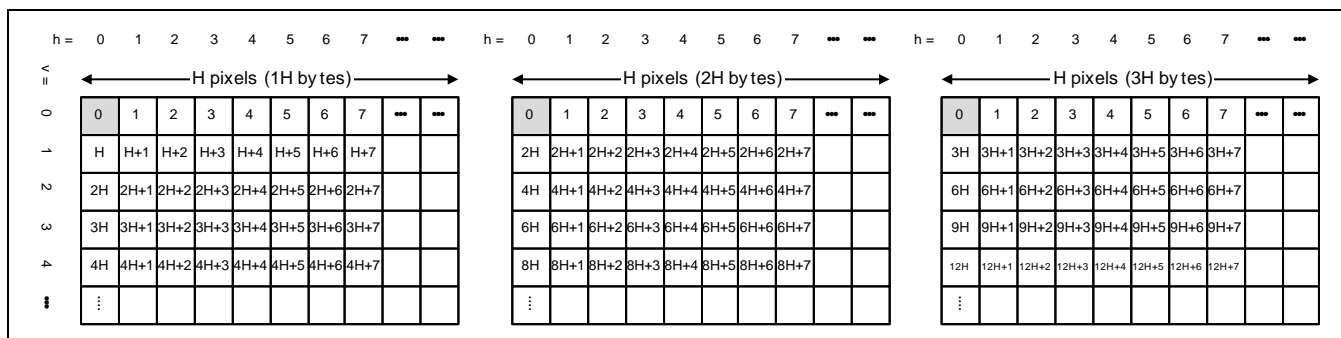


Figure 33.18 Memory address for the corresponding pixel's position

Legend of Figure 33.18

- H is horizontal image size in pixel unit.
- 1H, 2H and 3H is corresponding to the case 1 pixel has 1byte, 2bytes, 3bytes of data respectively.

[OSD-CLUT Setting]

To set up the OSD-CLUT, store the values to replace each of R (Cr), G (Y), and B (Cb) components in the OSD-CLUT allocated as shown in Table 33.12. The OSD-CLUT space has 256 32-bit entries, which should be accessed in 32 bits in the same way as for register access. The address for each entry is obtained from the start address of each space (Table 33.12) + entry number × 4. For example, the OSD-CLUT address of entry 7 for RPF2 is (offset address) + H'481C. Figure 33.19 shows the OSD-CLUT format.

Table 33.12 OSD-CLUT Space for Each RPF

| RPF | Address Area for CLUT Space |
|------|--|
| RPF0 | (offset address) + H'4000 to (offset address) + H'43FF |
| RPF1 | (offset address) + H'4400 to (offset address) + H'47FF |
| RPF2 | (offset address) + H'4800 to (offset address) + H'4BFF |

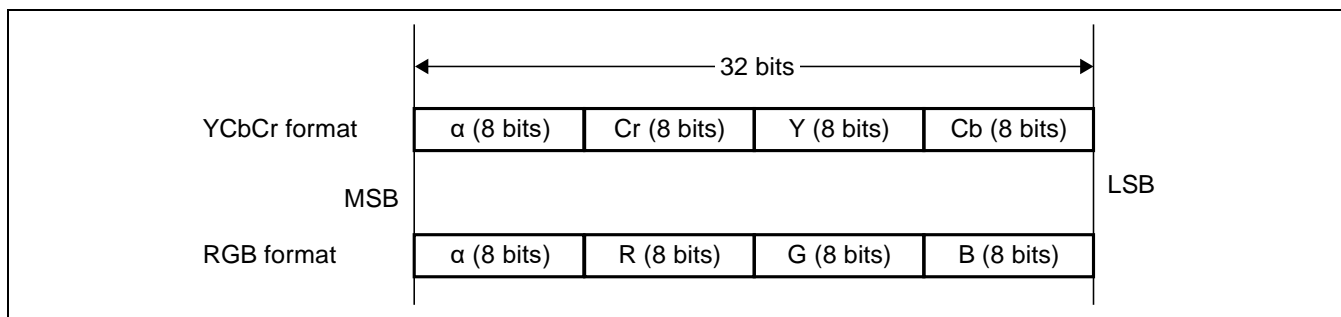


Figure 33.19 OSD-CLUT Format

According to the specified 256-entry replacement table (OSD-CLUT), input data is replaced as shown in Figure 33.20. Input data is 8 bits but output data is 32 bits including α and RGB or YCbCr components.

When writing to an entry in the OSD-CLUT, the host CPU writes all components at the same time. Therefore, specify replacement data for all components in the OSD-CLUT at the same time as shown in Figure 33.19.

Note that write access to the OSD-CLUT space is prohibited while the RPF is operating.

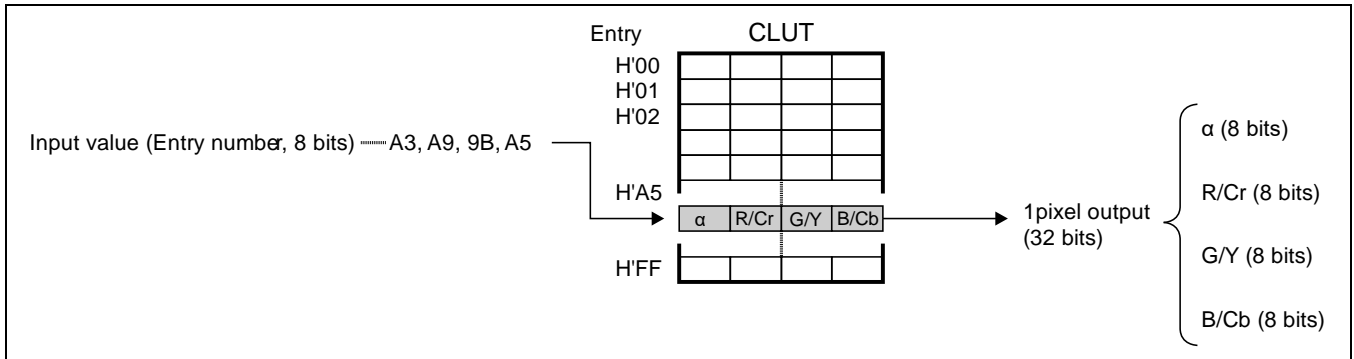


Figure 33.20 Replacement of Input Data with Pixel Values Specified in OSD-CLUT

33.2.6.4 RPFn Data Swapping Registers (VI6_RPFn_DSWAP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-------|-------|-------|-------|----|----|----|----|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | A_LLS | A_LWS | A_WDS | A_BTS | — | — | — | — | P_LLS | P_LWS | P_WDS | P_BTS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|------------------|----------|---------------|-----|---|
| 31 to 12, 7 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 | A_LLS | B'0 | R/W | α Plane Data Swapping in LONG LWORD Units The effect of this bit setting is defined in Table 33.13. 0: Data swapping in LONG LWORD (64-bit) units is disabled 1: Data swapping in LONG LWORD (64-bit) units is enabled |
| 10 | A_LWS | B'0 | R/W | α Plane Data Swapping in long word Units The effect of this bit setting is defined in Table 33.13. 0: Data swapping in long word (32-bit) units is disabled 1: Data swapping in long word (32-bit) units is enabled |
| 9 | A_WDS | B'0 | R/W | α Plane Data Swapping in Word Units The effect of this bit setting is defined in Table 33.13. 0: Data swapping in word (16-bit) units is disabled 1: Data swapping in word (16-bit) units is enabled |
| 8 | A_BTS | B'0 | R/W | α Plane Data Swapping in Byte Units The effect of this bit setting is defined in Table 33.13. 0: Data swapping in byte (8-bit) units is disabled 1: Data swapping in byte (8-bit) units is enabled |
| 3 | P_LLS | B'0 | R/W | Picture Plane Data Swapping in LONG LWORD Units The effect of this bit setting is defined in Table 33.13. 0: Data swapping in LONG LWORD (64-bit) units is disabled 1: Data swapping in LONG LWORD (64-bit) units is enabled |
| 2 | P_LWS | B'0 | R/W | Picture Plane Data Swapping in long word Units The effect of this bit setting is defined in Table 33.13. 0: Data swapping in long word (32-bit) units is disabled 1: Data swapping in long word (32-bit) units is enabled |
| 1 | P_WDS | B'0 | R/W | Picture Plane Data Swapping in Word Units The effect of this bit setting is defined in Table 33.13. 0: Data swapping in word (16-bit) units is disabled 1: Data swapping in word (16-bit) units is enabled |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | P_BTS | B'0 | R/W | Picture Plane Data Swapping in Byte Units The effect of this bit setting is defined in Table 33.13. 0: Data swapping in byte (8-bit) units is disabled 1: Data swapping in byte (8-bit) units is enabled |

When the virtual input function of the RPFn is used (VI6_RPFn_INFMT.VIR = 1), this register setting is ignored. Swapping of RPF input data can be specified separately for the α plane and picture plane.

Table 33.13 shows the data order before and after swapping according to the long long word, long word, word, and byte swapping settings.

When data order in memory for each format is the same as Table 33.10 for RGB format and Figure 33.15 to Figure 33.17 for YCbCr format, set 4'b1111 to {*_LLS, *_LWS, *_WDS, *_BTS}. If data order is not the same as the definition, change data order within 16byte unit by these bits as shown in Table 33.13.

Table 33.13 Changing data order according to swap register

Data order in memory

| Byte address | 16n+0 | 16n+1 | 16n+2 | 16n+3 | 16n+4 | 16n+5 | 16n+6 | 16n+7 | 16n+8 | 16n+9 | 16n+10 | 16n+11 | 16n+12 | 16n+13 | 16n+14 | 16n+15 | *_L LS | *_LW S | *_W DS | *_BT S |
|--------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| Data 0 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 3 | 2 | 5 | 4 | 7 | 6 | 9 | 8 | 11 | 10 | 13 | 12 | 15 | 14 | 1 | 1 | 1 | 0 |
| 2 | 2 | 3 | 0 | 1 | 6 | 7 | 4 | 5 | 10 | 11 | 8 | 9 | 14 | 15 | 12 | 13 | 1 | 1 | 0 | 1 |
| 3 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 11 | 10 | 9 | 8 | 15 | 14 | 13 | 12 | 1 | 1 | 0 | 0 |
| 4 | 4 | 5 | 6 | 7 | 0 | 1 | 2 | 3 | 12 | 13 | 14 | 15 | 8 | 9 | 10 | 11 | 1 | 0 | 1 | 1 |
| 5 | 5 | 4 | 7 | 6 | 1 | 0 | 3 | 2 | 13 | 12 | 15 | 14 | 9 | 8 | 11 | 10 | 1 | 0 | 1 | 0 |
| 6 | 6 | 7 | 4 | 5 | 2 | 3 | 0 | 1 | 14 | 15 | 12 | 13 | 10 | 11 | 8 | 9 | 1 | 0 | 0 | 1 |
| 7 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 1 | 0 | 0 | 0 |
| 8 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 0 | 1 | 1 | 1 |
| 9 | 9 | 8 | 11 | 10 | 13 | 12 | 15 | 14 | 1 | 0 | 3 | 2 | 5 | 4 | 7 | 6 | 0 | 1 | 1 | 0 |
| 10 | 10 | 11 | 8 | 9 | 14 | 15 | 12 | 13 | 2 | 3 | 0 | 1 | 6 | 7 | 4 | 5 | 0 | 1 | 0 | 1 |
| 11 | 11 | 10 | 9 | 8 | 15 | 14 | 13 | 12 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 0 | 1 | 0 | 0 |
| 12 | 12 | 13 | 14 | 15 | 8 | 9 | 10 | 11 | 4 | 5 | 6 | 7 | 0 | 1 | 2 | 3 | 0 | 0 | 1 | 1 |
| 13 | 13 | 12 | 15 | 14 | 9 | 8 | 11 | 10 | 5 | 4 | 7 | 6 | 1 | 0 | 3 | 2 | 0 | 0 | 1 | 0 |
| 14 | 14 | 15 | 12 | 13 | 10 | 11 | 8 | 9 | 6 | 7 | 4 | 5 | 2 | 3 | 0 | 1 | 0 | 0 | 0 | 1 |
| 15 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 0 | 0 | 0 | 0 |



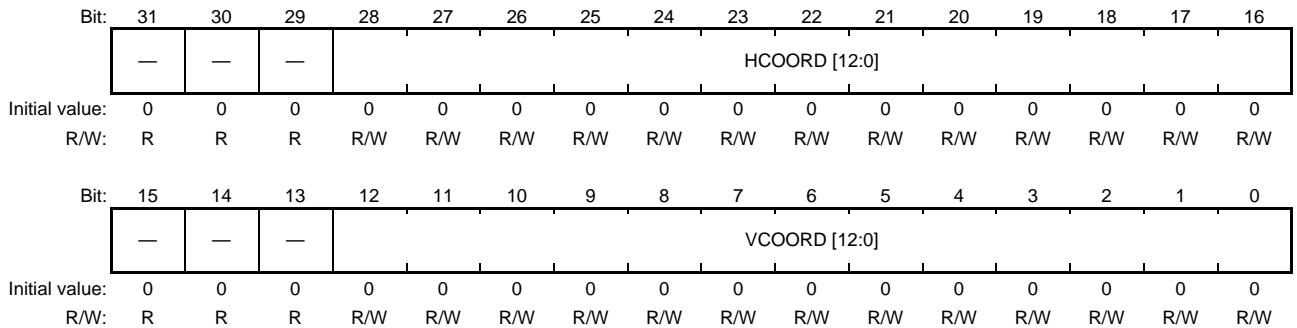
Data order defined in Table 33.10 for RGB format and Figure 33.15 to Figure 33.17 for YCbCr format

| Byte address | 16n+0 | 16n+1 | 16n+2 | 16n+3 | 16n+4 | 16n+5 | 16n+6 | 16n+7 | 16n+8 | 16n+9 | 16n+10 | 16n+11 | 16n+12 | 16n+13 | 16n+14 | 16n+15 |
|--------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|--------|--------|--------|--------|
| Data | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |

Note: When reading data is FCNL compressed data, follow section 35 FCP User's Manual about setting this swap register.

33.2.6.5 RPFn Display Location Registers (VI6_RPFn_LOC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|---------------|---------------|-----|--|
| 31 to 29, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | HCOORD [12:0] | All 0 | R/W | Horizontal Coordinate of Sublayer Display Location on Master Layer These bits specify the left-end location of the sublayer displayed by the RPFn and the subsequent module connected through the DPR. Specify the horizontal coordinate of the location in pixel units with the left-end pixel of the master layer set at coordinate 0. When the RPFn is the master layer, set these bits to 0. If the sublayer extends beyond the master layer according to the HCOORD setting, the extended section is cut off at the right end of the master layer. Even in this case, however, a bus transfer that is unnecessary for output image generation is executed since the whole sublayer data is read from the external memory. Appropriate coordinate setting is required so that the sublayer does not extend beyond the right end of the master layer. A value from 0 to 8189 can be specified. |
| 12 to 0 | VCOORD [12:0] | All 0 | R/W | Vertical Coordinate of Sublayer Display Location on Master Layer These bits specify the top-end location of the sublayer displayed by the RPFn and the subsequent module connected through the DPR. Specify the vertical coordinate of the location in pixel units with the top-end pixel of the master layer set at coordinate 0. When the RPFn is the master layer, set these bits to 0. If the sublayer extends beyond the master layer according to the VCOORD setting, the extended section is cut off at the bottom end of the master layer. Even in this case, however, a bus transfer that is unnecessary for output image generation is executed since the whole sublayer data is read from the external memory. Appropriate coordinate setting is required so that the sublayer does not extend beyond the bottom end of the master layer. A value from 0 to 8189 can be specified. |

Figure 33.21 shows an example of RPF1 and RPF2 offsets with respect to master layer RPF0. Although this figure only shows sublayers RPF1 and RPF2, specify offsets for all RPFs other than the master layer in the same way as shown in this example.

Whether an RPFn is the master layer or a sublayer is determined through the selection of the source RPF for WPFn (the VI6_WPFn_SRCRPF setting). For details, refer to section 33.2.7.1.

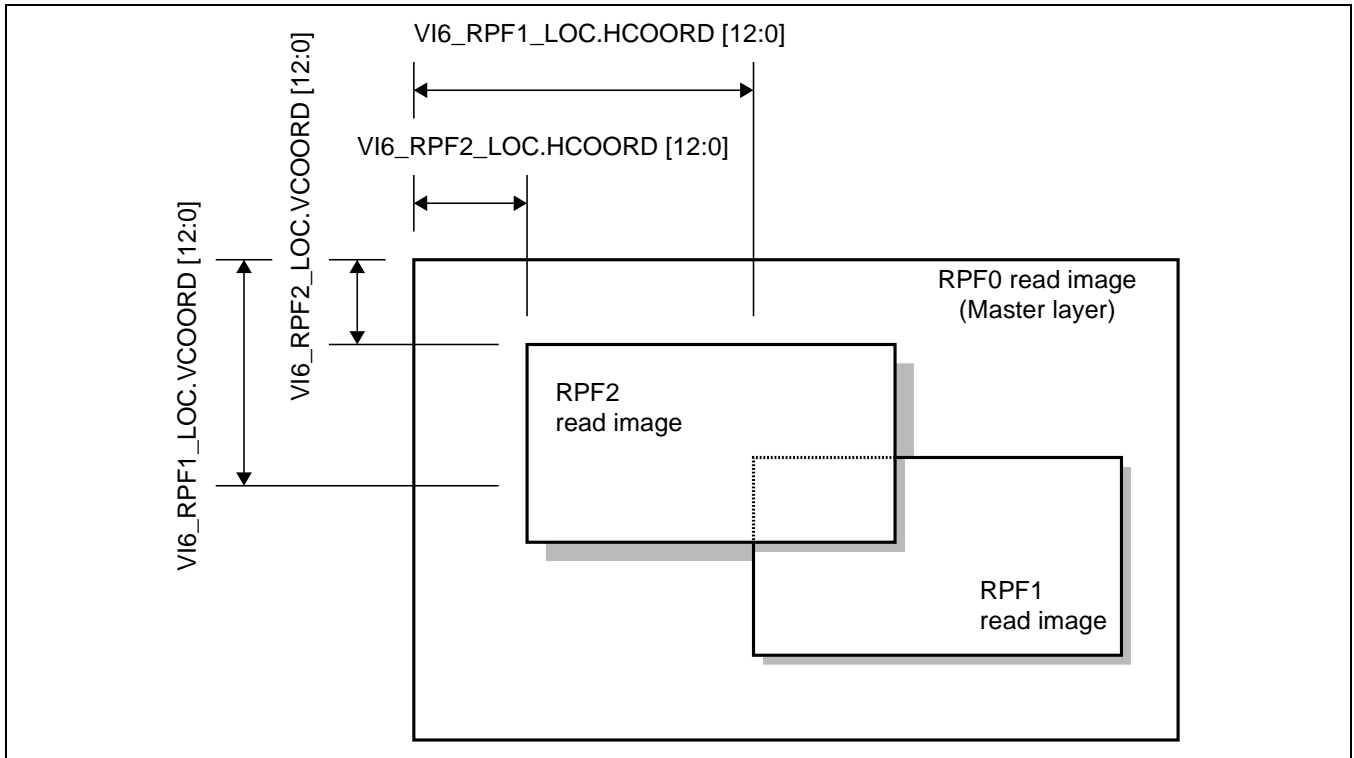


Figure 33.21 RPF1 and RPF2 Offsets from Master Layer

33.2.6.6 RPFn α Plane Selection Control Registers (VI6_RPFn_ALPH_SEL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------|-----------|-----|-----|-----------|-----|-----|-----|-------------|-----|-----|-----|-----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | ASEL[2:0] | | | IROP[3:0] | | | | BSEL | — | — | — | AEXT[1:0] | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | ALPHA1[7:0] | | | | | | | | ALPHA0[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------------------|-----------|---------------|-----|---|
| 31, 22 to 20, 17, 16 | — | All 0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 30 to 28 | ASEL[2:0] | B'000 | R/W | <p>α Format and Processing Method Select</p> <p>These bits select how to handle the α value to be used. The RPF handles two types of α value; 8-bit and 1-bit values. When a 1-bit α value is used, VSP2 assumes that the 1-bpp α value for each pixel is stored in the order from MSB to LSB in each byte (big endian).</p> <p>The α value is used as either transparency information or mask information. Transparency information is included in the α plane read from external memory when the ASEL bits are set to 1 or 3 and in the α value stored in the packed RGB bit field when these bits are set to 0 or 2. The α value as transparency information is sent as the destination value to the IROP as shown in Figure 33.22 and then output to the subsequent modules. The output α value is used, for example, for blending in the BRU.</p> <p>The α value as mask information is used for IROP operation in the RPF. The mask information is included in the α plane read from external RAM when the ASEL bits are set to 0 or 2 and the source value is used in IROP operation (IROP setting other than 0, 5, 10, or 15). This α value is sent as the source value to the IROP as shown in Figure 33.22.</p> <p>Note that the α value selected through the ASEL bits has a lower priority than the VI6_RPFn_CKEY_SET*.AP* value replaced through the color keying function. When the color keying function is used, the α value may be replaced with the VI6_RPFn_CKEY_SET*.AP* value regardless of the ASEL bit setting.</p> <p>When using the virtual input function (VI6_RPFn_INFMT.VIR = 1), specify 4.</p> <p>0: 1, 4, or 8-bit packed α + plane α (IROP != 0, 5, 10, 15)</p> <p>The α bit field in 1, 4, or 8-bit packed α is handled as transparency information. Be sure to specify the packed format that includes α through VI6_RPFn_INFMT.RDFMT.</p> <p>When VI6_RPFn_MSKCTRL.MSK_EN is 0 and the IROP bit value is not 0, 5, 10, or 15, the α plane should be read as mask information. Specify the number of α data bits (BSEL) stored in the α plane and the α plane read start address (VI6_RPFn_SRCM_ADDR_AI). When the IROP bits are set to 0, 5, 10, or 15, the α plane is not read.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 30 to 28 | ASEL[2:0] | B'000 | R/W | <p>B'001: 8-bit plane α</p> <p>The 8-bit α plane is read from external RAM as transparency information. When the packed RGB format has a bit field for α, the information in the α bit field is discarded. The α plane read start address (VI6_RPFn_SRCM_ADDR_AI) should be specified. The α value goes through the 8-bit transparent α generator shown in Figure 33.22 without change.</p> <p>When VI6_RPFn_MSKCTRL.MSK_EN is 0, IROP operation cannot be executed; set the IROP bits to 0 in this case. When VI6_RPFn_MSKCTRL.MSK_EN is 1, IROP operation can be executed.</p> <p>B'010: 1-bit packed α + plane α (IROP ! = 0, 5, 10, 15)</p> <p>The 1-bit packed α input is converted by the 8-bit transparent α generator shown in Figure 33.22 according to the ALPHA0/1 setting into the 8-bit α value as transparency information. Select the packed input format that includes a 1-bit α field.</p> <p>When VI6_RPFn_MSKCTRL.MSK_EN is 0 and the IROP value is not 0, 5, 10, or 15, the α plane should be read as mask information. Specify the number of α data bits (BSEL) stored in the α plane and the α plane read start address (VI6_RPFn_SRCM_ADDR_AI). When the IROP bits are set to 0, 5, 10, or 15, the α plane is not read.</p> <p>B'011: 1-bit plane α</p> <p>The 1-bit α plane is read from external RAM and converted by the 8-bit transparent α generator shown in Figure 33.22 according to the ALPHA0/1 setting into the 8-bit α value as transparency information. When the packed RGB format has a bit field for α, the information in the α bit field is discarded. The α plane read start address (VI6_RPFn_SRCM_ADDR_AI) should be specified.</p> <p>When VI6_RPFn_MSKCTRL.MSK_EN is 0, IROP operation cannot be executed; set the IROP bits to 0 in this case. When VI6_RPFn_MSKCTRL.MSK_EN is 1, IROP operation can be executed.</p> <p>B'100: Fixed α</p> <p>The fixed α value (VI6_RPFn_VRTCOL_SET.LAYA value) is output from the RPF. IROP operation cannot be executed; set the IROP bits to 0 in this case.</p> <p>B'101 to B'111: Setting prohibited.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 27 to 24 | IROP[3:0] | H'0 | R/W | <p>IROP Operation Setting</p> <p>These bits specify the operator to be executed in the IROP operation unit shown in Figure 33.22. The source (S) for the IROP operation is the pixel data and α data specified in the VI6_RPFn_MSKSET0 or VI6_RPFn_MSKSET1 IROP input value register, which is selected according to the value (0 or 1) generated by the 1-bit mask generator. The destination (D) is the image data (RGB/YCbCr) and 8-bit α data output from the unpack/OSD processor. IROP operation is applied both for the image data and α data between the source and destination data.</p> <p>If these bits are set to the operation that involves the source (S) (IROP setting other than 0, 5, 10, or 15) while VI6_RPFn_MSKCTRL.MSK_EN is 0, the α plane is read from the external RAM to be used for the α value for IROP operation; specify the α plane read start address (VI6_RPFn_SRCM_ADDR_A).</p> <p>When the virtual input function is used (VI6_RPFn_INFMT.VIR = 1), IROP operation is not available; set these bits to B'0000.</p> <p>B'0000: NOP(D) B'0001: AND(S & D) B'0010: AND_REVERSE(S & ~D) B'0011: COPY(S) B'0100: AND_INVERTED(~S & D) B'0101: CLEAR(0) B'0110: XOR(S ^ D) B'0111: OR(S D) B'1000: NOR(~(S D)) B'1001: EQUIV(~(S ^ D)) B'1010: INVERT(~D) B'1011: OR_REVERSE(S ~D) B'1100: COPY_INVERTED(~S) B'1101: OR_INVERTED(~S D) B'1110: NAND(~(S & D)) B'1111: SET(all 1)</p> |
| 23 | BSEL | B'0 | R/W | <p>α Bit Count Conversion Selection for 1-Bit Mask Generator</p> <p>Specifies the number of bits in the α plane to be read as mask information from the external RAM. The α value in mask information is used for the source (S) in IROP. When α plane data is 8 bits, it is converted to one bit through the 1-bit mask generator shown in Figure 33.22.</p> <p>Note that this bit setting is valid when the ASEL bits are set to 0 or 2 and VI6_RPFn_MSKCTRL.MSK_EN is set to 0. In other cases, this bit setting has no effect.</p> <p>0: 8-bit α is converted to 1-bit α through the 1-bit mask generator.</p> <p>When the 8-bit α value input to the RPF is not 0, it is converted to B'1; when the value is 0, it is converted to B'0.</p> <p>1: α value goes through the 1-bit mask generator.</p> <p>The 1-bit α value input to the RPF is output through the 1-bit mask generator without change.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|---|
| 19, 18 | AEXT[1:0] | B'00 | R/W | <p>Lower-Bit α Value Extension Method Set</p> <p>These bits specify the method for extending the input α data to 8 bits through the unpack processing.</p> <p>B'00: The lower-order bits of α value are extended with 0.</p> <p>B'01: The upper-order bits of α value are copied to the lower-order bits.</p> <p>B'10: The lower-order bits of α value are extended with 0. The maximum value is limited to H'FF.</p> <p>B'11: Setting prohibited</p> |
| 15 to 8 | ALPHA1[7:0] | All 0 | R/W | <p>8-Bit α Value Output when 1-Bit α Value is 1</p> <p>These bits specify the 8-bit α value to be output when 1-bit α data is input and the α value input to the 8-bit transparent α generator shown in Figure 33.22 is B'1. This setting is valid when the ASEL bits are set to B'010 or B'011.</p> <p>A value from 0 to 255 can be specified.</p> |
| 7 to 0 | ALPHA0 [7:0] | All 0 | R/W | <p>8-Bit α Value Output when 1-Bit α Value is 0</p> <p>These bits specify the 8-bit α value to be output when 1-bit α data is input and the α value input to the 8-bit transparent α generator shown in Figure 33.22 is B'0. This setting is valid when the ASEL bits are set to B'010 or B'011.</p> <p>A value from 0 to 255 can be specified.</p> |

Figure 33.22 shows the relationship between the α selector, IROP operation unit, color keying unit, and related registers. The IROP operation unit receives two inputs, source and destination. The image data input from the external memory is processed through the unpack processor and 8-bit transparent α generator and then input to the IROP operation unit as destination data. The α plane data input from the external memory is sent to the 8-bit transparent α generator when the ASEL bits are set to 1 or 3, or sent to the 1-bit mask α generator when the ASEL bits are set to 0 or 2. For the pixel data and 8-bit α value on the source side of the IROP operation unit, either the VI6_RPFn_MSKSET0 value or VI6_RPFn_MSKSET1 values will be selected according to the 1-bit α value output by the 1-bit mask α generator.

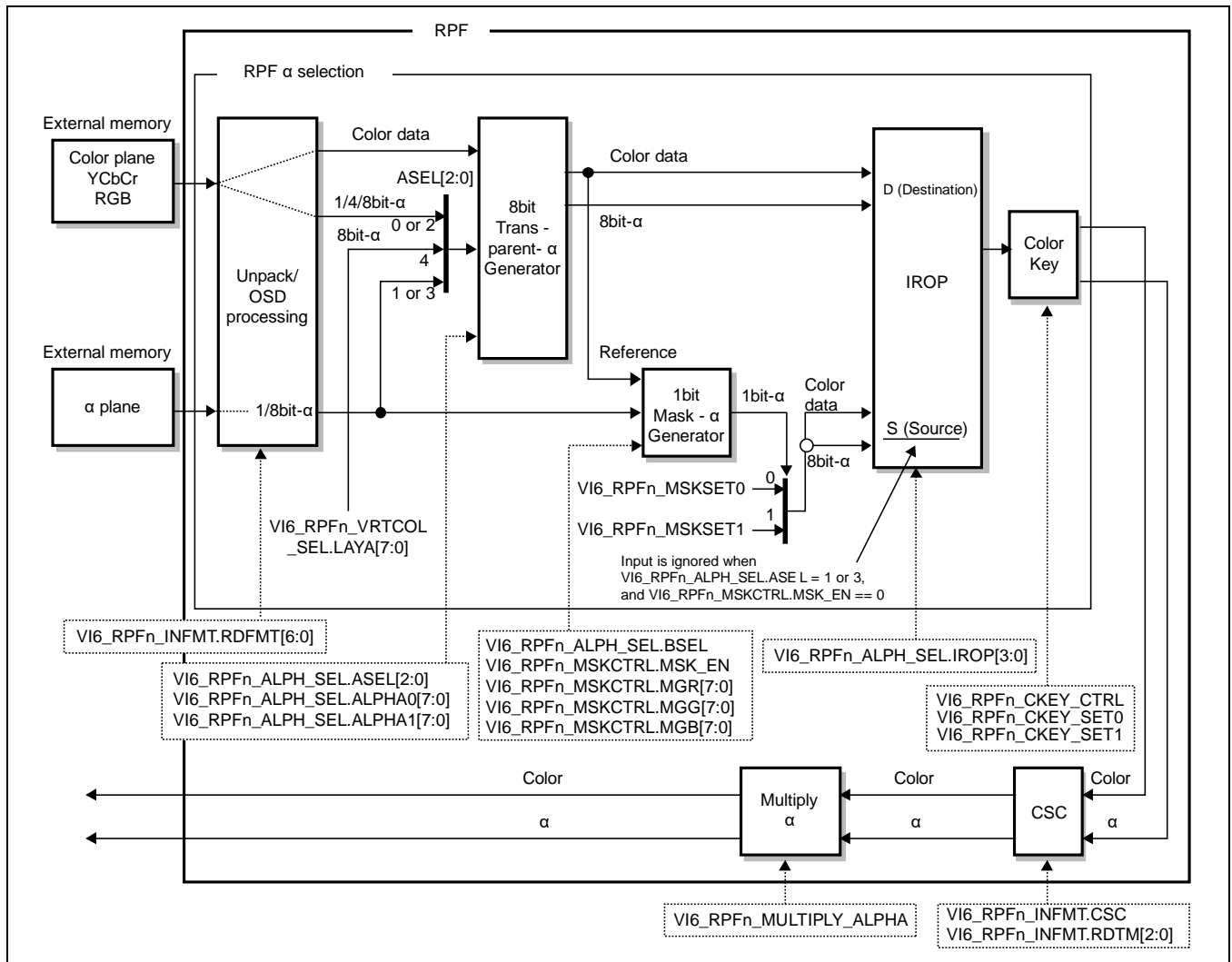


Figure 33.22 Configuration of alpha Selector and IROP Operation Unit in RPF

The following describes the function of each block shown in Figure 33.22. Read the following while referring to the figure as necessary.

Unpack/OSD processor:

Unpacks each component and α value of the image data according to the packed format specified in VI6_RPFn_INFMT.RDFMT.

8-bit transparent α generator:

Converts the input α value into 8-bit α when the input α is four bits or one bit.

When VI6_RPFn_ALPH_SEL.ASEL are set to 0 (8-, 4-, or 1-bit packed α), this generator outputs the input α value without change if the α bit field in the packed α data is 8 bits; if the α bit field is less than 8 bits, it is converted to an 8-bit α value by extending the LSB side according to the VI6_RPFn_ALPH_SEL.AEXT setting.

When VI6_RPFn_ALPH_SEL.ASEL are set to 1 (8-bit plane α) or 4 (fixed α), this generator outputs the input 8-bit plane α without change. If a packed α value is included in RGB data, it is discarded.

When VI6_RPFn_ALPH_SEL.ASEL are set to 2 (1-bit packed α) or 3 (8-bit α generated from 1-bit plane α), an 8-bit α value is generated by using the VI6_RPFn_ALPH_SEL.ALPHA0 [7:0] value when the input 1-bit α value is 0 or by using the VI6_RPFn_ALPH_SEL.ALPHA1 [7:0] value when the input 1-bit α value is 1. When VI6_RPFn_ALPH_SEL.ASEL is set to 3, a packed α value that is included in RGB data is discarded.

1-bit mask α generator:

Generates 1-bit α data from the input 8-bit α data or pixel data. When the input α data is one bit, this generator outputs it without change.

When VI6_RPFn_ALPH_SEL.ASEL are set to 0 (8-, 4-, or 1-bit packed α) or 2 (plane α) and VI6_RPFn_MSKCTRL.MSK_EN is set to 0, the α plane read from the external memory to be used in IROP is converted to 1-bit α data when necessary. When the α plane data read from the external RAM is 8 bits (BSEL = 0), if the value is 0, a 1-bit α value of B'0 is generated; if the value is not 0, a 1-bit α value of B'1 is generated. When the α plane data is one bit (BSEL = 1), this generator outputs it without change.

When the value of the 1-bit α generated by the 1-bit mask α generator is B'0, the 8-bit α and pixel data specified in VI6_RPFn_MSKSET0 are output as the source. When the generated 1-bit α value is B'1, the 8-bit α and pixel data specified in VI6_RPFn_MSKSET1 are output as the source.

As shown in Figure 33.22, when VI6_RPFn_ALPH_SEL.ASEL are set to 1 (8-bit plane α) or 3 (1-bit plane α), the α plane read from the external RAM is sent to the 8-bit transparent α generator as transparency information. When VI6_RPFn_MSKCTRL.MSK_EN is set to 0, the 1-bit α for masking is generated according to the input α plane (refer to section 33.2.6.8), but the 1-bit mask α generator does not refer to the input α plane because it is input to the 8-bit transparent α generator as transparency information. Accordingly, the 1-bit mask α generator does not generate a 1-bit α for masking and the data on the source side becomes invalid; that is, IROP operation cannot be executed. Set the IROP bits to 0 in this case. In contrast, when VI6_RPFn_MSKCTRL.MSK_EN is set to 1, the 1-bit mask α generator creates α data for masking according to the pixel data instead of the input α plane data, and IROP operation can be executed in this case.

IROP operation unit

Executes ROP operation according to the opcode specified in VI6_RPFn_ALPH_SEL.IROP. For ROP operation (other than NOP), valid values should be input both for the source and destination. As described in the above (description of the 1-bit mask α generator), when VI6_RPFn_ALPH_SEL.ASEL are set to 1 or 3 and VI6_RPFn_MSKCTRL.MSK_EN is set to 0, the source data for the IROP operation unit is treated as invalid; set VI6_RPFn_ALPH_SEL.IROP to 0 (NOP). When VI6_RPFn_ALPH_SEL.ASEL are set to 4, a fixed α value is output from the RPF and IROP operation is not available. In the same way as the above case, set VI6_RPFn_ALPH_SEL.IROP to 0 (NOP).

To specify a valid source value for the IROP operation unit and execute IROP operation (specify an opcode other than NOP in the IROP bits), specify register values as shown in Table 33.14. Where the source input state is indicated as "Valid" in the table, IROP operation can be executed. In the cases where IROP operation is not available, set the IROP bits to 0 (NOP).

Table 33.14 Source Input State in IROP Operation Unit

| VI6_RPFn_ALPH_SEL.ASEL [2:0] | VI6_RPFn_MSKCTRL.MSK_EN | | |
|---|--|--|-------|
| | 0 (Source data is generated according to input α plane) | 1 (Source data is generated according to the destination-side pixel data) | |
| B'000 (1-, 4-, or 8-bit packed α + plane α) | Valid | (α plane input) | Valid |
| B'001 (8-bit α plane) | Invalid | (IROP operation is not available; α plane is output to the subsequent modules behind RPF) | Valid |
| B'010 (8-bit α generated from 1-bit packed α + plane α) | Valid | (α plane input) | Valid |
| B'011 (8-bit α generated from 1-bit plane α) | Invalid | (IROP operation is not available; α plane is output to the subsequent modules behind RPF) | Valid |
| B'100 (Fixed α) | Invalid (IROP operation is not available; fixed α is output to the subsequent modules behind RPF) | | |

For the handling of the α values shown in Figure 33.22 and Table 33.14, the relationship between the RPF input format and RPF output α value is shown in Table 33.15. Where only bit names are shown in the table, the bits are in VI6_RPFn_ALPH_SEL described in this section.

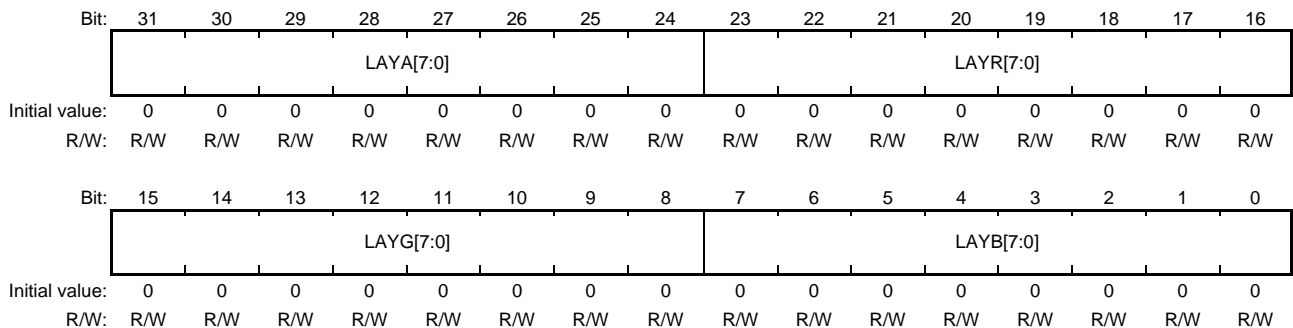
Table 33.15 α Value Selected and Output according to ASEL Bits in Each Input Format

*Fixed value H'FF is output because packed α is not included in YCbCr.

| ASEL Setting | α Value Output for Each Input Format | | |
|--|---|--------------------------|--------------------------|
| | RGB | YCbCr | OSD-CLUT |
| B'000 (8-, 4-, or 1-bit packed α is input) | 1-, 4-, or 8-bit pixel α | H'FF* | α value in CLUT |
| B'001 (8-bit plane α is input) | 8-bit α plane | 8-bit α plane | 8-bit α plane |
| B'010 (8-bit α is generated from the 1-bit packed α input) | ALPHA0 or ALPHA1 setting | H'FF* | H'FF |
| B'011 (8-bit α is generated from the 1-bit plane α input) | ALPHA0 or ALPHA1 setting | ALPHA0 or ALPHA1 setting | ALPHA0 or ALPHA1 setting |
| B'100 (Fixed α is output) | VI6_RPFn_VRTCOL_SET.LAYA setting | | |

33.2.6.7 RPFn Virtual Plane Color Information Registers (VI6_RPFn_VRTCOL_SET)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 24 | LAYA [7:0] | All 0 | R/W | <p>Virtual-Input Fixed α Value</p> <p>These bits specify the fixed α value generated in the RPFn when the virtual input function is enabled through the VI6_RPFn_INFMT.VIR setting.</p> <p>When the virtual input function is disabled (VI6_RPFn_INFMT.VIR = 0), these bits are used to specify the fixed α value to be output from the RPF while VI6_RPFn_ALPH_SEL.ASEL are set to 4. A value from 0 to 255 can be specified.</p> |
| 23 to 16 | LAYR [7:0] | All 0 | R/W | <p>Virtual-Input Fixed R/Cr Component Value</p> <p>These bits specify the fixed R or Cr value generated in the RPFn when the virtual input function is enabled through the VI6_RPFn_INFMT.VIR setting. When VI6_RPFn_INFMT.RDFMT is set to 19, the value specified in these bits is used as the R value. When VI6_RPFn_INFMT.RDFMT is set to 64, the value specified in these bits is used as the Cr value. A value from 0 to 255 can be specified.</p> |
| 15 to 8 | LAYG [7:0] | All 0 | R/W | <p>Virtual-Input Fixed G/Y Component Value</p> <p>These bits specify the fixed G or Y value generated in the RPFn when the virtual input function is enabled through the VI6_RPFn_INFMT.VIR setting. When VI6_RPFn_INFMT.RDFMT is set to 19, the value specified in these bits is used as the G value. When VI6_RPFn_INFMT.RDFMT is set to 64, the value specified in these bits is used as the Y value. A value from 0 to 255 can be specified.</p> |
| 7 to 0 | LAYB [7:0] | All 0 | R/W | <p>Virtual-Input Fixed B/Cb Component Value</p> <p>These bits specify the fixed B or Cb value generated in the RPFn when the virtual input function is enabled through the VI6_RPFn_INFMT.VIR setting. When VI6_RPFn_INFMT.RDFMT is set to 19, the value specified in these bits is used as the B value. When VI6_RPFn_INFMT.RDFMT is set to 64, the value specified in these bits is used as the Cb value. A value from 0 to 255 can be specified.</p> |

33.2.6.8 RPFn Mask Control Registers (VI6_RPFn_MSKCTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

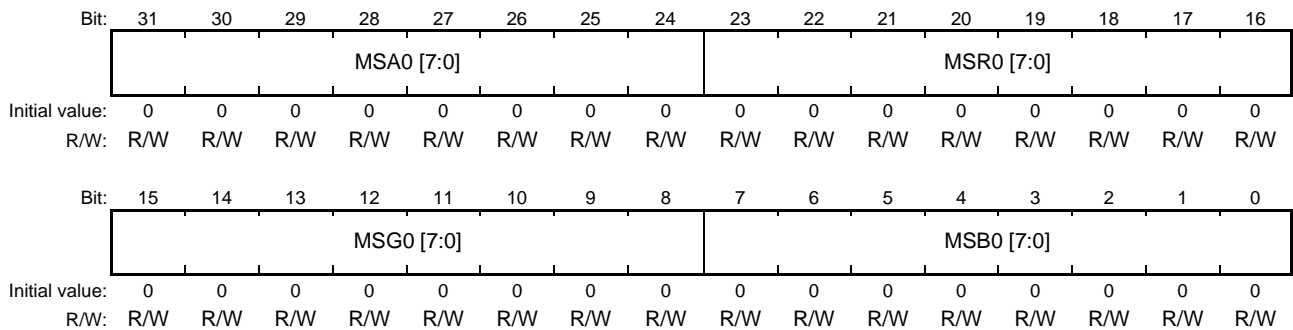
| | | | | | | | | | | | | | | | | |
|----------------|-----------|-----|-----|-----|-----|-----|-----|--------|-----------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | MSK_EN | MGR [7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MGG [7:0] | | | | | | | | MGB [7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | MSK_EN | B'0 | R/W | Mask Generation Specification Specifies the method of α value generation in the 1-bit mask α generator shown in Figure 33.22. 0: A 1-bit mask value is generated according to the input α plane value. When the input α is in the 1-bit format (VI6_RPFn_ALPH_SEL.BSEL = 1), the 1-bit mask value is output without change. When the input α is in the 8-bit format (VI6_RPFn_ALPH_SEL.BSEL = 0), the 1-bit mask value is 0 if the α value is H'00; otherwise, the 1-bit mask value is 1. 1: The R/Cr, G/Y, and B/Cb components of the image input to the destination side of the IROP operation unit are compared with the values specified in the MGR, MGG, and MGB bits, respectively. When all values match, 1 is output as the 1-bit mask value, and in other cases, 0 is output. When the generated 1-bit mask data is not used, set VI6_RPFn_ALPH_SEL.IROP to 0. |
| 23 to 16 | MGR [7:0] | All 0 | R/W | R/Cr Comparison Value for 1-Bit α Generation These bits specify the R/Cr value to be compared for 1-bit α generation by using the pixel data on the destination side. When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify an R value for comparison. When YCbCr is specified, specify a Cr value for comparison. This setting is ignored when the MSK_EN bit is set 0. A value from 0 to 255 can be specified. |
| 15 to 8 | MGG [7:0] | All 0 | R/W | G/Y Comparison Value for 1-Bit α Generation These bits specify the G/Y value to be compared for 1-bit α generation by using the pixel data on the destination side. When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a G value for comparison. When YCbCr is specified, specify a Y value for comparison. This setting is ignored when MSK_EN is set to 0. A value from 0 to 255 can be specified. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|---|
| 7 to 0 | MGB [7:0] | All 0 | R/W | <p>B/Cb Comparison Value for 1-Bit α Generation</p> <p>These bits specify the B/Cb value to be compared for 1-bit α generation by using the pixel data on the destination side. When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a B value for comparison. When YCbCr is specified, specify a Cb value for comparison.</p> <p>This setting is ignored when MSK_EN is set to 0.</p> <p>A value from 0 to 255 can be specified.</p> |

33.2.6.9 RPFn IROP-SRC Input Value Registers 0 (VI6_RPFn_MSKSET0)

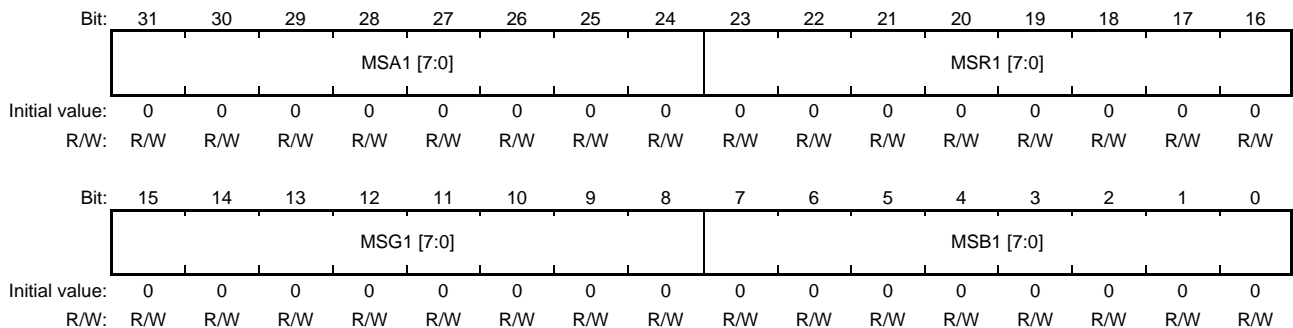
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 24 | MSA0 [7:0] | All 0 | R/W | IROP-Source Input α Value when 1-Bit α is 0 These bits specify the 8-bit α value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (Figure 33.22). A value from 0 to 255 can be specified. |
| 23 to 16 | MSR0 [7:0] | All 0 | R/W | IROP-Source Input R/Cr Value when 1-Bit α is 0 These bits specify the R/Cr value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (Figure 33.22). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify an R component value. When YCbCr is specified, specify a Cr component value. A value from 0 to 255 can be specified. |
| 15 to 8 | MSG0 [7:0] | All 0 | R/W | IROP-Source Input G/Y Value when 1-Bit α is 0 These bits specify the G/Y value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (Figure 33.22). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a G component value. When YCbCr is specified, specify a Y component value. A value from 0 to 255 can be specified. |
| 7 to 0 | MSB0 [7:0] | All 0 | R/W | IROP-Source Input B/Cb Value when 1-Bit α is 0 These bits specify the B/Cb value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 0 (Figure 33.22). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a B component value. When YCbCr is specified, specify a Cb component value. A value from 0 to 255 can be specified. |

33.2.6.10 RPFn IROP-SRC Input Value Registers 1 (VI6_RPFn_MSKSET1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 24 | MSA1 [7:0] | All 0 | R/W | IROP-Source Input α Value when 1-Bit α is 1 These bits specify the 8-bit α value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1. A value from 0 to 255 can be specified (Figure 33.22). |
| 23 to 16 | MSR1 [7:0] | All 0 | R/W | IROP-Source Input R/Cr Value when 1-Bit α is 1 These bits specify the R/Cr value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1 (Figure 33.22). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify an R component value. When YCbCr is specified, specify a Cr component value. A value from 0 to 255 can be specified. |
| 15 to 8 | MSG1 [7:0] | All 0 | R/W | IROP-Source Input G/Y Value when 1-Bit α is 1 These bits specify the G/Y value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1 (Figure 33.22). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a G component value. When YCbCr is specified, specify a Y component value. A value from 0 to 255 can be specified. |
| 7 to 0 | MSB1 [7:0] | All 0 | R/W | IROP-Source Input B/Cb Value when 1-Bit α is 1 These bits specify the B/Cb value to be input as the source to the IROP operation unit when the internal 1-bit α value generated through the 1-bit mask generator is 1 (Figure 33.22). When RGB is specified as the RPFn input format through VI6_RPFn_INFMT.RDFMT, specify a B component value. When YCbCr is specified, specify a Cb component value. A value from 0 to 255 can be specified. |

33.2.6.11 RPFn Color Keying Control Registers (VI6_RPFn_CKEY_CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

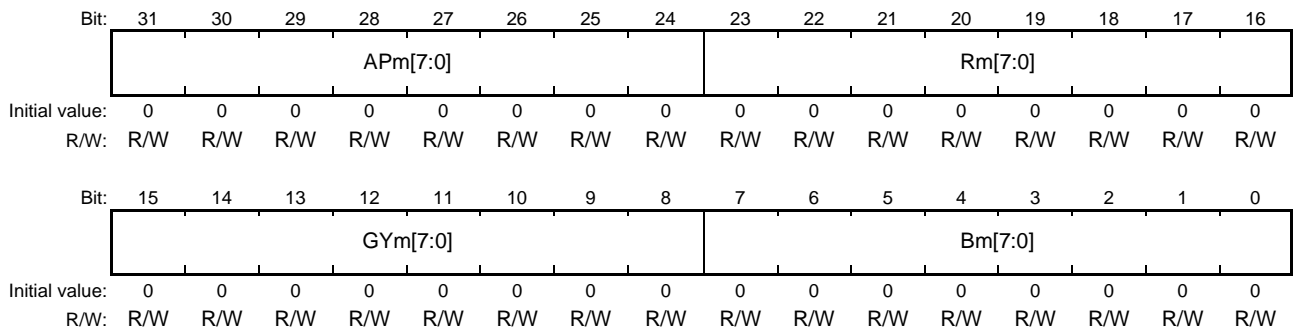
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----|----|----|----|-----|----|----|--------|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | LTH | — | — | — | CV | — | — | SAP E1 | SAP E0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R/W | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----------------------------|----------|---------------|-----|---|
| 31 to 9, 7 to 5, 3, 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | LTH | B'0 | R/W | <p>Transparent color-luma Threshold Mode Enable/Disable</p> <p>This bit enables or disables transparent color-luma threshold mode for the color keying module.</p> <p>In transparent color-luma threshold mode, color information 0 (GY0) specified in VI6_RPFn_CKEY_SET0 register is compared with the input luma values.</p> <p>When the input data is in YCbCr format, and if input Y value is equal to or smaller than specified color information (VI6_RPFn_CKEY_SET0.GY0), the input α value is replaced with the value specified in VI6_RPFn_CKEY_SET0.AP0. This bit is available only when the input data is in YCbCr format. When the input data is in an RGB format, set this bit to 0.</p> <p>This bit setting is valid only when the CV bit is set to 0; it is ignored when the CV bit is set to 1. When using the virtual input function (VI6_RPFn_INFMT.VIR = 1), set this bit to 0.</p> <p>0: Luma threshold mode is disabled (Matched color mode) 1: Luma threshold mode is enabled</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 4 | CV | B'0 | R/W | <p>Color Replacement Control</p> <p>This bit controls the color replacement function in the color keying module shown in Figure 33.7. When an RGB format is specified as the color space of the RPFn input data through VI6_RPFn_INFMT.RDFMT, and if all components of an input pixel match the color components specified in VI6_RPFn_CKEY_SET0, the color replacement function replaces the values of the input α and all RGB components with the α and color components specified in VI6_RPFn_CKEY_SET1. When a YCbCr format is specified as the color space of the RPFn input data through VI6_RPFn_INFMT.RDFMT, only the Y data is compared; if the luminance component of an input pixel matches the value specified in VI6_RPFn_CKEY_SET0.GY0, the color replacement function replaces the values of the input α and all YCbCr components with the α and color components specified in VI6_RPFn_CKEY_SET1.</p> <p>When the CV bit is set to 1, the color replacement function is enabled. When using the virtual input function (VI6_RPFn_INFMT.VIR = 1), set this bit to 0.</p> <p>0: Color replacement function is disabled (transparent color mode).</p> <p>1: Color replacement function is enabled.</p> |
| 1 | SAPE1 | B'0 | R/W | <p>Comparison Color Data Setting 1 Enable/Disable</p> <p>This bit enables or disables comparison color data setting 1 in the transparent color mode for the color keying module. This bit setting is valid when the CV bit and the LTH bit are set to 0; it is ignored when the CV bit or the LTH bit is set to 1.</p> <p>In transparent color mode, color information 1 (VI6_RPFn_CKEY_SET1.R1/GY1/B1) specified in VI6_RPFn_CKEY_SET1 is compared with the input component values. When the input data is in an RGB format, and if all input values match the specified color information, the input α value is replaced with the value specified in VI6_RPFn_CKEY_SET1.AP. When the input data is in YCbCr format, only the Y data is compared.</p> <p>When using the virtual input function (VI6_RPFn_INFMT.VIR = 1), set this bit to 0.</p> <p>0: Comparison color data setting 1 is disabled.</p> <p>1: Comparison color data setting 1 is enabled.</p> |
| 0 | SAPE0 | B'0 | R/W | <p>Comparison Color Data Setting 0 Enable/Disable</p> <p>This bit enables or disables comparison color data setting 0 in the transparent color mode for the color keying module. This bit setting is valid when the CV bit and the LTH bit are set to 0; it is ignored when the CV bit or the LTH bit is set to 1.</p> <p>In transparent color mode, color information 0 (VI6_RPFn_CKEY_SET0.R0/GY0/B0) specified in VI6_RPFn_CKEY_SET0 is compared with the input component values. When the input data is in an RGB format, and if all input values match the specified color information, the input α value is replaced with the value specified in VI6_RPFn_CKEY_SET0.AP. When the input data is in YCbCr format, only the Y data is compared.</p> <p>When using the virtual input function (VI6_RPFn_INFMT.VIR = 1), clear this bit to 0.</p> <p>0: Comparison color data setting 0 is disabled.</p> <p>1: Comparison color data setting 0 is enabled.</p> |

33.2.6.12 RPFn Color Keying Color Setting Registers-m (VI6_RPFn_CKEY_SETm: m = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



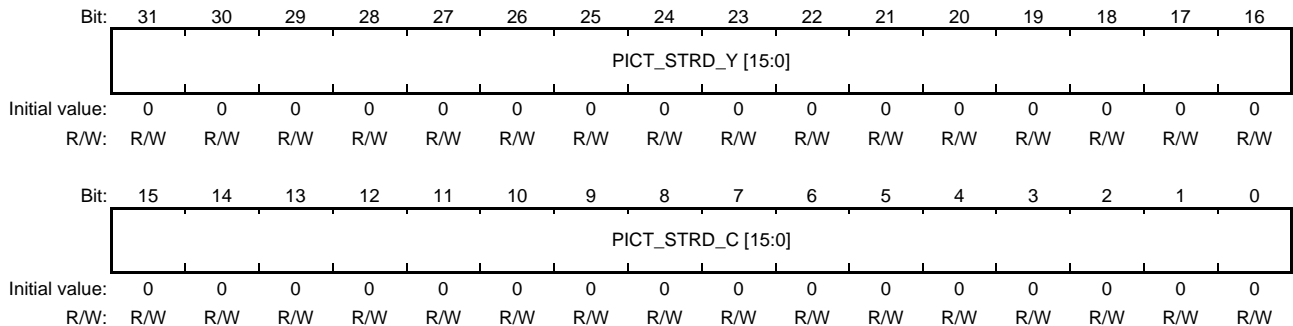
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 24 | APm [7:0] | All 0 | R/W | <p>α Data in Color Keying Color Information-m</p> <ul style="list-style-type: none"> In transparent color-matched color mode for color keying When the input data matches color setting-m (Rm, GYm, and Bm) for transparent color comparison in the color keying module, the input α value is replaced with the value specified in these bits. Specify the α value to replace the input value. In transparent color-luma threshold mode for color keying When the input data is in YCbCr format, and if input Y value is equal to or smaller than GY0, the input α value is replaced with the value specified in this bit field (AP0). Specify the α value in this bit field (AP0) to replace the input value. AP1 is not used in this mode. Set AP1 to 0 value. In color replacement mode for color keying These bits are not used in this mode. Clear them to 0. α value replacement through these bits in transparent color mode for color keying takes priority over the α value selected through the VI6_RPFn_ALPH_SEL.ASEL setting. A value from 0 to 255 can be specified. |
| 23 to 16 | Rm [7:0] | All 0 | R/W | <p>R*/Cr Component Data in Color Keying Color Information-m</p> <ul style="list-style-type: none"> In transparent color-matched color mode for color keying Specify the R component value for comparison enabled through the VI6_RPFn_CKEY_CTRL.SAPEm setting. When the RPFn input is in YCbCr format, the color keying module does not compare the Cr component, and the setting of these bits is ignored. In transparent color-luma threshold mode for color keying The color keying module does not refer this bit field, and the setting of these bits is ignored. In color replacement mode for color keying Specify the R component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 15 to 8 | GYm [7:0] | All 0 | R/W | <p>G*/Y Component Data in Color Keying Color Information-m</p> <ul style="list-style-type: none"> In transparent color-matched color mode for color keying Specify the G/Y component value for comparison enabled through the VI6_RPFn_CKEY_CTRL.SAPEm setting. In transparent color-luma threshold mode for color keying Specify the Y component value in the GY0 to be compared. In color replacement mode for color keying Specify the G/Y component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified. |
| 7 to 0 | Bm [7:0] | All 0 | R/W | <p>B*/Cb Component Data in Color Keying Color Information-m</p> <ul style="list-style-type: none"> In transparent color-matched color mode for color keying Specify the B component value for comparison enabled through the VI6_RPFn_CKEY_CTRL.SAPEm setting. When the RPFn input is in YCbCr format, the color keying module does not compare the Cb component, and the setting of these bits is ignored. In transparent color-luma threshold mode for color keying The color keying module does not refer this bit field, and the setting of these bits is ignored. In color replacement mode for color keying Specify the B component value to be compared with the input data in the color replacement function of the color keying module. A value from 0 to 255 can be specified. |

Note: * When comparison data is specified in an RGB format, if a packed format is selected for RGB input and each of the RGB components is not 8 bits, the lower-order bits of input data are extended as specified through VI6_RPFn_INFMT.CEXT before comparison. The RGB components to be compared with the input should also be extended in the same way and the extended values should be specified in this register.

33.2.6.13 RPFn Source Picture Memory Stride Setting Registers (VI6_RPFn_SRCM_PSTRIDE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------|---------------|-----|---|
| 31 to 16 | PICT_STRD_Y [15:0] | All 0 | R/W | <p>Memory Stride of Source Picture Y/RGB Plane</p> <p>These bits specify in 1-byte units the memory stride of the source picture Y/RGB plane read by the RPFn.</p> <p>A value from 0 to 65535 can be specified. Refer to Figure 33.23 for settings.</p> |
| 15 to 0 | PICT_STRD_C [15:0] | All 0 | R/W | <p>Memory Stride of Source Picture C Plane</p> <p>These bits specify in 1-byte units the memory stride of the source picture C plane read by the RPFn. When an RGB-format picture is read, these bits do not need to be set.</p> <p>A value from 0 to 65535 can be specified. Refer to Figure 33.23 for settings.</p> <p>In the YCbCr planar format, this setting is used as the memory stride of the Cb and Cr planes.</p> |

This register specifies the memory stride of the picture planes in the source area as shown in Figure 33.23. The memory stride of the α plane should be specified through VI6_RPFn_SRCM_ASTRIDE.ALPH_STRD. When the RPF input is in an RGB format, only the RGB plane is read; when the input is in YCbCr format, the Y and C planes are read as shown in Figure 33.23. When the α plane is used, the α plane is also read. According to the image format and the necessity of the α plane, specify the necessary addresses where the source image is stored (VI6_RPFn_SRCM_ADDR_Y, VI6_RPFn_SRCM_ADDR_C, and VI6_RPFn_SRCM_ADDR_AI).

Whether the α plane needs to be read is determined according to the α plane selection method and IROP operation type. For details, refer to section 33.2.6.6.

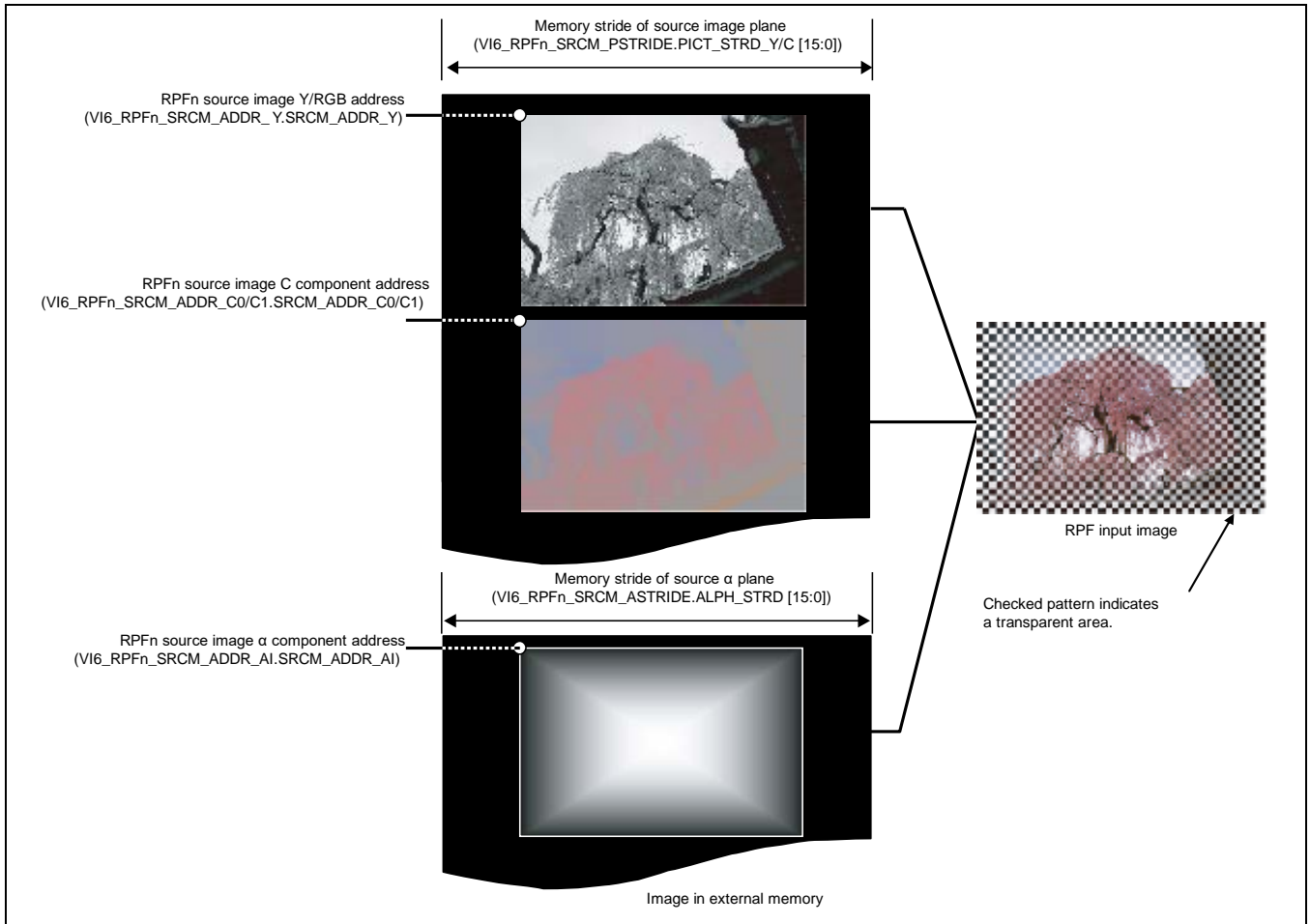
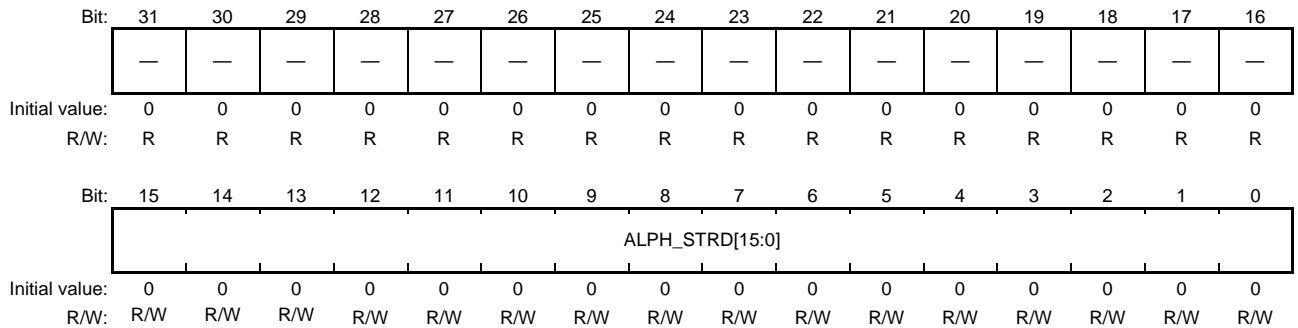


Figure 33.23 Reading an Image from RPFn Source Area

33.2.6.14 RPFn Source α Memory Stride Setting Registers (VI6_RPFn_SRCM_ASTRIDE)

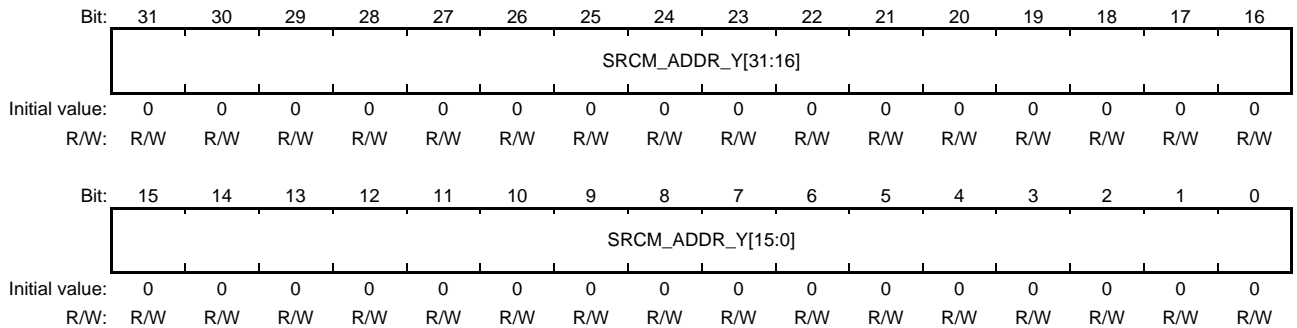
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | ALPH_STRD [15:0] | All 0 | R/W | Memory Stride of Source α Plane These bits specify in 1-byte units the memory stride of the source α plane read by the RPFn. A value from 0 to 65535 can be specified. Refer to Figure 33.23 for settings. |

33.2.6.15 RPFn Source Y/RGB Address Registers (VI6_RPFn_SRCM_ADDR_Y)

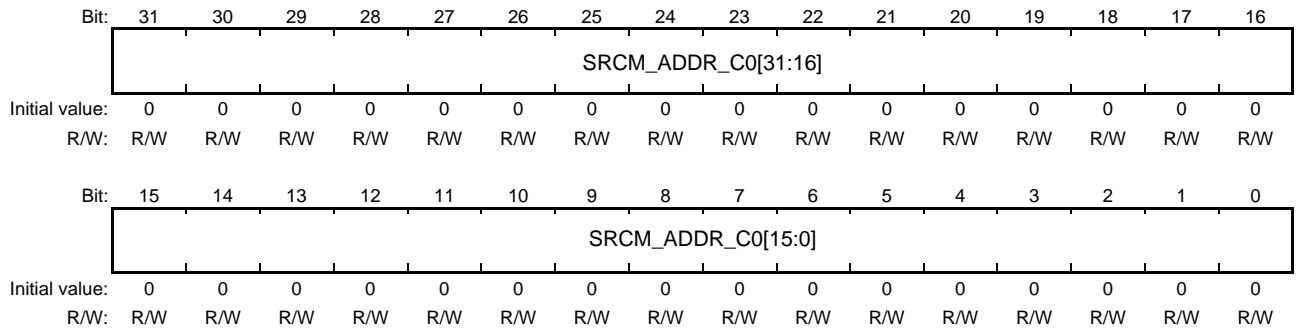
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------|---------------|-----|---|
| 31 to 0 | SRCM_ADDR_Y [31:0] | All 0 | R/W | <p>Source Image Y/RGB Plane Storing Address</p> <p>These bits specify in 1-byte units the start address of the source image Y plane and packed RGB plane read by the RPFn.</p> <p>A value from H'0000_0000 to H'FFFF_FFFF can be specified.</p> <p>Refer to Figure 33.23 in section 33.2.6.13 for settings.</p> |

33.2.6.16 RPFn Source Chroma Address Registers 0 (VI6_RPFn_SRCM_ADDR_C0)

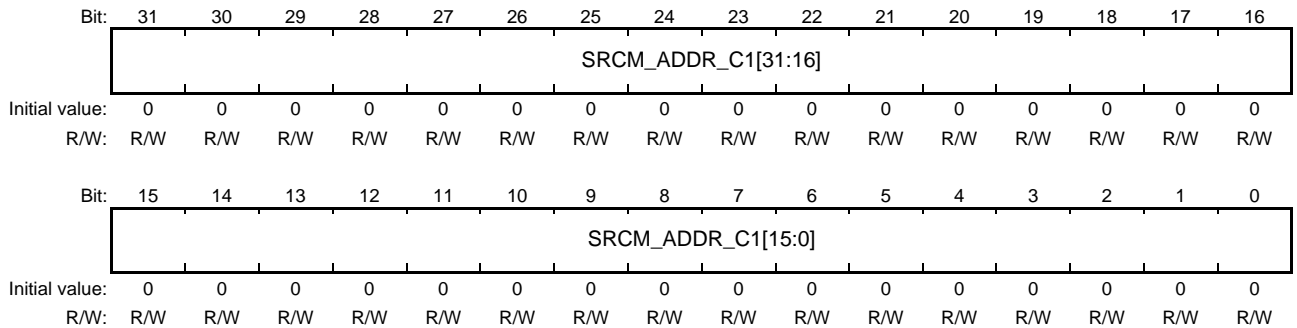
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------|---------------|-----|---|
| 31 to 0 | SRCM_ADDR_C0 [31:0] | All 0 | R/W | <p>Source Image C Plane Storing Address 0</p> <p>These bits specify in 1-byte units the start address of the source image C plane read by the RPFn. Here, the C plane indicates the combined CbCr plane when a semi-planar format is selected from the packed YCbCr formats shown in Table 33.11 or the Cb plane when a planar format is selected. When an interleaved format is selected or the RPF input is in an RGB format, this setting is not used.</p> <p>A value from H'0000_0000 to H'FFFF_FFFF can be specified. Refer to Figure 33.23 in section 33.2.6.13 for settings.</p> |

33.2.6.17 RPFn Source Chroma Address Registers 1 (VI6_RPFn_SRCM_ADDR_C1)

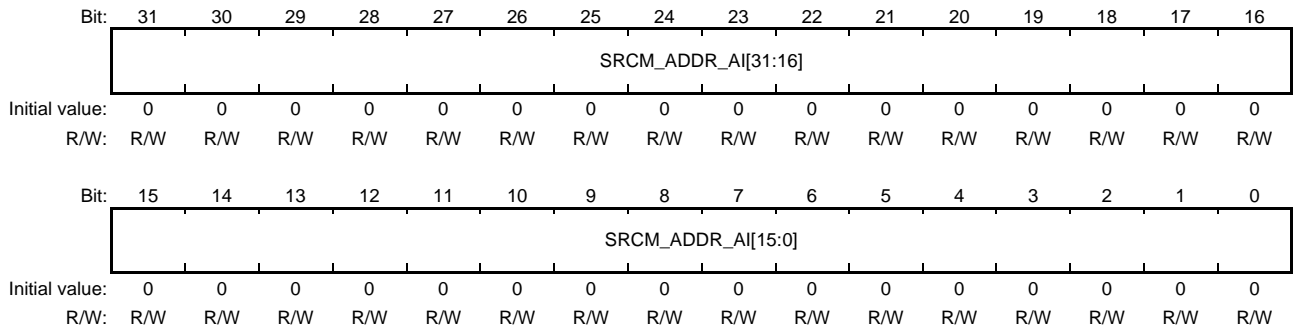
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------|---------------|-----|---|
| 31 to 0 | SRCM_ADDR_C1 [31:0] | All 0 | R/W | <p>Source Image C Plane Storing Address 1</p> <p>These bits specify in 1-byte units the start address of the Cr plane when a planar YCbCr format shown in Table 33.11 is read by the RPFn.</p> <p>This setting is not used when the RPF input is in YCbCr format that is not a planar format or in an RGB format.</p> <p>A value from H'0000_0000 to H'FFFF_FFFF can be specified. Refer to Figure 33.23 in section 33.2.6.13 for settings.</p> |

33.2.6.18 RPFn Source α Address Registers (VI6_RPFn_SRCM_ADDR_AI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------|---------------|-----|--|
| 31 to 0 | SRCM_ADDR_AI [31:0] | All 0 | R/W | <p>Source Image α Plane Storing Address</p> <p>These bits specify in 1-byte units the start address of the α plane of the source image read by the RPFn. Specify in the same way as the start address of the picture plane. When the α plane is not read from the source area, these bits do not need to be set.</p> <p>A value from H'0000_0000 to H'FFFF_FFFF can be specified. Refer to Figure 33.23 in section 33.2.6.13 for settings.</p> |

33.2.6.19 RPFn Bus Access Control Registers (VI6_RPFn_BAC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | — | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | B512 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|-------------------|----------|---------------|-----|--|
| 31 to 17, 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | B512 | B'0 | R/W | Burst access in 512 pixels' enable 0: burst access in 256 pixels 1: burst access in 512 pixels This bit is used to vary the bus access between VSPBC and VSPBD in order to enhance the bus efficiency. For the case using both VSPBC and VSPBD in RZ/G2H: Always set 1 to this bit for VSPBC and set 0 to this bit for VSPBD. For other cases: Always set 0 to this bit |

33.2.6.20 RPFn Multiple Alpha Control (VI6_RPFn_MULT_ALPH)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----------|----|----|------------|-----|------------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | A_M MD | — | — | P_MMD[1:0] | | ALPHA_RATIO[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|------------------|-------------------|---------------|-----|--|
| 31 to 13, 11, 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | A_MMD | B'0 | R/W | 0: Alpha data go through multiply-alpha unit. 1: Multiply-alpha unit multiplies alpha data by specified alpha (ALPHA_RATIO [7:0]). When output format from CSC unit is YCbCr, set 0 to this bit. |
| 9 to 8 | P_MMD[1:0] | B'00 | R/W | B'00: Image data go through multiply-alpha unit. B'01: Multiply-alpha unit multiplies image data by specified alpha (ALPHA_RATIO [7:0]). B'10: Multiply-alpha unit multiplies image data by alpha data. B'11: Multiply-alpha unit multiplies image data by alpha data and specified alpha (ALPHA_RATIO [7:0]). When output format from CSC unit is YCbCr, set 0 to this bit. |
| 7 to 0 | ALPHA_RATIO [7:0] | All 0 | R/W | Multiply alpha value |

ALPIn: Input Alpha data to Multiply-alpha unit

PIXIn(R): Input R data to Multiply-alpha unit

PIXIn(G): Input G data to Multiply-alpha unit

PIXIn(B): Input B data to Multiply-alpha unit

ALPout: Output Alpha data from Multiply-alpha unit

PIXout(R): Output R data from Multiply-alpha unit

PIXout(G): Output G data from Multiply-alpha unit

PIXout(B): Output B data from Multiply-alpha unit

Table 33.16 Expression of output alpha data from Multiply-alpha unit

| A_MMD | ALPHA_RATIO[7:0] | Expression |
|-------|------------------|---------------------------------------|
| 0 | Don't care | $ALPout = ALPin$ |
| 1 | Not 255 | $ALPout = ALPin * ALPHA_RATIO / 256$ |
| | 255 | $ALPout = ALPin$ |

Table 33.17 Expression of output pixel data from Multiply-alpha unit

| P_MMD[1:0] | ALPHA_RATIO[7:0] | ALPin | Expression |
|------------|------------------|------------|---|
| 0 | Don't care | Don't care | $PIXout(R) = PIXin(R)$ $PIXout(G) = PIXin(G)$ $PIXout(B) = PIXin(B)$ |
| | Not 255 | Don't care | $PIXout(R) = PIXin(R) * ALPHA_RATIO / 256$ $PIXout(G) = PIXin(G) * ALPHA_RATIO / 256$ $PIXout(B) = PIXin(B) * ALPHA_RATIO / 256$ |
| 1 | 255 | Don't care | $PIXout(R) = PIXin(R)$ $PIXout(G) = PIXin(G)$ $PIXout(B) = PIXin(B)$ |
| | Don't care | Not 255 | $PIXout(R) = PIXin(R) * ALPin / 256$ $PIXout(G) = PIXin(G) * ALPin / 256$ $PIXout(B) = PIXin(B) * ALPin / 256$ |
| 2 | Don't care | 255 | $PIXout(R) = PIXin(R)$ $PIXout(G) = PIXin(G)$ $PIXout(B) = PIXin(B)$ |
| | Not 255 | Not 255 | $PIXout(R) = PIXin(R) * ALPin * ALPHA_RATIO / 256 / 256$ $PIXout(G) = PIXin(G) * ALPin * ALPHA_RATIO / 256 / 256$ $PIXout(B) = PIXin(B) * ALPin * ALPHA_RATIO / 256 / 256$ |
| 3 | 255 | Not 255 | $PIXout(R) = PIXin(R) * ALPin / 256$ $PIXout(G) = PIXin(G) * ALPin / 256$ $PIXout(B) = PIXin(B) * ALPin / 256$ |
| | Not 255 | 255 | $PIXout(R) = PIXin(R) * ALPHA_RATIO / 256$ $PIXout(G) = PIXin(G) * ALPHA_RATIO / 256$ $PIXout(B) = PIXin(B) * ALPHA_RATIO / 256$ |
| 3 | 255 | 255 | $PIXout(R) = PIXin(R)$ $PIXout(G) = PIXin(G)$ $PIXout(B) = PIXin(B)$ |
| | Not 255 | Not 255 | $PIXout(R) = PIXin(R) * ALPHA_RATIO / 256$ $PIXout(G) = PIXin(G) * ALPHA_RATIO / 256$ $PIXout(B) = PIXin(B) * ALPHA_RATIO / 256$ |

33.2.7 WPF Control Registers

33.2.7.1 WPFn-Source-RPF Registers (VI6_WPFn_SRCRPF)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|--------------|-----|----|----|----------------|---------------|----------------|----------------|----------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | VIR_ACT[1:0] | — | — | — | VIR_ACT2 [1:0] | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R | R | R/W | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | RPF4_ACT [1:0] | RPF3_ACT[1:0] | RPF2_ACT [1:0] | RPF1_ACT [1:0] | RPF0_ACT [1:0] | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------------|---------------|---------------|-----|--|
| 31, 30, 27, 26, 23 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29, 28 | VIR_ACT [1:0] | B'00 | R/W | <p>Virtual RPF Start Enable in BRU</p> <p>These bits enable start of the virtual RPF in the BRU as the source RPF for the WPFn when the WPFn is started. For details of the virtual RPF, refer to the following.</p> <p>Section 33.2.15.1, BRU Input Control Register (VI6_BRU_INCTRL)</p> <p>Section 33.2.15.2, Size Register of BRU Input Virtual RPF (VI6_BRU_VIRRRPF_SIZE)</p> <p>Section 33.2.15.3, Display Location Register of BRU Input Virtual RPF (VI6_BRU_VIRRRPF_LOC)</p> <p>Section 33.2.15.4, Color Information Register of BRU Input Virtual RPF (VI6_BRU_VIRRRPF_COL)</p> <p>Note that the virtual RPF is in the BRU as shown in Figure 33.45 and there are no register bits for DPR setting related to the virtual RPF.</p> <p>B'00: The virtual RPF in the BRU is not started.</p> <p>B'01: The virtual RPF in the BRU is started as a sublayer source RPF for the WPFn.</p> <p>B'10: The virtual RPF in the BRU is started as the master-layer source RPF for the WPFn.</p> <p>B'11: Setting prohibited</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-------------------|---------------|-----|---|
| 25, 24 | VIR_ACT2 [1:0] | B'00 | R/W | <p>Virtual RPF Start Enable in BRS [RZ/G2H, RZ/G2N]</p> <p>These bits enable start of the virtual RPF in the BRS as the source RPF for the WPF0 when the WPF0 is started. For details of the virtual RPF, refer to the following.</p> <p>Section 33.2.19.1, BRS Input Control Register (VI6_BRS_INCTRL)</p> <p>Section 33.2.19.2, Size Register of BRS Input Virtual RPF (VI6_BRS_VIRRRPF_SIZE)</p> <p>Section 33.2.19.3, Display Location Register of BRS Input Virtual RPF (VI6_BRS_VIRRRPF_LOC)</p> <p>Section 33.2.19.4, Color Information Register of BRS Input Virtual RPF (VI6_BRS_VIRRRPF_COL)</p> <p>Note that the virtual RPF is in the BRS as shown in Figure 33.55 and there are no register bits for DPR setting related to the virtual RPF.</p> <p>B'00: The virtual RPF in the BRS is not started.</p> <p>B'01: The virtual RPF in the BRS is started as a sublayer source RPF for the WPFn.</p> <p>B'10: The virtual RPF in the BRS is started as the master-layer source RPF for the WPFn.</p> <p>B'11: Setting prohibited</p> |
| 9, 8 | RPF4_ACT [1:0] | B'00 | R/W | <p>RPFn Start Enable (RPFn_ACT, n = 0 to 4)</p> <p>These bits enable start of RPFn as the source RPF for the WPFn when the WPFn is started. When RPFn is not started by any of the WPF0 to WPF1, set the VI6_DPR_RPFn_ROUTE.RT_RPFn bits to D'63. When ILV is used and image size of ILV output is equal to output size of the WPFn, set the RPFn_ACT, which ILVL (ILV left) is connected, to 2.</p> |
| 7, 6 | RPF3_ACT [1:0] | B'00 | R/W | <p>B'00: RPFn is not started.</p> |
| 5, 4 | RPF2_ACT [1:0] | B'00 | R/W | <p>B'01: RPFn is started as a sublayer source RPF for the WPFn.</p> |
| 3, 2 | RPF1_ACT [1:0] | B'00 | R/W | <p>B'10: RPFn is started as the master-layer source RPF for the WPFn.</p> |
| 1, 0 | RPF0_ACT [1:0] | B'00 | R/W | <p>B'11: Setting prohibited</p> |

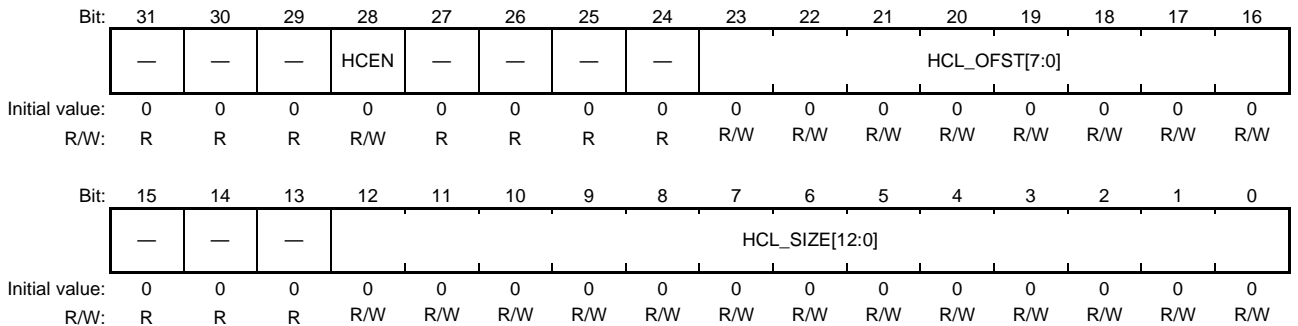
When the WPFn is started through the VSP2 start register n (VI6_CMDn: n = 0, 1), the RPF and virtual RPF in the BRU specified as the source RPF in this register are also started to supply data to the VSP2 internal modules.

Note the following when specifying the source RPF.

- (1) Each source RPF can be assigned to only one target WPFn; a single RPF cannot be assigned as the source RPF for multiple WPFs. For example, when setting VI6_WPF0_SRCRPF to H'0000_0008 and VI6_WPF1_SRCRPF to H'0000_0004 is attempted, the VSP2 will not operate correctly (these settings are prohibited) because RPF1 is assigned as both the master-layer source RPF for WPF0 and a sublayer source RPF for WPF1.
- (2) When blending or ROP operation is applied to multiple images through the BRU or BRS, multiple source RPFs are necessary for one WPF. When multiple source RPFs are used, images should be classified into a master layer and sublayers; assign one of the source RPFs as the master-layer source RPF and other RPFs as sublayer source RPFs. Do not assign all RPFs as sublayer source RPFs (VI6_WPF1_SRCRPF = H'0000_0015) or two or more RPFs as the master-layer source RPF (VI6_WPF0_SRCRPF = H'0000_025A) (such settings are prohibited).
- (3) When the BRU is not used, there should be only one source RPF for one WPF. In this case, the source RPF should be assigned as the master-layer source RPF.

33.2.7.2 WPFn Horizontal Input Size Clipping Registers (VI6_WPFn_HSZCLIP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|------------------------------|----------------|---------------|-----|---|
| 31 to 29, 27 to 24, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | HCEN | B'0 | R/W | Horizontal Size Clipping Enable/Disable Enables or disables clipping of the horizontal size of the WPFn input image. 0: Horizontal size clipping is disabled 1: Horizontal size clipping is enabled |
| 23 to 16 | HCL_OFST [7:0] | All 0 | R/W | Horizontal Size Clipping Offset Value Setting These bits specify the offset size (pixels) from the left end of the image in horizontal size clipping when the HCEN bit is 1 (Figure 33.24). The left side of the image input to the WPF is cut off for the size specified in these bits. When the HCEN bit is 0, this setting is ignored. For WPF0, specify the left-side offset size (pixels) for the image before rotation. A value from 0 to 255 can be specified. (HCL_OFST + HCL_SIZE) should not exceed the horizontal size of the WPF input. If the setting shown in the bottom example in Figure 33.24 is made, VSP2 does not operate correctly. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 12 to 0 | HCL_SIZE [12:0] | All 0 | R/W | <p>Horizontal Clipping Size Setting</p> <p>When the HCEN bit is 1, these bits specify the clipping size for horizontal clipping processing. Through this processing, the area of the horizontal size specified through the HCL_SIZE bits starting from the offset position specified through the HCL_OFST bits is determined as the valid image area. Accordingly, the right-side pixels beyond the (HCL_OFST + HCL_SIZE) size in the WPFn input image are discarded. When the HCEN bit is 0, this setting is ignored.</p> <p>For WPF0, specify the horizontal clipping size for the image before rotation.</p> <p>A value from 1 to 8190 can be specified. (HCL_OFST + HCL_SIZE) should not exceed the horizontal size of the WPF input. If the setting shown in the bottom example in Figure 33.24 is made, VSP2 does not operate correctly.</p> <p>(Note)</p> <p>When the WPFn does not execute 90° or 270° rotation (bit 2 in VI6_WPFn_OUTFMT.ROT is 0) and the WPFn output format is YCbCr4:2:2 or YCbCr4:2:0, specify an even value in these bits. When the WPFn executes 90° or 270° rotation (bit 2 in VI6_WPFn_OUTFMT.ROT is 1) and the WPFn output format is YCbCr4:2:0, specify an even value in these bits.</p> |

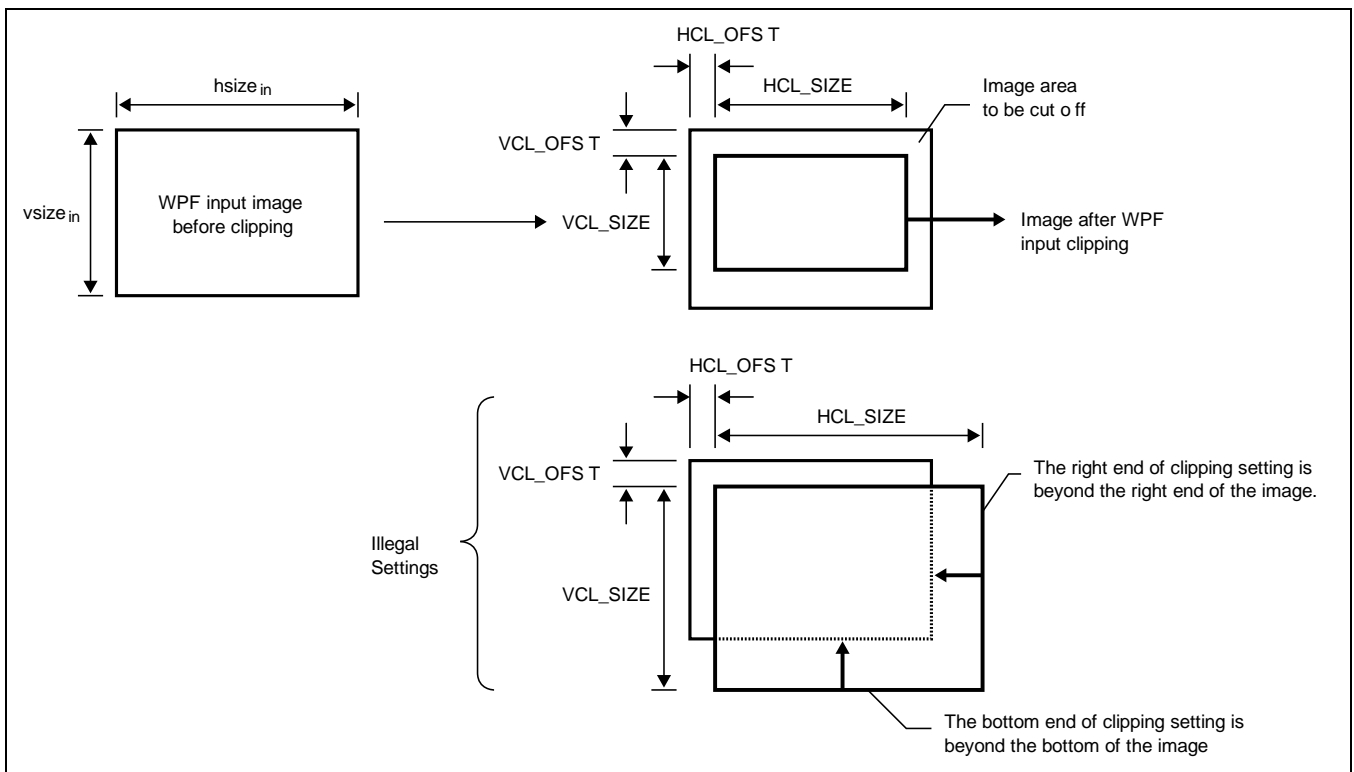
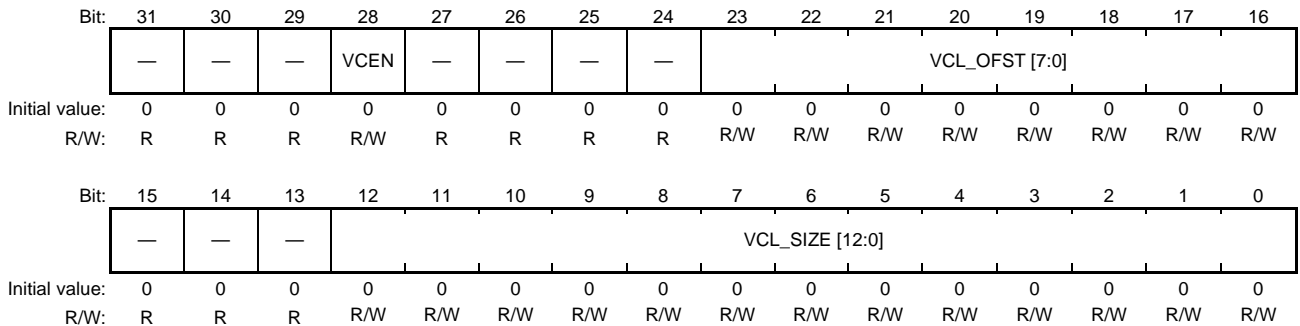


Figure 33.24 Image Clipping in WPF Input Section

33.2.7.3 WPFn Vertical Input Size Clipping Registers (VI6_WPFn_VSZCLIP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|------------------------------|----------------|---------------|-----|--|
| 31 to 29, 27 to 24, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | VCEN | B'0 | R/W | Vertical Size Clipping Enable/Disable Enables or disables clipping of the vertical size of the WPFn input image. 0: Vertical size clipping is disabled 1: Vertical size clipping is enabled |
| 23 to 16 | VCL_OFST [7:0] | All 0 | R/W | Vertical Size Clipping Offset Value Setting These bits specify the offset size (pixels) from the top end of the image in vertical size clipping when the VCEN bit is 1 (Figure 33.24). The top of the image input to the WPF is cut off for the size specified in these bits. When the VCEN bit is 0, this setting is ignored. For WPF0, specify the top offset size (pixels) for the image before rotation. A value from 0 to 255 can be specified. (VCL_OFST + VCL_SIZE) should not exceed the vertical size of the WPF input. If the setting shown in the bottom example in Figure 33.24 is made, VSP2 does not operate correctly. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------|---------------|-----|---|
| 12 to 0 | VCL_SIZE [12:0] | All 0 | R/W | <p>Vertical Clipping Size Setting</p> <p>When the VCEN bit is 1, these bits specify the clipping size for vertical clipping processing. Through this processing, the area of the vertical size specified through the VCL_SIZE bits starting from the offset position specified through the VCL_OFST bits is determined as the valid image area. Accordingly, the bottom pixels beyond the (VCL_OFST + VCL_SIZE) size in the WPFn input image are discarded. When the VCEN bit is 0, this setting is ignored.</p> <p>For WPF0, specify the vertical clipping size for the image before rotation.</p> <p>A value from 1 to 8190 can be specified. (VCL_OFST + VCL_SIZE) should not exceed the vertical size of the WPF input. If the setting shown in the bottom example in Figure 33.24 is made, VSP2 does not operate correctly.</p> <p>(Note)</p> <p>When the WPFn does not execute 90° or 270° rotation (bit 2 in VI6_WPFn_OUTFMT.ROT is 0) and the WPFn output format is YCbCr4:2:0, specify an even value in these bits. When the WPFn executes 90° or 270° rotation (bit 2 in VI6_WPFn_OUTFMT.ROT is 1) and the WPFn output format is YCbCr4:2:2 or YCbCr4:2:0, specify an even value in these bits.</p> |

33.2.7.4 WPFn Output Format Registers (VI6_WPFn_OUTFMT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|-------|------------|-----|------------|-----|-----|-----|-----|-------------|------|-----|-----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | PDV [7:0] | | | | | | | PXA | ODE | — | FCNL | — | ROT [2:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SPYCS | SPUVS | DITH [1:0] | | WRTM [2:0] | | | CSC | — | WRFMT [6:0] | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 21,19,7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 31 to 24 | PDV [7:0] | All 0 | R/W | PAD Value in Output Packed Data These bits specify the value to be stored in the bit field indicated as PAD or P in the output formats shown in Table 33.18. To store this value in PAD, specify 0 in the PXA bit. A value from 0 to 255 can be specified. |
| 23 | PXA | B'0 | R/W | PAD Data Select Selects the value to be stored in the bit field indicated as PAD or P in the packed RGB output formats shown in Table 33.18. Both the value specified in the PDV bits and the α data input from the DPR to WPF are 8 bits, but some of the PAD and P bit fields shown in Table 33.18 are four bits or one bit. When the target bit field is not 8 bits, the number of bits in the PDV value and the α data input from the DPR to WPF is reduced according to the VI6_WPFn_RNDCTRL.ABRM setting. For bit count reduction, refer to Figure 33.25 and the description of VI6_WPFn_RNDCTRL.ABRM. 0: The value specified in the PDV bits is stored in the PAD shown in Table 33.18. 1: The α value output from DPR in pixel units is stored in the PAD shown in Table 33.18. |
| 22 | ODE | B'0 | R/W | Ordered Dither (mode A) Enable/Disable 0: Ordered dither (mode A) is disabled. 1: Ordered dither (mode A) is enabled. When the output format specified through the WRFMT bits is YCbCr, specify 0 in this bit. And when VI6_WPFn_OUTFMT.CSC is set to 1, specify 0 in this bit even in the case that the output format specified through the WRFMT bits is RGB. Ordered dither is available only for 18bpp. So, when ODE bit is equal to 1, set WRFMT at 18bpp format. When ODE bit is equal to 0, WPF dither method is specified by DITH[1:0] in the register Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 20 | FCNL | B'0 | R/W | <p>FCNL compression activation.</p> <p>0: Not compress data with FCNL 1: Compress data with FCNL.</p> <p>FCNL compression is realized in the FCPV, but setting of FCNL is in the VSP2 register. Setting timing of the FCNL should be same with other bit in the register.</p> <p>FCNL compression cannot be enabled for VSPBS, VSPD and VSPDL. Set 0 to this bit for VSPBS, VSPD and VSPDL.</p> <p>See section 35 FCP User's Manual for register setting related to FCNL.</p> |
| 18 to 16 | ROT [2:0] | B'000 | R/W | <p>Rotation Processing Select</p> <p>These bits select the rotation/flipping processing to be applied to the WPFn output image. 90° rotation and 270° rotation are done clockwise. Figure 33.26 and Figure 33.27 show the correspondence between the original image and the rotation/flipping result according to each setting.</p> <p>B'000: No rotation or flipping B'001: Vertical flipping B'010: Horizontal flipping B'011: 180° rotation B'100: 90° rotation B'101: 90° rotation + vertical flipping (equal to 270° rotation + horizontal flipping) B'110: 90° rotation + horizontal flipping (equal to 270° rotation + vertical flipping) B'111: 270° rotation</p> <p>Rotation/flipping can be specified only in VSPI - WPF0. In other channels, refer to section 33.4.5 for restriction of these bits.</p> <p>Note that the destination address setting should be changed according to the setting of these bits. For details, refer to section 33.2.7.10.</p> <p>When the LIF module is used (VI6_LIF_CTRL.LIF_EN = 1), set 0 to ROT [2:0].</p> |
| 15 | SPYCS | B'0 | R/W | <p>WPF Output Mode Setting 1</p> <p>When the output format is YUY2, set this bit to 1 and set the WRFMT bits to 71 (H'47).</p> <p>When the output format is YVYU, set this bit and the SPUVS bit to 1 and set the WRFMT bits to 71 (H'47).</p> <p>In other cases, set this bit to 0.</p> |
| 14 | SPUVS | B'0 | R/W | <p>WPF Output Mode Setting 2</p> <p>When the output format is NV61, set this bit to 1 and set the WRFMT bits to 65 (H'41).</p> <p>When the output format is NV21, set this bit to 1 and set the WRFMT bits to 66 (H'42).</p> <p>When the output format is YVYU, set this bit and the SPYCS bit to 1 and set the WRFMT bits to 71 (H'47).</p> <p>In other cases, set this bit to 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 13, 12 | DITH [1:0] | B'00 | R/W | <p>Ordered Dither (mode B) Enable/Disable</p> <p>When the output format specified through the WRFMT bits is RGB with 18 bpp (260000 colors) or less, the color reduction processing is applied to match the number of colors. The color reduction processing may generate the artifacts of pseudo gradation, which can be suppressed through dithering. The DITH bits enable or disable dithering during color reduction. When the output format specified through the WRFMT bits is YCbCr, specify 0 in these bits.</p> <p>And when VI6_WPFn_OUTFMT.CSC is set to 1, specify 0 in these bits even in the case that the output format specified through the WRFMT bits is RGB.</p> <p>B'00: Dithering (mode B) is disabled B'11: Dithering (mode B) is enabled B'01, B'10: Setting prohibited</p> <p>When ODE bit in the register is 1, set this bit to 0.</p> |
| 11 to 9 | WRTM [2:0] | B'000 | R/W | <p>CSC Conversion Expression Setting</p> <p>These bits select the expression for color space conversion. The conversion direction is RGB to YCbCr when the format specified in the WRFMT bits is YCbCr, or YCbCr to RGB when the format specified in the WRFMT bits is RGB.</p> <p>B'000: BT.601 YCbCr [16,235/240] ↔ RGB [0,255] B'001: BT.601 YCbCr [0,255] ↔ RGB [0,255] B'010: BT.709 YCbCr [16,235/240] ↔ RGB [0,255] B'011: BT.709 YCbCr [16,235/240] ↔ RGB [16,235] B'100 to B'111: Setting prohibited</p> |
| 8 | CSC | B'0 | R/W | <p>Color Space Conversion Setting</p> <p>Enables or disables YCbCr ↔ RGB color space conversion to be executed in the WPFn. The characteristics of color space conversion are determined by the WRTM setting.</p> <p>There are some points to be noted about the relationship between the CSC setting and output format (WRFMT). For details refer to (*1) in section 33.2.6.3, RPFn Input Format Registers (VI6_RPFn_INFMT).</p> <p>0: Color space is not converted. 1: Color space is converted.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------|---------------|-----|--|
| 6 to 0 | WRFMT [6:0] | All 0 | R/W | <p>WPF Output Image Format Setting</p> <p>These bits select the format of the image output from the WPFn to the external memory from among those listed in Table 33.18 and Table 33.19.</p> <p>[Note 1] Number of output pixels</p> <p>When YCbCr4:2:2 is specified through WRFMT, the horizontal size of the output image should be a multiple of 2 pixels. When YCbCr4:2:0 is specified, the vertical and horizontal sizes of the output image should be multiples of 2 pixels. Specify an appropriate data flow of the source RPF → DPR → target WPF so that the size of the image input to the target WPF satisfies the above restrictions. In particular, when the data flow includes a module or a function that modifies (up-scales, down-scales, or clips) the image size, take special care about the module or function settings.</p> <p>In addition, note the rotation function of the WPF shown in Figure 33.10. When 90° or 270° rotation is used (bit 2 in the ROT bits has a value of 1), the horizontal size of the WPF output image is equal to the vertical size of the WPF input image, and the vertical size of the output image is equal to the horizontal size of the input image; that is, the horizontal and vertical sizes are switched between the input and output. For example, because of the above restriction on YCbCr4:2:2 that the horizontal output size should be a multiple of 2 pixels, the vertical size of the WPF input should be a multiple of 2 pixels.</p> <p>[Note 2] Output lines in YCbCr4:2:0</p> <p>In the YCbCr4:2:0 output format, the number of chrominance lines in the vertical direction is one-half the number of luminance lines. For this reason, the WPF outputs only even-numbered chrominance lines (lines 0, 2, 4, 6, ...) (conversion from (A) to (B) in. When vertical flipping is also specified through the ROT bits, the flipping processing is executed last and the chrominance line locations are inverted (lines 1, 3, 5, 7, ...) in the output image ((C) in Figure 33.28).</p> <p>[Note 3] Down sampling of CbCr in horizontal direction in YCbCr4:2:0 or YCbCr4:2:2</p> <p>In the YCbCr4:2:2 or YCbCr4:2:0 output format, method of down sample of Cb/Cr in horizontal direction is average of neighbor two pixels.</p> |

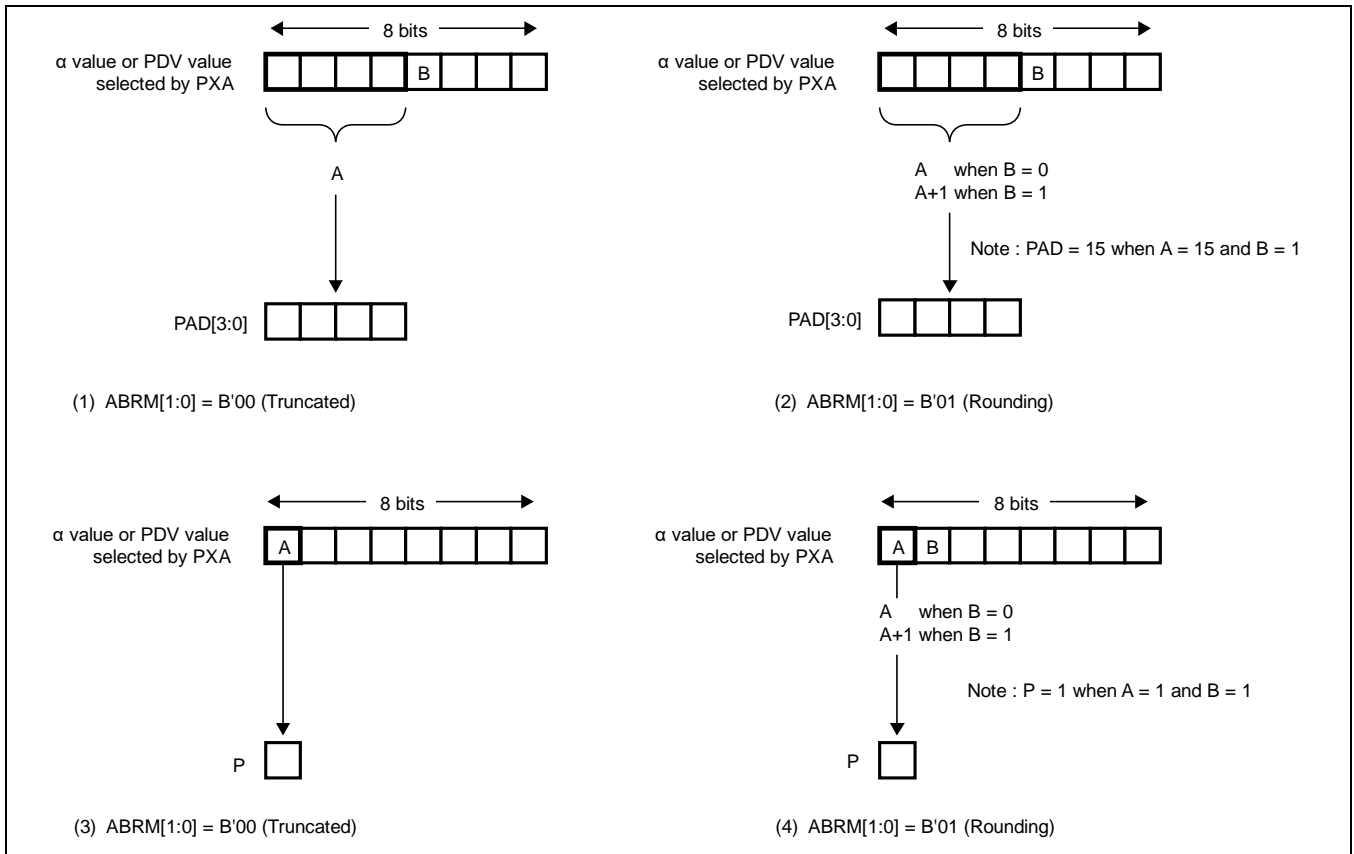


Figure 33.25 Selection of PAD Value and Reduction of Bit Count through PXA Setting

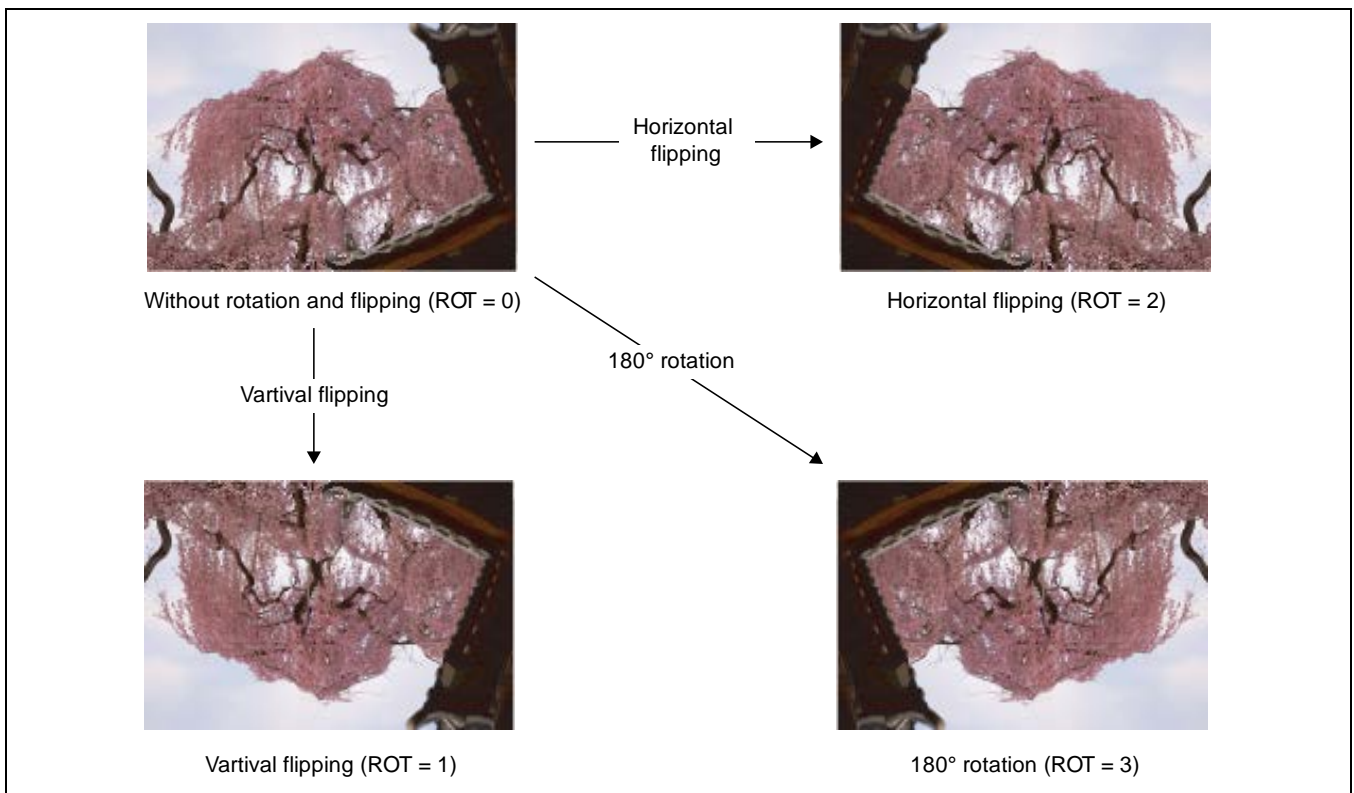


Figure 33.26 Correspondence between Original Image and Rotation/Flipping Result according to ROT Setting (1)

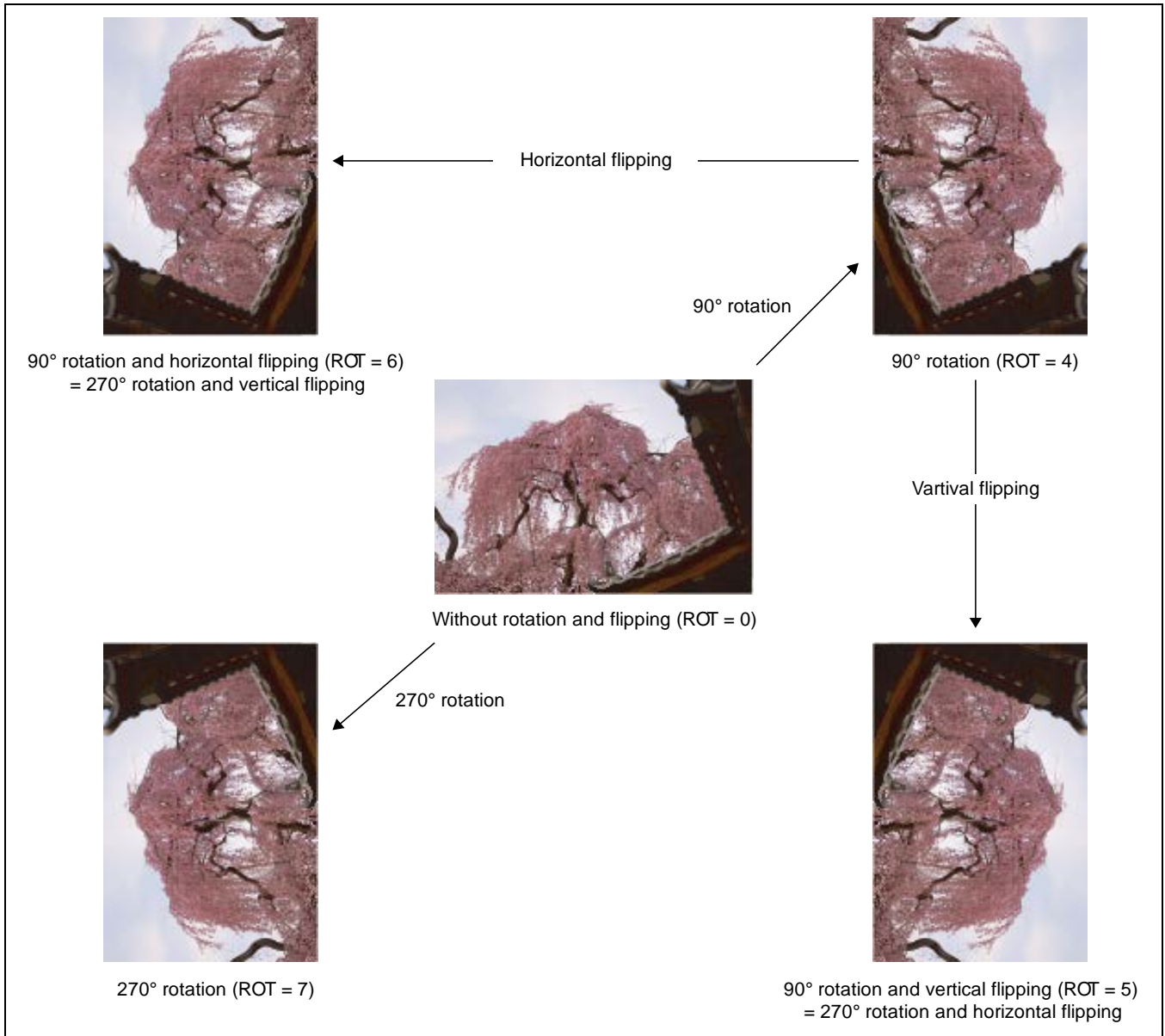


Figure 33.27 Correspondence between Original Image and Rotation/Flipping Result according to ROT Setting (2)

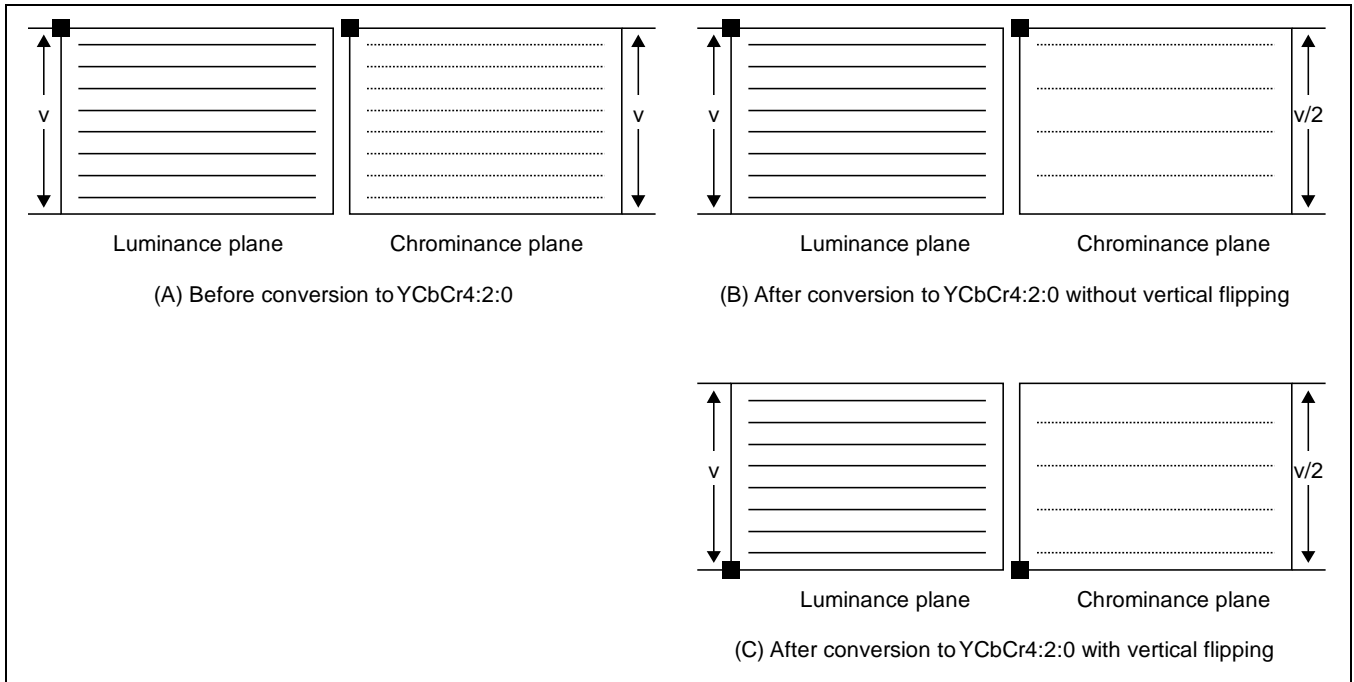


Figure 33.28 Chrominance Output Lines in YCbCr4:2:0 and Vertical Flipping Result

Table 33.19 Packed YCbCr Formats for WPF Output

| WRFMT [6:0] | Packed YCbCr Output Format | Reference |
|--------------|--|----------------|
| H'40 | YCbCr4:4:4 semi-planar | Figure 33.15*4 |
| H'41 | YCbCr4:2:2 semi-planar (NV16, NV61*1) | |
| H'42 | YCbCr4:2:0 semi-planar (NV12, NV21*1) | |
| H'43 to H'45 | Reserved | — |
| H'46 | YCbCr4:4:4 interleaved | Figure 33.16*4 |
| H'47 | YCbCr4:2:2 interleaved type 0 (UYVY, YUY2*2, YVYU*3) | |
| H'48 | YCbCr4:2:2 interleaved type 1 | |
| H'49 | YCbCr4:2:0 interleaved *5 | |
| H'4A | YCbCr4:4:4 planar | Figure 33.17*4 |
| H'4B | YCbCr4:2:2 planar (YV16) | |
| H'4C | YCbCr4:2:0 planar (YV12, I420) | |
| H'4D to H'7F | Reserved | — |

- Notes: 1. When the output format is NV61 or NV21, set SPUVS (bit 14) to 1.
 2. When the output format is YUY2, set SPYCS (bit 15) to 1.
 3. When the output format is YVYU, set SPUVS (bit 14) to 1 and SPYCS (bit 15) to 1.
 4. Figure 33.18 shows the definition of memory address for each pixel in Figure 33.15 to Figure 33.17.
 5. Each line of plane is written twice, so byte/pixel of YCbCr420ITL is 3 byte/pixel (same as YCbCr444ITL).

For details of each YCbCr format, refer to Figure 33.15, Figure 33.16, Figure 33.17, and Figure 33.18. In these figures, registers for the RPF are indicated; read them as registers for the WPF as follows.

(RPF registers in the figures) → (Corresponding WPF registers)

VI6_RPFn_SRCM_ADDR_Y.SRCM_ADDR_Y → VI6_WPFn_DSTN_ADDR_Y.DSTM_ADDR_Y

VI6_RPFn_SRCM_ADDR_C0.SRCM_ADDR_C0 → VI6_WPFn_DSTN_ADDR_C0.DSTM_ADDR_C0

VI6_RPFn_SRCM_ADDR_C1.SRCM_ADDR_C1 → VI6_WPFn_DSTN_ADDR_C1.DSTM_ADDR_C1

33.2.7.5 WPFn Data Swapping Registers (VI6_WPFn_DSWAP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | P_LLS | P_LWS | P_WDS | P_BTS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 | P_LLS | B'0 | R/W | WPF Output Data Swapping in LONG LWORD Units The effect of this bit setting is the same as data swapping in the RPF; refer to Table 33.13. 0: Data swapping in LONG LWORD (64-bit) units is disabled 1: Data swapping in LONG LWORD (64-bit) units is enabled |
| 2 | P_LWS | B'0 | R/W | WPF Output Data Swapping in long word Units The effect of this bit setting is the same as data swapping in the RPF; refer to Table 33.13. 0: Data swapping in long word (32-bit) units is disabled 1: Data swapping in long word (32-bit) units is enabled |
| 1 | P_WDS | B'0 | R/W | WPF Output Data Swapping in Word Units The effect of this bit setting is the same as data swapping in the RPF; refer to Table 33.13. 0: Data swapping in word (16-bit) units is disabled 1: Data swapping in word (16-bit) units is enabled |
| 0 | P_BTS | B'0 | R/W | WPF Output Data Swapping in Byte Units The effect of this bit setting is the same as data swapping in the RPF; refer to Table 33.13. 0: Data swapping in byte (8-bit) units is disabled 1: Data swapping in byte (8-bit) units is enabled |

Table 33.13 shows the data order before and after swapping according to the long long word, long word, word, and byte swapping settings.

When data order in memory for each format is the same as in Table 33.18 for RGB format and Figure 33.15 to Figure 33.17 for YCbCr format, set 4'b1111 to {*_LLS, *_LWS, *_WDS, *_BTS}. If data order is not the same as the definition, change data order within 16byte unit by these bits as shown in Table 33.13.

Note: When writing data is FCNL compressed data, follow section 35 FCP User's Manual about setting this swap register.

33.2.7.6 WPFn Rounding Control Registers (VI6_WPFn_RNDCTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|------------|------|----|----|------------|---------------|-----|-----|-----|-----|-----|-----|-----|-----|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | CBRM | — | — | ABRM [1:0] | ATHRESH [7:0] | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | CLMD [1:0] | | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----------------------------------|---------------|---------------|-----|---|
| 31 to 29, 27, 26, 15, 14, 11 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | CBRM | B'0 | R/W | Bit Count Reduction Method Selection for Data Storage in Packed RGB This bit specifies the method for reducing the number of bits when data is stored in the bit fields indicated as R, G, and B in Table 33.18 and the target bit fields are not 8 bits. 0: Bit count conversion: The lower-order bits are truncated 1: Bit count conversion: Rounding (rounding off) |
| 25, 24 | ABRM [1:0] | B'00 | R/W | Bit Count Reduction Method Selection for Data Storage in PAD These bits specify the method for reducing the number of bits when the data selected through VI6_WPFn_OUTFMT.PXA is stored in the bit fields indicated as PAD or P in Table 33.18 and the target bit field is four bits or one bit. A value of B'10 can be specified only when the packed RGB format specified through VI6_WPFn_OUTFMT.WRFMT includes a 1-bit P field. In this case, when the data selected through VI6_WPFn_OUTFMT.PXA is greater than the ATHRESH value, 1 is stored in the P field; when the selected data is not greater than the ATHRESH value, 0 is stored. B'00: Bit count conversion: The lower-order bits are truncated B'01: Bit count conversion: Rounding (rounding off) B'10: Bit count conversion: Comparison with the threshold value (this setting is allowed only when the storage field is one bit) B'11: Setting prohibited |
| 23 to 16 | ATHRESH [7:0] | All 0 | R/W | Threshold for Conversion to 1-Bit α Data These bits specify the threshold value used for conversion from 8-bit α data to one bit when the ABRM bits are set to B'10. When the 8-bit α value before bit count reduction is equal to or smaller than the ATHRESH value, 0 is stored as the reduced 1-bit α data. In other cases, 1 is stored as the 1-bit α data. A value from 0 to 255 can be specified. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------|---------------|-----|--|
| 13, 12 | CLMD [1:0] | B'00 | R/W | <p>Color Data Clipping</p> <p>These bits specify the method for clipping the YCbCr color data output from the WPF. When RGB color data is output from the WPF, specify 0 in these bits.</p> <p>B'00: Output value is not clipped (0-255)</p> <p>B'01: Output value is clipped:</p> <p>YCbCr mode 1 (16-235 (Y), 16-240 (Cb/Cr))</p> <p>B'10: Output value is clipped: YCbCr mode 2 (Y/Cb/Cr = 1-254)</p> <p>B'11: Setting prohibited</p> |

33.2.7.7 WPF0 Rotation Control Registers (VI6_WPF0_ROT_CTRL)

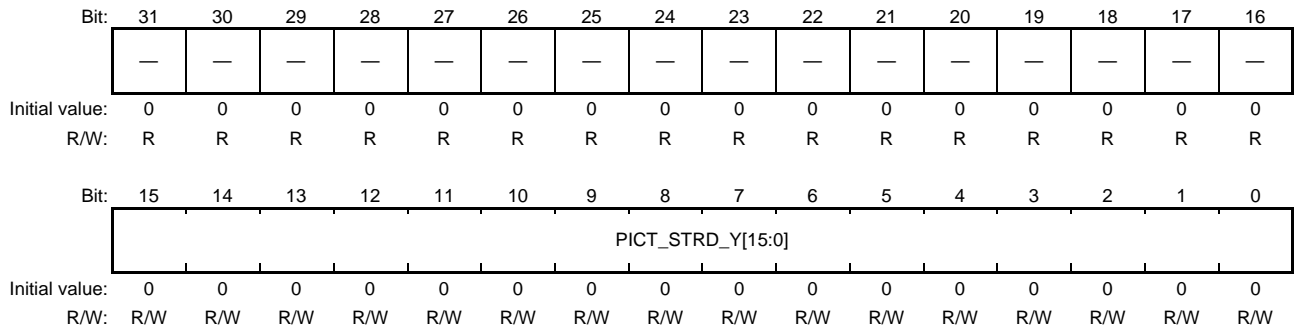
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | LN16 | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | LMEM_WD [12:0] | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|----------------|---------------|-----|--|
| 31 to 18, 16 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 17 | LN16 | B'0 | R/W | 16-Line Rotation Mode Selects the mode for rotation processing. The 16-line rotation mode can improve the bus transfer efficiency in rotation processing, but the maximum horizontal image size that can be rotated is one half of that in 8-line rotation mode. 0: 8-line rotation mode 1: 16-line rotation mode Set 1 to this bit for transfer efficiency. |
| 12 to 0 | LMEM_WD [12:0] | All 0 | R/W | Rotation Memory Word Count Always specify 256 here. |

33.2.7.8 WPFn Destination Y Plane Memory Stride Registers (VI6_WPFn_DSTM_STRIDE_Y)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | PICT_STRD_Y [15:0] | All 0 | R/W | Memory Stride of Destination Picture Y/RGB Plane These bits specify in 1-byte units the memory stride of the destination picture in the external memory to be written to by the WPFn as shown in Figure 33.29. A value from H'0000 to H'FFFF can be specified. |

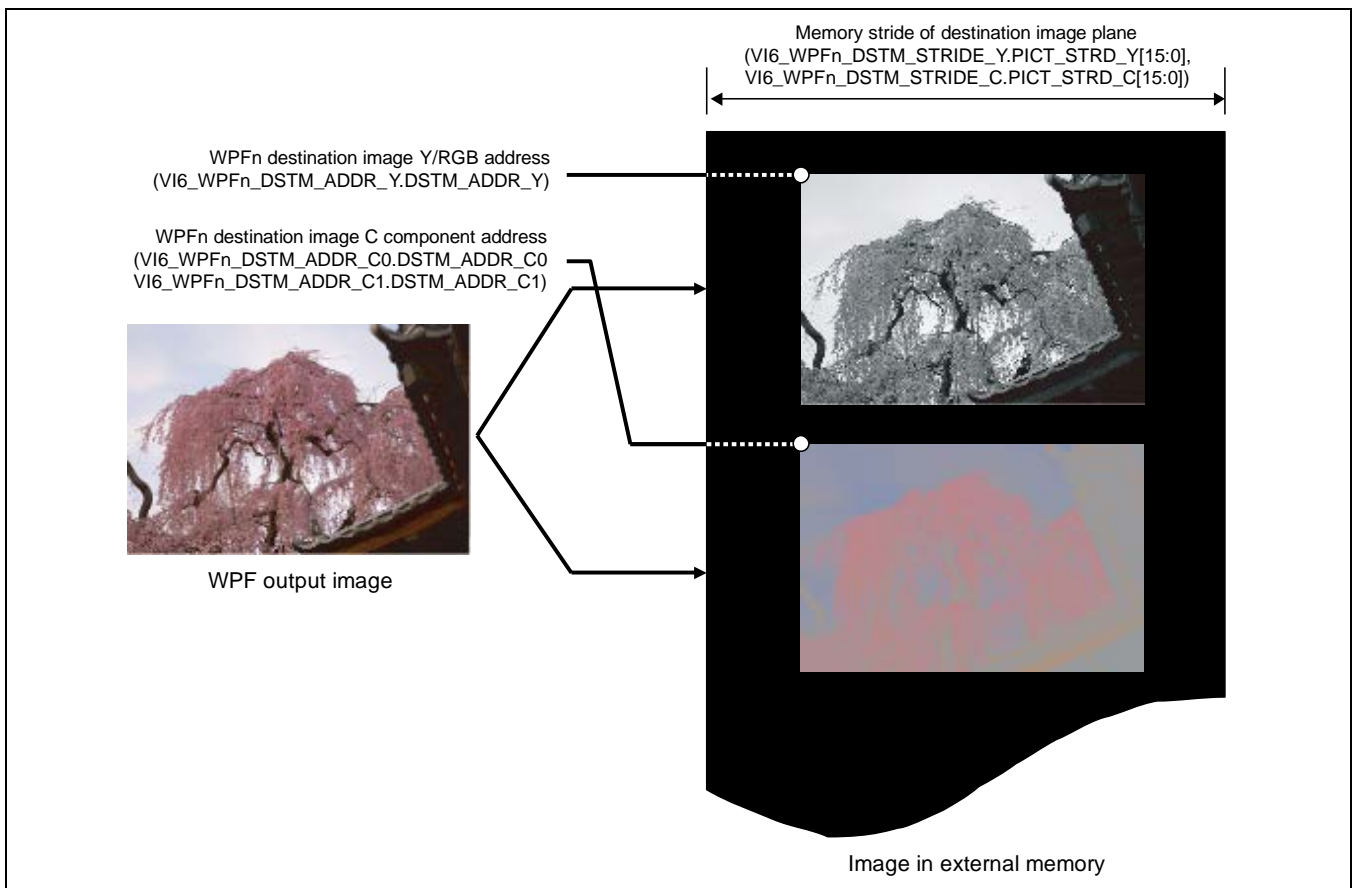
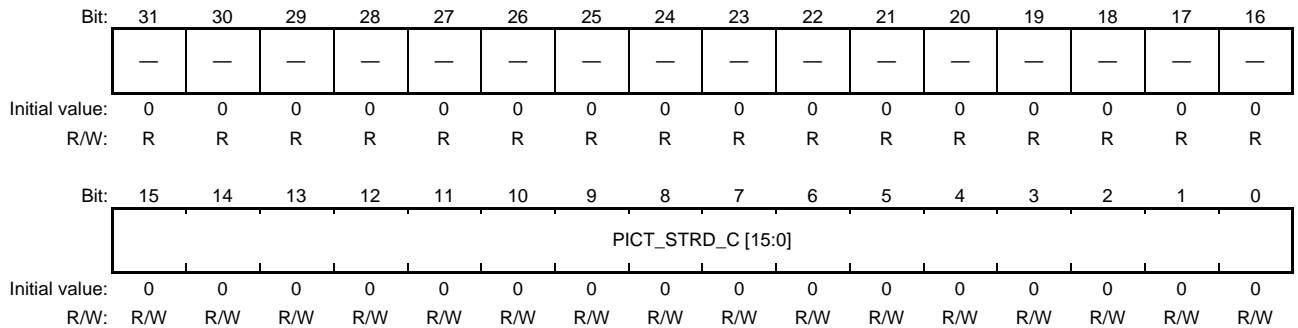


Figure 33.29 Writing Image Data to Destination Area in WPFn

33.2.7.9 WPFn Destination C Plane Memory Stride Registers (VI6_WPFn_DSTM_STRIDE_C)

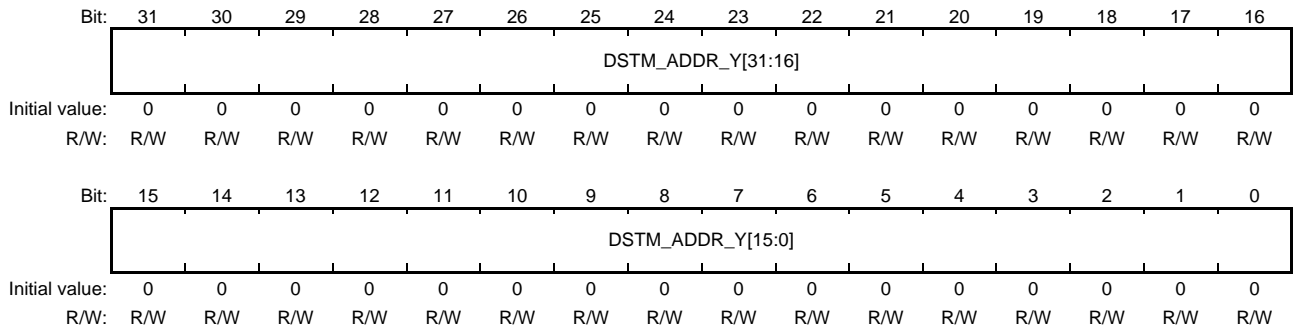
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | PICT_STRD_C [15:0] | All 0 | R/W | Memory Stride of Destination Picture C Plane These bits specify in 1-byte units the memory stride for the C plane of the destination picture in the external memory to be written to by the WPFn as shown in Figure 33.29. When the WPFn outputs images in an RGB format, this setting is not used. When the WPFn outputs images in YCbCr planar format, this setting is applied to both the Cb and Cr planes. A value from H'0000 to H'FFFF can be specified. |

33.2.7.10 WPFn Destination Y/RGB Address Registers (VI6_WPFn_DSTM_ADDR_Y)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------|---------------|-----|---|
| 31 to 0 | DSTM_ADDR_Y [31:0] | All 0 | R/W | <p>Destination Image Y/RGB Plane Storing Address</p> <p>These bits specify in 1-byte units the address for storing the destination-image Y plane or packed RGB plane to be written to by the WPFn in the method described later.</p> <p>A value from H'0000_0000 to H'FFFF_FFFF can be specified.</p> |

(Destination Address Specification Method)

When rotation/flipping is not used, the start address of a frame (address FHA shown in Figure 33.30) should be specified as the destination address. When flipping/rotation is used, the destination address is not the frame start address (FHA); one of addresses A₀ to A₃ shown in Figure 33.30 should be selected according to the combination of desired rotation and flipping (VI6_WPFn_OUTFMT.ROT setting).

To strictly define locations A₀ to A₃, let the horizontal size of the output image after rotation be H, the vertical size of the output image after rotation be V, and the memory stride (VI6_WPFn_DSTM_STRIDE_Y/C setting) be S as shown in Figure 33.30. Calculate the destination address (one of A₀ to A₃) using the formula shown in Table 33.20 and specify it in the destination address storing register.

The values of variables N, P, and L in Table 33.20 depend on the other register settings and luminance and chrominance components. These values should be obtained by referring to Table 33.21 to Table 33.23 when calculating the address to be specified in VI6_WPFn_DSTM_ADDR_Y, or Table 33.21 and Table 33.24 to Table 33.25 when calculating the address to be specified in VI6_WPFn_DSTM_ADDR_C0 or VI6_WPFn_DSTM_ADDR_C1.

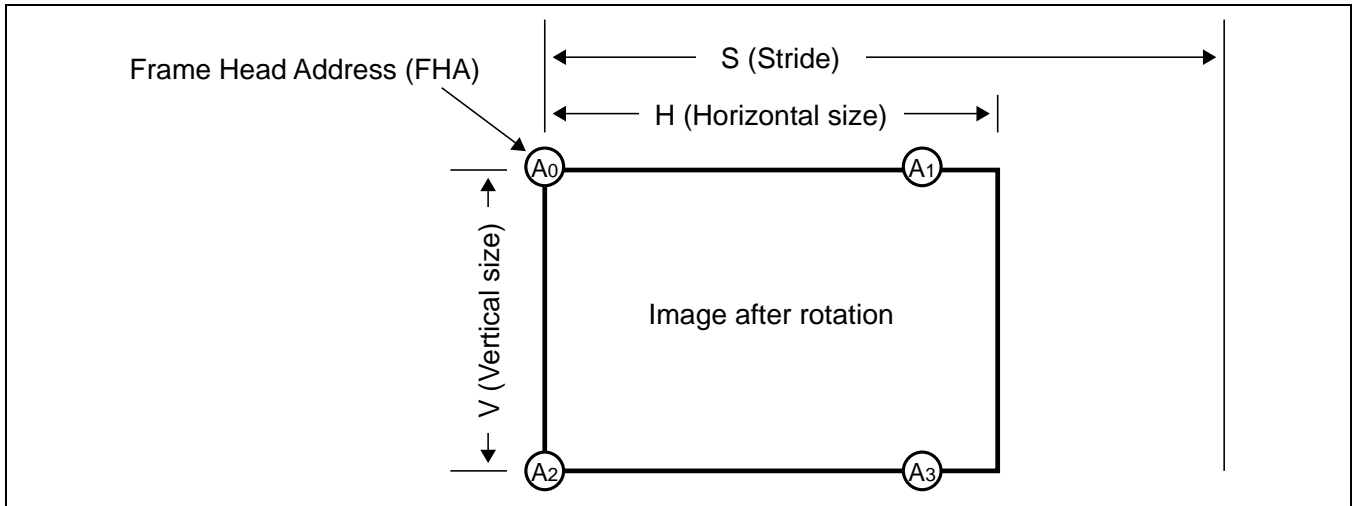


Figure 33.30 Location of Destination Address to be Specified for Rotation/Flipping

Table 33.20 Destination Address A0, A1, A2, and A3 Calculation Formulas

| VI6_WPFn_OUTFMT.ROT Setting | Formula for Calculating Address to be Set in VI6_WPFn_DSTM_ADDR_Y, VI6_WPFn_DSTM_ADDR_C0, and VI6_WPFn_DSTM_ADDR_C1 |
|-----------------------------|---|
| 0, 2, or 6 | $A_0 = \text{FHA}$ |
| 4 | $A_1 = \text{FHA} + \text{clip0}((H - N) \times P)$ |
| 1, 3, or 7 | $A_2 = \text{FHA} + (V \times L - 1) \times S$ |
| 5 | $A_3 = \text{FHA} + (V \times L - 1) \times S + \text{clip0}((H - N) \times P)$ |

Table 33.21 Value of N according to VI6_WPFn_ROT_CTRL.LN16 Setting (Used in Common for RGB, Y, and C)

| VI6_WPFn_ROT_CTRL.LN16 | N |
|------------------------|----|
| 0 | 8 |
| 1 | 16 |

Table 33.22 Value of P according to VI6_WPFn_OUTFMT.WRFMT Setting (for RGB and Luminance Y Address Calculation)

| VI6_WPFn_OUTFMT.WRFMT | P |
|--|---|
| 0, 64 to 66, or 74 to 76 | 1 |
| 1 to 2, 4 to 6, 25 to 32, or 71 to 72, | 2 |
| 15 to 18, 21, 24, 33, 70, or 73 | 3 |
| 7 to 14, 19 to 20, 22 to 23, or 34 to 35 | 4 |

Table 33.23 Value of L according to VI6_WPFn_OUTFMT.WRFMT Setting (for RGB and Luminance Y Address Calculation)

| VI6_WPFn_OUTFMT.WRFMT | L |
|--|-----|
| 73 | 0.5 |
| 0 to 2, 4 to 35, 64 to 66, 70 to 72, or 74 to 76 | 1 |

**Table 33.24 Value of P according to VI6_WPFn_OUTFMT.WRFMT Setting
(for Chrominance C0 and C1 Address Calculation)**

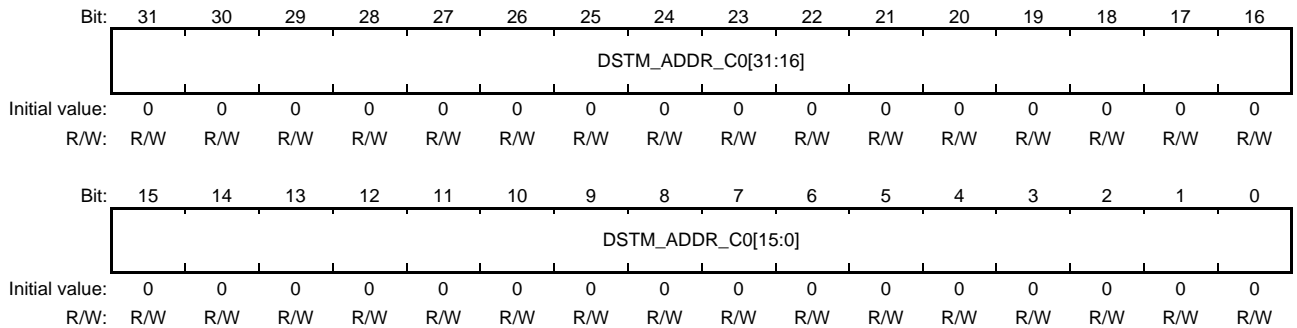
| VI6_WPFn_OUTFMT.WRFMT | P |
|-----------------------|-------------|
| 70 to 73 | Not defined |
| 75 to 76 | 0.5 |
| 65 to 66, 74 | 1 |
| 64 | 2 |

**Table 33.25 Value of L according to VI6_WPFn_OUTFMT.WRFMT Setting
(for Chrominance C0 and C1 Address Calculation)**

| VI6_WPFn_OUTFMT.WRFMT | L |
|-----------------------|-------------|
| 70 to 73 | Not defined |
| 66, 76 | 0.5 |
| 64 to 65, 74 to 75 | 1 |

33.2.7.11 WPFn Destination Chroma Address Registers 0 (VI6_WPFn_DSTM_ADDR_C0)

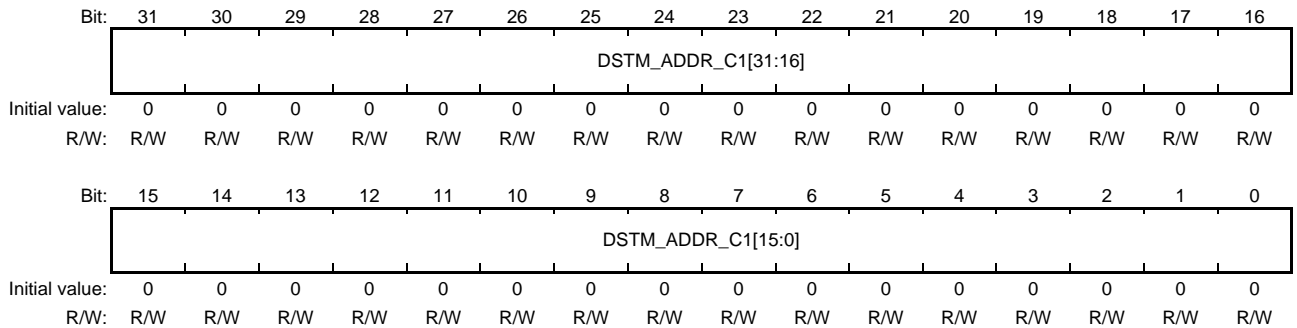
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------|---------------|-----|---|
| 31 to 0 | DSTM_ADDR_C0 [31:0] | All 0 | R/W | <p>Destination Image C Plane Storing Address 0</p> <p>These bits specify in 1-byte units the address for storing the destination-image C plane to be written to by the WPFn. Refer to the description of VI6_WPFn_DSTM_ADDR_Y for settings.</p> <p>Here, the C plane indicates the combined CbCr plane when a semi-planar format is selected from the packed YCbCr formats shown in Table 33.19 or the Cb plane when a planar format is selected. When an interleaved format is selected or the output is in an RGB format, this setting is not used.</p> <p>A value from H'0000_0000 to H'FFFF_FFFF can be specified. Refer to Figure 33.30 in section 33.2.7.10 for settings.</p> |

33.2.7.12 WPFn Destination Chroma Address Registers 1 (VI6_WPFn_DSTM_ADDR_C1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------|---------------|-----|---|
| 31 to 0 | DSTM_ADDR_C1 [31:0] | All 0 | R/W | <p>Destination Image C Plane Storing Address 1</p> <p>These bits specify in 1-byte units the address for storing the Cr plane when the WPFn outputs images to the external memory in YCbCr planar format shown in Table 33.19. Refer to the description of VI6_WPFn_DSTM_ADDR_Y for settings.</p> <p>This setting is not used when the WPF outputs in YCbCr format that is not a planar format or in an RGB format.</p> <p>A value from H'0000_0000 to H'FFFF_FFFF can be specified. Refer to Figure 33.30 in section 33.2.7.10 for settings.</p> |

33.2.7.13 WPFn LIF Write Back Control Registers (VI6_WPFn_WRBCK_CTRL: n = 0, 1)

[for n = 0]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[for n = 1]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | WBMD[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1, 0 | WBMD[1:0] | B'00 | R/W | Display Data Write Back Control This bit is used for selecting the write back mode when the value of VI6_LIFn_CTRL.LIF_EN bit is set to 1. 0: Write Back Disabled 1: Write Back Enabled |

33.2.8 DPR Control Registers

33.2.8.1 Concept of DPR Settings

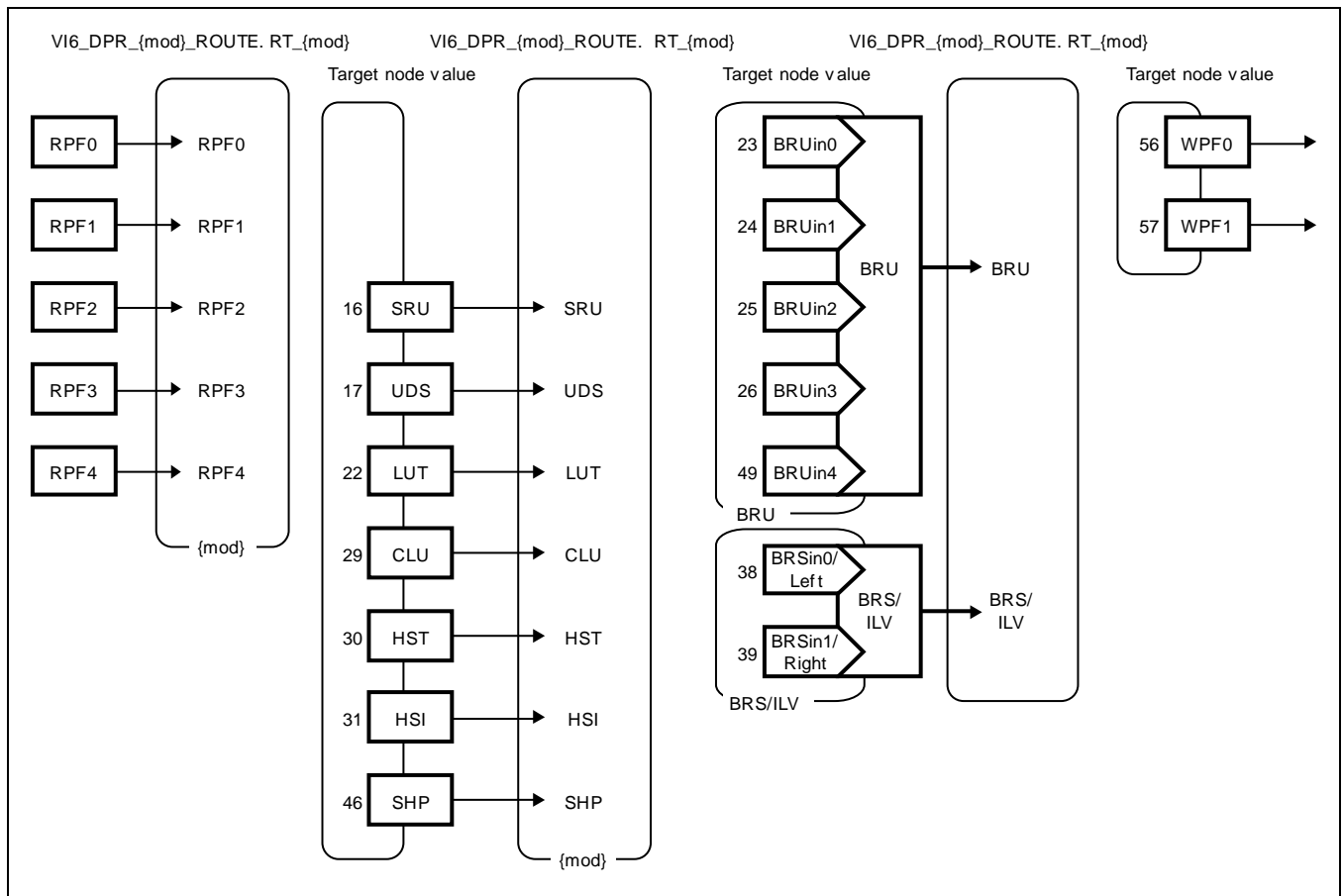


Figure 33.31 Node Register Names and Target Node Values on Data Path Router [RZ/G2H, RZ/G2N]

Note: * Figure 33.31 shows node value of all sub modules implemented in all VSPs of RZ/G2H, RZ/G2N.

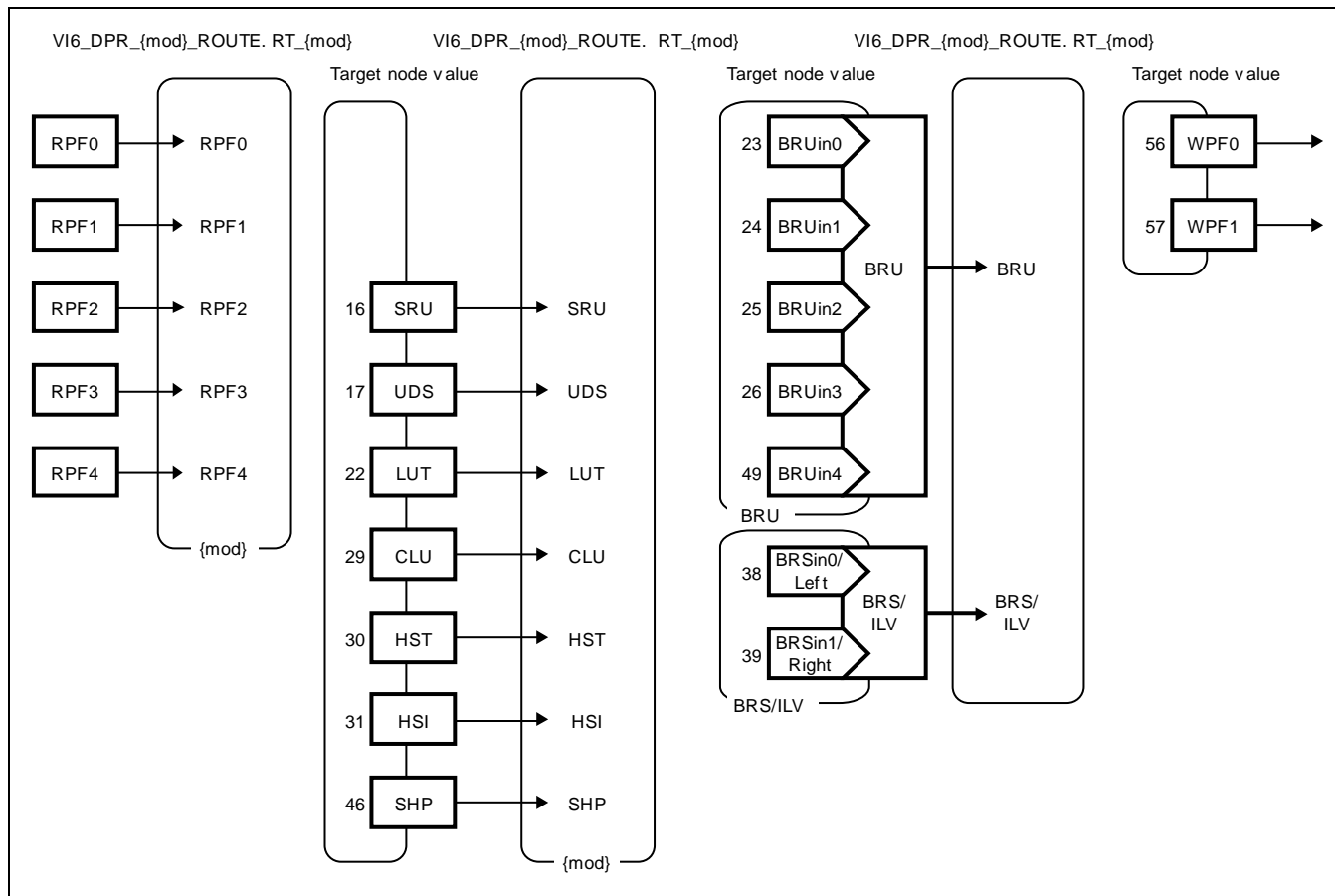


Figure 33.32 Node Register Names and Target Node Values on Data Path Router [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]

Note: * Figure 33.32 shows node value of all sub modules implemented in all VSPs of RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E.

In the VSP2 internal data path, the order of processes can be specified as desired. The module for performing each process has a unique node value. Set each bit field in VI6_DPR_*_ROUTE to an appropriate node value shown in Figure 33.31 (RZ/G2H, RZ/G2N), Figure 33.32 (RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E) to specify the target node to be connected behind each module.

For DPR settings, all of the following restrictions should be observed. If any of them is violated, even the WPF paths operating correctly at that time will be affected as well as the WPF paths connected through the DPR, and correct operation will not be guaranteed (for example, if a restriction is violated for the DPR setting related to WPF1, even WPF0 that are operating correctly will be affected).

1. Specify 63 for the output node values of all RPFs and processing modules that are not used in the DPR. Here, make sure that no module is connected to a module for which 63 is specified as the node value.

Note: VI6_DPR_{mod}_ROUTE and VI6_DPR_{mod}_SMPPT corresponding to {mod} are not implemented in VSP which doesn't have {mod}. VI6_DPR_RPFn_ROUTE (n = 1 to 4) corresponding to RPFn is not implemented in VSP which doesn't have RPFn (n = 1 to 4). See Figure 33.1 to Figure 33.6 about module configuration of each VSP.

2. When specifying a value other than 63 for an output node value in the DPR, make sure that valid inputs (RPF0 to RPF4 or virtual RPF) and a target WPF are determined.
3. Only one module can be connected to each module; specifying the same target node value for two or more modules is prohibited.

4. Desired modules can be connected between each RPF and BRU input port, but all RPFs specified as the sources for a BRU input port should have the same target WPF. Same manner must be applied for ILV and BRS.
5. Make appropriate routing or RPF register settings so that the color space formats (RGB/YCbCr) for all BRU input ports are the same. Same manner must be applied for ILV and BRS.
6. Do not connect the output of any module as the input to the same module (in the BRU case, any input port) even when there is another module between the output and input (creating a loop is prohibited).
7. Each node can be used only once throughout all paths from RPF_n to WPF_n. When a module shown in Figure 33.31 is assigned in one RPF → WPF path, it cannot be used in another RPF → WPF path.
8. While a WPF is operating, modifying the DPR connection settings in VI6_DPR_*_ROUTE is prohibited for modules used by the WPF but allowed for modules not used by the WPF. Be careful not to accidentally modify the settings of the modules included in the WPF path that is operating.
9. Be careful to do the following setting when ILV or BRS is used.
 VI6_DPR_ILV_BRS_ROUTE.BRSSEL = 0 (ILV is used)
 VI6_DPR_ILV_BRS_ROUTE.BRSSEL = 1 (BRS is used)
 BRS and ILV can be used exclusively.

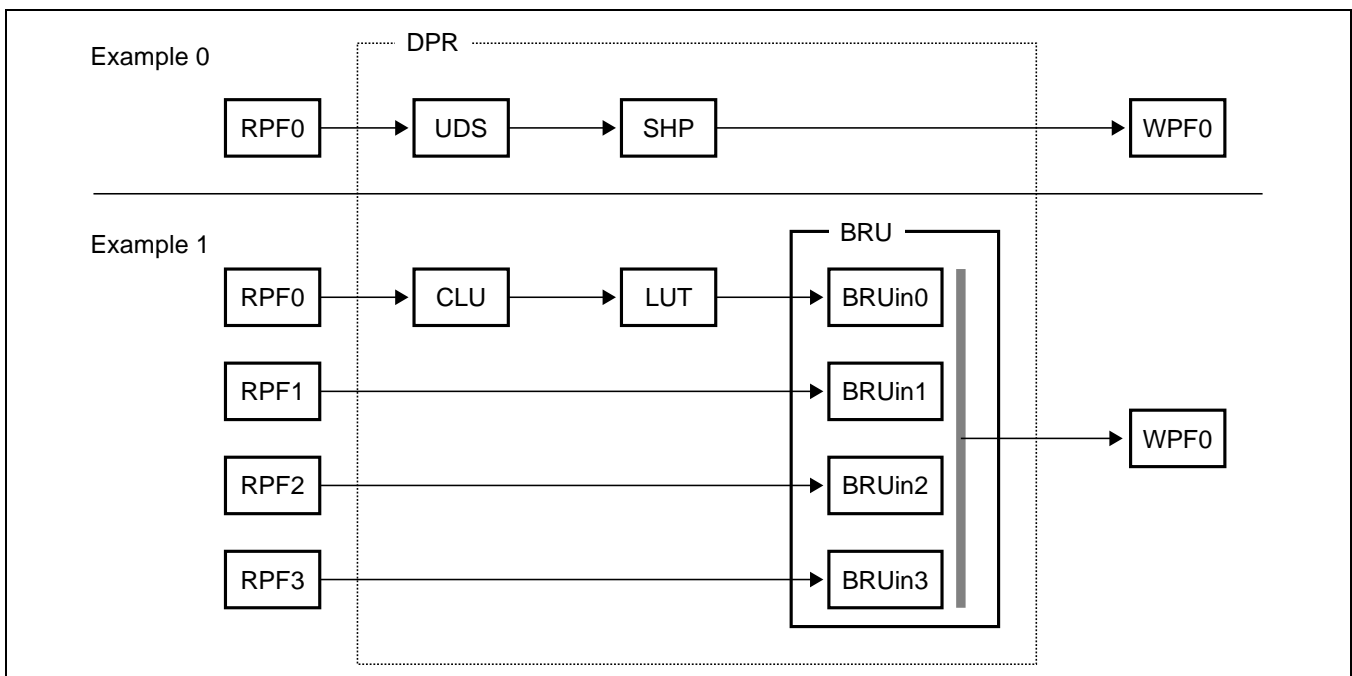


Figure 33.33 Examples of Internal Data Path Routing

Figure 33.33 shows examples of internal data path routing. Example 0 is WPF0 processing (RPF0 is the source RPF), and example 1 is also WPF0 (RPF0 to RPF3 are the source RPFs). Each example has the configuration shown in Figure 33.33. Example 0 performs UDS (up/down scaling) and SHP (sharpness) processing. Example 1 performs CLU and LUT processing (e.g. γ correction), for input 0 (RPF0) and then applies blending or raster operation between the resultant data and input data 1 to 3 (RPF1 to RPF3). The VI6_DPR_*_ROUTE settings for these examples are shown in table 33.26. The bit fields for the modules that are not used in the examples should be set to 63.

Table 33.26 VI6_DPR*_ROUTE Register Settings in Connection Examples Shown in Figure 33.33

| | Register Name | Setting |
|------------------|----------------------------|----------------|
| Example 0 | VI6_DPR_RPF0_ROUTE.RT_RPF0 | 17 (To UDS) |
| | VI6_DPR_RPF1_ROUTE.RT_RPF1 | 63 (UNUSED) |
| | VI6_DPR_RPF2_ROUTE.RT_RPF2 | 63 (UNUSED) |
| | VI6_DPR_RPF3_ROUTE.RT_RPF3 | 63 (UNUSED) |
| | VI6_DPR_RPF4_ROUTE.RT_RPF4 | 63 (UNUSED) |
| | VI6_DPR_SRU_ROUTE.RT | 63 (UNUSED) |
| | VI6_DPR_UDS_ROUTE.RT | 46 (To SHP) |
| | VI6_DPR_UIF4_ROUTE.RT | 63 (UNUSED) |
| | VI6_DPR_SHP_ROUTE.RT | 56 (To WPF0) |
| | VI6_DPR_LUT_ROUTE.RT | 63 (UNUSED) |
| | VI6_DPR_CLU_ROUTE.RT | 63 (UNUSED) |
| | VI6_DPR_HST_ROUTE.RT | 63 (UNUSED) |
| | VI6_DPR_HSI_ROUTE.RT | 63 (UNUSED) |
| | VI6_DPR_ILV_BRS_ROUTE.RT | 63 (UNUSED) |
| | VI6_DPR_BRU_ROUTE.RT | 63 (UNUSED) |
| Example 1 | VI6_DPR_RPF0_ROUTE.RT_RPF0 | 29 (To CLU) |
| | VI6_DPR_RPF1_ROUTE.RT_RPF1 | 24 (To BRUin1) |
| | VI6_DPR_RPF2_ROUTE.RT_RPF2 | 25 (To BRUin2) |
| | VI6_DPR_RPF3_ROUTE.RT_RPF3 | 26 (To BRUin3) |
| | VI6_DPR_RPF4_ROUTE.RT_RPF4 | 63 (UNUSED) |
| | VI6_DPR_SRU_ROUTE.RT | 63 (UNUSED) |
| | VI6_DPR_UDS_ROUTE.RT | 63 (UNUSED) |
| | VI6_DPR_UIF4_ROUTE.RT | 63 (UNUSED) |
| | VI6_DPR_SHP_ROUTE.RT | 63 (UNUSED) |
| | VI6_DPR_LUT_ROUTE.RT | 56 (To WPF0) |
| | VI6_DPR_CLU_ROUTE.RT | 22 (To LUT) |
| | VI6_DPR_HST_ROUTE.RT | 63 (UNUSED) |
| | VI6_DPR_HSI_ROUTE.RT | 63 (UNUSED) |
| | VI6_DPR_ILV_BRS_ROUTE.RT | 63 (UNUSED) |
| | VI6_DPR_BRU_ROUTE.RT | 63 (UNUSED) |

Although both examples shown in figure 33.33 include operation in image processing modules, connect the RPF to the WPF directly when only image format conversion or packed format conversion is required.

33.2.8.2 RPFn Routing Register (VI6_DPR_RPFn_ROUTE : n = 0, 1, 2, 3, 4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|--------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | RT_RPFn[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 31 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | RT_RPFn [5:0] | All 0 | R/W | RPFn Target Node Value These bits specify the target node value for RPFn. When using RPFn, refer to Figure 33.31 to Figure 33.32 for settings. When RPFn is not started through the VI6_WPFn_SRCRPF setting, specify 63. |

33.2.8.3 WPFn Timing Control Register (VI6_DPR_WPFn_FPORCH: n = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|--------------|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | FP_WPFn[5:0] | | | | | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|------------------|---------------|---------------|-----|--|
| 31 to 14, 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 to 8 | FP_WPFn [5:0] | All 0 | R/W | WPFn Internal Operation Timing Setting Specify 5. |

33.2.8.4 {mod} Routing Register (VI6_DPR_{mod}_ROUTE: {mod} = ILV_BRS, SRU, UDS, LUT, CLU, HST, HSI, BRU, SHP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|---------|------------|-----|-----|-----|-----|----------|---------|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | BRS SEL | — | — | — | — | FXA[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | FP[5:0] | | | | | — | — | RT[5:0] | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--|-----------|---------------|-----|---|
| 31 to 29, 27 to 24, 15, 14, 7, 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | BRSSSEL | B'0 | R/W | Select ILV or BRS 0: ILV is selected 1: BRS is selected. BRS and ILV shares data path route. Setting VI6_DPR_ILV_BRS_ROUTE is needed in case of using ILV or BRS. This bit select ILV or BRS for VI6_DPR_ILV_BRS_ROUTE. Set 0 to this bit in case of using ILV. Set 1 to this bit in case of using BRS. This bit is available in only VI6_DPR_ILV_BRS_ROUTE. |
| 23 to 16 | FXA [7:0] | All 0 | R/W | Fixed α Output Value for {mod} The {mod} does not support input/output of the α value. The α value input to the {SHP, SRU, LUT, CLU, HST and HSI} is discarded, and the fixed α value specified in these bits is always output from the {SHP, SRU, LUT, CLU, HST and HSI}. A value from 0 to 255 can be specified. These bits are valid for SHP, SRU, LUT, CLU, HST and HSI modules. For UDS, ILV, BRS and BRU modules, these bits are reserved. |
| 13 to 8 | FP [5:0] | All 0 | R/W | {mod} Internal Operation Timing Setting Specify 0. |
| 5 to 0 | RT [5:0] | All 0 | R/W | {mod}Target Node Value These bits specify the target node value for the {mod}. When using the {mod}, refer to Figure 33.31 to Figure 33.32 for settings. When not using the {mod}, specify 63. |

33.2.8.5 {mod} Sampling Point Register (VI6_DPR_{mod}_SMPPT: {mod} = HGO, HGT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----------|-----|-----|----|----|---------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | TGW[2:0] | | | — | — | PT[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------------|----------|---------------|-----|---|
| 31 to 11, 7, 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 to 8 | TGW[2:0] | B'000 | R/W | Target WPF Index for {mod} These bits are used for specifying the target WPF index of {mod}. If {mod} is not used, set 7 to these bits. |
| 5 to 0 | PT[5:0] | All 0 | R/W | Target Node Index for {mod} Histogram Sampling {mod} generates the histogram for the target node specified by these bits. For example, if these bits are set to 0, {mod} module generates the histogram for the output data of RPF0. Refer to below table for the target node index which can be used for {mod}. If {mod} is not used, set 63 to these bits. |

Table 33.27 Target node index for {mod} module

| Node Index | Module for Histogram Generation |
|------------|---------------------------------|
| 0 to 4 | RPF0 to 4 |
| 16 | SRU |
| 17 | UDS |
| 22 | LUT |
| 27 | BRU |
| 29 | CLU |
| 30 | HST |
| 31 | HSI |
| 46 | SHP |

33.2.9 SRU Control Registers

33.2.9.1 Super Resolution Control Register 0 (VI6_SRU_CTRL0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----------------|-----|-----|-----|-----------------|----------------|-----|-----|----------------|----------------|----------------|------------|-----|--|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | | |
| | — | — | — | — | — | — | — | SRU_PARAM0[8:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | |
| | — | — | — | SRU_PARAM1[4:0] | | | | — | SRU_MODE [2:0] | | | SRU_PAR AM2 | SRU_PAR AM3 | SRU_PAR AM4 | SRU _EN | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----------------------------|------------------|---------------|-----|--|
| 31 to 25, 15 to 13, 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 to 16 | SRU_PARAM0 [8:0] | All 0 | R/W | Super Resolution Parameter 0 Specify an appropriate value shown in Table 33.28; do not specify any other value. |
| 12 to 8 | SRU_PARAM1 [4:0] | All 0 | R/W | Super Resolution Parameter 1 Specify an appropriate value shown in Table 33.28; do not specify any other value. |
| 6 to 4 | SRU_MODE [2:0] | B'000 | R/W | Super Resolution Mode Setting These bits specify a super resolution mode. In super resolution without scaling, the output size is the same as the input size. In super resolution with double scale-up, the output size is twice the input size. B'000: Super resolution without scaling B'100: Super resolution with double scale-up Other settings are prohibited. |
| 3 | SRU_PARAM2 | B'0 | R/W | Super Resolution Parameter 2 This parameter setting depends on the color space of the image input to the SRU. Specify an appropriate value shown in Table 33.29; do not specify any other value. |
| 2 | SRU_PARAM3 | B'0 | R/W | Super Resolution Parameter 3 Specify an appropriate value shown in Table 33.29; do not specify any other value. |
| 1 | SRU_PARAM4 | B'0 | R/W | Super Resolution Parameter 4 This parameter setting depends on the color space of the image input to the SRU. Specify an appropriate value shown in Table 33.29; do not specify any other value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | SRU_EN | B'0 | R/W | Super Resolution Processing Enable/Disable Enables or disables super resolution processing. This setting has the highest priority over all other SRU registers. 0: Super resolution processing is disabled (input goes through the SRU without change) 1: Super resolution processing is enabled |

YCbCr format is recommended for the super resolution processing. If the super resolution processing is applied to an RGB-format image, adverse effects such as color blur may be generated; when using an RGB format, evaluate the image quality carefully before applying the processing to practical use.

To execute the super resolution processing in YCbCr format, set up the RPF and DPR appropriately so that YCbCr-format image data is input to the SRU. Specifically, make the RPFn output YCbCr-format image data and send the output data to the SRU through the DPR. For details of RPFn input/output settings, refer to section 33.2.6.3. For DPR settings, refer to section 33.2.8.1 to 33.2.8.4.

Table 33.28 Super Resolution Parameter Setting 1

| Intensity | | VI6_SRU_CTRL0 | | VI6_SRU_CTRL1 | VI6_SRU_CTRL2 | | | |
|-----------|---|---------------|------------|---------------|---------------|------------|------------|------------|
| | | SRU_MODE | SRU_PARAM0 | SRU_PARAM1 | SRU_PARAM5 | SRU_PARAM6 | SRU_PARAM7 | SRU_PARAM8 |
| Weak | 1 | 0 or 4 | 256 | 4 | H'7FF | 24 | 40 | 255 |
| | 2 | 0 or 4 | 256 | 4 | H'7FF | 8 | 16 | 255 |
| | 3 | 0 or 4 | 384 | 5 | H'7FF | 36 | 60 | 255 |
| | 4 | 0 or 4 | 384 | 5 | H'7FF | 12 | 27 | 255 |
| | 5 | 0 or 4 | 511 | 6 | H'7FF | 48 | 80 | 255 |
| Strong | 6 | 0 or 4 | 511 | 6 | H'7FF | 16 | 36 | 255 |

Table 33.29 Super Resolution Parameter Setting 2

| Color Space of SRU Input Image | SRU_PARAM2 | SRU_PARAM3 | SRU_PARAM4 |
|--------------------------------|------------|------------|------------|
| YCbCr* | 0 | 1 | 0 |
| RGB | 1 | 1 | 1 |

Note: * YCbCr format is recommended for the image input to the SRU.

33.2.9.2 Super Resolution Control Register 1 (VI6_SRU_CTRL1)

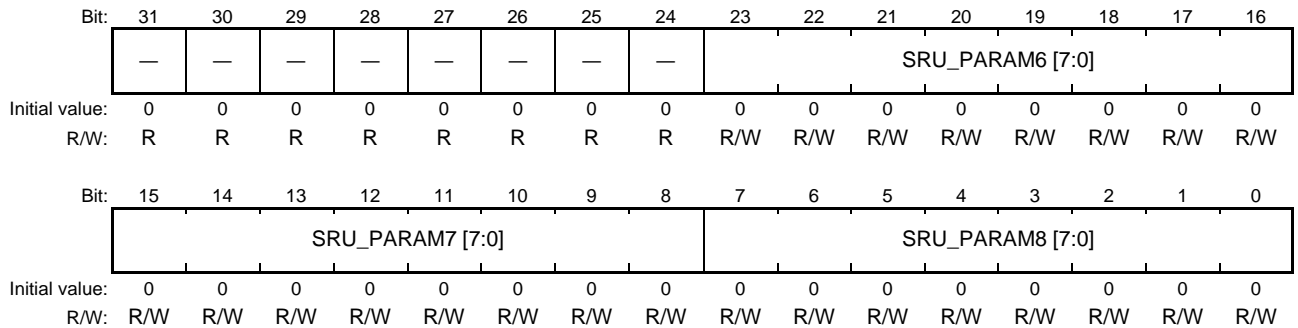
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|-------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | SRU_PARAM5 [10:0] | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------|---------------|-----|--|
| 31 to 11 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 to 0 | SRU_PARAM5 [10:0] | All 0 | R/W | Super Resolution Parameter 5 Specify an appropriate value shown in Table 33.28; do not specify any other value. |

33.2.9.3 Super Resolution Control Register 2 (VI6_SRU_CTRL2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | SRU_PARAM6 [7:0] | All 0 | R/W | Super Resolution Parameter 6 Specify an appropriate value shown in Table 33.28; do not specify any other value. |
| 15 to 8 | SRU_PARAM7 [7:0] | All 0 | R/W | Super Resolution Parameter 7 Specify an appropriate value shown in Table 33.28; do not specify any other value. |
| 7 to 0 | SRU_PARAM8 [7:0] | All 0 | R/W | Super Resolution Parameter 8 Specify an appropriate value shown in Table 33.28; do not specify any other value. |

33.2.10 UDS Control Registers

33.2.10.1 Scaling Control Registers (VI6_UDS_CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|-----|-----|-----------|----|----|-----|-----------|----|----|----|-----|------|------------|--------------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | AMD | FMD | BLA DV | — | — | AON | ATH ON | — | — | — | BC | NE_A | NE_RC R | NE_G Y | NE_ BCB |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R | R | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | AMD SLH | EXT (TDIP C) | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---|----------|---------------|-----|--|
| 31, 27, 26, 23 to 21, 15 to 3, 0 | — | All 0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 30 | AMD | B'0 | R/W | Pixel Count at Scale-Up Specifies the number of pixels generated through scale-up in the UDS. This bit setting is ignored for scale-down. 0: Pixel count after scale-up is $1 + \langle n - 1 \rangle \times \text{scale-up factor}$ 1: Pixel count after scale-up is $\langle n \rangle \times \text{scale-up factor}$ *n: Number of pixels input to UDS |
| 29 | FMD | B'0 | R/W | Padding for Insufficient Clipping Size When the scaling filter outputs an image that is smaller than the clipping size (VI6_UDS_CLIP_SIZE), pixels are interpolated to match the clipping size. This bit specifies the pixel filling method. 0: Pixels are filled by copying pixels at the right edge and the bottom edge 1: Pixels are filled with the color specified by VI6_UDS_FILL_COLOR |
| 28 | BLADV | B'0 | R/W | Bilinear or Nearest Neighbor Interpolation Characteristic Control Controls the characteristics of bilinear or nearest neighbor interpolation. Setting this bit to 1 improves the aliasing characteristics at $\times 1/2$ to $1/8$ scaling (VI6_UDS_SCALE.VMANT/HMANT = 2 to 8). Note that the apparent resolution around edges may deteriorate. In multi-tap mode, this control is not available; to use this control, set the BC bit to 0. |
| 25 | AON | B'0 | R/W | Scale-Up/Down of α Plane These bits specify whether to enable or disable scale-up/down of the α plane when scaling up/down in the RGB format. When the AON bit is set to 0, the UDS outputs the value of the VI6_UDS_ALPVAL.ALPH_VAL0 bits as a fixed α value. 0: α scale-up/down is not performed 1: α scale-up/down is performed |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 24 | ATHON | B'0 | R/W | <p>α Output Data Threshold Comparison Enable/Disable</p> <p>Enables or disables comparison with the α output data threshold. When this bit is 1, the output α value is replaced according to the VI6_UDS_ALPTH and VI6_UDS_ALPVAL values.</p> <p>When the AON bit is 0 (scale-up/-down of the α plane is disabled), this bit setting has no effect.</p> <p>0: α output data threshold comparison is disabled 1: α output data threshold comparison is enabled</p> |
| 20 | BC | B'0 | R/W | <p>Pixel Component Interpolation Method at Scale-Up/Down</p> <p>Specifies the method for interpolating pixel components at scale-up/-down.</p> <p>0: Bilinear or nearest neighbor interpolation method is used 1: Interpolation method equivalent to 4 to 17 taps in accordance with the scaling factor is used (multi-tap mode)</p> <p>Setting 1 to BC bit (Multi tap mode) is recommended.</p> <p>Note that the α component is interpolated by the method specified in the NE_A bit instead of the multi-tap method.</p> <p>When the BC and AON bits are both set to 1, the scaling shall be scale-up, no scaling, or scale-down with scaling factor 1/1 to 1/2.</p> |
| 19 | NE_A | B'0 | R/W | <p>α Interpolation Method</p> <p>Specifies the interpolation method of the α plane. When the AON bit is 0 (scale-up/-down of the α plane is disabled), this bit setting has no effect.</p> <p>0: Bilinear method 1: Nearest neighbor method*</p> |
| 18 | NE_RCR | B'0 | R/W | <p>R/Cr Interpolation Method When Bilinear/Nearest Neighbor Interpolation is Selected</p> <p>Specifies the interpolation method of the R/Cr component when bilinear/nearest neighbor interpolation is selected (BC = 0).</p> <p>0: Bilinear method 1: Nearest neighbor method*</p> |
| 17 | NE_GY | B'0 | R/W | <p>G/Y Interpolation Method When Bilinear/Nearest Neighbor Interpolation is Selected</p> <p>Specifies the interpolation method of the G/Y component when bilinear/nearest neighbor interpolation is selected (BC = 0).</p> <p>0: Bilinear method 1: Nearest neighbor method*</p> |
| 16 | NE_BCB | B'0 | R/W | <p>B/Cb Interpolation Method When Bilinear/Nearest Neighbor Interpolation is Selected</p> <p>Specifies the interpolation method of the B/Cb component when bilinear/nearest neighbor interpolation is selected (BC = 0).</p> <p>0: Bilinear method 1: Nearest neighbor method*</p> |
| 2 | AMDSLH | B'0 | R/W | <p>Horizontal Filter Phase</p> <p>0: The horizontal start and end phase is automatically determined by the setting of AMD. In this case, the value of VI6_UDS_HPHASE.HSTP/HEDP is not valid.</p> <p>1: The horizontal start and end phase is determined by the value of VI6_UDS_HPHASE.HSTP/HEDP.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------------|---------------|-----|---|
| 1 | EXT (TDIPC) | B'0 | R/W | Extended UDS function Enable/Disable Select 0: Extended UDS function Disable 1: 2D IPC function or low pass filter Enable UDS execute 2D IPC function based on this bit and the setting of VI6_UDS_IPC register. UDS execute low pass filter based on this bit and the setting of VI6_UDS_IPC register. See section 33.2.10.7, 2D IPC Setting Register (VI6_UDS_IPC) for more details. |

Note: * This method can be used only when the scale-up/-down factor is 1/1 to 1/4.

33.2.10.2 Scaling Factor Registers (VI6_UDS_SCALE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----|-----|-----|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | HMANT[3:0] | | | | HFRAC[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | VMANT[3:0] | | | | VFRAC[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 28 | HMANT [3:0] | All 0 | R/W | <p>Multiplier (Integral Part) of Horizontal Scaling Factor</p> <p>These bits specify the integral part of the horizontal scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size.</p> <p>A value from H'0 to H'F can be specified. Select a value within the range shown in Table 33.30.</p> |
| 27 to 16 | HFRAC [11:0] | All 0 | R/W | <p>Multiplier (Fractional Part) of Horizontal Scaling Factor</p> <p>These bits specify the fractional part of the horizontal scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size.</p> <p>A value from H'100 to H'FFF can be specified when an image is up-scaled (the HMANT value is 0) in the horizontal direction. A value from H'000 to H'FFF can be specified when an image is downscaled (the HMANT value is not 0) in the horizontal direction. Select a value within the range shown in Table 33.30.</p> |
| 15 to 12 | VMANT [3:0] | H'0 | R/W | <p>Multiplier (Integral Part) of Vertical Scaling Factor</p> <p>These bits specify the integral part of the vertical scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size.</p> <p>A value from H'0 to H'F can be specified. Select a value within the range shown in Table 33.30.</p> |
| 11 to 0 | VFRAC [11:0] | All 0 | R/W | <p>Multiplier (Fractional Part) of Vertical Scaling Factor</p> <p>These bits specify the fractional part of the vertical scaling factor. The image size to be obtained changes according to this setting. Calculate an appropriate value using the formula shown later to obtain a desired image size.</p> <p>A value from H'100 to H'FFF can be specified when an image is up-scaled (the VMANT value is 0) in the vertical direction. A value from H'000 to H'FFF can be specified when an image is downscaled (the VMANT value is not 0) in the vertical direction. Select a value within the range shown in Table 33.30.</p> |

The HMANT and HFRAC bits set the scale-up/-down factor for an image in the horizontal direction, and the VMANT and VFRAC bits set the scale-up/-down factor for an image in the vertical direction. The UDS operation switches between horizontal scale-up and horizontal scale-down according to the HMANT and HFRAC bit settings, as shown in Table 33.30. (Setting a value outside the ranges of Table 33.30 in UDS operation is prohibited). Table 33.30 is for the horizontal direction, but it is similar for the vertical direction. In this case, replace HMANT with VMANT and HFRAC with VFRAC when reading.

Note that settings for scaling in the horizontal direction and settings for scaling in the vertical direction can be made independently. Therefore, a setting for scale-up in the horizontal direction and scale-down in the vertical direction is possible. In such a case, because the formula (described later) for obtaining the image size after scale-up/-down is different between scale-up and scale-down, a formula matching the scale-up or scale-down operation should be selected independently for the horizontal direction and vertical direction.

Table 33.30 Switching of Horizontal Scale-Up/Down Operation According to HMANT and HFRAC Bit Settings

| HMANT | HFRAC | UDS Operation |
|------------|----------------|-------------------------------|
| H'0 | H'100 to H'FFF | Scale-up |
| H'1 | H'000 | Same size (no scale-up/-down) |
| | H'001 to H'FFF | Scale-down |
| H'2 to H'F | H'000 to H'FFF | |

Described here is the method for calculating the size of the up-scaled/downscaled image that was obtained based on this register setting. First, define the variables necessary for calculating the horizontal size of the up-scaled/downscaled image, as shown below.

$$hscale = \frac{4096}{4096 \times m_h + f_h}$$

m_h is the value of VI6_UDS_SCALE.HMANT, and f_h is the value of VI6_UDS_SCALE.HFRAC. This formula expresses the estimate of the scale-up/-down factor processed by the UDS. If the horizontal size of the image before scale-up/-down is set as $hsize_{org}$, the horizontal size of the image after scale-up/-down can be roughly obtained through $hsize_{org} \times hscale$.

Similarly, define the variables necessary for calculating the vertical size of the up-scaled/down-scaled image.

$$vscale = \frac{4096}{4096 \times m_v + f_v}$$

When setting m_v as the value of VI6_UDS_SCALE.VMANT, f_v as the value of VI6_UDS_SCALE.VFRAC, and $vsize_{org}$ as the vertical size of the image before scale-up/-down, the vertical size of the image after scale-up/-down can be roughly obtained through $vsize_{org} \times vscale$.

When the UDS performs scale-down with the settings of Table 33.30, using the variables defined so far, the horizontal size of the downscaled image $hsize_{down_scaled}$ and the vertical size of the downscaled image $vsize_{down_scaled}$ become as follows:

$$hsize_{down_scaled} = \left\langle 1 + \left(\left(\left(1 + \frac{\langle hsize_{org} - 1 \rangle}{m_h'} \right) - 1 \right) \times m_h' \times 4096 + hstp + hedp \right) / (4096 \times m_h + f_h) \right\rangle$$

$$vsize_{down_scaled} = \left\langle 1 + \left(\left(\left\langle 1 + \frac{\langle vsize_{org} - 1 \rangle}{m_v'} \right\rangle - 1 \right) \times m_v' \times 4096 \right) / (4096 \times m_v + f_v) \right\rangle$$

$hstp$ is the value of VI6_UDS_HPHASE.HSTP, $hedp$ is the value of VI6_UDS_HPHASE.HEDP.

The value of m_h' or m_v' , which is shown in Table 33.31, changes according to the setting of VI6_UDS_SCALE.HMANT or VI6_UDS_SCALE.VMANT and VI6_UDS_CTRL.BLADV.

Table 33.31 m_h' or m_v' Setting

| VI6_UDS_SCALE.HMANT Setting (VI6_UDS_SCALE.VMANT Setting) | VI6_UDS_CTRL.BLADV | m_h' (m_v') |
|--|--------------------|----------------------|
| 0 to 3 | 0 | 1 |
| 4 to 7 | 0 | 2 |
| 8 to 15 | 0 | 4 |
| 2 to 3 | 1 | 2 |
| 4 to 7 | 1 | 4 |
| 0, 1, 8 to 15 | 1 | Forbidden |

When the UDS performs scale-up with the settings of Table 33.30 and VI6_UDS_CTRL.AMD is 0, the horizontal size of the up-scaled image $hsize_{up_scaled}$ and the vertical size of the up-scaled image $vsize_{up_scaled}$ become as follows:

$$hsize_{up_scaled} = \left\langle 1 + ((hsize_{org} - 1) \times 4096 + hstp + hedp) / (4096 \times m_h + f_h) \right\rangle$$

$$vsize_{up_scaled} = \left\langle 1 + ((vsize_{org} - 1) \times 4096) / (4096 \times m_v + f_v) \right\rangle$$

When VI6_UDS_CTRL.AMD is 1, the horizontal size of the up-scaled image $hsize_{up_scaled}$ and the vertical size of the up-scaled image $vsize_{up_scaled}$ become as follows:

$$hsize_{up_scaled} = \langle hsize_{org} \times hscale \rangle$$

$$vsize_{up_scaled} = \langle vsize_{org} \times vscale \rangle$$

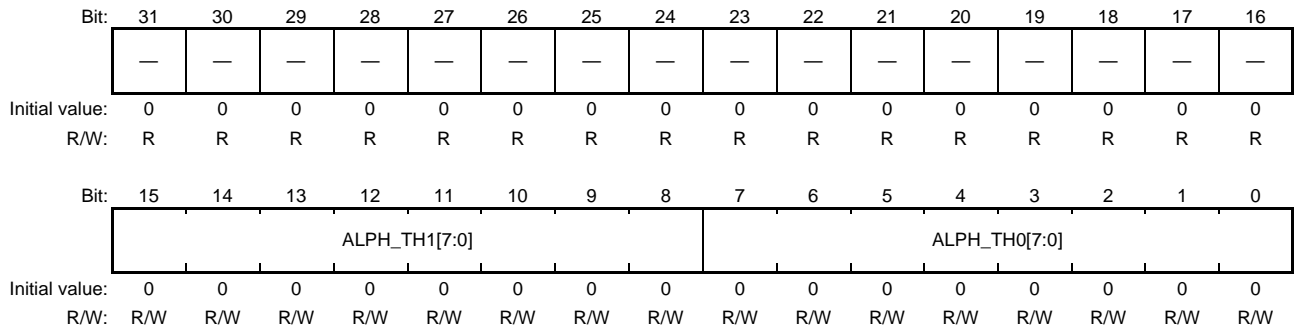
After considering the division of $hscale$ and $vscale$, the formulas become as follows:

$$hsize_{up_scaled} = \langle (hsize_{org} \times 4096) / (4096 \times m_h + f_h) \rangle$$

$$vsize_{up_scaled} = \langle (vsize_{org} \times 4096) / (4096 \times m_v + f_v) \rangle$$

33.2.10.3 α Data Threshold Setting Registers (VI6_UDS_ALPTH)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 8 | ALPH_TH1 [7:0] | All 0 | R/W | α Data Threshold Setting 1 When the α value is equal to or greater than the value of the ALPH_TH1 bits, the α value is replaced with that of VI6_UDS_ALPVAL.ALPH_VAL2. When VI6_UDS_CTRL.AON is 0 (scale-up/-down of the α plane is disabled), the setting of these bits has no effect. A value from H'00 to H'FF can be specified. |
| 7 to 0 | ALPH_TH0 [7:0] | All 0 | R/W | α Data Threshold Setting 0 When the α value is equal to or smaller than the value of the ALPH_TH0 bits, the α value is replaced with that of VI6_UDS_ALPVAL.ALPH_VAL0. When VI6_UDS_CTRL.AON is 0 (scale-up/-down of the α plane is disabled), the setting of these bits has no effect. A value from H'00 to H'FF can be specified. |

33.2.10.4 α Data Replacing Value Setting Registers (VI6_UDS_ALPVAL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----------------|-----|-----|-----|-----|-----|-----|-----|----------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | ALPH_VAL2[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | ALPH_VAL1[7:0] | | | | | | | | ALPH_VAL0[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|---|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | ALPH_VAL2 [7:0] | All 0 | R/W | Replacing α Value Setting 2 These bits set a value that replaces the α value when it is equal to or greater than the value of VI6_UDS_ALPTH.ALPH_TH1. When VI6_UDS_CTRL.AON is 0 (scale-up/-down of the α plane is disabled), the setting of these bits has no effect. A value from H'00 to H'FF can be specified. |
| 15 to 8 | ALPH_VAL1 [7:0] | All 0 | R/W | Replacing α Value Setting 1 These bits set a value that replaces the α value when it is greater than the value of VI6_UDS_ALPTH.ALPH_TH0 and also smaller than that of VI6_UDS_ALPTH.ALPH_TH1. When VI6_UDS_CTRL.AON is 0 (scale-up/-down of the α plane is disabled), the setting of these bits has no effect. A value from H'00 to H'FF can be specified. |
| 7 to 0 | ALPH_VAL0 [7:0] | All 0 | R/W | Replacing α Value Setting 0 These bits set a value that replaces the α value when it is equal to or smaller than the value of VI6_UDS_ALPTH.ALPH_TH0. When VI6_UDS_CTRL.AON is 0 (scale-up/-down of the α plane is disabled), α output value for the UDS is the fixed value specified in these bits. A value from H'00 to H'FF can be specified. |

33.2.10.5 Passband Registers (VI6_UDS_PASS_BWIDTH)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|---------------|-----|-----|-----|-----|-----|-----|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | BWIDTH_H[6:0] | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | — | BWIDTH_V[6:0] | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-------------------|----------------|---------------|-----|---|
| 31 to 23, 15 to 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 22 to 16 | BWIDTH_H [6:0] | All 0 | R/W | Horizontal Signal Passband at Image Scale-Up/Down Set these bits following the method described later. |
| 6 to 0 | BWIDTH_V [6:0] | All 0 | R/W | Vertical Signal Passband at Image Scale-Up/Down Set these bits following the method described later. |

The method for setting the passband in the horizontal direction of an image is described. When the VI6_UDS_SCALE.HMANT bits for horizontal scale-up/-down factor setting are not 0, set the BWIDTH_H bits according to the following formula. When the VI6_UDS_SCALE.HMANT bits are 0, set 64 in the BWIDTH_H bits.

$$BWIDTH_H = \left\langle 64 \times \frac{4096 \times m_h'}{4096 \times m_h + f_h} \right\rangle \quad (\text{VI6_UDS_SCALE.HMANT} \neq 0)$$

$$BWIDTH_H = 64 \quad (\text{VI6_UDS_SCALE.HMANT} = 0)$$

m_h is the value of VI6_UDS_SCALE.HMANT, and f_h is the value of VI6_UDS_SCALE.HFRAC. For the m_h' value, see Table 33.31. The method for setting the passband in the vertical direction of an image is similar to that for the horizontal direction described earlier. Since only the correspondence relationship of the registers is changed as shown below, replace the variables in the previous explanation as shown below when reading.

BWIDTH_H → BWIDTH_V

$m_h' \rightarrow m_v'$

$m_h \rightarrow m_v$

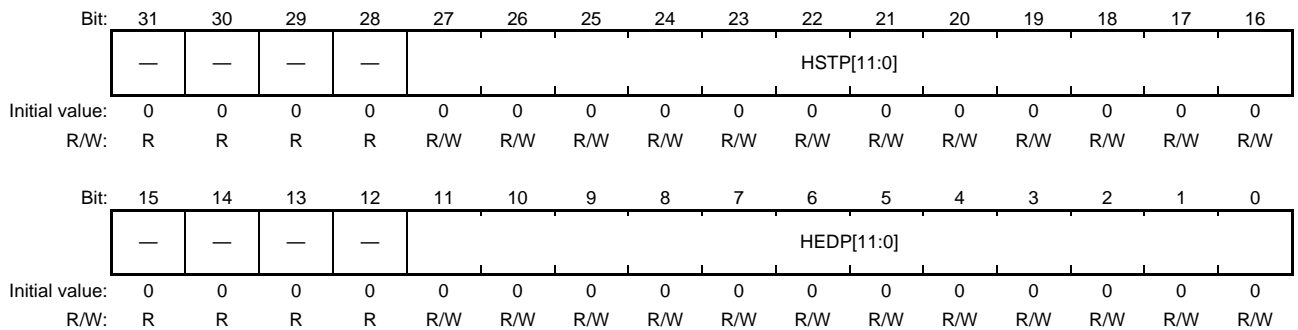
$f_h \rightarrow f_v$

VI6_UDS_SCALE.HMANT → VI6_UDS_SCALE.VMANT

VI6_UDS_SCALE.HFRAC → VI6_UDS_SCALE.VFRAC

33.2.10.6 Scaling Filter Horizontal Phase Registers (VI6_UDS_HPHASE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|-------------|---------------|-----|--|
| 31 to 28, 15 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 to 16 | HSTP [11:0] | All 0 | R/W | Horizontal Starting Phase |
| 11 to 0 | HEDP [11:0] | All 0 | R/W | Horizontal Ending Phase |

This register can be used for the image division mode. When this register is not used, set VI6_UDS_CTRL.AMDSLH to 0, and set HSTP and HEDP to 0. When the value of HSTP or HEDP is not equal to 0, the size is different from the value shown in section 33.2.10.2.

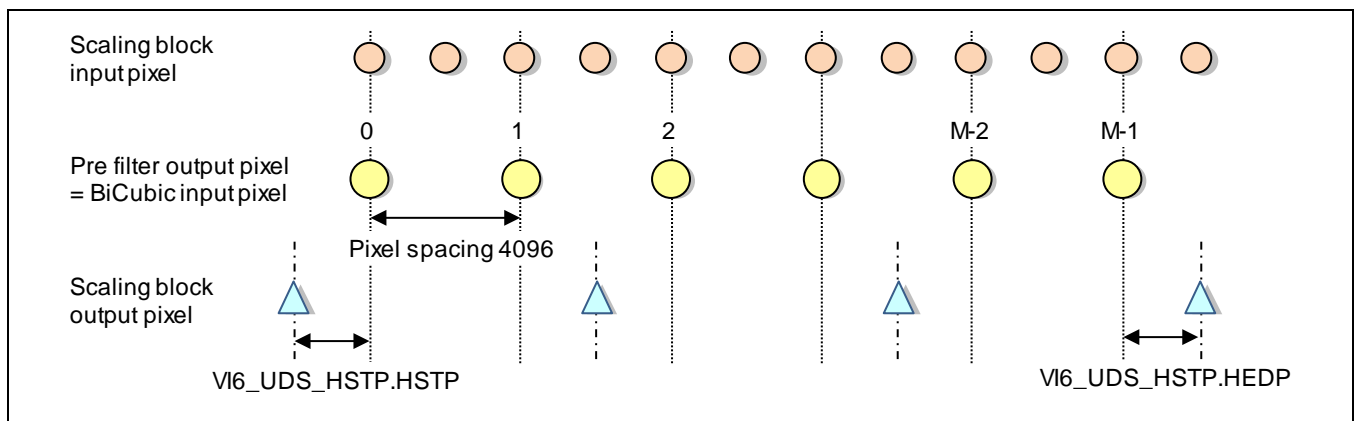


Figure 33.34 Horizontal Starting and Ending Phase

33.2.10.7 2D IPC Setting Register (VI6_UDS_IPC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-------|----|----|----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | FIELD | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | LPF E |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-------------------|----------|---------------|-----|--|
| 31 to 28, 26 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 | FIELD | B'0 | R/W | Top/Bottom Field Select Select Top/Bottom Field of image data when 2D IPC function is enabled (VI6_UDSn_CTRL.EXT = 1). 0: Top Field 1: Bottom Field |
| 0 | LPFE | B'0 | R/W | Low Pass Filter Enable 0: low pass filter disable 1: low pass filter enable |

IPC setting is shown below.

- Set 1 to VI6_UDSn_CTRL.EXT.
- Set UDS register as double up-scale in vertical direction and no scaling in horizontal direction.
- Set appropriate setting of processing image.

Top Field: H'0000_0000

Bottom Field: H'0800_0000

- Set 0 to VI6_UDS_CTRL.FMD
- Set 1 to VI6_UDS_CTRL.BC

Low pass filter setting is shown below

- Set 1 to VI6_UDS0_CTRL.EXT and set 1 to VI6_UDS0_CTRL.BC
- Set equal setting (x1) to VI6_UDS_SCALE
- Set 0 - 64 to VI6_UDS_PASS_BWIDTH. BWIDTH_H for low pass filter in horizontal direction.
- Set 0 - 64 to VI6_UDS_PASS_BWIDTH. BWIDTH_V for low pass filter in vertical direction.

Strength of low pass filter can be set by VI6_UDS_PASS_BWIDTH.

When UDS is used for IPC processing, low pass filter cannot be enabled.

| | | | |
|---------------------|--------|-------|------|
| Intensity of strong | Strong | | Weak |
| VI6_UDS_PASS_BWIDTH | 0 | | 64 |

Figure 33.35 Strength of Low Pass Filter and setting of pass band width

33.2.10.8 UDS Horizontal Input Clipping Registers (VI6_UDS_HSZCLIP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----------------|-----|-----|-----|-----|---------------|-----|-----|-----|-----|-----|-----|-----|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | HCE N | — | — | — | — | HCL_OFST[7:0] | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R/W | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | HCL_SIZE[12:0] | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|------------------------------------|--------------------|---------------|-----|---|
| 31 to 29, 27 to 24, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | HCEN | B'0 | R/W | Horizontal Input Size Clipping Enable This bit is used for setting the horizontal size clipping of UDS. 0: Disable Horizontal Size Clipping 1: Enable Horizontal Size Clipping |
| 23 to 16 | HCL_OFST [7:0] | All 0 | R/W | Horizontal Offset Value for Input Size Clipping |
| 12 to 0 | HCL_SIZE [12:0] | All 0 | R/W | Horizontal Clipping Size |

This register can be used for the image division mode. If the image partition mode is used, contact to RENESAS for detail.

33.2.10.9 UDS Output Size Clipping Registers (VI6_UDS_CLIP_SIZE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | CL_HSIZE[12:0] | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | CL_VSIZE[12:0] | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|-----------------|---------------|-----|---|
| 31 to 29, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | CL_HSIZE [12:0] | All 0 | R/W | Clipping Size of Horizontal Pixel Count after Scale-Up/-Down The horizontal width of an image output from the scaling filter is adjusted (clipped or padded) to match the pixel count set in the CL_HSIZE bits. The setting range is 4 to 2048 in a scale-down operation (see Table 33.30) and 4 to 8190 in a scale-up operation. These bits always have to be set when using the UDS, regardless of the scale-up, scale-down, or no-scaling setting by the VI6_UDS_SCALE register. |
| 12 to 0 | CL_VSIZE [12:0] | All 0 | R/W | Clipping Size of Vertical Pixel Count after Scale-Up/-Down The vertical width of an image output from the scaling filter is adjusted (clipped or padded) to match the pixel count set in the CL_VSIZE bits. The setting range is 4 to 2048 in a scale-down operation (see Table 33.30) and 4 to 8190 in a scale-up operation. These bits always have to be set when using the UDS, regardless of the scale-up, scale-down, or no-scaling setting by the VI6_UDS_SCALE register. |

Figure 33.36 shows the configuration of the UDS. The UDS consists of a scaling filter and clipping circuit, such as the configuration shown in Figure 33.36. The scaling filter and clipping circuit are independent of each other.

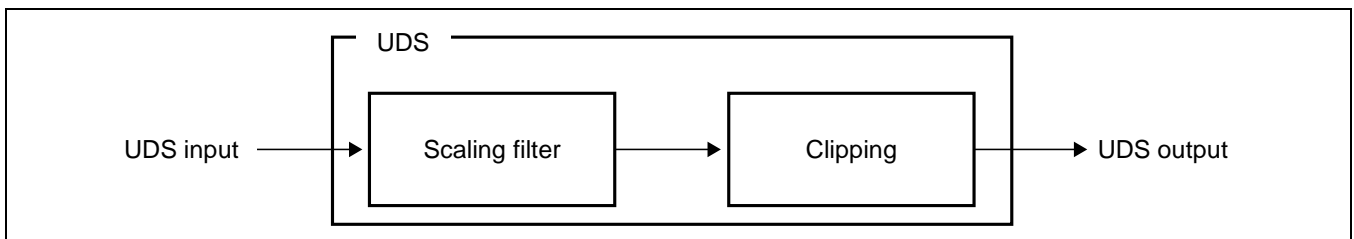


Figure 33.36 UDS Configuration

The size of the image actually output by the scaling filter (refer to section 33.2.10.2 for calculation) is determined from the size of the image input to the scaling filter and the VI6_UDS_SCALE setting; that is, three types of image are output by the UDS according to the relationship between the size of the image actually output from the scaling filter ($hsize_{scaled}$ and $vsize_{scaled}$) and this register setting as shown in Figure 33.37.

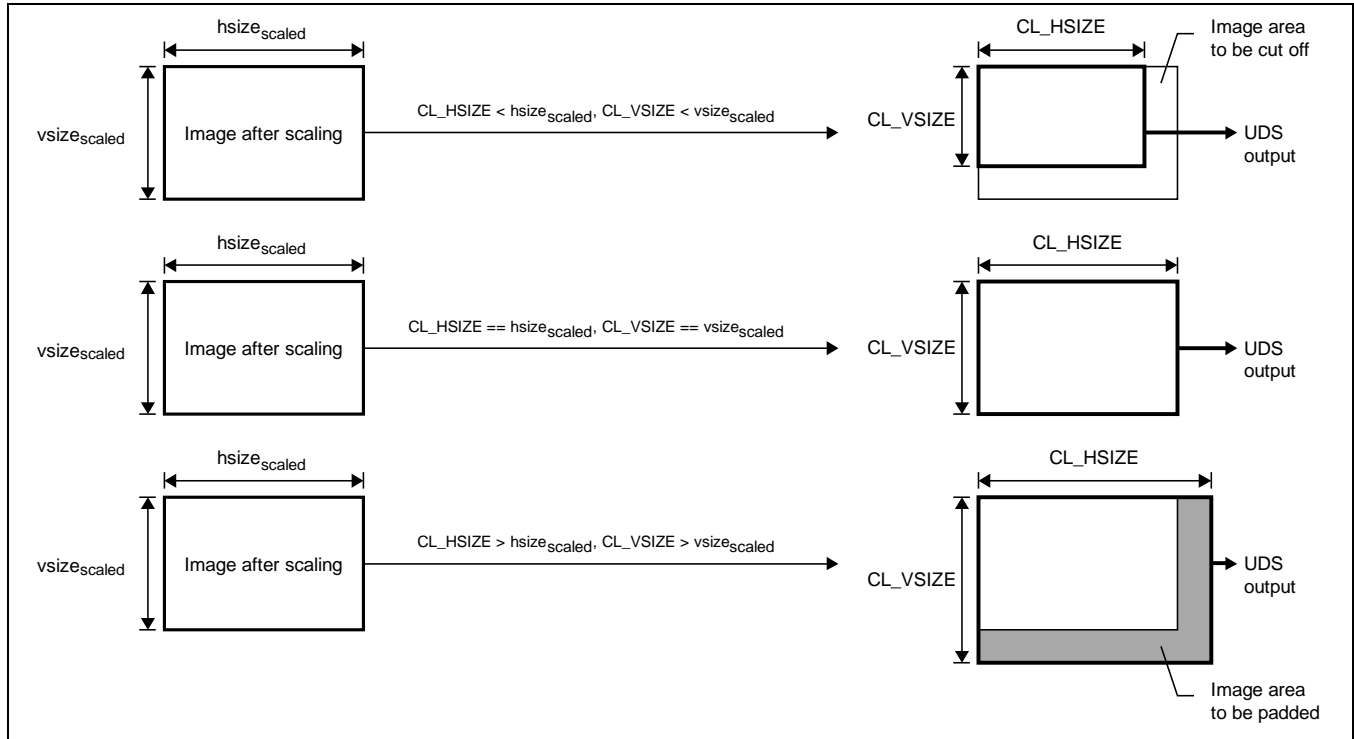


Figure 33.37 UDS Output Image for Each CL_HSIZE/VSIZE Setting

When the settings of the CL_HSIZE and CL_VSIZE bits are smaller than the horizontal and vertical pixel counts ($hsize_{scaled}$ and $vsize_{scaled}$) actually output by the scaling filter, the up-scaled/down-scaled image is clipped to become the UDS output image.

When the settings of the CL_HSIZE and CL_VSIZE bits are equal to the horizontal and vertical pixel counts ($hsize_{scaled}$ and $vsize_{scaled}$) actually output by the scaling filter, the image actually output by the scaling filter becomes the UDS output image without change.

When the settings of the CL_HSIZE and CL_VSIZE bits are greater than the horizontal and vertical pixel counts ($hsize_{scaled}$ and $vsize_{scaled}$) actually output by the scaling filter, in order to obtain the UDS output image, the image is padded in the mode set by VI6_UDS_CTRL.FMD for the area in which the settings of the CL_HSIZE and CL_VSIZE bits have exceeded the number of pixels actually output by the scaling filter.

33.2.10.10 Color Fill Register (VI6_UDS_FILL_COLOR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----|-----|-----|-----|-----|-----|-----|------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | RFILC[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | GFILC[7:0] | | | | | | | | BFILC[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | RFILC [7:0] | All 0 | R/W | R/Cr Component of Fill Color A value from H'00 to H'FF can be specified. |
| 15 to 8 | GFILC [7:0] | All 0 | R/W | G/Y Component of Fill Color A value from H'00 to H'FF can be specified. |
| 7 to 0 | BFILC [7:0] | All 0 | R/W | B/Cb Component of Fill Color A value from H'00 to H'FF can be specified. |

The scaling filter creates an up-scaled/down-scaled image based on the scaling factor settings of the VI6_UDS_SCALE register. If the size of the up-scaled/down-scaled image created by the scaling filter smaller than the clipping size set in the VI6_UDS_CLIP_SIZE register (lowest case in Figure 33.37) while VI6_UDS_CTRL.FMD is set to 1, the color specified by this register is used to fill the space until the clipping size is reached.

The α value of the area to be padded by this register is dependent on VI6_UDS_CTRL.FMD setting. When VI6_UDS_CTRL.FMD is 0 (repetition), the pixel value at the right edge (bottom edge) of the image is repeated as the α value. When VI6_UDS_CTRL.FMD is 1 (color information of this register is used), the α value is 0.

33.2.11 LUT Control Register

33.2.11.1 LUT Control Register (VI6_LUT_CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | LUT_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | LUT_EN | B'0 | R/W | 1D-LUT Enable/Disable Enables or disables the 1D-LUT function by the LUT. When the 1D-LUT is used, the color component curve information needs to be set separately in the LUT table. For the LUT table settings, refer to section 33.3.5.1. 0: 1D-LUT function is disabled 1: 1D-LUT function is enabled |

In the LUT, various image processing, such as curves with high operation load (e.g., γ correction), negative-positive conversion, and gain adjustment of images, can be achieved by the data replacement processing by the 1D-LUT. As shown in Figure 33.38, the LUT replaces each component of the input pixel data using the set replacement table of 256 entries. For example, if the LUT is set as in Figure 33.39, when there is an input of 150, the data stored in address 150 of the 1D-LUT is read and output as the LUT output. Figure 33.39 shows a case in which the input and output become equal for convenience in explaining.

In the LUT settings shown in Figure 33.40, the input bits are reversed. This has the effect of negative-positive flipping. In Figure 33.41, γ correction ($\gamma = 1.8$ is shown as an example) is possible. As described above, information to be set in the LUT indicates LUT processing characteristics. If the same value is set for each component in the LUT, an equal effect can be obtained for each component of the input image when it is processed. If the LUT is set with different characteristics for each component, the processing characteristics can be changed for each component.

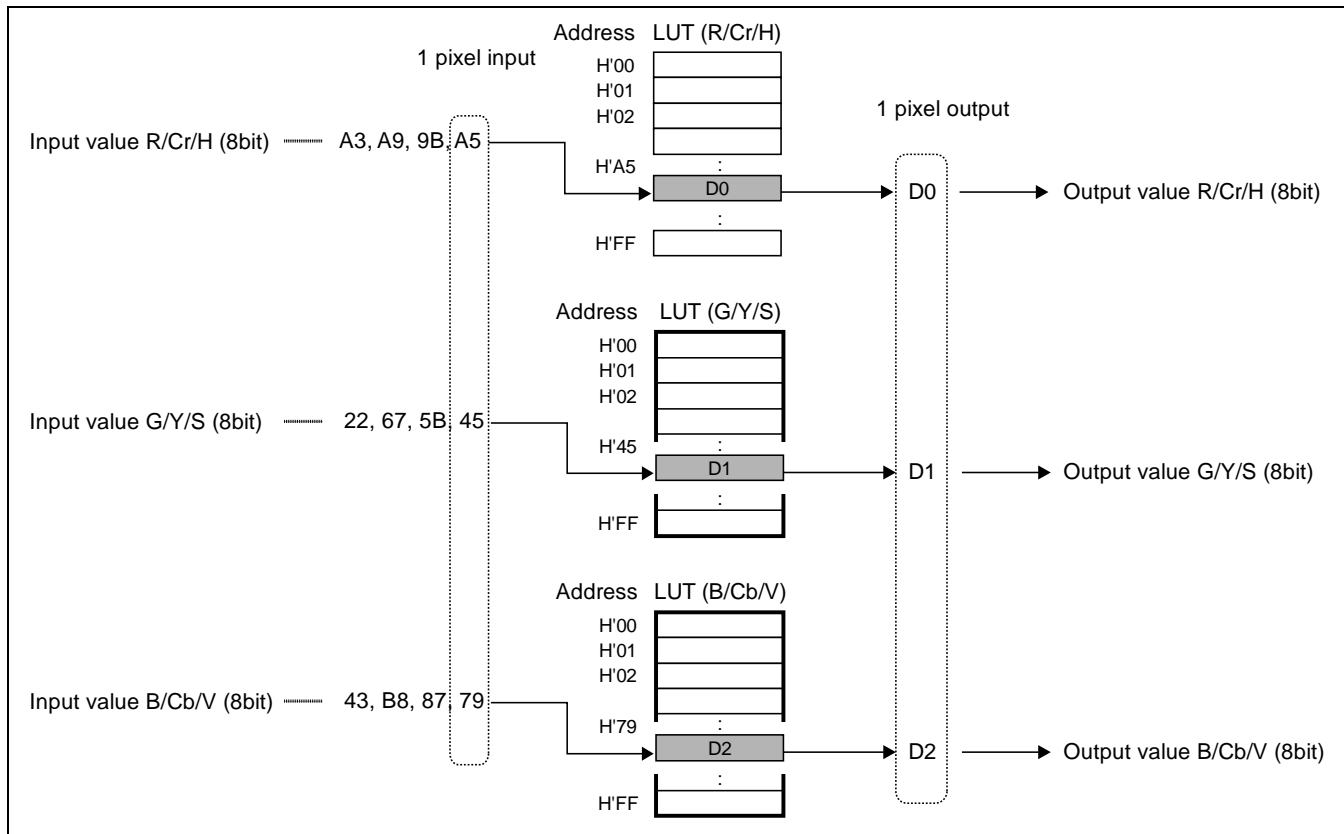


Figure 33.38 Relationship between Input and Output for 1D-LUT Table

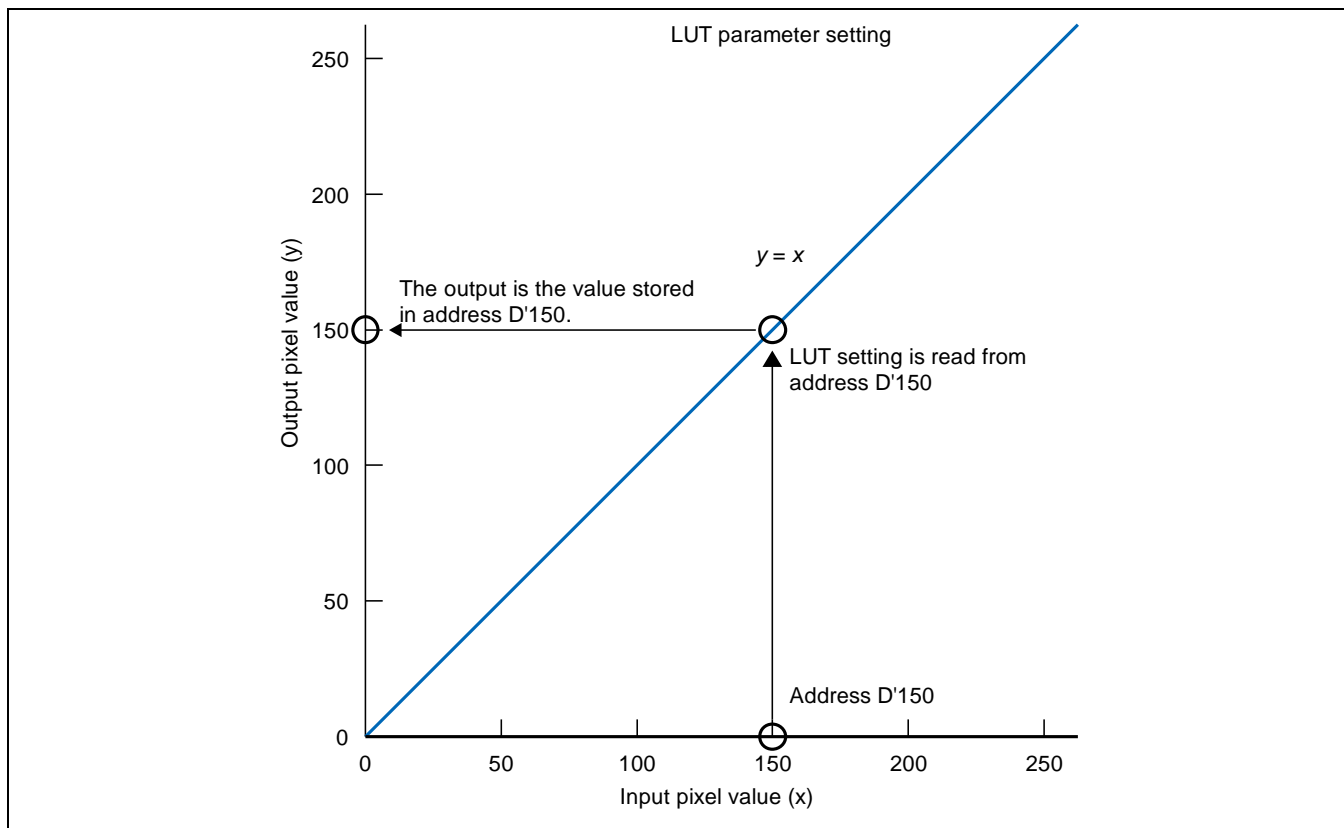


Figure 33.39 Setting Example in Which Output Becomes Equal to Input

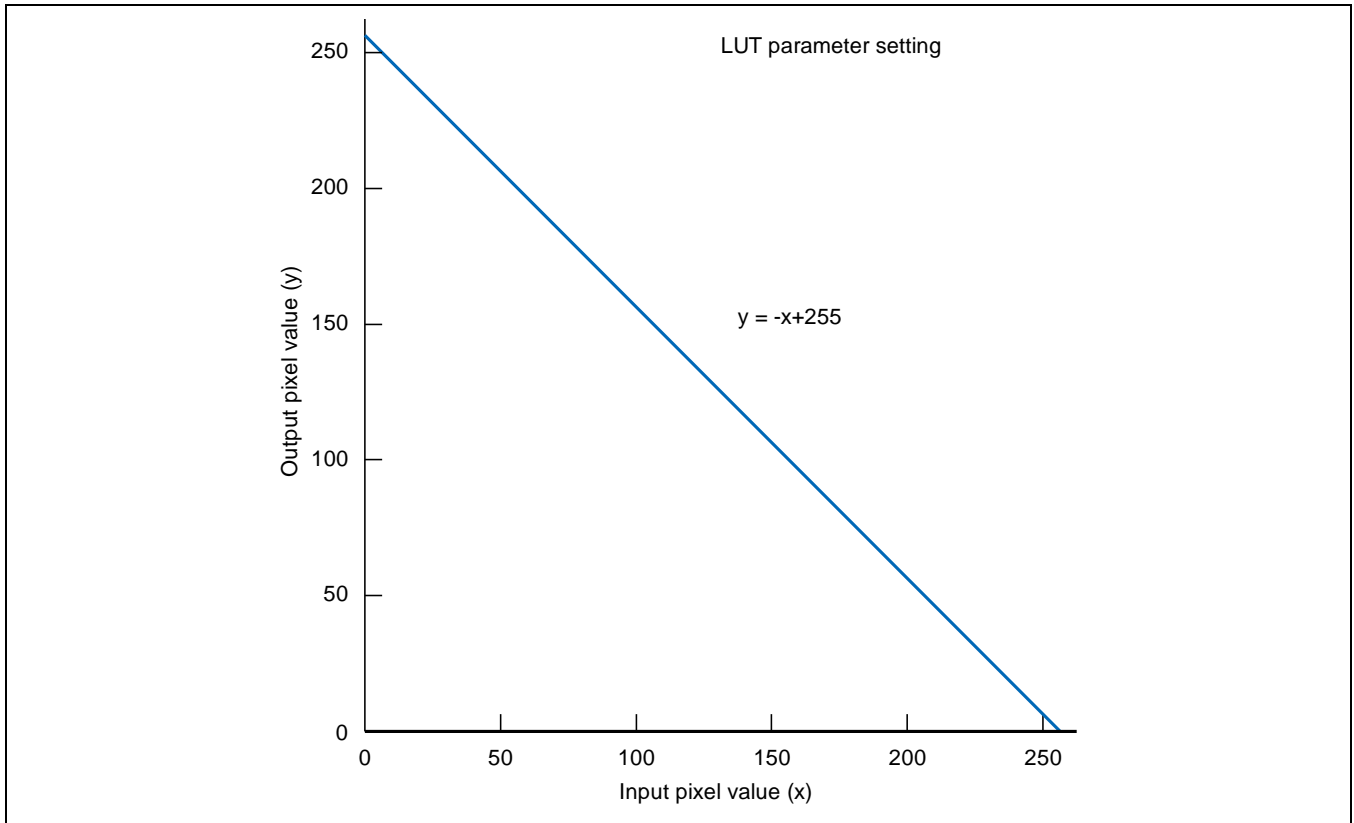


Figure 33.40 Setting Example of Negative-Positive Conversion

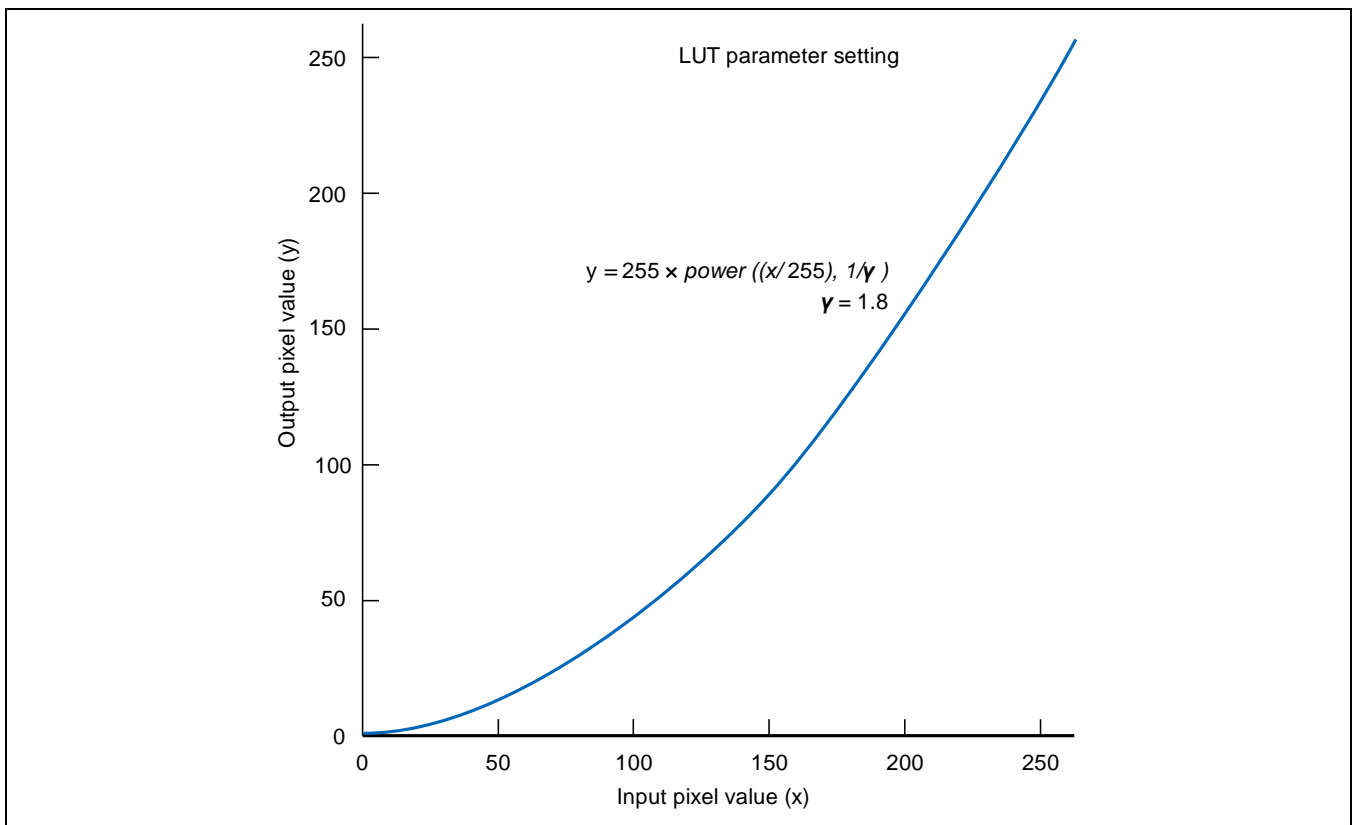


Figure 33.41 Setting Example of γ Correction

33.2.12 CLU Control Register

33.2.12.1 CLU Control Register (VI6_CLU_CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----------|-----|----------|-----|----|----|----------|-----|----------|-----|----------|-----|----|----|-----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | AAI | — | — | — | MVS | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | AX1[1:0] | | AX2[1:0] | | — | — | OS0[1:0] | | OS1[1:0] | | OS2[1:0] | | — | — | M2D | CLU_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--|---------------|---------------|-----|--|
| 31 to 29, 27 to 25, 23 to 16, 11, 10, 3, 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | AAI | B'0 | R/W | Automatic Table Address Increment Selects whether to specify the CLU table address in VI6_CLU_ADDR (refer to section 33.3.5.2) every time or update the address automatically. For details, refer to section 33.3.5.2 0: CLU table address should be specified every time 1: CLU table address is automatically incremented |
| 24 | MVS | B'0 | R/W | Max Value Stretch Select calculation method in max value region. 0: Method 0 (Lower compatible) 1: Method 1 (Improved characteristics) Setting 1 to MVS bit is recommended. |
| 15, 14 | AX1I [1:0] | B'00 | R/W | Input Control 0 in 2D Mode When setting the M2D bit to 0, be sure to set these bits to 0. When setting the M2D bit to 1, be sure to set these bits to 3. |
| 13, 12 | AX2I [1:0] | B'00 | R/W | Input Control 1 in 2D Mode When setting the M2D bit to 0, be sure to set these bits to 0. When setting the M2D bit to 1, be sure to set these bits to 1. |
| 9, 8 | OS0 [1:0] | B'00 | R/W | Output Control 0 in 2D Mode When setting the M2D bit to 0, be sure to set these bits to 0. When setting the M2D bit to 1, be sure to set these bits to 3. |
| 7, 6 | OS1 [1:0] | B'00 | R/W | Output Control 1 in 2D Mode When setting the M2D bit to 0, be sure to set these bits to 0. When setting the M2D bit to 1, be sure to set these bits to 1. |
| 5, 4 | OS2 [1:0] | B'00 | R/W | Output Control 2 in 2D Mode When setting the M2D bit to 0, be sure to set these bits to 0. When setting the M2D bit to 1, be sure to set these bits to 3. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | M2D | B'0 | R/W | <p>LUT Dimension Number</p> <p>Specifies the number of LUT dimensions. The details of each mode will be described later.</p> <p>2D mode can be used only when the CLU input color space is YCbCr.</p> <p>0: Operates in 3D mode 1: Operates in 2D mode</p> |
| 0 | CLU_EN | B'0 | R/W | <p>CLU Processing Enable/Disable</p> <p>Enables or disables the 3D/2D color correction function by the CLU. When the CLU is used, the color component information needs to be set in the CLU table. For how to set data in the CLU table, see section 33.3.5.2.</p> <p>0: CLU color correction is disabled 1: CLU color correction is enabled</p> |

The CLU handles three-dimensional or two-dimensional LUTs. A three-dimensional LUT is called a 3D-LUT and a two-dimensional LUT is called a 2D-LUT. The 3D or 2D operating mode of the CLU is selected by the M2D bit.

In 3D mode, a three-dimensional space such as that shown in Figure 33.42(a) is considered. The total number of coordinate points that can exist in this three-dimensional space is the total number of combinations of input data. For the CLU, this becomes the cube of each 8-bit component (= 16777216 points). Since it is impractical to have table values (= table memory) for the same number of these coordinate points, the CLU divides the three-dimensional space of Figure 33.42 (a) into grids as shown in Figure 33.42 (b) in 3D mode. The apex in each divided grid is defined as the coordinate point in the three-dimensional space, and a table value with three component values is set for all of these coordinate points. The total number of coordinate points in Figure 33.42 (b) is $17^3 = 4913$. For the method of setting the table value for these 4913 points, refer to section 33.3.5.2.

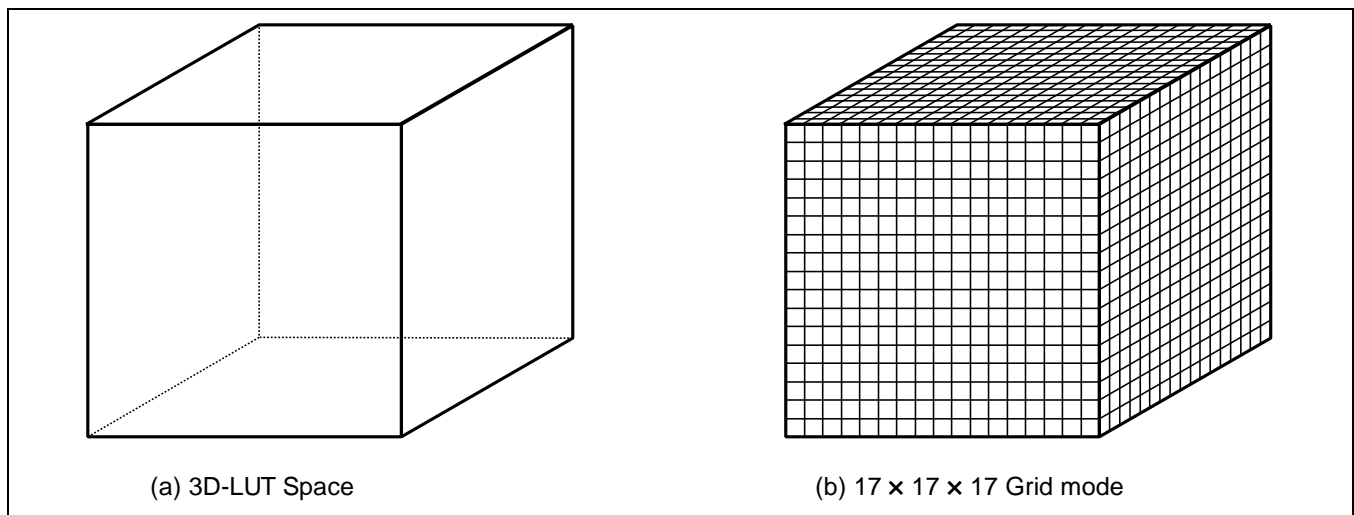


Figure 33.42 Concept and Division Count of a 3-Dimensional LUT

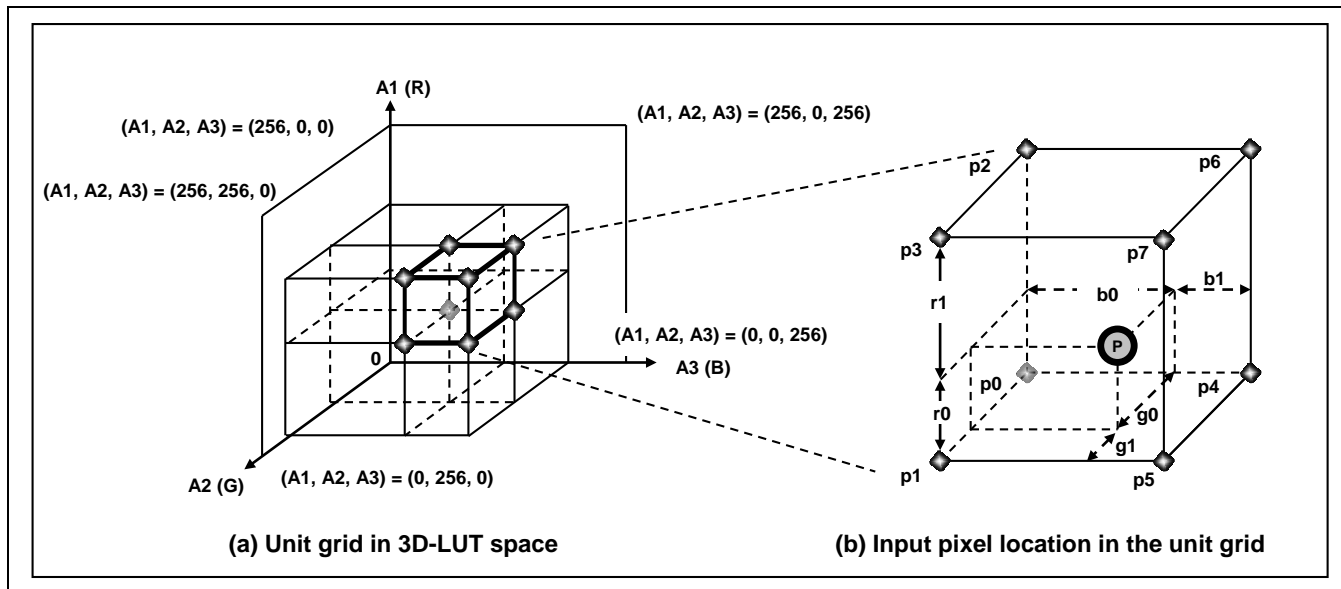


Figure 33.43 Conceptual Diagram of LUT Space When CLU Is in 3D Mode

To express a three-dimensional space such as that shown in Figure 33.42, consider the three-dimensional coordinates and axes shown in Figure 33.43 and define the minimum value and maximum value of each axis as 0 and 256, respectively. In 3D-LUT operation, the CLU forcibly assigns the four components of input data (α , R/Cr/H, G/Y/S, and B/Cb/V) to axes: R, Cr, and H components to the first axis (A1), G, Y, and S components to the second axis (A2), and B, Cb, and V components to the third axis (A3). (The case of 2D mode is described later.) Among multiple components assigned to each axis (e.g., R, Cr, and H components for the first axis), one component is selected according to the format of the data input to the CLU. For example, when the color space of the data input to the CLU is RGB, the first axis (A1) shows the R component, the second axis (A2) shows the G component, and the third axis (A3) shows the B component. In a YCbCr color space, the first axis (A1) shows the Cr component, the second axis (A2) shows the Y component, and the third axis (A3) shows the Cb component.

Each component value of the CLU input data becomes a value on each axis. Since the input data ranges from 0 to 255, the maximum value of 256 is handled as a virtual point in a grid space like that shown in Figure 33.42. In a case where the coordinates obtained by plotting the values of the input three components on axes are exactly on a grid point (e.g., p0 in Figure 33.43 (b)), the 3-component value of the table value (p0) set to that location becomes the CLU output. If the input value of the three components result in an intermediate location between grid points and not on a grid point, such as position P in Figure 33.43 (b), value P of the three components is interpolated from the table values (p0 to p7 in Figure 33.43 (b)) for the eight surrounding grid points and this is used as the CLU output.

By setting arbitrary table value for a three-dimensional LUT space (three-dimensional color space) such as that shown in Figure 33.42 using the processing described above, not only color conversion or color correction for the entire color space, but color conversion or color correction (memory color correction, skin smoothing, etc.) for a particular color area can be achieved.

Next, a 2D-LUT that is used when the CLU is operating in 2D mode is described (M2D bit should be set to 1 for operation in 2D mode). 2D-LUT mode is similar to 3D-LUT mode. However, since the axes for selecting the LUT become two-dimensional, the concept of a 3D-LUT space illustrated in Figure 33.43 changes to a 2D-LUT space like that shown in Figure 33.44.

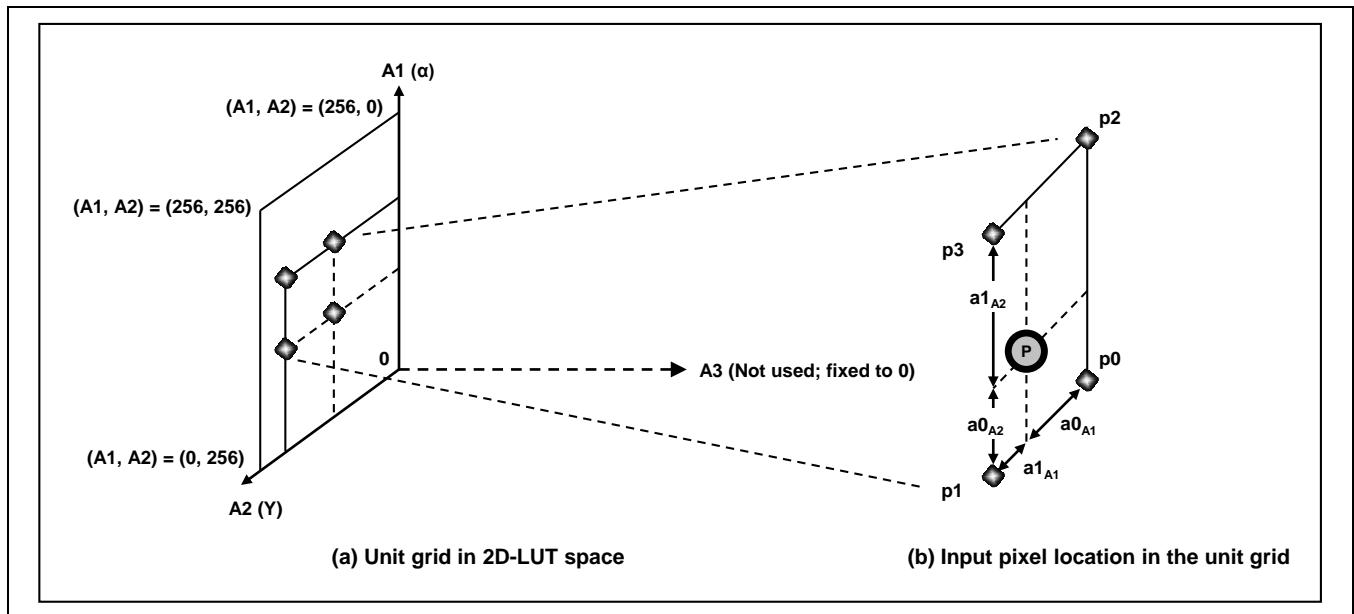


Figure 33.44 Conceptual Diagram of LUT Space When CLU Is in 2D Mode

As shown in Figure 33.44 (a), the first axis (A1) and second axis (A2) are used, and the A1-A2 plane becomes the LUT space in 2D mode. Because the third axis (A3) is not used, the CLU always handles the value on the A3 axis as 0.

2D mode can be used only when the CLU input is in the YCbCr color space. The input value of axis A1 is the input α value of the CLU, and the input value of axis A2 is the input luminance Y of the CLU. If the coordinates in the 2D-LUT space which were determined by the input values are exactly on a grid point (e.g., p2 in Figure 33.44 (b)), the table value (p2) set to that location becomes the output value of the 2D-LUT. If the input 2-component value exists at an intermediate location between grid points and not on a grid point, such as position P in Figure 33.44 (b), value P is interpolated from the table values (p0 to p3 in Figure 33.44 (b)) for the four surrounding grid points and this is used as the 2D-LUT output. The interpolated component of the second axis of the 2D-LUT output acquired in this manner becomes the Y component output of the CLU. In 2D mode, the Cb/Cr component is a pass-through output.

For the CLU table values in Figure 33.44 in 2D mode, set a value for only the component value of the second axis, and set 0 for the component values of the first and third axes. For details on the setting method, see section 33.3.5.2.

33.2.13 HST Control Register

33.2.13.1 HST Control Register (VI6_HST_CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | HST_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | HST_EN | B'0 | R/W | HSV Conversion (RGB → HSV) Enable/Disable Enables or disables RGB → HSV conversion. H indicates hue, S indicates saturation, and V indicates a value (brightness). 0: RGB → HSV conversion is disabled 1: RGB → HSV conversion is enabled |

Various color correction processing can be applied to HSV-format images by combining the 1D-LUT and 3D-LUT. For details on the 1D-LUT and 3D-LUT, see sections 33.2.11.1 and 33.2.12.1.

To return HSV-converted data to RGB data, HSV → RGB conversion is used (see section 33.2.14.1). RGB ↔ HSV conversion causes an operation error. Therefore, even when only just performing conversion of RGB → HSV → RGB, the original RGB value and converted RGB value will not completely match.

33.2.14 HSI Control Register

33.2.14.1 HSI Control Register (VI6_HSI_CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | HSI_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | HSI_EN | B'0 | R/W | Reversed HSV Conversion (HSV → RGB) Enable/Disable Enables or disables HSV → RGB conversion to return HSV data that was converted by HSV conversion (see section 33.2.13.1) back to RGB data. 0: HSV → RGB conversion is disabled 1: HSV → RGB conversion is enabled |

RGB ↔ HSV conversion causes an operation error. Therefore, even when only just performing conversion of RGB → HSV → RGB, the original RGB value and converted RGB value will not completely match.

33.2.15 BRU Control Registers

33.2.15.1 BRU Input Control Register (VI6_BRU_INCTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | | | |
|----------------|------|-----|------------|-----|------|--------------|-----|-------------|-----|-----|------|------|-------------|------|------|------|-------------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | NRM | ODE4 | DITH 4 [2:0] | | | — | — | — | D4ON | D3ON | D2ON | D1ON | D0ON | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | ODE3 | | DITH3[2:0] | | | ODE2 | | DITH2 [2:0] | | | ODE1 | | DITH1 [2:0] | | ODE0 | | DITH0 [2:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|------------|---------------|-----|--|
| 31 to 29, 23 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | NRM | B'0 | R/W | Color Data Normalization Enables or disables division by the α value of the color data in BRU blending operation. This is used when converting the RGB color data format to which the α value is multiplied (pre-multiplied color) into the RGB color data format to which the α value is not multiplied (non pre-multiplied color). Do not use this for the YCbCr format. 0: Divider (DIV unit in Figure 33.45) does not divide the color value by α 1: Divider (DIV unit in Figure 33.45) divides the color value by α |
| 27 | ODE4 | B'0 | R/W | Ordered Dither (mode A) of CH4 Input to BRU Enable/Disable 0: Ordered dither (mode A) of BRUin4 is disabled. 1: Ordered dither (mode A) of BRUin4 is enabled. Ordered dither is available only for 18bpp. So, when ODE4 bit is equal to 1, set DITH4 = 1. When ODE4 bit is equal to 0, BRUin4 dither method is specified by D4ON in the register Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp. |
| 26 to 24 | DITH4[2:0] | B'000 | R/W | Dithering of CH4 Input to BRU These bits specify how to perform dithering of the CH4 input to the BRU. The setting method is the same as that for the DITH3 bits. Read the description of the DITH3 bits with BRUin4 and D4ON replacing BRUin4 and D4ON, respectively. |
| 20 | D4ON | B'0 | R/W | Ordered dither (mode B) Enable of BRU Input 4 Enables or disables dithering (color reduction) of BRU input 4 (BRUin4 in Figure 33.45)). 0: Dithering (mode B) of BRUin4 is disabled 1: Dithering (mode B) of BRUin4 is enabled When ODE4 in this bit is set to 1, set 0 to this bit. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 19 | D3ON | B'0 | R/W | <p>Ordered dither (mode B) Enable of BRU Input 3</p> <p>Enables or disables dithering (color reduction) of BRU input 3 (BRUin3 in Figure 33.45).</p> <p>0: Dithering (mode B) of BRUin3 is disabled</p> <p>1: Dithering (mode B) of BRUin3 is enabled</p> <p>When ODE3 in this bit is set to 1, set 0 to this bit.</p> |
| 18 | D2ON | B'0 | R/W | <p>Ordered dither (mode B) Enable of BRU Input 2</p> <p>Enables or disables dithering (color reduction) of BRU input 2 (BRUin2 in Figure 33.45).</p> <p>0: Dithering (mode B) of BRUin2 is disabled</p> <p>1: Dithering (mode B) of BRUin2 is enabled</p> <p>When ODE2 in this bit is set to 1, set 0 to this bit.</p> |
| 17 | D1ON | B'0 | R/W | <p>Ordered dither (mode B) Enable of BRU Input 1</p> <p>Enables or disables dithering (color reduction) of BRU input 1 (BRUin1 in Figure 33.45).</p> <p>0: Dithering (mode B) of BRUin1 is disabled</p> <p>1: Dithering (mode B) of BRUin1 is enabled</p> <p>When ODE1 in this bit is set to 1, set 0 to this bit.</p> |
| 16 | D0ON | B'0 | R/W | <p>Ordered dither (mode B) Enable of BRU Input 0</p> <p>Enables or disables dithering (color reduction) of BRU input 0 (BRUin0 in Figure 33.45).</p> <p>0: Dithering (mode B) of BRUin0 is disabled</p> <p>1: Dithering (mode B) of BRUin0 is enabled</p> <p>When ODE0 in this bit is set to 1, set 0 to this bit.</p> |
| 15 | ODE3 | B'0 | R/W | <p>Ordered Dither (mode A) of CH3 Input to BRU Enable/Disable</p> <p>0: Ordered dither (mode A) of BRUin3 is disabled.</p> <p>1: Ordered dither (mode A) of BRUin3 is enabled.</p> <p>Ordered dither is available only for 18bpp. So, when ODE3 bit is equal to 1, set DITH3 = 1.</p> <p>When ODE3 bit is equal to 0, BRUin3 dither method is specified by D3ON in the register</p> <p>Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp.</p> |
| 14 to 12 | DITH3[2:0] | B'000 | R/W | <p>Dithering of CH3 Input to BRU</p> <p>These bits specify the number of colors for pixels after dithering (color reduction) when dithering (color reduction) for pixel information is enabled through the D3ON bit. When dithering (color reduction) for pixel information is disabled, specify 0 in these bits.</p> <p>B'000: Dithering of BRUin3 input image is disabled</p> <p>B'001: Dithering of BRUin3 input image at 18 bpp (RGB666: 260000 colors)</p> <p>B'010: Dithering of BRUin3 input image at 16 bpp (RGB565: 65535 colors)</p> <p>B'011: Dithering of BRUin3 input image at 15 bpp (RGB555: 32768 colors)</p> <p>B'100: Dithering of BRUin3 input image at 12 bpp (RGB444: 4096 colors)</p> <p>B'101: Dithering of BRUin3 input image at 8 bpp (RGB332: 256 colors)</p> <p>B'110, B'111: Setting prohibited</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|---|
| 11 | ODE2 | B'0 | R/W | <p>Ordered Dither (mode A) of CH2 Input to BRU Enable/Disable</p> <p>0: Ordered dither (mode A) of BRUin2 is disabled.</p> <p>1: Ordered dither (mode A) of BRUin2 is enabled.</p> <p>Ordered dither is available only for 18bpp. So, when ODE2 bit is equal to 1, set DITH2 = 1.</p> <p>When ODE2 bit is equal to 0, BRUin2 dither method is specified by D2ON in the register</p> <p>Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp.</p> |
| 10 to 8 | DITH2 [2:0] | B'000 | R/W | <p>Dithering of CH2 Input to BRU</p> <p>These bits specify how to perform dithering of the CH2 input to the BRU. The setting method is the same as that for the DITH3 bits. Read the description of the DITH3 bits with BRUin2 and D2ON replacing BRUin3 and D3ON, respectively.</p> |
| 7 | ODE1 | B'0 | R/W | <p>Ordered Dither (mode A) of CH1 Input to BRU Enable/Disable</p> <p>0: Ordered dither (mode A) of BRUin1 is disabled.</p> <p>1: Ordered dither (mode A) of BRUin1 is enabled.</p> <p>Ordered dither is available only for 18bpp. So, when ODE1 bit is equal to 1, set DITH1 = 1.</p> <p>When ODE1 bit is equal to 0, BRUin1 dither method is specified by D1ON in the register</p> <p>Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp.</p> |
| 6 to 4 | DITH1 [2:0] | B'000 | R/W | <p>Dithering of CH1 Input to BRU</p> <p>These bits specify how to perform dithering of the CH1 input to the BRU. The setting method is the same as that for the DITH3 bits. Read the description of the DITH3 bits with BRUin1 and D1ON replacing BRUin3 and D3ON, respectively.</p> |
| 3 | ODE0 | B'0 | R/W | <p>Ordered Dither (mode A) of CH0 Input to BRU Enable/Disable</p> <p>0: Ordered dither (mode A) of BRUin0 is disabled.</p> <p>1: Ordered dither (mode A) of BRUin0 is enabled.</p> <p>Ordered dither is available only for 18bpp. So, when ODE0 bit is equal to 1, set DITH0 = 1.</p> <p>When ODE0 bit is equal to 0, BRUin0 dither method is specified by D0ON in the register</p> <p>Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp.</p> |
| 2 to 0 | DITH0 [2:0] | B'000 | R/W | <p>Dithering of CH0 Input to BRU</p> <p>These bits specify how to perform dithering of the CH0 input to the BRU. The setting method is the same as that for the DITH3 bits. Read the description of the DITH3 bits with BRUin0 and D0ON replacing BRUin3 and D3ON, respectively.</p> |

Figure 33.45 shows the configuration of the BRU. For the BRU inputs, there are five inputs from the DPR and one internal input as a virtual RPF. BRUin0 to BRUin4 are input ports that have the target node values shown in Figure 33.31 to Figure 33.32, and they can be connected to any module on the DPR. The same color space (YCbCr or RGB) has to be used for the five inputs from the DPR to the BRU.

The virtual RPF inside the BRU is an input unit not connected to the DPR. It is called the "virtual RPF" because it outputs images internally created by the BRU. Starting of the virtual RPF is controlled by VI6_WPFn_SRCRPF.VIR_ACT, and the single-color data created at the virtual RPF can be used for blending or raster operation (ROP) with data from the

other input units BRUin0 to BRUin4. The color space for the single color to be set for the virtual RPF needs to match the color space of the five inputs from the DPR to the BRU. For this setting method, see section 33.2.15.4.

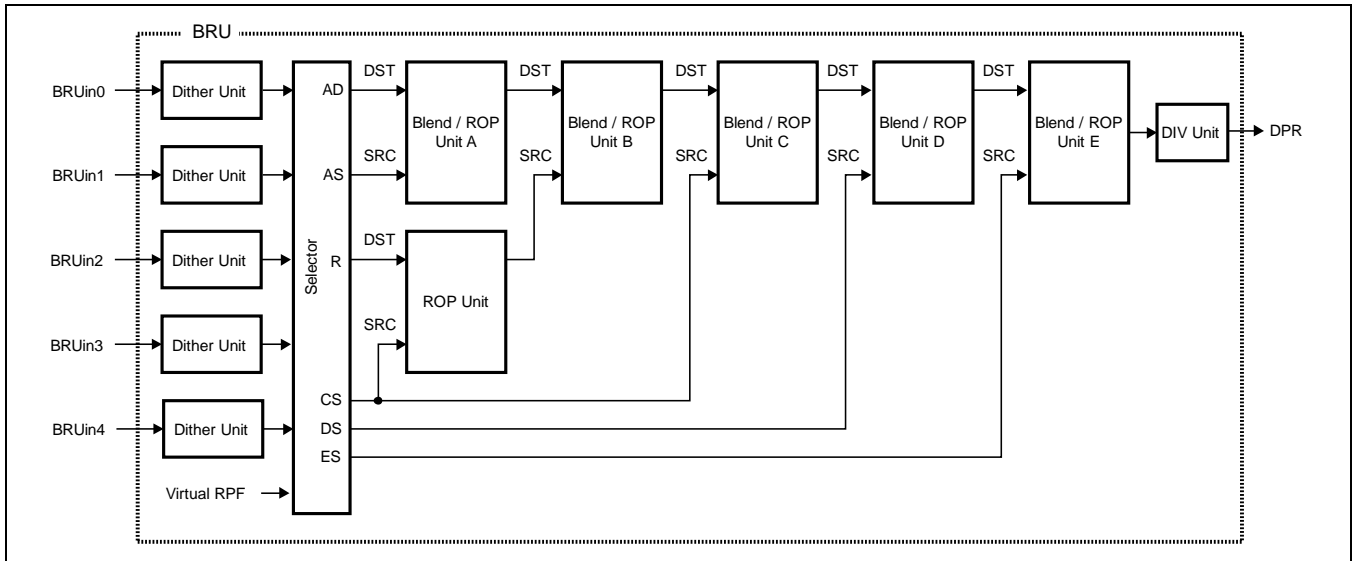


Figure 33.45 BRU Configuration

The selector in Figure 33.45 is used to select the SRC and DST inputs to blending/ROP units A to E and the ROP unit from BRUin0 to BRUin4 which are inputs from the DPR and the virtual RPF. The SRC and DST input sources for blending/ROP units A to E and the ROP unit are either uniquely determined based on the configuration shown in Figure 33.45 (Table 33.32) or selected as desired by registers. The input sources that can be arbitrarily selected by registers are AD, AS, R, CS, DS and ES, which correspond to the registers shown in Table 33.33.

Table 33.32 SRC and DST with Unique Input Sources

| Input | Input Source | Register Bits |
|---------------------------|---------------------------------|----------------------|
| Blending/ROP unit B - DST | Output from blending/ROP unit A | — |
| Blending/ROP unit B - SRC | Output from ROP unit | — |
| Blending/ROP unit C - DST | Output from blending/ROP unit B | — |
| Blending/ROP unit D - DST | Output from blending/ROP unit C | — |
| Blending/ROP unit E - DST | Output from blending/ROP unit D | — |
| ROP unit - SRC | CS which is a selector output | VI6_BRUC_CTRL.SRCSEL |

Table 33.33 Correspondence between Selector Output Destinations and Register Bits

| Selector Output | Output Destination | Register Bits |
|-----------------|---------------------------|----------------------|
| AD | Blending/ROP unit A - DST | VI6_BRUA_CTRL.DSTSEL |
| AS | Blending/ROP unit A - SRC | VI6_BRUA_CTRL.SRCSEL |
| R | ROP unit - DST | VI6_BRU_ROP.DSTSEL |
| CS | Blending/ROP unit C - SRC | VI6_BRUC_CTRL.SRCSEL |
| DS | Blending/ROP unit D - SRC | VI6_BRUD_CTRL.SRCSEL |
| ES | Blending/ROP unit E - SRC | VI6_BRUE_CTRL.SRCSEL |

33.2.15.2 Size Register of BRU Input Virtual RPF (VI6_BRU_VIRRPF_SIZE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | VIR_HSIZE[12:0] | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | VIR_VSIZE[12:0] | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

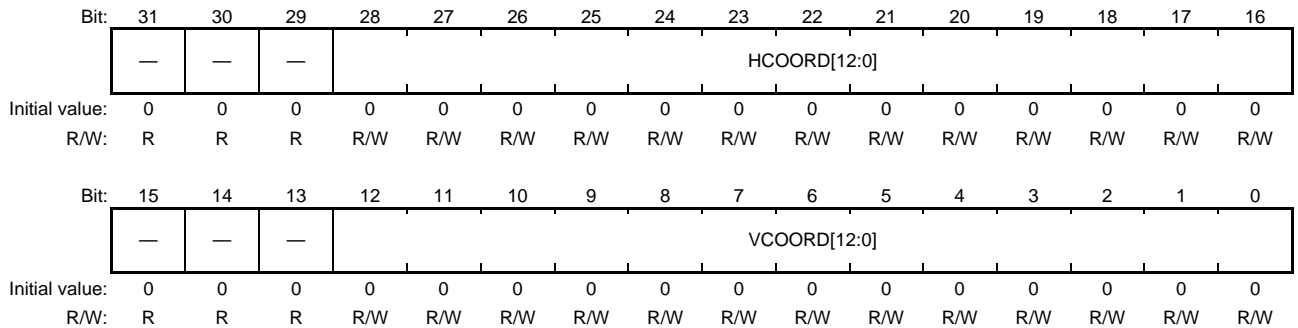
| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|------------------|---------------|-----|--|
| 31 to 29, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | VIR_HSIZE [12:0] | All 0 | R/W | Virtual RPF Horizontal Size These bits set the horizontal size of an image from the virtual RPF shown in Figure 33.45. A value from 1 to 8190 can be specified. |
| 12 to 0 | VIR_VSIZE [12:0] | All 0 | R/W | Virtual RPF Vertical Size These bits set the vertical size of an image from the virtual RPF shown in Figure 33.45. A value from 1 to 8190 can be specified. |

The virtual RPF has only a function to output a fixed α value and a fixed pixel value. The virtual RPF can internally create a single-color image without accessing external memory via the MAU. Same as images from the other BRU input ports, a sublayer can be blended on an image created in this manner with the image used as the background (master layer). In turn, when using the image as a sublayer, it can be drawn on the master layer as a window.

Note that the virtual RPF is fixed to input port 5 of the BRU, and the data path route cannot be changed by the DPR.

33.2.15.3 Display Location Register of BRU Input Virtual RPF (VI6_BRU_VIRRPF_LOC)

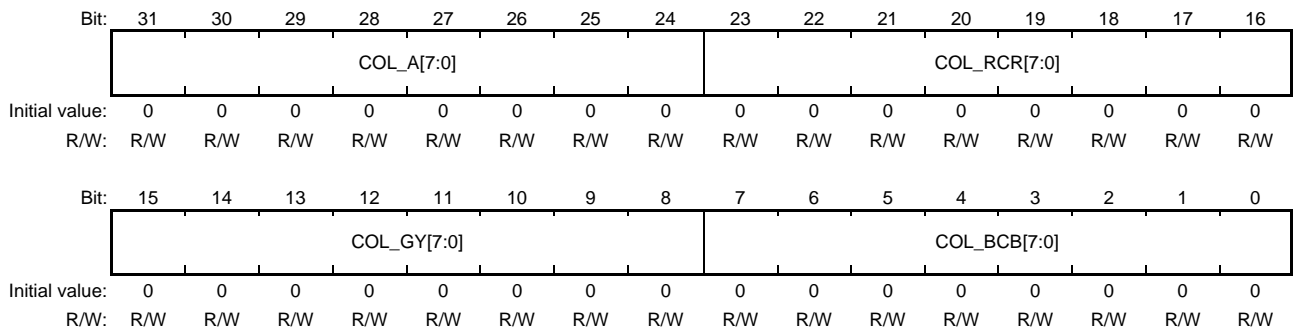
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|---------------|---------------|-----|---|
| 31 to 29, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | HCOORD [12:0] | All 0 | R/W | Horizontal Coordinate of Virtual RPF Location on Master Layer These bits specify the horizontal coordinate of where to locate the left-edge pixel of the virtual RPF's layer, with the left-edge pixel of the master layer set at coordinate 0. This setting should be made in pixel units. A value from 0 to 8189 can be specified. When the virtual RPF is specified as the master layer by VI6_WPFn_SRCRPF.VIR_ACT, set these bits to 0. |
| 12 to 0 | VCOORD [12:0] | All 0 | R/W | Vertical Coordinate of Virtual RPF Location on Master Layer These bits specify the vertical coordinate of where to locate the top-edge pixel of the virtual RPF's layer, with the top-edge pixel of the master layer set at coordinate 0. This setting should be made in pixel units. A value from 0 to 8189 can be specified. When the virtual RPF is specified as the master layer by VI6_WPFn_SRCRPF.VIR_ACT, set these bits to 0. |

33.2.15.4 Color Information Register of BRU Input Virtual RPF (VI6_BRU_VIRRRPF_COL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 24 | COL_A [7:0] | All 0 | R/W | Fixed α of Virtual RPF These bits set the fixed α value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified. |
| 23 to 16 | COL_RCR [7:0] | All 0 | R/W | Fixed R/Cr of Virtual RPF These bits set the fixed R/Cr value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified. |
| 15 to 8 | COL_GY [7:0] | All 0 | R/W | Fixed G/Y of Virtual RPF These bits set the fixed G/Y value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified. |
| 7 to 0 | COL_BCB [7:0] | All 0 | R/W | Fixed B/Cb of Virtual RPF These bits set the fixed B/Cb value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified. |

The transparency information and color information of the single color that is created by the virtual RPF are set in the bits of this register. As described earlier, the color information is set for the YCbCr or RGB color space. The color space to be set in this register depends on the register settings of the environment and other modules to which the BRU is connected by the DPR. Two cases can be considered. Since the α value (COL_A) is transparency information and irrelevant to the concept of color space, the same setting is made for either the YCbCr or RGB color space.

(Case 1: When an input other than the virtual RPF is used)

When the source RPF is connected to any one of the BRU input ports (BRUin0 to BRUin4) other than the virtual RPF and valid data is being supplied, the same color space data as the color space for the BRU inputs should be set in this register as the color space for the virtual RPF's color information. This is based on the restriction of "all BRU inputs must have the same color space", as described in section 33.2.8.1 or 33.2.15.1.

(Case 2: When only the virtual RPF is used)

When only the virtual RPF is used as the source RPF of WPFn, RPFn is not connected to the BRU, as shown in Figure 33.46. Thus, there is no color space for another input that the color space for the virtual RPF has to follow, as in case 1. This means that the color space of the data output by the BRU is determined by the WPF setting.

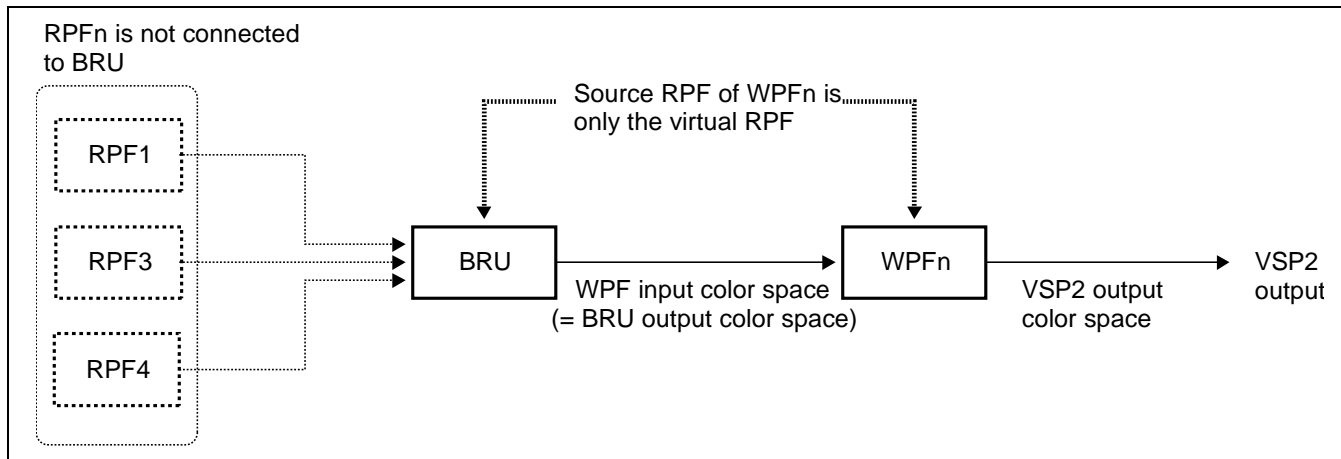


Figure 33.46 Relationship between DPR Connection and Color Space When Only Virtual RPF is Used

Figure 33.46, the output color space of the VSP2 (= output color space of WPFn) is determined by VI6_WPFn_OUTFMT.WRFMT. When bit 6 in VI6_WPFn_OUTFMT.WRFMT (WRFMT [6]) is 0, the color space is RGB, while when it is 1, the color space is YCbCr. Next, the WPF input color space (= BRU output color space) is determined by the relationship between the WPF output color space and VI6_WPFn_OUTFMT.CSC. When VI6_WPFn_OUTFMT.CSC is 0, the WPF output color space and WPF input color space (= BRU output color space) are the same. When VI6_WPFn_OUTFMT.CSC is 1, the WPF output color space and WPF input color space (= BRU output color space) are the opposite. This relationship is summarized in Table 33.34.

The color space for the virtual RPF's color information should be set in this register according to the "BRU output color space" shown in Table 33.34.

Table 33.34 Relationship between WPF Output Color Space and BRU Output Color Space

| VI6_WPFn_OUTFMT Register Bit Settings | | BRU Output Color Space (= WPF Input Color Space) | |
|---------------------------------------|--|--|--|
| Bit 6 in WRFMT | CSC | | |
| 0 (WPF output is RGB) | 0 (YCbCr → RGB conversion is disabled) | RGB | |
| 0 (WPF output is RGB) | 1 (YCbCr → RGB conversion is enabled) | YCbCr | |
| 1 (WPF output is YCbCr) | 0 (RGB → YCbCr conversion is disabled) | YCbCr | |
| 1 (WPF output is YCbCr) | 1 (RGB → YCbCr conversion is enabled) | RGB | |

33.2.15.5 BRU Control Registers (VI6_BRUm_CTRL: m = A, B, C, D, E)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|----|----|----|----|----|----|----|-----|--------------|-----|-----|------------|--------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RBC | — | — | — | — | — | — | — | — | DSTSEL [2:0] | | | — | SRCSEL [2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | CROP [3:0] | | | AROP [3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----------------------|--------------|---------------|-----|---|
| 31 | RBC | B'0 | R/W | Operation Type of Blending/ROP Unit m (m = A, B, C, D, E) Specifies the operation type for blending/ROP unit m (m = A, B, C, D, E) shown in Figure 33.45. 0: ROP (raster operation) 1: Blending operation |
| 30 to 23, 19, 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 22 to 20 | DSTSEL [2:0] | B'000 | R/W | Input Selection for DST Side of Blending/ROP Unit A These bits select the input for the DST side of blending/ROP unit A shown in Figure 33.45. These bits specify the connection between the BRU input port and the DST separately from the setting of connections between other modules and the BRU input port through the DPR. B'000: BRU input 0 (BRUin0) is input to DST B'001: BRU input 1 (BRUin1) is input to DST B'010: BRU input 2 (BRUin2) is input to DST B'011: BRU input 3 (BRUin3) is input to DST B'100: Virtual RPF is input to DST B'101: BRU input 4 (BRUin4) is input to DST B'110 to B'111: Setting prohibited [Note] The DSTSEL bits for blending/ROP unit m (m = B, C, D, E) are reserved according to Table 33.32. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|---|
| 18 to 16 | SRCSEL [2:0] | B'000 | R/W | <p>Input Selection for SRC Side of Blending/ROP Unit m (m = A, C, D, E)</p> <p>These bits select the input for the SRC side of blending/ROP unit m (m = A, C, D, E) shown in Figure 33.45. These bits specify the connection between the BRU input port and the SRC separately from the setting of connections between other modules and the BRU input port through the DPR.</p> <p>B'000: BRU input 0 (BRUin0) is input to SRC B'001: BRU input 1 (BRUin1) is input to SRC B'010: BRU input 2 (BRUin2) is input to SRC B'011: BRU input 3 (BRUin3) is input to SRC B'100: Virtual RPF is input to SRC B'101: BRU input 4 (BRUin4) is input to SRC B'110 to B'111: Setting prohibited</p> <p>[Note] The SRCSEL bits for blending/ROP unit B are reserved according to Table 33.32. The write value should always be 0.</p> |
| 7 to 4 | CROP [3:0] | H'0 | R/W | <p>Color Data ROP Operator</p> <p>These bits select the ROP operator of the color data in blending/ROP unit m (m = A, B, C, D, E). Select the opcode for ROP operation from Table 33.35.</p> |
| 3 to 0 | AROP [3:0] | H'0 | R/W | <p>α Data ROP Operator</p> <p>These bits select the ROP operator of the α data in blending/ROP unit m (m = A, B, C, D, E). Select the opcode for ROP operation from Table 33.35.</p> |

Table 33.35 ROP Operator of Blending/ROP Unit m (m = A, B, C, D)

| Opcode | Operator |
|--------|----------------------|
| B'0000 | NOP(D) |
| B'0001 | AND(S & D) |
| B'0010 | AND_REVERSE(S & ~D) |
| B'0011 | COPY(S) |
| B'0100 | AND_INVERTED(~S & D) |
| B'0101 | CLEAR(0) |
| B'0110 | XOR(S ^ D) |
| B'0111 | OR(S D) |
| B'1000 | NOR(~(S D)) |
| B'1001 | EQUIV(~(S ^ D)) |
| B'1010 | INVERT(~D) |
| B'1011 | OR_REVERSE(S ~D) |
| B'1100 | COPY_INVERTED(~S) |
| B'1101 | OR_INVERTED(~S D) |
| B'1110 | NAND(~(S & D)) |
| B'1111 | SET(all1) |

33.2.15.6 BRU Blend Control Registers (VI6_BRUm_BLD: m = A, B, C, D, E)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|-----|-----|-----|------------|-----|-----|------------|------------|-----|-----|-----|------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CBES | CCMDX[2:0] | | | — | CCMDY[2:0] | | | ABES | ACMDX[2:0] | | | — | ACMDY[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | COEFX[7:0] | | | | | | | | COEFY[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 | CBES | B'0 | R/W | Blending Expression Selection Selects the blending expression of the color data in the BRU (VI6_BRUm_CTRL.RBC = 1). Blending coefficients are specified by the CCMDX and CCMDY bits. 0: CCMDX * (DST color data) + CCMDY * (SRC color data) 1: CCMDX * (DST color data) – CCMDY * (SRC color data) |
| 30 to 28 | CCMDX [2:0] | B'000 | R/W | Blending Coefficient X Selection These bits specify coefficient X used in the blending expression determined by the CBES bit. B'000: DST α data is used as blending coefficient X B'001: 255 – (DST α data) is used as blending coefficient X B'010: SRC α data is used as blending coefficient X B'011: 255 – (SRC α data) is used as blending coefficient X B'100: Fixed α value 0 (COEFX setting) |
| 27, 19 | — | All 0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 26 to 24 | CCMDY [2:0] | B'000 | R/W | Blending Coefficient Y Selection These bits specify coefficient Y used in the blending expression determined by the CBES bit. B'000: DST α data is used as blending coefficient Y B'001: 255 – (DST α data) is used as blending coefficient Y B'010: SRC α data is used as blending coefficient Y B'011: 255 – (SRC α data) is used as blending coefficient Y B'100: Fixed α value 1 (COEFY setting) |
| 23 | ABES | B'0 | R/W | Blending α Creation Expression Specifies the expression for creating α data after blending by blending/ROP unit m (m = A, B, C, D, E). α creation coefficients are specified by the ACMDX and ACMDY bits. 0: ACMDX * (DST α data) + ACMDY * (SRC α data) 1: ACMDX * (DST α data) – ACMDY * (SRC α data) |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 22 to 20 | ACMDX [2:0] | B'000 | R/W | <p>α Creation Coefficient X</p> <p>These bits specify α creation coefficient X used in the α creation expression determined by the ABES bit.</p> <p>B'000: (α creation coefficient X) = (DST α data)</p> <p>B'001: (α creation coefficient X) = 255 – (DST α data)</p> <p>B'010: (α creation coefficient X) = (SRC α data)</p> <p>B'011: (α creation coefficient X) = 255 – (SRC α data)</p> <p>B'100: (α creation coefficient X) = Fixed α value 0 (COEFX setting)</p> <p>B'101 to B'111: Setting prohibited</p> |
| 18 to 16 | ACMDY [2:0] | B'000 | R/W | <p>α Creation Coefficient Y</p> <p>These bits specify α creation coefficient Y used in the α creation expression determined by the ABES bit.</p> <p>B'000: (α creation coefficient Y) = (DST α data)</p> <p>B'001: (α creation coefficient Y) = 255 – (DST α data)</p> <p>B'010: (α creation coefficient Y) = (SRC α data)</p> <p>B'011: (α creation coefficient Y) = 255 – (SRC α data)</p> <p>B'100: (α creation coefficient Y) = Fixed α value 1 (COEFY setting)</p> <p>B'101 to B'111: Setting prohibited</p> |
| 15 to 8 | COEFX [7:0] | All 0 | R/W | <p>Fixed α Value 0</p> <p>These bits specify fixed α value 0 used when the CCMDX or ACMDX bits are set to B'100. A value from H'00 to H'FF can be specified.</p> |
| 7 to 0 | COEFY [7:0] | All 0 | R/W | <p>Fixed α Value 1</p> <p>These bits specify fixed α value 1 used when the CCMDY or ACMDY bits are set to B'100. A value from H'00 to H'FF can be specified.</p> |

33.2.15.7 BRU Raster Operation Control Register (VI6_BRU_ROP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-----------|-------------|-----|-----------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | DSTSEL[2:0] | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | CROP[3:0] | | | AROP[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-------------------|--------------|---------------|-----|---|
| 31 to 23, 19 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 22 to 20 | DSTSEL [2:0] | B'000 | R/W | Input Selection for DST Side of ROP Unit These bits select the input for the DST side of the ROP unit. For the SRC side, the same data as the input for the SRC side of blending/ROP unit C is input. B'000: BRU input 0 (BRUin0) is input to DST B'001: BRU input 1 (BRUin1) is input to DST B'010: BRU input 2 (BRUin2) is input to DST B'011: BRU input 3 (BRUin3) is input to DST B'100: Virtual RPF is input to DST B'101: BRU input 4 (BRUin4) is input to DST B'110 to B'111: Setting prohibited |
| 7 to 4 | CROP [3:0] | H'0 | R/W | Color Data ROP Operator These bits select the ROP operator of the color data in the ROP unit. Select the opcode for ROP operation from Table 33.35. |
| 3 to 0 | AROP [3:0] | H'0 | R/W | α Data ROP Operator These bits select the ROP operator of the α data in the ROP unit. Select the opcode for ROP operation from Table 33.35. |

33.2.16 HGO Control Registers

33.2.16.1 HGO Detection Window Offset Register (VI6_HGO_OFFSET)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|---------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | HOFFSET[13:0] | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | VOFFSET[13:0] | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------------|---------------|---------------|-----|---|
| 31, 30, 15, 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 16 | HOFFSET[13:0] | All 0 | R/W | Horizontal Offset of Histogram Detection Window The HGO creates a histogram for the detection window in the input image (Figure 33.47). In these bits, specify the value of horizontal offset (hoffset shown in Figure 33.47) in pixel units. A value from 0 to 8189 can be specified. The value of HOFFSET shall be smaller than that of the input image size of HGO. |
| 13 to 0 | VOFFSET[13:0] | All 0 | R/W | Vertical Offset of Histogram Detection Window The HGO creates a histogram for the detection window in the input image (Figure 33.47). In these bits, specify the value of vertical offset (voffset shown in Figure 33.47) in pixel units. A value from 0 to 8189 can be specified. The value of VOFFSET shall be smaller than that of the input image size of HGO. |

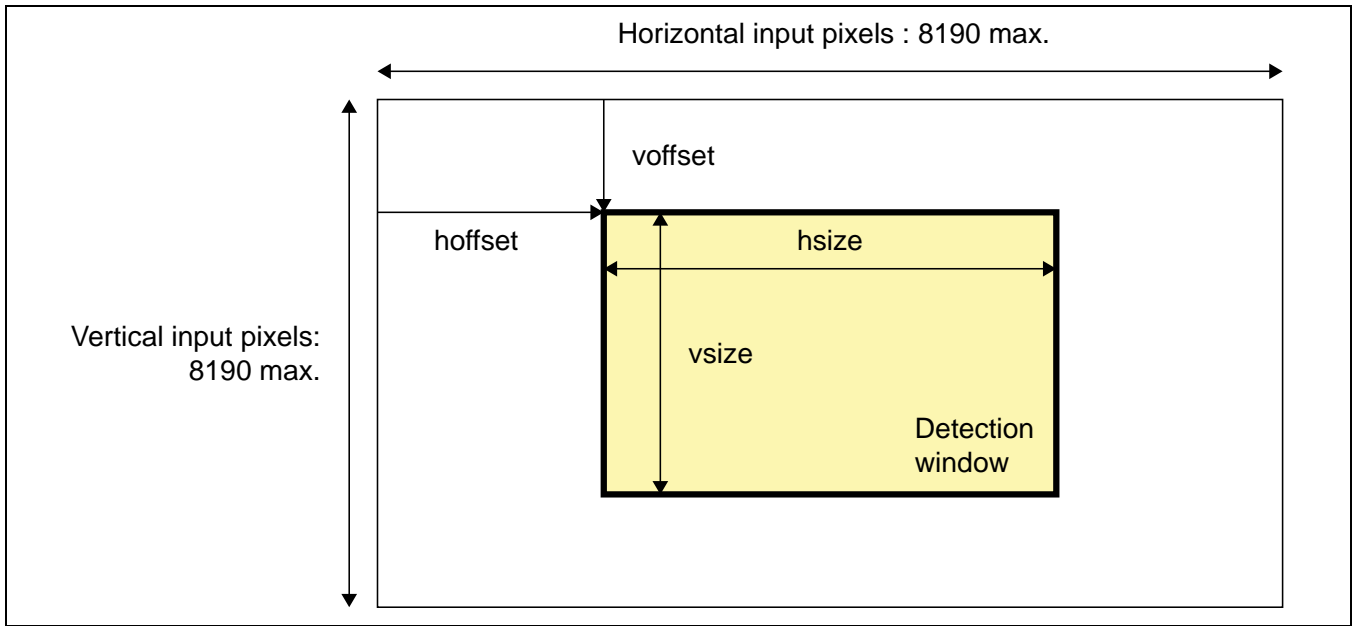
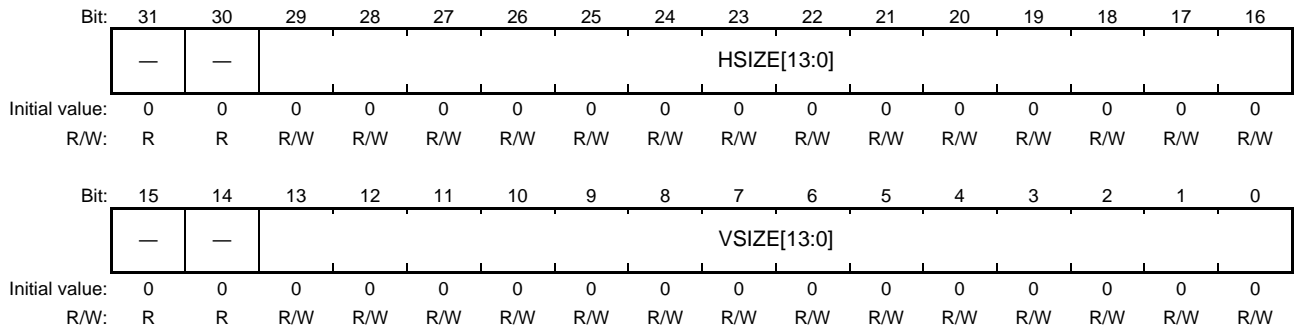


Figure 33.47 Histogram Detection Window of HGO

33.2.16.2 HGO Detection Window Size Register (VI6_HGO_SIZE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------------|-------------|---------------|-----|---|
| 31, 30, 15, 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 16 | HSIZE[13:0] | All 0 | R/W | Horizontal Size of Histogram Detection Window The HGO creates a histogram for the detection window in the input image (Figure 33.47). In these bits, specify the value of horizontal size (hsize shown in Figure 33.47) in pixel units. A value from 1 to 8190 can be specified. |
| 13 to 0 | VSIZE[13:0] | All 0 | R/W | Vertical Size of Histogram Detection Window The HGO creates a histogram for the detection window in the input image (Figure 33.47). In these bits, specify the value of vertical size (vsize shown in Figure 33.47) in pixel units. A value from 1 to 8190 can be specified. |

33.2.16.3 HGO Mode Register (VI6_HGO_MODE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----------|----|----------|----|----|------------|------------|------------|------------|-------------|-------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | AUT OW | — | STE P | — | — | MAXR GB | OFS B_R | OFS B_G | OFS B_B | HRATIO[1:0] | VRATIO[1:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|----------|---------------|-----|---|
| 31 to 13, 11, 9, 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | AUTOW | B'0 | R/W | HGO automatically histogram write. [RZ/G2H/RZ/G2N, RZ/G2E] 0: Histogram is not dumped to external memory automatically. S/W needs to dump histogram from histogram registers to external memory. 1: Histogram is dumped to SDRAM automatically. Dump format is shown in Figure 33.48. HGO histogram swapping register in section 33.2.16.16 specifies data order in 4byte unit as Table 33.36. |
| 10 | STEP | B'0 | R/W | Histogram step of Y/MaxRGB. 0: 64 step mode. 1: 256 step mode for Y/MaxRGB histogram. Restriction of color format is shown in Table 33.37 about 64 step mode and Table 33.38 about 256 step mode. |
| 7 | MAXRGB | B'0 | R/W | Histogram Source Component Setting 0: The histogram is generated from 3 color components independently. 1: The histogram is generated from the maximum value of input R, G and B data. |
| 6 | OFSB_R | B'0 | R/W | Offset Binary Mode for R/Cr/H Component 0: Straight binary 1: Offset binary In offset binary mode, values are converted to absolute values before they are used to detect the maximum value, minimum value, sum, and black band. Note that values without conversion are always used for histogram creation regardless of this mode setting. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|-------------|---------------|-----|---|
| 5 | OFSB_G | B'0 | R/W | <p>Offset Binary Mode for G/Y/S Component</p> <p>0: Straight binary 1: Offset binary</p> <p>In offset binary mode, values are converted to absolute values before they are used to detect the maximum value, minimum value, sum, and black band. Note that values without conversion are always used for histogram creation regardless of this mode setting.</p> |
| 4 | OFSB_B | B'0 | R/W | <p>Offset Binary Mode for B/Cb/V Component</p> <p>0: Straight binary 1: Offset binary</p> <p>In offset binary mode, values are converted to absolute values before they are used to detect the maximum value, minimum value, sum, and black band. Note that values without conversion are always used for histogram creation regardless of this mode setting.</p> |
| 3, 2 | HRATIO[1:0] | B'00 | R/W | <p>Horizontal Pixel Skipping Mode for Histogram Detection</p> <p>B'00: No skipping for horizontal pixels. B'01: Horizontal 1/2 skipping. One pixel is discarded from every two pixels before a histogram is created. B'10: Horizontal 1/4 skipping. Three pixels are discarded from every four pixels before a histogram is created. B'11: Setting prohibited.</p> <p>The first pixel in the horizontal direction is not discarded but subsequent pixels are discarded (regardless of the histogram detection window). Among the pixels that have not been discarded, only the pixels within the detection window are used to create a histogram.</p> |
| 1, 0 | VRATIO[1:0] | B'00 | R/W | <p>Vertical Pixel Skipping Mode for Histogram Detection</p> <p>B'00: No skipping for vertical pixels. B'01: Vertical 1/2 skipping. One pixel is discarded from every two pixels before a histogram is created. B'10: Vertical 1/4 skipping. Three pixels are discarded from every four pixels before a histogram is created. B'11: Setting prohibited.</p> <p>The first pixel in the vertical direction is not discarded but subsequent pixels are discarded (regardless of the histogram detection window). Among the pixels that have not been discarded, only the pixels within the detection window are used to create a histogram.</p> |

| Mode | 64 step mode & maxRGB disable | | 64 step mode & maxRGB enable | | 256 step mode | |
|------------------|-------------------------------|--------------------|------------------------------|--------------------|---------------|-----------------------------|
| Address (Offset) | Component | Bit[31:0] | Component | Bit[31:0] | Component | Bit[31:0] |
| 4 x 0 | R/Cr/H | VI6_HGO_R_HISTO_0 | N.A | N.A | Y | VI6_HGO_EXT_HIST_DATA (0) |
| 4 x 1 | | VI6_HGO_R_HISTO_1 | | | or | VI6_HGO_EXT_HIST_DATA (1) |
| ~ | | ~ | | | max | ~ |
| 4 x 62 | | VI6_HGO_R_HISTO_62 | | | (R, G, B) | VI6_HGO_EXT_HIST_DATA (62) |
| 4 x 63 | | VI6_HGO_R_HISTO_63 | | | | VI6_HGO_EXT_HIST_DATA (63) |
| 4 x 64 | G/Y/S | VI6_HGO_G_HISTO_0 | max | VI6_HGO_G_HISTO_0 | | VI6_HGO_EXT_HIST_DATA (64) |
| 4 x 65 | | VI6_HGO_G_HISTO_1 | (R, G, B) | VI6_HGO_G_HISTO_1 | | VI6_HGO_EXT_HIST_DATA (65) |
| ~ | | ~ | | ~ | | ~ |
| 4 x 126 | | VI6_HGO_G_HISTO_62 | | VI6_HGO_G_HISTO_62 | | VI6_HGO_EXT_HIST_DATA (126) |
| 4 x 127 | | VI6_HGO_G_HISTO_63 | | VI6_HGO_G_HISTO_63 | | VI6_HGO_EXT_HIST_DATA (127) |
| 4 x 128 | B/Cb/V | VI6_HGO_B_HISTO_0 | N.A | N.A | | VI6_HGO_EXT_HIST_DATA (128) |
| 4 x 129 | | VI6_HGO_B_HISTO_1 | | | | VI6_HGO_EXT_HIST_DATA (129) |
| ~ | | ~ | | | | ~ |
| 4 x 190 | | VI6_HGO_B_HISTO_62 | | | | VI6_HGO_EXT_HIST_DATA (190) |
| 4 x 191 | | VI6_HGO_B_HISTO_63 | | | | VI6_HGO_EXT_HIST_DATA (191) |
| 4 x 192 | N.A | N.A | N.A | N.A | | VI6_HGO_EXT_HIST_DATA (192) |
| 4 x 193 | | | | | | VI6_HGO_EXT_HIST_DATA (193) |
| ~ | | | | | | ~ |
| 4 x 254 | | | | | | VI6_HGO_EXT_HIST_DATA (254) |
| 4 x 255 | | | | | | VI6_HGO_EXT_HIST_DATA (255) |
| 4 x 256 | R/Cr/H | VI6_HGO_R_MAXMIN | N.A | N.A | N.A | N.A |
| 4 x 257 | | VI6_HGO_R_SUM | | | | |
| 4 x 258 | | VI6_HGO_R_LB_DET | | | | |
| 4 x 259 | N.A | N.A | | | | |
| 4 x 260 | G/Y/S | VI6_HGO_G_MAXMIN | max | VI6_HGO_G_MAXMIN | max | VI6_HGO_G_MAXMIN |
| 4 x 261 | | VI6_HGO_G_SUM | (R, G, B) | VI6_HGO_G_SUM | (R, G, B) | VI6_HGO_G_SUM |
| 4 x 262 | | VI6_HGO_G_LB_DET | | VI6_HGO_G_LB_DET | | VI6_HGO_G_LB_DET |
| 4 x 263 | N.A | N.A | N.A | N.A | N.A | N.A |
| 4 x 264 | B/Cr/B | VI6_HGO_B_MAXMIN | N.A | N.A | N.A | N.A |
| 4 x 265 | | VI6_HGO_B_SUM | | | | |
| 4 x 266 | | VI6_HGO_B_LB_DET | | | | |
| 4 x 267 | N.A | N.A | | | | |
| 4 x 268 | N.A | N.A | N.A | N.A | N.A | N.A |
| 4 x 269 | | | | | | |
| 4 x 270 | | | | | | |
| 4 x 271 | | | | | | |

Figure 33.48 HGO Histogram dump format in case of automatically histogram write

Table 33.36 Data order of HGO histogram in 4 byte

| Memory address | Data order in 4 byte | |
|-------------------|----------------------|------------|
| | H_LWS = 1 | H_LWS = 1 |
| | H_WDS = 1 | H_WDS = 0 |
| | H_BTS = 1 | H_BTS = 0 |
| 4n + 0 | Bit[31:24] | Bit[7:0] |
| 4n + 1 | Bit[23:15] | Bit[15:8] |
| 4n + 2 | Bit[15:8] | Bit[23:15] |
| 4n + 3 | Bit[7:0] | Bit[31:24] |

33.2.16.4 HGO LB Detection Threshold Register (VI6_HGO_LB_TH)

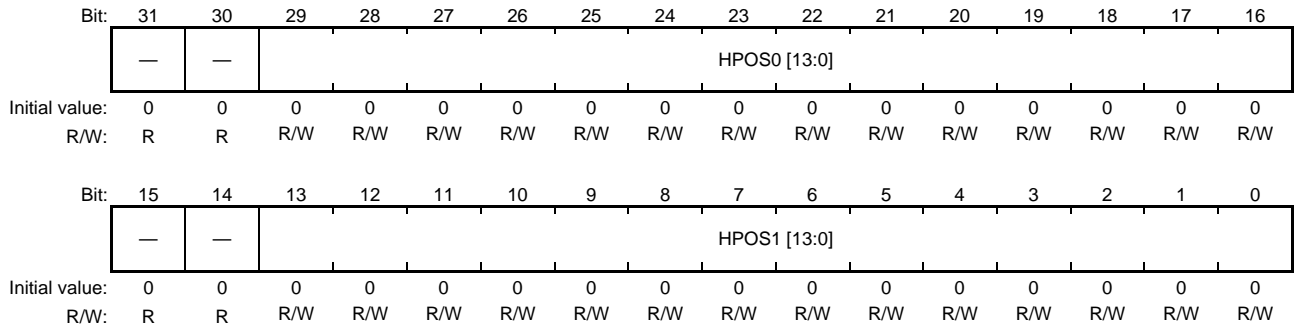
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | BLACK_TH[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | BLACK_TH[7:0] | All 0 | R/W | Threshold for Black Level Determination in Letter Box Detection In letter box detection, when all input pixels have the value equal to or smaller than the value specified in these bits, the image data is determined as black. Note that the letter box detection is not affected by the histogram detection window illustrated in Figure 33.47. A value from 0 to 255 can be specified. |

**33.2.16.5 HGO Horizontal Position Register for LB Detection Zone-n
(VI6_HGO_LBn_H: n = 0, 1, 2, 3)**

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------------|-------------|---------------|-----|--|
| 31, 30, 15, 14 | — | All 0 | R | Reserved |
| 29 to 16 | HPOS0[13:0] | All 0 | R/W | Horizontal Start Position for Letter Box Detection Zone-n When all pixels on a line (shown in Figure 33.49) in the input image have the value equal to or smaller than the threshold value, the HGO determines that there is a letter box. In these bits, specify the value of hpos0 of the detection zone-n (Figure 33.47) in pixel units. A value from 0 to 8189 can be specified. $hpos0 \leq hpos1$ should be satisfied. These bits are valid for Zone-0 and 1. For Zone-2 and 3, these bits are reserved. |
| 13 to 0 | HPOS1[13:0] | All 0 | R/W | Horizontal End Position for Letter Box Detection Zone-n When all pixels on a line (shown in Figure 33.49) in the input image have the value equal to or smaller than the threshold value, the HGO determines that there is a letter box. In these bits, specify the value of hpos1 of the detection zone-n (Figure 33.49) in pixel units. A value from 0 to 8189 can be specified. $hpos0 \leq hpos1$ should be satisfied. |

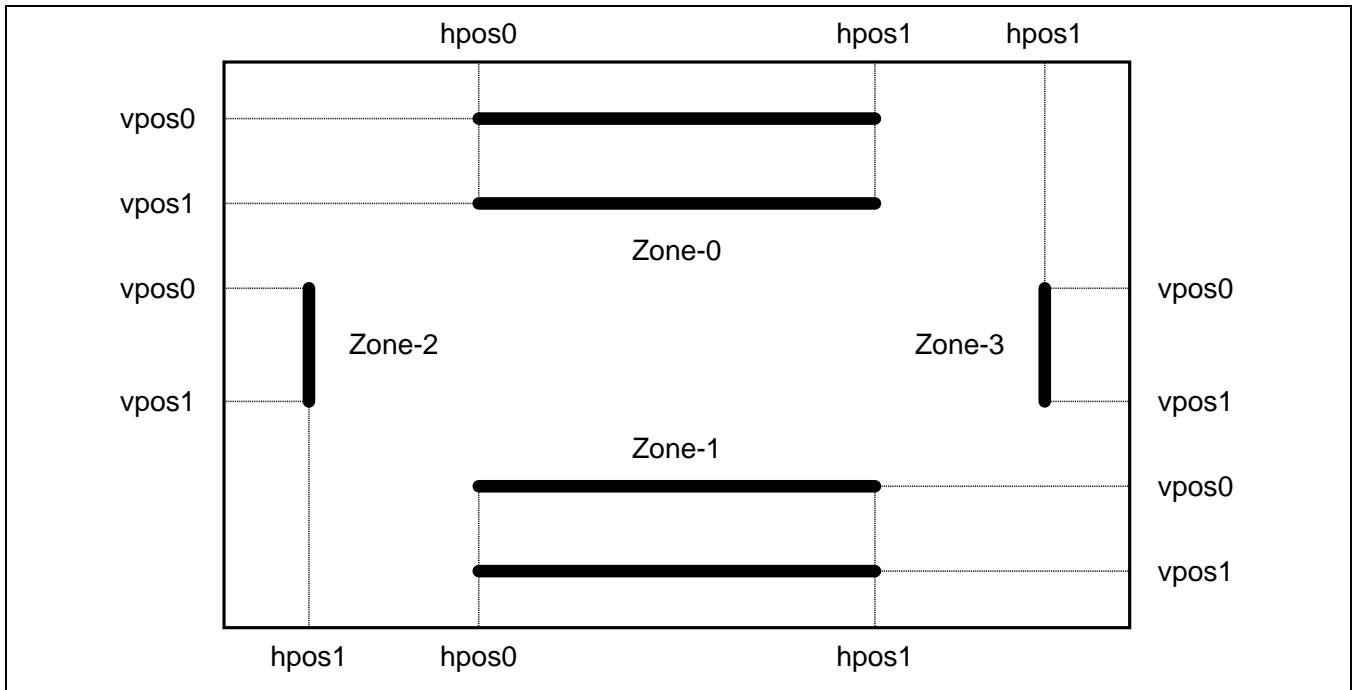
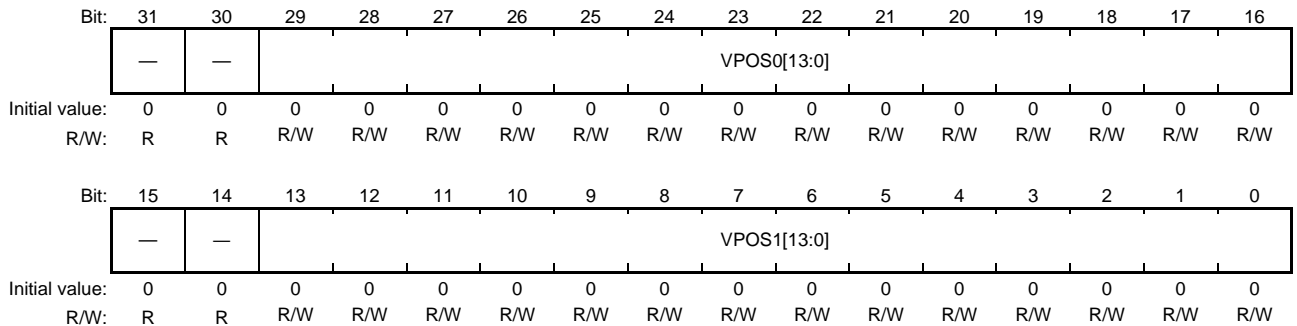


Figure 33.49 Letter Box Detection Position Settings for HGO

33.2.16.6 HGO Vertical Position Register for LB Detection Zone-n (VI6_HGO_LBn_V: n = 0, 1, 2, 3)

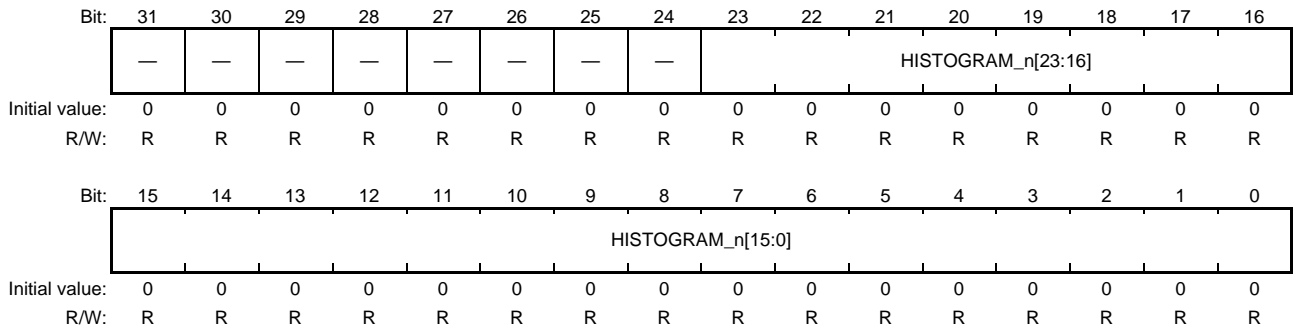
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------------|-------------|---------------|-----|--|
| 31, 30, 15, 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 16 | VPOS0[13:0] | All 0 | R/W | Vertical Start Position for Letter Box Detection Zone-n When all pixels on a line (shown in Figure 33.49) in the input image have the value equal to or smaller than the threshold value, the HGO determines that there is a letter box In these bits, specify the value of vpos0 of the detection zone-n (Figure 33.49) in pixel units. A value from 0 to 8189 can be specified. $vpos0 \leq vpos1$ should be satisfied. |
| 13 to 0 | VPOS1[13:0] | All 0 | R/W | Vertical End Position for Letter Box Detection Zone-n When all pixels on a line (shown in Figure 33.49) in the input image have the value equal to or smaller than the threshold value, the HGO determines that there is a letter box. In these bits, specify the value of vpos1 of the detection zone-n (Figure 33.49) in pixel units. A value from 0 to 8189 can be specified. $vpos0 \leq vpos1$ should be satisfied. |

33.2.16.7 HGO Component-m Histogram Register (VI6_HGO_m_HISTO_n: n = 0 to 63: m = R, G, B)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------|---------------|-----|---|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | HISTOGRAM_n [23:0] | All 0 | R | Frequency of Component-m in the value range-n From these bits, the count of the pixels whose component-m value (val) satisfies the following condition within the histogram detection window (Figure 33.47) is read (after pixel skipping is applied when skipping mode is selected). $4*n \leq val < 4*(n + 1)$ Counting starts after the HGO is activated. To read the histogram, read these bits after the HGO has completed processing of one screen of data and before the HGO is activated again. These bits are available when VI6_HGO_MODE.step is 0 (64 step mode). When VI6_HGO_MODE.step is 1 (256 step mode), read Y/MaxRGB histogram from the VI6_HGO_EXT_HIST_DATA after setting histogram address in the VI6_HGO_EXT_HIST_ADDR. |

The component-m in a color space is determined by the following table in case of VI6_HGO_MODE.step = 0 (64 step mode)

Table 33.37 The component of index-m for each color space in 64 step mode.

| Index-m | Color Space | | | |
|---------|-----------------------------|-----------------------------|---------------------------------|---------------------------------|
| | RGB | | YCbCr | HSV |
| | VI6_HGO_MODE. MAXRGB = 0 | VI6_HGO_MODE. MAXRGB = 1 | VI6_HGO_MODE MAXRGB = 0 (*1) | VI6_HGO_MODE. MAXRGB = 0(*1) |
| R | R | n/a (*2) | Cr | H |
| G | G | max(R, G, B) (*3) | Y | S |
| B | B | n/a (*2) | Cb | V |

Notes: 1. When color space of input data is in YCbCr or HSV, set VI6_HGO_MODE.MAXRGB = 0
 2. When VI6_HGO_MODE.MAXRGB = 1, the histogram of index-R / index-B are not ensured.
 3. max(R, G, B) indicates maximum value of input R, G and B data.

33.2.16.8 HGO Component-m Min/Max Value Register (VI6_HGO_m_MAXMIN: m = R, G, B)

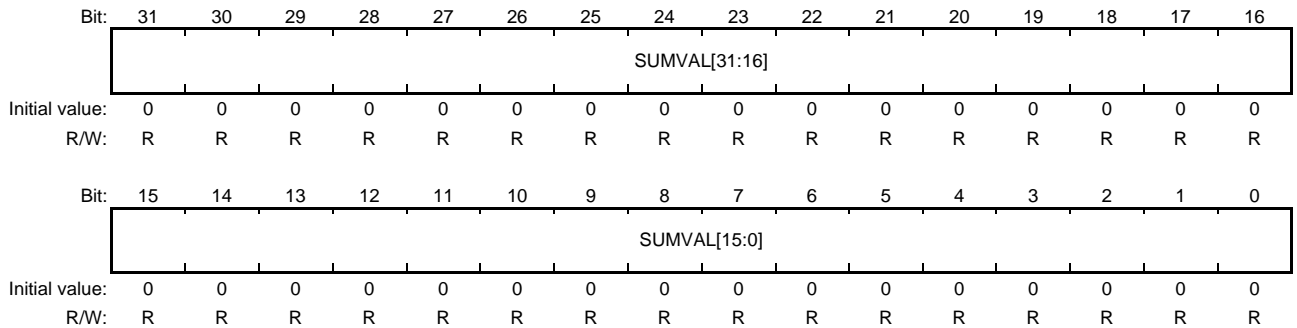
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | MAXVAL[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | MINVAL[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|-------------------|-------------|---------------|-----|---|
| 31 to 24, 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | MAXVAL[7:0] | All 0 | R | Maximum Value of Component-m From these bits, the maximum value of the component-m of the pixels within the histogram detection window (Figure 33.47) is read (after pixel skipping is applied when skipping mode is selected). Maximum value detection starts after the HGO is activated. To read the maximum value, read these bits after the HGO has completed processing of one screen of data and before the HGO is activated again. |
| 7 to 0 | MINVAL[7:0] | All 0 | R | Minimum Value of Component-m From these bits, the minimum value of the component-m of the pixels within the histogram detection window (Figure 33.47) is read (after pixel skipping is applied when skipping mode is selected). Minimum value detection starts after the HGO is activated. To read the minimum value, read these bits after the HGO has completed processing of one screen of data and before the HGO is activated again. |

33.2.16.9 HGO Component-m Sum Register (VI6_HGO_m_SUM: m = R, G, B)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|---|
| 31 to 0 | SUMVAL[31:0] | All 0 | R | <p>Sum of Component-m</p> <p>From these bits, the sum of the component-m of the pixels within the histogram detection window (Figure 33.47) is read (after pixel skipping is applied when skipping mode is selected).</p> <p>Accumulation starts after the HGO is activated. To read the sum, read these bits after the HGO has completed processing of one screen of data and before the HGO is activated again.</p> |

33.2.16.10 HGO Component-m LB Detection Result Register (VI6_HGO_m_LB_DET: m = R, G, B)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-------------|-------------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | LTRB OX1 | LTRB OX2 | SIDE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | LTRBOX1 | B'0 | R | Letter Box Detection Result #1 of Zone-0/1 for Component-m This bit is set to 1 when none of the component-m of the pixels on the lines of zone-0/vpos0 and zone-1/vpos1 (two lines in total) in Figure 33.49 exceeds the threshold (BLACK_TH). If any pixel is greater than the threshold, this bit is set to 0. Read this bit after the HGO has completed processing of one screen of data and before the HGO is activated again. The value read during processing is not guaranteed to be correct. |
| 1 | LTRBOX2 | B'0 | R | Letter Box Detection Result #2 of Zone-0/1 for Component-m This bit is set to 1 when none of the component-m of the pixels on the four lines of zone-0 and 1 in Figure 33.49 exceeds the threshold (BLACK_TH). If any pixel is greater than the threshold, this bit is set to 0. Read this bit after the HGO has completed processing of one screen of data and before the HGO is activated again. The value read during processing is not guaranteed to be correct. |
| 0 | SIDE | B'0 | R | Letter Box Detection Result of Zone-2/3 for Component-m This bit is set to 1 when none of the component-m of the pixels on the two lines of zone-2 and 3 in Figure 33.49 exceeds the threshold (BLACK_TH). If any pixel is greater than the threshold, this bit is set to 0. Read this bit after the HGO has completed processing of one screen of data and before the HGO is activated again. The value read during processing is not guaranteed to be correct. |

33.2.16.11 HGO Y/MaxRGB Extended Histogram Address Register (VI6_HGO_EXT_HIST_ADDR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|--------------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | EXT_HIST_ADDR[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | EXT_HIST_ADDR [7:0] | All 0 | R/W | Histogram address in case of 256 step histogram mode. Set value-m when user wants to read frequency of Y or MaxRGB pixel value-m. After setting VI6_HGO_EXT_HIST_ADDR, read VI6_HGO_EXT_HIST_DATA to read frequency of Y or MaxRGB pixel value-m. This is in-direct addressing way. This in-direct addressing way is available in case of VI6_HGO_MODE.STEP=1(256 step histogram mode). |

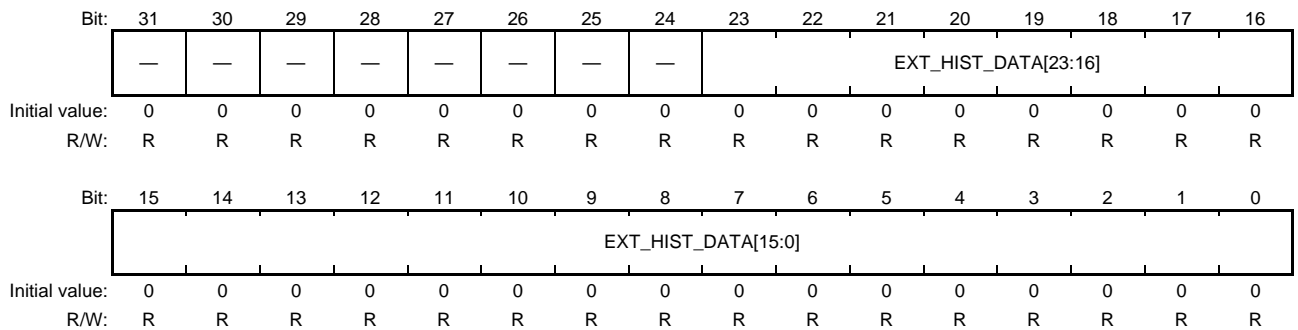
Table 33.38 Restriction in case of 256 step mode (VI6_HGO_MODE.STEP = 1)

| Color Space | | | |
|-------------------------|-------------------------|------------------------------|-----------------------------|
| RGB | YCbCr | | HSV |
| VI6_HGO_MODE.MAXRGB = 0 | VI6_HGO_MODE.MAXRGB = 1 | VI6_HGO_MODE.MAXRGB = 0 (*1) | VI6_HGO_MODE.MAXRGB = 0(*1) |
| N/A | max(R, G, B) (*2) | Y | N/A |

Notes: 1. When color space of input data is in YCbCr or HSV, set VI6_HGO_MODE.MAXRGB = 0
 2. max(R, G, B) indicates maximum value of input R, G and B data.

33.2.16.12 HGO Y/MaxRGB Extended Histogram Data Register (VI6_HGO_EXT_HIST_DATA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------|---------------|-----|---|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | EXT_HIST_DATA [23:0] | All 0 | R | Histogram data in case of 256 step histogram mode. See VI6_HGO_EXT_HIST_ADDR register's explanation. This in-direct addressing way is available in case of VI6_HGO_MODE.STEP = 1 (256 step histogram mode). |

33.2.16.13 HGO Write Buffer Side (VI6_HGO_WBUFS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | — | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | WRB UFS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | WRBUFS | B'0 | R/W | HGO-Histogram buffer side (Writing side by H/W) Setting value is reflected to internal register at the timing of frame start. Set this bit before setting VI6_DL_HDR_ADDRn (reservation of updating two plane registers). 0: 1D-Histogram side A is written by next VSP-operation. 1: 1D-Histogram side B is written by next VSP-operation. |

33.2.16.14 HGO Read Buffer Side (VI6_HGO_RBUFS)

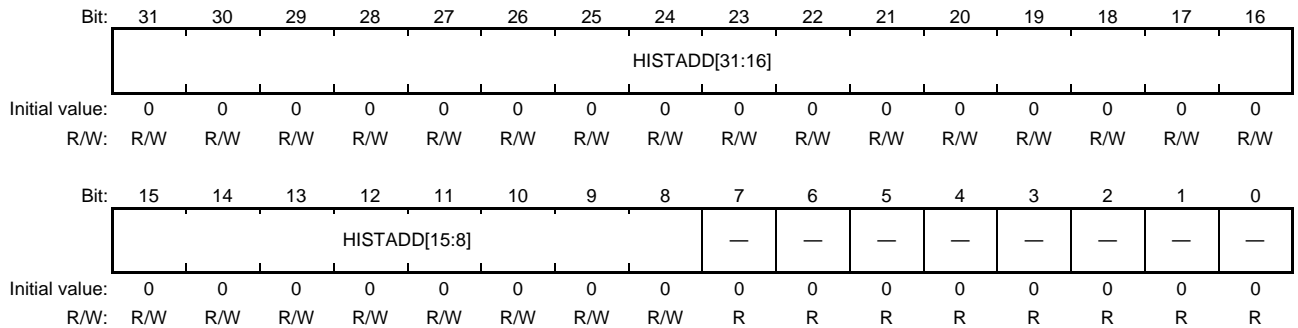
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | — | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | RDB UFS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | RDBUFS | B'0 | R/W | HGO-Histogram buffer side (Read side by S/W) Set this bit to specify reading side of 1D-Histogram before reading 1D-histogram registers. 0: 1D-Histogram side A can be read by S/W. 1: 1D-Histogram side B can be read by S/W. |

33.2.16.15 HGO Histogram Storing Address Register (VI6_HGO_HISTADD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 8 | HISTADD[31:8] | All 0 | R/W | HGO histogram storing address These bits specify the start address of storing the HGO histogram plane to be written in case of enabling histogram auto write (VI6_HGO_MODE.AUTOW = 1). Start address of HGO histogram can be set in 256-byte units. And HISTADD [31:8] × 256 is recognized as start address to write HGO histogram. |
| 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

33.2.16.16 HGO Histogram Swapping Register (VI6_HGO_HSWAP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-----------|-----------|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | H_L WS | H_W DS | H_B TS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | H_LWS | B'0 | R/W | HGO histogram Swapping in Longword Units 0: Data swapping in longword (32-bit) units is disabled 1: Data swapping in longword (32-bit) units is enabled |
| 1 | H_WDS | B'0 | R/W | HGO histogram Swapping in Word Units 0: Data swapping in word (16-bit) units is disabled 1: Data swapping in word (16-bit) units is enabled |
| 0 | H_BTS | B'0 | R/W | HGO histogram Swapping in Byte Units 0: Data swapping in byte (8-bit) units is disabled 1: Data swapping in byte (8-bit) units is enabled |

See Figure 33.48 and Table 33.36 for setting HGO histogram swap.

33.2.16.17 HGO Parameter Register Reset (VI6_HGO_REGRST)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|------------|----|----|----|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | RCP ART | — | — | — | RCL EA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | W | R | R | R | W |

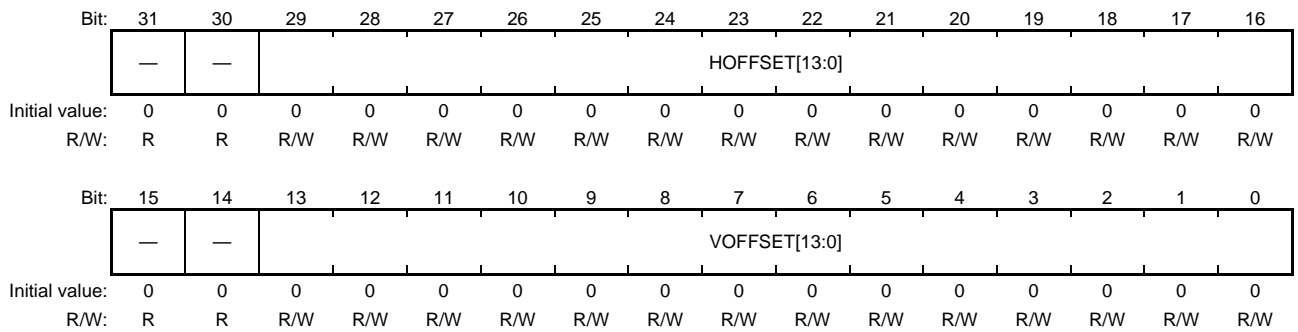
| Bit | Bit Name | Initial Value | R/W | Description |
|-----------------|----------|---------------|-----|--|
| 31 to 5, 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | RCPART | B'0 | W | Status Reset Writing 1 to this bit resets counter bits in HGO. |
| 0 | RCLEA | B'0 | W | Register Reset Writing 1 to this bit resets all read-only registers (33.2.16.7 to 33.2.16.10) to their initial values. This register is write-only and is always read as 0. Note that R/W registers (33.2.16.1 to 33.2.16.6) are not affected by write access to this register. |

When image partition is used, refer to section 33.3.7.7.

33.2.17 HGT Control Registers

33.2.17.1 HGT Detection Window Offset Register (VI6_HGT_OFFSET)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------------|----------------|---------------|-----|--|
| 31, 30, 15, 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 16 | HOFFSET [13:0] | All 0 | R/W | Horizontal Offset of Histogram Detection Window The HGT creates a histogram for the detection window in the input image (Figure 33.50). In these bits, specify the value of horizontal offset (hoffset shown Figure 33.50) in pixel units. A value from 0 to 8189 can be specified. The value of HOFFSET shall be smaller than that of the input image size of HGT. |
| 13 to 0 | VOFFSET [13:0] | All 0 | R/W | Vertical Offset of Histogram Detection Window The HGT creates a histogram for the detection window in the input image (Figure 33.50). In these bits, specify the value of vertical offset (voffset shown in Figure 33.50) in pixel units. A value from 0 to 8189 can be specified. The value of VOFFSET shall be smaller than that of the input image size of HGT. |

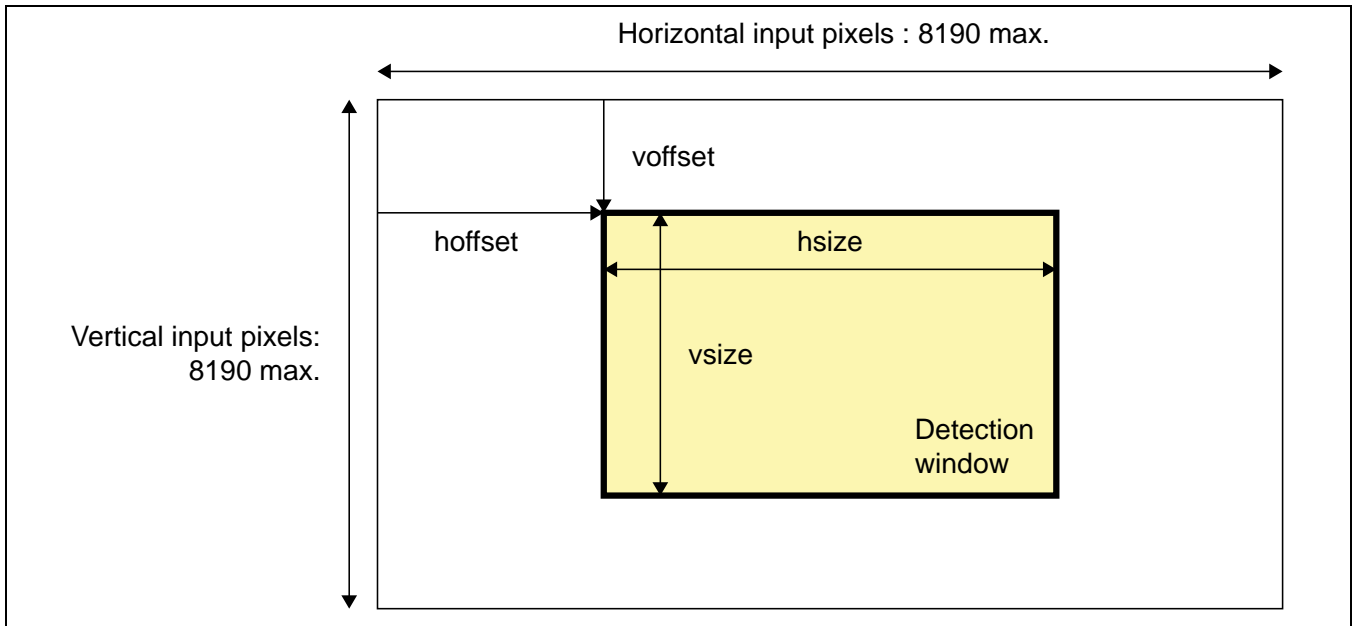
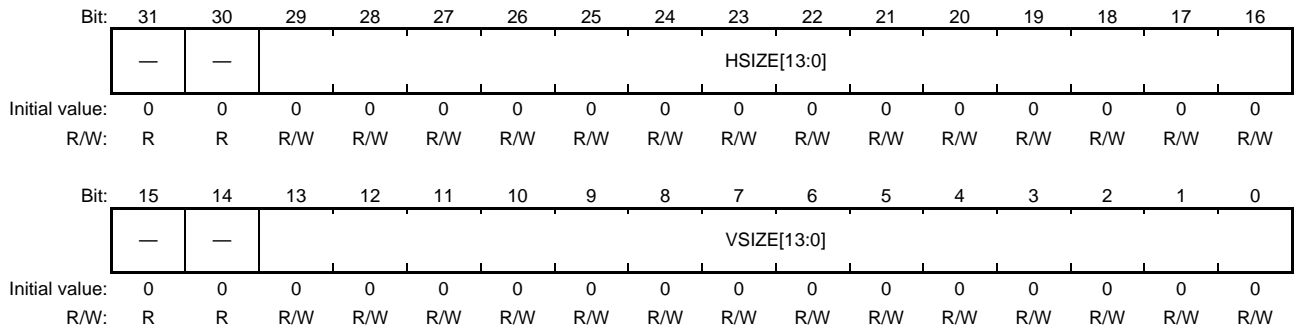


Figure 33.50 Histogram Detection Window of HGT

33.2.17.2 HGT Detection Window Size Register (VI6_HGT_SIZE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------------|-------------|---------------|-----|---|
| 31, 30, 15, 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 16 | HSIZE[13:0] | All 0 | R/W | Horizontal Size of Histogram Detection Window The HGT creates a histogram for the detection window in the input image (Figure 33.50). In these bits, specify the value of horizontal offset (hsize shown in Figure 33.50) in pixel units. A value from 1 to 8190 can be specified. |
| 13 to 0 | VSIZE[13:0] | All 0 | R/W | Vertical Size of Histogram Detection Window The HGT creates a histogram for the detection window in the input image (Figure 33.50). In these bits, specify the value of vertical offset (vsize shown in Figure 33.50) in pixel units. A value from 1 to 8190 can be specified. |

33.2.17.3 HGT Mode Register (VI6_HGT_MODE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----------|----|----|----|----|----|----|----|----|-------------|-------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | AUT OW | — | — | — | — | — | — | — | — | HRATIO[1:0] | VRATIO[1:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------------------|-------------|---------------|-----|---|
| 31 to13, 11 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | AUTOW | B'0 | R/W | HGT automatically histogram write. [RZ/G2H/RZ/G2N, RZ/G2E] 0: Histogram is not dumped to external memory automatically. S/W needs to dump histogram from histogram registers to external memory. 1: Histogram is dumped to SDRAM automatically. Dump format is shown in Figure 33.51. HGT Histogram Swapping Register in section 33.2.17.16 specifies data order in 4byte unit as Table 33.39. |
| 3, 2 | HRATIO[1:0] | B'00 | R/W | Horizontal Pixel Skipping Mode for Histogram Detection B'00: No skipping for horizontal pixels. B'01: Horizontal 1/2 skipping. One pixel is discarded from every two pixels before a histogram is created. B'10: Horizontal 1/4 skipping. Three pixels are discarded from every four pixels before a histogram is created. B'11: Setting prohibited. The first pixel in the horizontal direction is not discarded but subsequent pixels are discarded (regardless of the histogram detection window). Among the pixels that have not been discarded, only the pixels within the detection window are used to create a histogram. |
| 1, 0 | VRATIO[1:0] | B'00 | R/W | Vertical Pixel Skipping Mode for Histogram Detection B'00: No skipping for vertical pixels. B'01: Vertical 1/2 skipping. One pixel is discarded from every two pixels before a histogram is created. B'10: Vertical 1/4 skipping. Three pixels are discarded from every four pixels before a histogram is created. B'11: Setting prohibited. The first pixel in the vertical direction is not discarded but subsequent pixels are discarded (regardless of the histogram detection window). Among the pixels that have not been discarded, only the pixels within the detection window are used to create a histogram. |

| Address (Offset) | Hue area | Bit[31:0] |
|---------------------|-----------|--------------------|
| 4 x 0 | HueArea 0 | VI6_HGT_HISTO_0_0 |
| 4 x 1 | | VI6_HGT_HISTO_0_1 |
| ~ | | ~ |
| 4 x 30 | | VI6_HGT_HISTO_0_30 |
| 4 x 31 | | VI6_HGT_HISTO_0_31 |
| 4 x 32 | HueArea 1 | VI6_HGT_HISTO_1_0 |
| 4 x 33 | | VI6_HGT_HISTO_1_1 |
| ~ | | ~ |
| 4 x 62 | | VI6_HGT_HISTO_1_30 |
| 4 x 63 | | VI6_HGT_HISTO_1_31 |
| 4 x 64 | HueArea 2 | VI6_HGT_HISTO_2_0 |
| 4 x 65 | | VI6_HGT_HISTO_2_1 |
| ~ | | ~ |
| 4 x 94 | | VI6_HGT_HISTO_2_30 |
| 4 x 95 | | VI6_HGT_HISTO_2_31 |
| 4 x 96 | HueArea 3 | VI6_HGT_HISTO_3_0 |
| 4 x 97 | | VI6_HGT_HISTO_3_1 |
| ~ | | ~ |
| 4 x 126 | | VI6_HGT_HISTO_3_30 |
| 4 x 127 | | VI6_HGT_HISTO_3_31 |
| 4 x 128 | HueArea 4 | VI6_HGT_HISTO_4_0 |
| 4 x 129 | | VI6_HGT_HISTO_4_1 |
| ~ | | ~ |
| 4 x 158 | | VI6_HGT_HISTO_4_30 |
| 4 x 159 | | VI6_HGT_HISTO_4_31 |
| 4 x 160 | HueArea 5 | VI6_HGT_HISTO_5_0 |
| 4 x 161 | | VI6_HGT_HISTO_5_1 |
| ~ | | ~ |
| 4 x 190 | | VI6_HGT_HISTO_5_30 |
| 4 x 191 | | VI6_HGT_HISTO_5_31 |
| 4 x 192 | S | VI6_HGT_MAXMIN |
| 4 x 193 | S | VI6_HGT_SUM |
| 4 x 194 | V | VI6_HGT_LB_DET |
| 4 x 195 | N.A | N.A |
| ~ | | |
| 4 x 199 | | |

Figure 33.51 HGT Histogram dump format in case of automatically histogram write

Table 33.39 Data order of HGT histogram in 4 byte

| Memory address | Data order in 4 byte | |
|----------------|----------------------|------------|
| | H_LWS = 1 | H_LWS = 1 |
| | H_WDS = 1 | H_WDS = 0 |
| | H_BTS = 1 | H_BTS = 0 |
| 4n + 0 | Bit[31:24] | Bit[7:0] |
| 4n + 1 | Bit[23:15] | Bit[15:8] |
| 4n + 2 | Bit[15:8] | Bit[23:15] |
| 4n + 3 | Bit[7:0] | Bit[31:24] |

33.2.17.4 HGT Hue Area Register (VI6_HGT_HUE_AREA_n: n = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | HUE_LOWER_n[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | HUE_UPPER_n[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------------------|-------------------|---------------|-----|---|
| 31 to 24, 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | HUE_LOWER_n [7:0] | All 0 | R/W | Lower Boundary Value for Hue Area - n The HGT creates a two-dimensional histogram of H (hue) and S (saturation) components. Division of hue areas and overlapping of adjacent hue areas can be specified through these bits. In these bits, specify the value of n L (hue_lower n) shown in Figure 33.52. A value from 0 to 255 can be specified. The specified value should satisfy the restrictions shown in Figure 33.52. |
| 7 to 0 | HUE_UPPER_n [7:0] | All 0 | R/W | Upper Boundary Value for Hue Area - n The HGT creates a two-dimensional histogram of H (hue) and S (saturation) components. Division of hue areas and overlapping of adjacent hue areas can be specified through these bits. In these bits, specify the value of n U (hue_upper n) shown in Figure 33.52. A value from 0 to 255 can be specified. The specified value should satisfy the restrictions shown in Figure 33.52. |

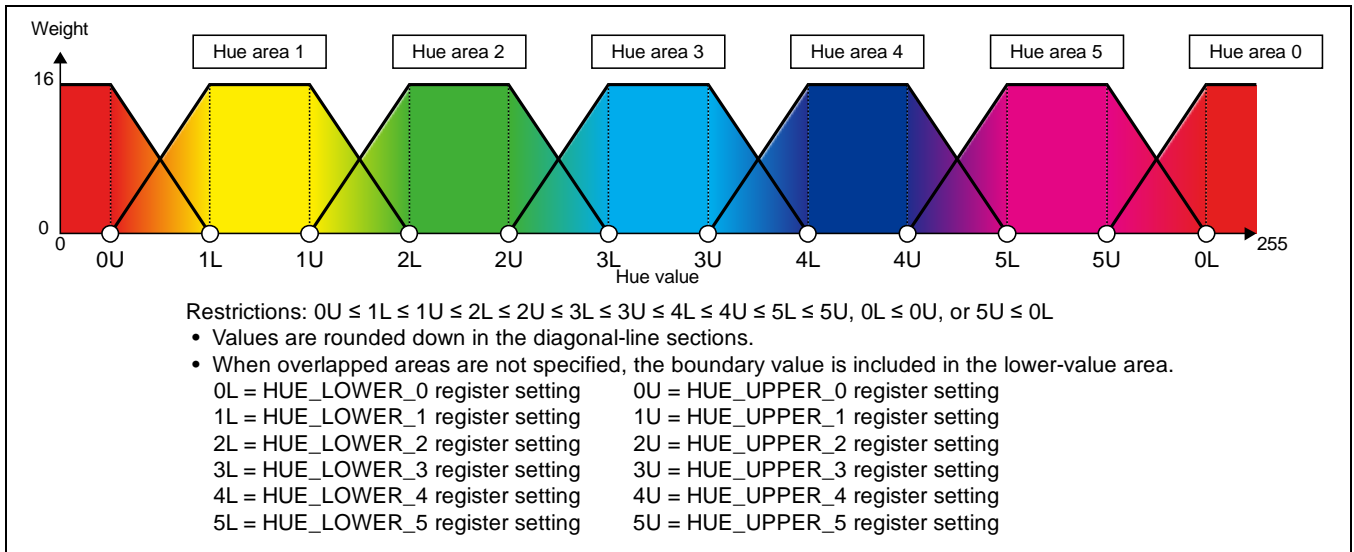


Figure 33.52 Weighting Histogram Using Hue

33.2.17.5 HGT LB Detection Threshold Register (VI6_HGT_LB_TH)

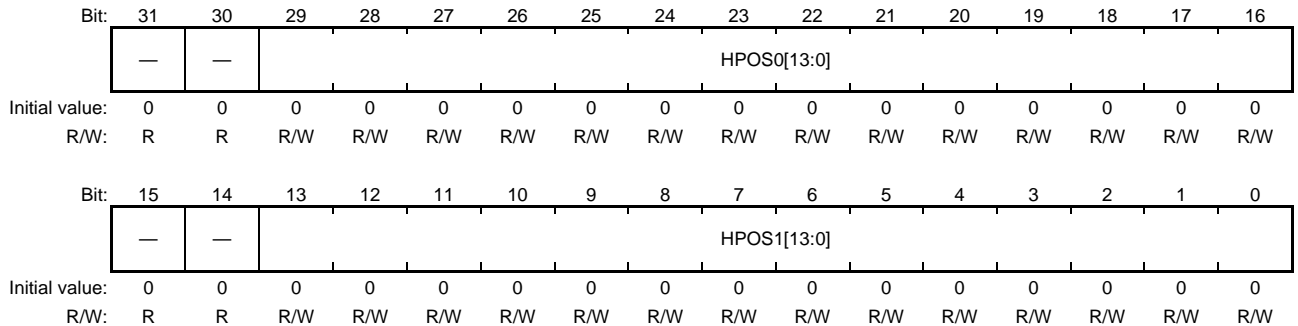
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | BLACK_TH[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | BLACK_TH[7:0] | All 0 | R/W | Threshold for Black Level Determination in Letter Box Detection In letter box detection, when all pixels of the input image data have the value equal to or smaller than the value specified in these bits, the data is determined as black. Note that the letter box detection is not affected by the histogram detection window illustrated in Figure 33.50. A value from 0 to 255 can be specified. |

**33.2.17.6 HGT Horizontal Position Register for LB Detection Zone-n
(VI6_HGT_LBn_H : n = 0, 1, 2, 3)**

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------------|-------------|---------------|-----|--|
| 31, 30, 15, 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 16 | HPOS0[13:0] | All 0 | R/W | Horizontal Start Position for Letter Box Detection Zone-n When all pixels on a line (shown in Figure 33.53) in the input image have the value equal to or smaller than the threshold value, the HGT determines that there is a letter box. In these bits, specify the value of hpos0 of the detection zone-n (Figure 33.53) in pixel units. A value from 0 to 8189 can be specified. $hpos0 \leq hpos1$ should be satisfied. These bits are valid for Zone-0 and 1. For Zone-2 and 3, these bits are reserved. |
| 13 to 0 | HPOS1[13:0] | All 0 | R/W | Horizontal End Position for Letter Box Detection Zone-n When all pixels on a line (shown in Figure 33.53) in the input image have the value equal to or smaller than the threshold value, the HGT determines that there is a letter box. In these bits, specify the value of hpos1 of the detection zone-n (Figure 33.53) in pixel units. A value from 0 to 8189 can be specified. $hpos0 \leq hpos1$ should be satisfied. |

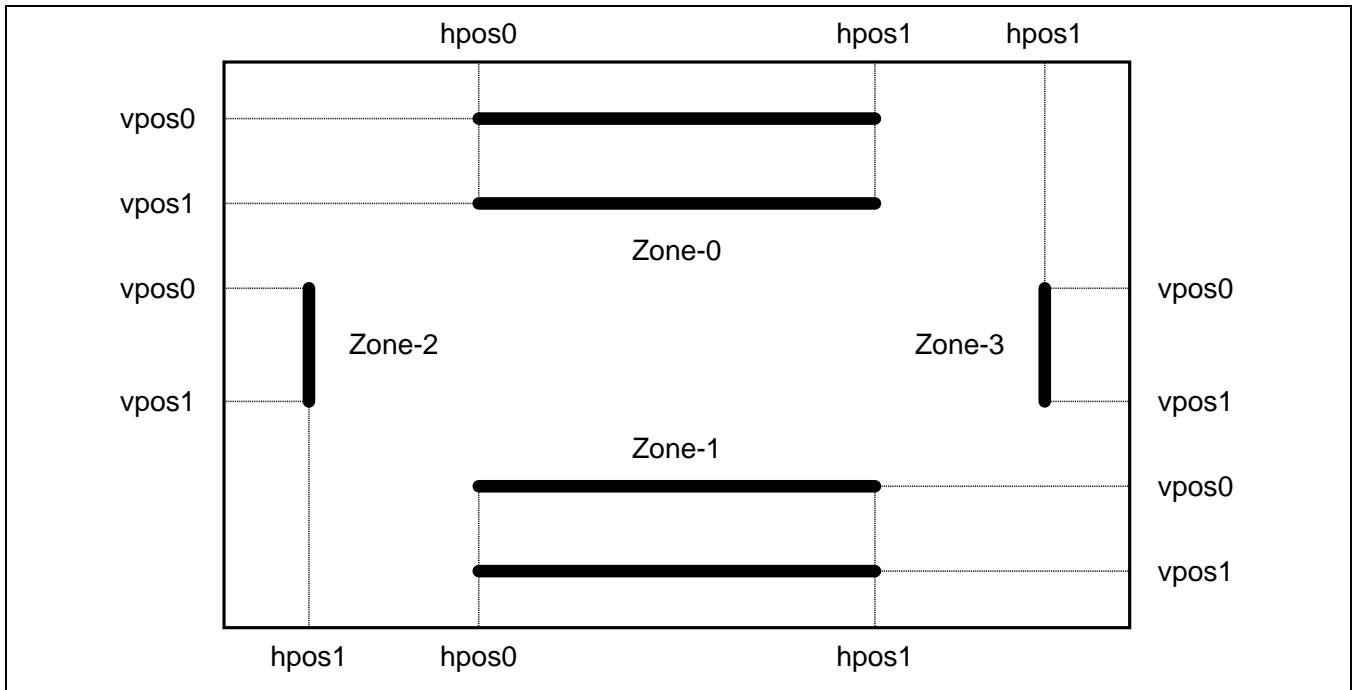
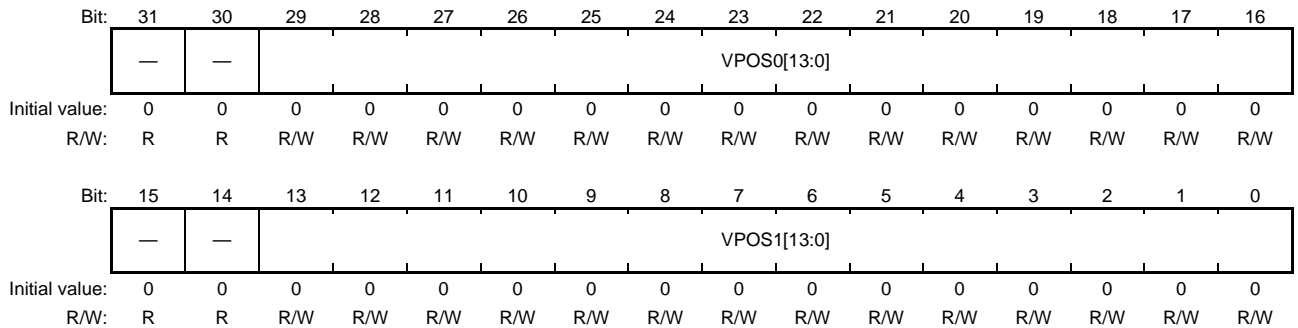


Figure 33.53 Letter Box Detection Position Settings for HGT

33.2.17.7 HGT Vertical Position Register for LB Detection Zone-n (VI6_HGT_LBn_V: n = 0, 1, 2, 3)

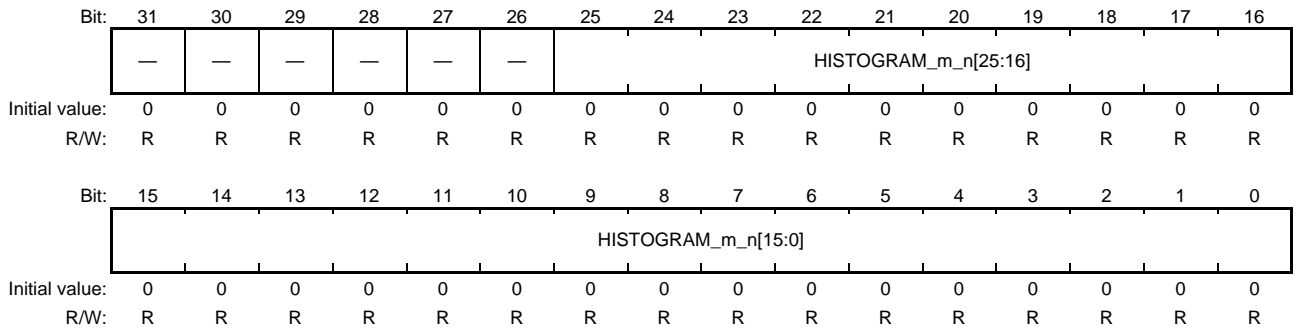
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------------|-------------|---------------|-----|---|
| 31, 30, 15, 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 16 | VPOS0[13:0] | All 0 | R/W | Vertical Start Position for Letter Box Detection Zone-n When all pixels on a line (shown in Figure 33.53) in the input image have the value equal to or smaller than the threshold value, the HGT determines that there is a letter box. In these bits, specify the value of vpos0 of the detection zone-n (Figure 33.53) in pixel units. A value from 0 to 8189 can be specified. $vpos0 \leq vpos1$ should be satisfied. |
| 13 to 0 | VPOS1[13:0] | All 0 | R/W | Vertical Position for Letter Box Detection Zone-n When all pixels on a line (shown in Figure 33.53) in the input image have the value equal to or smaller than the threshold value, the HGT determines that there is a letter box. In these bits, specify the value of vpos1 of the detection zone-n (Figure 33.53) in pixel units. A value from 0 to 8189 can be specified. $vpos0 \leq vpos1$ should be satisfied. |

33.2.17.8 HGT Histogram Register (VI6_HGT_HISTO_m_n: m = 0 to 5, n = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|---|
| 31 to 26 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25 to 0 | HISTOGRAM_m_n[25:0] | All 0 | R | Weighted Frequency of Hue Area-m and Saturation Area-n From these bits, the count of weighting for the pixels whose H component value is in hue area m (Figure 33.52) and whose S component value (val) satisfies the following condition within the histogram detection window (Figure 33.50) is read (after pixel skipping is applied when skipping mode is selected). $8 * n \leq val < 8 * (n + 1)$ The weight is determined by the H component value as shown in Figure 33.52 and the maximum weight is 16. Counting starts after the HGT is activated. To read the histogram, read these bits after the HGT has completed processing of one screen of data and before the HGT is activated again. |

33.2.17.9 HGT Max/Min Value Register (VI6_HGT_MAXMIN)

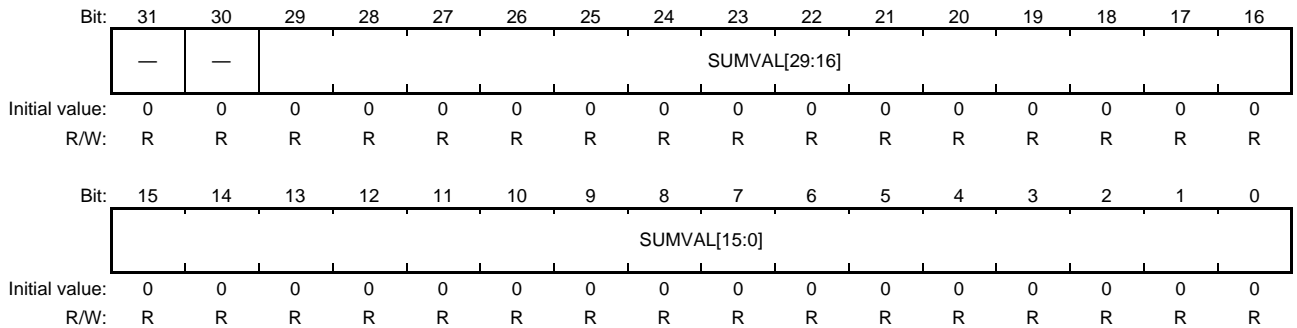
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | MAXVAL[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | MINVAL[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|-------------------|-------------|---------------|-----|---|
| 31 to 24, 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | MAXVAL[7:0] | All 0 | R | Maximum Value of S Components From these bits, the maximum value of the S components of the pixels within the histogram detection window (Figure 33.50) is read (after pixel skipping is applied when skipping mode is selected). Maximum value detection starts after the HGT is activated. To read the maximum value, read these bits after the HGT has completed processing of one screen of data and before the HGT is activated again. |
| 7 to 0 | MINVAL[7:0] | All 0 | R | Minimum Value of S Components From these bits, the minimum value of the S components of the pixels within the histogram detection window (Figure 33.50) is read (after pixel skipping is applied when skipping mode is selected). Minimum value detection starts after the HGT is activated. To read the minimum value, read these bits after the HGT has completed processing of one screen of data and before the HGT is activated again. |

33.2.17.10 HGT Sum Register (VI6_HGT_SUM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|--|
| 31, 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 0 | SUMVAL[29:0] | All 0 | R | Sum of S Components From these bits, the sum of the S components of the pixels within the histogram detection window (Figure 33.50) is read (after pixel skipping is applied when skipping mode is selected). Accumulation starts after the HGT is activated. To read the sum, read these bits after the HGT has completed processing of one screen of data and before the HGT is activated again. |

33.2.17.11 HGT LB Detection Result Register (VI6_HGT_LB_DET)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----------|----------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | LTR BOX1 | LTR BOX2 | SIDE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | LTRBOX1 | B'0 | R | Letter Box Detection Result #1 of Zone-0/1 for V Component This bit is set to 1 when none of the V components of the pixels on the lines of zone-0/vpos0 and zone-1/vpos1 (two lines in total) in Figure 33.53 exceeds the threshold (BLACK_TH). If any pixel is greater than the threshold, this bit is set to 0. Read this bit after the HGT has completed processing of one screen of data and before the HGT is activated again. The value read during processing is not guaranteed to be correct. |
| 1 | LTRBOX2 | B'0 | R | Letter Box Detection Result #2 of Zone-0/1 for V Component This bit is set to 1 when none of the V components of the pixels on the four lines of zone-0 and 1 in Figure 33.53 exceeds the threshold (BLACK_TH). If any pixel is greater than the threshold, this bit is set to 0. Read this bit after the HGT has completed processing of one screen of data and before the HGT is activated again. The value read during processing is not guaranteed to be correct. |
| 0 | SIDE | B'0 | R | Letter Box Detection Result of Zone-2/3 for V Component This bit is set to 1 when none of the V components of the pixels on the two lines of zone-2 and 3 in Figure 33.53 exceeds the threshold (BLACK_TH). If any pixel is greater than the threshold, this bit is set to 0. Read this bit after the HGT has completed processing of one screen of data and before the HGT is activated again. The value read during processing is not guaranteed to be correct. |

33.2.17.12 HGT Parameter Register Reset (VI6_HGT_REGRST)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|------------|----|----|----|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | RCP ART | — | — | — | RCL EA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | W | R | R | R | W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----------------|----------|---------------|-----|--|
| 31 to 5, 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | RCPART | B'0 | W | Status Reset Writing 1 to this bit resets counter bits in HGT. |
| 0 | RCLEA | B'0 | W | Register Reset Writing 1 to this bit resets all read-only registers (33.2.17.8 to 33.2.17.11) to their initial values. This register is write-only and is always read as 0. Note that R/W registers (33.2.17.1 to 33.2.17.7) are not affected by write access to this register. |

When image partition is used, refer to section 33.3.7.7.

33.2.17.13 HGT Write Buffer Side (VI6_HGT_WBUFS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | — | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | WRB UFS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | WRBUFS | B'0 | R/W | HGT-Histogram buffer side (Writing side by H/W) Setting value is reflected to internal register at the timing of frame start. Set this bit before setting VI6_DL_HDR_ADDRn (reservation of updating two plane registers). 0: 2D-Histogram side A is written by next VSP-operation. 1: 2D-Histogram side B is written by next VSP-operation. |

33.2.17.14 HGT Read Buffer Side (VI6_HGT_RBUFS)

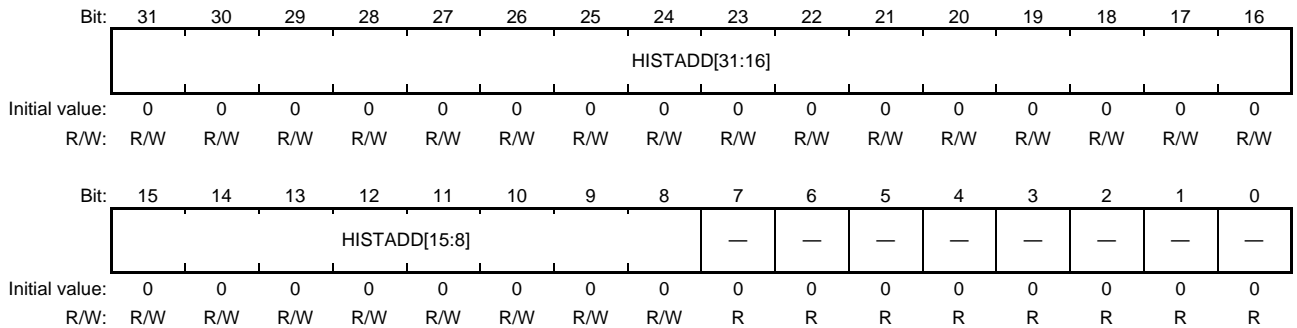
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | — | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | RDB UFS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | RDBUFS | B'0 | R/W | HGT-Histogram buffer side (Read side by S/W) Set this bit to specify reading side of 2D-Histogram before reading 2D-histogram registers. 0: 2D-Histogram side A can be read by S/W. 1: 2D-Histogram side B can be read by S/W. |

33.2.17.15 HGT Histogram Storing Address Register (VI6_HGT_HISTADD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 8 | HISTADD[31:8] | All 0 | R/W | HGT histogram storing address These bits specify the start address of storing the HGT histogram plane to be written in case of enabling histogram auto write (VI6_HGT_MODE.AUTOW = 1). Start address of HGT histogram can be set in 256-byte units. And HISTADD [31:8] × 256 is recognized as start address to write HGT histogram. |
| 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

33.2.17.16 HGT Histogram Swapping Register (VI6_HGT_HSWAP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-----------|-----------|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | H_L WS | H_W DS | H_B TS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | H_LWS | B'0 | R/W | HGT histogram Swapping in Longword Units 0: Data swapping in longword (32-bit) units is disabled 1: Data swapping in longword (32-bit) units is enabled |
| 1 | H_WDS | B'0 | R/W | HGT histogram Swapping in Word Units 0: Data swapping in word (16-bit) units is disabled 1: Data swapping in word (16-bit) units is enabled |
| 0 | H_BTS | B'0 | R/W | HGT histogram Swapping in Byte Units 0: Data swapping in byte (8-bit) units is disabled 1: Data swapping in byte (8-bit) units is enabled |

See Figure 33.51 and Table 33.39 for setting HGT histogram swap.

33.2.18 ILV Control Register

33.2.18.1 ILV Control Register (VI6_ILV_CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|-------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ILV_MODE [1:0] |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ILV_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-------------------|-------------------|---------------|-----|--|
| 31 to 18, 15 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 17, 16 | ILV_MODE [1:0] | B'0 | R/W | Format Setting for 3D-Format Conversion These bits specify the format for 3D-format conversion. B'00: Side by side (output is enlarged twice in horizontal direction) B'01: Line by line (output is enlarged twice in vertical direction) B'10: Pixel by pixel (output is enlarged twice in horizontal direction) B'11: Doubler (output is enlarged twice in horizontal and vertical directions) |
| 0 | ILV_EN | B'0 | R/W | 3D-Format Conversion Enable/Disable Enables or disables 3D-format conversion by the ILV. Before setting this bit to 1, make appropriate connections with the DPR so that valid image data is sent to two input lines of the ILV. When this bit is set to 0, the input to the ILVL passes through the ILV and the output size becomes the same as the input size. 0: 3D-format conversion is disabled. 1: 3D-format conversion is enabled. |

Interleave method of ILVL and ILVR is shown in Figure 33.54. "ILVL" or "L" in the figure is output image from module that target node setting is 38. "ILVR" or "R" in the figure is output image from module that target node setting is 39. See section 33.2.8.1 about target node.

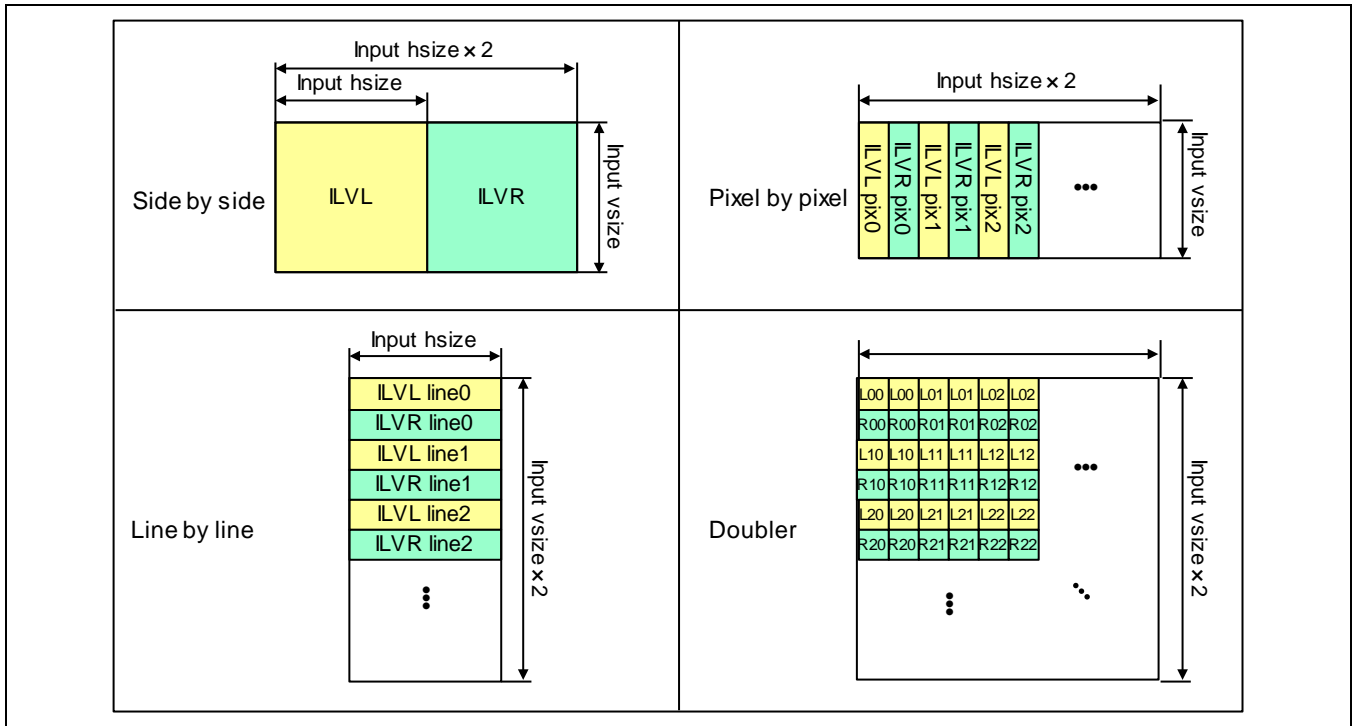


Figure 33.54 Interleave method of ILVL and ILVR

33.2.19 BRS Control Registers

33.2.19.1 BRS Input Control Register (VI6_BRS_INCTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----|----|----|----|----|----|----|----|----|----|----|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | NRM | — | — | — | — | — | — | — | — | — | — | D1 ON | D0 ON |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|-------|------------|-----|-----|-------|------------|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | ODE 1 | DITH1[2:0] | | | ODE 0 | DITH0[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----------------------------|----------|---------------|-----|--|
| 31 to 29, 27 to 18, 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | NRM | B'0 | R/W | Color Data Normalization Enables or disables division by the α value of the color data in BRS blending operation. This is used when converting the RGB color data format to which the α value is multiplied (pre-multiplied color) into the RGB color data format to which the α value is not multiplied (non pre-multiplied color). Do not use this for the YCbCr format. 0: Divider (DIV unit in Figure 33.55) does not divide the color value by α 1: Divider (DIV unit in Figure 33.55) divides the color value by α |
| 17 | D1ON | B'0 | R/W | Ordered dither (mode B) Enable of BRS Input 1 Enables or disables dithering (color reduction) of BRS input 1 (BRSin1 in Figure 33.55). 0: Dithering (mode B) of BRSin1 is disabled 1: Dithering (mode B) of BRSin1 is enabled When ODE1 in this bit is set to 1, set 0 to this bit. |
| 16 | D0ON | B'0 | R/W | Ordered dither (mode B) Enable of BRS Input 0 Enables or disables dithering (color reduction) of BRS input 0 (BRSin0 in Figure 33.55). 0: Dithering (mode B) of BRSin0 is disabled 1: Dithering (mode B) of BRSin0 is enabled When ODE0 in this bit is set to 1, set 0 to this bit. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-------------|---------------|-----|---|
| 7 | ODE1 | B'0 | R/W | <p>Ordered Dither (mode A) of CH1 Input to BRS Enable/Disable</p> <p>0: Ordered dither (mode A) of BRSin1 is disabled.</p> <p>1: Ordered dither (mode A) of BRSin1 is enabled.</p> <p>Ordered dither is available only for 18bpp. So, when ODE1 bit is equal to 1, set DITH1 = 1.</p> <p>When ODE1 bit is equal to 0, BRSin1 dither method is specified by D1ON in the register</p> <p>Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp.</p> |
| 6 to 4 | DITH1 [2:0] | B'000 | R/W | <p>Dithering of CH1 Input to BRS</p> <p>These bits specify the number of colors for pixels after dithering (color reduction) when dithering (color reduction) for pixel information is enabled through the D1ON bit. When dithering (color reduction) for pixel information is disabled, specify 0 in these bits.</p> <p>B'000: Dithering of BRSin1 input image is disabled</p> <p>B'001: Dithering of BRSin1 input image at 18 bpp (RGB666: 260000 colors)</p> <p>B'010: Dithering of BRSin1 input image at 16 bpp (RGB565: 65535 colors)</p> <p>B'011: Dithering of BRSin1 input image at 15 bpp (RGB555: 32768 colors)</p> <p>B'100: Dithering of BRSin1 input image at 12 bpp (RGB444: 4096 colors)</p> <p>B'101: Dithering of BRSin1 input image at 8 bpp (RGB332: 256 colors)</p> <p>B'110, B'111: Setting prohibited</p> |
| 3 | ODE0 | B'0 | R/W | <p>Ordered Dither (mode A) of CH0 Input to BRS Enable/Disable</p> <p>0: Ordered dither (mode A) of BRSin0 is disabled.</p> <p>1: Ordered dither (mode A) of BRSin0 is enabled.</p> <p>Ordered dither is available only for 18bpp. So, when ODE0 bit is equal to 1, set DITH0 = 1.</p> <p>When ODE0 bit is equal to 0, BRSin0 dither method is specified by D0ON in the register</p> <p>Ordered dither (mode A) is recommended rather than ordered dither (mode B) in case of 18bpp.</p> |
| 2 to 0 | DITH0 [2:0] | B'000 | R/W | <p>Dithering of CH0 Input to BRS</p> <p>These bits specify how to perform dithering of the CH0 input to the BRS. The setting method is the same as that for the DITH1 bits. Read the description of the DITH1 bits with BRSin0 and D0ON replacing BRSin1 and D1ON, respectively.</p> |

Figure 33.55 shows the configuration of the BRS. For the BRS inputs, there are two inputs from the DPR and one internal input as a virtual RPF. BRSin0 to BRSin1 are input ports that have the target node values shown in Figure 33.31, and they can be connected to any module on the DPR. The same color space (YCbCr or RGB) has to be used for the two inputs from the DPR to the BRS.

The virtual RPF inside the BRS is an input unit not connected to the DPR. It is called the "virtual RPF" because it outputs images internally created by the BRS. Starting of the virtual RPF is controlled by VI6_WPFn_SRCRPF.VIR_ACT2, and the single-color data created at the virtual RPF can be used for blending or raster operation (ROP) with data from the other input units BRSin0 to BRSin1. The color space for the single color to be set for the virtual RPF needs to match the color space of the two inputs from the DPR to the BRS. For this setting method, see section 33.2.19.4.

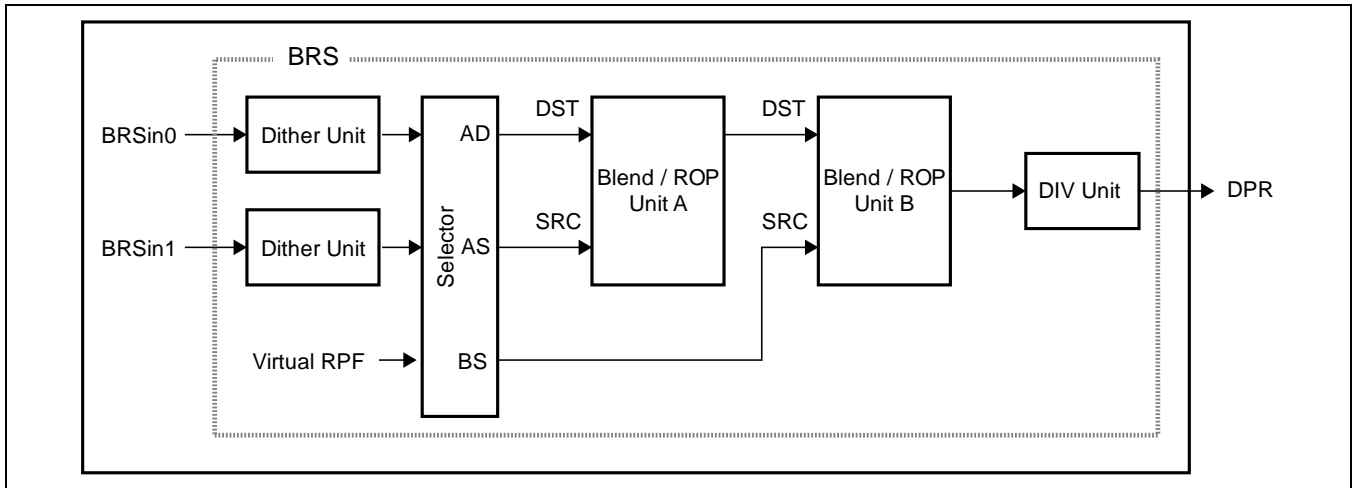


Figure 33.55 BRS Configuration

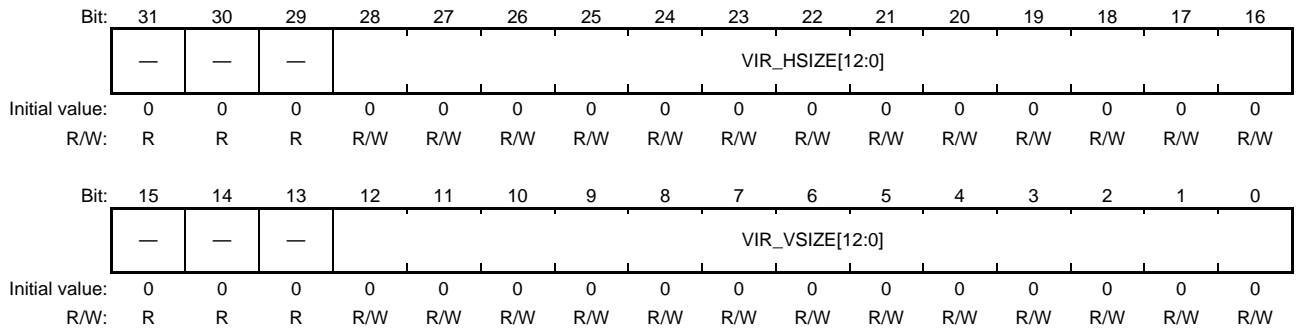
The selector in Figure 33.55 is used to select the SRC and DST inputs to blending/ROP units A to B from BRSin0 to BRSin1 which are inputs from the DPR and the virtual RPF. The SRC and DST input sources for blending/ROP units A to B are either uniquely determined based on the configuration shown in Figure 33.55 or selected as desired by registers. The input sources that can be arbitrarily selected by registers are AD, AS and BS, which correspond to the registers shown in Table 33.40.

Table 33.40 Correspondence between Selector Output Destinations and Register Bits for BRS

| Selector Output | Output Destination | Register Bits |
|-----------------|---------------------------|----------------------|
| AD | Blending/ROP unit A - DST | VI6_BRSA_CTRL.DSTSEL |
| AS | Blending/ROP unit A - SRC | VI6_BRSA_CTRL.SRCSEL |
| BS | Blending/ROP unit B - SRC | VI6_BRSB_CTRL.SRCSEL |

33.2.19.2 Size Register of BRS Input Virtual RPF (VI6_BRS_VIRPF_SIZE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

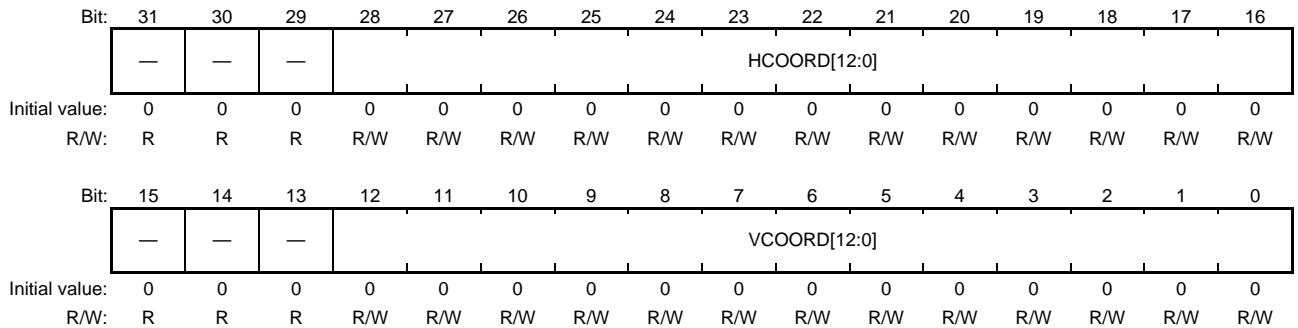


| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|------------------|---------------|-----|--|
| 31 to 29, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | VIR_HSIZE [12:0] | All 0 | R/W | Virtual RPF Horizontal Size These bits set the horizontal size of an image from the virtual RPF shown in Figure 33.55. A value from 1 to 8190 can be specified. |
| 12 to 0 | VIR_VSIZE [12:0] | All 0 | R/W | Virtual RPF Vertical Size These bits set the vertical size of an image from the virtual RPF shown in Figure 33.55. A value from 1 to 8190 can be specified. |

The virtual RPF has only a function to output a fixed α value and a fixed pixel value. The virtual RPF can internally create a single-color image without accessing external memory via the MAU. Same as images from the other BRS input ports, a sublayer can be blended on an image created in this manner with the image used as the background (master layer). In turn, when using the image as a sublayer, it can be drawn on the master layer as a window.

33.2.19.3 Display Location Register of BRS Input Virtual RPF (VI6_BRS_VIRPF_LOC)

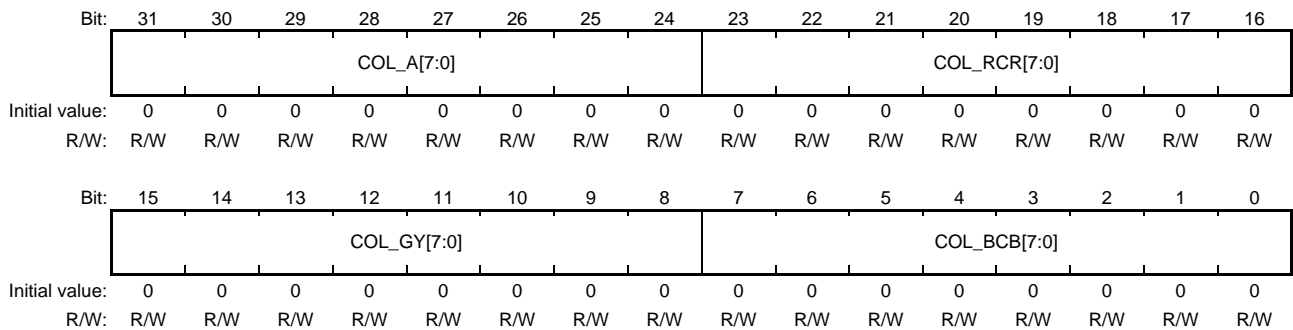
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |



| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|---------------|---------------|-----|--|
| 31 to 29, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | HCOORD [12:0] | All 0 | R/W | Horizontal Coordinate of Virtual RPF Location on Master Layer These bits specify the horizontal coordinate of where to locate the left-edge pixel of the virtual RPF's layer, with the left-edge pixel of the master layer set at coordinate 0. This setting should be made in pixel units. A value from 0 to 8189 can be specified. When the virtual RPF is specified as the master layer by VI6_WPFn_SRCRPF.VIR_ACT2, set these bits to 0. |
| 12 to 0 | VCOORD [12:0] | All 0 | R/W | Vertical Coordinate of Virtual RPF Location on Master Layer These bits specify the vertical coordinate of where to locate the top-edge pixel of the virtual RPF's layer, with the top-edge pixel of the master layer set at coordinate 0. This setting should be made in pixel units. A value from 0 to 8189 can be specified. When the virtual RPF is specified as the master layer by VI6_WPFn_SRCRPF.VIR_ACT2, set these bits to 0. |

33.2.19.4 Color Information Register of BRS Input Virtual RPF (VI6_BRS_VIRRPF_COL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 24 | COL_A [7:0] | All 0 | R/W | Fixed α of Virtual RPF These bits set the fixed α value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified. |
| 23 to 16 | COL_RCR [7:0] | All 0 | R/W | Fixed R/Cr of Virtual RPF These bits set the fixed R/Cr value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified. |
| 15 to 8 | COL_GY [7:0] | All 0 | R/W | Fixed G/Y of Virtual RPF These bits set the fixed G/Y value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified. |
| 7 to 0 | COL_BCB [7:0] | All 0 | R/W | Fixed B/Cb of Virtual RPF These bits set the fixed B/Cb value when the virtual RPF creates a virtual plane. A value from 0 to 255 can be specified. |

The way to set this register is same as BRU virtual RPF shown in section 33.2.15.4. See section 33.2.15.4 for more detail explanation.

33.2.19.5 BRS Control Registers (VI6_BRSm_CTRL: m = A, B)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|-----|----|----|----|----|----|----|----|-----|-------------|-----|-----|-----------|-------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RBC | — | — | — | — | — | — | — | — | DSTSEL[2:0] | | | — | SRCSEL[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | CROP[3:0] | | | AROP[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----------------------|--------------|---------------|-----|--|
| 31 | RBC | B'0 | R/W | Operation Type of Blending/ROP Unit m (m = A, B) Specifies the operation type for blending/ROP unit m (m = A, B) shown in Figure 33.55. 0: ROP (raster operation) 1: Blending operation |
| 30 to 23, 19, 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 22 to 20 | DSTSEL [2:0] | B'000 | R/W | Input Selection for DST Side of Blending/ROP Unit A These bits select the input for the DST side of blending/ROP unit A shown in Figure 33.55. These bits specify the connection between the BRS input port and the DST separately from the setting of connections between other modules and the BRS input port through the DPR. B'000: BRS input 0 (BRSin0) is input to DST B'001: BRS input 1 (BRSin1) is input to DST B'100: Virtual RPF is input to DST B'010, B'011, B'101 to B'111: Setting prohibited [Note] The DSTSEL bits for blending/ROP unit B is reserved. The write value should always be 0. |
| 18 to 16 | SRCSEL [2:0] | B'000 | R/W | Input Selection for SRC Side of Blending/ROP Unit m (m = A, B) These bits select the input for the SRC side of blending/ROP unit m (m = A, B) shown in Figure 33.55. These bits specify the connection between the BRS input port and the SRC separately from the setting of connections between other modules and the BRS input port through the DPR. B'000: BRS input 0 (BRSin0) is input to SRC B'001: BRS input 1 (BRSin1) is input to SRC B'100: Virtual RPF is input to SRC B'010, B'011, B'101 to B'111: Setting prohibited [Note] The SRCSEL bits for blending/ROP unit B are reserved. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------|---------------|-----|--|
| 7 to 4 | CROP [3:0] | H'0 | R/W | Color Data ROP Operator These bits select the ROP operator of the color data in blending/ROP unit m (m = A, B). Select the opcode for ROP operation from Table 33.35. |
| 3 to 0 | AROP [3:0] | H'0 | R/W | α Data ROP Operator These bits select the ROP operator of the α data in blending/ROP unit m (m = A, B). Select the opcode for ROP operation from Table 33.35. |

33.2.19.6 BRS Blend Control Registers (VI6_BRSm_BLD: m = A, B)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|-----|-----|-----|------------|-----|-----|------------|------------|-----|-----|-----|------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CBES | CCMDX[2:0] | | | — | CCMDY[2:0] | | | ABES | ACMDX[2:0] | | | — | ACMDY[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | COEFX[7:0] | | | | | | | | COEFY[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 | CBES | B'0 | R/W | Blending Expression Selection Selects the blending expression of the color data in the BRS (VI6_BRSm_CTRL.RBC = 1). Blending coefficients are specified by the CCMDX and CCMDY bits. 0: CCMDX * (DST color data) + CCMDY * (SRC color data) 1: CCMDX * (DST color data) – CCMDY * (SRC color data) |
| 30 to 28 | CCMDX [2:0] | B'000 | R/W | Blending Coefficient X Selection These bits specify coefficient X used in the blending expression determined by the CBES bit. B'000: DST α data is used as blending coefficient X B'001: 255 – (DST α data) is used as blending coefficient X B'010: SRC α data is used as blending coefficient X B'011: 255 – (SRC α data) is used as blending coefficient X B'100: Fixed α value 0 (COEFX setting) |
| 27, 19 | — | All 0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 26 to 24 | CCMDY [2:0] | B'000 | R/W | Blending Coefficient Y Selection These bits specify coefficient Y used in the blending expression determined by the CBES bit. B'000: DST α data is used as blending coefficient Y B'001: 255 – (DST α data) is used as blending coefficient Y B'010: SRC α data is used as blending coefficient Y B'011: 255 – (SRC α data) is used as blending coefficient Y B'100: Fixed α value 1 (COEFY setting) |
| 23 | ABES | B'0 | R/W | Blending α Creation Expression Specifies the expression for creating α data after blending by blending/ROP unit m (m = A, B, C, D, E). α creation coefficients are specified by the ACMDX and ACMDY bits. 0: ACMDX * (DST α data) + ACMDY * (SRC α data) 1: ACMDX * (DST α data) – ACMDY * (SRC α data) |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 22 to 20 | ACMDX [2:0] | B'000 | R/W | <p>α Creation Coefficient X</p> <p>These bits specify α creation coefficient X used in the α creation expression determined by the ABES bit.</p> <p>B'000: (α creation coefficient X) = (DST α data)</p> <p>B'001: (α creation coefficient X) = 255 – (DST α data)</p> <p>B'010: (α creation coefficient X) = (SRC α data)</p> <p>B'011: (α creation coefficient X) = 255 – (SRC α data)</p> <p>B'100: (α creation coefficient X) = Fixed α value 0 (COEFX setting)</p> <p>B'101 to B'111: Setting prohibited</p> |
| 18 to 16 | ACMDY [2:0] | B'000 | R/W | <p>α Creation Coefficient Y</p> <p>These bits specify α creation coefficient Y used in the α creation expression determined by the ABES bit.</p> <p>B'000: (α creation coefficient Y) = (DST α data)</p> <p>B'001: (α creation coefficient Y) = 255 – (DST α data)</p> <p>B'010: (α creation coefficient Y) = (SRC α data)</p> <p>B'011: (α creation coefficient Y) = 255 – (SRC α data)</p> <p>B'100: (α creation coefficient Y) = Fixed α value 1 (COEFY setting)</p> <p>B'101 to B'111: Setting prohibited</p> |
| 15 to 8 | COEFX [7:0] | All 0 | R/W | <p>Fixed α Value 0</p> <p>These bits specify fixed α value 0 used when the CCMDX or ACMDX bits are set to B'100. A value from H'00 to H'FF can be specified.</p> |
| 7 to 0 | COEFY [7:0] | All 0 | R/W | <p>Fixed α Value 1</p> <p>These bits specify fixed α value 1 used when the CCMDY or ACMDY bits are set to B'100. A value from H'00 to H'FF can be specified.</p> |

33.2.20 LIF Control Registers

33.2.20.1 LIFn Control Register (VI6_LIFn_CTRL)

[for n = 0]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[for n = 1]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|------------|-----|-----|-----|-----|-----|-----|------|-----|-----|--------|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | OBTH[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | CFMT | — | — | REQSEL | LIF_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------------------------------|----------------|---------------|-----|---|
| 31 to 28, 15 to 5, 3 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 to 16 | OBTH [11:0] | All 0 | R/W | Buffer Threshold for Start Ready Notification to Display Module [For VSPD] Always specify 3000, when LIF_EN is set to 1. [For VSPDL] RZ/G2H, RZ/G2N: Always specify 1500, when LIF_EN is set to 1. |
| 4 | CFMT | B'0 | R/W | Chroma Format This bit selects the output format from the LIF module to the display module. When RGB format is used, this bit shall be set to 0. 0: YCbCr444 or RGB Format 1: YCbCr422 Format [Note] The DU cannot receive YCbCr422 format. Therefore, when CFMT is 1, YCbCr444 data which information contents is equal to YCbCr422 is output to DU as shown in Figure 33.56. Set 1 to DEF10Rm.YCDF{0,1} (DU register) for DU to perform simple down sample for Cb/Cr. |
| 1 | REQSEL | B'0 | R/W | External Display Module Selection 0: This value is setting prohibited when 1 is set to LIF_EN 1: DU is selected as the destination external display module. |
| 0 | LIF_EN | B'0 | R/W | Enable/Disable of Data Output to External Display Module Enables or disables data output from the LIF to the external display module (DU). 0: Data output to the external display module is disabled. 1: Data output to the external display module is enabled. |

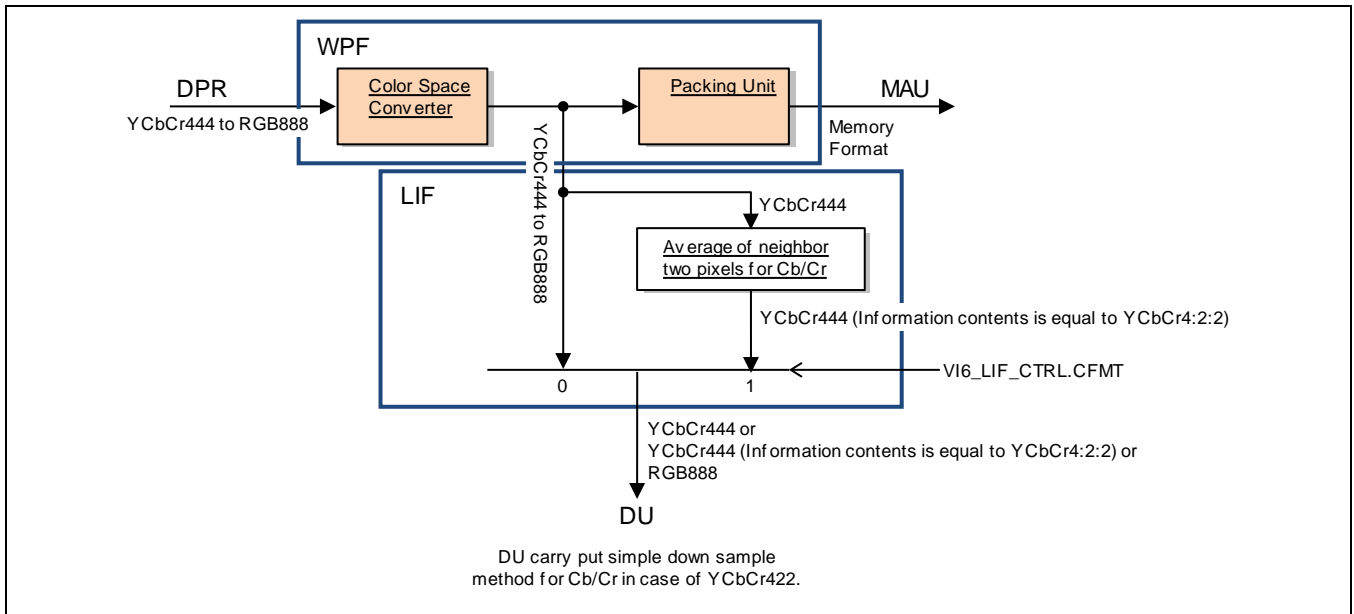


Figure 33.56 Data flow between LIF and DU

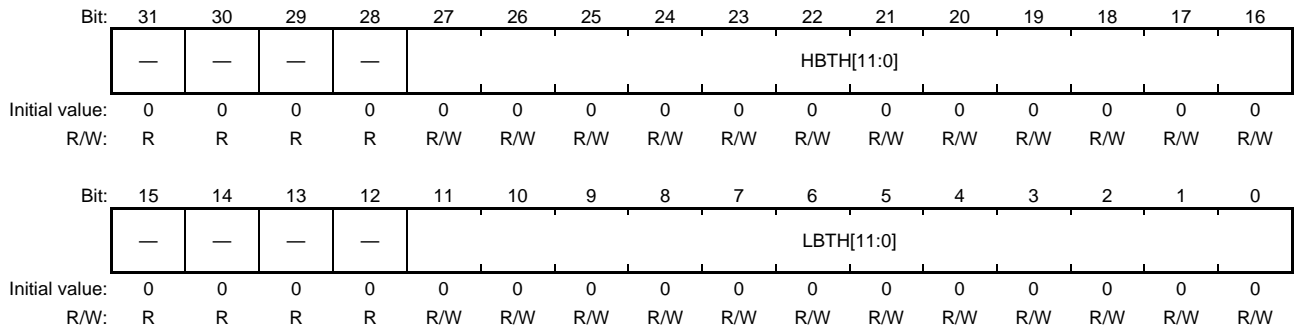
33.2.20.2 LIFn Clock Stop Buffer Control Register (VI6_LIFn_CSBTH)

[for n = 0]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[for n = 1]

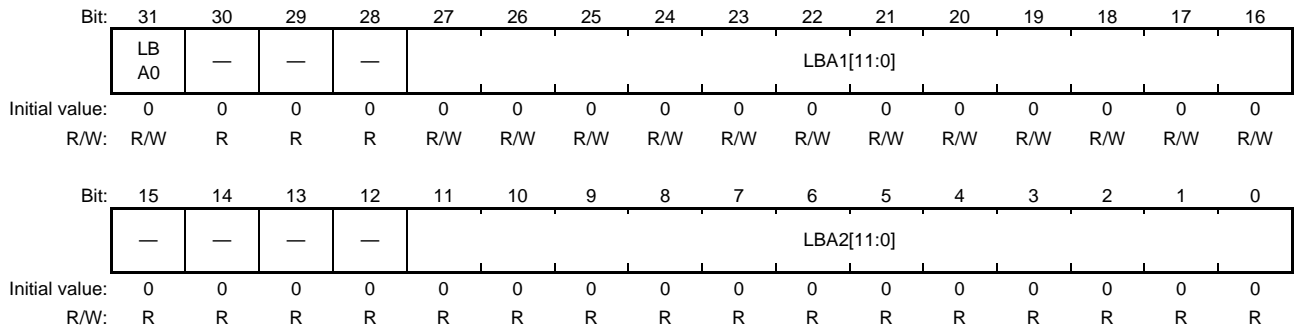
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |



| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|-------------|---------------|-----|---|
| 31 to 28, 15 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 to 16 | HBTH [11:0] | All 0 | R/W | Buffer Threshold for Clock Stop in Dynamic Clock Control Set HBTH[11:0] = 0 (fixed value) |
| 11 to 0 | LBTH [11:0] | All 0 | R/W | Buffer Threshold for Clock Start in Dynamic Clock Control Set LBTH[11:0] = 0 (fixed value) |

33.2.20.3 LIF0 Buffer Attribute Register (VI6_LIF0_LBA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | — | — |



| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|-------------|---------------|-----|---|
| 30 to 28, 15 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 31 | LBA0 | B'0 | R/W | LIF Buffer Attribute Register0 RZ/G2M V1.3, RZ/G2M V3.0: Always specify 0, when LIF_EN is set to 1 |
| 27 to 16 | LBA1 [11:0] | All 0 | R/W | LIF Buffer Attribute Register1 RZ/G2M V1.3, RZ/G2M V3.0: Always specify 0, when LIF_EN is set to 1 |
| 11 to 0 | LBA2 [11:0] | All 0 | R | LIF Buffer Attribute Register2 These bits are internal status for purpose of h/w debugging. |

33.2.21 SHP control registers

33.2.21.1 SHP Control Register0 (VI6_SHP_CTRL0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------|-----|-----|-----|-----|-----|-----|-----|-------|-----|-----|-----|-----|-----|------|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Limit0 | | | | | | | | Gain0 | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SIGN | SHP_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 15 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 31 to 24 | Limit0 | All 0 | R/W | Sharpness Parameter Limit0 There are 24 types of setting value to apply sharpness as shown in Table 33.41 and 24 types of setting value to apply blurring as shown in Table 33.42. Specify an appropriate value from among its combination. |
| 23 to 16 | Gain0 | All 0 | R/W | Sharpness Parameter Gain0 There are 24 types of setting value to apply sharpness as shown in Table 33.41 and 24 types of setting value to apply blurring as shown in Table 33.42. Specify an appropriate value from among its combination. |
| 1 | SIGN | B'0 | R/W | Sharpness Parameter SIGN There are 24 types of setting value to apply sharpness as shown in Table 33.41 and 24 types of setting value to apply blurring as shown in Table 33.42. Specify an appropriate value from among its combination. |
| 0 | SHP_EN | B'0 | R/W | Sharpness Processing Enable/Disable 0: Sharpness processing is disabled (input goes through the SHP without change) 1: Sharpness processing is enabled When this bit is set to 1, input format to the SHP module should be YCbCr format. Effect of sharpness is applied to Y component, not applied to Cb, Cr component. Effect of sharpness is applied to horizontal direction, not applied to vertical direction. |

To execute sharpness processing in YCbCr format, set up the RPF and DPR appropriately so that YCbCr-format image data is input to the SHP. Refer to section 33.2.6.3. For DPR settings, refer to section 33.2.8.1 to 33.2.8.4. Refer to Table 33.41 to apply sharpness. The SHP also has a function of blurring. Refer to Table 33.42 to apply blurring.

Table 33.41 Sharpness Parameter Setting (Sharpness)

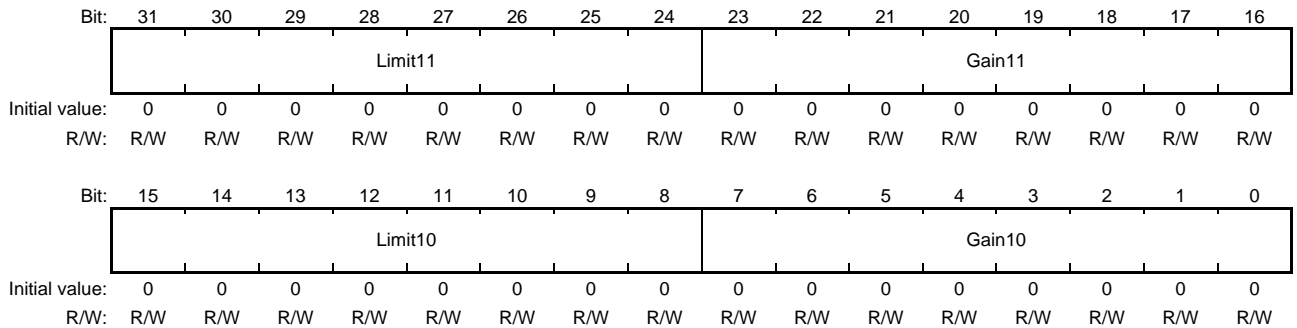
| Intensity | VI6_SHP_CTRL0 | | | | VI6_SHP_CTRL1 | | | | VI6_SHP_CTRL2 | | | | |
|-------------------|---------------|------|-------|--------|---------------|---------|--------|---------|---------------|---------|--------|---------|-----|
| | shp_en | sign | Gain0 | Limit0 | Gain10 | Limit10 | Gain11 | Limit11 | Gain20 | Limit20 | Gain21 | Limit21 | |
| Enhance weak | 1 | 1 | 0 | 4 | 64 | 0 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 2 | 1 | 0 | 8 | 64 | 0 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 3 | 1 | 0 | 12 | 64 | 0 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 4 | 1 | 0 | 16 | 64 | 0 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 5 | 1 | 0 | 20 | 64 | 0 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 6 | 1 | 0 | 24 | 64 | 0 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 7 | 1 | 0 | 24 | 64 | 16 | 64 | 224 | 255 | 0 | 64 | 0 | 0 |
| | 8 | 1 | 0 | 24 | 64 | 32 | 64 | 224 | 255 | 0 | 64 | 0 | 0 |
| | 9 | 1 | 0 | 24 | 64 | 48 | 64 | 224 | 255 | 0 | 64 | 0 | 0 |
| | 10 | 1 | 0 | 24 | 64 | 64 | 64 | 224 | 255 | 0 | 64 | 0 | 0 |
| | 11 | 1 | 0 | 24 | 64 | 80 | 64 | 224 | 255 | 0 | 64 | 0 | 0 |
| | 12 | 1 | 0 | 24 | 64 | 96 | 64 | 224 | 255 | 0 | 64 | 0 | 0 |
| | 13 | 1 | 0 | 24 | 64 | 120 | 64 | 224 | 255 | 0 | 64 | 0 | 0 |
| | 14 | 1 | 0 | 24 | 64 | 144 | 64 | 224 | 255 | 0 | 64 | 0 | 0 |
| | 15 | 1 | 0 | 24 | 64 | 168 | 64 | 224 | 255 | 0 | 64 | 0 | 0 |
| | 16 | 1 | 0 | 24 | 64 | 192 | 64 | 224 | 255 | 0 | 64 | 0 | 0 |
| | 17 | 1 | 0 | 24 | 64 | 192 | 64 | 224 | 255 | 24 | 64 | 224 | 255 |
| | 18 | 1 | 0 | 24 | 64 | 192 | 64 | 224 | 255 | 48 | 64 | 224 | 255 |
| | 19 | 1 | 0 | 24 | 64 | 192 | 64 | 224 | 255 | 72 | 64 | 224 | 255 |
| | 20 | 1 | 0 | 24 | 64 | 192 | 64 | 224 | 255 | 96 | 64 | 224 | 255 |
| | 21 | 1 | 0 | 26 | 64 | 208 | 64 | 224 | 255 | 104 | 64 | 224 | 255 |
| | 22 | 1 | 0 | 28 | 64 | 224 | 64 | 224 | 255 | 112 | 64 | 224 | 255 |
| | 23 | 1 | 0 | 30 | 64 | 240 | 64 | 224 | 255 | 120 | 64 | 224 | 255 |
| Enhance strong | 24 | 1 | 0 | 32 | 64 | 255 | 64 | 224 | 255 | 128 | 64 | 224 | 255 |

Table 33.42 Sharpness Parameter Setting (Blurring)

| Intensity | VI6_SHP_CTRL0 | | | | VI6_SHP_CTRL1 | | | | VI6_SHP_CTRL2 | | | | |
|-------------|---------------|------|-------|--------|---------------|---------|--------|---------|---------------|---------|--------|---------|---|
| | shp_en | sign | Gain0 | Limit0 | Gain10 | Limit10 | Gain11 | Limit11 | Gain20 | Limit20 | Gain21 | Limit21 | |
| Blur weak | 1 | 1 | 1 | 3 | 64 | 0 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 2 | 1 | 1 | 6 | 64 | 0 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 3 | 1 | 1 | 8 | 64 | 0 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 4 | 1 | 1 | 11 | 64 | 0 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 5 | 1 | 1 | 14 | 64 | 0 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 6 | 1 | 1 | 16 | 64 | 0 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 7 | 1 | 1 | 16 | 64 | 3 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 8 | 1 | 1 | 16 | 64 | 6 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 9 | 1 | 1 | 16 | 64 | 8 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 10 | 1 | 1 | 16 | 64 | 11 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 11 | 1 | 1 | 16 | 64 | 14 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 12 | 1 | 1 | 16 | 64 | 16 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 13 | 1 | 1 | 16 | 64 | 19 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 14 | 1 | 1 | 16 | 64 | 22 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 15 | 1 | 1 | 16 | 64 | 24 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 16 | 1 | 1 | 16 | 64 | 27 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 17 | 1 | 1 | 16 | 64 | 30 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 18 | 1 | 1 | 16 | 64 | 32 | 64 | 0 | 0 | 0 | 64 | 0 | 0 |
| | 19 | 1 | 1 | 16 | 64 | 32 | 64 | 0 | 0 | 3 | 64 | 0 | 0 |
| | 20 | 1 | 1 | 16 | 64 | 32 | 64 | 0 | 0 | 6 | 64 | 0 | 0 |
| | 21 | 1 | 1 | 16 | 64 | 32 | 64 | 0 | 0 | 8 | 64 | 0 | 0 |
| | 22 | 1 | 1 | 16 | 64 | 32 | 64 | 0 | 0 | 11 | 64 | 0 | 0 |
| | 23 | 1 | 1 | 16 | 64 | 32 | 64 | 0 | 0 | 14 | 64 | 0 | 0 |
| Blur Strong | 24 | 1 | 1 | 16 | 64 | 32 | 64 | 0 | 0 | 16 | 64 | 0 | 0 |

33.2.21.2 SHP Control Register1 (VI6_SHP_CTRL1)

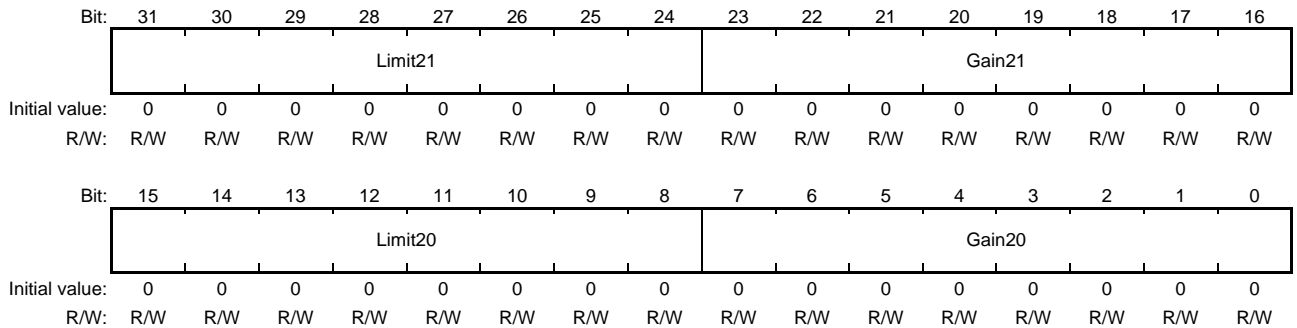
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 24 | Limit11 | All 0 | R/W | Sharpness Parameter Limit11 There are 24 types of setting value to apply sharpness as shown in Table 33.41 and 24 types of setting value to apply blurring as shown in Table 33.42. Specify an appropriate value from among its combination. |
| 23 to 16 | Gain11 | All 0 | R/W | Sharpness Parameter Gain11 There are 24 types of setting value to apply sharpness as shown in Table 33.41 and 24 types of setting value to apply blurring as shown in Table 33.42. Specify an appropriate value from among its combination. |
| 15 to 8 | Limit10 | All 0 | R/W | Sharpness Parameter Limit10 There are 24 types of setting value to apply sharpness as shown in Table 33.41 and 24 types of setting value to apply blurring as shown in Table 33.42. Specify an appropriate value from among its combination. |
| 7 to 0 | Gain10 | All 0 | R/W | Sharpness Parameter Gain10 There are 24 types of setting value to apply sharpness as shown in Table 33.41 and 24 types of setting value to apply blurring as shown in Table 33.42. Specify an appropriate value from among its combination. |

33.2.21.3 SHP Control Register2 (VI6_SHP_CTRL2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

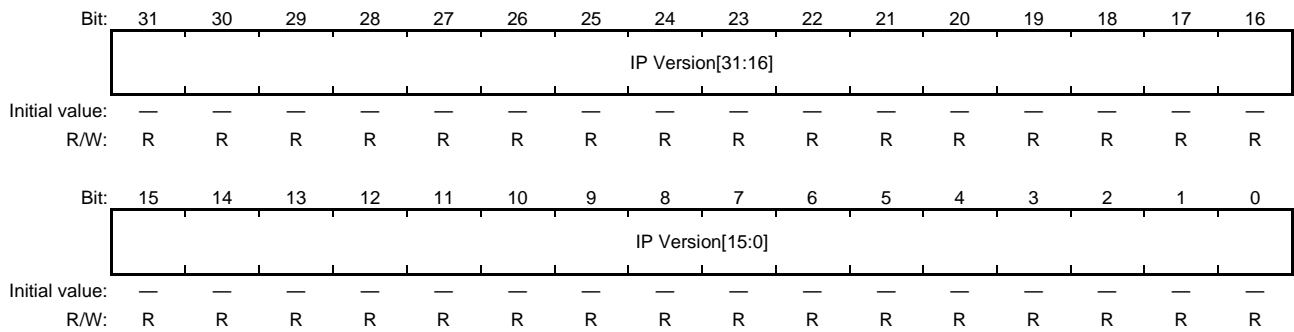


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 24 | Limit21 | All 0 | R/W | Sharpness Parameter Limit21 There are 24 types of setting value to apply sharpness as shown in Table 33.41 and 24 types of setting value to apply blurring as shown in Table 33.42. Specify an appropriate value from among its combination. |
| 23 to 16 | Gain21 | All 0 | R/W | Sharpness Parameter Gain21 There are 24 types of setting value to apply sharpness as shown in Table 33.41 and 24 types of setting value to apply blurring as shown in Table 33.42. Specify an appropriate value from among its combination. |
| 15 to 8 | Limit20 | All 0 | R/W | Sharpness Parameter Limit20 There are 24 types of setting value to apply sharpness as shown in Table 33.41 and 24 types of setting value to apply blurring as shown in Table 33.42. Specify an appropriate value from among its combination. |
| 7 to 0 | Gain20 | All 0 | R/W | Sharpness Parameter Gain20 There are 24 types of setting value to apply sharpness as shown in Table 33.41 and 24 types of setting value to apply blurring as shown in Table 33.42. Specify an appropriate value from among its combination. |

33.2.22 VSP internal registers

33.2.22.1 IP Version Register (VI6_IP_VERSION)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------------|---------------|-----|---|
| 31 to 0 | IP Version [31:0] | — | R | IP version register to indicate module kind and IP version. RZ/G2H VSPI: H'0101_1403 RZ/G2H VSPBD: H'0101_1503 RZ/G2H VSPBC: H'0101_1603 RZ/G2H VSPD: H'0101_1703 RZ/G2H VSPDL: H'0101_1903 RZ/G2M V1.3, RZ/G2M V3.0 VSPI: H'0101_1402 RZ/G2M V1.3, RZ/G2M V3.0 VSPB: H'0101_1502 RZ/G2M V1.3, RZ/G2M V3.0 VSPD: H'0101_1702 RZ/G2N VSPI: H'0101_1404 RZ/G2N VSPB: H'0101_1504 RZ/G2N VSPD: H'0101_1704 RZ/G2N VSPDL: H'0101_1904 RZ/G2E VSPI: H'0101_1404 RZ/G2E VSPB: H'0101_1504 RZ/G2E VSPD: H'0101_1704 |

33.2.23 RPFn OSD-CLUT Table

Refer to section 33.3.5.1.

33.2.24 LUT table

Refer to section 33.3.5.1.

33.2.25 CLU table address register and table data register

Refer to section 33.3.5.2.

33.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

33.3.1 Registers to set fixed value

Set fixed value to following registers in any case.

- [1] Always set H'0000 0808 to VI6_CLK_DCSWT
- [2] Always set D'256 to VI6_DL_CTRL.AR_WAIT [7:0] in case of using display list.
- [3] Always set D'2 to VI6_DL_EXT_CTRLn.POLINT [5:0] in case of using extended display list of WPFn.
- [4] Always set D'1 to VI6_DL_EXT_CTRLn.DLPRI in case of using extended display list of WPFn.
- [5] Always set D'0 to VI6_DL_EXT_CTRLn.EXPRI in case of using extended display list of WPFn.
- [6] Always set D'256 to VI6_WPF0_ROT_CTRL.LMEM_WD [12:0] in case of using rotation.
- [7] Always set H'0000 0500 to VI6_DPR_WPFn_FPORCH (n = 0, 1) in case of using WPFn.
- [8] For VSPD, always set D'3000 to VI6_LIF0_CTRL.OBTH[11:0] for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N or RZ/G2E in case of using LIF0 (VI6_LIF0_CTRL.LIF_EN = 1).
For VSPDL, always set D'1500 to VI6_LIFn_CTRL.OBTH[11:0] for RZ/G2H or RZ/G2N in case of using LIFn (VI6_LIFn_CTRL.LIF_EN = 1).
- [9] Always set H' 0000_0000 to VI6_LIF0_LBA for RZ/G2M V1.3, RZ/G2M V3.0 in case of using LIF0 (VI6_LIF0_CTRL.LIF_EN = 1).
- [10] Always set D'1 to VI6_LIFn_CTRL.REQSEL in case of using LIFn (VI6_LIFn_CTRL.LIF_EN = 1).
- [11] Always set D'0 to VI6_LIFn_CSBTH in case of using LIFn (VI6_LIFn_CTRL.LIF_EN = 1).

33.3.2 Concept of VSP2 Operation Starting and Stopping

The VSP2 provides two channels of image processing. Each channel is started by setting the corresponding start register. Here, starting a processing channel means starting one of WPF0 to WPF1, which are output modules of the VSP2. Use the start registers shown in Table 33.43 to start WPF modules.

After a WPF module is started and specified processing is completed, the corresponding channel stops operation and notifies the end of processing through an end interrupt. End interrupts are generated through the end interrupt source registers shown in Table 33.43; clearing a source register cancels the corresponding interrupt signal (for details of interrupt processing, refer to section 33.3.4, Interrupt Processing). Each of the operating status registers shown in the table indicates the busy state after the corresponding channel is started through the start register until processing is completed and operation stops. Figure 33.57 shows these operation timings.

Table 33.43 Target Module and Corresponding Registers for Starting and Stopping Operation

| Target Module | Start Register | End Interrupt Source Register | Operating Status Register |
|---------------|-----------------|-------------------------------|---------------------------|
| WPF0 | VI6_CMD0.STRCMD | VI6_WPF0_IRQ_STA. FRE | VI6_STATUS.SYS0_ACT |
| WPF1 | VI6_CMD1.STRCMD | VI6_WPF1_IRQ_STA. FRE | VI6_STATUS.SYS1_ACT |

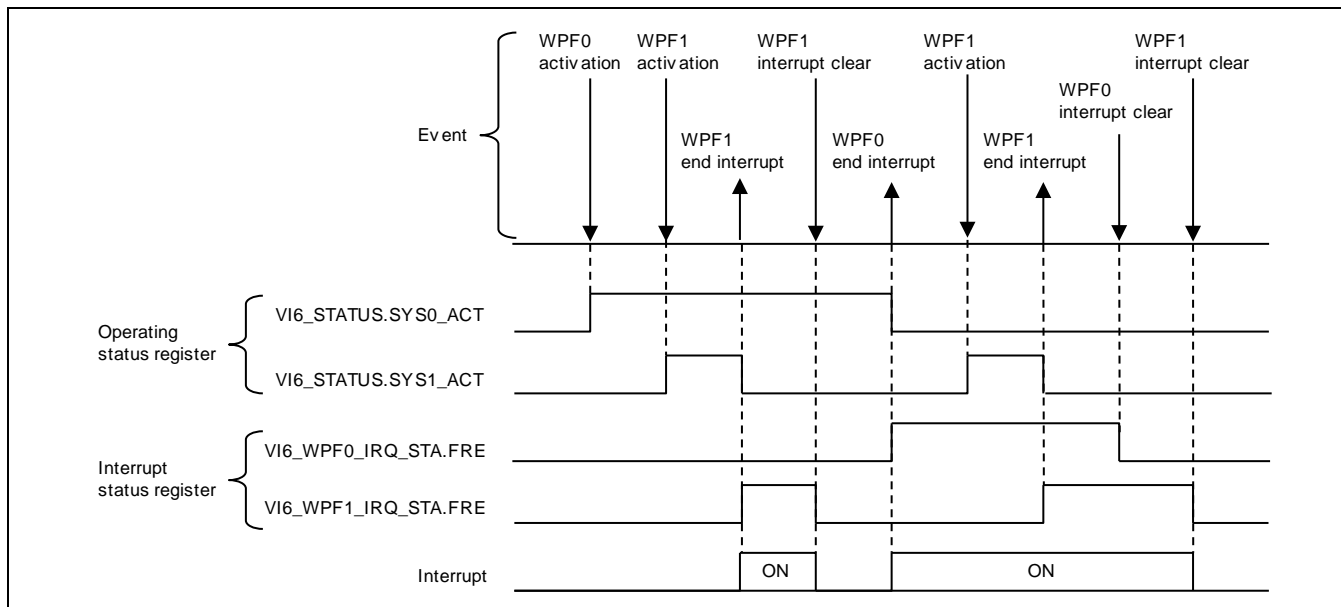


Figure 33.57 VSP2 Startup and Status of Each Register and Interrupt

The following describes the operating states (operating or stopped) of VSP2 internal modules. As described in section 33.1.2, the VSP2 has several image processing modules and the connections between modules are determined by the DPR. Accordingly, the operating state of a module is the same as that of the target WPF for that module. For example, when the target WPF for the BRU is WPF0, the BRU operating state is the same as that of WPF0; that is, the BRU operating state is indicated by the VI6_STATUS.SYS0_ACT as shown in Table 33.43 and the status change timing is shown as VI6_STATUS.SYS0_ACT in Figure 33.57. Likewise, the operating states of all modules connected to WPF0 is indicated by VI6_STATUS.SYS0_ACT.

After connections through the DPR are changed and the target WPF for a module is changed, refer to the correct register for the new target WPF to check the operating state of that module. In the above example, after the target WPF for the BRU is changed from WPF0 to WPF1, the BRU operating state is indicated by VI6_STATUS.SYS1_ACT.

Connections should be changed through the DPR-related registers (described later) while all modules to be affected by any change in connections are stopped. If connections through the DPR are changed during operation, the VSP2 will hang.

Figure 33.58 shows relation between start reservation and operating status. When start is reserved while VSP2 is operating, the reservation is accepted after VSP2 status is moved from "operating" into "idle" as shown in Figure 33.58. Therefore, when start reservation is used for VSPI and VSPB as shown in section 33.3.7.3 and section 33.3.8, take following procedure to confirm VSP2 isn't operating.

- <1> Check VI6_CMD0.STRCMD is zero (Start is not reserved)
- <2> Check VI6_STATUS.SYS0_ACT is zero (VSP2 is IDLE status)

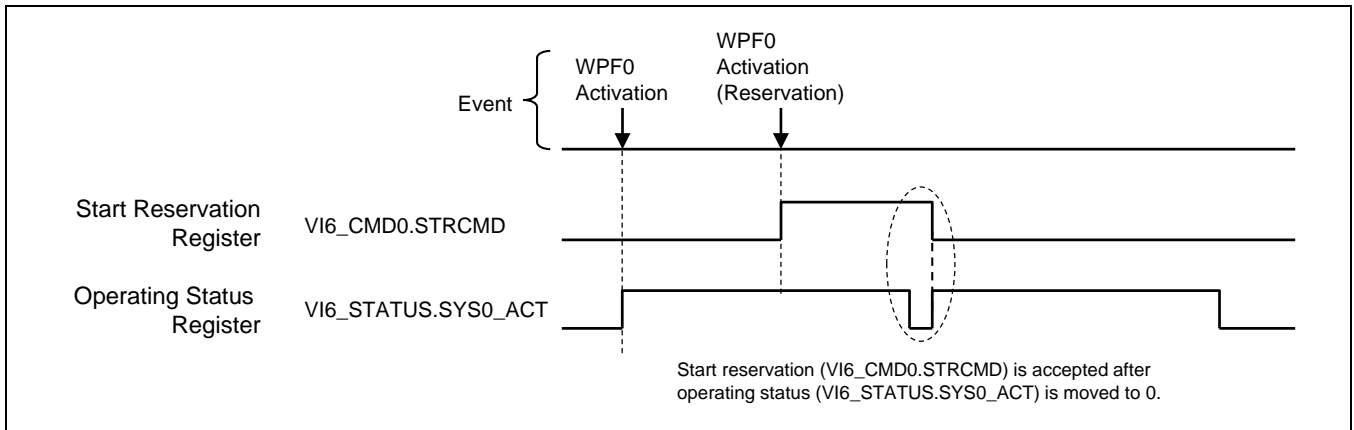


Figure 33.58 VSP2 Start reservation and Operating status

Note: WPF0 and WPF1 cannot be executed at same time for VSPD

33.3.3 Display List

33.3.3.1 Functional Description

The VSP2 provides the display list function. As a display list, the VSP2 automatically downloads the register settings except for the control registers (section 33.2.4 and 33.2.5) from external memory and stores the settings in the VSP2 registers. This function is advantageous in that the interrupt processing or register setting modification processing can be executed without CPU intervention during multiple-frame processing because the register settings used for VSP2 processing are prepared in advance in external memory such as SDRAM.

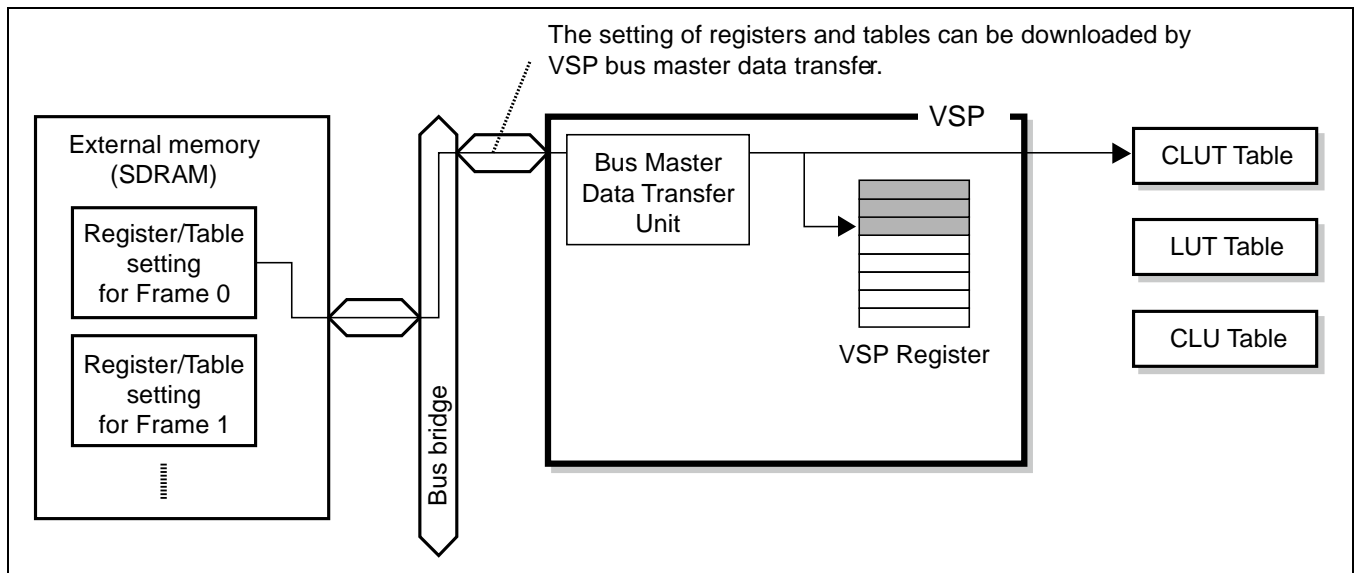


Figure 33.59 Concept of Display List

To use display lists, specify the external memory addresses to the display list control registers described in section 33.2.5. The register settings or various types of information should be stored in external memory in the format described in section 33.3.3.2. Figure 33.60 shows the difference between VSP2 operation through normal register settings and through display lists.

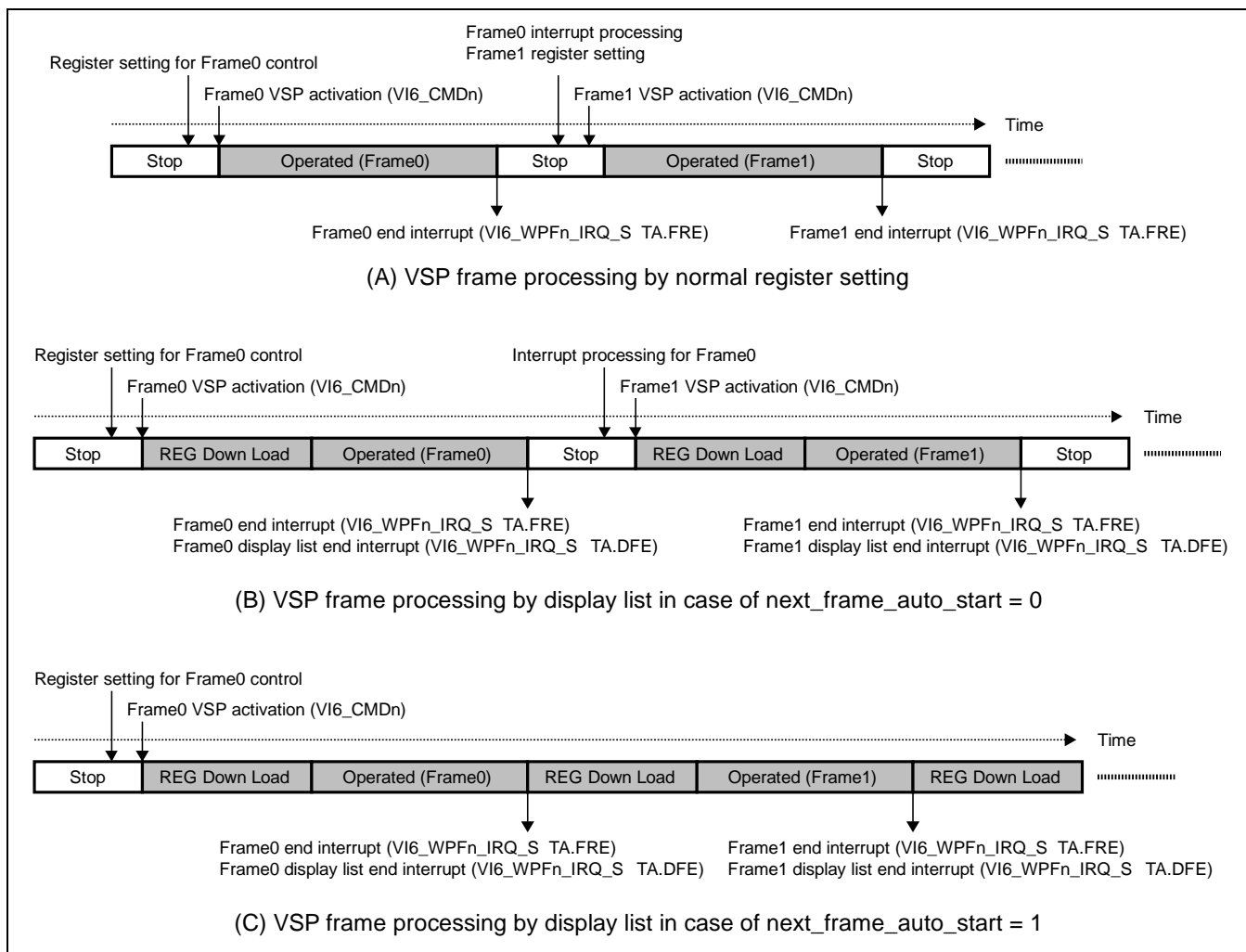


Figure 33.60 Comparison of VSP2 Operation between Normal Register Settings and Display Lists

As shown in Figure 33.60 (A), in the VSP2 processing through normal register settings, all registers should be set up before the VSP2 is started for each frame. After the VSP2 processing is completed, the VSP2 outputs a frame end interrupt (VI6_WPFn_IRQ_STA.FRE). This method requires a certain amount of time for register settings or interrupt processing by the CPU between frames. In contrast, when display lists are used, the VSP2 automatically downloads register settings from external memory as shown in Figure 33.60 (B) and (C), which reduces the load on the CPU between frames.

Figure 33.60 (B) shows the display list usage where the VSP2 stops at the end of every frame; only the VSP2 start processing for each frame is done by the CPU. This is suitable for the cases when the CPU controls synchronization of frame processing in frame buffer management or when the amount of register values or table data to be set in the VSP2 is large. In the case shown in Figure 33.60 (C), as soon as the frame processing ends, the VSP2 automatically begins next frame operation and starts downloading new register settings. This is the fastest operation using display lists.

Table 33.44 shows the modes of the display list and the supported functions for each mode. The detail of each mode is described in the following sections.

Table 33.44 Display list mode and supported functions

| Mode | Extended Display List | Continuous Frames |
|-------------------------------|-----------------------|--|
| Normal Display List Mode | Supported | Controlled by “next frame auto start” in the header of the display list. |
| Header-less Display List Mode | Not Supported | Controlled by VI6_DL_CTRL.CFM0 bit. |

33.3.3.2 Normal Display List Mode

The VSP2 display lists include control information as well as simple register settings in order to control multiple-frame processing in an optimum way for each application. Figure 33.61 shows the display list structure.

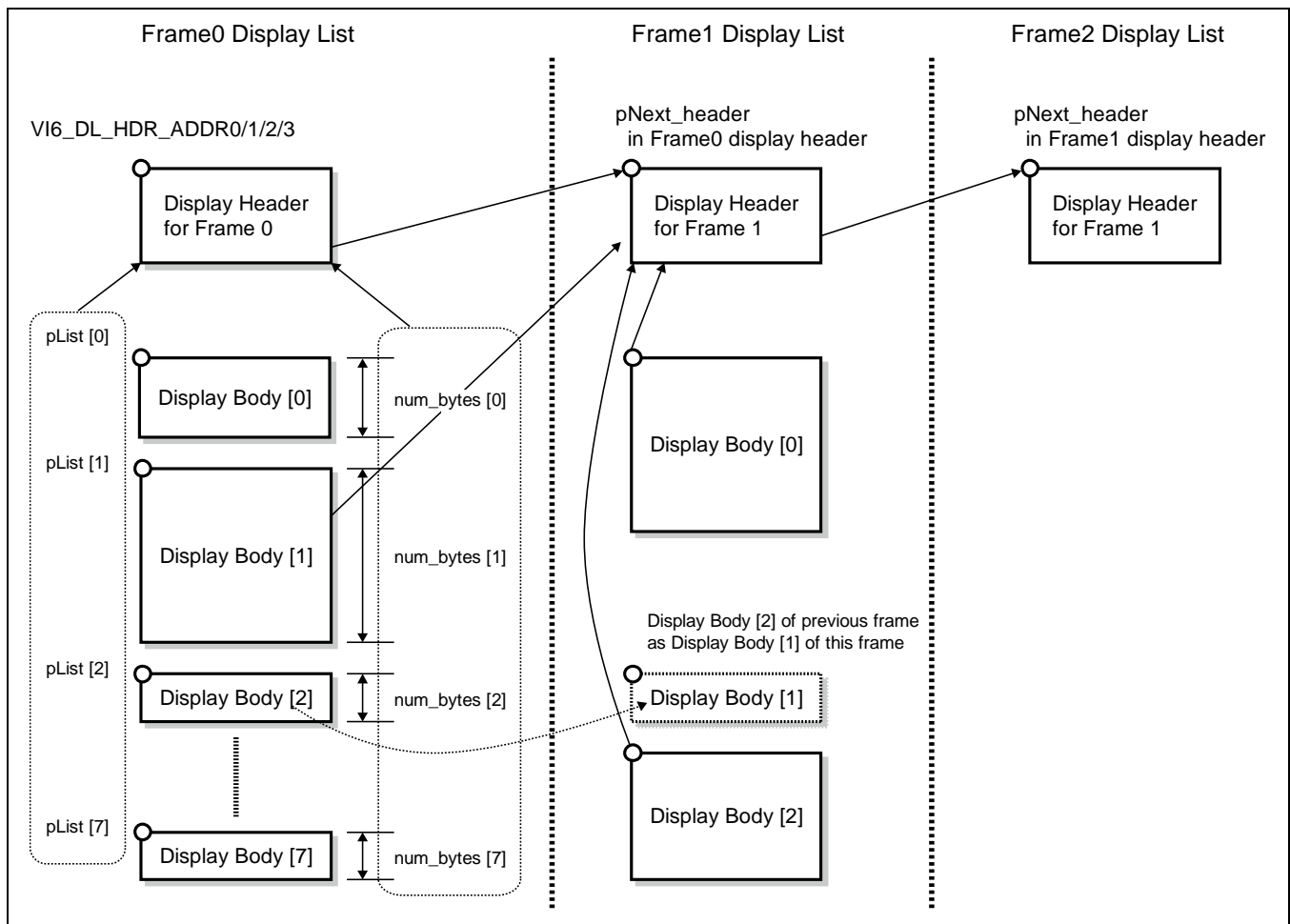


Figure 33.61 Structure and Concept of VSP2 Display List

A VSP2 display list consists of two sections; a header section for storing various information and control flags and a body section for storing register and table settings. A combination of these two sections is defined as a display list for a frame. The register and table settings can be divided and stored in up to eight separate bodies allocated in memory. Therefore, the separate bodies storing the register settings for one frame (for example, frame 0) can have non-sequential start addresses; that is, the bodies for one frame can be allocated to areas distant from each other in memory. To gather these bodies and configure the register settings for one frame, a header is used. A display header stores the number of bodies linked with the header and the start address and size of each body.

The VSP2 analyzes the header to gather register and table settings stored in separate memory areas and reconfigures the complete register settings.

The addresses of display headers should be specified in the VSP2 registers described in section 33.2.5. When activated in a mode that uses display lists, the VSP2 downloads display headers from the addresses specified in VI6_DL_HDR_ADDR0/1/2/3 (numbers 0 to 3 correspond to the WPF channel index numbers), analyzes the numbers of bodies and the address and data size of each body, downloads the bodies, and completes register and table settings. After display list downloading is completed, the VSP2 becomes ready for frame image processing; the VSP2 then starts the actual frame processing.

After processing of a frame ends, the VSP2 proceeds to the next frame processing. Here, there are two modes for starting the next frame processing as shown in Figure 33.60. In one mode, the VSP2 stops operation and waits for the next

activation by the CPU in the same way as when display lists are not used. In this mode, the address used for downloading the display header of the next frame is kept by the internal hardware. Therefore, if valid address information is not stored in the display header for the previous frame, a correct value should be specified in VI6_DL_HDR_ADDR0/1/2/3 while the VSP2 is stopped. When the VSP2 is started after a correct value is specified, the VSP2 starts next frame processing with the same procedure for the previous frame. In contrast to this mode, which stops the VSP2 after the end of one-frame processing, there is another mode for automatically starting next frame processing. In automatic start mode, the VSP2 downloads the next display header as soon as the previous frame processing ends. After downloading ends, the VSP2 starts image processing. The information regarding mode selection, that is, whether to automatically start next frame processing, should be stored in the display header downloaded for the previous frame.

In the automatic start mode, the VSP2 continues processing until the display header for a frame specifies that the next frame should not be started automatically. To stop processing during automatic execution, use a software reset (VI6_SRESET).

To strictly define the display list format described above, the following shows the grammatical structure of a display list using pseudo-code. First, to simplify the description of the display list format in the following pages, Table 33.45 defines a function. Function zero bits (num bits) generates a string of one-bit 0s for the number of bits specified by the parameter for the function. By using this function, Table 33.46 defines the header section format of a display list and Table 33.47 defines the body section format and Table 33.48 defines the extended display list body section format.

Data order of Display List Header Section, Display List Body Section and Extended Display List Body Section are assumed as the data is stored in SDRAM by big endian. Table 33.49 shows an example of data order. If data order is not same as the definition, data order within 8byte unit can be adjusted by setting VI6_DL_SWAP (33.2.5.3).

Table 33.45 Definition of a Function for Simple Description

| Syntax | Bit Count |
|------------------------------|---|
| zero_bits (num_bits) | |
| { | |
| for (i=0; i<num_bits; i++) { | |
| zero_bit | 1 |
| } | |
| } | |
| | |
| Bit String | Contents |
| zero_bit | zero_bit indicates a 1-bit integer having a value of 0. |

Table 33.46 Format of Display List Header Section

| Syntax | Bit Count |
|--|-----------|
| display_header () /* Fixed length */ | |
| { | |
| zero_bits (29) | |
| num_list_minus1 | 3 |
| for (i = 0; i<8; i++) | |
| zero_bits (15) | |
| num_bytes [i] | 17 |
| pList [i] | 32 |
| } | |
| pNext_header | 32 |
| zero_bits (30) | |
| current_frame_int_enable | 1 |
| next_frame_auto_start /* 76 bytes from the beginning of this header*/ | 1 |
| if (VI6_DL_EXT_CTRL.EXT) { | |
| zero_bits (32) /* padding zero 4 bytes for alignment */ | |
| zero_bits (6) | |
| pre_ext_dl_exec | 1 |
| post_ext_dl_exec | 1 |
| zero_bits (8) | |
| pre_ext_dl_num_cmd | 16 |
| pre_ext_dl_pList | 32 |
| zero_bits (16) | |
| post_ext_dl_num_cmd | 16 |
| post_ext_dl_pList /* 96 bytes from the beginning of this header*/ | 32 |
| } | |
| } | |

| Bit String | Contents |
|--------------------------|---|
| num_list_minus1 | Specifies the value obtained by subtracting 1 from the total number of display list bodies linked with the display header. For example, when this bit field is set to 0, this display list uses one body. |
| num_bytes [i] | Specifies the number of bytes in the i-th display list body (indicated by index i). Be sure to specify a multiple of eight bytes. For the bodies that are not defined in num_list_minus1 (for example, i = 5 to 7 when num_list_minus1 is set to 4), specify 0. |
| pList [i] | Specifies the start address of the i-th display list body (indicated by index i). Be sure to specify an address aligned with an 8-byte boundary (the lower-order three bits are 0). For the bodies that are not defined in num_list_minus1 (for example, i = 5 to 7 when num_list_minus1 is set to 4), specify 0. |
| pNext_header | Specifies the address of the display list header for the next frame. After display list downloading ends, the VSP2 keeps its value in the internal memory and uses it in the next display list header downloading. Be sure to specify an address aligned with an 8-byte boundary (the lower-order three bits are 0). |
| current_frame_int_enable | This is a flag that indicates whether to set the display list end interrupt source (VI6_WPFn_IRQ_STA.DFE) to 1 when the current frame processing ends. If this flag is set to 0, the display list end interrupt source (VI6_WPFn_IRQ_STA.DFE) is not set to 1 when one-frame processing by this display header ends. In this state, even if the display list end interrupt is enabled (VI6_WPFn_IRQ_ENB.DFEE is set to 1), no interrupt will be generated. If this flag is set to 1, the display list end interrupt source (VI6_WPFn_IRQ_STA.DFE) is set to 1 when one-frame processing by this display header ends. In this state, if the display list end interrupt is enabled (VI6_WPFn_IRQ_ENB.DFEE is set to 1), the VSP2 generates an interrupt. |
| Next_frame_auto_start | Enables or disables automatic start of next frame processing when one-frame processing by this display header ends. If this bit is set to 1, the VSP2 starts next frame processing as soon as one-frame processing by this display header ends, and starts downloading the next frame display header from the pNext_header address specified in this display header. If this bit is set to 0, the VSP2 stops operation when one-frame processing by this display header ends. In this case, start the VSP2 through VI6_CMDn to process the next frame. |
| pre_ext_dl_exec | Enables execution of the extended display list for frame preprocessing when VI6_DL_EXT_CTRL.EXT is 1. If this bit is set to 1, the VSP2 executes the extended display list for frame preprocessing. The VSP2 does not execute it if this bit is set to 0. This bit setting is ignored when VI6_DL_EXT_CTRL.EXT is 0. |
| post_ext_dl_exec | Set 0 to this bit. |
| pre_ext_dl_num_cmd | Specifies the number of commands in the extended display list body section for frame preprocessing when VI6_DL_EXT_CTRL.EXT is 1. The number of commands that can be specified is 1, and a command is 16 bytes. When pre_ext_dl_exec is set to 0, the extended display list for frame preprocessing is not executed; specify 0 in this bit. This bit setting is ignored when VI6_DL_EXT_CTRL.EXT is 0. |
| pre_ext_dl_pList | Specifies the start address of the area where the extended display list body section for frame preprocessing is stored when VI6_DL_EXT_CTRL.EXT is 1. Be sure to specify an address aligned with a 16-byte boundary (lower-order four bits are 0). When pre_ext_dl_exec is set to 0, the extended display list for frame preprocessing is not executed; specify 0 in this bit. This bit setting is ignored when VI6_DL_EXT_CTRL.EXT is 0. |
| post_ext_dl_num_cmd | Set 0 to this bit. |
| post_ext_dl_pList | Set 0 to this bit. |

Table 33.47 Format of Display List Body Section

| Syntax | Bit Count |
|--|--|
| display_list (num_bytes) /* Variable length (num_bytes) */ | |
| for (i = 0; i < num_bytes; i + = 8) { | |
| set_address | 32 |
| set_data | 32 |
| } | |
| } | |
| Bit String | Contents |
| set_address | Specifies the address where the value specified by set_data is to be stored. Specify a register address. Note: Register space of VSP is 32Kbyte (H'0000 - H'7FFC). So, set 0 to upper 17bit of set_address in Display List Body Section. |
| set_data | Specifies the value to be stored in the address specified by set_address. Specify a value to be set in a register. |

Table 33.48 Format of Extended Display List Body Section

| Syntax | Bit width |
|---|-----------|
| ext_dl_display_list (num_llw) /* Variable length (pre/post_ext_dl_num_bytes) */ | |
| for (i = 0; i < num_llw; i + = 2) { | |
| ext_dl_cmd | 64 |
| ext_dl_data | 64 |
| } | |
| } | |

See section 33.3.6.3 and section 33.3.6.4 for setting extended display list body section.

Table 33.49 Data order of display list body

| Address | Syntax |
|------------|----------------------------|
| pList | set_address [0] (bit31-24) |
| pList + 1 | set_address [0] (bit23-16) |
| pList + 2 | set_address [0] (bit15-8) |
| pList + 3 | set_address [0] (bit7-0) |
| pList + 4 | set_data [0] (bit31-24) |
| pList + 5 | set_data [0] (bit23-16) |
| pList + 6 | set_data [0] (bit15-8) |
| pList + 7 | set_data [0] (bit7-0) |
| pList + 8 | set_address [1] (bit31-24) |
| pList + 9 | set_address [1] (bit23-16) |
| pList + 10 | set_address [1] (bit15-8) |
| pList + 11 | set_address [1] (bit7-0) |
| pList + 12 | set_data [1] (bit31-24) |
| pList + 13 | set_data [1] (bit23-16) |
| pList + 14 | set_data [1] (bit15-8) |

| Address | Syntax |
|----------------|-----------------------|
| pList + 15 | set_data [1] (bit7-0) |
| ~~~ | ~~~ |

33.3.3.3 Header-less Display List Mode

The header-less display list does not have the display header listed in Table 33.46, and it has the simplest structure that has only single body. The extended display list function is not available in case of the header-less display list mode. Set the value 1 to VI6_DL_CTRL.NH0 (see the section 33.2.5.1) to use the header-less display list. The start address downloaded by the header-less display list should be set to VI6_DL_HDR_ADDR0. And the size of the display body which is originally defined in the display header should be set to VI6_DL_BODY_SIZE0. The header-less display list is available only in WPF0.

33.3.3.4 Restrictions on Display List Usage

Access to the general control registers and display list control registers through a display list is prohibited. When using display lists, be sure to observe the following restrictions on register access by the CPU.

1. Do not execute write access to the same register (same address) from the CPU and through a display list at the same time. If such a conflict occurs, correct operation of the VSP2 is not guaranteed.
2. Do not execute write access to the same OSD-CLUT, CLU, or LUT lookup table from the CPU and through a display list at the same time. If such a conflict occurs, correct operation of the VSP2 is not guaranteed. Here, the same lookup table means the address space having the same space name shown in Table 33.50 or Table 33.51.
3. When read access by the CPU and write access through a display list to the same register (same address) occur at the same time, the read value returned to the CPU is not guaranteed.
4. When read access by the CPU and write access through a display list to the same OSD-CLUT, CLU, or LUT lookup table occur at the same time, the read value returned to the CPU is not guaranteed. Here, the same lookup table means the address space having the same space name shown in Table 33.50 or Table 33.51.
5. For other restrictions on values and timing of register setting through display lists, refer to the restrictions on normal register settings described in section 33.2.3.
6. Manipulation and setting of the registers described in section 33.2.4 and 33.2.5 through a display list is prohibited.
7. The extended display list function enabled by VI6_DL_EXT_CTRL.EXT is to control the registers of the ICB connected with the VSP2. Do not use extended display lists for any other purpose.

33.3.4 Interrupt Processing

The VSP2 provides multiple image processing channels (WPF0 to WPF1) and they can operate simultaneously in parallel. When a WPF module generates an internal source that should be notified, an interrupt signal is output. As internal sources are generated in each WPF independently, the VSP2 has the following registers to control interrupts in each WPF.

WPFn Interrupt Enable Registers (VI6_WPFn_IRQ_ENB: n = 0, 1) (refer to section 33.2.4.12)

WPFn Interrupt Status Registers (VI6_WPFn_IRQ_STA: n = 0, 1) (refer to section 33.2.4.13)

While multiple WPF modules operate in parallel, they may output interrupt requests at the same time (this is called a multiple interrupt state). Read the WPFn Interrupt Status Registers (VI6_WPFn_IRQ_STA: n = 0, 1) in sequence and execute the interrupt response processing for each interrupt source.

(Note1)

To use interrupts, enable them through WPF interrupt enable registers. If an interrupt source register has already been set to 1 for some reason, an unintended interrupt will occur as soon as the corresponding interrupt enable register is set as enabled. To avoid this, before enabling interrupts through WPF interrupt enable registers, be sure to clear all WPF interrupt sources to be enabled to 0. Be careful about this procedure when setting up registers before starting the VSP2.

(Note2)

WPF0 and WPF1 cannot be executed at same time for VSPD.

33.3.5 Lookup Table Settings

33.3.5.1 OSD-CLUT or LUT

For a single entry to the OSD-CLUT space (see Table 33.50) or LUT space (see Table 33.50) of the VSP2, the OSD-CLUT or LUT data is set by a write access in the format shown in Figure 33.62.

The entry address of each space is (start address of the space) + (entry number counting from the base point 0 × 4). For example, the address of entry 7 to the LUT space is (offset address) + H'7000 + 7 × 4 = (offset address) + H'701C.

Table 33.50 shows the spaces for which entries can be made. Note that if the module that references that space is operating, write accesses to the relevant space are prohibited. For example, while RPF2 is operating, accesses to the entire space of VI6_CLUT2_TBL are prohibited. When a read access is made to the relevant space during operation of the referencing module, undefined values will be read out.

The operating/stopped state of each module in Table 33.50 is the operating state of the WPF to which each module is connected. Determine whether the module is operating or stopped using each WPF operating status bit in the VI6_STATUS register.

Table 33.50 OSD-CLUT Space and LUT Space Addresses

| Space Name | Space Addresses | Entry Count | Module that References the Space in the Left Column |
|---------------|--|-------------|---|
| VI6_CLUT0_TBL | (offset address) + H'4000 to (offset address) + H'43FC | 256 | RPF0 |
| VI6_CLUT1_TBL | (offset address) + H'4400 to (offset address) + H'47FC | 256 | RPF1 |
| VI6_CLUT2_TBL | (offset address) + H'4800 to (offset address) + H'4BFC | 256 | RPF2 |
| VI6_LUT_TBL | (offset address) + H'7000 to (offset address) + H'73FC | 256 | LUT |

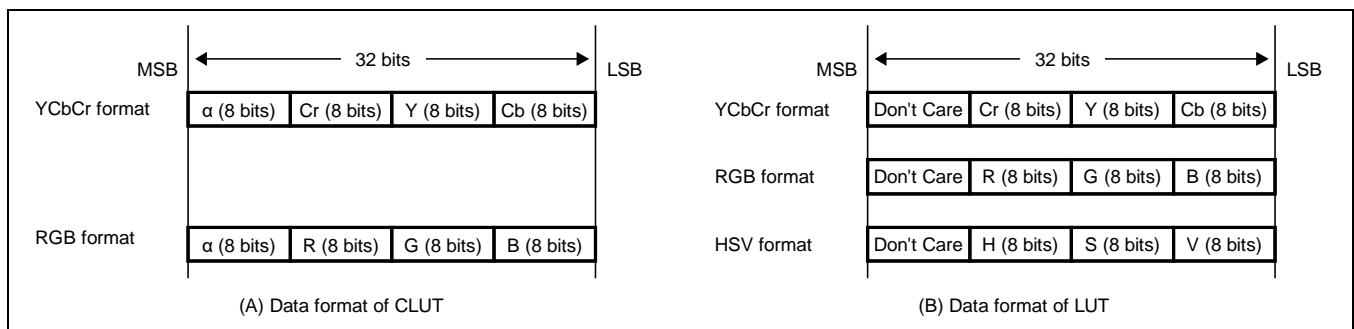


Figure 33.62 VI6_CLUT_TBL and VI6_LUT_TBL Formats

33.3.5.2 CLU Table

Data is set in the CLU table in the space shown in Table 33.51 through indirect addressing.

Table 33.51 CLU Space Addresses

| Space Name | Space Addresses | Module that References the Space in the Left Column |
|-------------|--|---|
| VI6_CLU_TBL | (offset address) + H'7400 to (offset address) + H'7407 | CLU |

The addresses used in indirect addressing (VI6_CLU_ADDR, VI6_CLU_DATA) are shown in Table 33.52. The CLU table can be accessed by setting the VI6_CLU_ADDR register and then reading from or writing to the VI6_CLU_DATA register. To write-access the CLU table continuously, such as when setting data to the CLU table, perform the following.

- (1) Set the first CLU table coordinates in VI6_CLU_ADDR.
 - (2) Write the first data to be set in VI6_CLU_DATA.
 - (3) Set the next CLU table coordinates in VI6_CLU_ADDR.
 - (4) Write the next data to be set in VI6_CLU_DATA.
- Execute these processes in a similar manner.

Accesses to VI6_CLU_ADDR and VI6_CLU_DATA registers are repeated in this way. The location where to access the CLU table to write the VI6_CLU_DATA contents immediately takes effect after a value is set in the VI6_CLU_ADDR register.

To read-access the CLU table continuously, change the write-access operation to the read-access operation in the sequence of processes of (1) to (4) above.

The following describes how to specify VI6_CLU_ADDR.

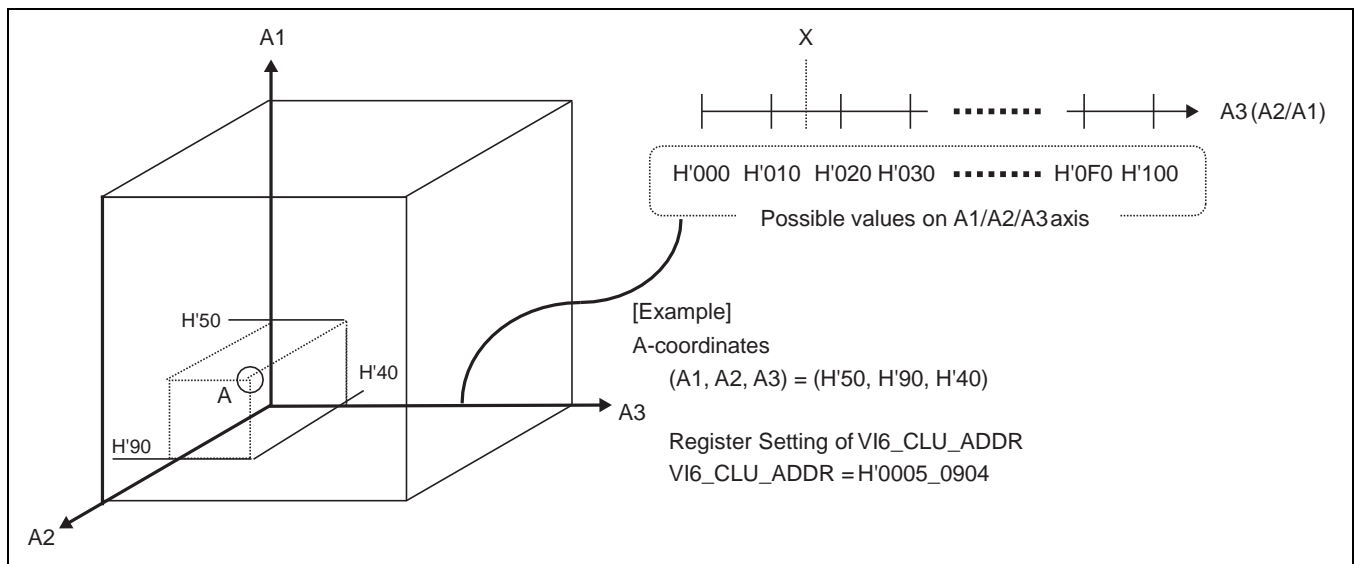


Figure 33.63 VI6_CLU_ADDR Setting

In the CLU table, a table value should be specified for every increase by H'10 along each of the axes and for every intersection between two axes. For a certain axis, as shown in the top right of Figure 33.63, a value to be set in the table corresponds to a scale mark at an interval of H'10. The location of a point between two scale marks (for example, point X in Figure 33.63) is calculated through interpolation by the CLU (refer to section 33.2.12.1). Accordingly, the lower four bits of table settings for each axis has no meaning; only the upper-order five bits of the desired coordinate for each component should be specified in VI6_CLU_ADDR (which should be shifted to the right by four bits) as shown in Figure 33.63.

Table 33.52 Address Spaces Used in Indirect Addressing to CLU Table

| Address (Name) | Bit | Initial Value | R/W | Description |
|--|----------|---------------|-----|--|
| (offset address) + H'7400 (VI6_CLU_ADDR) | 31 to 24 | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| | 23 to 16 | All 0 | R/W | Coordinate Value of First Axis These bits set the coordinate value of the first axis on the three-dimensional space coordinates of the CLU table. A value from 0 to 16 can be specified. |
| | 15 to 8 | All 0 | R/W | Coordinate Value of Second Axis These bits set the coordinate value of the second axis on the three-dimensional space coordinates of the CLU table. A value from 0 to 16 can be specified. |
| | 7 to 0 | All 0 | R/W | Coordinate Value of Third Axis These bits set the coordinate value of the third axis on the three-dimensional space coordinates of the CLU table. A value from 0 to 16 can be specified when the CLU is in 3D mode. In 2D mode, these bits must be set to 0. |
| (offset address) + H'7404 (VI6_CLU_DATA) | 31 to 24 | Undefined | R | Reserved These bits are always read as 0. The write value should always be 0. |
| | 23 to 16 | Undefined | R/W | Component Value of First Axis These bits set the component value of the first axis on the three-dimensional space coordinates of the CLU table, which is defined by the VI6_CLU_ADDR register. A value from 0 to 255 can be specified when the CLU is in 3D mode. In 2D mode, these bits must be set to 0. |
| | 15 to 8 | Undefined | R/W | Component Value of Second Axis These bits set the component value of the second axis on the three-dimensional space coordinates of the CLU table, which is defined by the VI6_CLU_ADDR register. A value from 0 to 255 can be specified. |
| | 7 to 0 | Undefined | R/W | Component Value of Third Axis These bits set the component value of the third axis on the three-dimensional space coordinates of the CLU table, which is defined by the VI6_CLU_ADDR register. A value from 0 to 255 can be specified when the CLU is in 3D mode. In 2D mode, these bits must be set to 0. |

Accessing way of "Automatic Table Address Increment (VI6_CLU_CTRL.AAI = 1)" is shown below.

```
(1) Set 1 in VI6_CLU_CTRL.AAI
(2) Set H'0000_0000 in VI6_CLU_ADDR (initial coordinate)
(3) Write the data in VI6_CLU_DATA by following order (pseudo-code).
for (ax3 = 0; ax3 < 17; ax3 ++) {
    for (ax2 = 0; ax2 < 17; ax2 ++) {
        for (ax1 = 0; ax1 < 17; ax1 ++) {
            Write the data in VI6_CLU_DATA
            for the coordinate of "VI6_CLU_ADDR = (ax1 << 16) + (ax2 << 8) + ax3".
        }
    }
}
```


33.3.6 Linked with DU

33.3.6.1 Operation flow of VSPD and DU

This section shows operation flow of VSPD and VSPDL by using Normal Display List Mode. For details of Normal Display List Mode, refer to section 33.3.3.2.

Figure 33.64 to Figure 33.66 shows a procedure (register setting flow) to display image data to display panel. Setting not only VSPD or VSPDL registers, but also FCPV/ DU registers are needed. So, see section 35 FCP User's Manual / section 36 DU User's Manual to know the contents of FCPV/ DU registers mentioned in Figure 33.64 and Figure 33.66.

Figure 33.64 shows setting flow to start VSPD/LIF0 linked with DU. Figure 33.65 shows setting flow to update display image. Figure 33.66 shows setting flow to stop VSPD/LIF0 linked with DU.

Using display list with enabling next_frame_auto_start shown in section 33.3.3 is required to take the flow from Figure 33.64 to Figure 33.66.

Same flow can be applied for VSPDL/LIF0 and VSPDL/LIF1. Correspondence between ch number of VSPD/LIF and ch number of DU is shown in Table 33.6. VSPDL/LIF0 and VSPDL/LIF1 can be operated in parallel.

Note: If display sync operation ([1-8] step in Figure 33.64) is set before activating the VSPD/VSPDL, background color specified in DU register (BPORn or DOORn) is displayed until image data read from external memory is ready to be transferred to DU. In this case, image data from external memory is displayed late by one vsync or more.

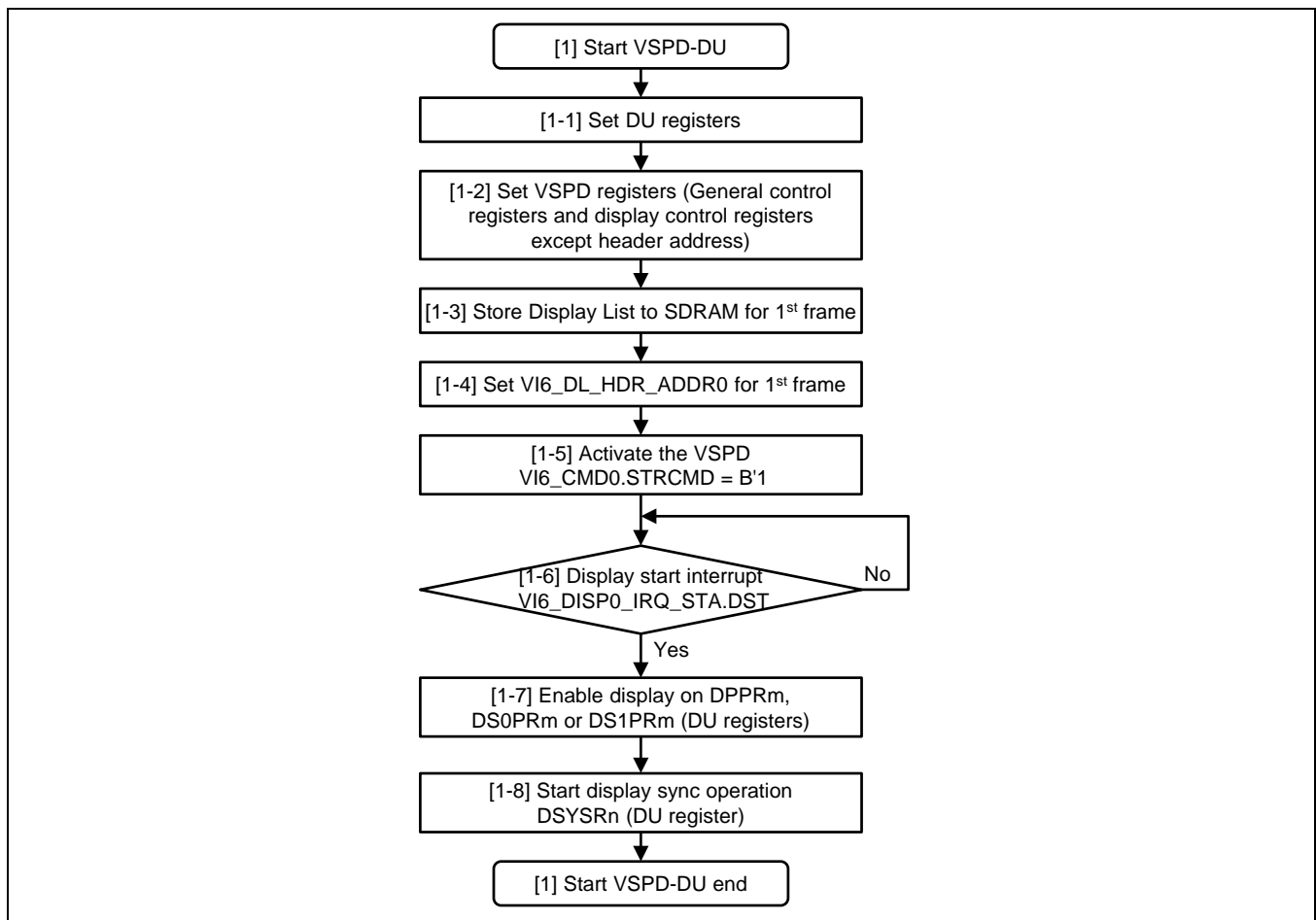


Figure 33.64 Setting flow to start VSPD linked with DU

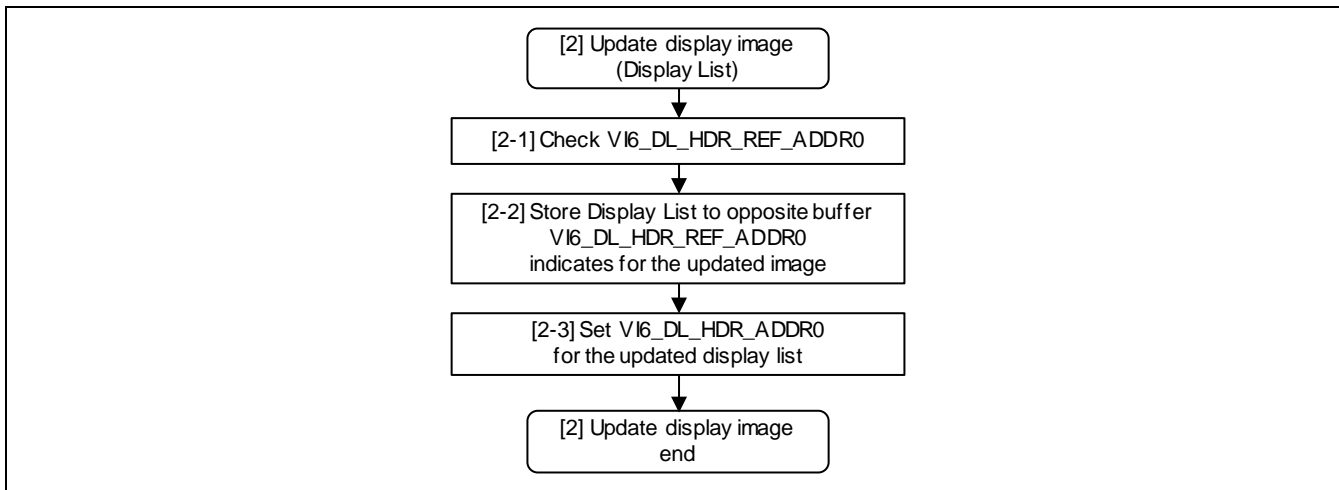
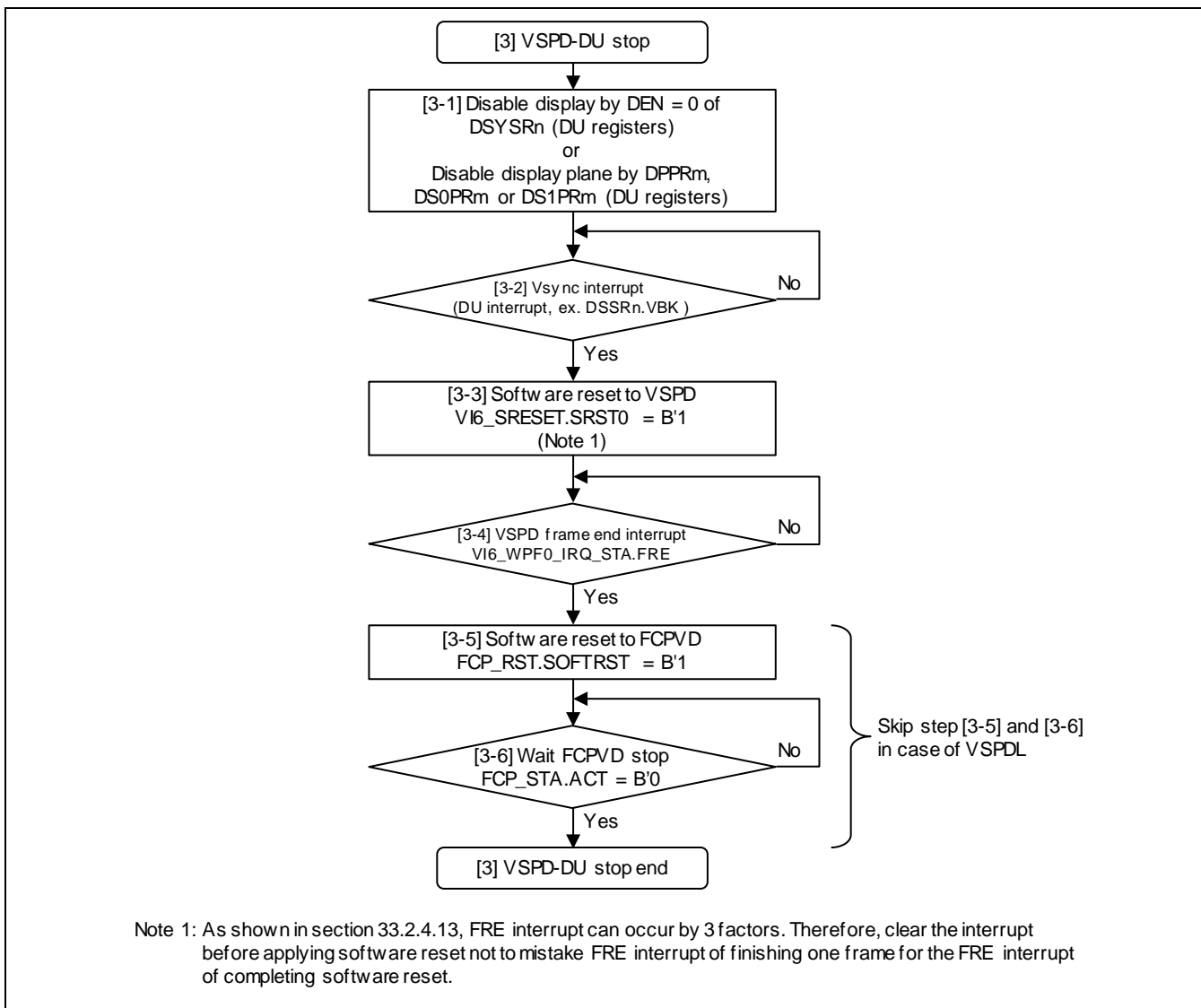


Figure 33.65 Setting flow to update display image



Note 1: As shown in section 33.2.4.13, FRE interrupt can occur by 3 factors. Therefore, clear the interrupt before applying software reset not to mistake FRE interrupt of finishing one frame for the FRE interrupt of completing software reset.

Figure 33.66 Setting flow to stop VSPD linked with DU

Note1: As shown in section 33.2.4.13, FRE interrupt can occur by 3 factors. Therefore, clear the interrupt before applying software reset not to mistake FRE interrupt of finishing one frame for the FRE interrupt of completing software reset.

It is possible to update display list at any time (arbitrary timing) as shown in Figure 33.65. Don't overwrite display list in external memory pointed by VI6_DL_HDR_REF_ADDR. It is because VSP2 is reading the display list pointed by VI6_DL_HDR_REF_ADDR or will be reading the display list soon.

Detail explanation is shown below.

As shown in section 33.2.5.6, when VSP2 is reading display list from external memory (Period [A] in Figure 33.67), VI6_DL_HDR_REF_ADDRn indicates header address of the display list referred by VSP2. When VSP2 is not reading display list from external memory (Period [B] in Figure 33.67), VI6_DL_HDR_REF_ADDRn indicates the value of VI6_DL_HDR_ADDRn, and there is possibility that the display list is being read soon by VSP2 at next frame start timing. Therefore, keep display list in external memory pointed by VI6_DL_HDR_REF_ADDR. Detail software sequence to update display list and VSP2-H/W behavior are shown below.

- [1] Store latest display list in external memory area different from the area pointed by VI6_DL_HDR_REF_ADDR, because the area pointed by VI6_DL_HDR_REF_ADDR is being read or is being read soon by VSP2.
- [2] Set header address of new display list into VI6_DL_HDR_ADDRn.
- [3] VSP2 reads display list from the external memory area pointed by VI6_DL_HDR_ADDRn at frame start timing.

Refer also to section 33.2.5.2 and section 33.2.5.6.

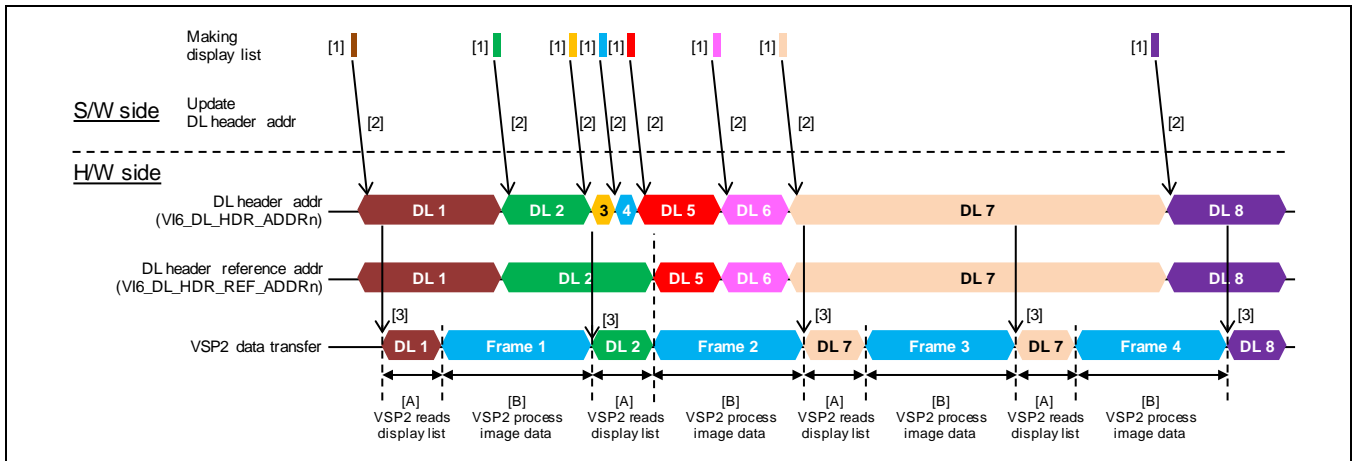


Figure 33.67 Updating display list at arbitrary timing

33.3.6.2 Controlling Two Register Planes Using Display Lists

This section shows operation flow of VSPD and VSPDL (*) by using Header-less Display List Mode. Start procedure and stop procedure are same with Normal Display List Mode in section 33.3.6.1 except for the point that VI6_DL_BODY_SIZE0 should be set before step [1-5] in start procedure shown in Figure 33.64. The procedure to update display list is different with Normal Display List Mode, and is described later. Figure 33.68 shows the control of two register planes using header-less display lists (see section 33.3.3.3) and its timing. In the description hereafter, the use of header-less display lists is always assumed and they are simply called display lists.

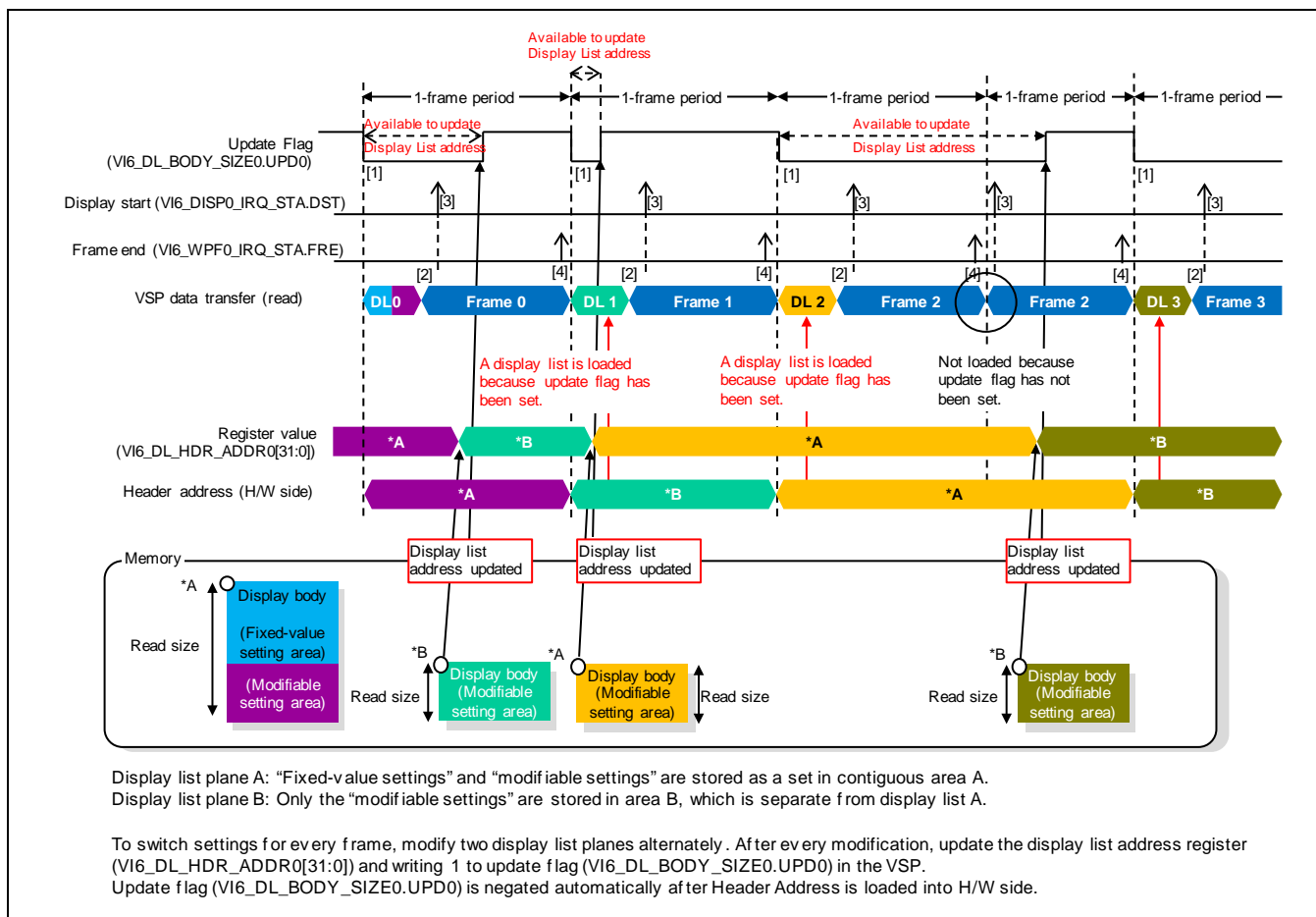


Figure 33.68 Controlling Two Register Planes Using a Display List

The VSP2 downloads a display list immediately after activation. In the start frame, download the display list that contains all necessary settings. From the next frame on, only the necessary register or table values should be specified in a display list.

When the update flag of VI6_DL_BODY_SIZE0.UPD0 is set to 1, VSP2 downloads a new display list at the start of the next frame. When the update flag of VI6_DL_BODY_SIZE0.UPD0 is set to 0, the register settings acquired from the display list previously downloaded are retained and used for the next operation without downloading a new display list.

Order of each VSP2-H/W Events mentioned as [1], [2], [3] and [4] in Figure 33.68 are shown below.

- [1] VI6_DL_BODY_SIZE0.BS0 and VI6_DL_HDR_ADDR0 are downloaded in H/W side. And Update Flag (VI6_DL_BODY_SIZE0.UPD0) is negated automatically.
- [2] Reading DisplayList from external memory into VSP2-H/W is finished through master access.
- [3] Display start interrupt status (VI6_DISP0_IRQ_STA.DST) asserts.
- [4] Frame end interrupt status (VI6_WPF0_IRQ_STA.FRE) asserts.

Don't overwrite the display list in external memory for a period from "[4] VI6_WPF0_IRQ_STA.FRE" to next "[3] VI6_DISPO_IRQ_STA.DST", because VSP is reading the display list for the period.

Note: * Header-less Display List Mode is supported only in LIF0 and is not supported in LIF1. Use Normal Display List Mode for VSPDL/LIF1.

33.3.6.3 Auto display function (AUTO-DISP)

VSPD and VSPDL have AUTO-DISP function. This function allows VSPD/VSPDL to display latest image in triple buffer output from VIN without S/W control of triple buffer. And VSPD/VSPDL selects the field which should be read from TOP or BOTTOM automatically without S/W control. Outline of this function is shown in

Figure 33.69. VSPD/VSPDL replaces source address registers with source address information stored in pre extended display list according to latest buffer information from VIN and field information from DU.

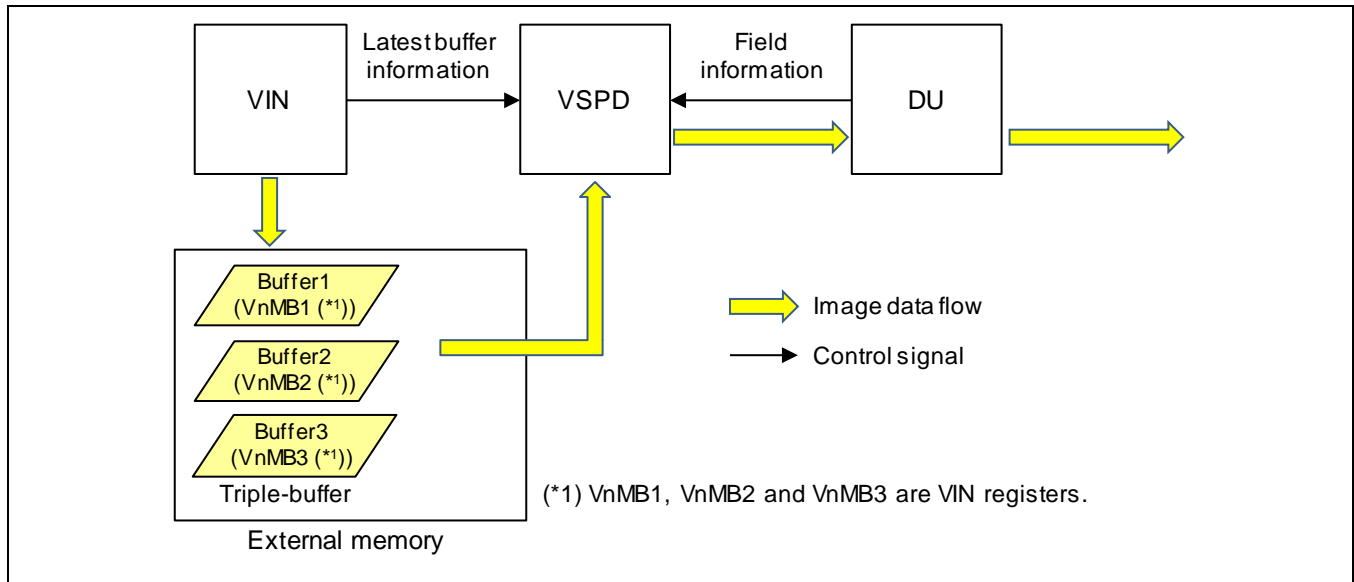


Figure 33.69 Outline of AUTO-DISP

Method to realize this function is shown below.

[1] Use Display List (Normal Display List Mode)

Register setting about display list in case of LIFn AUTO-DISP is shown below.

VI6_DL_CTRL.DLEn = 1

VI6_DL_CTRL.NH0 = 0 [Only for LIF0]

VI6_DL_CTRL.CFM0 = 0 [Only for LIF0]

VI6_DL_CTRL.RLM0 = 0 [Only for LIF0]

VI6_DL_HDR_ADDRn = <Set start address of Display list header>

[2] Use extended display list.

Register setting about extended display list in case of LIFn AUTO-DISP is shown below.

VI6_DL_EXT_CTRLn = H'0000_0221

[3] Contents of display list

Structure of display list to realize AUTO-DISP function is shown in Figure 33.70. Contents of each display list is shown below.

— Display List Header: Table 33.53

— Display List Body: Prepare it in the same way with the case which AUTO-DISP is not used.

— Pre Extended Display List Body: Table 33.54

— Pre Extended Display List Source Address Set: Table 33.55

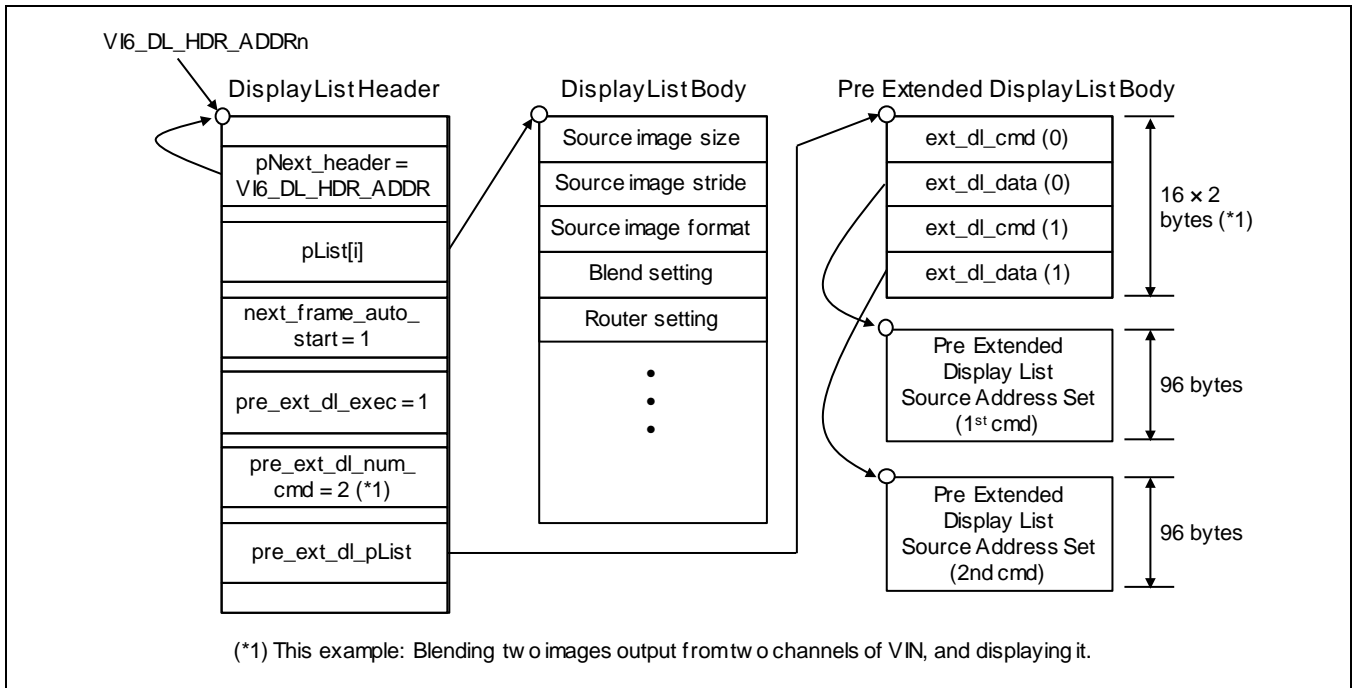


Figure 33.70 Structure of DisplayList of AUTO-DISP

Table 33.53 Display List Header for AUTO-DISP

| Syntax | Value |
|--------------------------|---|
| num_list_minus1 | Set these bits in the same way with the case which AUTO-DISP is not used. |
| num_bytes [i] | Set these bits in the same way with the case which AUTO-DISP is not used. |
| pList [i] | Set these bits in the same way with the case which AUTO-DISP is not used. |
| pNext_header | Set same address of VI6_DL_HDR_ADDRn for progressive display. See Figure 33.72 and Figure 33.73 for interlace display. Be sure to specify an address aligned with a 8-byte boundary (lower-order three bits are 0). |
| current_frame_int_enable | 0 |
| next_frame_auto_start | 1 |
| pre_ext_dl_exec | 1 |
| post_ext_dl_exec | 0 |
| pre_ext_dl_num_cmd | Set the number of connected VIN channels <1 to 5> |
| pre_ext_dl_pList | Set start address of extended display list body Be sure to specify an address aligned with a 16-byte boundary (lower-order four bits are 0). |
| post_ext_dl_num_cmd | 0 |
| post_ext_dl_pList | 0 |

Table 33.54 Pre Extended Display List Body for AUTO-DISP

| Bit String | bits | Symbol | value |
|-------------|------------|--|---|
| ext_dl_cmd | 63 to 40 | — | 0 |
| | 39 to 32 | Opcode | 2 (AUTO_DISP is selected) |
| | 31 to 14 | — | 0 |
| | 13 to 11 | VIN ch no | Set VIN channel number <0 ~ 7> |
| | 10 to 8 | RPF ch no | Set RPF channel number <0 ~ 4> |
| | 7 to 3 | — | 0 |
| | 2 | 1st field | 0: TOP field first (DSMRn.ODEV = 1) 1: BOT field first (DSMRn.ODEV = 0) |
| 1 | Correction | Set 0 for progressive display. See Figure 33.72 and Figure 33.73 for interlace display. | |
| 0 | Int | 0: Progressive display 1: Interlace display | |
| ext_dl_data | 63 to 32 | pAddres_set | Set start address of Source Address Set Be sure to specify an address aligned with a 16-byte boundary (lower-order four bits are 0). |
| | 31 to 0 | Reserved | 0 |

Note: Don't specify RPF channels which are not connected to the LIF in ext_dl_cmd [10:8]. Otherwise, operation of the WPF causes wrong on the other WPF in case of VSPDL.

Table 33.55 Pre Extended Display List Source Address Set for AUTO-DISP

| Address | Contents |
|--------------------------|---------------------------------------|
| ext_dl_data [63:32] + 0 | Buffer1 TOP field address of Y/RGB |
| ext_dl_data [63:32] + 4 | Buffer1 BOTTOM field address of Y/RGB |
| ext_dl_data [63:32] + 8 | Buffer1 TOP field address of C0 |
| ext_dl_data [63:32] + 12 | Buffer1 BOTTOM field address of C0 |
| ext_dl_data [63:32] + 16 | 0 |
| ext_dl_data [63:32] + 20 | 0 |
| ext_dl_data [63:32] + 24 | 0 |
| ext_dl_data [63:32] + 28 | 0 |
| ext_dl_data [63:32] + 32 | buffer2 TOP field address of Y/RGB |
| ext_dl_data [63:32] + 36 | buffer2 BOTTOM field address of Y/RGB |
| ext_dl_data [63:32] + 40 | buffer2 TOP field address of C0 |
| ext_dl_data [63:32] + 44 | buffer2 BOTTOM field address of C0 |
| ext_dl_data [63:32] + 48 | 0 |
| ext_dl_data [63:32] + 52 | 0 |
| ext_dl_data [63:32] + 56 | 0 |
| ext_dl_data [63:32] + 60 | 0 |
| ext_dl_data [63:32] + 64 | buffer3 TOP field address of Y/RGB |
| ext_dl_data [63:32] + 68 | buffer3 BOTTOM field address of Y/RGB |
| ext_dl_data [63:32] + 72 | buffer3 TOP field address of C0 |
| ext_dl_data [63:32] + 76 | buffer3 BOTTOM field address of C0 |
| ext_dl_data [63:32] + 80 | 0 |
| ext_dl_data [63:32] + 84 | 0 |
| ext_dl_data [63:32] + 88 | 0 |
| ext_dl_data [63:32] + 92 | 0 |

How VSPD/VSPDL replaces source address registers is described as below. VSP reads extended display list body specified in Table 33.54 and source address set specified in Table 33.55 during the term of “Ext-DL” specified in Figure 33.71. And VSPD/VSPDL replaces source address registers by step 1 to step 7 as below.

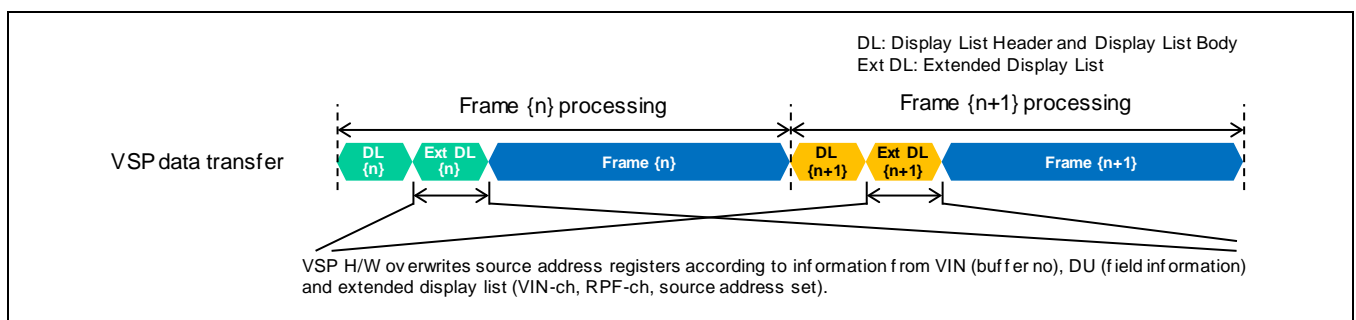


Figure 33.71 Timing to read extended display list

1. VSP reads extended display list body from external memory
2. VSP gets latest buffer information from VINs.
Note: information is updated at only 1st field in case of interlace.
3. VSP gets field information from DU.
4. VSP reads address information of the latest buffer of the VIN `ext_dl_cmd` [13:11] indicates from source address set specified in Table 33.55.
5. VSP selects source address from TOP (*) address or BOTTOM (*) address in source address set according to `ext_dl_cmd` [2:0].
Note: Set start address of progressive image in both TOP address bits and BOTTOM address bits in extended display list source address set in case of progressive display.
6. VSP overwrites source address registers for the RPF specified in `ext_dl_cmd` [10:8] at source address read from external memory.
Note: C1 address and Alpha address registers for the RPF are overwritten by 0 value for the RPF specified in `ext_dl_cmd` [10:8].
7. VSP repeats step 4 to step 6 for number of times of `pre_ext_dl_num_cmd`.

Note: * In this document, TOP field is used as field image including 0th, 2nd, 4th ... line. BOTTOM field is used as field image including 1st, 2nd, 5th ... line.

[4] Progressive and Interlace

There are restrictions about combination of interlace and progressive for input/output as shown in Table 33.56. When VIN captures interlace image, DU should display image in interlace method. When VIN captures progressive image, DU should display image in progressive method.

[4-1] Particular restriction for VIN registers

- Set full interlace mode in VIN register (`VnMC.IM[1:0] = 3`) in case of interlace display.
- No particular restriction is required in case of progressive display.

[4-2] Particular restriction for DU registers

- No particular restriction is required

[4-3] Particular setting and status of VSP

- Set `<2 x source stride>` to source stride register (`VI6_RPFn_SRCM_PSTRIDE`) in case of interlace display.
- `VI6_STATUS.FLDSTn` in section 33.2.4.11 indicates field information of last displayed frame.

[4-4] The number of Display List Header and Pre Extended Display List body

One display list header and one pre extended display list body is needed for progressive display as shown in Figure 33.70. All frames can be controlled by one display list. Set same value of `VI6_DL_HDR_ADDRn` to `pNext_haeder`.

Field signal from DU is not stable in first two fields. Therefore, to display first two fields by appropriate field (TOP or BOTTOM), set bit1 in `ext_dl_cmd` as shown in Figure 33.72 for TOP field first and Figure 33.73 for BOTTOM field first. Three display list headers and three pre extended display list bodies are needed to adjust first two fields for interlace display.

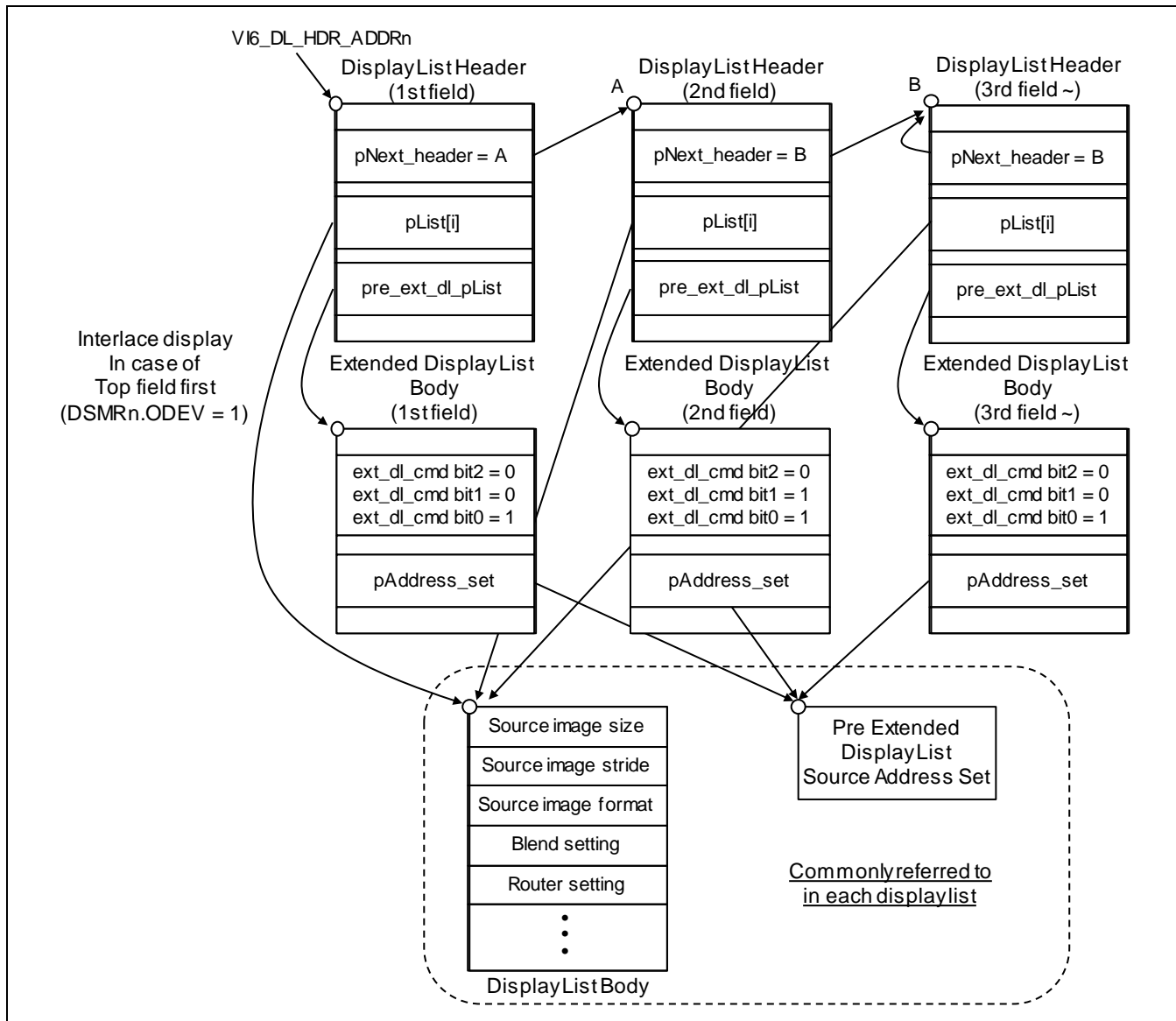


Figure 33.72 Correcting first two fields in case of TOP field first

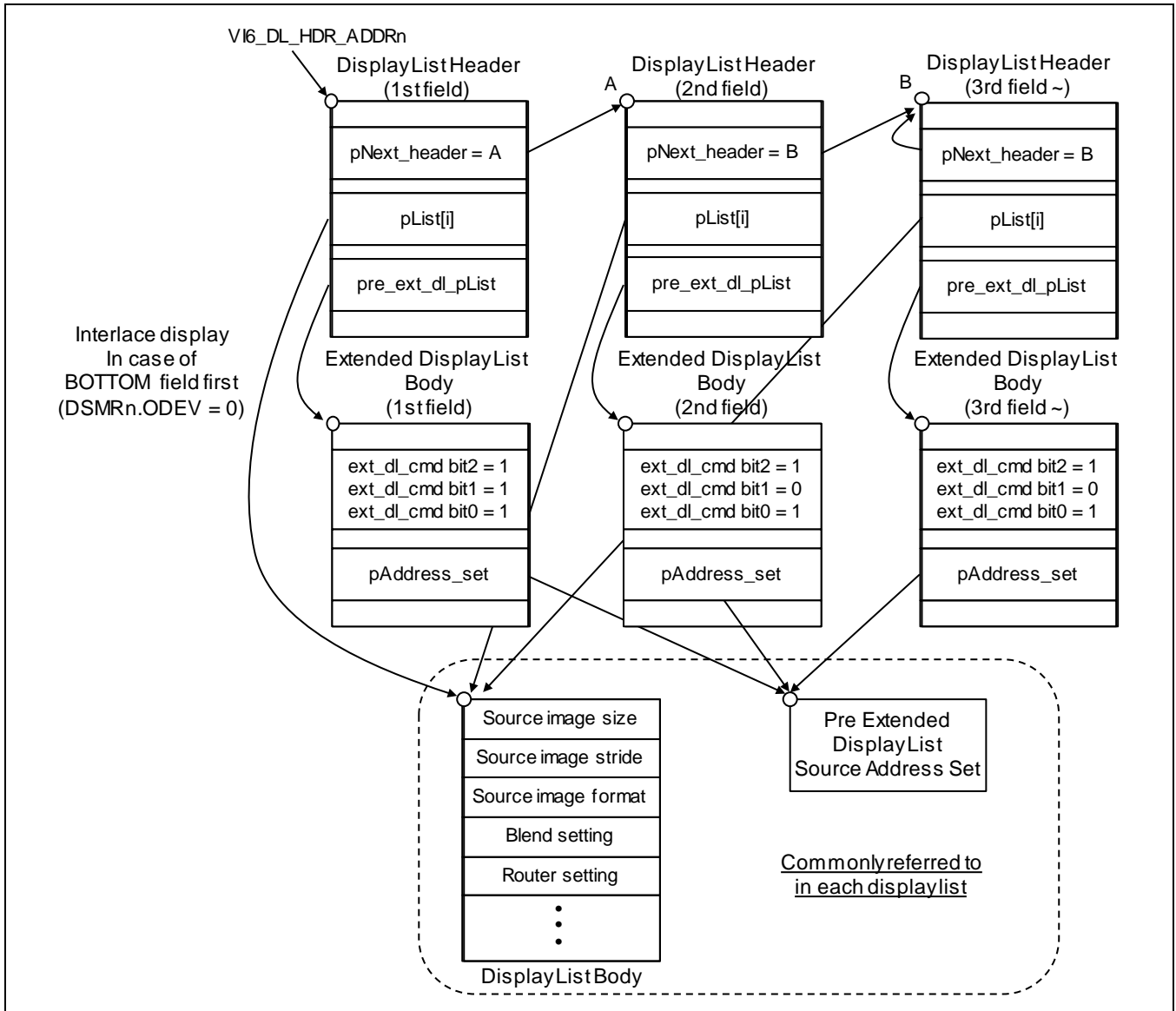


Figure 33.73 Correcting first two fields in case of BOTTOM field first

[5] Startup image before first image is output from VIN

VSP reads image data in buffer3 address specified in extended display list source address set until VIN completes writing first frame's image into buffer1. Therefore, prepare startup image in buffer3. This image is overwritten after VIN is activated.

[6] Procedure of activation

Activate VIN with continuous frame capture mode (VnFC.CC = 1). After that, activate VSPD and DU by procedure of Figure 33.64

[7] Restrictions

There are several restrictions as shown below.

[7-1] Restriction about combination of interlace and progressive for input/output

There are restriction about combination of interlace and progressive for input/output as shown in Table 33.56.

Table 33.56 Restriction about combination of interlace and progressive for input/output

| Video IN | Display | Support |
|-------------|-------------|---------|
| Progressive | Progressive | √ |
| Progressive | Interlace | |
| Interlace | Progressive | |
| Interlace | Interlace | √ |

[7-2] Restriction about frame rate of input and output

Frame rate of VIN should be equal to or smaller than frame rate of display.

[7-3] Restriction about selecting VIN channel.

- (a) For RZ/G2M V1.3, RZ/G2M V3.0, there are following restrictions about connection between VIN and VSPD.
- VSPD0 can read only VIN0 output image when this function is used.
 - VSPD1 can read only VIN2 output image when this function is used.
 - VSPD2 can read only VIN4 output image when this function is used.
- (b) For RZ/G2H, RZ/G2N and RZ/G2E, there is no restriction about selecting VIN channel.

33.3.6.4 Auto field function (AUTO-FLD)

VSPD and VSPDL have AUTO-FLD function. VSPD/VSPDL selects the field which should be read from TOP or BOTTOM automatically without S/W control. Outline of this function is shown in Figure 33.74. VSPD/VSPDL replaces source address registers with source address information stored in pre extended display list according to field information from DU.

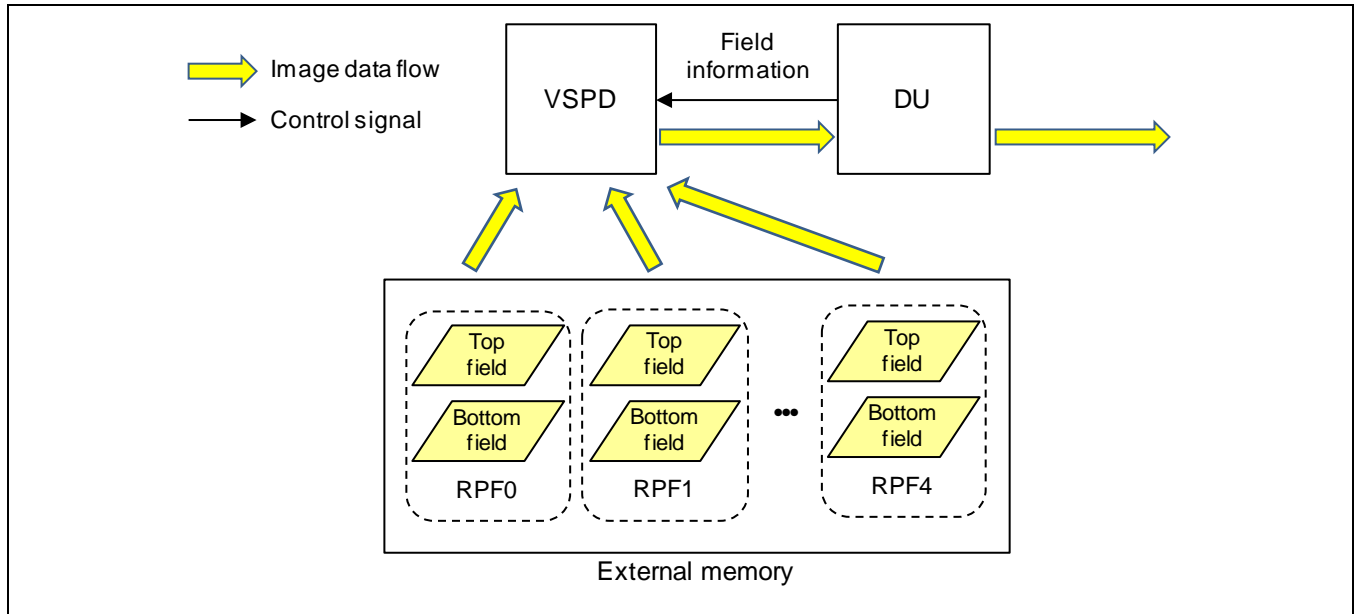


Figure 33.74 Outline of AUTO-FLD

Method to realize this function is shown below.

[1] Use Display List (Normal Display List Mode)

Take same setting with AUTO-DISP shown in 33.3.6.3.

[2] Use extended display list.

Take same setting with AUTO-DISP shown in 33.3.6.3.

[3] Contents of display list for AUTO-FLD

Structure of display list to realize AUTO-FLD function is shown in Figure 33.75. Contents of each display list is shown below.

- Display List Header: Table 33.57
- Display List Body: Prepare it in the same way with the case which AUTO-FLD is not used.
- Pre Extended Display List Body: Table 33.58
- Pre Extended Display List Source Address Set: Table 33.59

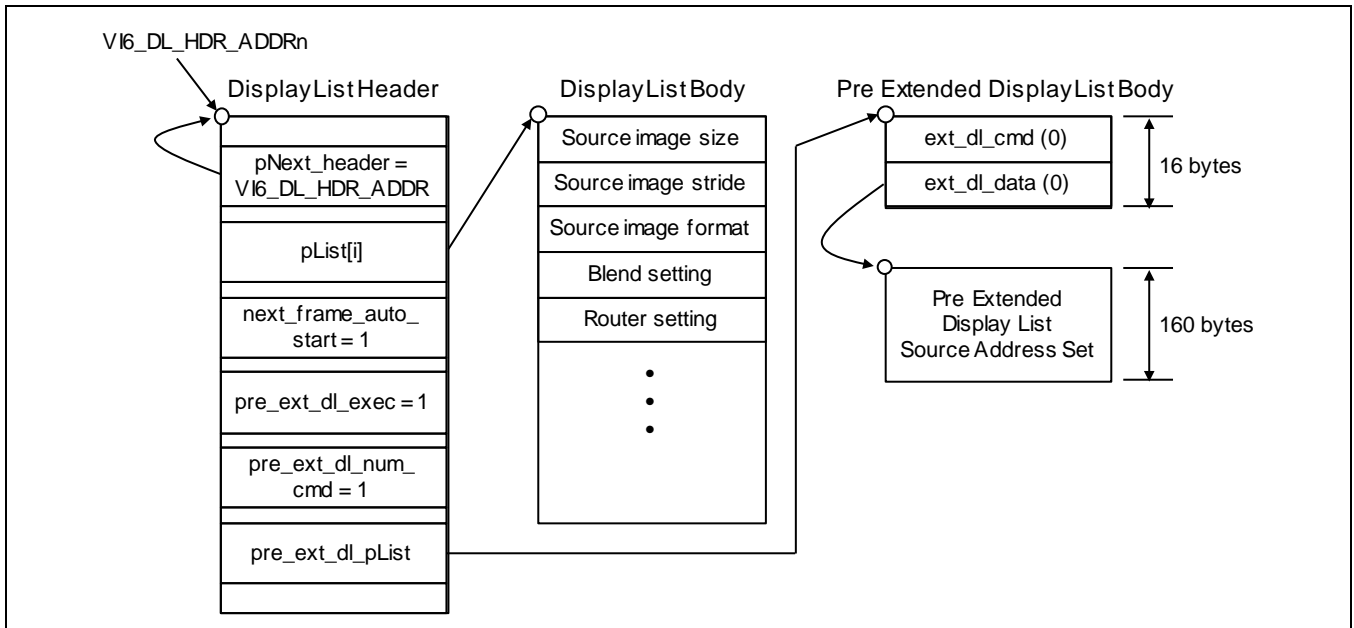


Figure 33.75 Structure of DisplayList of AUTO-FLD

Table 33.57 Display List Header for AUTO-FLD

| Syntax | Value |
|--------------------------|---|
| num_list_minus1 | Set these bits in the same way with the case which AUTO-FLD is not used. |
| num_bytes [i] | Set these bits in the same way with the case which AUTO-FLD is not used. |
| pList [i] | Set this bits in the same way with the case which AUTO-FLD is not used. |
| pNext_header | Set same address of VI6_DL_HDR_ADDRn for progressive display. See Figure 33.72 and Figure 33.73 for interlace display. Be sure to specify an address aligned with a 8-byte boundary (lower-order three bits are 0). |
| current_frame_int_enable | 0 |
| next_frame_auto_start | 1 |
| pre_ext_dl_exec | 1 |
| post_ext_dl_exec | 0 |
| pre_ext_dl_num_cmd | 1 |
| pre_ext_dl_pList | Set start address of extended display list body Be sure to specify an address aligned with a 16-byte boundary (lower-order four bits are 0). |
| post_ext_dl_num_cmd | 0 |
| post_ext_dl_pList | 0 |

Table 33.58 Pre Extended Display List Body for AUTO-FLD

| Bit String | bits | Symbol | value |
|-------------|----------|-------------|---|
| ext_dl_cmd | 63 to 40 | — | 0 |
| | 39 to 32 | Opcode | 3 (AUTO_FLD is selected) |
| | 31 to 21 | — | 0 |
| | 20 | RPF4 enable | 0: Not replace RPF4 source address registers. 1: Replace RPF4 source address registers. |
| | 19 | RPF3 enable | 0: Not replace RPF3 source address registers. 1: Replace RPF3 source address registers. |
| | 18 | RPF2 enable | 0: Not replace RPF2 source address registers. 1: Replace RPF2 source address registers. |
| | 17 | RPF1 enable | 0: Not replace RPF1 source address registers. 1: Replace RPF1 source address registers. |
| | 16 | RPF0 enable | 0: Not replace RPF0 source address registers. 1: Replace RPF0 source address registers. |
| | 15 to 2 | — | 0 |
| | 1 | Correction | Set 0 for progressive display. See Figure 33.72 and Figure 33.73 for interlace display. |
| 0 | Int | 1 | |
| ext_dl_data | 63 to 32 | pAddres_set | Set start address of Source Address Set Be sure to specify an address aligned with a 16-byte boundary (lower-order four bits are 0). |
| | 31 to 0 | Reserved | 0 |

Note: For RPFn enable in pre extended display list body shown in Table 33.58, don't enable RPF channels which are not connected to the LIF. Otherwise, operation of the WPF causes wrong on the other WPF in case of VSPDL.

Example. When RPF0, RPF1 and RPF2 are connected to WPF0/LIF0, and RPF3 and RPF4 are connected to WPF1/LIF1,

- Disable RPF3 and RPF4 (ext_dl_cmd [20:16] = 5'b00111) for WPF0 display list.
- Disable RPF0, RPF1 and RPF2 (ext_dl_cmd [20:16] = 5'b11000) for WPF1 display list.

Table 33.59 Pre Extended Display List Source Address Set for AUTO-FLD

| Address | Contents |
|---------------------------|------------------------------------|
| ext_dl_data [63:32] + 0 | RPF0 TOP field address of Y/RGB |
| ext_dl_data [63:32] + 4 | RPF0 BOTTOM field address of Y/RGB |
| ext_dl_data [63:32] + 8 | RPF0 TOP field address of C0 |
| ext_dl_data [63:32] + 12 | RPF0 BOTTOM field address of C0 |
| ext_dl_data [63:32] + 16 | RPF0 TOP field address of C1 |
| ext_dl_data [63:32] + 20 | RPF0 BOTTOM field address of C1 |
| ext_dl_data [63:32] + 24 | 0 |
| ext_dl_data [63:32] + 28 | 0 |
| ext_dl_data [63:32] + 32 | RPF1 TOP field address of Y/RGB |
| ext_dl_data [63:32] + 36 | RPF1 BOTTOM field address of Y/RGB |
| ext_dl_data [63:32] + 40 | RPF1 TOP field address of C0 |
| ext_dl_data [63:32] + 44 | RPF1 BOTTOM field address of C0 |
| ext_dl_data [63:32] + 48 | RPF1 TOP field address of C1 |
| ext_dl_data [63:32] + 52 | RPF1 BOTTOM field address of C1 |
| ext_dl_data [63:32] + 56 | 0 |
| ext_dl_data [63:32] + 60 | 0 |
| ext_dl_data [63:32] + 64 | RPF2 TOP field address of Y/RGB |
| ext_dl_data [63:32] + 68 | RPF2 BOTTOM field address of Y/RGB |
| ext_dl_data [63:32] + 72 | RPF2 TOP field address of C0 |
| ext_dl_data [63:32] + 76 | RPF2 BOTTOM field address of C0 |
| ext_dl_data [63:32] + 80 | RPF2 TOP field address of C1 |
| ext_dl_data [63:32] + 84 | RPF2 BOTTOM field address of C1 |
| ext_dl_data [63:32] + 88 | 0 |
| ext_dl_data [63:32] + 92 | 0 |
| ext_dl_data [63:32] + 96 | RPF3 TOP field address of Y/RGB |
| ext_dl_data [63:32] + 100 | RPF3 BOTTOM field address of Y/RGB |
| ext_dl_data [63:32] + 104 | RPF3 TOP field address of C0 |
| ext_dl_data [63:32] + 108 | RPF3 BOTTOM field address of C0 |
| ext_dl_data [63:32] + 112 | RPF3 TOP field address of C1 |
| ext_dl_data [63:32] + 116 | RPF3 BOTTOM field address of C1 |
| ext_dl_data [63:32] + 120 | 0 |
| ext_dl_data [63:32] + 124 | 0 |
| ext_dl_data [63:32] + 128 | RPF4 TOP field address of Y/RGB |
| ext_dl_data [63:32] + 132 | RPF4 BOTTOM field address of Y/RGB |
| ext_dl_data [63:32] + 136 | RPF4 TOP field address of C0 |
| ext_dl_data [63:32] + 140 | RPF4 BOTTOM field address of C0 |
| ext_dl_data [63:32] + 144 | RPF4 TOP field address of C1 |
| ext_dl_data [63:32] + 148 | RPF4 BOTTOM field address of C1 |
| ext_dl_data [63:32] + 152 | 0 |
| ext_dl_data [63:32] + 156 | 0 |

How VSPD/VSPDL replaces source address registers is described as below. VSP reads extended display list body specified in Table 33.58 and source address set specified in Table 33.59 during the term of “Ext-DL” specified in Figure 33.71. And VSPD/VSPDL replaces source address registers by step 1 to step 5 as below.

1. VSP reads extended display list body from external memory.
2. VSP gets field information from DU.
3. VSP reads address information of RPFs which `ext_dl_cmd [20:16]` indicates from source address set.
4. VSP selects source address from TOP address or BOTTOM address in source address set according to `ext_dl_cmd [2:0]`
5. VSP overwrites source address registers for the RPF specified in `ext_dl_cmd [20:16]` at source address read from external memory.

Note: Alpha address registers for the RPF are overwritten by 0 value for the RPF specified in `ext_dl_cmd [20:16]`.

[4] Interlace

[4-1] Status of VSP

- `VI6_STATUS.FLDSTn` in section 33.2.4.11 indicates field information of last displayed frame.

[4-2] The number of Display List Header and Pre Extended Display List body

Field signal from DU is not stable in first two fields. Therefore, to display first two fields by appropriate field (TOP or BOTTOM), set bit1 in `ext_dl_cmd` as shown in Figure 33.72 for TOP field first and Figure 33.73 for BOTTOM field first. Three display list headers and three pre extended display list bodies are needed to adjust first two fields for interlace display.

Note: For `AUTO_FLD`, set 0 to `ext_dl_cmd` bit2 regardless of that first field is TOP or BOTTOM.

[5] Procedure of activation

Activate VSPD and DU by procedure of Figure 33.64

[6] Restrictions

There is no restriction about `AUTO-FLD`.

33.3.7 Image partition for VSPI processing

33.3.7.1 Necessity of image partition

Because SRU, UDS and WPF/Rotation ($VI6_WPF0_OUTFMT.ROT > 1$) need to satisfy horizontal size restriction as shown in Table 33.60. Several executions of VSPI by partitioning one frame's image is needed to realize larger size in horizontal direction. In the chapter, the method is called as image partition. This section shows how to realize image partition for VSPI.

When either restriction shown in Table 33.60 applies, image partition method is needed.

Table 33.60 Condition of necessity to take image partition

| Location of restriction | Max horizontal size | Condition of necessity to take the restriction. |
|---|---------------------|--|
| Input to UDS after clipping horizontal size by VI6_UDS_HSZCLIP register | 304 pixels | UDS is used by up-scaling in horizontal direction. |
| Output from UDS after clipping horizontal size by VI6_UDS_CLIP_SIZE | 304 pixels | UDS is used by down-scaling or no scaling in horizontal direction. |
| Input to SRU | 288 pixels | SRU is used. |
| Input to WPF after clipping horizontal size by VI6_WPF0_HSZCLIP | 256 pixels | WPF/rotation ($VI6_WPF0_OUTFMT.ROT > 1$) is used. |

To realize image partition, see five items below.

- (1) Auto execution of VSPI by using automatic start of next frame processing (`next_frame_auto_start`) of DisplayList is recommended. See section 33.3.7.2 for more detail.
- (2) Start reservation is available to reduce overhead of interrupt response. See section 33.3.7.3 to use start reservation.
- (3) Discontinuous line is generated by VSPI end processing at partition boundary. So, reading overlapped pixels with neighbor partition and clipping discontinuous line at partition boundary is needed. Figure 33.76 shows example of overlap pixel and clipping in case of 3 partition. See section 33.3.7.4 for more detail.
- (4) The number of taps for each function's filter in horizontal direction is shown in section 33.3.7.5 for user to realize how many overlapped pixels to be read and how many pixels to be clipped.
- (5) Horizontal size clipping registers are explained in section 33.3.7.6.
- (6) How to set HGO and HGT registers related to image partition is shown in section 33.3.7.7.
- (7) There are some restrictions when VSPI image partition is used. See section 33.3.7.8.

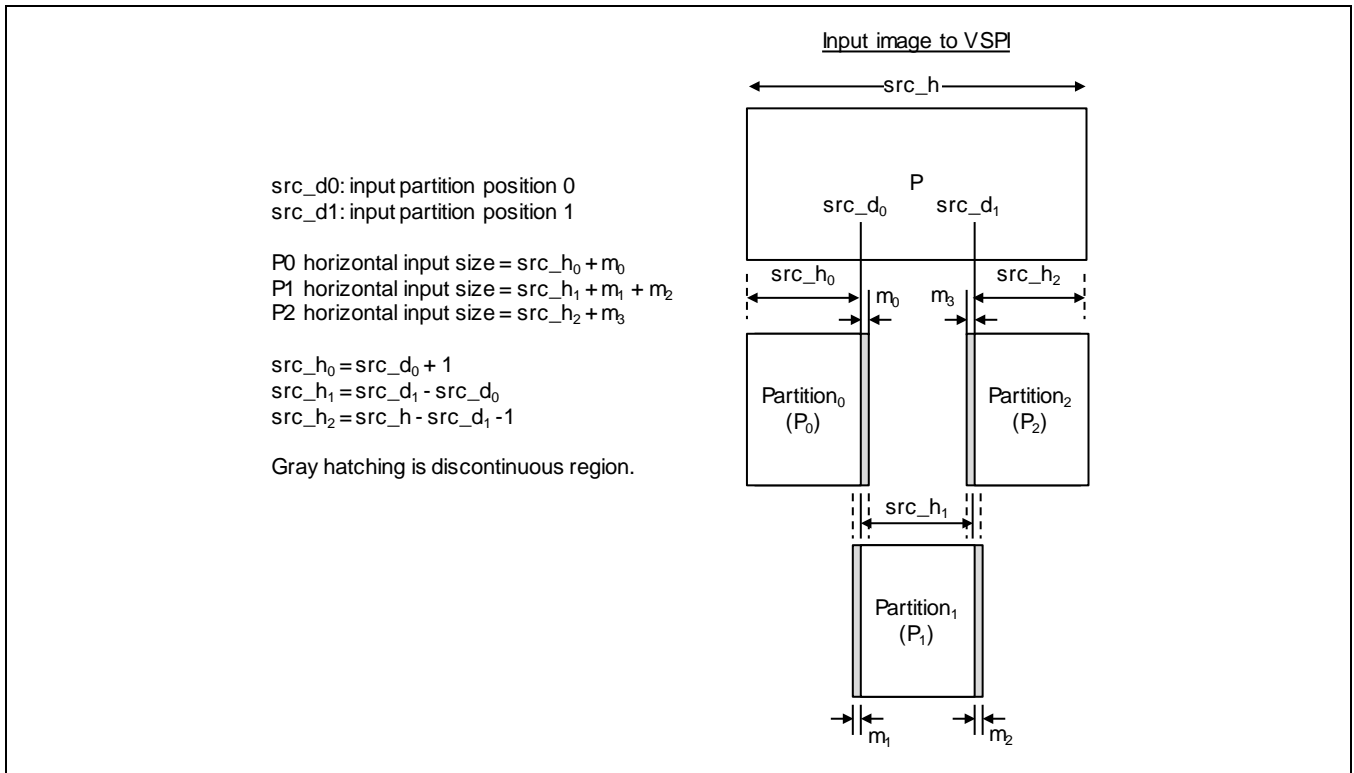


Figure 33.76 Example of overlap pixel and clipping in case of 3 partitions

33.3.7.2 Auto execution by using display list

Because partitioning one frame image into several partitioned images, several activations of VSPI for one frame processing are needed. So, using display list and auto execution by next frame auto start is recommended. Detail way is shown below.

- (1) Store all partitions' display list in SDRAM as shown in Figure 33.77.
- (2) In order to get better performance, don't set registers which are not updated per partition in display body from 2nd partition to last partition as shown in Figure 33.77. Registers which need to be updated per partition is listed in Table 33.61.
- (3) Set 1 to next_frame_auto_start in display list header from 1st partition to 2nd partition from the last. Then S/W has to activate VSPI only once for 1st partition.
- (4) Set 1 to current_frame_int_enable in display list header for only last partition. Set H'0000_0002 to VI6_WPF0_IRQ_ENB to enable Display List Frame End (DFE) interrupt and to disable Frame End (FRE) interrupt. Then DFE interrupt occurs after all partitions' processing is finished by VSPI-H/W.
- (5) Set below to VI6_DL_CTRL
 - VI6_DL_CTRL.DLE0 = 1 (Use display list)
 - VI6_DL_CTRL.NH0 = 0 (Use Header Mode)
 - VI6_DL_CTRL.RLM0 = 1 (Registers with two planes are not downloaded into H/W side at partition boundary)

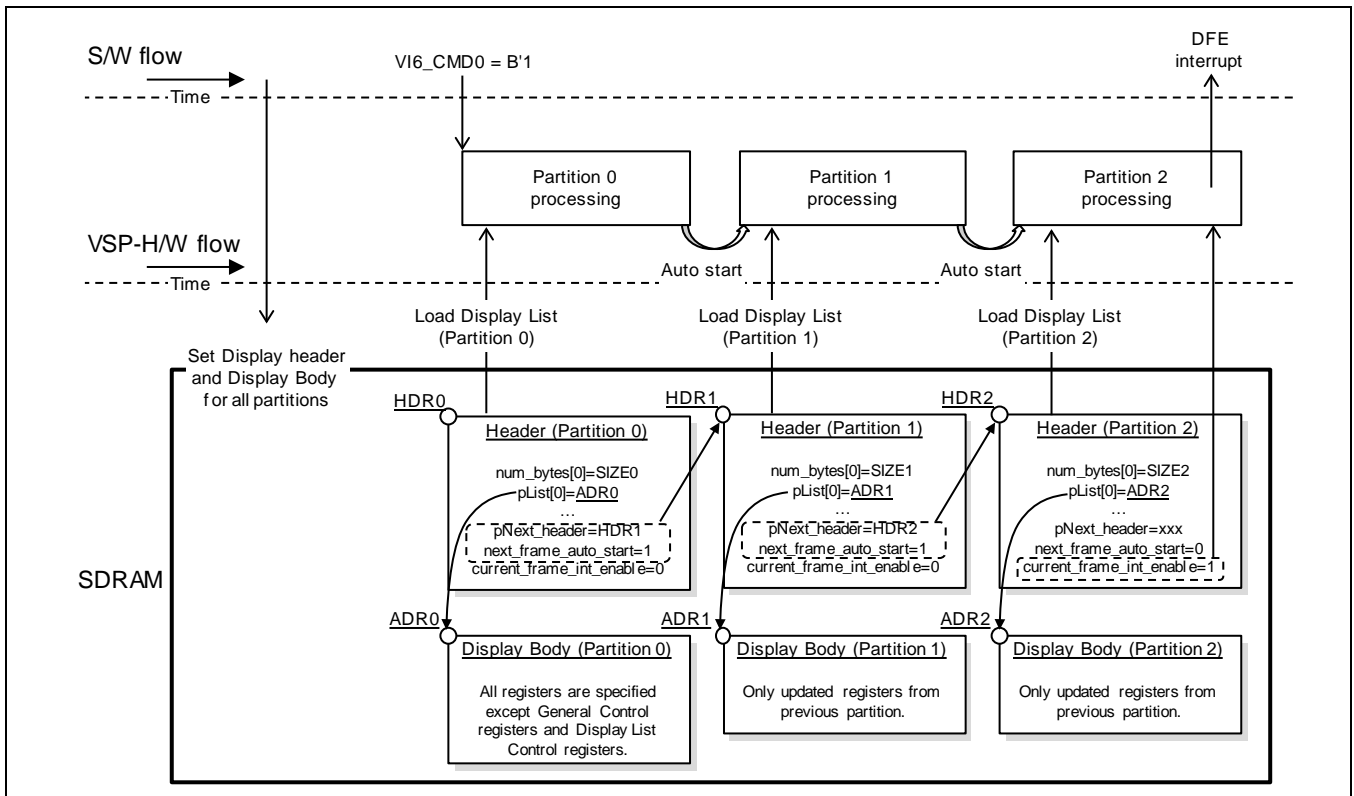


Figure 33.77 Flow of image partition process (Example of 3 partitions case)

Table 33.61 Registers which need to be updated per partition

| | |
|----|-----------------------|
| 1 | VI6_RPF0_SRC_BSIZE |
| 2 | VI6_RPF0_SRC_ESIZE |
| 3 | VI6_RPF0_SRCM_ADDR_Y |
| 4 | VI6_RPF0_SRCM_ADDR_C0 |
| 5 | VI6_RPF0_SRCM_ADDR_C1 |
| 6 | VI6_RPF0_SRCM_ADDR_AI |
| 7 | VI6_WPF0_HSZCLIP |
| 8 | VI6_WPF0_DSTM_ADDR_Y |
| 9 | VI6_WPF0_DSTM_ADDR_C0 |
| 10 | VI6_WPF0_DSTM_ADDR_C1 |
| 11 | VI6_UDS_HPHASE |
| 12 | VI6_UDS_HSZCLIP |
| 13 | VI6_UDS_CLIP_SIZE |
| 14 | VI6_HGO_MODE |
| 15 | VI6_HGO_OFFSET |
| 16 | VI6_HGO_SIZE |
| 17 | VI6_HGO_REGRST |
| 18 | VI6_HGT_MODE |
| 19 | VI6_HGT_OFFSET |
| 20 | VI6_HGT_SIZE |
| 21 | VI6_HGT_REGRST |
| 22 | VI6_DPR_HGO_SMPPT |
| 23 | VI6_DPR_HGT_SMPPT |

33.3.7.3 Start Reservation for VSPI

This section shows sequence of software when start reservation is used. Software can reserve starting VSPI before VSPI processing of one frame is completed. Start reservation can remove interrupt response time and software overhead by reserving execution of VSPI before completion of previous frame.

(a) below shows software sequence of start reservation.

(b) below shows software sequence to manage the number of interrupt generation.

Note: Follow section 33.3.7.2 about the setting of display list and VI6_WPF0_IRQ_ENB.

(a) Software sequence of start reservation

[1] ~ [9] shown in Figure 33.78 and Figure 33.79 shows S/W flow to set registers for start reservation of VSPI

[1] Wait until 0 is read from VI6_CMD0.STRCMD to confirm that starting WPF0 is not reserved.

[2] Wait until 0 is read from VI6_STATUS.SYS0_ACT to confirm WPF0 is IDLE.

[3] Set registers with two planes for frame 0 (1st frame) except DLH (Display List Header) address.

VI6_DL_HDR_ADDR0, VI6_HGO_WBUFS and VI6_HGT_WBUFS has two planes.

[4] Set header address of display list for frame 0 to VI6_DL_HDR_ADDR0. Then VI6_CMD0.UPDHDR is asserted automatically by VSP2 to indicate downloading registers with two planes is reserved.

[5] Activate WPF0 for frame 0 (1st frame) by writing H'0000_0001 to VI6_CMD0.

If next frame's execution of VSPI is needed, go to step [6].

[6] Wait until start reservation is accepted and registers with two planes are downloaded into H/W side. If 0 is read from VI6_CMD0, it means that starting WPF0 is accepted and register with two planes are downloaded into internal registers H/W side refers.

[7] Set registers with two planes for next frame (fr{m}) except DLH (Display List Header) address.

[8] Set header address of display list for next frame (fr{m}) to VI6_DL_HDR_ADDR0. Then VI6_CMD0.UPDHDR is asserted automatically by VSP2 to indicate downloading registers with two planes is reserved.

[9] Reserve activating WPF0 for next frame (fr{m}) by writing H'0000_0001 to VI6_CMD0.

If next frame's execution of VSPI is needed, go to step [6].

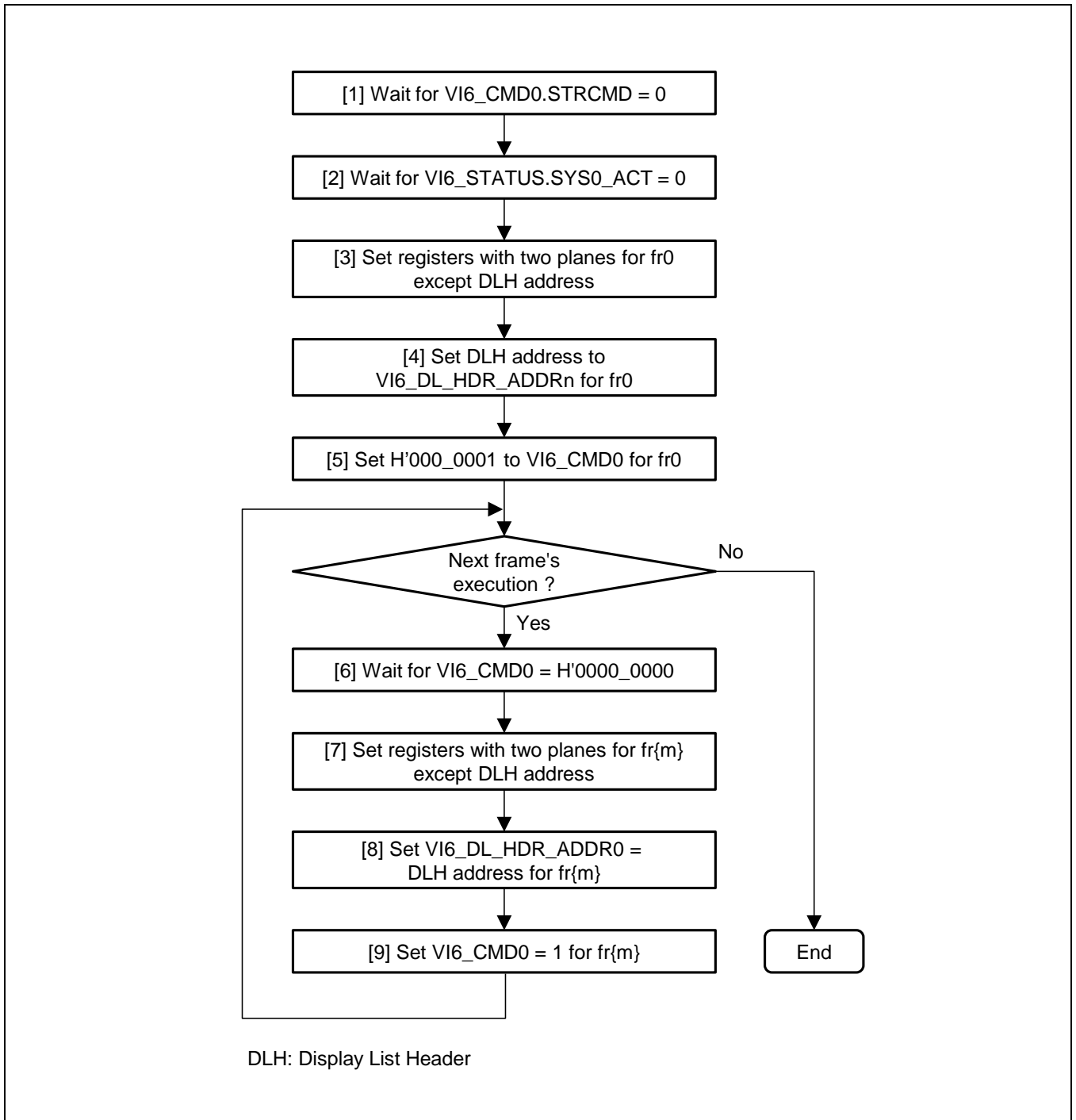


Figure 33.78 Flow chart of setting registers by S/W in case of VSPI start reservation

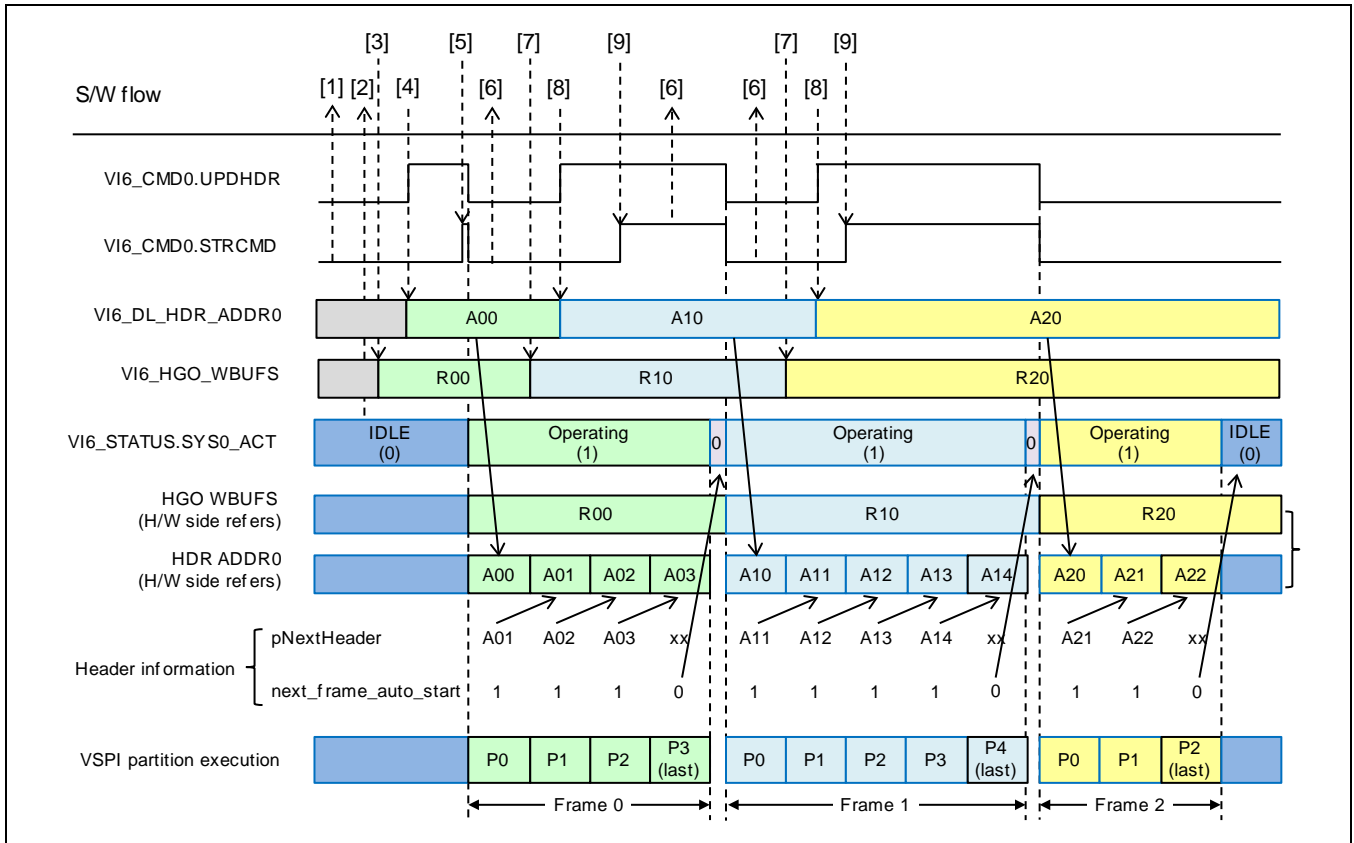


Figure 33.79 Timing of setting registers by S/W in case of VSPI start reservation

(b) Software sequence to manage the number of interrupt generation.

The way of using interrupt is shown below.

- [1] Set 1 to the VI6_WPF0_IRQ_ENB.DFEE to use frame end interrupt (VI6_WPF0_IRQ_STA.DFE)
- [2] Write 0 to VI6_WPF0_IRQ_STA.DFE to clear status after interrupt processing as shown in **Figure 33.80**.

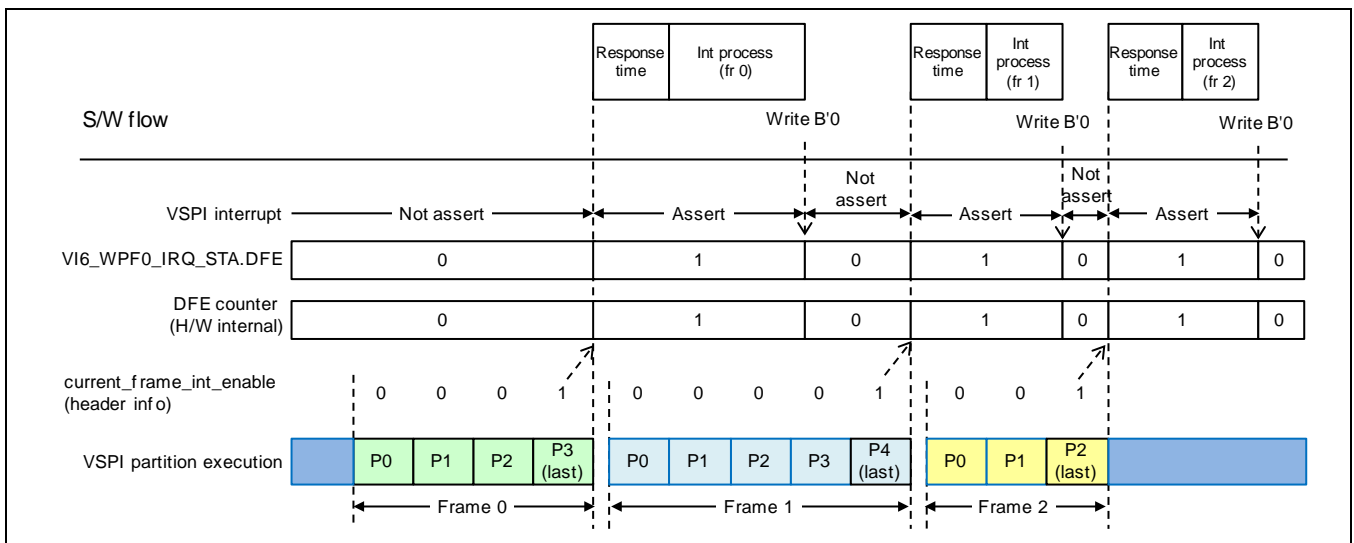


Figure 33.80 Timing of interrupt processing by S/W in case of VSPI start reservation (Case1)

Be careful following points as shown in Figure 33.81.

- [1] Set 0 to VI6_WPF0_IRQ_STA.DFE for the number of interrupt generation to clear VI6_WPF0_IRQ_STA.DFE.
- [2] The VSPI H/W can count at most two of interrupt generation. If three times of interrupts occur before clearing the interrupt factor, interrupt counter overflows. Therefore, wait to reserve starting VSPI of frame {n+2} until interrupt factor of the frame {n} is cleared.

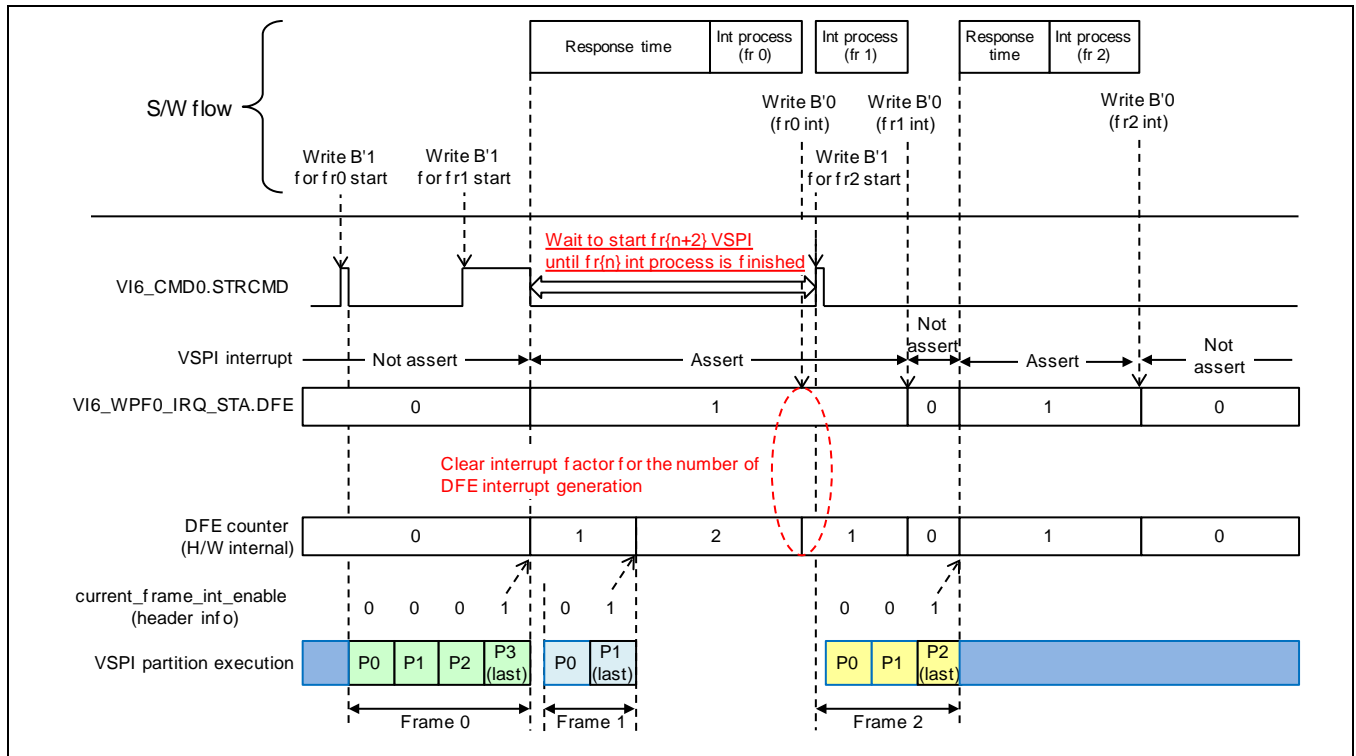


Figure 33.81 Timing of interrupt processing by S/W in case of VSPI start reservation (Case2)

33.3.7.4 Reading overlapped pixels and clipping discontinuous line

The VSPI processes partition boundary as end processing, and discontinuous line is generated at partition boundary. Figure 33.82 shows why reading overlapped pixels and clipping discontinuous line are needed. As example that 3 tap filter is used inside VSPI, right neighbor pixel and left neighbor pixel are referred to generate output pixel. (a), (b) and (c) pixels are referred to generate (b') in left partition. But if the VSPI doesn't read (c), right end pixel (b) is copied inside VSPI. And (b) is referred instead of (c) to generate (b'). And (b') is recognized as discontinuous line. So reading overlapped pixels and clipping discontinuous line are needed.

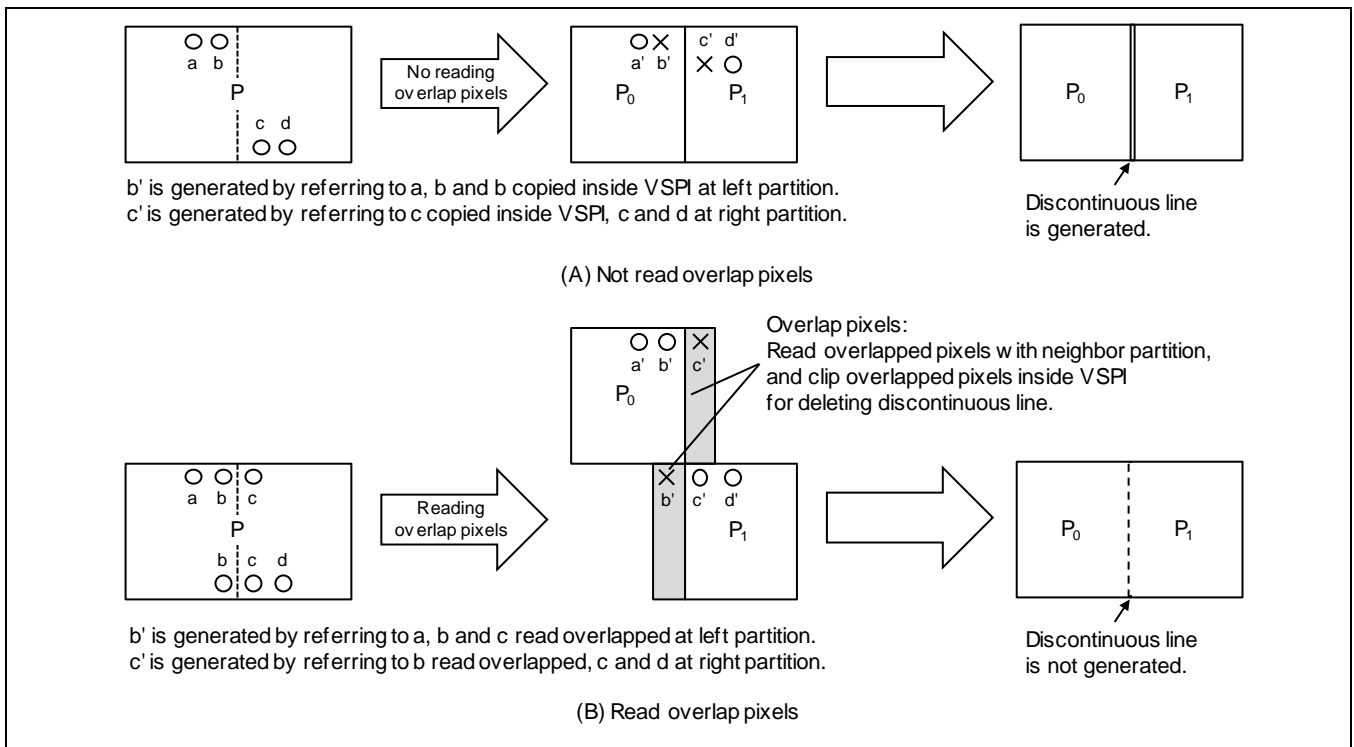


Figure 33.82 Discontinuous line at partition boundary (The example is 3 tap filter)

33.3.7.5 The number of taps for each function's filter in horizontal direction

SRU, SHP, UDS and RPF have filter function which refer to not only current position's pixel but also several neighbor pixels to generate current position's output pixel. The number of taps for each function's filter are shown in this section to realize how many overlapped pixels to be read and how many pixels to be clipped.

(1) RPF (YCbCr42x -> YCbCr444 up sampling for Chroma)

Treatment of Cb and Cr is same as up sampling method. So C is used as Cb or Cr in the explanation.

(1-1) In case of VI6_RPF0_INFMT.CIPM = 0 for YCbCr42x input format

- Duplicate C component twice in horizontal direction.
- Horizontal size setting is restricted as 2-pixels units as shown in Table 33.65

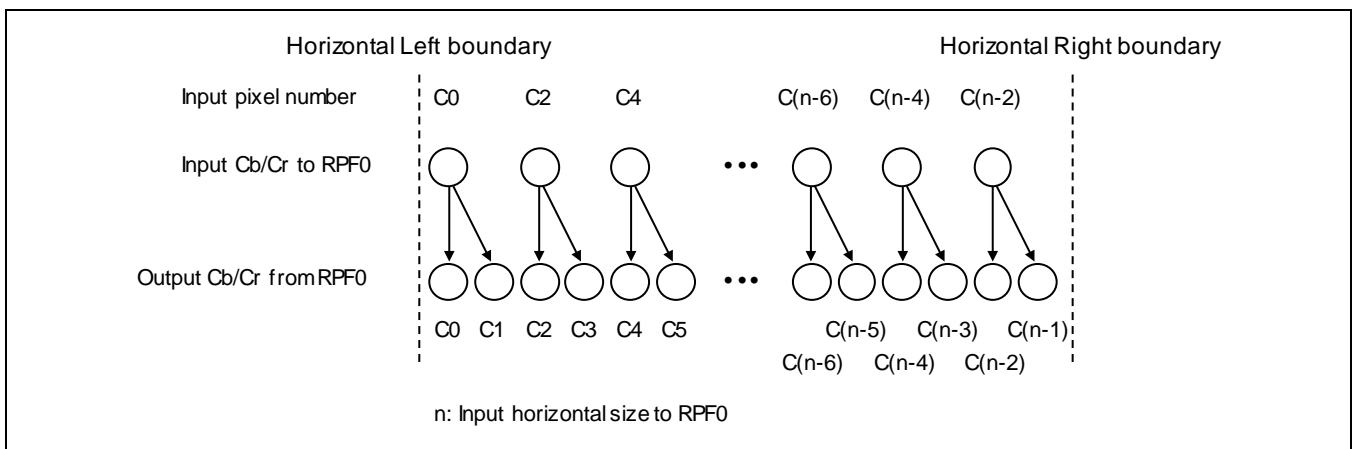


Figure 33.83 Chroma (Cb, Cr) interpolation way in RPF0 (VI6_RPF0_INFMT.CIPM = 0)

(1-2) In case of VI6_RPF0_INFMT.CIPM = 1 for YCbCr42x input format

- Even (0, 2, 4 ...) output pixel of C is generated as copy of input pixel.
- Odd (1, 3, 5 ...) output pixel of C is generated as 2 taps filter.
- Output pixel at right boundary is influenced by copied pixel C (n-2) of input pixel as shown in Figure 33.84. So it needs to clip C (n-1) output pixel for deleting discontinuous line when horizontal partition's right boundary is not frame boundary.
- Horizontal size setting is restricted as 2-pixels units as shown in Table 33.65.

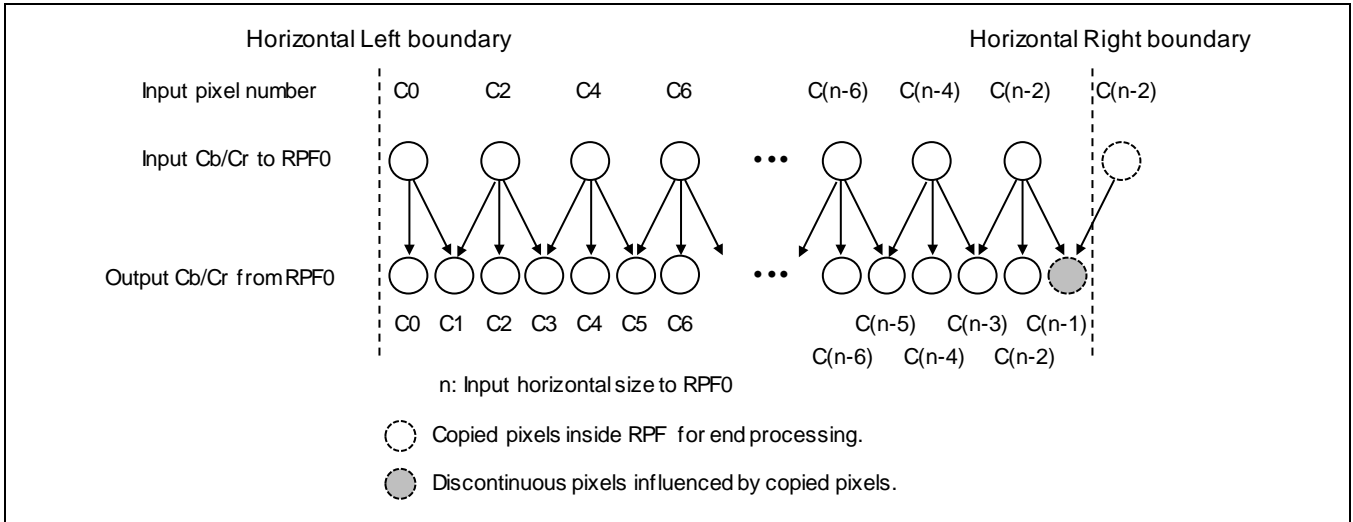


Figure 33.84 Chroma (Cb, Cr) interpolation way in RPF0 (VI6_RPF0_INFMT.CIPM = 1)

(2) SHP

- 9 taps filter
- From n-4 to n-1 output pixels are influenced by copied pixel of n-1 input pixel as shown in Figure 33.85. So it needs to clip from n-4 to n-1 output pixel for deleting discontinuous line when horizontal partition's right boundary is not frame boundary.
- From 0 to 3 output pixels are influenced by copied pixel of 0 input pixel as shown in Figure 33.85. So it needs to clip from 0 to 3 output pixels for deleting discontinuous line when horizontal partition's left boundary is not frame boundary.

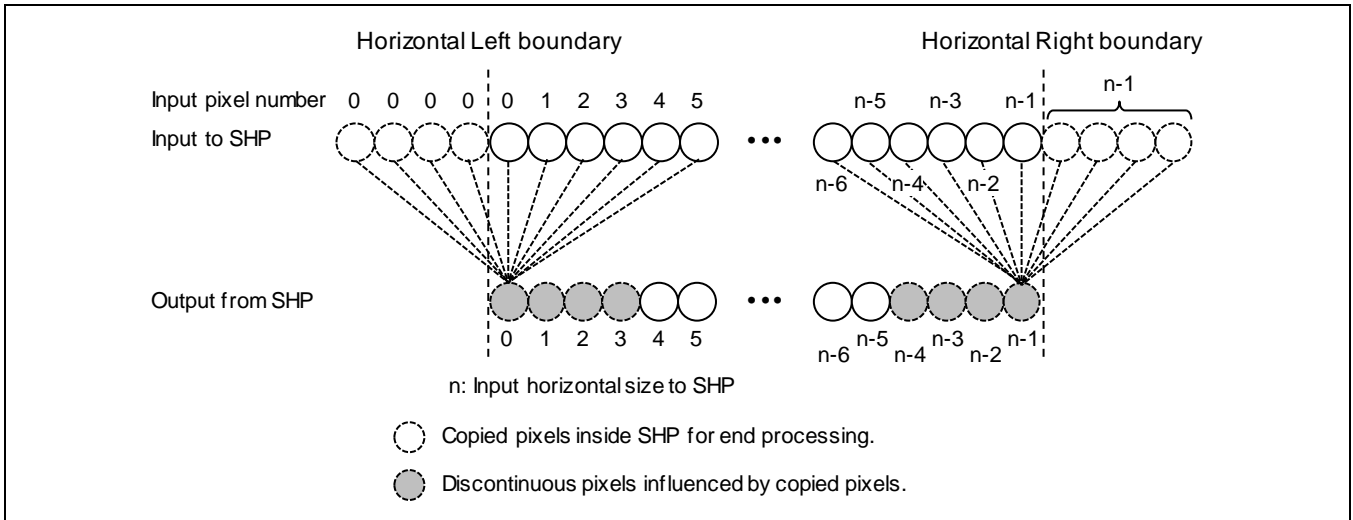


Figure 33.85 The way of applying tap in SHP filter

(3) **SRU**

SRU filter processing is composed of pre filter and post filter as shown in Figure 33.86.

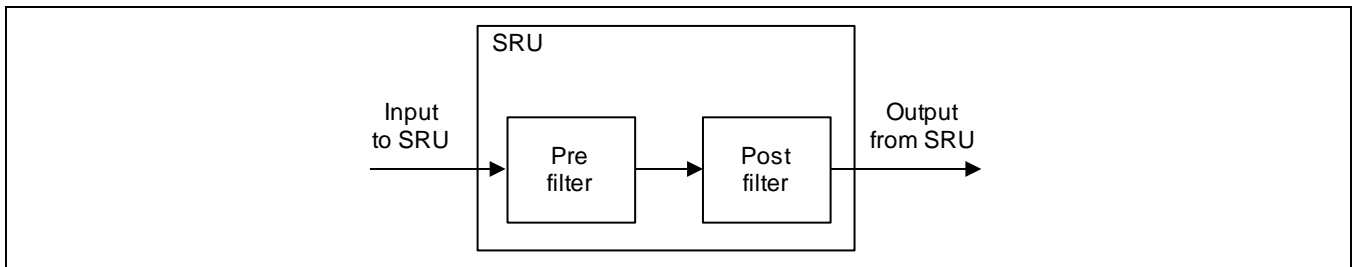


Figure 33.86 Structure of SRU filter processing

(3-1) In case of super resolution without scaling (VI6_SRU_CTRL0.SRU_MODE= B'000)

- Pre filter is 3 taps filter and go through post filter
- N-1 output pixel is influenced by copied pixel of n-1 input pixel as shown in Figure 33.87. So it needs to clip from n-1 output pixel for deleting discontinuous line when horizontal partition's right boundary is not frame boundary.
- 0 output pixel is influenced by copied pixel of 0 input pixel as shown in Figure 33.87. So it needs to clip 0 output pixel for deleting discontinuous line when horizontal partition's left boundary is not frame boundary.

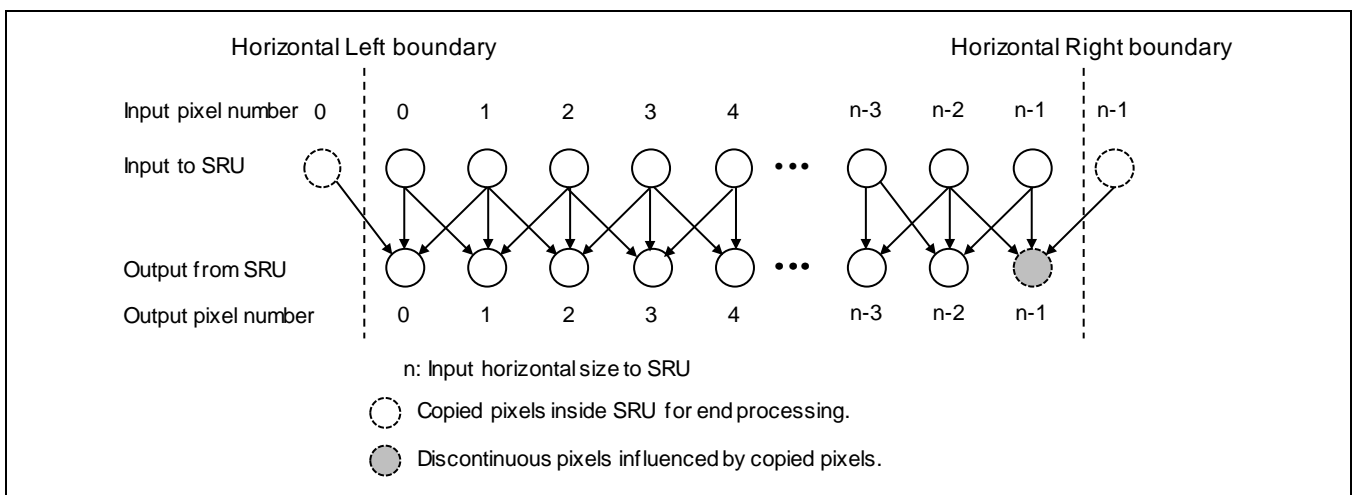


Figure 33.87 The way of applying tap in SRU filter (VI6_SRU_CTRL0.SRU_MODE = B'000)

(3-2) In case of super resolution with double scale-up (VI6_SRU_CTRL0.SRU_MODE = B'100)

- Pre filter is 3 taps filter and post filter is 2 taps filter.
- Even (0, 2, 4 ...) output pixel is generated by 3 pixels (3 taps).
- Odd (1, 3, 5 ...) output pixel is generated by 4 pixels (4 taps).
- From m-3 to m-1 output pixels are influenced by copied pixel of n-1 input pixel as shown in Figure 33.88. So it needs to clip from m-3 to m-1 output pixels for deleting discontinuous line when horizontal partition's right boundary is not frame boundary.
- From 0 to 1 output pixels are influenced by copied pixel of 0 input pixel as shown in Figure 33.88. So it needs to clip from 0 to 1 output pixels for deleting discontinuous line when horizontal partition's left boundary is not frame boundary.

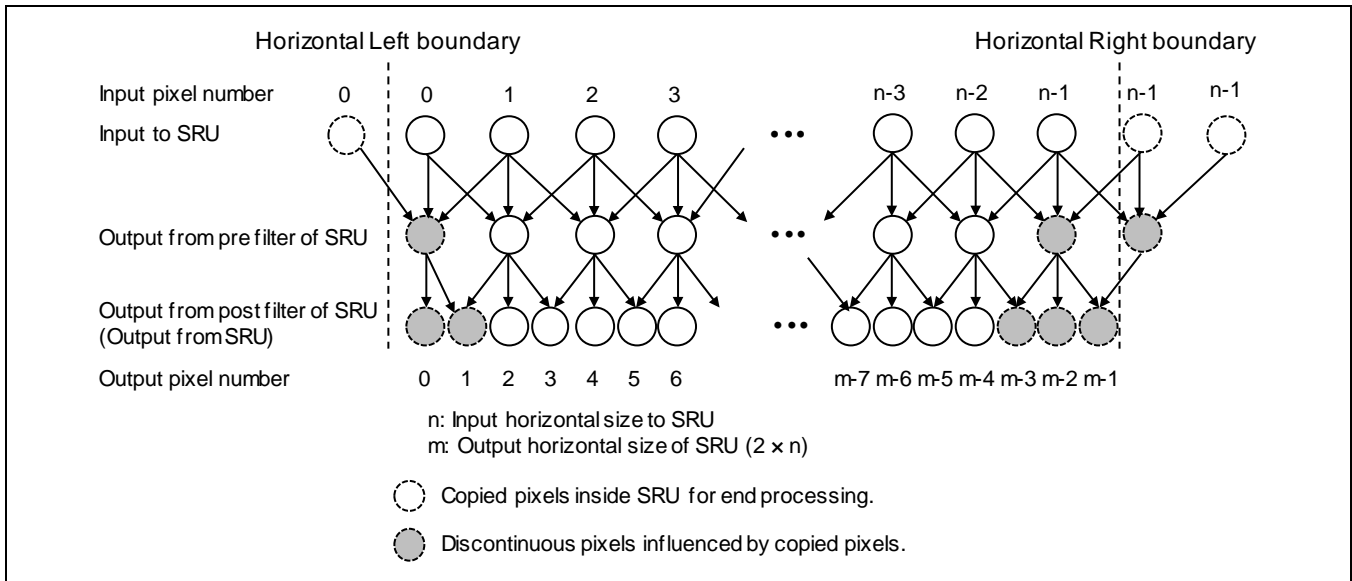


Figure 33.88 The way of applying tap in SRU filter (VI6_SRU_CTRL0.SRU_MODE = B'100)

(4) UDS

This section shows the way to calculate necessary input pixels of UDS (from src_pos0 position to src_pos1 position shown in Figure 33.89), discontinuous pixels generated by UDS (left position of discontinuous pixels is dst_pos0_pb as shown in Figure 33.89), necessary setting of horizontal start phase (hstp) and necessary setting of horizontal end phase (hedp) for UDS in order to generate valid output pixels from dst_pos0 position to dst_pos1 position as shown in Figure 33.89. Each parameter's definition is shown in (a) and (b) below. The way to calculate each parameter is shown in (c) below.

(a) Input parameters decided by user

- dst_pos0: Left position of desired valid pixels output from UDS.
- dst_pos1: Right position of desired valid pixels output from UDS.

(b) Parameters calculated in this section

- dst_pos0_pb: Left position of discontinuous pixels output from UDS.
- src_pos0: Left position of input pixels to UDS to generate valid pixels from dst_pos0 position.
- src_pos1: Right position of input pixels to UDS to generate valid pixels to dst_pos1 position.
- hstp: Horizontal start phase corresponding to dst_pos0_pb position.
- hedp: Horizontal end phase.

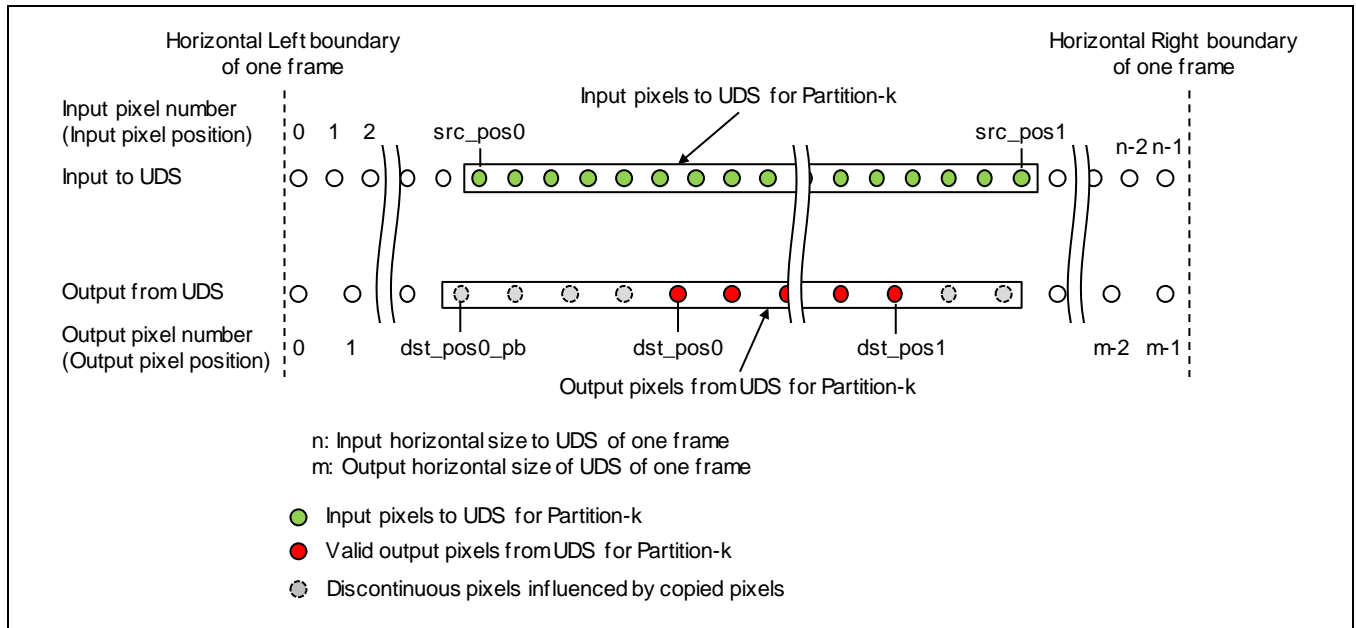


Figure 33.89 Relation between valid output pixels' position and necessary input pixels' position of UDS

(c) The way to calculate each parameter.

<procedure1> Calculate temporary src_pos0 position

For 1st partition (left end partition of source image)

$$\text{src_pos0 (temporal)} = 0$$

For from 2nd partition to last partition (except left end partition of source image)

$$\text{src_pos0 (temporal)} = \text{rdown}((\text{dst_pos0} \times \alpha - \text{phase_edge} \times m_h') / (4096 \times m_h')) - \text{sub_src} \times m_h'$$

m_h is the value of VI6_UDS_SCALE.HMANT, and f_h is the value of VI6_UDS_SCALE.HFRAC. m_h' is defined in Table 33.31. phase_edge is defined in Table 33.62. rdown is defined in section 33.4.7. sub_src is 1 in case of $m_h' = 1$, sub_src is 2 in case of $m_h' = 2$ or 4. alpha can be expressed as below.

$$\alpha = 4096 \times m_h + f_h$$

Table 33.62 Value of phase_edge referenced in above calculation

| VI6_UDS_SCALE.HMANT | VI6_UDS_CTRL.AMD | phase_edge |
|---------------------|------------------|---|
| 0 | 0 | 0 |
| | 1 | $\text{rdown}((4096 - \text{VI6_UDS_SCALE.HFRAC})/2)$ |
| 1 to 15 | 0 to 1 | 0 |

<procedure2> Calculate temporary dst_pos0_pb position

For 1st partition (left end partition of source image)

$$\text{dst_pos0_pb (temporal)} = 0$$

For from 2nd partition to last partition (except left end partition of source image)

$$\text{dst_pos0_pb (temporal)} = \text{rdown} ((\text{src_pos0 (temporal)} \times 4096 + \text{phase_edge} \times m_h') / \text{alpha})$$

<procedure3> Decide dst_pos0_pb position from temporal dst_pos0_pb

There is a restriction about the left position which UDS outputs including discontinuous pixels. So it's necessary to shift the position of dst_pos0_pb (temporal) to the left so that the restrictions may be satisfied. dst_pos0_pb can be expressed as below.

$$\text{dst_pos0_pb} = \text{dst_pos0_pb (temporal)} - \text{dst_pos0_pb_shift}$$

Here, dst_pos0_pb_shift means the number of pixels by which dst_pos0_pb (temporal) is shifted to the left. Decide dst_pos0_pb_shift as satisfying restriction of [(dst_pos0_pb x alpha) should be multiple of m_h']. The restriction can be expressed as below in other word.

There is no restriction in case of m_h' = 1

- (dst_pos0_pb x alpha) must be multiple of 2 in case of m_h' = 2
- (dst_pos0_pb x alpha) must be multiple of 4 in case of m_h' = 4

Because clip register of UDS output (VI6_UDS_CLIP_SIZE) doesn't have clip offset setting, set the number of pixels of sum of valid output pixels and left discontinuous pixels to VI6_UDS_CLIP_SIZE.CL_HSIZE as shown in Figure 33.90. The number of sum of valid output pixels and left discontinuous pixels is (dst_pos1 - dst_pos0_pb + 1) in the case. Left discontinuous pixels output from UDS should be clipped by offset register of WPF input clip (VI6_WPF0_HSZCLIP.HCL_OFST) as shown in Figure 33.90. The number of left discontinuous pixels is (dst_pos0 - dst_pos0_pb) in the case. When modules which output discontinuous pixels such as SRU and/or SHP are connected after UDS as shown in Figure 33.91, discontinuous pixels are generated by also those modules. Add the number of pixels to the offset setting in the case.

The number of valid pixels from UDS should be set in size setting of WPF input clip register (VI6_WPF0_HSZCLIP.HCL_SIZE) as shown in Figure 33.90. The number of valid pixels is (dst_pos1 - dst_pos0 + 1) in the case. When modules which output discontinuous pixels such as SRU and/or SHP is connected after UDS as shown in Figure 33.91, subtract the number of the discontinuous pixels from the size setting.

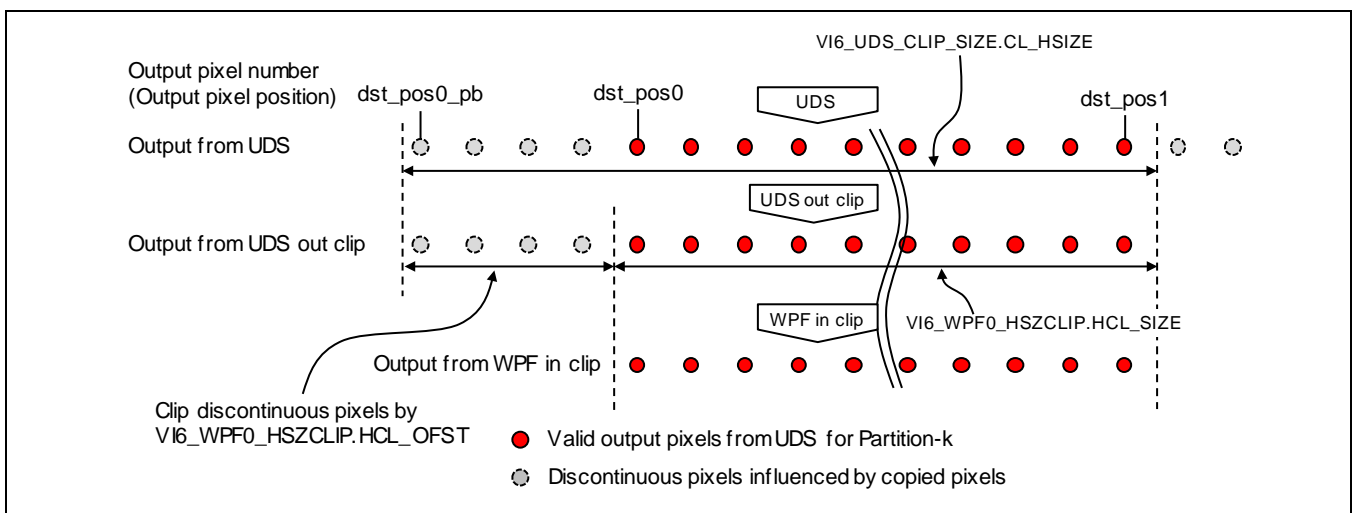


Figure 33.90 UDS output pixels and clip register's setting (Case 1)

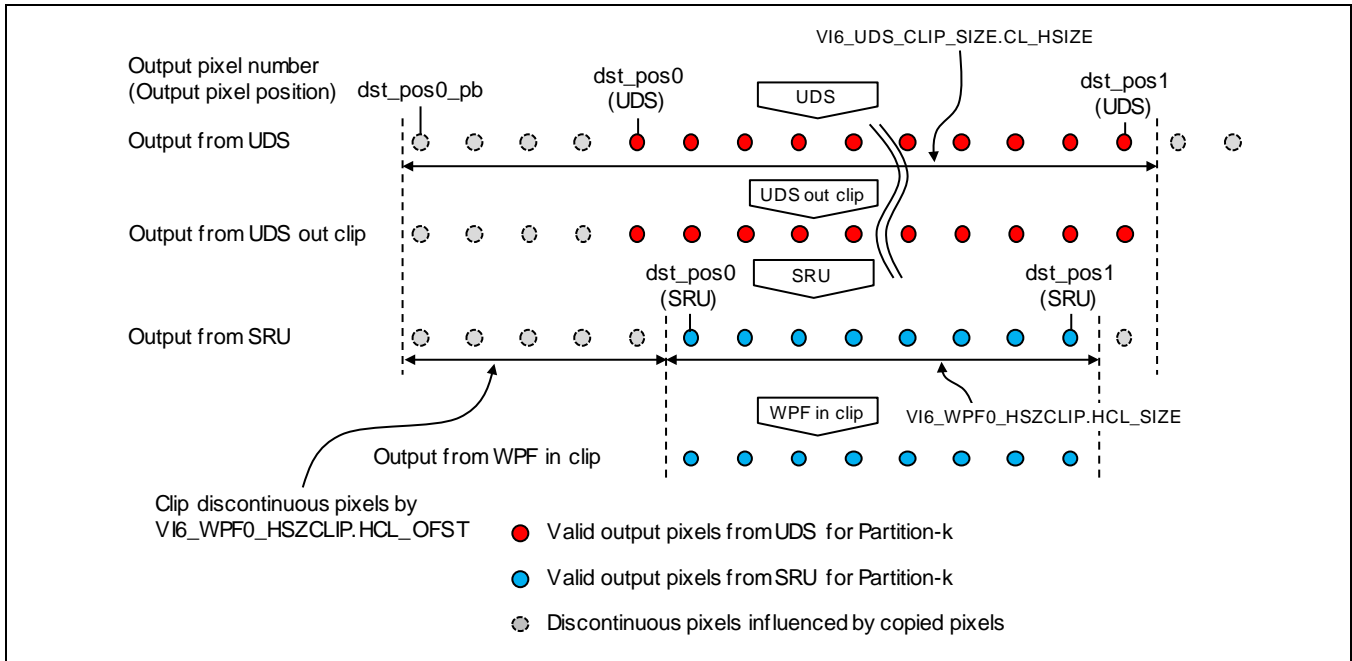


Figure 33.91 UDS output pixels and clip register's setting (Case 2)

<procedure4> Calculate src_pos0 position from dst_pos0_pb

For 1st partition (left end partition of source image)

$$\text{src_pos0} = 0$$

For from 2nd partition to last partition (except left end partition of source image)

$$\text{src_pos0} = \text{rdown} \left(\frac{(\text{dst_pos0_pb} \times \alpha - \text{phase_edge} \times m_h')}{(4096 \times m_h')} + \text{add_src} \right) \times m_h'$$

add_src is 0 in case that phase_residual expressed below is 0, add_src is 1 in case that phase_residual expressed below is not 0. phase_residual can be expressed as below.

$$\text{phase_residual} = \text{rdown} \left((\alpha \times \text{dst_pos0_pb} - \text{phase_edge} \times m_h') \% (4096 \times m_h') \right)$$

<procedure5> Calculate src_pos1 position from dst_pos1

For from 1st partition to 2nd last partition (except right end partition of source image)

$$\text{src_pos1} = (\text{rdown} \left(\frac{(\text{dst_pos1} \times \alpha - \text{phase_edge} \times m_h')}{(4096 \times m_h')} \right) + 2) \times m_h' + \text{rdown} (m_h'/2)$$

For last partition (right end partition of source image)

$$\text{src_pos1} = n - 1$$

n is horizontal input size of UDS of one frame as shown in Figure 33.89

<procedure6> Calculate hstp

For 1st partition (left end partition of source image)

hstp = phase_edge

For from 2nd partition to last partition (except left end partition of source image)

hstp = 4096 – rdown (phase_residual/ m_h) in case that phase_residual is not 0

hstp = 0 in case that phase_residual is 0

Set the value of hstp to VI6_UDS_HPHASE.HSTP. To enable VI6_UDS_HPHASE, set 1 to VI6_UDS_CTRL.AMDSLH.

<procedure7> Calculate hedp

For from 1st partition to 2nd last partition (except right end partition of source image)

hedp = 0

For last partition (right end partition of source image)

hedp = phase_edge

Set the value of hedp to VI6_UDS_HPHASE.HEDP. To enable VI6_UDS_HPHASE, set 1 to VI6_UDS_CTRL.AMDSLH.

33.3.7.6 Horizontal size clipping registers for image partition

As explained in section 33.3.7.1 and section 33.3.7.4, discontinuous line is generated at partition boundary. So clipping discontinuous line is needed at boundary of image partition.

Horizontal input size clipping can be realized at input point of UDS module, input point of WPF0 module, input point of HGO module, input point of HGT module and output point of UDS module as shown in Figure 33.92. Table 33.63 shows horizontal size clipping registers of each modules.

Other than an output point of UDS, arbitrary horizontal size can be specified from arbitrary horizontal offset position.

For an output point of UDS, arbitrary horizontal size can be specified from the left edge.

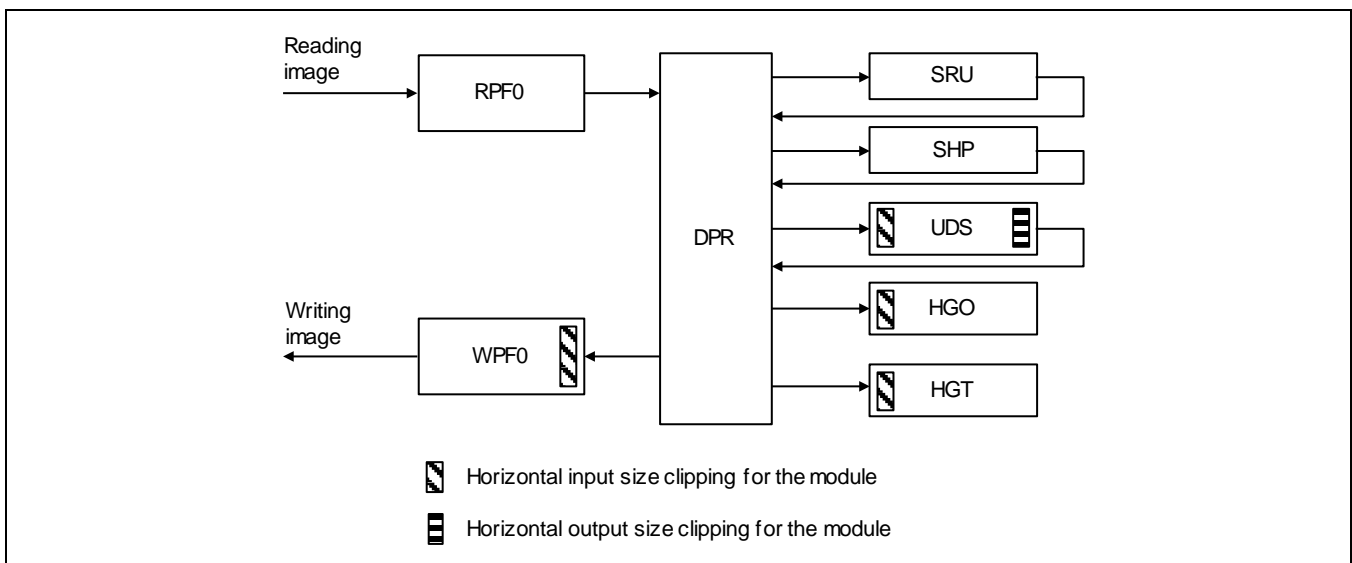


Figure 33.92 Horizontal input size clipping in VSPI

Table 33.63 Horizontal size clipping registers of each modules

| Point | Size or offset | Registers |
|-----------------|-------------------|-----------------------------------|
| UDS input (*1) | Horizontal offset | VI6_UDS_HSZCLIP.HCL_OFST [7:0] |
| | Horizontal size | VI6_UDS_HSZCLIP.HCL_SIZE [12:0] |
| UDS output (*2) | Horizontal size | VI6_UDS_CLIP_SIZE.CL_HSIZE [12:0] |
| WPF input (*3) | Horizontal offset | VI6_WPF0_HSZCLIP.HCL_OFST [7:0] |
| | Horizontal size | VI6_WPF0_HSZCLIP.HCL_SIZE [12:0] |
| HGO input | Horizontal offset | VI6_HGO_OFFSET.HOFFSET [13:0] |
| | Horizontal size | VI6_HGO_SIZE.HSIZE [13:0] |
| HGT input | Horizontal offset | VI6_HGT_OFFSET.HOFFSET [13:0] |
| | Horizontal size | VI6_HGT_SIZE.HSIZE [13:0] |

Notes: 1. Set 1 to VI6_UDS_HSZCLIP.HCEN to enable clipping horizontal size at input to UDS
 2. Horizontal offset at the UDS output point cannot be specified
 3. Set 1 to VI6_WPF0_HSZCLIP.HCEN to enable clipping horizontal size at input to WPF0

33.3.7.7 Histogram of HGO/HGT

This section shows the setting value of registers about histogram detection of HGO and HGT. Figure 33.93 shows the example of getting HGO histogram for UDS output. Those registers shown in Figure 33.93 can be set by display list.

If user wants to get histogram for partial window (like Figure 33.93), set H'0000_073F to VI6_DPR_HGO_SMPPT/VI6_DPR_HGT_SMPPT for partitions without detection window of histogram.

Set 1 to VI6_HGO_REGRST.RCLEA (VI6_HGT_REGRST.RCLEA) to initialize histogram counter for 1st partition including detection window of histogram (Partition1 in case of Figure 33.93). Then histogram counter is initialized before starting processing of the partition. To accumulate counting histogram for all partitions to get one frame's histogram, don't set 1 to VI6_HGO_REGRST.RCLEA (VI6_HGT_REGRST.RCLEA) after the partition.

Set 1 to VI6_HGO_REGRST.RCPART (VI6_HGT_REGRST.RCPART) for all partitions to initialize HGO (HGT) except histogram counter.

If user wants to dump histogram to external memory automatically [RZ/G2H, RZ/G2N, RZ/G2E], set 1 to VI6_HGO_MODE.AUTOW (VI6_HGT_MODE.AUTOW) for last partition with detection window (partition3 in case of Figure 33.93). Then histogram is dumped to external memory after processing the partition with VI6_HGO_MODE.AUTOW =1 (VI6_HGT_MODE.AUTOW =1).

hsize and hoffset in Figure 33.93 means horizontal size and horizontal offset as one frame. Set hsize and hoffset as each partition to VI6_HGO_SIZE.HSIZE, VI6_HGT_SIZE.HSIZE, VI6_HGO_OFFSET.HOFFSET and VI6_HGT_OFFSET.HOFFSET. If target node of histogram sampling generates discontinuous pixels, consider the number of discontinuous pixels and set offset register (VI6_HGO_OFFSET.HOFFSET, VI6_HGT_OFFSET.HOFFSET) as clipping also discontinuous pixels.

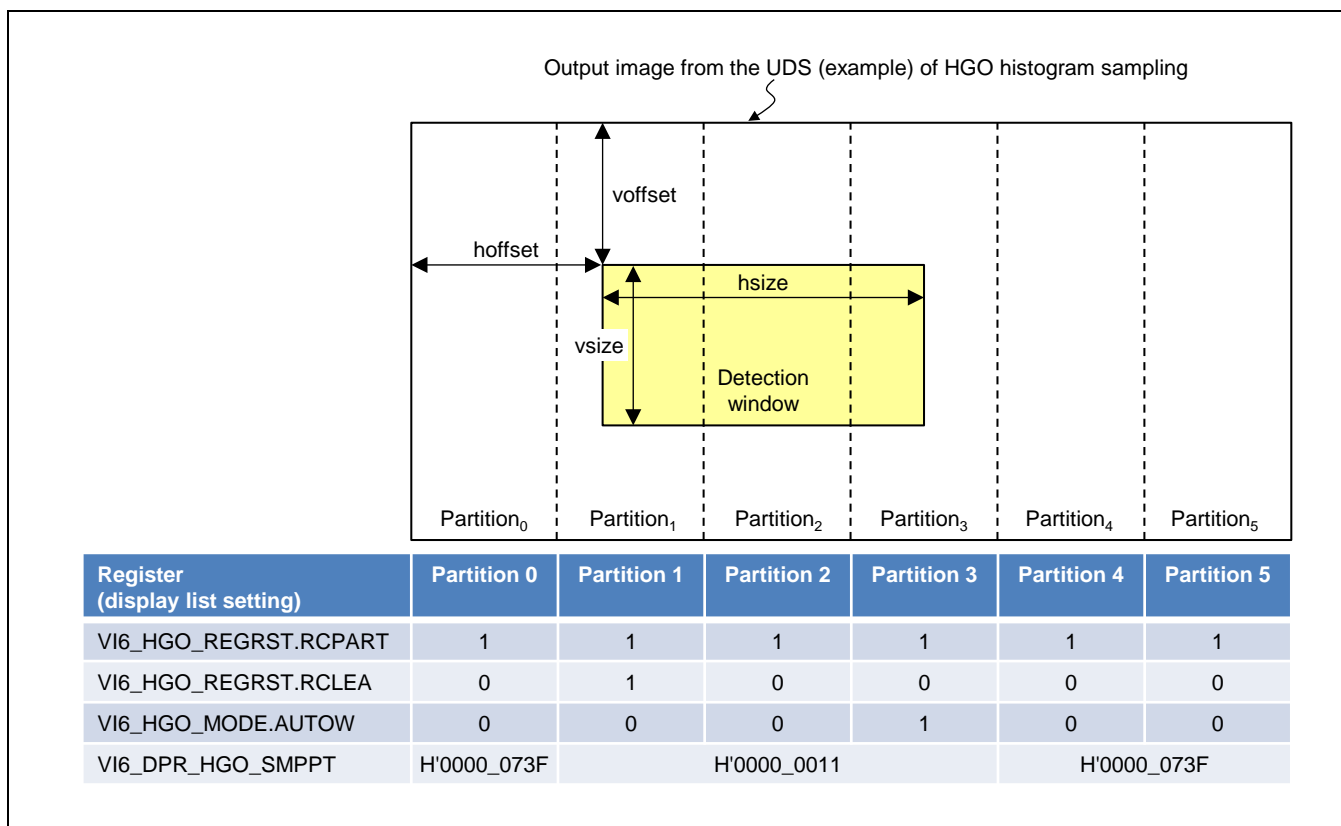


Figure 33.93 Example of histogram detection for image partition

33.3.7.8 Restriction when VSPI image partition is used

When VSPI image partition is used, there are some restrictions below.

Note: SRU (x1) means SRU without scaling. SRU (x2) means SRU with double scale-up.

- RPF
 - Tiling (VI6_RPF0_SRC_BSIZE.BHSIZE != VI6_RPF0_SRC_ESIZE.EHSIZE) is prohibited.
 - Reading 1bit/pixel alpha plane is prohibited.
- SRU and UDS
 - There are restriction when SRU and UDS are used in a VSP execution. The restriction depends on SRU resolution mode, UDS horizontal scaling factor, the order of UDS and SRU and usage of SHP. The restriction is shown in Table 33.64.
 - Example1. RPF0 -> UDS -> LUT -> SRU (x2) -> WPF0 is prohibited.
 - Example2. RPF0 -> SRU (x2) -> LUT -> UDS (hscale=2) -> WPF0 can be used.
 - Example3. RPF0 -> SRU (x1) -> LUT -> UDS (hscale=4) -> WPF0 can be used.
 - Example4. RPF0 -> SRU (x1) -> UDS (hscale=1/2) -> WPF0 is prohibited.

Table 33.64 Restriction in case of connecting SRU and UDS

| Order | SRU | UDS (*) | VSP2 supports | Note |
|-----------------|-----|---------------------|---------------|---------------------------------|
| UDS -> SRU (x1) | x1 | 1/16 < hscale <= 16 | √ | |
| SRU (x1) -> UDS | x1 | hscale < 1 | N.A | |
| | | 1 <= hscale | √ | |
| UDS -> SRU (x2) | x2 | 1/16 < hscale <= 16 | N.A | |
| SRU (x2) -> UDS | x2 | hscale =< 1/4 | N.A | |
| | | 1/4 < hscale | √ | SHP cannot be used in this case |

Note: * hscale is defined in section 33.2.10.2

- HGO/HGT
 - HGO/HGT sampling points should not be specified at modules before UDS or SRU(x2).
 - Ex. In case of RPF0 -> SHP -> UDS -> SRU(x1) -> CLU -> LUT -> WPF0 connection
 - UDS, SRU, CLU or LUT can be specified as HGO/HGT sampling points
 - RPF0 or SHP cannot be specified as HGO/HGT sampling points.
 - Ex. In case of RPF0 -> SHP -> SRU(x2) -> CLU -> LUT -> WPF0 connection
 - SRU, CLU or LUT can be specified as HGO/HGT sampling points
 - RPF0 or SHP cannot be specified as HGO/HGT sampling points.
 - Letter box detection (VI6_HGO_m_LB_DET: m = R, G, B and VI6_HGT_LB_DET) cannot be available.
 - Horizontal Pixel Skipping Mode cannot be used. Fix 0 to VI6_HGO_MODE.HRATIO [1:0], Fix 0 to VI6_HGT_MODE.HRATIO [1:0].
- WPF
 - Mode B dither cannot be used.

33.3.8 Start Reservation for VSPBD, VSPBC, VSPBS or VSPB

This section shows sequence of software when start reservation is used. The way of start reservation is common in VSPBD, VSPBC, VSPBS and VSPB, so we call VSPB* in this section. Software can reserve starting VSPB* before VSPB* processing of one frame is completed. Start reservation can remove interrupt response time and software overhead by reserving execution of VSPB* before completion of previous frame. Use frame end interrupt (VI6_WPF0_IRQ_ENB.FRE) for VSPB*.

(a) below shows software sequence of start reservation.

(b) below shows software sequence to manage the number of interrupt generation.

(a) Software sequence of start reservation

Same step of [1] ~ [9] shown in Figure 33.78 (Flow chart of setting registers by S/W in case of VSPI start reservation) can be applied for start reservation of VSPB*. Figure 33.94 shows timing chart of S/W flow to set registers for start reservation of VSPB*.

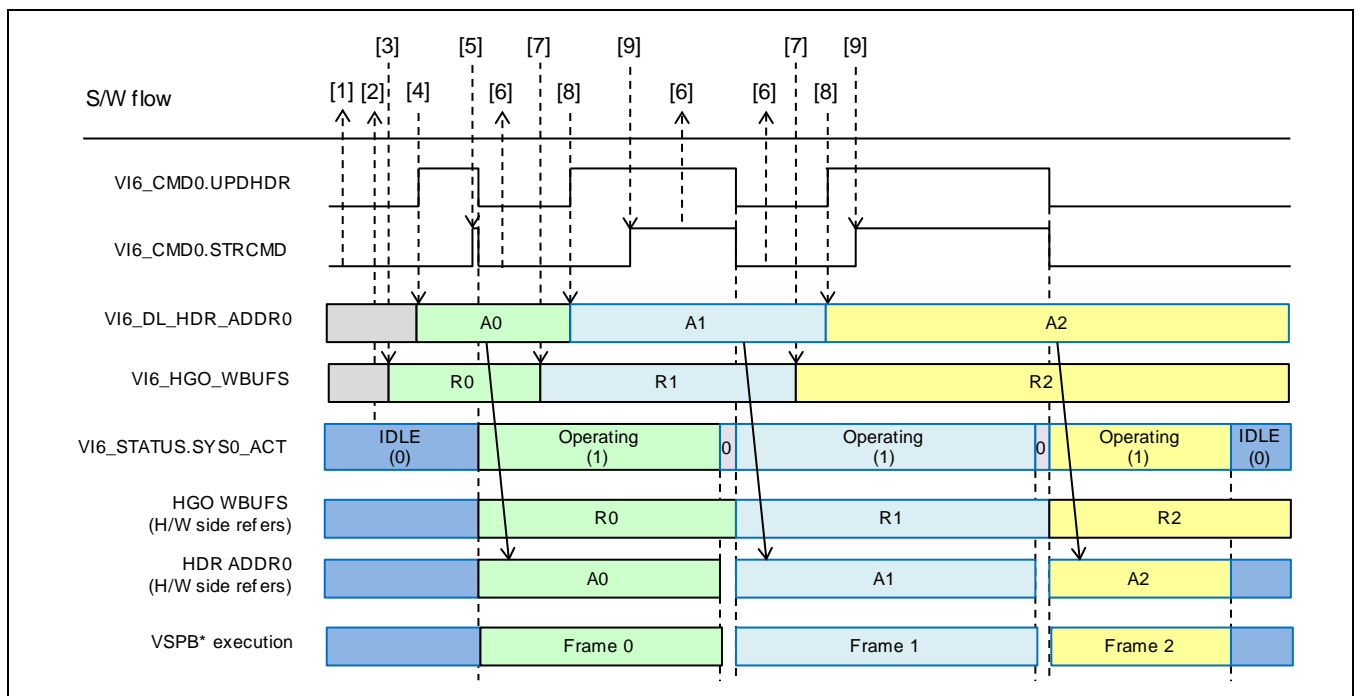


Figure 33.94 Timing of setting registers by S/W in case of VSPB* start reservation

(b) Software sequence to manage the number of interrupt generation.

The way of using interrupt is shown below.

[1] Set 1 to the VI6_WPF0_IRQ_ENB.FRE to use frame end interrupt (VI6_WPF0_IRQ_STA.FRE)

[2] Write 0 to VI6_WPF0_IRQ_STA.FRE to clear status after interrupt processing as shown in Figure 33.95.

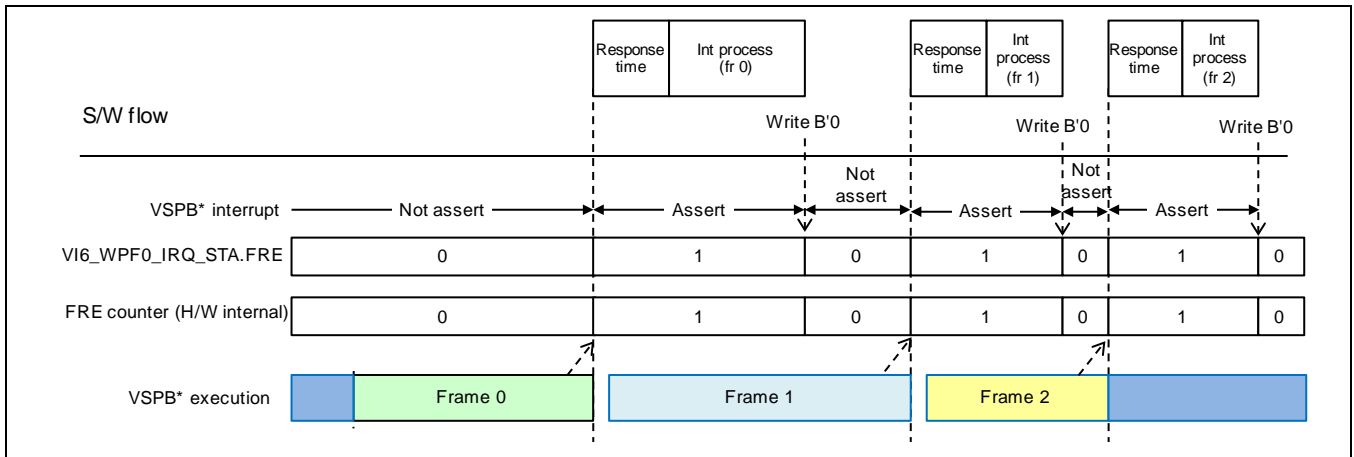


Figure 33.95 Timing of interrupt processing by S/W in case of VSPB* start reservation (Case1)

Be careful following points as shown in Figure 33.96.

- [1] Set 0 to `VI6_WPF0_IRQ_STA.FRE` for the number of interrupt generation to clear `VI6_WPF0_IRQ_STA.FRE`.
- [2] The VSPB* H/W can count at most two of interrupt generation. If three times of interrupts occur before clearing the interrupt factor, interrupt counter overflows. Therefore, wait to reserve starting VSPB* of frame {n+2} until interrupt factor of the frame {n} is cleared.

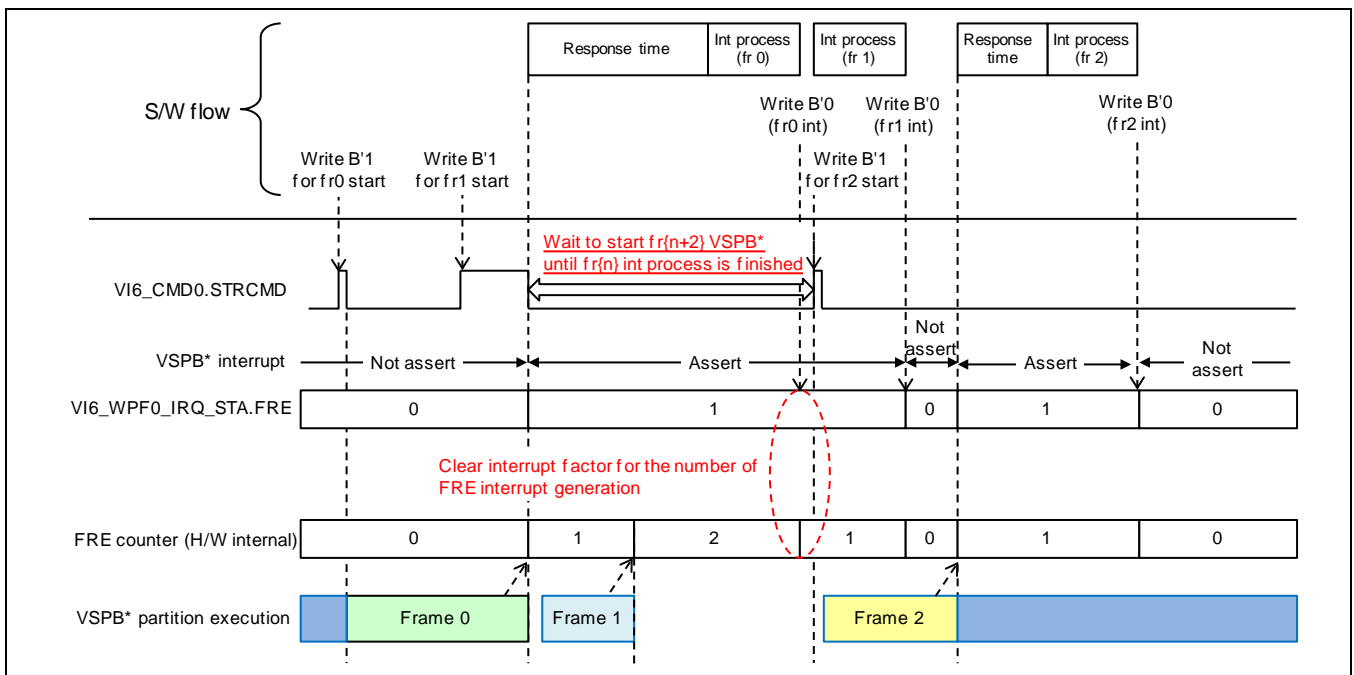


Figure 33.96 Timing of interrupt processing by S/W in case of VSPB start reservation (Case2)

33.3.9 The step to read HGO histogram registers of 256 step mode by indirect addressing

When HGO is used as 256 step mode (VI6_HGO_MODE.STEP = 1) and HGO histogram is not automatically dumped to external memory (VI6_HGO_MODE.AUTOW = 0), read HGO histogram registers by indirect addressing. The step is shown below.

- (1) Set VI6_HGO_RBUFS (section 33.2.16.14) to specify buffer side of reading HGO histogram. [RZ/G2M V1.3, RZ/G2M V3.0 only]
- (2) Set value-m in VI6_HGO_EXT_HIST_ADDR (section 33.2.16.11). m is frequency of Y or MaxRGB pixel with value-m.
- (3) Read VI6_HGO_EXT_HIST_DATA (section 33.2.16.12) to get frequency of Y or MaxRGB pixel with value-m. Repeat step (2) and step (3) 256 times from m = 0 to m = 255.

33.4 Usage Notes

RZ/G2H

RZ/G2M V1.3
RZ/G2NRZ/G2M V3.0
RZ/G2E

33.4.1 Assignment in Memory Space

Make sure that VSP2 memory space shall be mapped to Non-Cache region.

33.4.2 Limitations on Software Reset and Module Standby

Refer to Usage Notes of section 35 FCP User's Manual

33.4.3 Input Image Size

Table 33.65 is a list of input size specifications.

Table 33.65 List of Input Size Specifications

| Module | Min. Input Size | Max. Input Size | Restrictions on Setting Unit |
|--------|---|---|--|
| RPF | 1 (horizontal) × 1 (vertical) pixel | 8190 (horizontal) × 8190 (vertical) pixels | YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. YCbCr420: 2-pixel units both horizontally and vertically. Other formats: 1-pixel units both horizontally and vertically. Notes: 1. When the 1-bpp alpha plane*1 is input, the size can always be specified in 8-pixel units both horizontally and vertically regardless of the input format. 2. These restrictions including note 1 are applied to the following. VI6_RPFn_SRC_BSIZE VI6_RPFn_SRC_ESIZE |
| SRU*2 | 4 (horizontal) × 4 (vertical) pixels | Without scaling (same size): 288 (horizontal) × 8190 (vertical) pixels With double scale-up: 288 (horizontal) × 4095 (vertical) pixels | 1-pixel units both horizontally and vertically. |
| UDS*2 | 4 (horizontal) × 4 (vertical) pixels | 304 (horizontal) *3 × 8190 (vertical) pixels | 1-pixel units both horizontally and vertically. |
| SHP | 4 (horizontal) × 4 (vertical) pixels | 8190 (horizontal) × 8190 (vertical) pixels | 1-pixel units both horizontally and vertically. |
| ILV | 1 (horizontal) × 1 (vertical) pixel | Side by Side Mode 4095 (horizontal) × 8190 (vertical) pixels Line by Line Mode 8190 (horizontal) × 4095 (vertical) pixels Pixel by Pixel Mode 4095 (horizontal) × 8190 (vertical) pixels Doubler Mode 4095 (horizontal) × 4095 (vertical) pixels | 1-pixel units both horizontally and vertically. |
| LUT | 1 (horizontal) × 1 (vertical) pixel | 8190 (horizontal) × 8190 (vertical) pixels | 1-pixel units both horizontally and vertically. |
| CLU | 1 (horizontal) × 1 (vertical) pixel | 8190 (horizontal) × 8190 (vertical) pixels | 1-pixel units both horizontally and vertically. |
| HST | 1 (horizontal) × 1 (vertical) pixel | 8190 (horizontal) × 8190 (vertical) pixels | 1-pixel units both horizontally and vertically. |
| HSI | 1 (horizontal) × 1 (vertical) pixel | 8190 (horizontal) × 8190 (vertical) pixels | 1-pixel units both horizontally and vertically. |
| BRU | 1 (horizontal) × 1 (vertical) pixel | 8190 (horizontal) × 8190 (vertical) pixels | 1-pixel units both horizontally and vertically. |

| Module | Min. Input Size | Max. Input Size | Restrictions on Setting Unit |
|--------|---|---|--|
| BRS | 1 (horizontal) × 1 (vertical) pixel | 8190 (horizontal) × 8190 (vertical) pixels | 1-pixel units both horizontally and vertically. |
| HGO | 1 (horizontal) × 1 (vertical) pixel | 8190 (horizontal) × 8190 (vertical) pixels | 1-pixel units both horizontally and vertically. |
| HGT | 1 (horizontal) × 1 (vertical) pixel | 8190 (horizontal) × 8190 (vertical) pixels | 1-pixel units both horizontally and vertically. |
| LIF | 128 (horizontal) × 96 (vertical) pixels | 8190 (horizontal) × 8190 (vertical) pixels | 1-pixel units both horizontally and vertically. |
| WPF*2 | 1 (horizontal) × 1 (vertical) pixel | When VI6_WPF0_OUTFMT.ROT is smaller than 2 8190 (horizontal) × 8190 (vertical) pixels When VI6_WPF0_OUTFMT.ROT is bigger than 1, 256(horizontal) × 8190(vertical) pixels | YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. YCbCr420: 2-pixel units both horizontally and vertically. Other formats: 1-pixel units both horizontally and vertically. Note 1: This restriction on the WPF only applies to the WPF output size. The WPF input size should be specified in 1-pixel units. Note 2: In case of image rotation and flipping, other restrictions are applied. Refer to section 33.2.7.7, which explains LMEM_WD. |

Notes: 1. When VI6_RPFn_ALPH_SEL.ASEL is set to B'011.

- To support horizontal input size is larger than 256 pixels about SRU, UDS and/or WPF/ROT (VI6_WPF0_OUTFMT.ROT>1), the way of image partition is needed. Refer to section 33.3.7 for realizing the way of image partition.
- The restriction of horizontal size is applied to input size of UDS in case of up-scaling in horizontal direction. The restriction of horizontal size is applied to output size of UDS in case of down-scaling or no-scaling in horizontal direction.

The most important restriction shown in Table 33.65 is the setting unit. Make appropriate register settings so that the size of the image input to each module does comply with setting units and does not exceed the limits shown in Table 33.65. Especially for the register settings of SRU and UDS, where the size can be changed within the modules, be careful not to exceed the input size limit for the module connected behind each of them.

33.4.4 Output Image Size

The size of the output from each WPF is determined by the results of processing in the modules connected with the DPR. As shown in Figure 33.1 to Figure 33.6, the data input to the VSP2 is sent to the WPF output modules through the RPF and the modules connected with the DPR. When there is no processing that changes the image size through this data path, the WPF output size is the same as the RPF input size. Table 33.66 is a list of the processing that changes image size.

Table 33.66 Image Processing that changes image size

| Module | Function*1 | Related Register | Size of Output from Module |
|--------|---------------------------------------|--------------------------------------|--|
| SRU | Super resolution with double scale-up | VI6_SRU_CTRL0 | Super resolution without scaling (same size): Input size Super resolution with double scale-up: Input size × 2 |
| UDS | Output size clipping | VI6_UDS_CLIP_SIZE | Horizontal output size: VI6_UDS_CLIP_SIZE.CL_HSIZE setting Vertical output size: VI6_UDS_CLIP_SIZE.CL_VSIZE setting |
| WPF | Input size clipping | VI6_WPFn_HSZCLIP VI6_WPFn_VSZCLIP | When this function is disabled, the input size and the output size are the same. When this function is enabled, the output is in the following size. Horizontal output size: VI6_WPFn_HSZCLIP.HCL_SIZE *2 setting Vertical output size: VI6_WPFn_VSZCLIP.VCL_SIZE *2 setting |
| ILV | 3D format conversion | VI6_ILV_CTRL | Side by side mode or Pixel by pixel mode Horizontal: Input size × 2, Vertical: Input size Line by line mode Horizontal: Input size, Vertical: Input size × 2 Doubler mode Horizontal: Input size × 2, Vertical: Input size × 2 |

Notes: 1. For details of each function, refer to the descriptions of the related registers.

2. Refer to Section 33.2.1 "Notational Conventions for Registers and Bit Fields" for explanation of register bit field.

The input image size can be changed only with the modules and functions shown in Table 33.66. With the other modules and functions, the input size and output size are the same. Accordingly, after module connections with the DPR are determined, the VSP2 output size can also be determined through the following steps.

1. The image size (VI6_RPFn_SRC_ESIZE) read from the external memory and sent to the DPR by the RPFn is the initial value.
2. When the module connected with the DPR is not a module (function) shown in Table 33.66, the output size from the module is the same as the input size; the image size does not need to be updated.
3. When a module connected with the DPR is a module (function) shown in Table 33.66, the output image size should be updated to the size shown in the table, which should be used as the input image size for the module connected behind it.
4. When the final image size at the WPFn is determined, this size is the VSP2 output size for that WPFn path.

Figure 33.97 shows how to determine the image size in a sample DPR connection through the above steps. The related conditions that determine the image size are also shown.

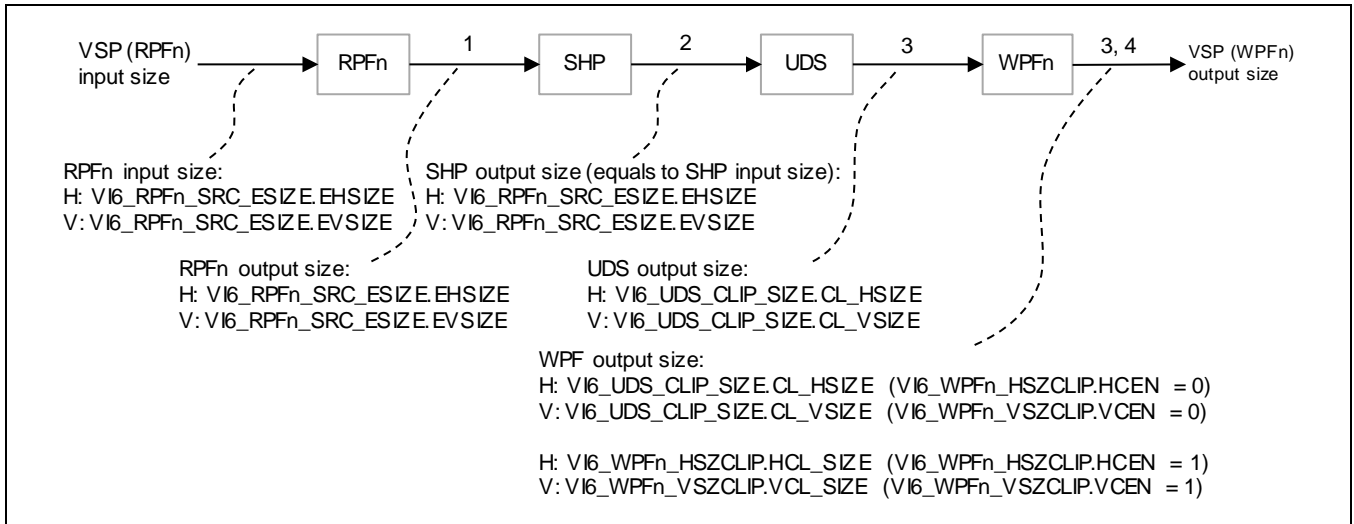


Figure 33.97 Input/output Size for Each Module in a Sample DPR Connection

Make appropriate register settings in each module so that the VSP2 output image size determined as shown in the figure does not violate the restrictions shown in Table 33.66.

33.4.5 Configuration of VSP2

Table 33.67 to Table 33.72 show the configuration of each type of VSP2 unit on this LSI.

All the registers and functions listed in [Module] category of Table 33.67 to Table 33.72 should be treated as un-implemented functions in each VSP2 unit. Do not set any registers related to the un-implemented function without any statement.

Table 33.67 VSPI configuration

| Category | Configuration | Restricted contents |
|-------------|------------------|--|
| Restriction | WPF/format | When FCNL compression is enabled, write format is restricted. See section 35 FCP User's Manual for more detail. Related registers: VI6_WPF0_OUTFMT.ROT, VI6_WPF0_OUTFMT.FCNL, VI6_WPF0_OUTFMT.WRFMT |
| | WPF/Addr, Stride | When FCNL compression is enabled, destination address and destination memory stride are restricted. See section 35 FCP User's Manual for more detail. Related registers : VI6_WPFn_STRIDE_Y. VI6_WPFn_STRIDE_C VI6_WPFn_DSTM_ADDR_Y VI6_WPFn_DSTM_ADDR_C0 VI6_WPFn_DSTM_ADDR_C1 |

Table 33.68 VSPBC configuration [RZ/G2H]

| Category | Configuration | Restricted contents |
|-------------|-----------------------------|---|
| Function | RPF/OSD-CLUT | OSD-CLUT is available only in RPF1 and RPF2. OSD-CLUT is not available in RPF0, RPF3 and RPF4. Related registers: VI6_RPFn_INFMT.RDFMT, VI6_CLUT0_TBL*, VI6_CLUT3_TBL*, VI6_CLUT4_TBL* |
| | WPF/rotation and reflection | Only vertical flipping is available. Horizontal flipping and rotation are not available. ROT = 0 and ROT = 1 are available. ROT = 2, 3, 4, 5, 6 and 7 are not available. Related registers: VI6_WPF0_OUTFMT.ROT, VI6_WPFn_ROT_CTRL |
| Restriction | WPF/format | When FCNL compression is enabled, write format is restricted. See section 35 FCP User's Manual for more detail. Related registers: VI6_WPF0_OUTFMT.ROT, VI6_WPF0_OUTFMT.FCNL, VI6_WPF0_OUTFMT.WRFMT |
| | WPF/Addr, Stride | When FCNL compression is enabled, destination address and destination memory stride are restricted. See section 35 FCP User's Manual for more detail. Related registers: VI6_WPFn_STRIDE_Y. VI6_WPFn_STRIDE_C VI6_WPFn_DSTM_ADDR_Y VI6_WPFn_DSTM_ADDR_C0 VI6_WPFn_DSTM_ADDR_C1 |

Table 33.69 VSPBD configuration [RZ/G2H]

| Category | Configuration | Restricted contents |
|-------------|-----------------------------|---|
| Function | RPF/OSD-CLUT | OSD-CLUT is available only in RPF1 and RPF2. OSD-CLUT is not available in RPF0, RPF3 and RPF4. Related registers: VI6_RPFn_INFMT.RDFMT, VI6_CLUT0_TBL*, VI6_CLUT3_TBL*, VI6_CLUT4_TBL* |
| | WPF/rotation and reflection | Only vertical flipping is available. Horizontal filliping and rotation are not available. ROT = 0 and ROT = 1 are available. ROT = 2, 3, 4, 5, 6 and 7 are not available. Related registers: VI6_WPF0_OUTFMT.ROT, VI6_WPFn_ROT_CTRL |
| Restriction | WPF/format | When FCNL compression is enabled, write format is restricted. See section 35 FCP User's Manual for more detail. Related registers: VI6_WPF0_OUTFMT.ROT, VI6_WPF0_OUTFMT.FCNL, VI6_WPF0_OUTFMT.WRFMT |
| | WPF/Addr, Stride | When FCNL compression is enabled, destination address and destination memory stride are restricted. See section 35 FCP User's Manual for more detail. Related registers: VI6_WPFn_STRIDE_Y. VI6_WPFn_STRIDE_C VI6_WPFn_DSTM_ADDR_Y VI6_WPFn_DSTM_ADDR_C0 VI6_WPFn_DSTM_ADDR_C1 |

Table 33.70 VSPB configuration [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Category | Configuration | Restricted contents |
|-------------|-----------------------------|---|
| Function | RPF/OSD-CLUT | OSD-CLUT is available only in RPF1 and RPF2. OSD-CLUT is not available in RPF0, RPF3 and RPF4. Related registers: VI6_RPFn_INFMT.RDFMT, VI6_CLUT0_TBL*, VI6_CLUT3_TBL*, VI6_CLUT4_TBL* |
| | WPF/rotation and reflection | Only vertical flipping is available. Horizontal filliping and rotation are not available. ROT = 0 and ROT=1 are available. ROT = 2, 3, 4, 5, 6 and 7 are not available. Related registers: VI6_WPF0_OUTFMT.ROT, VI6_WPFn_ROT_CTRL |
| Restriction | WPF/format | When FCNL compression is enabled, write format is restricted. See section 35 FCP User's Manual for more detail. Related registers: VI6_WPF0_OUTFMT.ROT, VI6_WPF0_OUTFMT.FCNL, VI6_WPF0_OUTFMT.WRFMT |
| | WPF/Addr, Stride | When FCNL compression is enabled, destination address and destination memory stride are restricted. See section 35 FCP User's Manual for more detail. Related registers: VI6_WPFn_STRIDE_Y. VI6_WPFn_STRIDE_C VI6_WPFn_DSTM_ADDR_Y VI6_WPFn_DSTM_ADDR_C0 VI6_WPFn_DSTM_ADDR_C1 |

Table 33.71 VSPD configuration

| Category | Configuration | Restricted contents |
|-------------|-----------------------------|---|
| Function | RPF/OSD-CLUT | OSD-CLUT is available only in RPF2. OSD-CLUT is not available in RPF0, RPF1, RPF3 and RPF4. Related registers. VI6_RPFn_INFMT.RDFMT, VI6_CLUT0_TBL*, VI6_CLUT1_TBL*, VI6_CLUT3_TBL*, VI6_CLUT4_TBL* |
| | WPF/rotation and reflection | Horizontal filling, vertical flipping and rotation are not available. ROT = 0 is available. ROT = 1, 2, 3, 4, 5, 6 and 7 are not available. Related registers: VI6_WPF0_OUTFMT.ROT |
| | Role of WPF0 and WPF1 | WPF0 and WPF1 cannot be executed in parallel. Following 3 use cases is possible for VSPD (Use Case 1) WPF0 output image data to DU without writing back the image data to memory. WPF1 cannot be executed. (Use Case 2) WPF0 output image data to DU with writing back the image data to memory. WPF1 cannot be executed. (Use Case 3) WPF1 is executed by memory to memory. WPF0 cannot be executed. |
| Restriction | WPF/format | FCNL compression cannot be used. Related registers : VI6_WPFn_OUTFMT.FCNL |

Table 33.72 VSPDL configuration [RZ/G2H, RZ/G2N]

| Category | Configuration | Restricted contents |
|-------------|-----------------------------|---|
| Function | RPF/OSD-CLUT | OSD-CLUT is available only in RPF2. OSD-CLUT is not available in RPF0, RPF1, RPF3 and RPF4. Related registers. VI6_RPFn_INFMT.RDFMT, VI6_CLUT0_TBL*, VI6_CLUT1_TBL*, VI6_CLUT3_TBL*, VI6_CLUT4_TBL* |
| | WPF/rotation and reflection | Horizontal filling, vertical flipping and rotation are not available. ROT = 0 is available. ROT = 1, 2, 3, 4, 5, 6 and 7 are not available. Related registers: VI6_WPF0_OUTFMT.ROT |
| Restriction | WPF/format | FCNL compression cannot be used. Related registers : VI6_WPFn_OUTFMT.FCNL |

33.4.6 Performance of VSP2

VSPI, VSPBD, VSPBC, VSPBS, VSPB, VSPD and VSPDL performance is shown in this section. VSP performance shown in below is based on evaluation under the condition that only one IP is operated.

33.4.6.1 Performance of VSPI

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

500 Mpix/s process rate per 1 set of VSPI except following cases. When one of following conditions is true, performance of VSPI is lower than 500 Mpix/s process rate. The execution of the VSPI is up to at most 480 frames per second.

[RZ/G2E]

125 Mpix/s process rate per 1 set of VSPI except following cases. When one of following conditions is true, performance of VSPI is lower than 125 Mpix/s process rate. The execution of the VSPI is up to at most 120 frames per second.

- (1) 90 degree or 270 degree rotation (VI6_WPF0_OUTFMT.ROT > 3) is enabled
- (2) Horizontal scaling ratio is larger than x4 times and SRU (x1) is used at same time.
- (3) At least one RPF has all following conditions.
 - RGB32bpp or YCbCr planar or YCbCr444 semi-planar
 - Reading alpha plane

33.4.6.2 Performance of VSPBD, VSPBC or VSPB

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

500 Mpix/s process rate per 1 set of VSPBD, VSPBC or VSPB except following cases. If one of following conditions is true, performance of VSPBD/VSPBC/VSPB is lower than 500 Mpix/s process rate. The execution of the VSPBD/VSPBC/VSPB are up to at most 480 frames per second.

[RZ/G2E]

125 Mpix/s process rate per 1 set of VSPB except following cases. If one of following conditions is true, performance of VSPB is lower than 125 Mpix/s process rate. The execution of the VSPB is up to at most 120 frames per second.

- (1) At least one RPF has all following conditions.
 - RGB32bpp or YCbCr planar or YCbCr444 semi-planar
 - Reading alpha plane
 - Source stride or start address of either plane is not multiple of 256 bytes.
- (2) Two RPFs have RGB32bpp and alpha plane.
- (3) At least three RPFs are used, and at least one of the RPF has RGB32bpp and alpha plane.
- (4-1) Total amount of byte/pixel of all RPFs is larger than 16 bytes. [for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]

Following example shows 16 byte/pixel.

 - RPF0 = YCbCr420SPL (2 byte/pixel)
 - RPF1 = YCbCr420SPL (2 byte/pixel)
 - RPF2 = ARGB (4 byte/pixel)
 - RPF3 = ARGB (4 byte/pixel)
 - RPF4 = ARGB (4 byte/pixel)
- (4-2) Total amount of byte/pixel of all RPFs and WPF is larger than 12 bytes. [for RZ/G2N]

Following example shows 12 byte/pixel.

 - RPF0 = YCbCr420SPL (2 byte/pixel)

- RPF1 = YCbCr420SPL (2 byte/pixel)
- RPF2 = YCbCr420SPL (2 byte/pixel)
- RPF3 = YCbCr420SPL (2 byte/pixel)
- WPF0 = ARGB (4 byte/pixel)

33.4.6.3 Performance of VSPD or VSPDL

In this section, the maximum dot clock frequency is the one VSPD/VSPDL can support regardless of display type. Therefore, it may be higher than the maximum dot clock frequency specified by a particular display type such as LVDS or DPAD can support. Refer to section 36 DU User's Manual and Electrical Characteristics to grasp the maximum dot clock frequency.

[RZ/G2H, RZ/G2N]

250 Mpix/s process rate (support up to 300MHz in case of DU display) per 1 set of VSPD.

125 Mpix/s process rate (support up to 150MHz in case of DU display) per 1 video path (1WPF) of VSPDL.

[RZ/G2M V1.3, RZ/G2M V3.0]

250 Mpix/s process rate (support up to 300MHz in case of DU display) per 1 set of VSPD.

[RZ/G2E]

125 Mpix/s process rate (support up to 150MHz in case of DU display) per 1 set of VSPD except following cases. If one of following conditions is true, the maximum process rate is 83.3 Mpix/sec (support up to 100MHz in case of DU display).

- (1) At least one RPF has all following conditions.
 - RGB32bpp or YCbCr planar or YCbCr444 semi-planar
 - Reading alpha plane
 - Source stride or start address of either plane is not multiple of 256 bytes.
- (2) Two RPFs have RGB32bpp and alpha plane.
- (3) At least three RPFs are used, and at least one of the RPF has RGB32bpp and alpha plane.
- (4) Total amount of byte/pixel of all RPFs is larger than 16 bytes.

Following example shows 16 byte/pixel.

- RPF0 = YCbCr420SPL (2 byte/pixel)
- RPF1 = YCbCr420SPL (2 byte/pixel)
- RPF2 = ARGB (4 byte/pixel)
- RPF3 = ARGB (4 byte/pixel)
- RPF4 = ARGB (4 byte/pixel)

33.4.7 Legend of terms

This section describes the configurations of VSP2 on this LSI. Descriptions in this section show the full-function specifications of the VSP2 IP. Some of the functions described in this section are not available on VSP2 with restrictions written here.

(1) Terminology

This section explains the terminology for VSP2.

Virtual RPF

There is an image compositing sub module named as BRU which can execute image blending or raster operation. The BRU has 5 input ports for compositing multiple images. There must be a RPF module as the start of data processing. Each RPF can be a source of data input to BRU. In addition to these RPF modules, the BRU has a function which generates monochrome color image internally. This functionality is useful because the data access to an external memory is not required. This function is called as virtual RPF.

Refer to section 33.2.7.1, 33.2.15.2, 33.2.15.3 and 33.2.15.4 for virtual RPF.

The BRS has also virtual RPF. Refer to section 33.2.7.1, 33.2.19.2, 33.2.19.3 and 33.2.19.4 for virtual RPF in the BRS.

Target WPF

The VSP2 has five RPFs (RPF 0 to 4) and four WPFs (WPF0 to 1) and execute image processing. One data path consists of more than one RPF, one WPF and other image functional modules if necessary. The WPF which is reached from specific RPF(s) in the data path is called as target WPF for the RPF(s). The number of target WPFs is always one for the data path.

Source RPF

The data source in the data path for the specific WPF is called as source RPF for that WPF. The number of source RPFs is not always one, may be more than one for image blend operation or raster operation.

Layer

In case of image composite operation such as image blending or raster operation for multiple images, there is hierarchical relationship for each image. One layer is assigned to one RPF or one virtual RPF indicating one image in hierarchical relationship.

Master Layer

The master layer is a base layer for image compositing of multiple images, and the size of the master layer equals to that of BRU output. When the image compositing is not applied, there is one layer. This layer is also called as the master layer.

Sub Layer

The Layers except the master layer are called as sub layer.

Alpha

The VSP2 can execute both color data and its transparent data. The alpha is 8-bit data and expresses the transparency of pixel. The value H'FF represents opaque pixel, and the value H'00 represents transparent pixel.

Plane

One layer consists of color components such as R/G/B or Y/Cb/Cr, and alpha data representing its transparency of the pixel. The frame which is one of color components and alpha data is called as a plane.

Picture Plane

The picture plane represents color data in one layer.

Alpha Plane

The alpha plane represents transparent data in one layer.

Blending

The image blending means the image composite operation with alpha data. The composite ratio of image blending depends on the relationship of each layer and alpha values.

Raster Operation (ROP)

The bitwise operation such as AND or OR is called as raster operation (ROP).

(2) Definition of operators and functions

The following operators, notations are defined for explaining the functions of VSP2.

 $\langle x \rangle$

Discard decimal places of value x

 $\text{rdown}(x)$

Discard decimal places of value x : Ex, $\text{rdown}(3.81) = 3$

 $\text{clip0}(x)$

Clip to 0 value if x is less than 0. [$x = (x < 0)? 0 : x$]

 $\text{clip3}(min, max, x)$

The value x shall be clipped so that x shall be $min < x < max$. [$x = (x < min)? min : ((x > max)? max : x)$]

(3) Unit

The unit is defined here for explaining VSP2's functions.

[bpp]: bits per pixel

VSP2 can handle a number of image formats. Each image format has different number of data bits. The unit [bpp] shows a number of data bits for one pixel. For example, RGB 8bpp means that its format is RGB and the pixel consists of 8 data bits.

34. Fine Display Processor (FDP1)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

34.1 Overview

The FDP1 is the de-interlacing module which converts the interlaced video to progressive video. Require software or the library to RENESAS for operating this module.

The RZ/G2H provides two channels.

The RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E provides one channel respectively.

34.1.1 Features

This module has the following features.

- High image quality motion adaptive de-interlacing algorithm (*)
 - Combines the best aspects of both Bob (2D) and Weave (3D)
 - 2D and 3D comparisons are performed to decide whether or not an individual pixel has motion.
 - Diagonal interpolation is supported in 2D compensation.
- Support 8190 x 8190 resolution
- Performance of FDP is as following:
 - Output performance of each channel is 500Mpixs/s for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N.
 - Output performance of each channel is 125Mpixs/s for RZ/G2E.
 - Input performance (interlace mode) per one active RPF of each channel is half of output performance.
 - Input performance (progressive mode) per one active RPF of each channel is equal to output performance.

Note: * This algorithm is applied for luma only. The chroma is de-interlaced by fixed 2D without diagonal interpolation.

Main Function of the FDP1

Read Pixel Formatter (RPF)

| Number of channels | | Three channels (RPF0 to RPF2) | |
|--------------------|------------|-------------------------------|--|
| Image Format | Input (*1) | YCbCr | YCbCr444-PL, SP, ILV (*2) YCbCr422-PL, SP, ILV (*2) YCbCr420-PL, SP (*2) |
| | | Maximum size | H8190 x V8190 pixels (Frame) H8190 x V4095 pixels (Field) |
| | | Minimum size | H3 2 x V32 pixels (Frame) H32 x V16 pixels (Field) |
| | | Size setting unit | YCbCr420: 2-pixel units both horizontally and vertically. YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. In other formats, the size can be set in 1-pixel units. |

Write Pixel Formatter (WPF)

| Number of channels | | | One channel (WPF) |
|---------------------------|--------------|--------------------------|--|
| Image format | Output | RGB | RGB332, RGB444, RGB565, RGB666, RGB888, α RGB8666, α RGB8888, α RGB4444, α RGB1555 α value is fixed. |
| | | YCbCr | YCbCr444-PL, SP, ILV (*2) YCbCr422-PL, SP, ILV (*2) YCbCr420-PL, SP (*2) |
| | | Maximum size | H8190 x V8190 pixels |
| | | Minimum size | H32 x V32 pixel |
| | | Size setting unit | YCbCr420: 2-pixel units both horizontally and vertically. YCbCr422: 2-pixel units horizontally and 1-pixel units vertically. In other formats, the size can be set in 1-pixel units. |
| Color space conversion | YCbCr to RGB | Conversion expression | BT.601 (16, 235/240) to RGB (0, 255) BT.709 (16, 235/240) to RGB (0, 255) BT.601 (0, 255) to RGB (0, 255) BT.709 (16, 235/240) to RGB (16, 235) |
| Changing number of colors | Output | Reducing RGB color depth | Dithering, lower-order bit truncation, or rounding |
| | | YCbCr422/420 | CbCr skipping or CbCr vertical skipping and horizontal skipping |

Interlace to Progressive Converter (IPC)

| | | |
|-----------------------------|--------------------|---|
| De-interlacing | Output frame rate | Per one input field output one frame (Example: 60i to 60p) |
| | Interpolation mode | 2D-3D adaptive (*3) 2D fixed 3D fixed (*3) |
| | Algorithm | Motion adaptive (luma only) |
| Diagonal line interpolation | | Supported (luma only) |

Notes: 1. Chroma component (Cb, Cr or U, V) of RPF0, RPF2 is not read from bus in case of planar or semi-planar format and discarded after reading in package with luma component in case of interleaved format because it is processed in fixed 2D de-interlacing.

2. PL: Planar, SP: Semi-planar, ILV: Interleaved

3. Source picture structures (field-structure or frame-structure) should be the same structure among the all input pictures.

34.1.2 Block Diagram

Figure 34.1 shows the block diagram of FDP1.

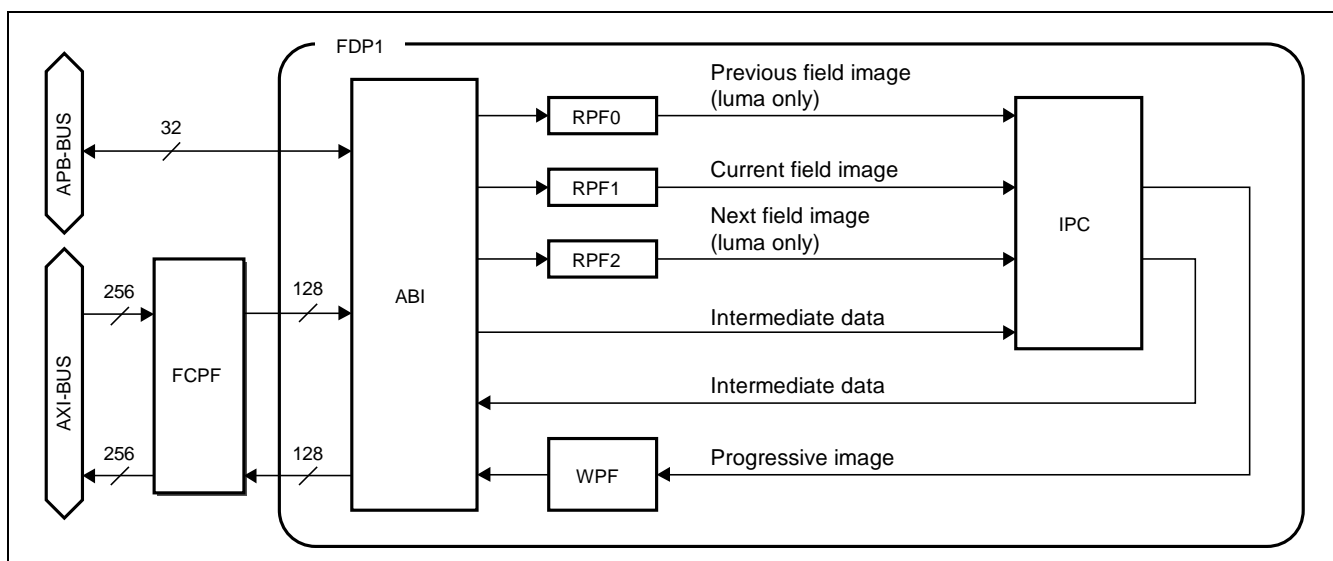


Figure 34.1 FDP1 Block diagram

The following gives an overview of each block function shown in Figure 34.1.

The operation of these functions is determined by the register setting explained in section 34.2. See section 34.2 for the relationship between the register setting and the functional behavior.

34.1.2.1 ABI (AXI Bus Interface)

The FDP1 applies processing to the image data stored in the external memory and writes the resultant data back to the external memory. The data transfer between the external memory and FDP1 necessary for this operation is done by the ABI, which works as the bus master, according to the register settings. The ABI executes this data transfer between the external memory and FDP1 through FCPF.

34.1.2.2 RPF (Read Pixel Formatter)

The RPF reads image data from the external memory through the ABI, unpacks data according to the specified format, and outputs the resultant data to the IPC. The input format unpacking unit expands the image data input from the ABI into the image format for internal processing.

The FDP1 provides three RPF modules (RPF0 to RPF2). RPF0 reads the previous field, RPF1 reads the current field and RPF2 reads the next field for processing de-interlacing operation.

34.1.2.3 IPC (Interlace to Progressive Converter)

The IPC is the de-interlacing module which reads the field image through RPF_n (n = 0, 1, 2) and writes back to the progressive image to the external memory through WPF. The IPC uses the motion adaptive algorithm which enhances the image resolution of still pixels compared with conventional 2D de-interlacing.

Note: the motion adaptive algorithm is applied to only luma component. Conventional 2D de-interlacing is applied to chroma component.

34.1.2.4 WPF (Write Pixel Formatter)

The WPF is an output module that receives image data from the IPC, converts the color space, number of colors, and format of the data, and outputs the results of FDP1 image processing to external memory through the ABI. The WPF is mainly configured from a color space converter and an output format converter (the packing unit).

The color space converter converts the color space between RGB and YCbCr, and the packing unit converts the format into the picture plane storing format. The FDP1 provides one WPF.

34.1.2.5 Input/output Data Format

The FDP1 input/output data should be stored in a format where the address is incremented in the direction from the MSB to the LSB as in big endian. For example, the FDP1 reads and writes data to and from the external memory through the dedicated bus interface with a 128-bit width, and the data that should come first should be stored in the MSB side regardless of the unit data size as shown in Figure 34.2.

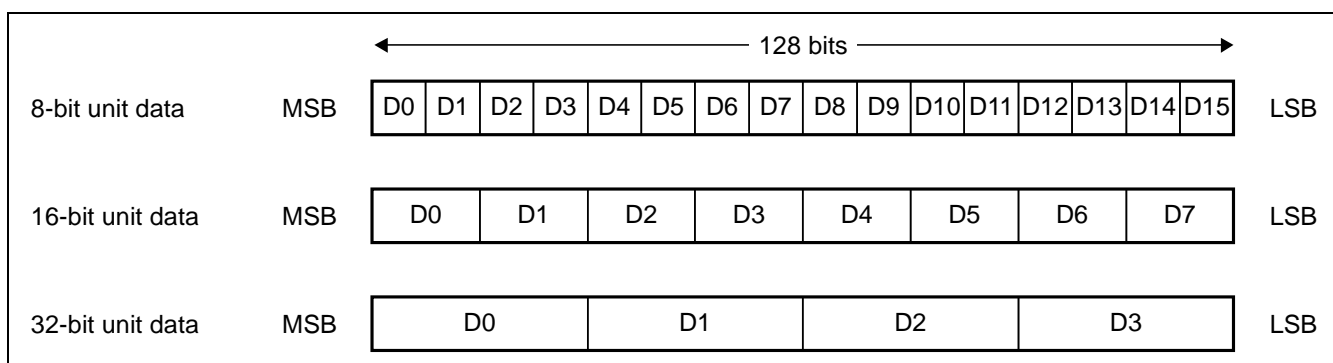


Figure 34.2 FDP1 Input/output Data Format

However, data may be arranged in a way unexpected by the FDP1 depending on the address handling method (such as little endian) of the FDP1 access destination in some cases as in the external memory format shown in Figure 34.3. In such a case, use the data swapping function (Refer to section 34.2.7.6 and 34.2.8.4) in the FDP1 to convert the data format to be suitable for the FDP1.

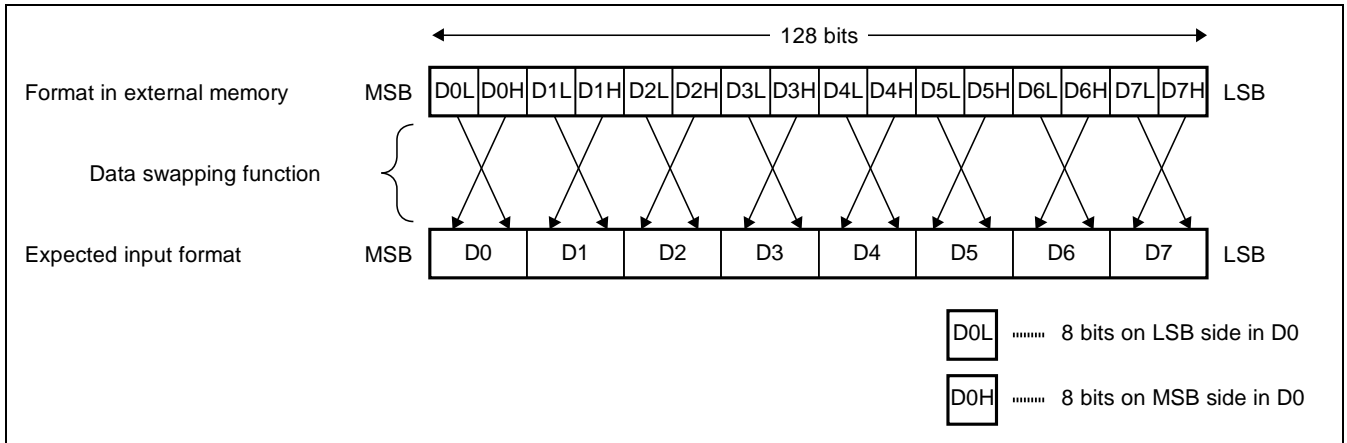


Figure 34.3 Input Data Alignment by Data Swapping Function

34.1.3 Register Configuration

Table 34.1 shows the FDP1 memory map. Immediate registers are specified as “Imm”, the V-update registers are indicated as “Vupdt” and the V-update status registers are indicated as “VupdtSt” in register type column in Table 34.1.

Base address of each FDP1:

Ch0: H'FE94_0000

Ch1: H'FE94_4000 (RZ/G2H only)

Table 34.1 List of FDP1 registers

| Name of Registers | Abbreviation | Register Type | R/W | Address (offset from base address) | Initial value | size | Second Generation RZ/G Series Products | | | |
|--|---------------------|---------------|-----|------------------------------------|---------------|------|--|----------------------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| FDP1 Start Register | FD1_CTL_CMD | Imm | R/W | H'0000 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sync Generator Register | FD1_CTL_SGCMD | Imm | R/W | H'0004 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Register Set End Register | FD1_CTL_REGEND | Imm | R/W | H'0008 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Channel Activation Register | FD1_CTL_CHACT | Vupdt | R/W | H'000C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Operation Mode Register | FD1_CTL_OPMODE | Vupdt | R/W | H'0010 | H'0000_0000 | 32 | √ | √ | √ | √ |
| V-Period Register | FD1_CTL_VPERIOD | Vupdt | R/W | H'0014 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Clock Control Register | FD1_CTL_CLKCTRL | Imm | R/W | H'0018 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Software Reset Register | FD1_CTL_SRESET | Imm | R/W | H'001C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Operating Status Register | FD1_CTL_STATUS | VupdtSt | R | H'0024 | H'0000_0000 | 32 | √ | √ | √ | √ |
| V-Cycles Status Register | FD1_CTL_VCYCLE_STAT | VupdtSt | R | H'0028 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Interrupt Enable Register | FD1_CTL_IRQENB | Imm | R/W | H'0038 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Interrupt Status Register | FD1_CTL_IRQSTA | Imm | R/W | H'003C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Interrupt Control Register | FD1_CTL_IRQFSET | Imm | R/W | H'0040 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Source Picture Size Register | FD1_RPF_SIZE | Vupdt | R/W | H'0060 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Source Picture Format Register | FD1_RPF_FORMAT | Vupdt | R/W | H'0064 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Source Picture Stride Register | FD1_RPF_PSTRIDE | Vupdt | R/W | H'0068 | H'0000_0000 | 32 | √ | √ | √ | √ |
| RPF0 Source Component Y Address Register | FD1_RPF0_ADDR_Y | Vupdt | R/W | H'006C | H'0000_0000 | 32 | √ | √ | √ | √ |

| | | | | | | | Second Generation RZ/G Series Products | | | |
|---|---------------------|---------------|-----|------------------------------------|---------------|------|--|----------------------------|--------|--------|
| Name of Registers | Abbreviation | Register Type | R/W | Address (offset from base address) | Initial value | size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| | | | | | | | √ | √ | √ | √ |
| RPF1 Source Component Y Address Register | FD1_RPF1_ADDR_Y | Vupdt | R/W | H'0078 | H'0000_0000 | 32 | √ | √ | √ | √ |
| RPF1 Source Component C0 Address Register | FD1_RPF1_ADDR_C0 | Vupdt | R/W | H'007C | H'0000_0000 | 32 | √ | √ | √ | √ |
| RPF1 Source Component C1 Address Register | FD1_RPF1_ADDR_C1 | Vupdt | R/W | H'0080 | H'0000_0000 | 32 | √ | √ | √ | √ |
| RPF2 Source Component Y Address Register | FD1_RPF2_ADDR_Y | Vupdt | R/W | H'0084 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Still Mask Address Register | FD1_RPF_SMSK_ADDR | Vupdt | R/W | H'0090 | H'0000_0000 | 32 | √ | √ | √ | √ |
| RPF Data Swap Register | FD1_RPF_SWAP | Vupdt | R/W | H'0094 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Destination Picture Format Register | FD1_WPF_FORMAT | Vupdt | R/W | H'00C0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Destination Picture Rounding Control Register | FD1_WPF_RNDCTL | Vupdt | R/W | H'00C4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Destination Picture Stride Register | FD1_WPF_PSTRIDE | Vupdt | R/W | H'00C8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Destination Component Y Address Register | FD1_WPF_ADDR_Y | Vupdt | R/W | H'00CC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Destination Component C0 Address Register | FD1_WPF_ADDR_C0 | Vupdt | R/W | H'00D0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Destination Component C1 Address Register | FD1_WPF_ADDR_C1 | Vupdt | R/W | H'00D4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| WPF Data Swap Register | FD1_WPF_SWAP | Vupdt | R/W | H'00D8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| IPC Mode Register | FD1_IPC_MODE | Vupdt | R/W | H'0100 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Still Mask Threshold Register | FD1_IPC_SMSK_THRESH | Vupdt | R/W | H'0104 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Comb Detection Parameter Register | FD1_IPC_COMB_DET | Vupdt | R/W | H'0108 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Motion Decision Parameter Register | FD1_IPC_MOTDEC | Vupdt | R/W | H'010C | H'0000_0000 | 32 | √ | √ | √ | √ |
| DLI Blend Parameter Register | FD1_IPC_DLI_BLEND | Vupdt | R/W | H'0120 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DLI Horizontal Frequency Gain Register | FD1_IPC_DLI_HGAIN | Vupdt | R/W | H'0124 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DLI Suppression Parameter Register | FD1_IPC_DLI_SPRS | Vupdt | R/W | H'0128 | H'0000_0000 | 32 | √ | √ | √ | √ |

| Name of Registers | Abbreviation | Register Type | R/W | Address (offset from base address) | Initial value | size | Second Generation RZ/G Series Products | | | |
|--|---------------------|---------------|-----|------------------------------------|---------------|------|--|----------------------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DLI Angle Parameter Register | FD1_IPC_DLI_ANGLE | Vupdt | R/W | H'012C | H'0000_0000 | 32 | √ | √ | √ | √ |
| DLI Isolated Pixel Parameter Register0 | FD1_IPC_DLI_ISOPIX0 | Vupdt | R/W | H'0130 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DLI Isolated Pixel Parameter Register1 | FD1_IPC_DLI_ISOPIX1 | Vupdt | R/W | H'0134 | H'0000_0000 | 32 | √ | √ | √ | √ |
| IPC Sensor Threshold Register0 | FD1_IPC_SENSOR_TH0 | Vupdt | R/W | H'0140 | H'0000_0000 | 32 | √ | √ | √ | √ |
| IPC Sensor Threshold Register1 | FD1_IPC_SENSOR_TH1 | Vupdt | R/W | H'0144 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Control Register0 | FD1_SENSOR_CTL0 | Vupdt | R/W | H'0170 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Control Register1 | FD1_SENSOR_CTL1 | Vupdt | R/W | H'0174 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Control Register2 | FD1_SENSOR_CTL2 | Vupdt | R/W | H'0178 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Control Register3 | FD1_SENSOR_CTL3 | Vupdt | R/W | H'017C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register0 | FD1_SENSOR_0 | Imm | R | H'0180 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register1 | FD1_SENSOR_1 | Imm | R | H'0184 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register2 | FD1_SENSOR_2 | Imm | R | H'0188 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register3 | FD1_SENSOR_3 | Imm | R | H'018C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register4 | FD1_SENSOR_4 | Imm | R | H'0190 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register5 | FD1_SENSOR_5 | Imm | R | H'0194 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register6 | FD1_SENSOR_6 | Imm | R | H'0198 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register7 | FD1_SENSOR_7 | Imm | R | H'019C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register8 | FD1_SENSOR_8 | Imm | R | H'01A0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register9 | FD1_SENSOR_9 | Imm | R | H'01A4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register10 | FD1_SENSOR_10 | Imm | R | H'01A8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register11 | FD1_SENSOR_11 | Imm | R | H'01AC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register12 | FD1_SENSOR_12 | Imm | R | H'01B0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register13 | FD1_SENSOR_13 | Imm | R | H'01B4 | H'0000_0000 | 32 | √ | √ | √ | √ |

| Name of Registers | Abbreviation | Register Type | R/W | Address (offset from base address) | Initial value | size | Second Generation RZ/G Series Products | | | |
|-----------------------------------|---------------------------------------|---------------|-----|------------------------------------|----------------|------|--|----------------------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Sensor Register14 | FD1_SENSOR_14 | Imm | R | H'01B8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register15 | FD1_SENSOR_15 | Imm | R | H'01BC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register16 | FD1_SENSOR_16 | Imm | R | H'01C0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Sensor Register17 | FD1_SENSOR_17 | Imm | R | H'01C4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Line Memory Pixel Number Register | FD1_IPC_LMEM | Vupdt | R/W | H'01E0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| IP Internal Data | FD1_IP_INTDATA | Imm | R | H'0800 | INTDATA_INIT * | 32 | √ | √ | √ | √ |
| LUT DIF_ADJ | FD1_DIF_ADJ_00 to FD1_DIF_ADJ_FF | Imm | R/W | H'1000 to H'13FF | H'0000_0000 | 32 | √ | √ | √ | √ |
| LUT SAD_ADJ | FD1_SAD_ADJ_00 to FD1_SAD_ADJ_FF | Imm | R/W | H'1400 to H'17FF | H'0000_0000 | 32 | √ | √ | √ | √ |
| LUT BLD_GAIN | FD1_BLD_GAIN_00 to FD1_BLD_GAIN_FF | Imm | R/W | H'1800 to H'1BFF | H'0000_0000 | 32 | √ | √ | √ | √ |
| LUT DIF_GAIN | FD1_DIF_GAIN_00 to FD1_DIF_GAIN_FF | Imm | R/W | H'1C00 to H'1FFF | H'0000_0000 | 32 | √ | √ | √ | √ |
| LUT MDET | FD1_MDET_00 to FD1_MDET_FF | Imm | R/W | H'2000 to H'23FF | H'0000_0000 | 32 | √ | √ | √ | √ |

Note: * INTDATA_INIT: depend on product generation, refer to section 34.2.9.19 for detail

34.1.4 Connected Module

Following table shows the modules connected to the FDP1.

Table of connected Module

| Module | Connected Module | Description |
|---------------|-------------------------|--------------------|
| FDP1 | FCPF | FCP for FDP |

34.2 Register Description

34.2.1 Access Restrictions

FDP1 supports only 32 bit access to read or write all addresses in FDP1 area. Do not access any of otherwise access units. Therefore, read-modify-write is needed to change partial bits in 32bits register.

34.2.2 Register Type Definitions

There are 3 types of registers in FDP1: immediate register, V-update Register and V-update Status Register. The values of immediate registers are reflected to H/W behavior immediately after the changing its values. Hence, do not change these immediate registers while FDP1 is processing unless this manual allows it explicitly. The values of V-update registers are reflected to H/W behavior at the timing of V-interruption (at the timing of FD1_CTL_IRQSTA.VINT is set to 1). Users can change these V-update registers while FDP1 is processing. To reflect V-update registers to the H/W, users have to set FD1_CTL_REGEND.REGEND to 1 after the all V-update registers which are intended to be changed are completely set. The values of V-update Status Registers are updated at the timing of V-interruption. Therefore, the status (FD1_CTL_STATUS) of de-interlacing process which is conducted before a V-interruption must be referred after that V-interruption.

To confirm the types of registers, refer to Table 34.1.

34.2.3 Notational Conventions for Registers and Bit Fields

This document uses the following notational conventions for the FDP1 registers and bit fields.

1. The names of registers and bits are written in uppercase.
2. A bit or bit field in a register is indicated as [register name.bit name]. For example, the STRCMD bit in the FD1_CTL_CMD register is indicated as FD1_CTL_CMD.STRCMD.
3. Lowercase "n" in a register name or a bit name indicates an integer and the range of value n is defined when necessary. For example, FD1_RPFn_ADDR_Y.PSRC_ADDR (n = 0, 1, 2) indicates the PSRC_ADDR bits in three registers FD1_RPF0_ADDR_Y.PSRC_ADDR, FD1_RPF1_ADDR_Y.PSRC_ADDR, and FD1_RPF2_ADDR_Y.PSRC_ADDR. For RPFn when the range of value n is not defined, RPFn (n = 0, 1, 2) are assumed.
4. In each subsection for register description in section 34.2, when only a bit name is written without showing its register name, the bit is in the register described in that subsection.
5. A wildcard (*) indicates any characters in a name and represents all registers or bits that match the specified first part of a name. For example, when there are three registers FD1_RPF0_ADDR_Y, FD1_RPF0_ADDR_C0 and FD1_RPF0_ADDR_C1, FD1_RPF0_ADDR_* indicates three registers.

34.2.4 Register Classification

The FDP1 registers are arranged in the following order; the general control registers control the operation of the entire FDP1, and the other registers control each image processing and specify parameters for the processing. The functions of the registers are described in this order starting from section 34.2.6.

1. General control registers (FD1_CTL_*)
2. RPF control registers (FD1_RPF_*)
3. WPF control registers (FD1_WPF_*)
4. IPC control registers (FD1_IPC_*)

34.2.5 Restrictions on Access to Registers and Lookup Tables

The FDP1 has control registers and lookup tables. When accessing the addresses where these registers and lookup tables are allocated, the following restrictions should be satisfied. If any restriction is violated, the FDP1 will not operate correctly.

1. For the read-only bits and reserved bits in all FDP1 registers, writing 1 is prohibited unless otherwise specified.
2. Addresses undefined in section 34.1.3 are reserved areas and write access is prohibited in these areas.
3. For all immediate registers and lookup tables, except FD1_CTL_SGCMD, FD1_CTL_SRESET and FD1_CTL_IRQ*, modifying register values during operation of the module is prohibited. Modify registers while the FDP1 is stopped. For the operating status of the FDP1, refer to section 34.2.6.9.

34.2.6 General Control Registers

34.2.6.1 FDP1 Start Register (FD1_CTL_CMD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | STRCMD |
| | | | | | | | | | | | | | | | | D |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | STRCMD | B'0 | R/W | Start Command FDP1 starts the de-interlacing processing if this bit is set to 1. Set this bit after all relevant registers are surely set except for FD1_CTL_REGEND and FD1_CTL_SGCMD. Set this bit to 1 before the FD1_CTL_REGEND.REGEND and FD1_CTL_SGCMD if FDP1 has to execute process. This bit will keep the value 1 after the set. To clear this bit, write the value 0 to this bit. 0: NOP or stops process at the next timing of V-Interruption. 1: Start FDP1 process at the next timing of V-Interruption. Note that setting of this register is valid if the register FD1_CTL_REGEND.REGEND is set to the value 1 at the timing of next V-Interruption. |

34.2.6.2 Sync Generator Register (FD1_CTL_SGCMD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SGEN |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | SGEN | B'0 | R/W | V-Interruption Generator Enable This bit controls V-Interruption. This bit should be set to the value 1 after other all registers are set, and only at the first field of the sequence. Set this bit to the value 1 to execute process in all V-Interruption modes described in the section 34.2.6.5. 0: Disable V-Interruption 1: Enable V-Interruption [Important] In case of software reset (FD1_CTL_SRESET.SRST), set this SGEN bit to the value 0. If this bit is cleared during process V-interruption will not occur after FDP1 finished current frame. As a result, FDP1 will not start the next frame because STRCMD is not captured and V-update status registers will not be updated about process information of current frame. |

34.2.6.3 Register Set End Register (FD1_CTL_REGEND)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | REGEND |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | REGEND | B'0 | R/W | Register Setting Ready Flag Set the value 1 to this bit after completing the setting of parameter registers. Do not write the value 0 to this bit. This bit will be cleared by FDP1 automatically. 0: Writing 0 is prohibited except for after the completion of S/W reset sequence. 1: Registers which are set by SW is ready to be reflected for Hardware reference. The V-update status registers is ready to be reflected for Software reference. The second or later V-interruption in the case No interrupt mode can be generated. |

34.2.6.4 Channel Activation Register (FD1_CTL_CHACT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-----|-----|----|----|----|----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | SMW | WR | — | — | — | — | SMR | RD2 | RD1 | RD0 |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|------------------|-----------------------|---------------|-----|---|
| 31 to 10, 7 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 to 8, 3 to 0 | SMW, WR, SMR, RD[2:0] | All 0 | R/W | Refer to Table 34.2 for setting these register fields. The setting of each register field depends on the setting of FD1_IPC_MODE.DIM register bits. Do not set the other values which is specified in Table 34.2. |

Table 34.2 FD1_CTL_CHACT setting for FD1_IPC_MODE.DIM parameter

| Use case | FD1_CTL_CHACT | | | | | | | | | | LWord Expression |
|------------------|--|----------------|-----|----|-----|-----|-----|-----|---|-------------|------------------|
| | PR G*1 | DIM [2:0] *2 | SMW | WR | SMR | RD2 | RD1 | RD0 | | | |
| Interlace Mode | Adaptive 2D/3D | First field *4 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | H'0000_0302 |
| | | Second field | | 0 | 1 | 1 | 0 | 1 | 1 | 1 | H'0000_0307 |
| | | Final field | | 1 | 0 | 1 | 0 | 0 | 1 | 0 | H'0000_0102 |
| | | Otherwise | | 0 | 1 | 1 | 1 | 1 | 1 | 1 | H'0000_030F |
| | Fixed 3D | First field *4 | | 1 | 0 | 1 | 0 | 0 | 1 | 0 | H'0000_0102 |
| | | Second field | | 2 | 0 | 1 | 0 | 1 | 1 | 1 | H'0000_0107 |
| | | Final field | | 1 | 0 | 1 | 0 | 0 | 1 | 0 | H'0000_0102 |
| | | Otherwise | | 2 | 0 | 1 | 0 | 1 | 1 | 1 | H'0000_0107 |
| Fixed 2D | First field *4 | | 1 | 0 | 1 | 0 | 0 | 1 | 0 | H'0000_0102 | |
| | Second field | | 2 | 0 | 1 | 0 | 1 | 1 | 1 | H'0000_0107 | |
| | Final field | | 1 | 0 | 1 | 0 | 0 | 1 | 0 | H'0000_0102 | |
| | Otherwise | | 2 | 0 | 1 | 0 | 1 | 1 | 1 | H'0000_0107 | |
| Progressive Mode | Select previous field for interpolated lines | | 3 | 0 | 1 | 0 | 0 | 1 | 1 | H'0000_0103 | |
| | Select next field for interpolated lines | | 4 | 0 | 1 | 0 | 1 | 1 | 0 | H'0000_0106 | |
| | Progressive Mode | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | H'0000_0102 | |

- Notes:
1. FD1_CTL_OPMODE.PRG
 2. FD1_IPC_MODE.DIM [2:0]
 3. Value 1 can be set but the corresponding data is not read
 4. the first field input for FDP process.

34.2.6.5 Operation Mode Register (FD1_CTL_OPMODE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|-----|----|----|-----------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | PRG | — | — | VIMD[1:0] | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------------|-----------|---------------|-----|---|
| 31 to 5, 3, 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | PRG | B'0 | R/W | Progressive Mode When the register field is set to 1, the de-interlacing processing is not executed, and the incoming image data is passed through IPC module. 0: Interlace Mode 1: Progressive Mode |
| 1, 0 | VIMD[1:0] | B'00 | R/W | V-Interruption Mode This register field sets V-Sync interrupt mode of FDP1. Refer to Table 34.3 for detail. 0: Interrupt mode 1: Best Effort mode 2: No interrupt mode |

Table 34.3 V-Interrupt Mode

| VIMD[1:0] | V-Interrupt Timing | | STRCMD Captured Timing |
|--------------------|---|--|----------------------------|
| | First time | Second time or later | |
| 0 (Interrupt mode) | V-Interrupt occurs when changes FD1_CTL_SGCMD.S GEN from 0 to 1 and FDP1 is not doing de-interlacing process. | Fixed period interrupt specified by FD1_CTL_VPERIOD in spite of the de-interlacing processing | Captured at V-Interruption |
| 1 (Best effort) | | Fixed period interrupt specified by FD1_CTL_VPERIOD If the de-interlacing processing is not completed at the timing specified by FD1_CTL_VPERIOD, V-Interruption is deferred to the end of the processing. | |
| 2 (No interrupt) | | If FD1_CTL_REGEND is set 1 before the end of the processing (frame end interruption FD1_CTL_IRQSTA .FRE is set as 1) V-Interruption occurs right after the end of the processing. If FD1_CTL_REGEND is set 1 after the end of processing (frame end interruption FD1_CTL_IRQSTA .FRE is set as 1) V-Interruption occurs right after FD1_CTL_REGEND is set. If FD1_CTL_REGEND is not set 1 V-Interruption does not occur. | |

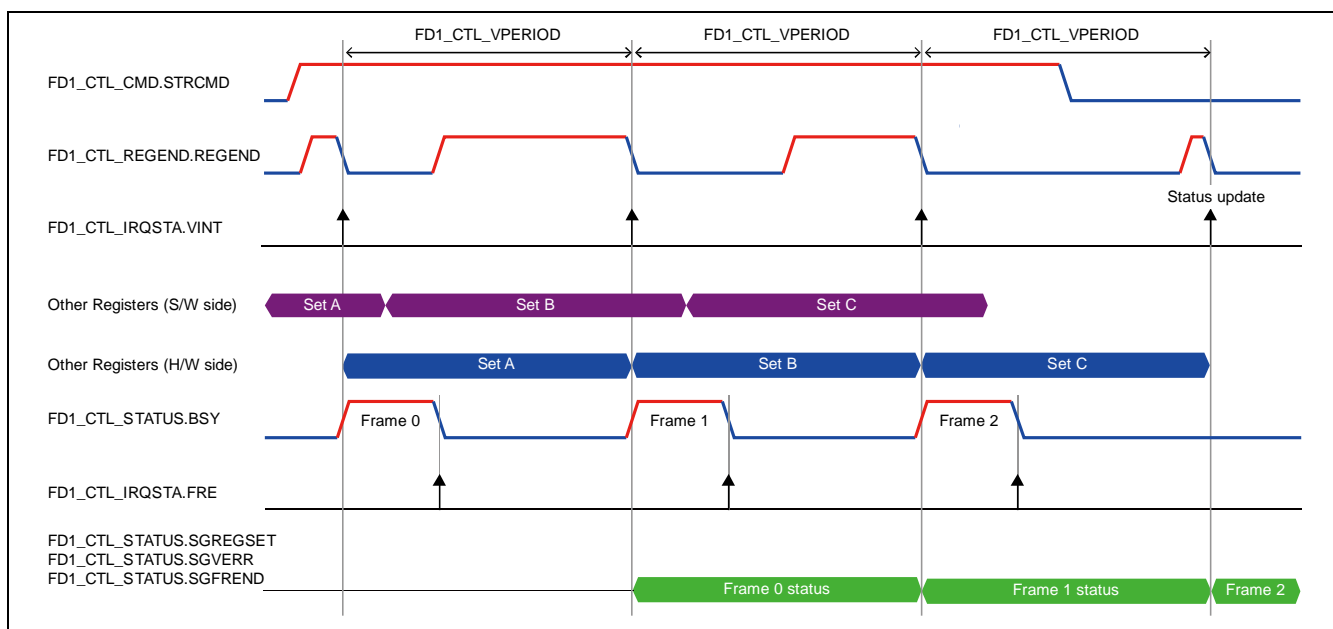


Figure 34.4 Timing Chart of Interrupt Mode (VIMD = 0) and Best Effort Mode (VIMD = 1)

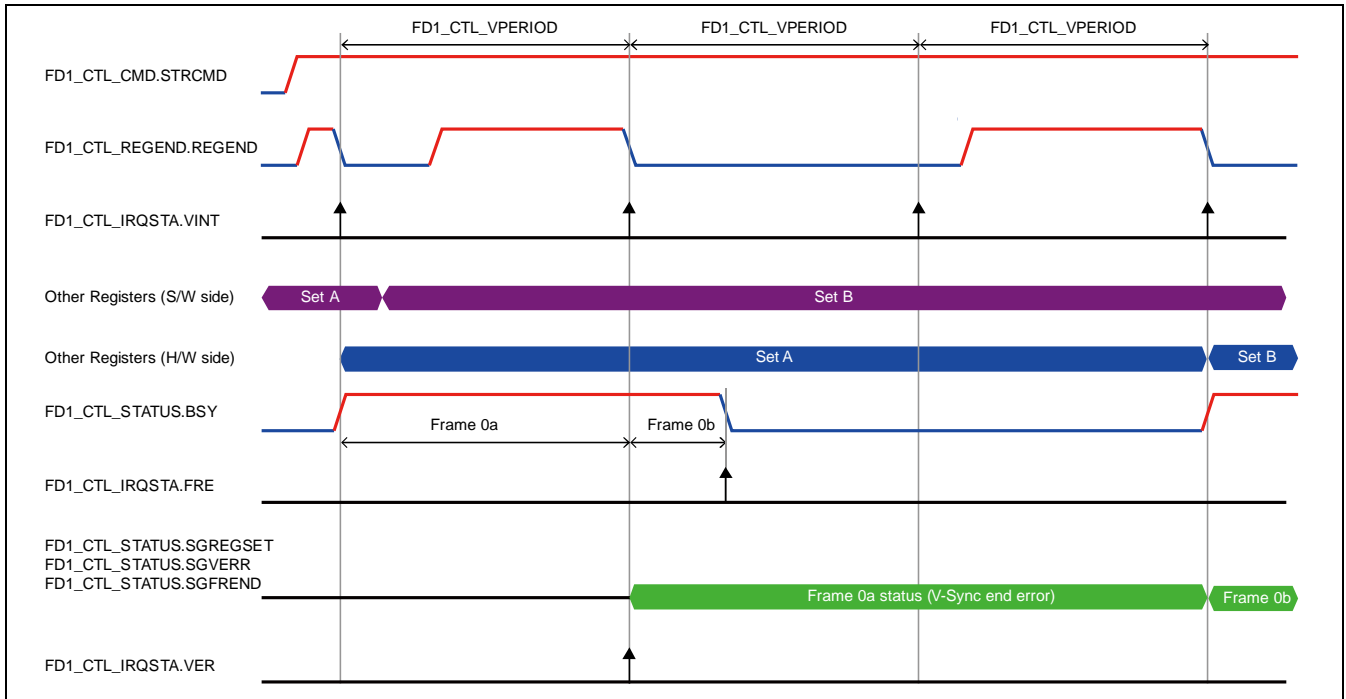


Figure 34.5 Timing Chart of Interrupt Mode (VIMD = 0) with Error

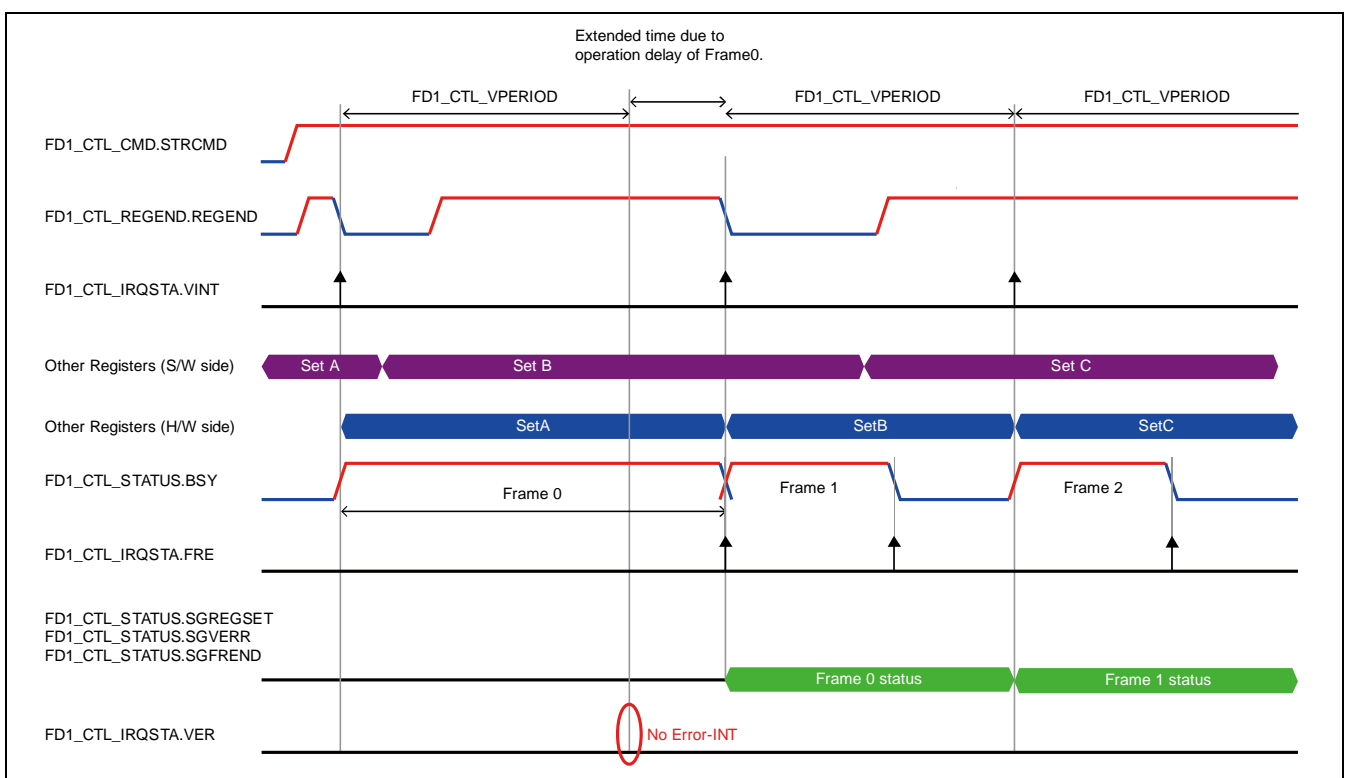


Figure 34.6 Timing Chart of Best Effort Mode (VIMD = 1) (In Case of Error at Interrupt Mode)

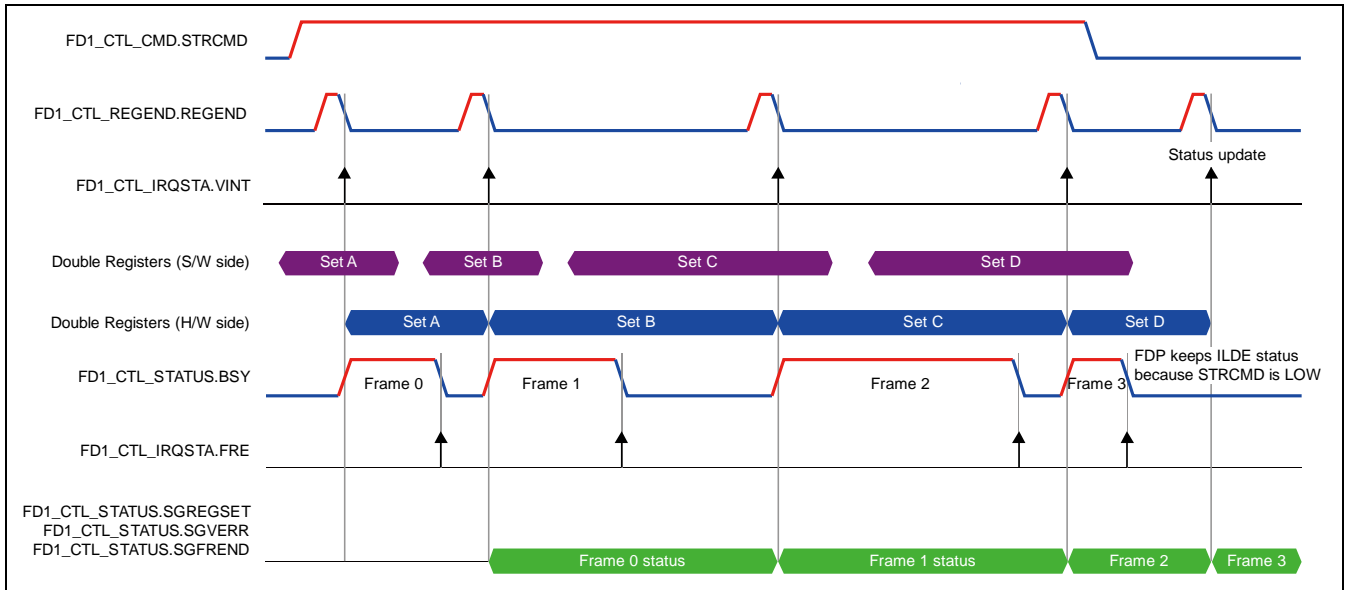
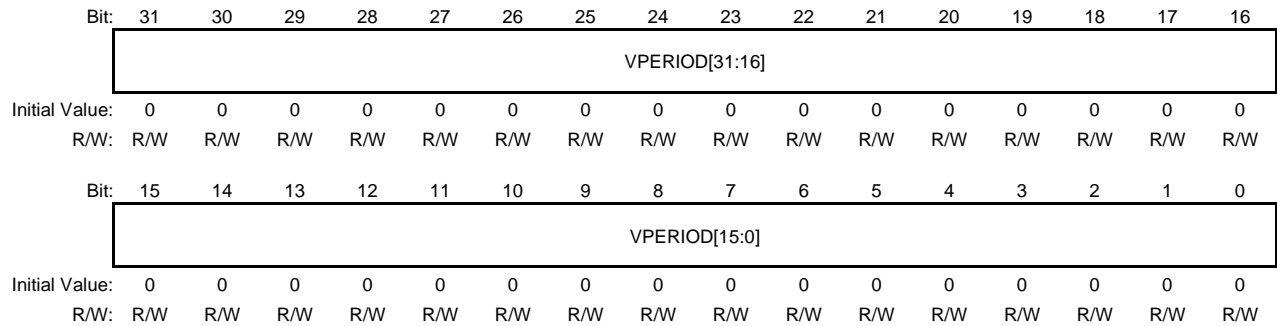


Figure 34.7 Timing Chart of No Interrupt Mode (VIMD = 2)

34.2.6.6 V-Period Register (FD1_CTL_VPERIOD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------------|---------------|-----|--|
| 31 to 0 | VPERIOD [31:0] | All 0 | R/W | V-Sync Period Set the period from the V-Sync to the next V-Sync in the unit of FDP1 clock frequency cycles. |

34.2.6.7 Clock Control Register (FD1_CTL_CLKCTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CSTP_N |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | CSTP_N | B'0 | R/W | Set the value 1 to this bit. |

34.2.6.8 Software Reset Register (FD1_CTL_SRESET)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SRST |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | SRST | B'0 | R/W | <p>FDP1 Software Reset</p> <p>FDP1 stops its operation after this bit is set to 1. FDP1 aborts all processing and quits bus/ memory accesses. Do not start next process by FD1_CTL_CMD until the software processing is completed. When this processing is completed, the FD1_CTL_IRQSTA.FRE interrupt source bit is set to 1, when the FRE interrupt is enabled. This interrupt notifies the end of the reset processing.</p> <p>The end of software reset processing is notified through the FRE bit, but the software reset issued while FDP1 is stopped is ignored as NOP. As it takes a while until the reset is actually issued after the reset bit is set, the FDP1 may complete operation before the reset is actually issued. In this case, no interrupt is output for the software reset that is issued after the FDP1 completes operation. But even through in this case, not that the output image may be destroyed due to software reset process.</p> <p>This bit is always read as 0.</p> <p>0: NOP 1: Software Reset</p> <p>[Important] Set FD1_CTL_SGCM.SGEN to the value 0 before write 1 to this SRST bit.</p> |

34.2.6.9 Operating Status Register (FD1_CTL_STATUS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------------|----|----|----|----|--------------|------------|-------------|----|----|----|----|----|----|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | VINT_CNT [15:0] | | | | | | | | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | SGREG SET | SGVER R | SGFRE ND | — | — | — | — | — | — | — | BSY |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------------------|-----------------|---------------|-----|---|
| 15 to 11, 7 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 31 to 16 | VINT_CNT [15:0] | All 0 | R | V-Sync Interrupt Counter Status This bit field shows the number of interruption times of V-Sync interrupt. Hence, the value of this bit indicates the number of progressive output frames which is generated by FDP1. This counter value is reset to 0 when FD1_CTL_SGCMD.SGEN is set to 1. |
| 10 | SGREGSET | B'0 | R | Register Set End Status This bit represents the register set status updated at V-Sync. The status is controlled by FD1_CTL_REGEND.REGEND bit and the internal timing. 0: Registers are not set for the operation 1: Registers are set for the operation |
| 9 | SGVERR | B'0 | R | V-Sync End Error Status This bit represents the V-Sync end error status updated at V-Sync. 0: No error 1: V-Sync end error (De-interlacing cannot be finished by the timing of V-Sync) |
| 8 | SGFREND | B'0 | R | Frame End Status This bit represents the frame end status updated at V-Sync. 0: The de-interlacing is not finished 1: Frame end (The de-interlacing is finished) |
| 0 | BSY | B'0 | R | FDP1 Operating Status This bit indicates the operating or stopped state of control FDP1. 0: FDP1 is stopped. 1: FDP1 is operating. |

34.2.6.10 V-Cycles Status Register (FD1_CTL_VCYCLE_STAT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | VCYC[31:16] | | | | | | | | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | VCYC[15:0] | | | | | | | | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|---|
| 31 to 0 | VCYC [31:0] | All 0 | R | Number of Cycles of the Previous Frame Processing This status register shows the number of cycles of previous frame de-interlacing processing. FDP1 updates this register at the timing of V-Sync interrupt. |

34.2.6.11 Interrupt Enable Register (FD1_CTL_IRQENB)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|-------|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | VERE |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | VINTE | — | — | — | FREE |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------------------------|----------|---------------|-----|---|
| 31 to 17, 15 to 5, 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | VERE | B'0 | R/W | Interrupt Enable for V-Sync End Error The interrupt of V-Sync end error is asserted at the timing that FDP1 does not complete the de-interlacing processing at the V-Sync. |
| 4 | VINTE | B'0 | R/W | Interrupt Enable for V-Sync The interrupt of V-Sync is asserted at the timing of V-Sync specified by FD1_CTL_OPMODE.VIMD register field. If VIMD is set to 2, this interrupt is not asserted. |
| 0 | FREE | B'0 | R/W | Interrupt Enable for Frame End The interrupt of frame end is asserted at the timing that FDP1 completes the de-interlacing processing. |

Each bit controls the interrupt enable of the corresponding interrupt source.

0: Interrupt Disabled

1: Interrupt Enabled

Each bit in FD1_CTL_IRQSTA is set to 1 when the corresponding interrupt source is generated. FD1_CTL_IRQENB specifies whether to output an interrupt signal for the generated source. When an interrupt is disabled in this register, no interrupt signal is generated even when the corresponding bit in FD1_CTL_IRQSTA is set to 1. When an interrupt is enabled in this register, an interrupt signal is output when the corresponding bit in FD1_CTL_IRQSTA is set to 1.

34.2.6.12 Interrupt Status Register (FD1_CTL_IRQSTA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|------|----|----|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | VER |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | VINT | — | — | — | FRE |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------------------------|----------|---------------|-----|--|
| 31 to 17, 15 to 5, 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | VER | B'0 | R/W | Interrupt Status and Clear for V-Sync End Error |
| 4 | VINT | B'0 | R/W | Interrupt Status and Clear for V-Sync |
| 0 | FRE | B'0 | R/W | Interrupt Status and Clear for Frame End |

The read value from each bit is the status of the interrupt source, and the write access to each bit controls the interrupt status.

[Read Access] Interrupt Status

- 0: No interrupt
- 1: Interrupt activated

[Write Access] Interrupt Clear

- 0: The interrupt status is cleared to 0
- 1: Hold the interrupt status value

FD1_CTL_IRQSTA indicates the state of the interrupt sources generated in the FDP1. Whether to output a FDP1 interrupt when an interrupt source is generated and the corresponding bit is set to 1 is determined by the corresponding bit setting in FD1_CTL_IRQENB.

While an interrupt is disabled in FD1_CTL_IRQENB, the FDP1 does not output an interrupt signal even when an interrupt source is generated, but the source flag in this register is set to 1.

34.2.6.13 Interrupt Control Register (FD1_CTL_IRQFSET)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | FRES |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | FRES | B'0 | R/W | This bit is used to issue the dummy frame end interruption. [Read Access] Always 0 value is read. [Write Access] 0: Hold the current interrupt status value 1: The interrupt status of frame end (bit FRE of register FD1_CTL_IRQSTA) is forced to 1. After being forced to 1, the status value 1 of interrupt status bit FRE is hold until it is cleared by being written 0. The bit FRES turns to 0 automatically after being written 1. |

34.2.7 RPF Control Registers

34.2.7.1 Source Picture Size Register (FD1_RPF_SIZE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | HSIZE[12:0] | | | | | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | VSIZE[12:0] | | | | | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|-------------|---------------|-----|--|
| 31 to 29, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | HSIZE[12:0] | All 0 | R/W | Horizontal Source Picture Size Specify the horizontal picture size of each input field in unit of pixels. |
| 12 to 0 | VSIZE[12:0] | All 0 | R/W | Vertical Source Picture Size Specify the vertical picture size of each input field in unit of pixels. |

34.2.7.2 Source Picture Format Register (FD1_RPF_FORMAT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|--------|--------|----|----|----|-----|----|------------|-----|-----|-----|-----|-----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CIPM |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | RSPYCS | RSPUVS | — | — | — | CF | — | RDFMT[6:0] | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R | R | R | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---|----------|---------------|-----|---|
| 31 to 17, 15 to 14, 11 to 9, 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | CIPM | B'0 | R/W | Horizontal Chrominance Interpolation Method Setting Image data is processed in the YCbCr444 format inside RPF in case of YCbCr color space. When the chrominance format of the input image is YCbCr422 or YCbCr420, data is up-sampled as shown in Figure 34.8. This bit specifies the method of up-sampling for this purpose. 0: The nearest-neighbor method is used for horizontal chrominance interpolation. 1: The bilinear method is used for horizontal chrominance interpolation. In the case all following conditions are true set 0 to this bit in order to prevent the image difference between input and output. If set 1 to this bit in this case the output image may be different from the input image, but it is not broken. 1) Operation mode is progressive (FD1_CTL_OPMODE.PRG = 1) 2) Input format is either YCbCr4:2:2 or YCbCr4:2:0 3) Output format is either YCbCr4:2:2 or YCbCr4:2:0 |
| 13 | RSPYCS | B'0 | R/W | RPF Input Mode Setting 1 When the input format is YUY2, set this bit to 1 and set the RDFM T bits to 71 (H'47). When the input format is YVYU, set this bit and the RSPUVS bit to 1 and set the RDFM T bits to 71 (H'47). In other cases, set this bit to 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-------------|---------------|-----|--|
| 12 | RSPUVS | B'0 | R/W | <p>RPF Input Mode Setting 2</p> <p>When the input format is NV61, set this bit to 1 and set the RDFM T bits to 65 (H'41).</p> <p>When the input format is NV21, set this bit to 1 and set the RDFM T bits to 66 (H'42).</p> <p>When the input format is YVYU, set this bit and the RSPYCS bit to 1 and set the RDFM T bits to 71 (H'47).</p> <p>In other cases, set this bit to 0.</p> |
| 8 | CF | B'0 | R/W | <p>Current Field</p> <p>This bit specifies current field parity. Note that the Previous or Next field pictures should be the opposite parity of the current field specified by this bit.</p> <p>0: Current field is top field 1: Current field Bottom field</p> |
| 6 to 0 | RDFMT [6:0] | All 0 | R/W | <p>RPF Input Image Format Setting</p> <p>These bits select the format of the image input from the external SDRAM to the RPFn (n = 0, 1, 2). Select a value corresponding to the desired format from those shown in Table 34.4.</p> <p>[Note 1] Number of input pixels When YCbCr4:2:2 is selected through the RDFMT bits, the horizontal size of the input image should be specified in 2-pixel units. When YCbCr4:2:0 is selected, the vertical and horizontal sizes should be specified in 2-pixel units.</p> <p>[Note 2] Input chroma horizontal up-sampling When the input format is YCbCr4:2:2 or YCbCr4:2:0 the chroma component is up-sampled horizontally to YCbCr4:4:4 (the format used for image process) with the method specified by the CIPM bit. In the interlace mode (FD1_CTL_OPMODE.PRG = 0) this up-sampling is processed before IP conversion.</p> <p>[Note 3] Input chroma vertical up-sampling When the input format is YCbCr4:2:0 the chroma component is up-sampled vertically to YCbCr4:4:4 (the format used for image process) by copying the neighbor upper input line. In the interlace mode (FD1_CTL_OPMODE.PRG = 0) this up-sampling is processed before IP conversion.</p> |

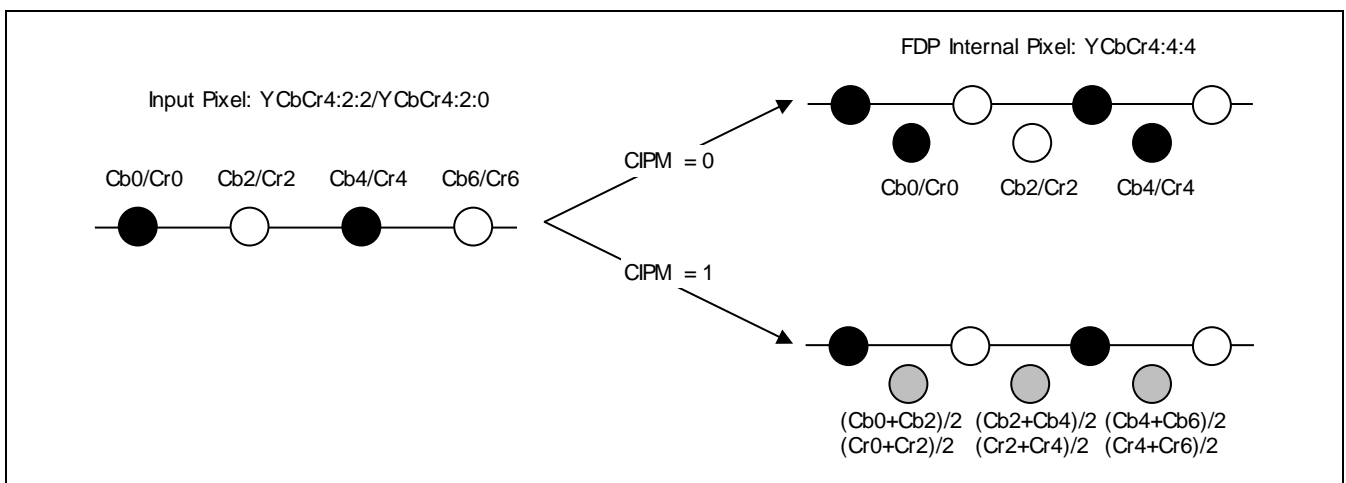


Figure 34.8. Chrominance Interpolation Methods Selectable through CIPM Setting

Table 34.4 YCbCr Formats for RPF Input*4 and WPF Output

| RDFMT [6:0], WRFMT[6:0] | Packed YCbCr input/output Format | Reference |
|------------------------------------|---|------------------|
| H'00 to H'3F | Reserved | — |
| H'40 | YCbCr4:4:4 semi-planar | Figure 34.9 |
| H'41 | YCbCr4:2:2 semi-planar (NV16, NV61*1) | |
| H'42 | YCbCr4:2:0 semi-planar (NV12, NV21*1) | |
| H'43 to H'45 | Reserved | — |
| H'46 | YCbCr4:4:4 interleaved | Figure 34.10 |
| H'47 | YCbCr4:2:2 interleaved type 0 (UYVY, YUY2*2, YVYU*3) | |
| H'48 | YCbCr4:2:2 interleaved type 1 | |
| H'49 | Reserved | |
| H'4A | YCbCr4:4:4 planar | Figure 34.11 |
| H'4B | YCbCr4:2:2 planar (YV16) | |
| H'4C | YCbCr4:2:0 planar (YV12, I420) | |
| H'4D to H'7F | Reserved | — |

Notes: 1. when the input format is NV61 or NV21, set the RSPUVS bit to 1.
 2. when the input format is YUY2, set the RSPYCS bit to 1.
 3. when the input format is YVYU, set the RSPUVS bit to 1 and RSPYCS bit to 1.
 4. Chroma component (Cb, Cr or U, V) of RPF0, RPF2 is not read from bus in case of YCbCr planar, YCbCr semi-planar formats, and discarded after reading in package with luma component in case YCbCr interleaved formats because it is processed in fixed 2D de-interlacing.

In write format case shown in section 34.2.8.1, FD1_WPF_FORMAT.WSPYCS, WSPUVS are corresponds to RSPYCS and RSPUVS, respectively.

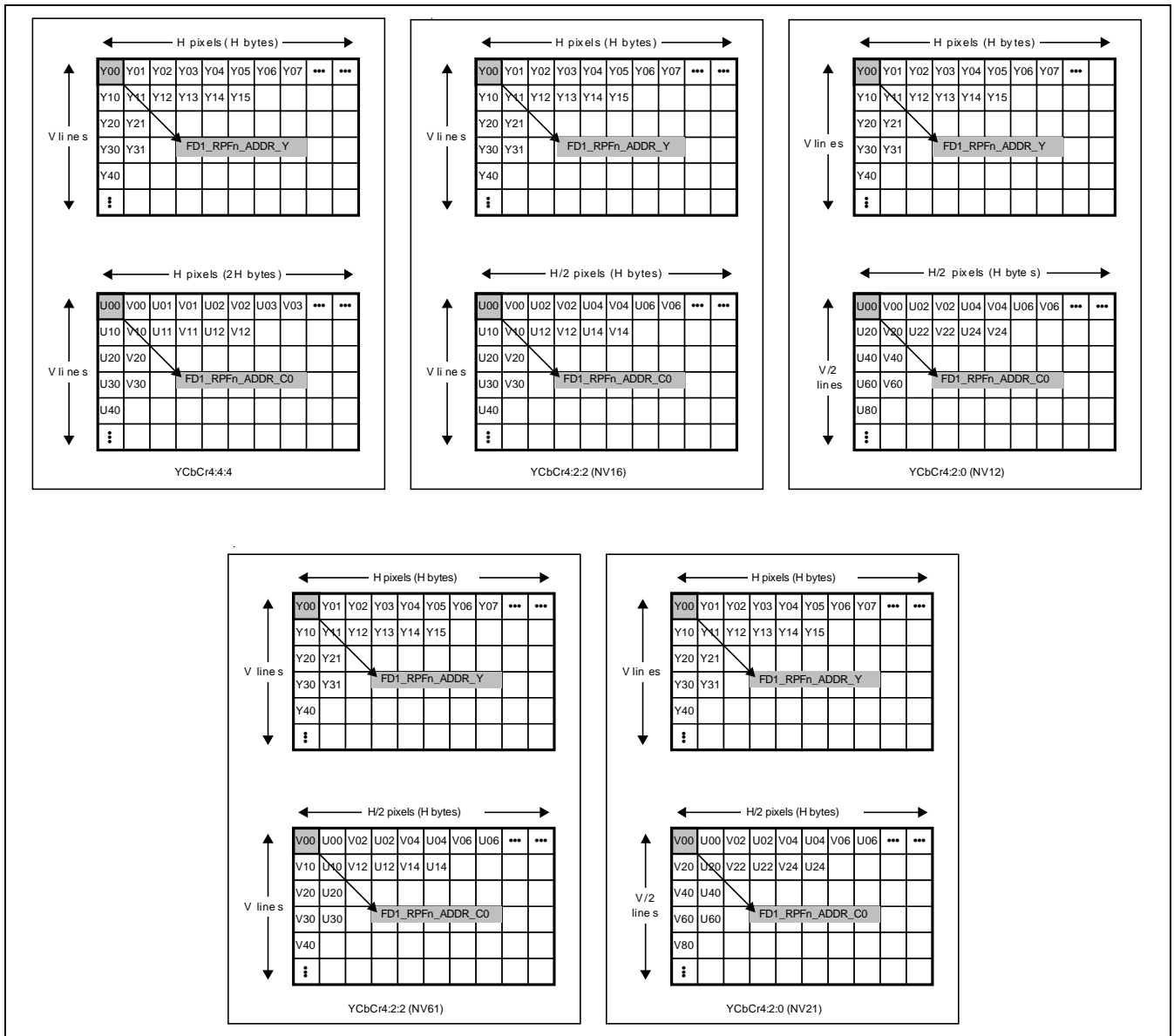


Figure 34.9. YCbCr Semi-Planar Formats

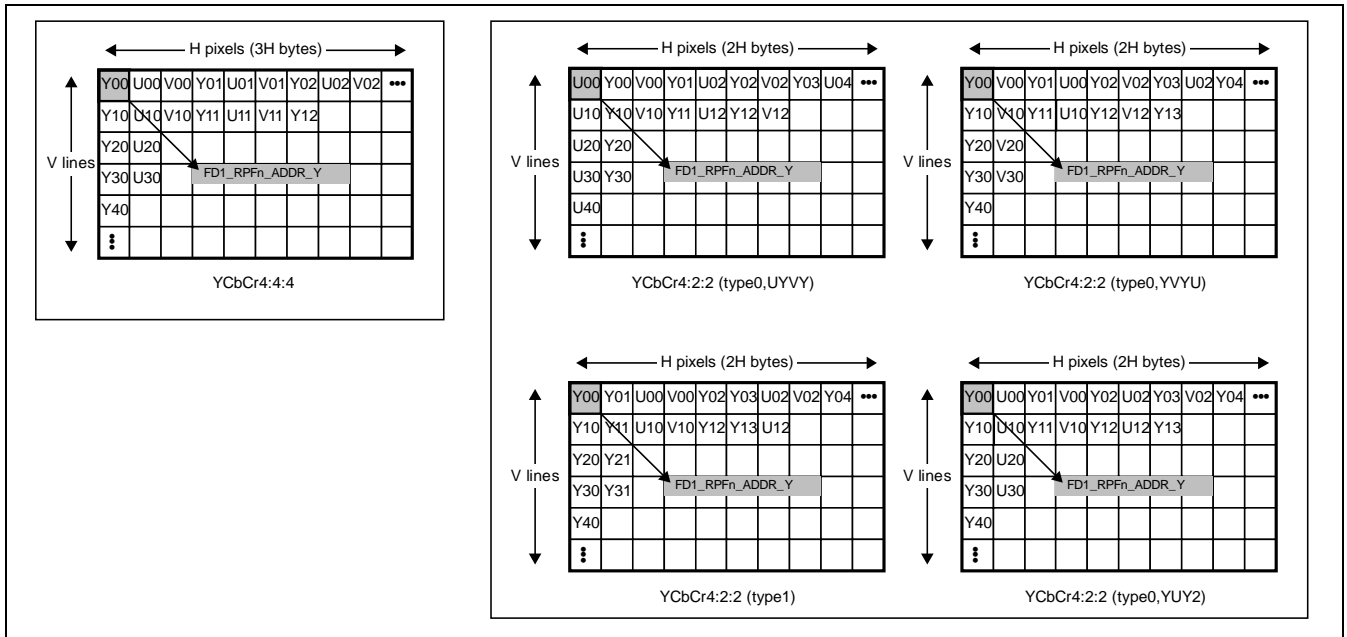


Figure 34.10. YCbCr Interleaved Formats

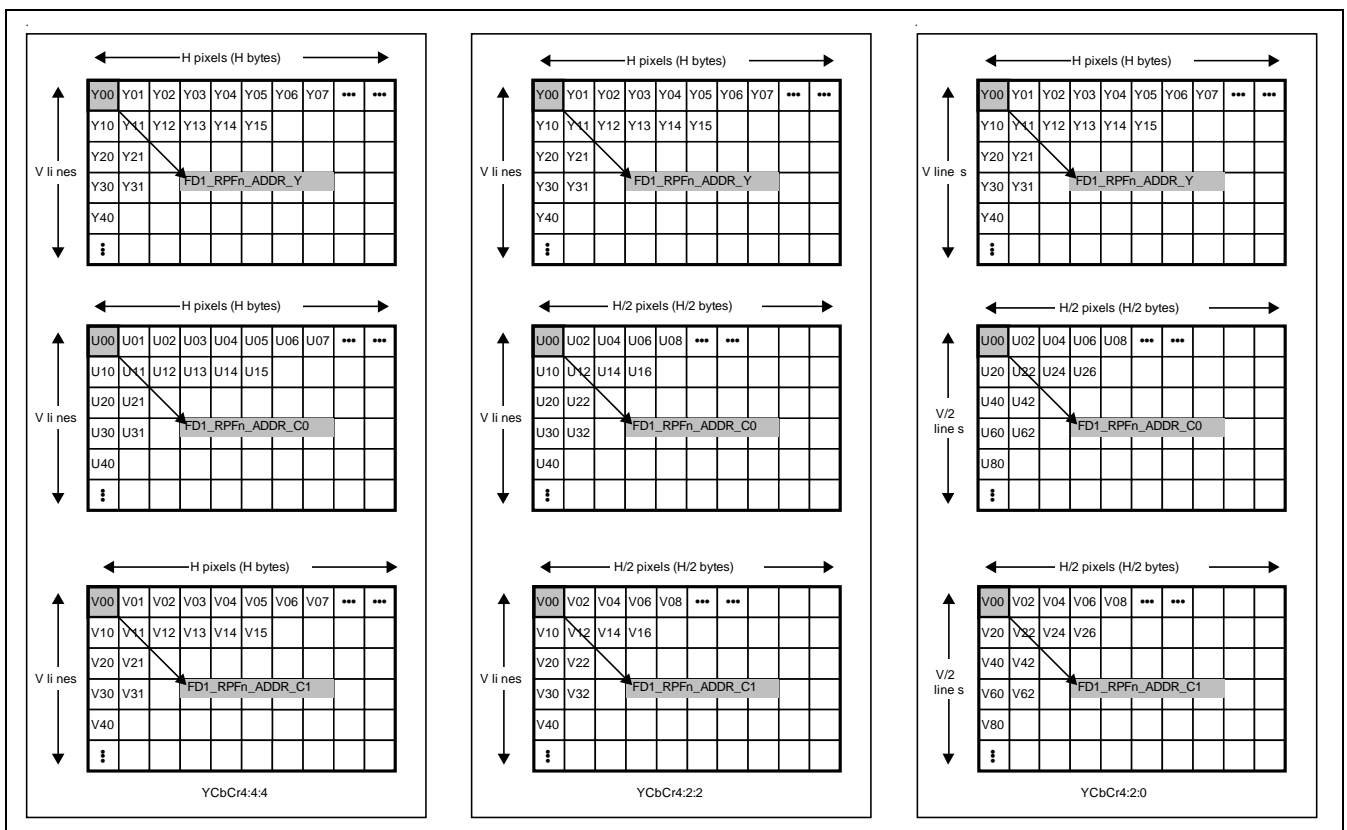


Figure 34.11. YCbCr Planar Formats

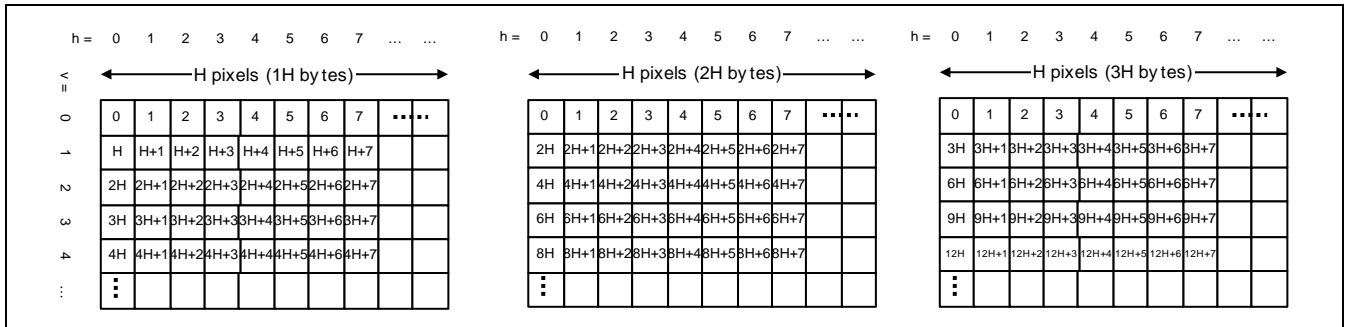
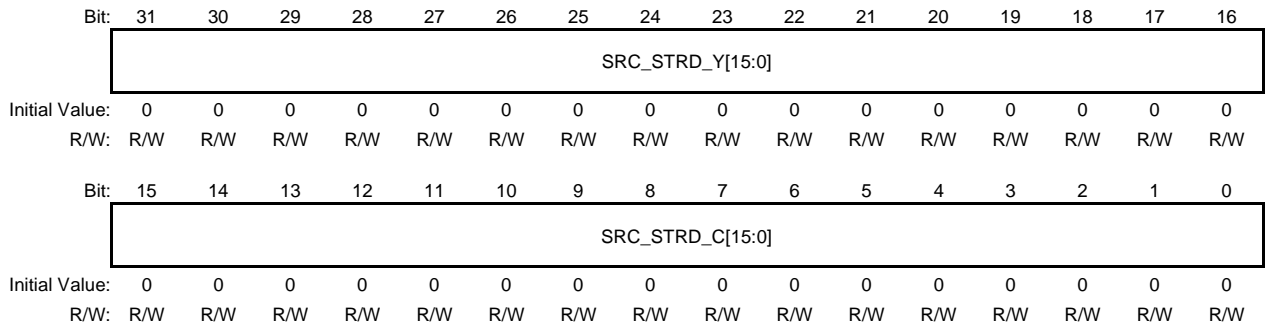


Figure 34.12. Memory address for the corresponding pixel's position

Legend of Figure 34.12, H is horizontal image size in pixel unit. 1H, 2H and 3H is corresponding to the case 1 pixel has 1byte, 2bytes, 3bytes of data respectively.

34.2.7.3 Source Picture Stride Register (FD1_RPF_PSTRIDE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



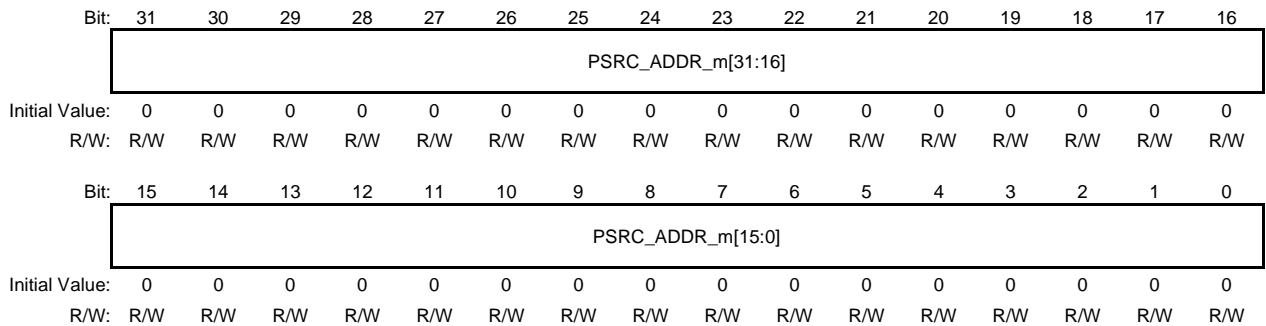
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------|---------------|-----|---|
| 31 to 16 | SRC_STRD_Y [15:0] | All 0 | R/W | Memory Stride of Source Picture Y Plane These bits specify in 1-byte units the memory stride of the source picture Y plane read by the RPFn (n = 0, 1, 2). A value from H'0020 to H'1FFE can be specified. |
| 15 to 0 | SRC_STRD_C [15:0] | All 0 | R/W | Memory Stride of Source Picture C Plane These bits specify in 1-byte units the memory stride of the source picture C plane read by the RPF1. A value from H'0020 to H'1FFE can be specified. In the YCbCr planar format, this setting is used as the memory stride of the Cb and Cr planes. |

[Note]

As described above, stride value of source pictures (past, current, future) are common one. Therefore, frame structures of all input pictures should be the same format. For example, the case that past picture which is supplied to RPF0 is frame-structure, and other input pictures to RPF1 and 2 are field picture, is prohibited.

34.2.7.4 RPFn Source Component-m Address Register (FD1_RPFn_ADDR_m: {n,m} is one of {0,Y},{1,Y},{1,C0},{1,C1},{2,Y})

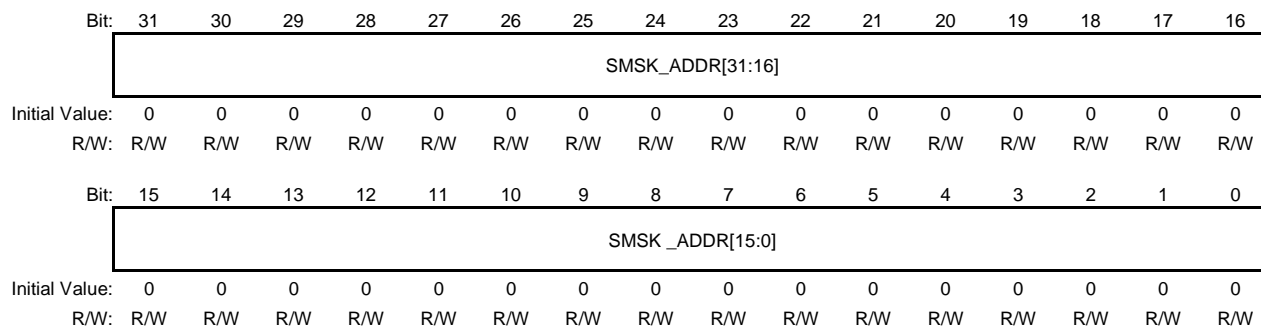
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------|---------------|-----|--|
| 31 to 0 | PSRC_ADDR_m [31:0] | All 0 | R/W | <p>RPFn Source Field Address for Component-m ({n,m} is one of {0,Y},{1,Y},{1,C0},{1,C1},{2,Y})</p> <p>These bits specify in 1-byte units the start address of the source component-m plane read by the RPFn.</p> <p>A value from H'0000_0000 to H'FFFF_FFFF can be specified.</p> <p>Set same value with LSB 4bit of RPFn source field address of component C0 to LSB 4bit of RPFn source field address of component C1, when input image format is Planar(FD1_RPF_FORMAT.RDFMT = 74,75,76).</p> |

34.2.7.5 Still Mask Address Register (FD1_RPF_SMSK_ADDR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------|---------------|-----|--|
| 31 to 0 | SMSK_ADDR [31:0] | All 0 | R/W | <p>Still Mask Buffer Address.</p> <p>These bits specify the memory address which the still mask data is located. This bit field should take the value in the range of H'0000_0000 to H'FFFF_FFFF.</p> <p>Still mask is intermediate data for IP conversion only when adaptive 2D/3D (FD1_CTL_OPMODE.PRG = 0 and FD1_IPC_MODE.DIM [2:0] = 0). This data is active when one of the two bits FD1_CTL_CHACT.SMW and FD1_CTL_CHACT.SMR is set as 1. This data is generated and re-used by FDP. It should be kept on SDRAM.</p> <p>FDP1 needs two separated buffers of the same size for this data, one is common for all top fields and the other is common for all bottom fields. Therefore, the source address must be different between top and bottom field. When the register bit FD1_RPF_FORMAT.CF is 0 set the address of buffer of top filed to register. Otherwise set the address of buffer of bottom filed to register.</p> <p>The memory stride of each buffer is expressed as the following equation.</p> $2 \times \{(FD1_RPF_SIZE.HSIZE + 7)/8\} \text{ [byte]}$ <p>The memory space of each buffer is expressed as below.</p> $2 \times \{(FD1_RPF_SIZE.HSIZE + 7)/8\} \times FD1_RPF_SIZE.VSIZE \text{ [byte]}$ <p>The Figure 34.13 shows the usage of still mask data buffers and the Table 34.5 shows how to set the register of still mask data buffer address.</p> |

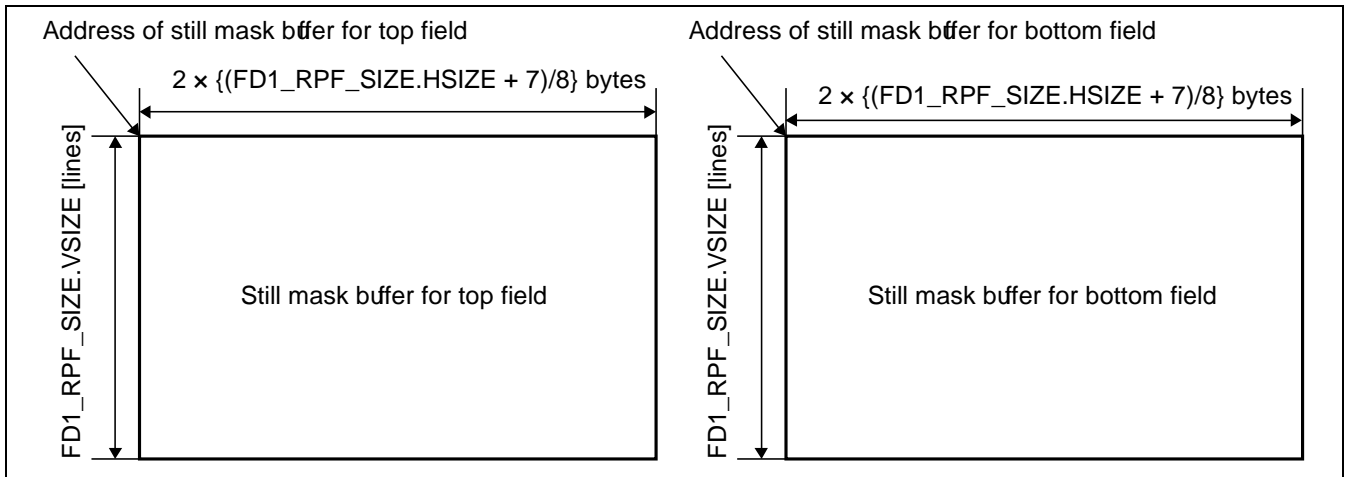


Figure 34.13 Usage of still mask buffers

Table 34.5 Setting of still mask buffer address register

| Current Field Parity | FD1_RPF_FORMAT.CF | FD1_RPF_SMSK_ADDR.SMSK_ADDR |
|----------------------|-------------------|---|
| Top Field | 0 | Address of Still Mask buffer for Top Field |
| Bottom Field | 1 | Address of Still Mask buffer for Bottom Field |

34.2.7.6 RPF Data Swap Register (FD1_RPF_SWAP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | ISWAP[3:0] | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|--|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | ISWAP [3:0] | H'0 | R/W | Input Swap Setting for Image Data This bit field specifies the data swapping function. Each bit corresponds to the following data swapping. Bit[3]: Long Long Word (64-bit) Swap Bit[2]: Long Word (32-bit) Swap Bit[1]: Word (16-bit) Swap Bit[0]: Byte (8-bit) Swap |

When data order in memory for each format is defined as same as Figure 34.9 to Figure 34.11 set 4'b1111 to ISWAP[3:0]. If data order in memory is different from definitions of these figures, it can be adjusted to definitions of these figures in 16 bytes unit as shown in Table 34.6.

Note: Follow the section “34.4 FCP for FDP (FCPF)” about setting this swap register.

Table 34.6 Changing data order according to swap register

| Data order in memory | | | | | | | | | | | | | | | | ISWAP[3:0]/OSWAP[3:0] | | | | |
|----------------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|---------|---------|---------|---------|-----------------------|------|------|------|------|
| Byte address | 16n+ 0 | 16n+ 1 | 16n+ 2 | 16n+ 3 | 16n+ 4 | 16n+ 5 | 16n+ 6 | 16n+ 7 | 16n+ 8 | 16n+ 9 | 16n+ 10 | 16n+ 11 | 16n+ 12 | 16n+ 13 | 16n+ 14 | 16n+ 15 | BIT3 | BIT2 | BIT1 | BIT0 |
| Data | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 1 | 1 | 1 | 1 |
| | 1 | 0 | 3 | 2 | 5 | 4 | 7 | 6 | 9 | 8 | 11 | 10 | 13 | 12 | 15 | 14 | 1 | 1 | 1 | 0 |
| | 2 | 3 | 0 | 1 | 6 | 7 | 4 | 5 | 10 | 11 | 8 | 9 | 14 | 15 | 12 | 13 | 1 | 1 | 0 | 1 |
| | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 11 | 10 | 9 | 8 | 15 | 14 | 13 | 12 | 1 | 1 | 0 | 0 |
| | 4 | 5 | 6 | 7 | 0 | 1 | 2 | 3 | 12 | 13 | 14 | 15 | 8 | 9 | 10 | 11 | 1 | 0 | 1 | 1 |
| | 5 | 4 | 7 | 6 | 1 | 0 | 3 | 2 | 13 | 12 | 15 | 14 | 9 | 8 | 11 | 10 | 1 | 0 | 1 | 0 |
| | 6 | 7 | 4 | 5 | 2 | 3 | 0 | 1 | 14 | 15 | 12 | 13 | 10 | 11 | 8 | 9 | 1 | 0 | 0 | 1 |
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 1 | 0 | 0 | 0 |
| | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 0 | 1 | 1 | 1 |
| | 9 | 8 | 11 | 10 | 13 | 12 | 15 | 14 | 1 | 0 | 3 | 2 | 5 | 4 | 7 | 6 | 0 | 1 | 1 | 0 |
| | 10 | 11 | 8 | 9 | 14 | 15 | 12 | 13 | 2 | 3 | 0 | 1 | 6 | 7 | 4 | 5 | 0 | 1 | 0 | 1 |
| | 11 | 10 | 9 | 8 | 15 | 14 | 13 | 12 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 0 | 1 | 0 | 0 |
| | 12 | 13 | 14 | 15 | 8 | 9 | 10 | 11 | 4 | 5 | 6 | 7 | 0 | 1 | 2 | 3 | 0 | 0 | 1 | 1 |
| | 13 | 12 | 15 | 14 | 9 | 8 | 11 | 10 | 5 | 4 | 7 | 6 | 1 | 0 | 3 | 2 | 0 | 0 | 1 | 0 |
| | 14 | 15 | 12 | 13 | 10 | 11 | 8 | 9 | 6 | 7 | 4 | 5 | 2 | 3 | 0 | 1 | 0 | 0 | 0 | 1 |
| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 0 | 0 | 0 | 0 |



Data order defined in Table 34.7 for RGB format and Figure 34.9 to Figure 34.11 for YCbCr format

| Byte address | 16n +0 | 16n +1 | 16n+ 2 | 16n+ 3 | 16n+ 4 | 16n+ 5 | 16n+ 6 | 16n+ 7 | 16n+ 8 | 16n+ 9 | 16n+ 10 | 16n+ 11 | 16n+ 12 | 16n+ 13 | 16n+ 14 | 16n+ 15 |
|--------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|---------|---------|---------|---------|---------|
| Data | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |

34.2.8 WPF Control Registers

34.2.8.1 Destination Picture Format Register (FD1_WPF_FORMAT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----------|--------|-----------|-----|-----------|-----|-----|-----|----|------------|------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | PDV[7:0] | | | | | | | — | — | — | FCNL | — | — | — | — | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | WSPYCS | WSPUVS | DITH[1:0] | | WRTM[2:0] | | | CSC | — | WRFMT[6:0] | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----------------------|----------|---------------|-----|--|
| 23 to 21, 19 to 16, 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 31 to 24 | PDV[7:0] | All 0 | R/W | PAD Value in Output Packed Data These bits specify the value to be stored in the bit field indicated as PAD or P in the output formats shown in Table 34.7. A value from 0 to 255 can be specified. |
| 20 | FCNL | B'0 | R/W | Frame Compress Near Lossless in FCPF 0: Not compress 1: Compress Although this bit is register of FCPF, the setting timing is same with other bits of FD1_WPF_FORMAT. |
| 15 | WSPYCS | B'0 | R/W | WPF Output Mode Setting 1 When the output format is YUY2, set this bit to 1 and set the WRFMT bits to 71 (H'47). When the output format is YVYU, set this bit and the WSPUVS bit to 1 and set the WRFMT bits to 71 (H'47). In other cases, set this bit to 0. |
| 14 | WSPUVS | B'0 | R/W | WPF Output Mode Setting 2 When the output format is NV61, set this bit to 1 and set the WRFMT bits to 65 (H'41). When the output format is NV21, set this bit to 1 and set the WRFMT bits to 66 (H'42). When the output format is YVYU, set this bit and the WSPYCS bit to 1 and set the WRFMT bits to 71 (H'47). In other cases, set this bit to 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 13 to 12 | DITH[1:0] | B'00 | R/W | <p>Dithering Enable/Disable</p> <p>When the output format specified through the WRFMT bits is RGB with 18 bpp (262144 colors) or less, the color reduction processing is applied to match the number of colors. The color reduction processing may generate the artifacts of pseudo gradation, which can be suppressed through dithering. The DITH bits enable or disable dithering during color reduction.</p> <p>When the output format specified through the WRFMT bits is YCbCr, specify 0 in these bits.</p> <p>0: Dithering is disabled 3: Dithering is enabled 1, 2: Setting prohibited</p> |
| 11 to 9 | WRTM[2:0] | B'000 | R/W | <p>CSC Conversion Expression Setting</p> <p>These bits select the expression for color space conversion. The conversion direction is RGB to YCbCr when the format specified in the WRFMT bits is RGB, or YCbCr to RGB when the format is YCbCr.</p> <p>0: BT.601 YCbCr [16,235/240] → RGB [0,255] 1: BT.601 YCbCr [0,255] → RGB [0,255] 2: BT.709 YCbCr [16,235/240] → RGB [0,255] 3: BT.709 YCbCr [16,235/240] → RGB [16,235] 4 to 7: Setting prohibited</p> |
| 8 | CSC | B'0 | R/W | <p>Color Space Conversion Setting</p> <p>Enables or disables YCbCr ↔ RGB color space conversion to be executed in the WPF. The characteristics of color space conversion are determined by the WRTM setting.</p> <p>0: Color space is not converted. 1: Color space is converted.</p> |
| 6 to 0 | WRFMT [6:0] | All 0 | R/W | <p>WPF Output Image Format Setting</p> <p>These bits select the format of the image output from the WPF to the external memory from among those listed in Table 34.7 and Table 34.4.</p> <p>[Note 1] Number of output pixels When YCbCr4:2:2 is specified through WRFMT, the horizontal size of the output image should be a multiple of 2 pixels. When YCbCr4:2:0 is specified, the vertical and horizontal sizes of the output image should be multiples of 2 pixels.</p> <p>[Note 2] Output chroma horizontal down-sampling When the output format is YCbCr4:2:2 or YCbCr4:2:0 the chroma component is down-sampled horizontally from YCbCr4:4:4 (the format of image process result) by averaging the values of neighbor two pixels. In the interlace mode (FD1_CTL_OPMODE.PRG = 0) this down-sampling is processed after IP conversion.</p> <p>[Note 3] Output chroma vertical down-sampling When the output format is YCbCr4:2:0 the chroma component is down-sampled vertically from YCbCr4:4:4 (the format of image process result) by outputting only the even number line (line 0, 2, etc). In the interlace mode (FD1_CTL_OPMODE.PRG = 0) this down-sampling is processed after IP conversion.</p> |

34.2.8.2 Destination Picture Rounding Control Register (FD1_WPF_RNDCTL)

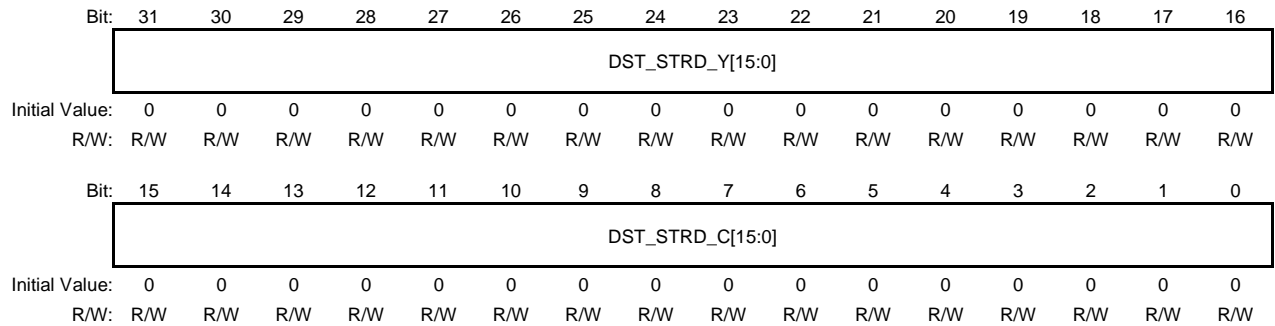
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|-----------|------|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | CBRM | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | CLMD[1:0] | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----------------------------|-----------|---------------|-----|---|
| 31 to 29, 27 to 14, 11 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | CBRM | B'0 | R/W | Bit Count Reduction Selection for Data Storage in Packed RGB This bit specifies the method for reducing the number of bits when data is stored in the bit fields indicated as R, G, and B in Table 34.7 and the target bit fields are not eight bits. 0: Bit count conversion: The lower-order bits are truncated 1: Bit count conversion: Rounding (rounding off) |
| 13 to 12 | CLMD[1:0] | B'00 | R/W | Color Data Clipping These bits specify the method for clipping the YCbCr color data output from the WPF. When RGB color data is output from the WPF, specify 0 in these bits. b'00: Output value is not clipped (0-255) b'01: Output value is clipped: YCbCr mode 1 (16-235 (Y), 16-240 (Cb/Cr)) b'10: Output value is clipped: YCbCr mode 2 (Y/Cb/Cr = 1-254) b'11: Setting prohibited |

34.2.8.3 Destination Picture Stride Register (FD1_WPF_PSTRIDE)

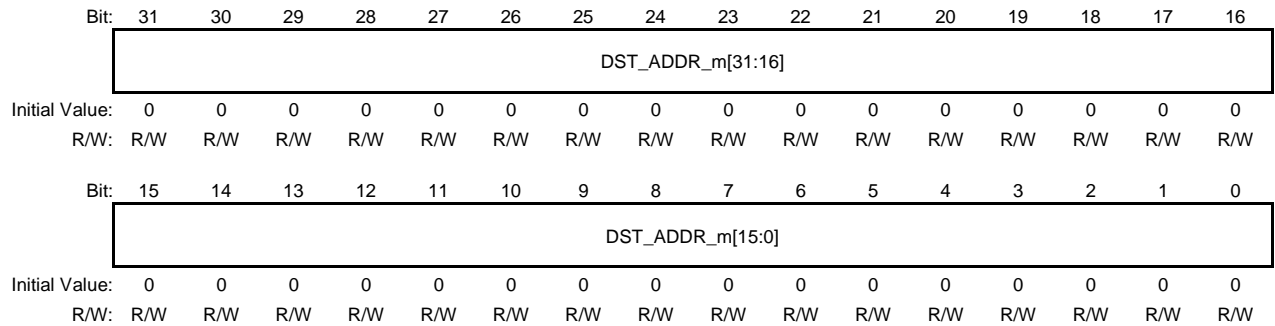
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------|---------------|-----|--|
| 31 to 16 | DST_STRD_Y [15:0] | All 0 | R/W | Memory Stride of Destination Picture Y/RGB Plane These bits specify in 1-byte units the memory stride of the destination picture in the external memory to be written to by the WPF. A value from H'0020 to H'1FFE can be specified. |
| 15 to 0 | DST_STRD_C [15:0] | All 0 | R/W | Memory Stride of Destination Picture C Plane These bits specify in 1-byte units the memory stride for the C plane of the destination picture in the external memory to be written to by the WPF. When the WPF outputs images in an RGB format, this setting is not used. When the WPF outputs images in YCbCr planar format, this setting is applied to both the Cb and Cr planes. A value from H'0020 to H'1FFE can be specified. |

34.2.8.4 Destination Component-m Address Register (FD1_WPF_ADDR_m: m = Y, C0, C1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------------|---------------|-----|---|
| 31 to 0 | DST_ADDR_m [31:0] | All 0 | R/W | Destination Address for Component-m (m = Y, C0, C1) These bits specify in 1-byte units the address for storing the destination component-m plane to be written to the external memory by the WPF. A value from H'0000_0000 to H'FFFF_FFFF can be specified. |

34.2.8.5 WPF Data Swap Register (FD1_WPF_SWAP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------------|-----|-----|------------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | SSWAP[3:0] | | | OSWAP[3:0] | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 4 | SSWAP [3:0] | All 0 | R/W | Set the same value with FD1_RPF_SWAP.ISWAP. |
| 3 to 0 | OSWAP [3:0] | H'0 | R/W | Output Swap Setting for Image Data This bit field specifies the data swapping function. Each bit corresponds to the following data swapping. Bit[3]: Long Long Word (64-bit) Swap Bit[2]: Long Word (32-bit) Swap Bit[1]: Word (16-bit) Swap Bit[0]: Byte (8-bit) Swap |

When data order in memory for each format is defined as same as Table 34.7 for RGB format and Figure 34.9 to Figure 34.11 for YCbCr format set 4'b1111 to OSWAP[3:0]. If data order in memory is different from definitions of these table and figures, it can be adjusted to definitions of these table and figures in 16 bytes unit as shown in Table 34.6.

Note: Follow the section “34.4 FCP for FDP (FCPF)” about setting this swap register.

34.2.9 IPC Control Registers

Table 34.8 shows registers categorized in function of IPC.

Table 34.8 IPC registers categorized in function

| Function | Registers |
|-------------------------------|--|
| IP Conversion Control | FD1_IPC_MODE |
| 3D de-interlace | FD1_IPC_COMB_DET |
| Adaptive 2D/3D de-interlacing | FD1_IPC_SMSK_THRESH FD1_IPC_MOTDEC |
| Diagonal line interpolation | FD1_IPC_DLI_BLEND FD1_IPC_DLI_HGAIN FD1_IPC_DLI_SPRS FD1_IPC_DLI_ANGLE FD1_IPC_DLI_ISOPIX0 FD1_IPC_DLI_ISOPIX1 |
| Film detection | FD1_SENSOR_m (m = 0, 1, ..., 17) FD1_SENSOR_CTL0 FD1_SENSOR_CTL1 FD1_SENSOR_CTL2 FD1_SENSOR_CTL3 FD1_IPC_SENSOR_TH0 FD1_IPC_SENSOR_TH1 |
| Line memory pixel number | FD1_IPC_LMEM |
| IP Internal data | FD1_IP_INTDATA |

For chroma component, IPC cannot process 3D de-interlace, adaptive 2D/3D de-interlacing, diagonal line interpolation and film detection but can process fixed 2D IP conversion.

34.2.9.1 IPC Mode Register (FD1_IPC_MODE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----|----|----|----|----|----|----------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | DLI | — | — | — | — | — | DIM[2:0] | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|----------|---------------|-----|--|
| 31 to 9, 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | DLI | B'0 | R/W | Set the value 1 to this bit. |
| 2 to 0 | DIM[2:0] | B'000 | R/W | De-Interlacing Mode These bits specify the de-interlacing mode. B'000: Adaptive 2D/3D de-interlacing B'001: Fixed 2D de-interlacing B'010: Fixed 3D de-interlacing B'011: Select previous field for interpolated lines B'100: Select next field for interpolated lines B'101, B'110, B'111: setting prohibited The de-interlacing process of chroma component is fixed 2D de-interlacing regardless of these bits. In the case of first, second and last frame, DIM [2:0] should take other value than the upper guide. Refer to Table 34.2 for setting this register and FD1_CTL_OPMODE, FD1_CTL_CHACT according to each use case. |

34.2.9.2 Still Mask Threshold Register (FD1_IPC_SMSK_THRESH)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|--------------|-----|-----|-----|-----|-----|-----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | FSM0 |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | SMSK_TH[7:0] | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------------------|------------------|---------------|-----|---|
| 31 to 17, 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | FSM0 | B'0 | R/W | Set the value 1. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 7 to 0 | SMSK_TH [7:0] | All 0 | R/W | Set the value 2 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |

34.2.9.3 Comb Detection Parameter Register (FD1_IPC_COMB_DET)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------------|-----|-----|-----|-----|-----|-----|---------------|---------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | CMB_OFST[7:0] | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CMB_MAX[7:0] | | | | | | | CMB_GRAD[7:0] | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | CMB_OFST [7:0] | All 0 | R/W | Set the value H'20 to this bit as default. |
| 15 to 8 | CMB_MAX [7:0] | All 0 | R/W | Set the value 0 to this bit as default. |
| 7 to 0 | CMB_GRAD [7:0] | All 0 | R/W | Set the value H'40 to this bit as default. |

When combing noise appears obviously it is recommended to set value H'20, H'ff, H'ff to register CMB_OFST, CMB_MAX, CMB_GRAD respectively.

34.2.9.4 Motion Decision Parameter Register (FD1_IPC_MOTDEC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------------|-----|-----|-----|-----|-----|-----|-----|---------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MOV_COEF[7:0] | | | | | | | | STL_COEF[7:0] | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 8 | MOV_COEF [7:0] | All 0 | R/W | Set the value H'80 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 7 to 0 | STL_COEF [7:0] | All 0 | R/W | Set the value H'20 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |

34.2.9.5 DLI Blend Parameter Register (FD1_IPC_DLI_BLEND)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------------|-----|-----|-----|-----|-----|-----|---------------|---------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | BLD_GRAD[7:0] | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | BLD_MAX[7:0] | | | | | | | BLD_OFST[7:0] | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | BLD_GRAD [7:0] | All 0 | R/W | Set the value H'80 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 15 to 8 | BLD_MAX [7:0] | All 0 | R/W | Set the value H'ff to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 7 to 0 | BLD_OFST [7:0] | All 0 | R/W | Set the value H'02 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |

34.2.9.6 DLI Horizontal Frequency Gain Register (FD1_IPC_DLI_HGAIN)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------------|-----|-----|-----|-----|-----|-----|-------------|--------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | HG_GRAD[7:0] | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | HG_OFST[7:0] | | | | | | | HG_MAX[7:0] | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | HG_GRAD [7:0] | All 0 | R/W | Set the value H'10 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 15 to 8 | HG_OFST [7:0] | All 0 | R/W | Set the value H'00 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 7 to 0 | HG_MAX [7:0] | All 0 | R/W | Set the value H'ff to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |

34.2.9.7 DLI Suppression Parameter register (FD1_IPC_DLI_SPRS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----------------|-----|-----|-----|-----|-----|-----|-----|----------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | SPRS_GRAD[7:0] | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SPRS_OFST[7:0] | | | | | | | | SPRS_MAX[7:0] | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | SPRS_GRAD [7:0] | All 0 | R/W | Set the value H'90 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 15 to 8 | SPRS_OFST [7:0] | All 0 | R/W | Set the value H'04 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 7 to 0 | SPRS_MAX [7:0] | All 0 | R/W | Set the value H'ff to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |

34.2.9.8 DLI Angle Parameter Register (FD1_IPC_DLI_ANGLE)

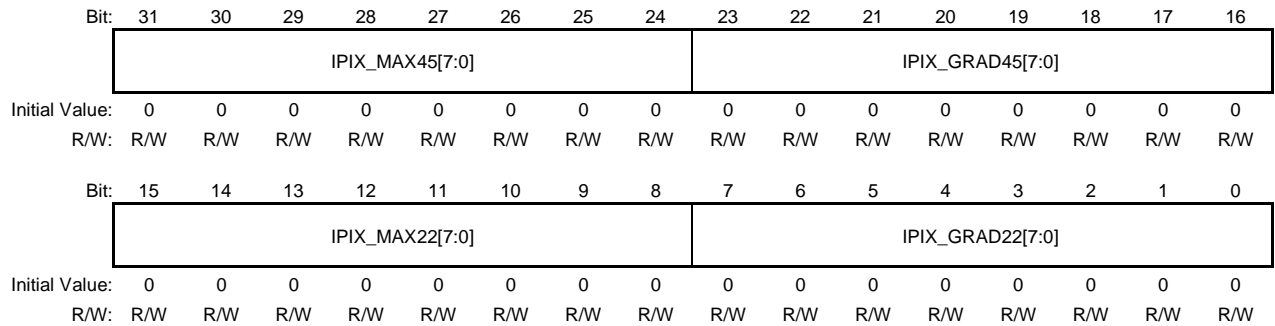
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------|-----|-----|-----|-----|-----|-----|-----|-------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | ASEL45[7:0] | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | ASEL22[7:0] | | | | | | | | ASEL15[7:0] | | | | | | | |
| Initial Value: | 0 | | | | | | | | 0 | | | | | | | |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | ASEL45 | All 0 | R/W | Set the value H'04 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 15 to 8 | ASEL22 | All 0 | R/W | Set the value H'08 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 7 to 0 | ASEL15 | All 0 | R/W | Set the value H'0c to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |

34.2.9.9 DLI Isolated Pixel Parameter Register0 (FD1_IPC_DLI_ISOPIX0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | IPIX_MAX45 | All 0 | R/W | Set the value H'ff to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 23 to 16 | IPIX_GRAD45 | All 0 | R/W | Set the value H'10 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 15 to 8 | IPIX_MAX22 | All 0 | R/W | Set the value H'ff to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 7 to 0 | IPIX_GRAD22 | All 0 | R/W | Set the value H'10 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |

34.2.9.10 DLI Isolated Pixel Parameter Register1 (FD1_IPC_DLI_ISOPIX1)

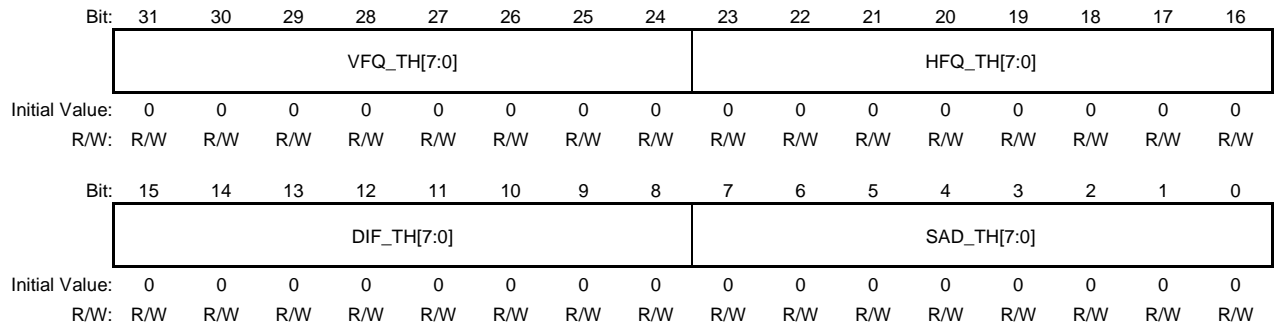
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------------|-----|-----|-----|-----|-----|-----|-----|------------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | IPIX_MAX15[7:0] | | | | | | | | IPIX_GRAD15[7:0] | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 8 | IPIX_MAX15 | All 0 | R/W | Set the value H'ff to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 7 to 0 | IPIX_GRAD15 | All 0 | R/W | Set the value H'10 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |

34.2.9.11 IPC Sensor Threshold Register0 (FD1_IPC_SENSOR_TH0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 24 | VFQ_TH [7:0] | All 0 | R/W | Set the value H'20 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 23 to 16 | HFQ_TH [7:0] | All 0 | R/W | Set the value H'20 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 15 to 8 | DIF_TH [7:0] | All 0 | R/W | Set the value H'80 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 7 to 0 | SAD_TH [7:0] | All 0 | R/W | Set the value H'80 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |

34.2.9.12 IPC Sensor Threshold Register1 (FD1_IPC_SENSOR_TH1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------------|-----|-----|-----|-----|-----|-----|--------------|-----|-----|-----|-------------------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | DETECTOR_SEL[4:0] | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | COMB_TH[7:0] | | | | | | | FREQ_TH[7:0] | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------|---------------|-----|--|
| 31 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 to 16 | DETECTOR_SEL [4:0] | All 0 | R/W | Set the value 0 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 15 to 8 | COMB_TH [7:0] | All 0 | R/W | Set the value 0 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 7 to 0 | FREQ_TH [7:0] | All 0 | R/W | Set the value 0 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |

34.2.9.13 Sensor Control Register0 (FD1_SENSOR_CTL0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------------|-----|-----|-----|---------------|-----|-----|-----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | FRM_LVTH[3:0] | | | | FLD_LVTH[3:0] | | | | — | — | — | — | — | — | — | FD_EN |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|------------------|----------------|---------------|-----|---|
| 31 to 16, 7 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 12 | FRM_LVTH [3:0] | H'0 | R/W | Set the value 2 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 11 to 8 | FLD_LVTH [3:0] | All 0 | R/W | Set the value 2 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 0 | FD_EN | B'0 | R/W | Set the value 1 to this bit. |

34.2.9.14 Sensor Control Register1 (FD1_SENSOR_CTL1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | XS[12:0] | | | | | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | YS[12:0] | | | | | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|----------|---------------|-----|---|
| 31 to 29, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | XS[12:0] | All 0 | R/W | Set the value 0 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |
| 12 to 0 | YS[12:0] | All 0 | R/W | Set the value 0 to this bit. Other setting value is used for purpose of h/w debugging. Do not set other value. |

34.2.9.15 Sensor Control Register2 (FD1_SENSOR_CTL2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | XE[12:0] | | | | | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | YE[12:0] | | | | | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|----------|---------------|-----|--|
| 31 to 29, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | XE[12:0] | All 0 | R/W | Set the value (FD1_RPF_SIZE.HSIZE - 1) to this bit. |
| 12 to 0 | YE[12:0] | All 0 | R/W | Set the value (FD1_RPF_SIZE.VSIZE - 1) to this bit if input picture is progressive. Set the value (FD1_RPF_SIZE.VSIZE x 2 - 1) to this bit if input picture is interlace. |

34.2.9.16 Sensor Control Register3 (FD1_SENSOR_CTL3)

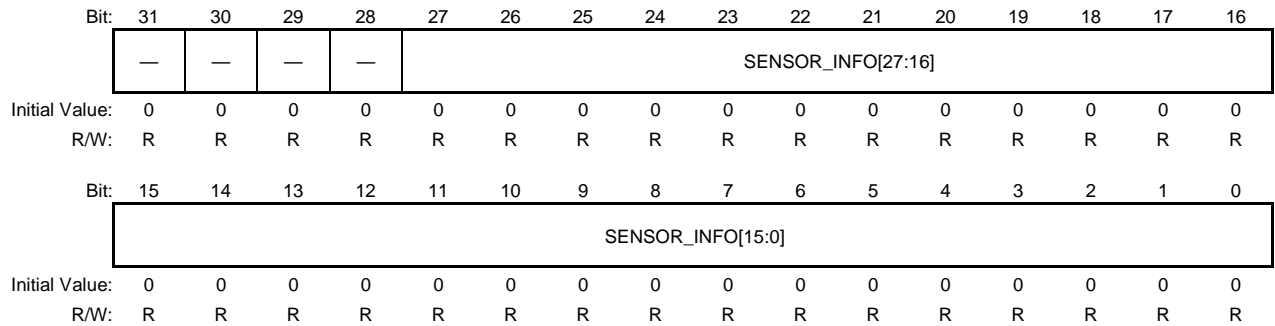
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|--------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | PO SX0[12:0] | | | | | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | PO SX1[12:0] | | | | | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------------------|---------------|---------------|-----|--|
| 31 to 29, 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | PO SX0 [12:0] | All 0 | R/W | Set the value INT (FD1_RPF_SIZE.HSIZE/3) to this bit. |
| 12 to 0 | PO SX1 [12:0] | All 0 | R/W | Set the value INT (2 * FD1_RPF_SIZE.HSIZE/3) to this bit. |

34.2.9.17 Sensor Register (FD1_SENSOR_m: m = 0, 1...17)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------|---------------|-----|--|
| 31 to 28 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 to 0 | SENSOR_INFO [27:0] | All 0 | R | Sensor information of FDP1. |

34.2.9.18 Line Memory Pixel Number Register (FD1_IPC_LMEM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | PNUM[11:0] | | | | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 0 | PNUM[11:0] | All 0 | R/W | Specify 1024 in case of linear addressing access. In case of tile addressing access, set 960 to this bit. Refer to section "Operation", section "FCP for FDP", FCP HWM. |

34.2.9.19 IP Internal Data (FD1_IP_INTDATA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------------------|----|----|----|----|----|----|----|----|----|----|----|----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | IP_INTERNAL_DATA[31:16] | | | | | | | | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | IP_INTERNAL_DATA[15:0] | | | | | | | | | | | | | | | |
| Initial Value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0/1 | 0/1 | 0/1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------------|--|-----|--|
| 31 to 0 | IP_INTERNAL_DATA[31:0] | H'0201_0203 or H'0201_0202 or H'0201_0204 or H'0201_0205 | R | Internal data specifies version of FDP H'0201_0203: FDP1 version is RZ/G2H H'0201_0202: FDP1 version is RZ/G2M V1.3, RZ/G2M V3.0 H'0201_0204: FDP1 version is RZ/G2N H'0201_0205: FDP1 version is RZ/G2E |

34.2.10 LUT**34.2.10.1 LUT DIF_ADJ**

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Refer to section 34.3.2.

34.2.10.2 LUT SAD_ADJ

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Refer to section 34.3.2.

34.2.10.3 LUT BLD_GAIN

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Refer to section 34.3.2.

34.2.10.4 LUT DIF_GAIN

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Refer to section 34.3.2.

34.2.10.5 LUT MDET

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Refer to section 34.3.2.

34.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

34.3.1 Procedure of FDP1 Start and Stop

34.3.1.1 Register usages for FDP1 Start and Stop

Registers FD1_CTL_CMD, FD1_CTL_SGCMD, and FD1_CTL_REGEND must be used correctly in the procedures of FDP1 start and stop as specified in Table 34.9 and Table 34.10.

Table 34.9 Register usages for starting FDP1

| Register | Start the first frame (FR[0]) | Start the second or later frame (FR[f], f = 1, 2, 3...) |
|-----------------------|--|---|
| FD1_CTL_SGCMD.SGEN | After all registers have been set completely, it must be set as 1 to generate V-Interruption. | It must be kept as 1 from the previous frame FR [f-1]. |
| FD1_CTL_REGEND.REGEND | After all registers except FD1_CTL_SGCMD.SGEN have been set completely, it must be set 1. As the results, at the first V-interruption, all V-update registers are reflected for hardware reference, FD1_CTL_CMD.STRCMD is referred to start the first frame. | After V-interruption of starting the previous frame FR [f-1], it must be set 1 after all registers about FR[f] have been set completely. As the results, at the V-interruption of the end of frame FR [f-1], the V-update registers will be reflected to hardware reference, FD1_CTL_CMD.STRCMD is referred to start the frame FR[f]. |
| FD1_CTL_CMD.STRCMD | It must be set 1 when all registers except FD1_CTL_SGCMD.SGEN, FD1_CTL_REGEND.REGEND are set. At the first V-Interruption it is referred by hardware to start the frame. | It must be set 1 when all registers except FD1_CTL_REGEND.REGEND are set at the next V-Interruption (boundary between FR [t-1] and FR[t]) it is referred by hardware to start the frame. |

Table 34.10 Register usages for stopping FDP1

| Register | Stopping immediately by Software reset | Stopping after current process finished |
|-----------------------|--|---|
| FD1_CTL_SGCMD.SGEN | It must be cleared before setting the value of FD1_CTL_SRESET.SRST as 1 so that no more V-Interruption after FDP1 stopped. | After V-interruption at the final frame end (FD1_CTL_IRQSTA.FRE is set as 1) occurred, it must be cleared so that no more V-Interruption is generated after FDP1 stopped |
| FD1_CTL_REGEND.REGEND | — | <p>1) No Interrupt Mode</p> <p>After V-interruption of starting the final frame, it must be set 1 right after clearing FD1_CTL_CMD.STRCMD so that the final V-interruption occurs and operation status of the final frame is updated to V-update status registers for Software reference.</p> <p>2) Other modes</p> <p>Right after V-interruption at starting the final frame, it must be set 1 right after clearing FD1_CTL_CMD.STRCMD so that operation status of the final frame is updated to V-update status registers for Software reference.</p> |
| FD1_CTL_CMD.STRCMD | After FDP1 stopped successfully (FD1_CTL_IRQSTA.FRE is set as 1), it must be cleared to prevent wrong operation when restart FDP1. | After V-interruption of starting the final frame, it must be cleared to prevent wrong operation when restart FDP1. |

34.3.1.2 Register setting order for FDP1 Start and Stop

(1) Interrupt Mode or Best Effort Mode

(1a) To start the first frame, set registers in following order.

1. Set 1 to FD1_CTL_CLKCTRL.CSTP_N
2. Set registers other than FD1_CTL_CMD, FD1_CTL_SGCMD, and FD1_CTL_REGEND.
3. Set FD1_CTL_CMD.STRCMD to the value 1. Set FD1_CTL_REGEND.REGEND to the value 1.
4. Set FD1_CTL_SGCMD.SGEN to the value 1, then FDP1 starts its process. FD1_CTL_SGCMD.SGEN must be not cleared if continue to the second or later frames.

(1b) To start the second or later frame, set registers in following order.

1. After V-interruption of the previous frame start timing occurred, set registers other than FD1_CTL_CMD and FD1_CTL_REGEND.
2. Set FD1_CTL_CMD.STRCMD to the value 1.
3. Set FD1_CTL_REGEND.REGEND to the value 1.
4. At the V-interruption of the previous frame end timing, FDP1 starts the target frame. The V-interruption of the previous frame end timing and the one of the target frame start timing are the same.

(1c) To stop FDP1 operation after current frame finished, set registers as the following.

1. After V-interruption of the current frame start timing occurred, set FD1_CTL_CMD.STRCMD to the value 0.
2. Set FD1_CTL_REGEND.REGEND to the value 1. If this register was completed before the V-Interruption of frame end timing, FDP1 stops its operation and current frame becomes the last frame of the image sequence. If these register setting could not be completed before the V-Interruption of frame end timing, FDP1 stops its operation in the next V-interruption. In the later case, V-update status registers are not updated at V-interruption of frame end but at the next V-interruption at which FD1_CTL_REGEND.REGEND is set as 1.
3. Set FD1_CTL_SGCMD.SGEN to the value 0, then FDP1 will stop generating V-interruption.

(1d) To stop FDP1 operation immediately, set registers as the following.

1. Set FD1_CTL_SGCMD.SGEN to the value 0, then FDP1 will stop generating V-interruption.
2. Set FD1_CTL_SRESET.SRST to the value 1. FDP1 will invoke termination process immediately.
3. Wait frame end interrupt from FDP1 or until the register bit FD1_CTL_IRQSTA.FRE is set to 1. After it occurs, FDP1 have finished its processing. With this operation, FDP1 can stop its process quickly, but the output frame of the last frame is corrupted.

(2) No interrupt mode

(2a) To start the first frame, set registers in following order.

Same as (1a).

(2b) To start the second or later frame, set registers in following order.

Same as (1b).

(2c) To stop FDP1 operation after current frame finished, set registers as the following.

1. After V-interruption of the current frame start timing occurred, set FD1_CTL_CMD.STRCMD to the value 0.
2. Set FD1_CTL_REGEND.REGEND to the value 1. At the V-Interruption of frame end timing, FDP1 stops its operation and current frame becomes the last frame of the image sequence.
3. Set FD1_CTL_SGCMD.SGEN to the value 0, then FDP1 will stop generating V-interruption.

(2d) To stop FDP1 operation immediately, set registers as the following.

Same as (1d).

34.3.2 Look Up Table (LUT) Setting

FDP1 has the Look-Up-Tables (LUT) shown in Table 34.11 to change the characteristic of the de-interlacing function. The LUT is an 8-bit entry and 8-bit output table. Note that users can access to LUT only by 32 bit access. Do not access other bit length such as 8bit access or 16bit access. LUT behaves like immediate registers described in section 34.2.2, so do not change the value in LUT while FDP1 is under processing.

Table 34.11 List of LUT

| LUT Name |
|----------|
| DIF_ADJ |
| SAD_ADJ |
| BLD_GAIN |
| DIF_GAIN |
| MDET |

The LUT is used for making the non-linear characteristic shown in Figure 34.14 where X and Y are input and output of LUT respectively.

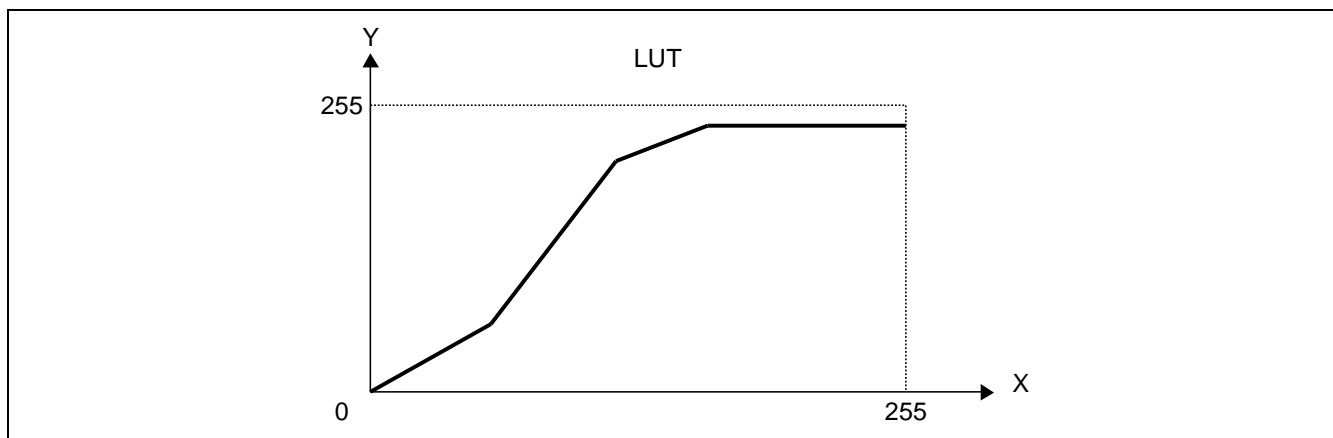


Figure 34.14 LUT for changing the de-interlacing characteristic

There are 3 types of setting values for those LUTs, “The setting 1 of each LUT”, “The setting 2 of each LUT” and “The setting 3 of each LUT”, respectively specified in Table 34.12, Table 34.13 and Table 34.14. “The setting 1 of each LUT” which keeps the balance ratio of 2D and 3D interpolated images on the output image, is recommended as default setting. When it is needed to strengthen the 3D interpolated images even if there is motion in the input image sequence, use “The setting 2 of each LUT” or “The setting 3 of each LUT” with which the ratio of 3D interpolated image increases as shown in the Figure 34.15.

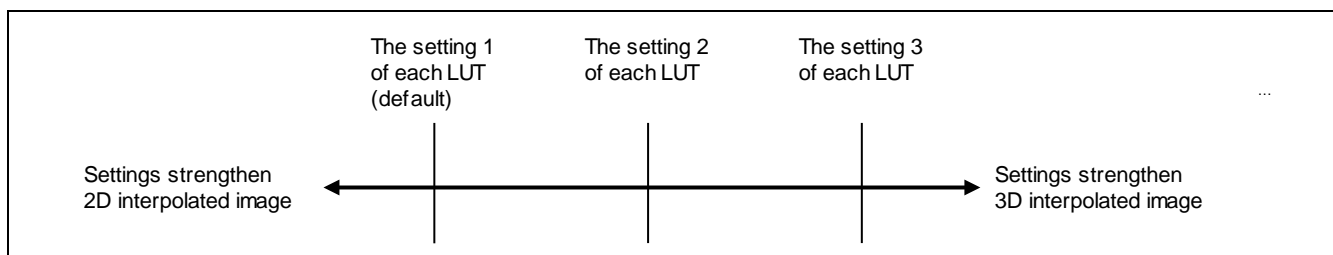


Figure 34.15 The target of each type of LUT settings

Table 34.12 The setting 1 of each LUT

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 0 | H'00 | H'00 | H'80 | H'80 | H'00 |
| 1 | H'24 | H'24 | H'80 | H'80 | H'01 |
| 2 | H'43 | H'43 | H'80 | H'80 | H'02 |
| 3 | H'5E | H'5E | H'80 | H'80 | H'03 |
| 4 | H'76 | H'76 | H'80 | H'80 | H'04 |
| 5 | H'8C | H'8C | H'80 | H'80 | H'05 |
| 6 | H'9E | H'9E | H'80 | H'80 | H'06 |
| 7 | H'AF | H'AF | H'80 | H'80 | H'07 |
| 8 | H'BD | H'BD | H'80 | H'80 | H'08 |
| 9 | H'C9 | H'C9 | H'80 | H'80 | H'09 |
| 10 | H'D4 | H'D4 | H'80 | H'80 | H'0A |
| 11 | H'DD | H'DD | H'80 | H'80 | H'0B |
| 12 | H'E4 | H'E4 | H'80 | H'80 | H'0C |
| 13 | H'EA | H'EA | H'80 | H'80 | H'0D |
| 14 | H'EF | H'EF | H'80 | H'80 | H'0E |
| 15 | H'F3 | H'F3 | H'80 | H'80 | H'0F |
| 16 | H'F6 | H'F6 | H'80 | H'80 | H'10 |
| 17 | H'F9 | H'F9 | H'80 | H'80 | H'11 |
| 18 | H'FB | H'FB | H'80 | H'80 | H'12 |
| 19 | H'FC | H'FC | H'80 | H'80 | H'13 |
| 20 | H'FD | H'FD | H'80 | H'80 | H'14 |
| 21 | H'FE | H'FE | H'80 | H'80 | H'15 |
| 22 | H'FE | H'FE | H'80 | H'80 | H'16 |
| 23 | H'FF | H'FF | H'80 | H'80 | H'17 |
| 24 | H'FF | H'FF | H'80 | H'80 | H'18 |
| 25 | H'FF | H'FF | H'80 | H'80 | H'19 |
| 26 | H'FF | H'FF | H'80 | H'80 | H'1A |
| 27 | H'FF | H'FF | H'80 | H'80 | H'1B |
| 28 | H'FF | H'FF | H'80 | H'80 | H'1C |
| 29 | H'FF | H'FF | H'80 | H'80 | H'1D |
| 30 | H'FF | H'FF | H'80 | H'80 | H'1E |
| 31 | H'FF | H'FF | H'80 | H'80 | H'1F |
| 32 | H'FF | H'FF | H'80 | H'80 | H'20 |
| 33 | H'FF | H'FF | H'80 | H'80 | H'21 |
| 34 | H'FF | H'FF | H'80 | H'80 | H'22 |
| 35 | H'FF | H'FF | H'80 | H'80 | H'23 |
| 36 | H'FF | H'FF | H'80 | H'80 | H'24 |
| 37 | H'FF | H'FF | H'80 | H'80 | H'25 |
| 38 | H'FF | H'FF | H'80 | H'80 | H'26 |
| 39 | H'FF | H'FF | H'80 | H'80 | H'27 |
| 40 | H'FF | H'FF | H'80 | H'80 | H'28 |
| 41 | H'FF | H'FF | H'80 | H'80 | H'29 |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 42 | H'FF | H'FF | H'80 | H'80 | H'2A |
| 43 | H'FF | H'FF | H'80 | H'80 | H'2B |
| 44 | H'FF | H'FF | H'80 | H'80 | H'2C |
| 45 | H'FF | H'FF | H'80 | H'80 | H'2D |
| 46 | H'FF | H'FF | H'80 | H'80 | H'2E |
| 47 | H'FF | H'FF | H'80 | H'80 | H'2F |
| 48 | H'FF | H'FF | H'80 | H'80 | H'30 |
| 49 | H'FF | H'FF | H'80 | H'80 | H'31 |
| 50 | H'FF | H'FF | H'80 | H'80 | H'32 |
| 51 | H'FF | H'FF | H'80 | H'80 | H'33 |
| 52 | H'FF | H'FF | H'80 | H'80 | H'34 |
| 53 | H'FF | H'FF | H'80 | H'80 | H'35 |
| 54 | H'FF | H'FF | H'80 | H'80 | H'36 |
| 55 | H'FF | H'FF | H'80 | H'80 | H'37 |
| 56 | H'FF | H'FF | H'80 | H'80 | H'38 |
| 57 | H'FF | H'FF | H'80 | H'80 | H'39 |
| 58 | H'FF | H'FF | H'80 | H'80 | H'3A |
| 59 | H'FF | H'FF | H'80 | H'80 | H'3B |
| 60 | H'FF | H'FF | H'80 | H'80 | H'3C |
| 61 | H'FF | H'FF | H'80 | H'80 | H'3D |
| 62 | H'FF | H'FF | H'80 | H'80 | H'3E |
| 63 | H'FF | H'FF | H'80 | H'80 | H'3F |
| 64 | H'FF | H'FF | H'80 | H'80 | H'40 |
| 65 | H'FF | H'FF | H'80 | H'80 | H'41 |
| 66 | H'FF | H'FF | H'80 | H'80 | H'42 |
| 67 | H'FF | H'FF | H'80 | H'80 | H'43 |
| 68 | H'FF | H'FF | H'80 | H'80 | H'44 |
| 69 | H'FF | H'FF | H'80 | H'80 | H'45 |
| 70 | H'FF | H'FF | H'80 | H'80 | H'46 |
| 71 | H'FF | H'FF | H'80 | H'80 | H'47 |
| 72 | H'FF | H'FF | H'80 | H'80 | H'48 |
| 73 | H'FF | H'FF | H'80 | H'80 | H'49 |
| 74 | H'FF | H'FF | H'80 | H'80 | H'4A |
| 75 | H'FF | H'FF | H'80 | H'80 | H'4B |
| 76 | H'FF | H'FF | H'80 | H'80 | H'4C |
| 77 | H'FF | H'FF | H'80 | H'80 | H'4D |
| 78 | H'FF | H'FF | H'80 | H'80 | H'4E |
| 79 | H'FF | H'FF | H'80 | H'80 | H'4F |
| 80 | H'FF | H'FF | H'80 | H'80 | H'50 |
| 81 | H'FF | H'FF | H'80 | H'80 | H'51 |
| 82 | H'FF | H'FF | H'80 | H'80 | H'52 |
| 83 | H'FF | H'FF | H'80 | H'80 | H'53 |
| 84 | H'FF | H'FF | H'80 | H'80 | H'54 |
| 85 | H'FF | H'FF | H'80 | H'80 | H'55 |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 86 | H'FF | H'FF | H'80 | H'80 | H'56 |
| 87 | H'FF | H'FF | H'80 | H'80 | H'57 |
| 88 | H'FF | H'FF | H'80 | H'80 | H'58 |
| 89 | H'FF | H'FF | H'80 | H'80 | H'59 |
| 90 | H'FF | H'FF | H'80 | H'80 | H'5A |
| 91 | H'FF | H'FF | H'80 | H'80 | H'5B |
| 92 | H'FF | H'FF | H'80 | H'80 | H'5C |
| 93 | H'FF | H'FF | H'80 | H'80 | H'5D |
| 94 | H'FF | H'FF | H'80 | H'80 | H'5E |
| 95 | H'FF | H'FF | H'80 | H'80 | H'5F |
| 96 | H'FF | H'FF | H'80 | H'80 | H'60 |
| 97 | H'FF | H'FF | H'80 | H'80 | H'61 |
| 98 | H'FF | H'FF | H'80 | H'80 | H'62 |
| 99 | H'FF | H'FF | H'80 | H'80 | H'63 |
| 100 | H'FF | H'FF | H'80 | H'80 | H'64 |
| 101 | H'FF | H'FF | H'80 | H'80 | H'65 |
| 102 | H'FF | H'FF | H'80 | H'80 | H'66 |
| 103 | H'FF | H'FF | H'80 | H'80 | H'67 |
| 104 | H'FF | H'FF | H'80 | H'80 | H'68 |
| 105 | H'FF | H'FF | H'80 | H'80 | H'69 |
| 106 | H'FF | H'FF | H'80 | H'80 | H'6A |
| 107 | H'FF | H'FF | H'80 | H'80 | H'6B |
| 108 | H'FF | H'FF | H'80 | H'80 | H'6C |
| 109 | H'FF | H'FF | H'80 | H'80 | H'6D |
| 110 | H'FF | H'FF | H'80 | H'80 | H'6E |
| 111 | H'FF | H'FF | H'80 | H'80 | H'6F |
| 112 | H'FF | H'FF | H'80 | H'80 | H'70 |
| 113 | H'FF | H'FF | H'80 | H'80 | H'71 |
| 114 | H'FF | H'FF | H'80 | H'80 | H'72 |
| 115 | H'FF | H'FF | H'80 | H'80 | H'73 |
| 116 | H'FF | H'FF | H'80 | H'80 | H'74 |
| 117 | H'FF | H'FF | H'80 | H'80 | H'75 |
| 118 | H'FF | H'FF | H'80 | H'80 | H'76 |
| 119 | H'FF | H'FF | H'80 | H'80 | H'77 |
| 120 | H'FF | H'FF | H'80 | H'80 | H'78 |
| 121 | H'FF | H'FF | H'80 | H'80 | H'79 |
| 122 | H'FF | H'FF | H'80 | H'80 | H'7A |
| 123 | H'FF | H'FF | H'80 | H'80 | H'7B |
| 124 | H'FF | H'FF | H'80 | H'80 | H'7C |
| 125 | H'FF | H'FF | H'80 | H'80 | H'7D |
| 126 | H'FF | H'FF | H'80 | H'80 | H'7E |
| 127 | H'FF | H'FF | H'80 | H'80 | H'7F |
| 128 | H'FF | H'FF | H'80 | H'80 | H'80 |
| 129 | H'FF | H'FF | H'80 | H'80 | H'81 |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 130 | H'FF | H'FF | H'80 | H'80 | H'82 |
| 131 | H'FF | H'FF | H'80 | H'80 | H'83 |
| 132 | H'FF | H'FF | H'80 | H'80 | H'84 |
| 133 | H'FF | H'FF | H'80 | H'80 | H'85 |
| 134 | H'FF | H'FF | H'80 | H'80 | H'86 |
| 135 | H'FF | H'FF | H'80 | H'80 | H'87 |
| 136 | H'FF | H'FF | H'80 | H'80 | H'88 |
| 137 | H'FF | H'FF | H'80 | H'80 | H'89 |
| 138 | H'FF | H'FF | H'80 | H'80 | H'8A |
| 139 | H'FF | H'FF | H'80 | H'80 | H'8B |
| 140 | H'FF | H'FF | H'80 | H'80 | H'8C |
| 141 | H'FF | H'FF | H'80 | H'80 | H'8D |
| 142 | H'FF | H'FF | H'80 | H'80 | H'8E |
| 143 | H'FF | H'FF | H'80 | H'80 | H'8F |
| 144 | H'FF | H'FF | H'80 | H'80 | H'90 |
| 145 | H'FF | H'FF | H'80 | H'80 | H'91 |
| 146 | H'FF | H'FF | H'80 | H'80 | H'92 |
| 147 | H'FF | H'FF | H'80 | H'80 | H'93 |
| 148 | H'FF | H'FF | H'80 | H'80 | H'94 |
| 149 | H'FF | H'FF | H'80 | H'80 | H'95 |
| 150 | H'FF | H'FF | H'80 | H'80 | H'96 |
| 151 | H'FF | H'FF | H'80 | H'80 | H'97 |
| 152 | H'FF | H'FF | H'80 | H'80 | H'98 |
| 153 | H'FF | H'FF | H'80 | H'80 | H'99 |
| 154 | H'FF | H'FF | H'80 | H'80 | H'9A |
| 155 | H'FF | H'FF | H'80 | H'80 | H'9B |
| 156 | H'FF | H'FF | H'80 | H'80 | H'9C |
| 157 | H'FF | H'FF | H'80 | H'80 | H'9D |
| 158 | H'FF | H'FF | H'80 | H'80 | H'9E |
| 159 | H'FF | H'FF | H'80 | H'80 | H'9F |
| 160 | H'FF | H'FF | H'80 | H'80 | H'A0 |
| 161 | H'FF | H'FF | H'80 | H'80 | H'A1 |
| 162 | H'FF | H'FF | H'80 | H'80 | H'A2 |
| 163 | H'FF | H'FF | H'80 | H'80 | H'A3 |
| 164 | H'FF | H'FF | H'80 | H'80 | H'A4 |
| 165 | H'FF | H'FF | H'80 | H'80 | H'A5 |
| 166 | H'FF | H'FF | H'80 | H'80 | H'A6 |
| 167 | H'FF | H'FF | H'80 | H'80 | H'A7 |
| 168 | H'FF | H'FF | H'80 | H'80 | H'A8 |
| 169 | H'FF | H'FF | H'80 | H'80 | H'A9 |
| 170 | H'FF | H'FF | H'80 | H'80 | H'AA |
| 171 | H'FF | H'FF | H'80 | H'80 | H'AB |
| 172 | H'FF | H'FF | H'80 | H'80 | H'AC |
| 173 | H'FF | H'FF | H'80 | H'80 | H'AD |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 174 | H'FF | H'FF | H'80 | H'80 | H'AE |
| 175 | H'FF | H'FF | H'80 | H'80 | H'AF |
| 176 | H'FF | H'FF | H'80 | H'80 | H'B0 |
| 177 | H'FF | H'FF | H'80 | H'80 | H'B1 |
| 178 | H'FF | H'FF | H'80 | H'80 | H'B2 |
| 179 | H'FF | H'FF | H'80 | H'80 | H'B3 |
| 180 | H'FF | H'FF | H'80 | H'80 | H'B4 |
| 181 | H'FF | H'FF | H'80 | H'80 | H'B5 |
| 182 | H'FF | H'FF | H'80 | H'80 | H'B6 |
| 183 | H'FF | H'FF | H'80 | H'80 | H'B7 |
| 184 | H'FF | H'FF | H'80 | H'80 | H'B8 |
| 185 | H'FF | H'FF | H'80 | H'80 | H'B9 |
| 186 | H'FF | H'FF | H'80 | H'80 | H'BA |
| 187 | H'FF | H'FF | H'80 | H'80 | H'BB |
| 188 | H'FF | H'FF | H'80 | H'80 | H'BC |
| 189 | H'FF | H'FF | H'80 | H'80 | H'BD |
| 190 | H'FF | H'FF | H'80 | H'80 | H'BE |
| 191 | H'FF | H'FF | H'80 | H'80 | H'BF |
| 192 | H'FF | H'FF | H'80 | H'80 | H'C0 |
| 193 | H'FF | H'FF | H'80 | H'80 | H'C1 |
| 194 | H'FF | H'FF | H'80 | H'80 | H'C2 |
| 195 | H'FF | H'FF | H'80 | H'80 | H'C3 |
| 196 | H'FF | H'FF | H'80 | H'80 | H'C4 |
| 197 | H'FF | H'FF | H'80 | H'80 | H'C5 |
| 198 | H'FF | H'FF | H'80 | H'80 | H'C6 |
| 199 | H'FF | H'FF | H'80 | H'80 | H'C7 |
| 200 | H'FF | H'FF | H'80 | H'80 | H'C8 |
| 201 | H'FF | H'FF | H'80 | H'80 | H'C9 |
| 202 | H'FF | H'FF | H'80 | H'80 | H'CA |
| 203 | H'FF | H'FF | H'80 | H'80 | H'CB |
| 204 | H'FF | H'FF | H'80 | H'80 | H'CC |
| 205 | H'FF | H'FF | H'80 | H'80 | H'CD |
| 206 | H'FF | H'FF | H'80 | H'80 | H'CE |
| 207 | H'FF | H'FF | H'80 | H'80 | H'CF |
| 208 | H'FF | H'FF | H'80 | H'80 | H'D0 |
| 209 | H'FF | H'FF | H'80 | H'80 | H'D1 |
| 210 | H'FF | H'FF | H'80 | H'80 | H'D2 |
| 211 | H'FF | H'FF | H'80 | H'80 | H'D3 |
| 212 | H'FF | H'FF | H'80 | H'80 | H'D4 |
| 213 | H'FF | H'FF | H'80 | H'80 | H'D5 |
| 214 | H'FF | H'FF | H'80 | H'80 | H'D6 |
| 215 | H'FF | H'FF | H'80 | H'80 | H'D7 |
| 216 | H'FF | H'FF | H'80 | H'80 | H'D8 |
| 217 | H'FF | H'FF | H'80 | H'80 | H'D9 |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 218 | H'FF | H'FF | H'80 | H'80 | H'DA |
| 219 | H'FF | H'FF | H'80 | H'80 | H'DB |
| 220 | H'FF | H'FF | H'80 | H'80 | H'DC |
| 221 | H'FF | H'FF | H'80 | H'80 | H'DD |
| 222 | H'FF | H'FF | H'80 | H'80 | H'DE |
| 223 | H'FF | H'FF | H'80 | H'80 | H'DF |
| 224 | H'FF | H'FF | H'80 | H'80 | H'E0 |
| 225 | H'FF | H'FF | H'80 | H'80 | H'E1 |
| 226 | H'FF | H'FF | H'80 | H'80 | H'E2 |
| 227 | H'FF | H'FF | H'80 | H'80 | H'E3 |
| 228 | H'FF | H'FF | H'80 | H'80 | H'E4 |
| 229 | H'FF | H'FF | H'80 | H'80 | H'E5 |
| 230 | H'FF | H'FF | H'80 | H'80 | H'E6 |
| 231 | H'FF | H'FF | H'80 | H'80 | H'E7 |
| 232 | H'FF | H'FF | H'80 | H'80 | H'E8 |
| 233 | H'FF | H'FF | H'80 | H'80 | H'E9 |
| 234 | H'FF | H'FF | H'80 | H'80 | H'EA |
| 235 | H'FF | H'FF | H'80 | H'80 | H'EB |
| 236 | H'FF | H'FF | H'80 | H'80 | H'EC |
| 237 | H'FF | H'FF | H'80 | H'80 | H'ED |
| 238 | H'FF | H'FF | H'80 | H'80 | H'EE |
| 239 | H'FF | H'FF | H'80 | H'80 | H'EF |
| 240 | H'FF | H'FF | H'80 | H'80 | H'F0 |
| 241 | H'FF | H'FF | H'80 | H'80 | H'F1 |
| 242 | H'FF | H'FF | H'80 | H'80 | H'F2 |
| 243 | H'FF | H'FF | H'80 | H'80 | H'F3 |
| 244 | H'FF | H'FF | H'80 | H'80 | H'F4 |
| 245 | H'FF | H'FF | H'80 | H'80 | H'F5 |
| 246 | H'FF | H'FF | H'80 | H'80 | H'F6 |
| 247 | H'FF | H'FF | H'80 | H'80 | H'F7 |
| 248 | H'FF | H'FF | H'80 | H'80 | H'F8 |
| 249 | H'FF | H'FF | H'80 | H'80 | H'F9 |
| 250 | H'FF | H'FF | H'80 | H'80 | H'FA |
| 251 | H'FF | H'FF | H'80 | H'80 | H'FB |
| 252 | H'FF | H'FF | H'80 | H'80 | H'FC |
| 253 | H'FF | H'FF | H'80 | H'80 | H'FD |
| 254 | H'FF | H'FF | H'80 | H'80 | H'FE |
| 255 | H'FF | H'FF | H'80 | H'80 | H'FF |

Table 34.13 The setting 2 of each LUT

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 0 | H'00 | H'00 | H'80 | H'80 | H'00 |
| 1 | H'13 | H'13 | H'80 | H'80 | H'01 |
| 2 | H'24 | H'24 | H'80 | H'80 | H'02 |
| 3 | H'34 | H'34 | H'80 | H'80 | H'03 |
| 4 | H'43 | H'43 | H'80 | H'80 | H'04 |
| 5 | H'51 | H'51 | H'80 | H'80 | H'05 |
| 6 | H'5E | H'5E | H'80 | H'80 | H'06 |
| 7 | H'6A | H'6A | H'80 | H'80 | H'07 |
| 8 | H'76 | H'76 | H'80 | H'80 | H'08 |
| 9 | H'81 | H'81 | H'80 | H'80 | H'09 |
| 10 | H'8C | H'8C | H'80 | H'80 | H'0A |
| 11 | H'95 | H'95 | H'80 | H'80 | H'0B |
| 12 | H'9E | H'9E | H'80 | H'80 | H'0C |
| 13 | H'A7 | H'A7 | H'80 | H'80 | H'0D |
| 14 | H'AF | H'AF | H'80 | H'80 | H'0E |
| 15 | H'B6 | H'B6 | H'80 | H'80 | H'0F |
| 16 | H'BD | H'BD | H'80 | H'80 | H'10 |
| 17 | H'C3 | H'C3 | H'80 | H'80 | H'11 |
| 18 | H'C9 | H'C9 | H'80 | H'80 | H'12 |
| 19 | H'CF | H'CF | H'80 | H'80 | H'13 |
| 20 | H'D4 | H'D4 | H'80 | H'80 | H'14 |
| 21 | H'D8 | H'D8 | H'80 | H'80 | H'15 |
| 22 | H'DD | H'DD | H'80 | H'80 | H'16 |
| 23 | H'E0 | H'E0 | H'80 | H'80 | H'17 |
| 24 | H'E4 | H'E4 | H'80 | H'80 | H'18 |
| 25 | H'E7 | H'E7 | H'80 | H'80 | H'19 |
| 26 | H'EA | H'EA | H'80 | H'80 | H'1A |
| 27 | H'ED | H'ED | H'80 | H'80 | H'1B |
| 28 | H'EF | H'EF | H'80 | H'80 | H'1C |
| 29 | H'F1 | H'F1 | H'80 | H'80 | H'1D |
| 30 | H'F3 | H'F3 | H'80 | H'80 | H'1E |
| 31 | H'F5 | H'F5 | H'80 | H'80 | H'1F |
| 32 | H'F6 | H'F6 | H'80 | H'80 | H'20 |
| 33 | H'F8 | H'F8 | H'80 | H'80 | H'21 |
| 34 | H'F9 | H'F9 | H'80 | H'80 | H'22 |
| 35 | H'FA | H'FA | H'80 | H'80 | H'23 |
| 36 | H'FB | H'FB | H'80 | H'80 | H'24 |
| 37 | H'FC | H'FC | H'80 | H'80 | H'25 |
| 38 | H'FC | H'FC | H'80 | H'80 | H'26 |
| 39 | H'FD | H'FD | H'80 | H'80 | H'27 |
| 40 | H'FD | H'FD | H'80 | H'80 | H'28 |
| 41 | H'FE | H'FE | H'80 | H'80 | H'29 |
| 42 | H'FE | H'FE | H'80 | H'80 | H'2A |
| 43 | H'FE | H'FE | H'80 | H'80 | H'2B |
| 44 | H'FE | H'FE | H'80 | H'80 | H'2C |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 45 | H'FF | H'FF | H'80 | H'80 | H'2D |
| 46 | H'FF | H'FF | H'80 | H'80 | H'2E |
| 47 | H'FF | H'FF | H'80 | H'80 | H'2F |
| 48 | H'FF | H'FF | H'80 | H'80 | H'30 |
| 49 | H'FF | H'FF | H'80 | H'80 | H'31 |
| 50 | H'FF | H'FF | H'80 | H'80 | H'32 |
| 51 | H'FF | H'FF | H'80 | H'80 | H'33 |
| 52 | H'FF | H'FF | H'80 | H'80 | H'34 |
| 53 | H'FF | H'FF | H'80 | H'80 | H'35 |
| 54 | H'FF | H'FF | H'80 | H'80 | H'36 |
| 55 | H'FF | H'FF | H'80 | H'80 | H'37 |
| 56 | H'FF | H'FF | H'80 | H'80 | H'38 |
| 57 | H'FF | H'FF | H'80 | H'80 | H'39 |
| 58 | H'FF | H'FF | H'80 | H'80 | H'3A |
| 59 | H'FF | H'FF | H'80 | H'80 | H'3B |
| 60 | H'FF | H'FF | H'80 | H'80 | H'3C |
| 61 | H'FF | H'FF | H'80 | H'80 | H'3D |
| 62 | H'FF | H'FF | H'80 | H'80 | H'3E |
| 63 | H'FF | H'FF | H'80 | H'80 | H'3F |
| 64 | H'FF | H'FF | H'80 | H'80 | H'40 |
| 65 | H'FF | H'FF | H'80 | H'80 | H'41 |
| 66 | H'FF | H'FF | H'80 | H'80 | H'42 |
| 67 | H'FF | H'FF | H'80 | H'80 | H'43 |
| 68 | H'FF | H'FF | H'80 | H'80 | H'44 |
| 69 | H'FF | H'FF | H'80 | H'80 | H'45 |
| 70 | H'FF | H'FF | H'80 | H'80 | H'46 |
| 71 | H'FF | H'FF | H'80 | H'80 | H'47 |
| 72 | H'FF | H'FF | H'80 | H'80 | H'48 |
| 73 | H'FF | H'FF | H'80 | H'80 | H'49 |
| 74 | H'FF | H'FF | H'80 | H'80 | H'4A |
| 75 | H'FF | H'FF | H'80 | H'80 | H'4B |
| 76 | H'FF | H'FF | H'80 | H'80 | H'4C |
| 77 | H'FF | H'FF | H'80 | H'80 | H'4D |
| 78 | H'FF | H'FF | H'80 | H'80 | H'4E |
| 79 | H'FF | H'FF | H'80 | H'80 | H'4F |
| 80 | H'FF | H'FF | H'80 | H'80 | H'50 |
| 81 | H'FF | H'FF | H'80 | H'80 | H'51 |
| 82 | H'FF | H'FF | H'80 | H'80 | H'52 |
| 83 | H'FF | H'FF | H'80 | H'80 | H'53 |
| 84 | H'FF | H'FF | H'80 | H'80 | H'54 |
| 85 | H'FF | H'FF | H'80 | H'80 | H'55 |
| 86 | H'FF | H'FF | H'80 | H'80 | H'56 |
| 87 | H'FF | H'FF | H'80 | H'80 | H'57 |
| 88 | H'FF | H'FF | H'80 | H'80 | H'58 |
| 89 | H'FF | H'FF | H'80 | H'80 | H'59 |
| 90 | H'FF | H'FF | H'80 | H'80 | H'5A |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 91 | H'FF | H'FF | H'80 | H'80 | H'5B |
| 92 | H'FF | H'FF | H'80 | H'80 | H'5C |
| 93 | H'FF | H'FF | H'80 | H'80 | H'5D |
| 94 | H'FF | H'FF | H'80 | H'80 | H'5E |
| 95 | H'FF | H'FF | H'80 | H'80 | H'5F |
| 96 | H'FF | H'FF | H'80 | H'80 | H'60 |
| 97 | H'FF | H'FF | H'80 | H'80 | H'61 |
| 98 | H'FF | H'FF | H'80 | H'80 | H'62 |
| 99 | H'FF | H'FF | H'80 | H'80 | H'63 |
| 100 | H'FF | H'FF | H'80 | H'80 | H'64 |
| 101 | H'FF | H'FF | H'80 | H'80 | H'65 |
| 102 | H'FF | H'FF | H'80 | H'80 | H'66 |
| 103 | H'FF | H'FF | H'80 | H'80 | H'67 |
| 104 | H'FF | H'FF | H'80 | H'80 | H'68 |
| 105 | H'FF | H'FF | H'80 | H'80 | H'69 |
| 106 | H'FF | H'FF | H'80 | H'80 | H'6A |
| 107 | H'FF | H'FF | H'80 | H'80 | H'6B |
| 108 | H'FF | H'FF | H'80 | H'80 | H'6C |
| 109 | H'FF | H'FF | H'80 | H'80 | H'6D |
| 110 | H'FF | H'FF | H'80 | H'80 | H'6E |
| 111 | H'FF | H'FF | H'80 | H'80 | H'6F |
| 112 | H'FF | H'FF | H'80 | H'80 | H'70 |
| 113 | H'FF | H'FF | H'80 | H'80 | H'71 |
| 114 | H'FF | H'FF | H'80 | H'80 | H'72 |
| 115 | H'FF | H'FF | H'80 | H'80 | H'73 |
| 116 | H'FF | H'FF | H'80 | H'80 | H'74 |
| 117 | H'FF | H'FF | H'80 | H'80 | H'75 |
| 118 | H'FF | H'FF | H'80 | H'80 | H'76 |
| 119 | H'FF | H'FF | H'80 | H'80 | H'77 |
| 120 | H'FF | H'FF | H'80 | H'80 | H'78 |
| 121 | H'FF | H'FF | H'80 | H'80 | H'79 |
| 122 | H'FF | H'FF | H'80 | H'80 | H'7A |
| 123 | H'FF | H'FF | H'80 | H'80 | H'7B |
| 124 | H'FF | H'FF | H'80 | H'80 | H'7C |
| 125 | H'FF | H'FF | H'80 | H'80 | H'7D |
| 126 | H'FF | H'FF | H'80 | H'80 | H'7E |
| 127 | H'FF | H'FF | H'80 | H'80 | H'7F |
| 128 | H'FF | H'FF | H'80 | H'80 | H'80 |
| 129 | H'FF | H'FF | H'80 | H'80 | H'81 |
| 130 | H'FF | H'FF | H'80 | H'80 | H'82 |
| 131 | H'FF | H'FF | H'80 | H'80 | H'83 |
| 132 | H'FF | H'FF | H'80 | H'80 | H'84 |
| 133 | H'FF | H'FF | H'80 | H'80 | H'85 |
| 134 | H'FF | H'FF | H'80 | H'80 | H'86 |
| 135 | H'FF | H'FF | H'80 | H'80 | H'87 |
| 136 | H'FF | H'FF | H'80 | H'80 | H'88 |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 137 | H'FF | H'FF | H'80 | H'80 | H'89 |
| 138 | H'FF | H'FF | H'80 | H'80 | H'8A |
| 139 | H'FF | H'FF | H'80 | H'80 | H'8B |
| 140 | H'FF | H'FF | H'80 | H'80 | H'8C |
| 141 | H'FF | H'FF | H'80 | H'80 | H'8D |
| 142 | H'FF | H'FF | H'80 | H'80 | H'8E |
| 143 | H'FF | H'FF | H'80 | H'80 | H'8F |
| 144 | H'FF | H'FF | H'80 | H'80 | H'90 |
| 145 | H'FF | H'FF | H'80 | H'80 | H'91 |
| 146 | H'FF | H'FF | H'80 | H'80 | H'92 |
| 147 | H'FF | H'FF | H'80 | H'80 | H'93 |
| 148 | H'FF | H'FF | H'80 | H'80 | H'94 |
| 149 | H'FF | H'FF | H'80 | H'80 | H'95 |
| 150 | H'FF | H'FF | H'80 | H'80 | H'96 |
| 151 | H'FF | H'FF | H'80 | H'80 | H'97 |
| 152 | H'FF | H'FF | H'80 | H'80 | H'98 |
| 153 | H'FF | H'FF | H'80 | H'80 | H'99 |
| 154 | H'FF | H'FF | H'80 | H'80 | H'9A |
| 155 | H'FF | H'FF | H'80 | H'80 | H'9B |
| 156 | H'FF | H'FF | H'80 | H'80 | H'9C |
| 157 | H'FF | H'FF | H'80 | H'80 | H'9D |
| 158 | H'FF | H'FF | H'80 | H'80 | H'9E |
| 159 | H'FF | H'FF | H'80 | H'80 | H'9F |
| 160 | H'FF | H'FF | H'80 | H'80 | H'A0 |
| 161 | H'FF | H'FF | H'80 | H'80 | H'A1 |
| 162 | H'FF | H'FF | H'80 | H'80 | H'A2 |
| 163 | H'FF | H'FF | H'80 | H'80 | H'A3 |
| 164 | H'FF | H'FF | H'80 | H'80 | H'A4 |
| 165 | H'FF | H'FF | H'80 | H'80 | H'A5 |
| 166 | H'FF | H'FF | H'80 | H'80 | H'A6 |
| 167 | H'FF | H'FF | H'80 | H'80 | H'A7 |
| 168 | H'FF | H'FF | H'80 | H'80 | H'A8 |
| 169 | H'FF | H'FF | H'80 | H'80 | H'A9 |
| 170 | H'FF | H'FF | H'80 | H'80 | H'AA |
| 171 | H'FF | H'FF | H'80 | H'80 | H'AB |
| 172 | H'FF | H'FF | H'80 | H'80 | H'AC |
| 173 | H'FF | H'FF | H'80 | H'80 | H'AD |
| 174 | H'FF | H'FF | H'80 | H'80 | H'AE |
| 175 | H'FF | H'FF | H'80 | H'80 | H'AF |
| 176 | H'FF | H'FF | H'80 | H'80 | H'B0 |
| 177 | H'FF | H'FF | H'80 | H'80 | H'B1 |
| 178 | H'FF | H'FF | H'80 | H'80 | H'B2 |
| 179 | H'FF | H'FF | H'80 | H'80 | H'B3 |
| 180 | H'FF | H'FF | H'80 | H'80 | H'B4 |
| 181 | H'FF | H'FF | H'80 | H'80 | H'B5 |
| 182 | H'FF | H'FF | H'80 | H'80 | H'B6 |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 183 | H'FF | H'FF | H'80 | H'80 | H'B7 |
| 184 | H'FF | H'FF | H'80 | H'80 | H'B8 |
| 185 | H'FF | H'FF | H'80 | H'80 | H'B9 |
| 186 | H'FF | H'FF | H'80 | H'80 | H'BA |
| 187 | H'FF | H'FF | H'80 | H'80 | H'BB |
| 188 | H'FF | H'FF | H'80 | H'80 | H'BC |
| 189 | H'FF | H'FF | H'80 | H'80 | H'BD |
| 190 | H'FF | H'FF | H'80 | H'80 | H'BE |
| 191 | H'FF | H'FF | H'80 | H'80 | H'BF |
| 192 | H'FF | H'FF | H'80 | H'80 | H'C0 |
| 193 | H'FF | H'FF | H'80 | H'80 | H'C1 |
| 194 | H'FF | H'FF | H'80 | H'80 | H'C2 |
| 195 | H'FF | H'FF | H'80 | H'80 | H'C3 |
| 196 | H'FF | H'FF | H'80 | H'80 | H'C4 |
| 197 | H'FF | H'FF | H'80 | H'80 | H'C5 |
| 198 | H'FF | H'FF | H'80 | H'80 | H'C6 |
| 199 | H'FF | H'FF | H'80 | H'80 | H'C7 |
| 200 | H'FF | H'FF | H'80 | H'80 | H'C8 |
| 201 | H'FF | H'FF | H'80 | H'80 | H'C9 |
| 202 | H'FF | H'FF | H'80 | H'80 | H'CA |
| 203 | H'FF | H'FF | H'80 | H'80 | H'CB |
| 204 | H'FF | H'FF | H'80 | H'80 | H'CC |
| 205 | H'FF | H'FF | H'80 | H'80 | H'CD |
| 206 | H'FF | H'FF | H'80 | H'80 | H'CE |
| 207 | H'FF | H'FF | H'80 | H'80 | H'CF |
| 208 | H'FF | H'FF | H'80 | H'80 | H'D0 |
| 209 | H'FF | H'FF | H'80 | H'80 | H'D1 |
| 210 | H'FF | H'FF | H'80 | H'80 | H'D2 |
| 211 | H'FF | H'FF | H'80 | H'80 | H'D3 |
| 212 | H'FF | H'FF | H'80 | H'80 | H'D4 |
| 213 | H'FF | H'FF | H'80 | H'80 | H'D5 |
| 214 | H'FF | H'FF | H'80 | H'80 | H'D6 |
| 215 | H'FF | H'FF | H'80 | H'80 | H'D7 |
| 216 | H'FF | H'FF | H'80 | H'80 | H'D8 |
| 217 | H'FF | H'FF | H'80 | H'80 | H'D9 |
| 218 | H'FF | H'FF | H'80 | H'80 | H'DA |
| 219 | H'FF | H'FF | H'80 | H'80 | H'DB |
| 220 | H'FF | H'FF | H'80 | H'80 | H'DC |
| 221 | H'FF | H'FF | H'80 | H'80 | H'DD |
| 222 | H'FF | H'FF | H'80 | H'80 | H'DE |
| 223 | H'FF | H'FF | H'80 | H'80 | H'DF |
| 224 | H'FF | H'FF | H'80 | H'80 | H'E0 |
| 225 | H'FF | H'FF | H'80 | H'80 | H'E1 |
| 226 | H'FF | H'FF | H'80 | H'80 | H'E2 |
| 227 | H'FF | H'FF | H'80 | H'80 | H'E3 |
| 228 | H'FF | H'FF | H'80 | H'80 | H'E4 |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 229 | H'FF | H'FF | H'80 | H'80 | H'E5 |
| 230 | H'FF | H'FF | H'80 | H'80 | H'E6 |
| 231 | H'FF | H'FF | H'80 | H'80 | H'E7 |
| 232 | H'FF | H'FF | H'80 | H'80 | H'E8 |
| 233 | H'FF | H'FF | H'80 | H'80 | H'E9 |
| 234 | H'FF | H'FF | H'80 | H'80 | H'EA |
| 235 | H'FF | H'FF | H'80 | H'80 | H'EB |
| 236 | H'FF | H'FF | H'80 | H'80 | H'EC |
| 237 | H'FF | H'FF | H'80 | H'80 | H'ED |
| 238 | H'FF | H'FF | H'80 | H'80 | H'EE |
| 239 | H'FF | H'FF | H'80 | H'80 | H'EF |
| 240 | H'FF | H'FF | H'80 | H'80 | H'F0 |
| 241 | H'FF | H'FF | H'80 | H'80 | H'F1 |
| 242 | H'FF | H'FF | H'80 | H'80 | H'F2 |
| 243 | H'FF | H'FF | H'80 | H'80 | H'F3 |
| 244 | H'FF | H'FF | H'80 | H'80 | H'F4 |
| 245 | H'FF | H'FF | H'80 | H'80 | H'F5 |
| 246 | H'FF | H'FF | H'80 | H'80 | H'F6 |
| 247 | H'FF | H'FF | H'80 | H'80 | H'F7 |
| 248 | H'FF | H'FF | H'80 | H'80 | H'F8 |
| 249 | H'FF | H'FF | H'80 | H'80 | H'F9 |
| 250 | H'FF | H'FF | H'80 | H'80 | H'FA |
| 251 | H'FF | H'FF | H'80 | H'80 | H'FB |
| 252 | H'FF | H'FF | H'80 | H'80 | H'FC |
| 253 | H'FF | H'FF | H'80 | H'80 | H'FD |
| 254 | H'FF | H'FF | H'80 | H'80 | H'FE |
| 255 | H'FF | H'FF | H'80 | H'80 | H'FF |

Table 34.14 The setting 3 of each LUT

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 0 | H'00 | H'00 | H'80 | H'80 | H'00 |
| 1 | H'0E | H'0E | H'80 | H'80 | H'01 |
| 2 | H'19 | H'19 | H'80 | H'80 | H'02 |
| 3 | H'24 | H'24 | H'80 | H'80 | H'03 |
| 4 | H'2F | H'2F | H'80 | H'80 | H'04 |
| 5 | H'39 | H'39 | H'80 | H'80 | H'05 |
| 6 | H'43 | H'43 | H'80 | H'80 | H'06 |
| 7 | H'4C | H'4C | H'80 | H'80 | H'07 |
| 8 | H'55 | H'55 | H'80 | H'80 | H'08 |
| 9 | H'5E | H'5E | H'80 | H'80 | H'09 |
| 10 | H'66 | H'66 | H'80 | H'80 | H'0A |
| 11 | H'6E | H'6E | H'80 | H'80 | H'0B |
| 12 | H'76 | H'76 | H'80 | H'80 | H'0C |
| 13 | H'7E | H'7E | H'80 | H'80 | H'0D |
| 14 | H'85 | H'85 | H'80 | H'80 | H'0E |
| 15 | H'8C | H'8C | H'80 | H'80 | H'0F |
| 16 | H'92 | H'92 | H'80 | H'80 | H'10 |
| 17 | H'98 | H'98 | H'80 | H'80 | H'11 |
| 18 | H'9E | H'9E | H'80 | H'80 | H'12 |
| 19 | H'A4 | H'A4 | H'80 | H'80 | H'13 |
| 20 | H'AA | H'AA | H'80 | H'80 | H'14 |
| 21 | H'AF | H'AF | H'80 | H'80 | H'15 |
| 22 | H'B4 | H'B4 | H'80 | H'80 | H'16 |
| 23 | H'B8 | H'B8 | H'80 | H'80 | H'17 |
| 24 | H'BD | H'BD | H'80 | H'80 | H'18 |
| 25 | H'C1 | H'C1 | H'80 | H'80 | H'19 |
| 26 | H'C5 | H'C5 | H'80 | H'80 | H'1A |
| 27 | H'C9 | H'C9 | H'80 | H'80 | H'1B |
| 28 | H'CD | H'CD | H'80 | H'80 | H'1C |
| 29 | H'D0 | H'D0 | H'80 | H'80 | H'1D |
| 30 | H'D4 | H'D4 | H'80 | H'80 | H'1E |
| 31 | H'D7 | H'D7 | H'80 | H'80 | H'1F |
| 32 | H'DA | H'DA | H'80 | H'80 | H'20 |
| 33 | H'DD | H'DD | H'80 | H'80 | H'21 |
| 34 | H'DF | H'DF | H'80 | H'80 | H'22 |
| 35 | H'E2 | H'E2 | H'80 | H'80 | H'23 |
| 36 | H'E4 | H'E4 | H'80 | H'80 | H'24 |
| 37 | H'E6 | H'E6 | H'80 | H'80 | H'25 |
| 38 | H'E8 | H'E8 | H'80 | H'80 | H'26 |
| 39 | H'EA | H'EA | H'80 | H'80 | H'27 |
| 40 | H'EC | H'EC | H'80 | H'80 | H'28 |
| 41 | H'EE | H'EE | H'80 | H'80 | H'29 |
| 42 | H'EF | H'EF | H'80 | H'80 | H'2A |
| 43 | H'F1 | H'F1 | H'80 | H'80 | H'2B |
| 44 | H'F2 | H'F2 | H'80 | H'80 | H'2C |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 45 | H'F3 | H'F3 | H'80 | H'80 | H'2D |
| 46 | H'F4 | H'F4 | H'80 | H'80 | H'2E |
| 47 | H'F5 | H'F5 | H'80 | H'80 | H'2F |
| 48 | H'F6 | H'F6 | H'80 | H'80 | H'30 |
| 49 | H'F7 | H'F7 | H'80 | H'80 | H'31 |
| 50 | H'F8 | H'F8 | H'80 | H'80 | H'32 |
| 51 | H'F9 | H'F9 | H'80 | H'80 | H'33 |
| 52 | H'FA | H'FA | H'80 | H'80 | H'34 |
| 53 | H'FA | H'FA | H'80 | H'80 | H'35 |
| 54 | H'FB | H'FB | H'80 | H'80 | H'36 |
| 55 | H'FB | H'FB | H'80 | H'80 | H'37 |
| 56 | H'FC | H'FC | H'80 | H'80 | H'38 |
| 57 | H'FC | H'FC | H'80 | H'80 | H'39 |
| 58 | H'FD | H'FD | H'80 | H'80 | H'3A |
| 59 | H'FD | H'FD | H'80 | H'80 | H'3B |
| 60 | H'FD | H'FD | H'80 | H'80 | H'3C |
| 61 | H'FE | H'FE | H'80 | H'80 | H'3D |
| 62 | H'FE | H'FE | H'80 | H'80 | H'3E |
| 63 | H'FE | H'FE | H'80 | H'80 | H'3F |
| 64 | H'FE | H'FE | H'80 | H'80 | H'40 |
| 65 | H'FE | H'FE | H'80 | H'80 | H'41 |
| 66 | H'FE | H'FE | H'80 | H'80 | H'42 |
| 67 | H'FF | H'FF | H'80 | H'80 | H'43 |
| 68 | H'FF | H'FF | H'80 | H'80 | H'44 |
| 69 | H'FF | H'FF | H'80 | H'80 | H'45 |
| 70 | H'FF | H'FF | H'80 | H'80 | H'46 |
| 71 | H'FF | H'FF | H'80 | H'80 | H'47 |
| 72 | H'FF | H'FF | H'80 | H'80 | H'48 |
| 73 | H'FF | H'FF | H'80 | H'80 | H'49 |
| 74 | H'FF | H'FF | H'80 | H'80 | H'4A |
| 75 | H'FF | H'FF | H'80 | H'80 | H'4B |
| 76 | H'FF | H'FF | H'80 | H'80 | H'4C |
| 77 | H'FF | H'FF | H'80 | H'80 | H'4D |
| 78 | H'FF | H'FF | H'80 | H'80 | H'4E |
| 79 | H'FF | H'FF | H'80 | H'80 | H'4F |
| 80 | H'FF | H'FF | H'80 | H'80 | H'50 |
| 81 | H'FF | H'FF | H'80 | H'80 | H'51 |
| 82 | H'FF | H'FF | H'80 | H'80 | H'52 |
| 83 | H'FF | H'FF | H'80 | H'80 | H'53 |
| 84 | H'FF | H'FF | H'80 | H'80 | H'54 |
| 85 | H'FF | H'FF | H'80 | H'80 | H'55 |
| 86 | H'FF | H'FF | H'80 | H'80 | H'56 |
| 87 | H'FF | H'FF | H'80 | H'80 | H'57 |
| 88 | H'FF | H'FF | H'80 | H'80 | H'58 |
| 89 | H'FF | H'FF | H'80 | H'80 | H'59 |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 90 | H'FF | H'FF | H'80 | H'80 | H'5A |
| 91 | H'FF | H'FF | H'80 | H'80 | H'5B |
| 92 | H'FF | H'FF | H'80 | H'80 | H'5C |
| 93 | H'FF | H'FF | H'80 | H'80 | H'5D |
| 94 | H'FF | H'FF | H'80 | H'80 | H'5E |
| 95 | H'FF | H'FF | H'80 | H'80 | H'5F |
| 96 | H'FF | H'FF | H'80 | H'80 | H'60 |
| 97 | H'FF | H'FF | H'80 | H'80 | H'61 |
| 98 | H'FF | H'FF | H'80 | H'80 | H'62 |
| 99 | H'FF | H'FF | H'80 | H'80 | H'63 |
| 100 | H'FF | H'FF | H'80 | H'80 | H'64 |
| 101 | H'FF | H'FF | H'80 | H'80 | H'65 |
| 102 | H'FF | H'FF | H'80 | H'80 | H'66 |
| 103 | H'FF | H'FF | H'80 | H'80 | H'67 |
| 104 | H'FF | H'FF | H'80 | H'80 | H'68 |
| 105 | H'FF | H'FF | H'80 | H'80 | H'69 |
| 106 | H'FF | H'FF | H'80 | H'80 | H'6A |
| 107 | H'FF | H'FF | H'80 | H'80 | H'6B |
| 108 | H'FF | H'FF | H'80 | H'80 | H'6C |
| 109 | H'FF | H'FF | H'80 | H'80 | H'6D |
| 110 | H'FF | H'FF | H'80 | H'80 | H'6E |
| 111 | H'FF | H'FF | H'80 | H'80 | H'6F |
| 112 | H'FF | H'FF | H'80 | H'80 | H'70 |
| 113 | H'FF | H'FF | H'80 | H'80 | H'71 |
| 114 | H'FF | H'FF | H'80 | H'80 | H'72 |
| 115 | H'FF | H'FF | H'80 | H'80 | H'73 |
| 116 | H'FF | H'FF | H'80 | H'80 | H'74 |
| 117 | H'FF | H'FF | H'80 | H'80 | H'75 |
| 118 | H'FF | H'FF | H'80 | H'80 | H'76 |
| 119 | H'FF | H'FF | H'80 | H'80 | H'77 |
| 120 | H'FF | H'FF | H'80 | H'80 | H'78 |
| 121 | H'FF | H'FF | H'80 | H'80 | H'79 |
| 122 | H'FF | H'FF | H'80 | H'80 | H'7A |
| 123 | H'FF | H'FF | H'80 | H'80 | H'7B |
| 124 | H'FF | H'FF | H'80 | H'80 | H'7C |
| 125 | H'FF | H'FF | H'80 | H'80 | H'7D |
| 126 | H'FF | H'FF | H'80 | H'80 | H'7E |
| 127 | H'FF | H'FF | H'80 | H'80 | H'7F |
| 128 | H'FF | H'FF | H'80 | H'80 | H'80 |
| 129 | H'FF | H'FF | H'80 | H'80 | H'81 |
| 130 | H'FF | H'FF | H'80 | H'80 | H'82 |
| 131 | H'FF | H'FF | H'80 | H'80 | H'83 |
| 132 | H'FF | H'FF | H'80 | H'80 | H'84 |
| 133 | H'FF | H'FF | H'80 | H'80 | H'85 |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 134 | H'FF | H'FF | H'80 | H'80 | H'86 |
| 135 | H'FF | H'FF | H'80 | H'80 | H'87 |
| 136 | H'FF | H'FF | H'80 | H'80 | H'88 |
| 137 | H'FF | H'FF | H'80 | H'80 | H'89 |
| 138 | H'FF | H'FF | H'80 | H'80 | H'8A |
| 139 | H'FF | H'FF | H'80 | H'80 | H'8B |
| 140 | H'FF | H'FF | H'80 | H'80 | H'8C |
| 141 | H'FF | H'FF | H'80 | H'80 | H'8D |
| 142 | H'FF | H'FF | H'80 | H'80 | H'8E |
| 143 | H'FF | H'FF | H'80 | H'80 | H'8F |
| 144 | H'FF | H'FF | H'80 | H'80 | H'90 |
| 145 | H'FF | H'FF | H'80 | H'80 | H'91 |
| 146 | H'FF | H'FF | H'80 | H'80 | H'92 |
| 147 | H'FF | H'FF | H'80 | H'80 | H'93 |
| 148 | H'FF | H'FF | H'80 | H'80 | H'94 |
| 149 | H'FF | H'FF | H'80 | H'80 | H'95 |
| 150 | H'FF | H'FF | H'80 | H'80 | H'96 |
| 151 | H'FF | H'FF | H'80 | H'80 | H'97 |
| 152 | H'FF | H'FF | H'80 | H'80 | H'98 |
| 153 | H'FF | H'FF | H'80 | H'80 | H'99 |
| 154 | H'FF | H'FF | H'80 | H'80 | H'9A |
| 155 | H'FF | H'FF | H'80 | H'80 | H'9B |
| 156 | H'FF | H'FF | H'80 | H'80 | H'9C |
| 157 | H'FF | H'FF | H'80 | H'80 | H'9D |
| 158 | H'FF | H'FF | H'80 | H'80 | H'9E |
| 159 | H'FF | H'FF | H'80 | H'80 | H'9F |
| 160 | H'FF | H'FF | H'80 | H'80 | H'A0 |
| 161 | H'FF | H'FF | H'80 | H'80 | H'A1 |
| 162 | H'FF | H'FF | H'80 | H'80 | H'A2 |
| 163 | H'FF | H'FF | H'80 | H'80 | H'A3 |
| 164 | H'FF | H'FF | H'80 | H'80 | H'A4 |
| 165 | H'FF | H'FF | H'80 | H'80 | H'A5 |
| 166 | H'FF | H'FF | H'80 | H'80 | H'A6 |
| 167 | H'FF | H'FF | H'80 | H'80 | H'A7 |
| 168 | H'FF | H'FF | H'80 | H'80 | H'A8 |
| 169 | H'FF | H'FF | H'80 | H'80 | H'A9 |
| 170 | H'FF | H'FF | H'80 | H'80 | H'AA |
| 171 | H'FF | H'FF | H'80 | H'80 | H'AB |
| 172 | H'FF | H'FF | H'80 | H'80 | H'AC |
| 173 | H'FF | H'FF | H'80 | H'80 | H'AD |
| 174 | H'FF | H'FF | H'80 | H'80 | H'AE |
| 175 | H'FF | H'FF | H'80 | H'80 | H'AF |
| 176 | H'FF | H'FF | H'80 | H'80 | H'B0 |
| 177 | H'FF | H'FF | H'80 | H'80 | H'B1 |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 178 | H'FF | H'FF | H'80 | H'80 | H'B2 |
| 179 | H'FF | H'FF | H'80 | H'80 | H'B3 |
| 180 | H'FF | H'FF | H'80 | H'80 | H'B4 |
| 181 | H'FF | H'FF | H'80 | H'80 | H'B5 |
| 182 | H'FF | H'FF | H'80 | H'80 | H'B6 |
| 183 | H'FF | H'FF | H'80 | H'80 | H'B7 |
| 184 | H'FF | H'FF | H'80 | H'80 | H'B8 |
| 185 | H'FF | H'FF | H'80 | H'80 | H'B9 |
| 186 | H'FF | H'FF | H'80 | H'80 | H'BA |
| 187 | H'FF | H'FF | H'80 | H'80 | H'BB |
| 188 | H'FF | H'FF | H'80 | H'80 | H'BC |
| 189 | H'FF | H'FF | H'80 | H'80 | H'BD |
| 190 | H'FF | H'FF | H'80 | H'80 | H'BE |
| 191 | H'FF | H'FF | H'80 | H'80 | H'BF |
| 192 | H'FF | H'FF | H'80 | H'80 | H'C0 |
| 193 | H'FF | H'FF | H'80 | H'80 | H'C1 |
| 194 | H'FF | H'FF | H'80 | H'80 | H'C2 |
| 195 | H'FF | H'FF | H'80 | H'80 | H'C3 |
| 196 | H'FF | H'FF | H'80 | H'80 | H'C4 |
| 197 | H'FF | H'FF | H'80 | H'80 | H'C5 |
| 198 | H'FF | H'FF | H'80 | H'80 | H'C6 |
| 199 | H'FF | H'FF | H'80 | H'80 | H'C7 |
| 200 | H'FF | H'FF | H'80 | H'80 | H'C8 |
| 201 | H'FF | H'FF | H'80 | H'80 | H'C9 |
| 202 | H'FF | H'FF | H'80 | H'80 | H'CA |
| 203 | H'FF | H'FF | H'80 | H'80 | H'CB |
| 204 | H'FF | H'FF | H'80 | H'80 | H'CC |
| 205 | H'FF | H'FF | H'80 | H'80 | H'CD |
| 206 | H'FF | H'FF | H'80 | H'80 | H'CE |
| 207 | H'FF | H'FF | H'80 | H'80 | H'CF |
| 208 | H'FF | H'FF | H'80 | H'80 | H'D0 |
| 209 | H'FF | H'FF | H'80 | H'80 | H'D1 |
| 210 | H'FF | H'FF | H'80 | H'80 | H'D2 |
| 211 | H'FF | H'FF | H'80 | H'80 | H'D3 |
| 212 | H'FF | H'FF | H'80 | H'80 | H'D4 |
| 213 | H'FF | H'FF | H'80 | H'80 | H'D5 |
| 214 | H'FF | H'FF | H'80 | H'80 | H'D6 |
| 215 | H'FF | H'FF | H'80 | H'80 | H'D7 |
| 216 | H'FF | H'FF | H'80 | H'80 | H'D8 |
| 217 | H'FF | H'FF | H'80 | H'80 | H'D9 |
| 218 | H'FF | H'FF | H'80 | H'80 | H'DA |
| 219 | H'FF | H'FF | H'80 | H'80 | H'DB |
| 220 | H'FF | H'FF | H'80 | H'80 | H'DC |
| 221 | H'FF | H'FF | H'80 | H'80 | H'DD |

| Index | DIF_ADJ [] | SAD_ADJ [] | BLD_GAIN [] | DIF_GAIN [] | MDET [] |
|-------|------------|------------|-------------|-------------|---------|
| 222 | H'FF | H'FF | H'80 | H'80 | H'DE |
| 223 | H'FF | H'FF | H'80 | H'80 | H'DF |
| 224 | H'FF | H'FF | H'80 | H'80 | H'E0 |
| 225 | H'FF | H'FF | H'80 | H'80 | H'E1 |
| 226 | H'FF | H'FF | H'80 | H'80 | H'E2 |
| 227 | H'FF | H'FF | H'80 | H'80 | H'E3 |
| 228 | H'FF | H'FF | H'80 | H'80 | H'E4 |
| 229 | H'FF | H'FF | H'80 | H'80 | H'E5 |
| 230 | H'FF | H'FF | H'80 | H'80 | H'E6 |
| 231 | H'FF | H'FF | H'80 | H'80 | H'E7 |
| 232 | H'FF | H'FF | H'80 | H'80 | H'E8 |
| 233 | H'FF | H'FF | H'80 | H'80 | H'E9 |
| 234 | H'FF | H'FF | H'80 | H'80 | H'EA |
| 235 | H'FF | H'FF | H'80 | H'80 | H'EB |
| 236 | H'FF | H'FF | H'80 | H'80 | H'EC |
| 237 | H'FF | H'FF | H'80 | H'80 | H'ED |
| 238 | H'FF | H'FF | H'80 | H'80 | H'EE |
| 239 | H'FF | H'FF | H'80 | H'80 | H'EF |
| 240 | H'FF | H'FF | H'80 | H'80 | H'F0 |
| 241 | H'FF | H'FF | H'80 | H'80 | H'F1 |
| 242 | H'FF | H'FF | H'80 | H'80 | H'F2 |
| 243 | H'FF | H'FF | H'80 | H'80 | H'F3 |
| 244 | H'FF | H'FF | H'80 | H'80 | H'F4 |
| 245 | H'FF | H'FF | H'80 | H'80 | H'F5 |
| 246 | H'FF | H'FF | H'80 | H'80 | H'F6 |
| 247 | H'FF | H'FF | H'80 | H'80 | H'F7 |
| 248 | H'FF | H'FF | H'80 | H'80 | H'F8 |
| 249 | H'FF | H'FF | H'80 | H'80 | H'F9 |
| 250 | H'FF | H'FF | H'80 | H'80 | H'FA |
| 251 | H'FF | H'FF | H'80 | H'80 | H'FB |
| 252 | H'FF | H'FF | H'80 | H'80 | H'FC |
| 253 | H'FF | H'FF | H'80 | H'80 | H'FD |
| 254 | H'FF | H'FF | H'80 | H'80 | H'FE |
| 255 | H'FF | H'FF | H'80 | H'80 | H'FF |

34.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

34.4.1 Assignment in Memory Space

Make sure that FDP1 memory space shall be mapped to Non-Cache region.

34.4.2 FCPF connection

Bus access of FDP is carried out through FCPF (Frame Compression IP for FDP) module in RZ/G Gen2. So refer to also the section “35.4 FCP for FDP (FCPF)” in this manual.

There are some notices in FCPF connection system below.

- 1) FDP1 must be started, stopped in companion with FCPF as guided in section “Operation” of the section “35.4 FCP for FDP (FCPF)”.
- 2) There are some restrictions about register setting in FCPF connection system. Refer to the section “35.4 FCP for FDP (FCPF)” about restrictions of register setting.

34.4.3 Manipulate FDP with module stop, software reset of CPG

Please refer to Usage Notes of the section “35.4 FCP for FDP (FCPF)” about manipulating FCPF+FDP with CPG module stop, software reset.

34.4.4 Terminology

(1) Definition of operators and functions

The following operators, notations are defined for explaining the functions of FDP1.

clip0 (x)

Clip to 0 value if x is less than 0. $[x = (x < 0)? 0: x]$

clip3 (min, max, x)

The value x shall be clipped so that x shall be $\min < x < \max$. $[x = (x < \min)? \min: ((x > \max)? \max:)]$

(2) Unit

The unit is defined here for explaining FDP1’s functions.

[bpp]: Bits per pixel

FDP1 can handle a number of image formats. Each image format has different number of data bits. The unit [bpp] shows number of data bits for one pixel. For example, RGB 8bpp means that its format is RGB and the pixel consists of 8 data bits.

35. Frame Compression Processor (FCP)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

35.1 Overview

The FCP is a companion module of a video processing module in RZ/G 2nd generation. It provides data compression and decompression, data caching, and converting of AXI transaction in order to reduce the memory bandwidth.

There are three types of FCP. The configuration and behavior of each FCP highly depend on the module to be paired. The detail information is described in sub section as follows:

- 35.2 FCP for Codec (FCPC)
- 35.3 FCP for VSP (FCPV)
- 35.4 FCP for FDP (FCPF)

35.2 FCP for Codec (FCPC)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The FCPC is a frame compression module which reduces the bandwidth with Renesas lossless data compression and decompression (FCL) for the Renesas video codec processor (VCP4). And it also has the reference picture cache for the VCP4. Require software or the library to RENESAS for operating this module.

The FCPC is connected to following codec IP cores which make up the VCP4.

1. VDPB: H.265/HEVC decoder
2. VCPLF: Multi-codec decoder and encoder

There are two configurations of the FCPC. One is FCPCS and other is FCPCI.

Table 35.1 Configuration Type of the FCPC

| Type | Description | | | | | |
|-------------|---|-------------|-------------|--------|--------|-------------|
| FCPCS | <table border="1"> <tr> <td>RZ/G2H</td> <td>RZ/G2M V3.0</td> <td>RZ/G2N</td> <td>RZ/G2E</td> </tr> </table> Shared among VDPB, VCPLF, and iVDP1C. <table border="1"> <tr> <td>RZ/G2M V1.3</td> </tr> </table> Shared between VDPB and VCPLF. | RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2M V1.3 |
| RZ/G2H | RZ/G2M V3.0 | RZ/G2N | RZ/G2E | | | |
| RZ/G2M V1.3 | | | | | | |
| FCPCI | <table border="1"> <tr> <td>RZ/G2M V1.3</td> </tr> </table> Dedicated for iVDP1C or VCPLF. | RZ/G2M V1.3 | | | | |
| RZ/G2M V1.3 | | | | | | |

Table 35.2 Number of AXI master ports

| Type | Number of AXI master ports | | | | | |
|-------------|--|-------------|-------------|-------------|--------|--------|
| FCPCS | <table border="1"> <tr> <td>RZ/G2H</td> <td>RZ/G2M V3.0</td> </tr> </table> 2 (Each AXI master port is referred to as ch0 or ch1 in this User's Manual.) <table border="1"> <tr> <td>RZ/G2M V1.3</td> <td>RZ/G2N</td> <td>RZ/G2E</td> </tr> </table> 1 | RZ/G2H | RZ/G2M V3.0 | RZ/G2M V1.3 | RZ/G2N | RZ/G2E |
| RZ/G2H | RZ/G2M V3.0 | | | | | |
| RZ/G2M V1.3 | RZ/G2N | RZ/G2E | | | | |
| FCPCI | <table border="1"> <tr> <td>RZ/G2M V1.3</td> </tr> </table> 1 | RZ/G2M V1.3 | | | | |
| RZ/G2M V1.3 | | | | | | |

For detail connection of the FCPCS, FCPCI and iVDP1C, refer to Section 31.

For detail connection of the FCPCS, FCPCI and VCP4, refer to Section 32.

35.2.1 Features

The FCPC has the following features.

- Support for Lossless compression for decoded picture and decompression for reference picture
- Support 3840 x 2160 resolution
- Support for reference picture cache
- Lossless compression ratio is typically 50%. It may not be compressed since Lossless compression relies on the statistical nature of the original image.

Table 35.3 Main Function of the FCPC

| Item | Description |
|------------------------------------|---|
| Data compression and decompression | Renesas lossless (FCL) compression and decompression, Compression for decoded picture, and decompression for reference picture YCbCr 4:2:0 semi-planar and 8-bit depth are supported |
| Reference picture cache | FCPCS: 64 Kbyte FCPCI: 32 Kbyte |
| AXI transaction | Read access: Out-of-order completion allowed. Outstanding transactions supported. Write access: Out-of-order completion allowed. Outstanding transactions supported. |

35.2.2 Connected Module

Table 35.4 shows modules connected to the FCPC.

Table 35.4 Connected Module

| Module | Connected Module | Description |
|--------|------------------|---|
| FCPCS | VDPB | RZ/G2H RZ/G2M V1.3 RZ/G2M V3.0 RZ/G2N RZ/G2E H.265 decoder |
| | VCPLF | Multi-codec decoder and encoder |
| | iVDP1C | RZ/G2H RZ/G2M V3.0 RZ/G2N RZ/G2E Low delay decoder |
| FCPCI | iVDP1C | Low delay decoder |
| | VCPLF | RZ/G2M V1.3 Multi-codec decoder and encoder |

35.2.3 Usage Notes

35.2.3.1 Limitations on Software Reset, Module Standby and power-shutoff

Before executing following operations for FCPC, confirm that connected modules (VDPB, VCPLF or iVDP1C) are not encoding or decoding a picture. If connected modules are encoding or decoding a picture, wait completion of encoding or decoding.

- Execute Software Reset by CPG
- Enter Module Standby (stop clock supply) by CPG
- Enter power-shutoff status (stop power supply) by SYSC

35.3 FCP for VSP (FCPV)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The FCPV is a frame compression module which reduces the bandwidth with Renesas near-lossless data compression (FCNL) and conducts the bus accesses with 256bit AXI bus for the Renesas video signal processor (VSP). This module is under control of software for the VSP with the control function of the FCPV executed on the host CPU.

There are three configurations of the FCPV. Those are FCPVI, FCPVB, and FCPVD.

Table 35.5 Configuration Type of the FCPV

| Type | Description | Products |
|-------|---|---|
| FCPVI | For VSPI | RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E |
| FCPVB | RZ/G2H: For VSPBC and VSPBD RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E: For VSPB | RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E |
| FCPVD | RZ/G2H, RZ/G2N: For VSPD and VSPDL RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E: For VSPD | RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E |

35.3.1 Features

The FCPVI, FCPVB has the following features.

- Support FCNL compression with 50% bandwidth reduction
— Note: FCNL decompression is executed by DBSC4, DDR3L/4 memory controller.
- Support out-of-order for the whole outstanding transactions

The FCPVD of has the following features.

- Support out-of-order for the whole outstanding transactions

Table 35.6 Main Function of the FCPV

| Item | Description |
|------------------------|--|
| Data compression | FCNL compression for formats ARGB8888, YCbCr Planar 444/422/420, YCbCr Interleave 422 YUY2 |
| Picture size supported | Minimum: 1pixels x 1lines Maximum: 8190pixels x 8190 lines The picture sizes must follow the restriction specified in the section “33. Video Signal Processor (VSP2)” (sections “Usage Notes”, “Input Image Size” and “Output Image Size”). |
| Color format supported | FCNL compression: ARGB8888, YCbCr Planar 444/422/420, YCbCr Interleave 422 YUY2 Others: support all formats specified in the section “33. Video Signal Processor (VSP2)” (sections “Register Descriptions”, “RPFn Input Format Registers (VI6_RPFn_INFMT)” and “WPFn Output Format Registers (VI6_WPFn_OUTFMT)”). |
| AXI transactions | Read access: Access at 256 bytes boundary Support out-of-order for the whole outstanding transactions Support data interleaving Write access: Access at 256 bytes boundary Support out-of-order for the whole outstanding transactions |

35.3.2 Block Diagram

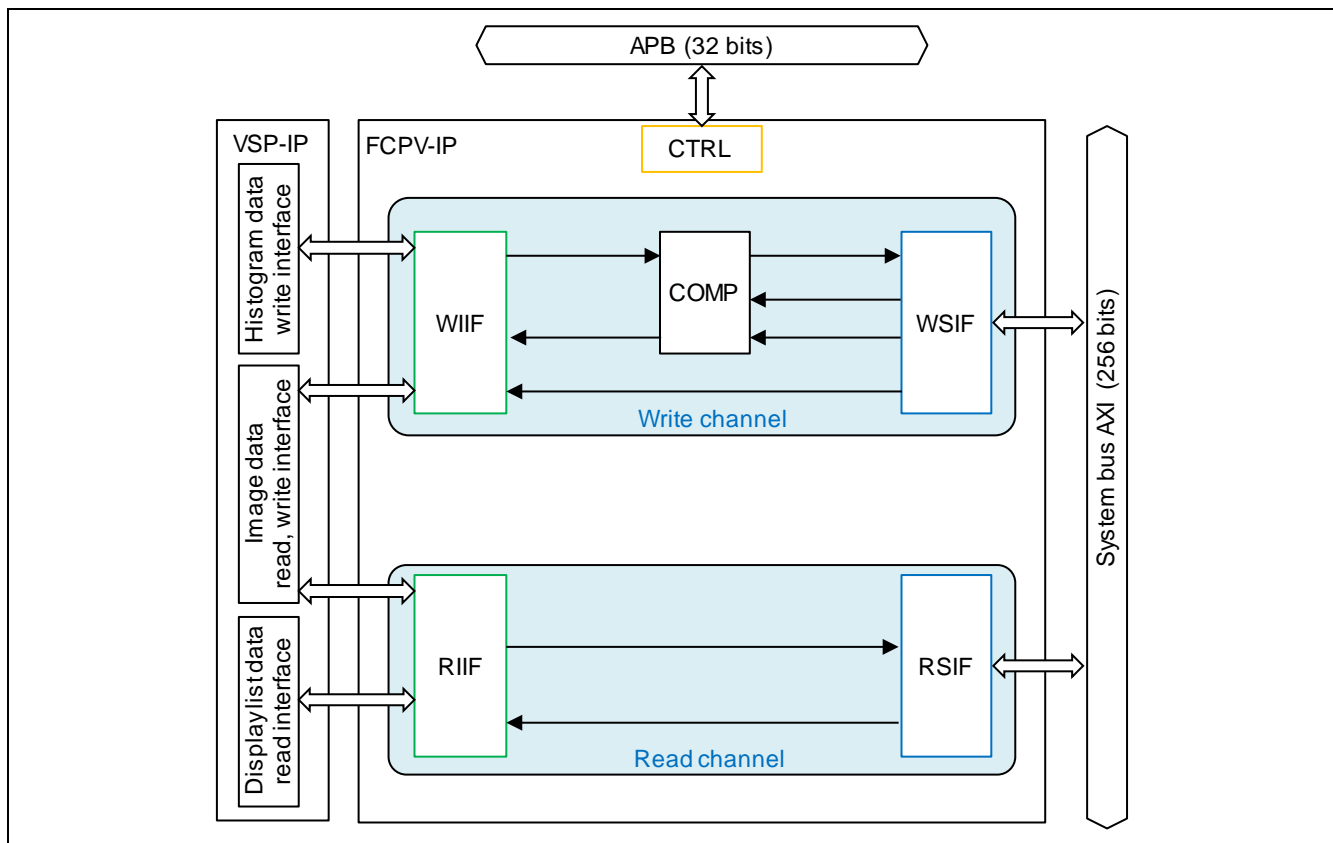


Figure 35.1 Overall Configuration of the FCPV

Table 35.7 Sub-module Description

| Abbr. | Description | Abbr. | Description |
|-------|------------------------------------|-------|-----------------------------------|
| COMP | Compression module | CTRL | FCPV controller |
| WIIF | VSP write channel interface | RIIF | VSP read channel interface |
| WSIF | System AXI write channel interface | RSIF | System AXI read channel interface |

35.3.3 IP Set and Bus Access Functions

FCPV does not work alone but in companion with other IPs (VSP {I/B/D}) in an IP set. For each IP set the bus access function conducted by FCPV is different.

Table 35.8 Bus access function for each IP set

| IP set | Bus access |
|--------------|--|
| VSPB + FCPVB | Following functions are conducted by FCPV |
| VSPD + FCPVD | <ul style="list-style-type: none"> Read linear addressing image data |
| VSPI + FCPVI | <ul style="list-style-type: none"> Read display list data Write image data Write histogram data * |

Note: * This function is active when FCPV is connected to VSPI of RZ/G2H, RZ/G2N, RZ/G2E; VSPBC of RZ/G2H and VSPB of RZ/G2N, RZ/G2E.

35.3.4 Register Configuration

The FCPV contains the registers which can directly be read and written to by the host CPU. The host can access any long word (32 bits) in the registers. Therefore, read-modify-write is needed to change partial bits in 32bits register.

The grayed portions represent reserved areas for debugging or future functional expansion. Do not access the reserved areas. Never access the registers when the FCPV is in the module standby mode (with the operation clock placed in hold status).

The following are the base addresses of the FCPV channels in case of RZ/G2H.

- FCPVI0: H'FE9A_F000
- FCPVI1: H' FE9B_F000
- FCPVB0 (for VSPBD): H' FE96_F000
- FCPVB1 (for VSPBC): H' FE92_F000
- FCPVD0: H'FEA2_7000
- FCPVD1: H'FEA2_F000

The following are the base addresses of the FCPV channels in case of RZ/G2M V1.3, RZ/G2M V3.0.

- FCPVI0: H'FE9A_F000
- FCPVB0: H' FE96_F000
- FCPVD0: H'FEA2_7000
- FCPVD1: H'FEA2_F000
- FCPVD2: H'FEA3_7000

The following are the base addresses of the FCPV channels in case of RZ/G2N, RZ/G2E.

- FCPVI0: H'FE9A_F000
- FCPVB0: H' FE96_F000
- FCPVD0: H'FEA2_7000
- FCPVD1: H'FEA2_F000

Table 35.9 List of FCPV Register

| Name of Register | Abbreviation | R/W | Offset from Base Address | Initial Value | Access size | Second Generation RZ/G Series Products | | | |
|-------------------------------|--------------|-----|--------------------------|---------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| FCP version control register | FCP_VCR | R | H'0000 | VCR_INIT * | 32 | √ | √ | √ | √ |
| FCPV configuration register 0 | FCP_CFG0 | R/W | H'0004 | H'0000_0000 | 32 | √ | √ | √ | √ |
| FCP reset register | FCP_RST | R/W | H'0010 | H'0000_0000 | 32 | √ | √ | √ | √ |
| | | | H'0014 | | | √ | √ | √ | √ |
| FCP status register | FCP_STA | R | H'0018 | H'0000_0000 | 32 | √ | √ | √ | √ |

Note: * VCR_INIT: depend on product generation, refer to section 35.3.6.1 for detail

35.3.5 Connected Module

Following table shows modules connected to the FCPV.

Table 35.10 Connected Module

| Module | Connected Module | Description |
|--------|---|--|
| FCPVI | VSP1 | VSP for image processing |
| FCPVB | VSPBC (RZ/G2H only) | VSP for image blending |
| | VSPBD (RZ/G2H only) | VSP for image blending |
| | VSPB (RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E) | VSP for image blending |
| FCPVD | VSPD | VSP for blending and display output |
| | VSPDL (RZ/G2H, RZ/G2N) | VSP for blending and 2 display outputs |

35.3.6 Register Description

35.3.6.1 FCP_VCR: FCP Version Control Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

FCP_VCR is the version control register of the FCPV. The value read is fixed.

| | | | | | | | | | | | | | | | | |
|----------------|----------|----|----|----|----|----|----|----|----------|----|----|----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CATEGORY | | | | | | | | REVISION | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0/1 | 0/1 | 0/1 | 0/1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|------------------------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 8 | CATEGORY | H'01 | R | FCP use case category. 1: RZ/G |
| 7 to 0 | REVISION | H'04 or H'02 or H'06 or H'08 | R | Version of LSI product H'04: FCP revision is RZ/G2H H'02: FCP revision is RZ/G2M V1.3, RZ/G2M V3.0 H'06: FCP revision is RZ/G2N H'08: FCP revision is RZ/G2E |

35.3.6.2 FCP_CFG0: FCPV Configuration Register 0

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

FCP_CFG0 specifies the configuration of the FCPV.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | FCPV SEL | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | FCPVSEL | B'0 | R/W | FCPV Mode Select This bit specifies the FCPV processing mode. 0: FCPVB, FCPVD 1: FCPVI This bit must not be changed during operation of the FCPV. |
| 0 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

35.3.6.3 FCP_RST: FCP Reset Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

FCP_RST controls the FCPV reset operation. The reset operation of the FCPV highly depends on the reset sequence of the connected VSP. For detailed information, refer to the section 35.3.7.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|--------|----|----|----|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | MODRST | — | — | — | SOFT RST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | MODRST | B'0 | R/W | FCPV MODule ReSeT This reset stops the FCPV module immediately regardless of the bus transaction state. To operate the FCPV after this reset, reconfigure all FCPV registers. After the active period of MODRST is finished, it is switched back to 0 automatically. 0: NOP 1: Forcibly terminate the FCPV operation. Note: Do not set both MODRST and SOFTRST bit to 1. |
| 3 to 1 | — | B'000 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | SOFTRST | B'0 | R/W | FCPV SOFT ReSeT This reset stops the FCPV module after finishing active bus transaction. To operate the FCPV after this reset, reconfigure all FCPV registers. After the active period of SOFTRST is finished, it is switched back to 0 automatically. 0: NOP 1: Terminates the FCPV operation. Note: Do not set both MODRST and SOFTRST bit to 1. |

35.3.6.4 FCP_STA: FCP Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

FCP_STA indicates the FCPV status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ACT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0 or 1. Do not care about this value. The write value should always be 0. |
| 0 | ACT | B'0 | R | FCPV Active Flag 0: Indicates the FCPV is not active. 1: Indicates the FCPV is active. |

35.3.7 Operation

FCPV is not invoked directly by register setting. It works if there are bus accesses from the connected IP VSP. This section explains how to control operation for FCPV connected with VSP.

35.3.7.1 Operation Control Setting IP set VSPI+FCPVI

(a) FCPVI initialization

FCPVI must be initialized with following settings. This initialization must be conducted before starting image process. After initialization, those settings are kept no change until power-off or hardware reset, therefore, they don't need to be set for every image process.

- Set 1 to FCP_CFG0.FCPVSEL to enable configuration of FCPVI.

(b) To start image process of VSPI + FCPVI

FCPVI registers do not need to be set when starting an image process. Therefore, refer to section "33. Video Signal Processor (VSP2)" for the guidance of starting an image process.

The register setting of VSPI should follow restrictions in section 35.3.7.5.

Set value 1 to VI6_WPF0_OUTFMT.FCNL if writing FCNL data.

(c) To stop image process of VSPI+FCPVI immediately

Do following steps to stop immediately an image process of VSPI + FCPVI

1. Set value 1 to VSPI register VI6_WPF0_IRQ_ENB.FREE so that frame end interruption can be issued.
2. Set value 1 to VSPI register VI6_SRESET.SRST0. VSPI will invoke termination process immediately.
3. Wait frame end interruption from VSPI or until the register bit VI6_WPF0_IRQ_STA.FRE is set to 1. After it occurs, VSPI has finished its processing.
4. Set value 1 to FCPVI register FCP_RST.SOFTRST. FCPVI will invoke termination process immediately.
5. Wait until ACT bit of the FCP_STA register and become 0. After it occurs, FCPVI has finished its processing.

With this procedure, VSPI can stop its process quickly, but the output frame of the last frame is corrupted.

35.3.7.2 Operation Control Setting for IP set VSPB+FCPVB

(a) FCPVB initialization

FCPVB must be initialized with following settings. This initialization must be conducted before starting image process. After initialization, those settings are kept no change until power-off or hardware reset, therefore, they don't need to be set for every image process.

- Set 0 to FCP_CFG0.FCPVSEL to enable configuration of FCPVB.

(b) To start image process of VSPB+FCPVB

FCPVB registers do not need to be set when starting an image process. Therefore, refer to the section "33. Video Signal Processor (VSP2)" for the guidance of starting an image process.

The register setting of VSPB should follow restrictions in section 35.3.7.5.

Set value 1 to VI6_WPF0_OUTFMT.FCNL if writing FCNL data.

(c) To stop image process of VSPB+FCPVB immediately

Do following steps to stop immediately an image process

1. Set value 1 to VSPB register VI6_WPF0_IRQ_ENB.FREE so that frame end interruption can be issued.
2. Set value 1 to VSPB register VI6_SRESET.SRST0. VSPB will invoke termination process immediately.
3. Wait frame end interruption from VSPB or until the register bit VI6_WPF0_IRQ_STA.FRE is set to 1. After it occurs, VSPB has finished its processing.
4. Set value 1 to FCPVB register FCP_RST.SOFTRST. FCPVB will invoke termination process immediately.
5. Wait until ACT bit of the FCP_STA register become 0. After it occurs, FCPVB has finished its processing.

With this procedure, VSP can stop its process quickly, but the output frame of the last frame is corrupted.

35.3.7.3 Operation Control Setting for IP set VSPD+FCPVD, VSPDL+FCPVD

(a) FCPVD initialization

FCPVD must be initialized with following settings. This initialization must be conducted before starting image process. After initialization, those settings are kept no change until power-off or hardware reset, therefore, they don't need to be set for every image process.

- Set 0 to FCP_CFG0.FCPVSEL to enable configuration of FCPVD.

(b) To start image process of VSPD+FCPVD, VSPDL+FCPVD

FCPVD registers do not need to be set when starting an image process. Therefore, refer to the section "33. Video Signal Processor (VSP2)" for the guidance of starting an image process.

The register setting of VSPD should follow restrictions in section 35.3.7.5.

(c) To stop image process of VSPD+FCPVD, VSPDL+FCPVD immediately

Do following steps to stop immediately an image process

1. Stop the DU first if it is connected with VSPD (VI6_LIF{0/1}_CTRL.LIF_EN = 1). Refer to the section "36. Display Unit (DU)" for the way of stopping the DU. Not do this step if DU is not connected with VSPD (VI6_LIF{0/1}_CTRL.LIF_EN = 0).
2. Set value 1 to VSPD register VI6_WPF {0/1} _IRQ_ENB.FREE so that frame end interruption can be issued.
3. Set value 1 to VSPD register VI6_SRESET.SRST {0/1}. VSPD will invoke termination process immediately.
4. Wait frame end interruption from VSPD or until the register bit VI6_WPF {0/1} _IRQ_STA.FRE is set to 1. After it occurs, VSPD has finished its processing.
5. Set value 1 to FCPVD register FCP_RST.SOFTRST. FCPVD will invoke termination process immediately. *
6. Wait until ACT bit of the FCP_STA register become 0. After it occurs, FCPVD has finished its processing. *

Note: * In case of VSPDL+FCPVD, not apply step 5, step 6.

With this procedure, VSP can stop its process quickly, but the output frame of the last frame is corrupted.

35.3.7.4 Reset Operation

FCPV reset can be conducted by software reset and power on reset. The software reset which is controlled by software, is used to stop the FCPV operation. The software reset operation is not executed immediately; it is executed on completion of the bus transaction. The other is power on reset which initialize the whole FCPV logic immediately.

(1) Software Reset

- 1) Activate software reset sequence
Set FCP_RST.SOFTRST to the value 1.
- 2) Confirm software reset sequence finished:
FCP_STA.ACT becomes 0. All remain transactions with the bus system finished.

The procedure to stop operation of FCPV and connected IP is described at section 35.3.7.1, 35.3.7.2.

(2) Power-on Reset, Hardware reset

The power-on reset is controlled by an externally input signal called hardware reset. This reset initializes the whole internal logic of FCPV.

35.3.7.5 Register Settings and Restrictions on Connected VSP

The connected IP VSP {I/B} have to follow the setting and restrictions in following tables.

(1) Register settings on VSP{I/B}

Table 35.11 Register setting when writing FCNL data

| Register | Register value | Use cases |
|----------------------|----------------|---------------------|
| VI6_WPF0_OUTFMT.FCNL | 0 | Write non FCNL data |
| | 1 | Write FCNL data |

(2) Restrictions on VSP {I/B} in case of writing data via FCPV

In case of writing FCNL data (VI6_WPF0_OUTFMT.FCNL = 1), the write data swap register of VSP must be set as shown in Table 35.12.

Table 35.12 Setting restriction about data swap

| Register | Register Value | Target IP |
|---|----------------|------------|
| VI6_WPF0_DSWAP.{P_LLS, P_LWS, P_WDS, P_BTS} | H'8 | VSP{I/B} * |

Table 35.13 Setting restriction about output color formats

| Register | Format setting (VI6_WPF0_OUTFMT) | | Restrictions | |
|--------------------------------|-------------------------------------|----------|--|--------------|
| | FCNL | ROT[2:0] | Register Value (explanation) | Target IP |
| VI6_WPF0_ROT_CTRL.LN 16 | Don't care | 4-7 | 1 (always output in 16 lines mode) | VSPI |
| VI6_WPF0_OUTFMT. WRFMT[6:0] | 1 | 0,1 | H'13 (ARGB8888) H'47 (YCbCr 422 Interleave YUY2) (*1) H'4A (YCbCr 444 Planar) H'4B (YCbCr 422 Planar) H'4C (YCbCr 420 Planar) | VSP{I/B}{*2} |
| | | 2-3 | H'13 (ARGB8888) H'47 (YCbCr 422 Interleave YUY2) (*1) H'4A (YCbCr 444 Planar) H'4B (YCbCr 422 Planar) H'4C (YCbCr 420 Planar) | VSPI |
| | | 4-7 | H'13 (ARGB8888) | VSPI |

Notes: 1 For the format YCbCr422 Interleave YUY2, VI6_WPF0_OUTFMT.SPYCS must be set to 1 and VI6_WPF0_OUTFMT.SPUVS must be set to 0.

Table 35.14 Setting restriction about destination stride and address for FCNL data, VSPB (*1)

| Register | Restriction value |
|---|-------------------|
| VI6_WPF0_STRIDE_Y.PICT_STRD_Y[15:0] | 256n (*2) |
| VI6_WPF0_STRIDE_C.PICT_STRD_C[15:0] | 256n |
| VI6_WPF0_DSTM_ADDR_Y.DSTM_ADDR_Y[31:0] | 256n |
| VI6_WPF0_DSTM_ADDR_m.DSTM_ADDR_m[31:0] (m = C0, C1) | 256n |

Notes: 1. VI6_WPF0_OUTFMT.FCNL = 1
2. 256n means multiple of 256

Table 35.15 Setting restriction about destination stride and address for FCNL data, VSPI (*1)(*5)

| Register | Format setting (VI6_WPF0_OUTFMT) | | Restrictions | |
|--|-------------------------------------|---------------------|---|--|
| | ROT[2:0] | WRFMT[6:0] | Value for whole frame or first partition (*4) | Value for second or later partition |
| VI6_WPF0_STRIDE_Y. PICT_STRD_Y[15:0] | — | — | 256n (*3) | 256n |
| VI6_WPF0_STRIDE_C. PICT_STRD_C[15:0] | — | — | 256n | 256n |
| VI6_WPF0_DSTM_ADDR_ Y. DSTM_ADDR_Y[31:0] | 0, 1, 2, 3 | H'47(*2), H'13 | 256n | 64n |
| | | H'4A, H'4B, H'4C | 256n | 16n |
| | 4, 5 | H'13 | 256n + 192 | 64n |
| | 6, 7 | H'13 | 256n | 64n |
| VI6_WPF0_DSTM_ADDR_ m. DSTM_ADDR_m[31:0] (m = C0, C1) | 0, 1, 2, 3 | H'4A, H'4B, H'4C | 256n | 16n |

Notes: 1. VI6_WPF0_OUTFMT.FCNL = 1

2. For the format YCbCr422 Interleave YUY2, VI6_WPF0_OUTFMT.SPYCS must be set to 1 and VI6_WPF0_OUTFMT.SPUVS must be set to 0.

3. 256n/64n/16n means multiple of 256/64/16

4. first partition means the left edge partition in case VI6_WPF0_OUTFMT.ROT[2:0] != 4, 5 and right edge partition in case VI6_WPF0_OUTFMT.ROT[2:0] = 4, 5, as shown in Figure 35.2.

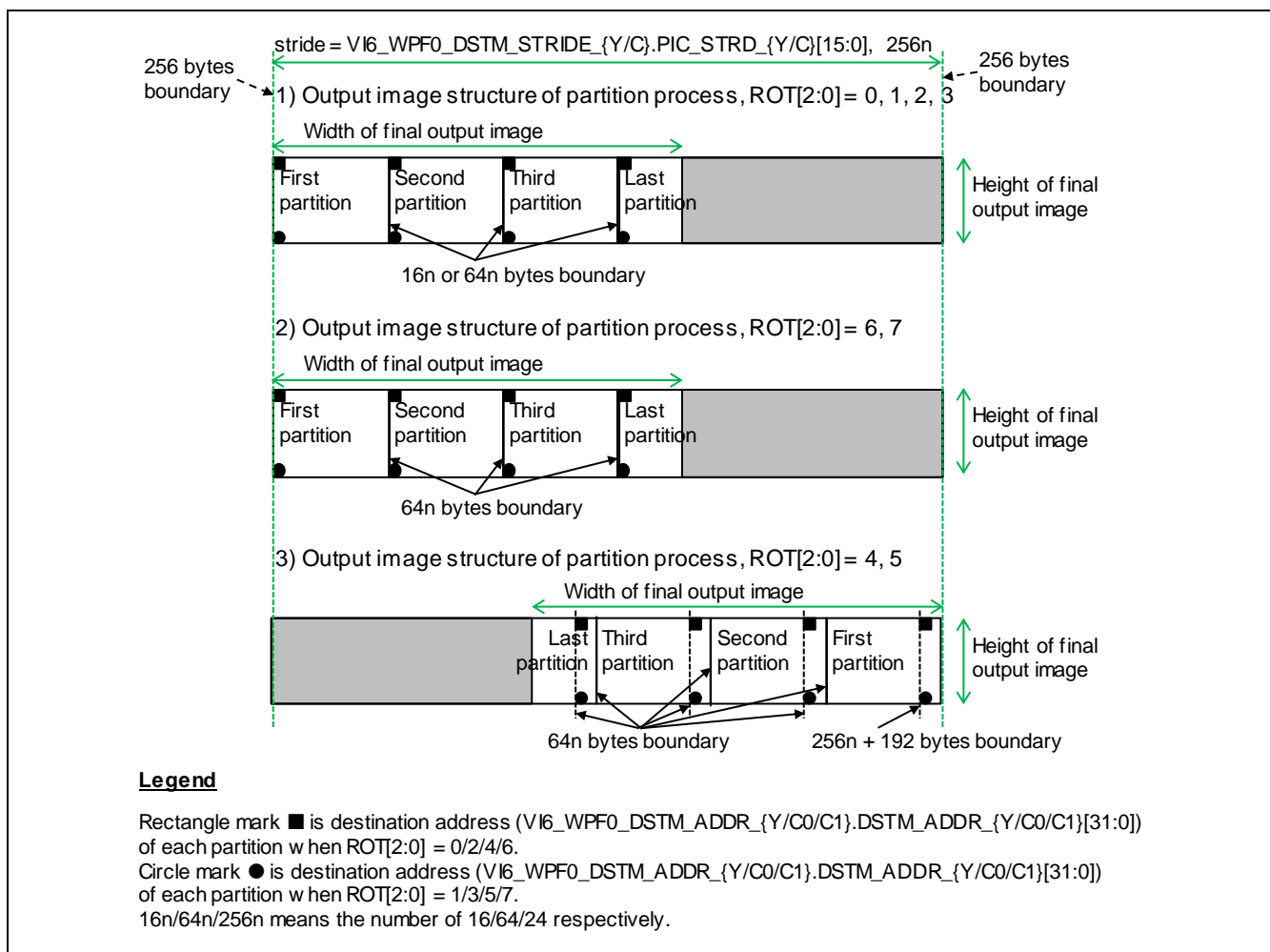


Figure 35.2 Restriction of write stride, address in case of partition process of VSPI

(3) Restrictions on VSP {I/B/D} in case of reading data via FCPV

In case of reading FCNL data, the read data swap register of VSP must be set as shown in Table 35.16. Note that this restriction is needed for only RZ/G2H, RZ/G2M V1.3, RZ/G2N, RZ/G2E which supports FCNL.

Table 35.2 Setting restriction on data swap in case of reading FCNL data

| Register | Value | Target IP |
|--|-------|------------|
| VI6_RPFm_DSWAP.{P_LLS, P_LWS, P_WDS, P_BTS} (m = 0-4) in case of reading FCNL data. | H'F | VSP{I/B/D} |

35.3.8 Memory Layout of FCNL Image Data

35.3.8.1 In Cases of VI6_WPF0_OUTFMT.ROT [2:0] = 0, 1, 2, 3, 6, 7

Figure 35.3 shows the memory layout of a FCNL image in cases of VI6_WPF0_OUTFMT.ROT [2:0] = 0, 1, 2, 3, 6, 7 for a final output image (distinct with output image of one partition in VSPI case). The output memory area of FCNL image data is specified as a rectangle linear address area with a destination address and a stride, as same as the raw data. However, the FCNL image data is stored in shadow address area of the rectangle linear address area. That is, in each 256 bytes area starting from 256 bytes boundary, FCNL data is stored on 128 bytes area of the lower side address.

The register setting of destination address and memory stride must follow the section “33. Video Signal Processor (VSP2)” and satisfy the restriction on section 35.3.7.5.

The buffer size for a FCNL image is equal to *stride* * *height*. Here, height is half of the picture height if the plane is Cb (U), Cr (V) for format YCbCr 420 planar and equal to the picture height for other cases. The buffer size doesn't change regardless of the FCNL compression.

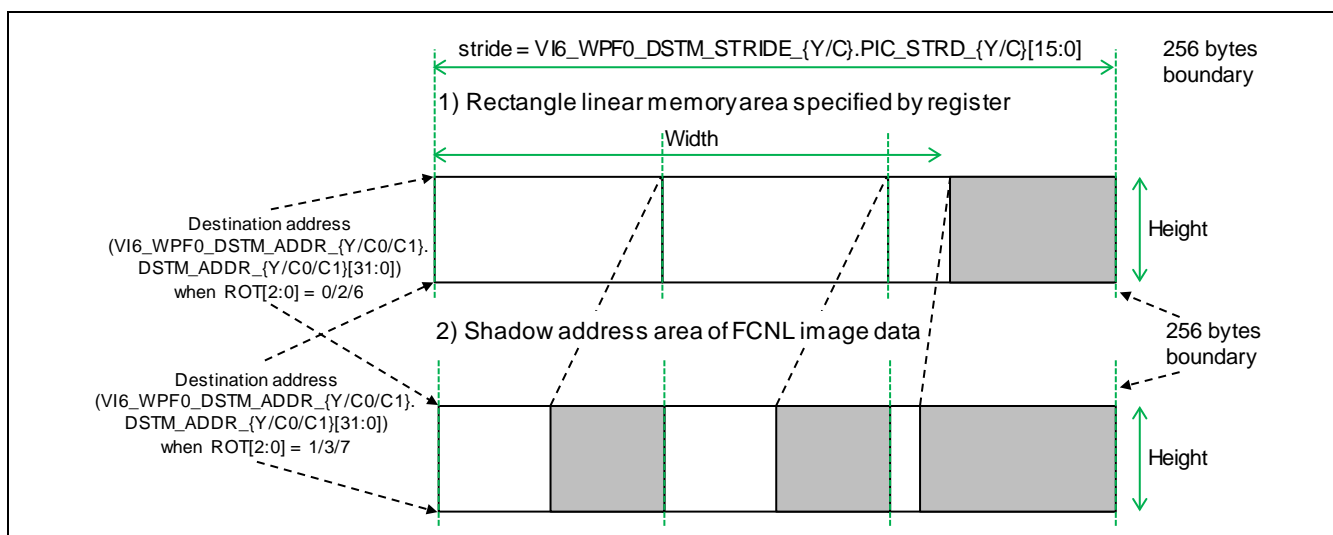


Figure 35.3 Memory layout of the FCNL image (ROT [2:0] = 0, 1, 2, 3, 6, 7)

When reading FCNL data, the source address, source stride, source image sizes must be specified basing on the rectangle linear address area. The source address must be inside the rectangle linear address area.

35.3.8.2 In Cases of VI6_WPF0_OUTFMT.ROT [2:0] = 4, 5

Figure 35.4 shows the memory layout of a FCNL image in cases of VI6_WPF0_OUTFMT.ROT [2:0] = 4, 5, for a final output image (distinct with output image of one of multiple partitions in VSPI case).

Same to the case 35.3.8.1, FCNL data is stored in the shadow address area of the rectangle linear address area specified by destination and stride registers.

As shown in the Figure 35.4, for the cases of VI6_WPF0_OUTFMT.ROT [2:0] = 4, 5, the write access is conducted from right to left of the memory area. Therefore, the destination address is specified at the right side of the rectangle linear address area.

The register setting of destination address and memory stride must follow the section “33. Video Signal Processor (VSP2)” and satisfy the restriction on 35.3.7.5.

The buffer size for a FCNL image is equal to $stride * height$. Here, height is half of the picture height if the plane is Cb (U), Cr (V) for format YCbCr 420 planar and equal to the picture height for other cases. The buffer size doesn’t change regardless of the FCNL compression.

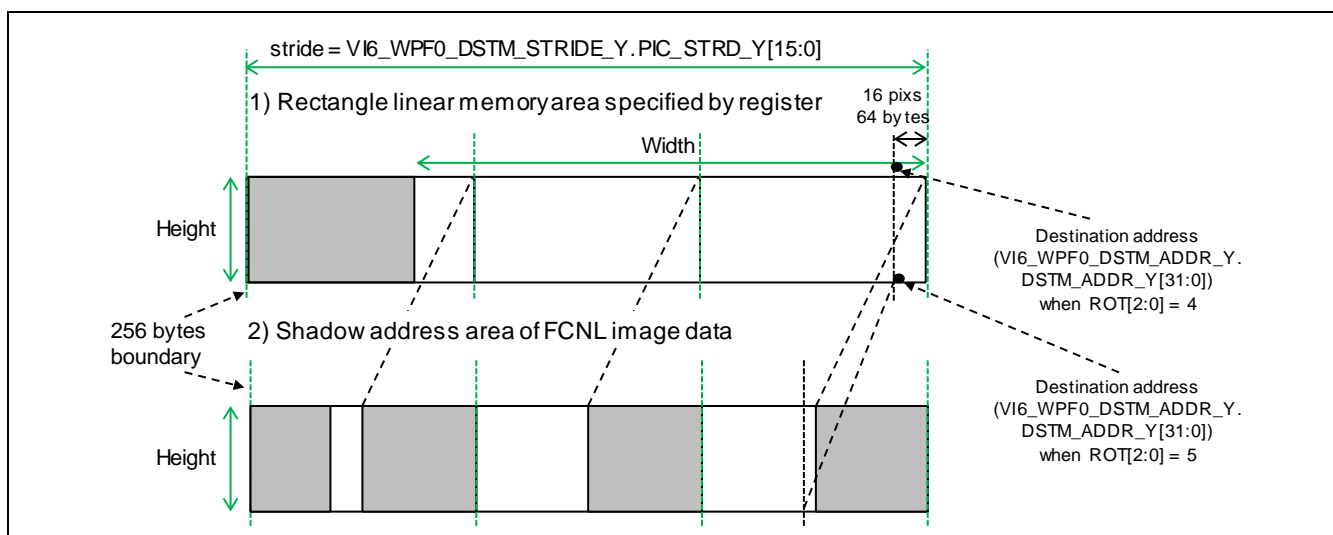


Figure 35.4 Memory layout of the FCNL image (ROT [2:0] = 4, 5)

When reading FCNL data, the source address, source stride, source image sizes must be specified basing on the rectangle linear address area. The source address must be inside the rectangle linear address area. The read access is conducted on left to right of the memory area.

35.3.9 Usage Notes

35.3.9.1 Manipulating FCPV+VSP with CPG module stop, software reset

The following notes 1) to 2) are to prevent FCPV, VSP from the unstable status which may cause hang or wrong operation.

Note 1) In the case of stopping FCPV{I/B/D} and VSP{I/B/D} using the corresponding register bit of module stop register (MSTPSR6), realtime module stop register (RMSTPCR6), system module stop register (SMSTPCR6) and software reset register (SRCR6) of the module CPG, do following steps. Note that it is needed to stop both FCPV{I/B/D} and VSP{I/B/D}.

Step1) Stop FCPV{I/B/D} + VSP{I/B/D} with the software reset sequence following to the guidance in the section 35.3.7. For VSPDL+FCPVD, it is needed to stop both two video paths corresponding to WPF0, WPF1 if both WPFs are operating.

Step2) Stop VSP{I/B/D}, FCPV{I/B/D} using the corresponding register bit of CPG register (MSTPSR6 or RMSTPCR6 or SMSTPCR6 or SRCR6). Either VSP or FCPV can be stopped first.

Note 2) In the case of re-starting FCPV{I/B/D} + VSP{I/B/D} after stopping those modules with CPG module stop, CPG software reset, be sure to run start process as defined in section “33. Video Signal Processor (VSP2)”.

35.4 FCP for FDP (FCPF)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The FCPF is a frame compression module which reduces the bandwidth with Renesas near-lossless (FCNL) compression and lossless (FCL) decompression and conducts the bus accesses with 256 bits AXI bus for the Renesas fine display processor (FDP). This module is under control of software for the FDP with the control function of the FCPF executed on the host CPU.

35.4.1 Features

The FCPF has the following features.

- Support FCNL near-lossless compression with 50% bandwidth reduction
— Note: FCNL decompression is executed by DBSC4, DDR3L/4 memory controller.
- Support for FCL lossless decompression for YCbCr420 semi-planar
- Support 8190 x 8190 resolution
- Support converting linear addressing access from FDP to tile addressing image
- Support out-of-order for the whole outstanding transactions

Table 35.17 Main Function of the FCPF

| Item | Description |
|------------------------------------|---|
| Data compression and decompression | Renesas lossless (FCL) decompression for YCbCr420 semi-planar Renesas near-lossless (FCNL) compression for formats ARGB8888, YCbCr 444/422/420 Planar, YCbCr 422 Interleave YUY2 |
| Picture size supported | Tile addressing image: Min: 32 pixels x 16 lines Max: Horizontal 8190 pixels x vertical 8190 lines Others: support all picture sizes specified in the section “34. Fine Display Processor (FDP1)” (sections “Overview”, “Features”) under additional restrictions at 35.4.7.3. |
| Color format | Tile addressing image: YCbCr420 semi-planar FCNL compression: ARGB8888, YCbCr Planar 444/422/420, YCbCr Interleave 422 YUY2 Others: support all formats specified in the section “34. Fine Display Processor (FDP1)” (sections “Register Descriptions”, “Source Picture Format Register (FD1_RPF_FORMAT)” and “Destination Picture Format Register (FD1_WPF_FORMAT)”) |
| AXI transaction | Read access: Support out-of-order for all outstanding transactions Support data interleaving Tile/linear address conversion (format YCbCr420 semi-planar only) Write access: Support out-of-order for all outstanding transactions |

35.4.2 Block Diagram

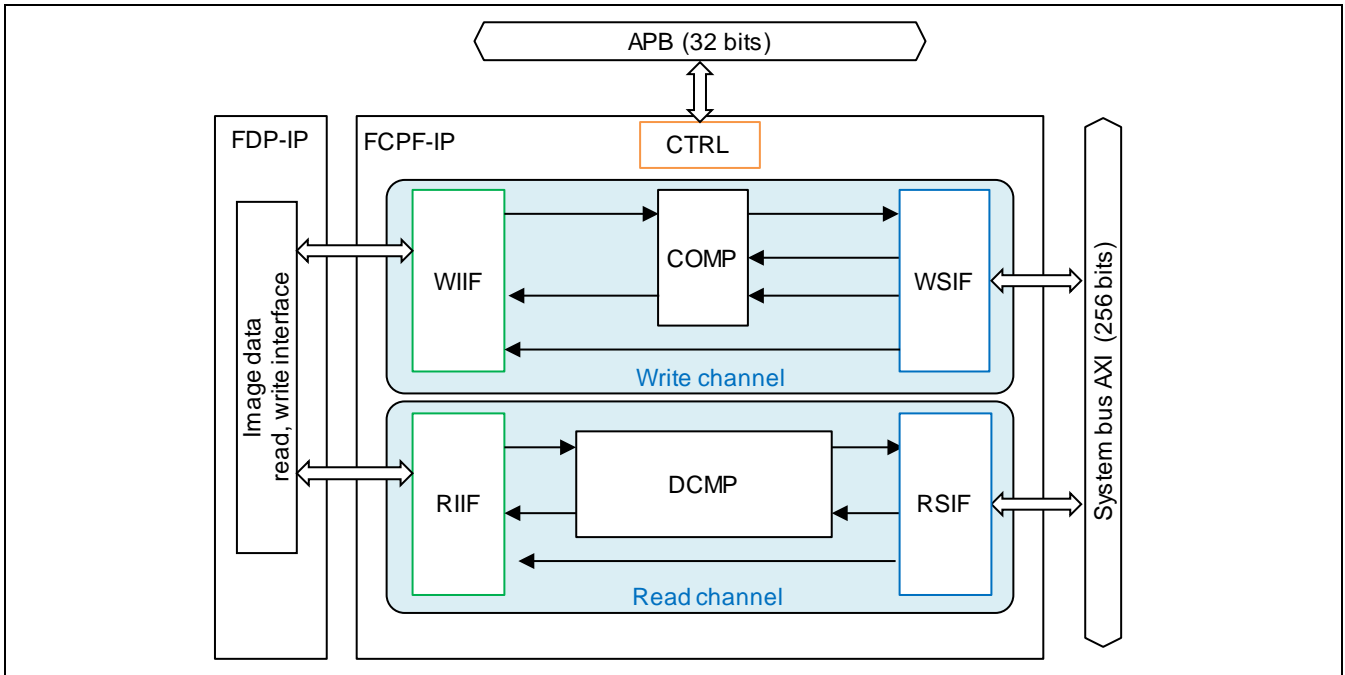


Figure 35.5 Overall Configuration of the FCPF

Table 35.18 Sub-module Description

| Abbr. | Description | Abbr. | Description |
|-------|--------------------------------------|-------|-------------------------------------|
| COMP | Compression module | DCMP | Decompression module |
| WIIF | Connected IP write channel interface | RIIF | Connected IP read channel interface |
| WSIF | System AXI write channel interface | RSIF | System AXI read channel interface |
| CTRL | FCPF controller | — | — |

35.4.3 IP set and bus access functions

FCPF works in companion with other IPs (FDP) in an IP set. As shown in Table 35.19 the bus access function conducted by FCPF is different according to each IP set.

Table 35.19 Bus access function for each IP set

| IP set | Bus access |
|------------|--|
| FDP + FCPF | Following functions are conducted by FCPF <ul style="list-style-type: none"> • Read linear addressing image data • Read tile addressing image data • Write image data |

35.4.4 Register Configuration

The FCPF contains the registers which can directly be read and written to by the host CPU. The host can access any long word (32 bits) in the registers. Therefore, read-modify-write is needed to change partial bits in 32bits register.

There are 2 types of registers in FCPF: immediate register, V-update Register. The values of immediate registers are reflected to H/W behavior immediately after changing its values. Hence, do not change these immediate registers while FCPF is processing (FCP_STA.ACT is set to 1) unless this manual allows it explicitly. The values of V-update registers are reflected to H/W behavior at the next V-interruption of FDP. For FDP connection case users can set V-update registers of the next frame/field while FDP+FCPF is processing the current frame/field.

Refer to the sections 35.4.7.1 to grasp the flow of setting registers of FCPF and connected IP.

There are many settings such as byte swaps, formats, etc which are set in connected IP and transferred to FCPF. Those registers must be set following to the section 35.4.7.3 of this document.

The grayed portions represent reserved areas for debugging or future functional expansion. Do not access the reserved areas. Never access the registers when the FCPF is in the module standby mode (with the operation clock placed in hold status).

The following are the base addresses of the FCPF channels.

- FCPF0: H'FE95_0000
- FCPF1: H'FE95_1000 (RZ/G2H only)

Table 35.20 FCPF Registers

| Name of Register | Abbreviation | Register type | R/W | Offset from base address | Initial Value | Access size | Second Generation RZ/G Series Products | | | | |
|---|---------------|---------------|-----|--------------------------|---------------|-------------|--|-------------|-------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| FCP version control register | FCP_VCR | Imm | R | H'0000 | VCR_INIT (*) | 32 | √ | √ | √ | √ | |
| FCP reset register | FCP_RST | Imm | R/W | H'0010 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| FCP status register | FCP_STA | Imm | R | H'0018 | VCR_INIT (*) | 32 | √ | √ | √ | √ | |
| FCP Tile/Linear address conversion control | FCP_TL_CTRL | Vupdt | R/W | H'0070 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| FCP picture information register 1 | FCP_PICINFO1 | Vupdt | R/W | H'00C4 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| FCP base address of ancillary information of plane Y0 | FCP_BA_ANC_Y0 | Vupdt | R/W | H'0100 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| FCP base address of ancillary information of plane Y1 | FCP_BA_ANC_Y1 | Vupdt | R/W | H'0104 | H'0000_0000 | 32 | √ | √ | √ | √ | |

| Name of Register | Abbreviation | Register type | R/W | Offset from base address | Initial Value | Access size | Second Generation RZ/G Series Products | | | |
|---|---------------|---------------|-----|--------------------------|---------------|-------------|--|----------------------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| FCP base address of ancillary information of plane Y2 | FCP_BA_ANC_Y2 | Vupdt | R/W | H'0108 | H'0000_0000 | 32 | √ | √ | √ | √ |
| FCP base address of ancillary information of plane C | FCP_BA_ANC_C | Vupdt | R/W | H'010C | H'0000_0000 | 32 | √ | √ | √ | √ |
| FCP base address of reference picture of plane Y0 | FCP_BA_REF_Y0 | Vupdt | R/W | H'0110 | H'0000_0000 | 32 | √ | √ | √ | √ |
| FCP base address of reference picture of plane Y1 | FCP_BA_REF_Y1 | Vupdt | R/W | H'0114 | H'0000_0000 | 32 | √ | √ | √ | √ |
| FCP base address of reference picture of plane Y2 | FCP_BA_REF_Y2 | Vupdt | R/W | H'0118 | H'0000_0000 | 32 | √ | √ | √ | √ |
| FCP base address of reference picture of plane C | FCP_BA_REF_C | Vupdt | R/W | H'011C | H'0000_0000 | 32 | √ | √ | √ | √ |

Imm: immediate register

Vupdt: V-update register

Note: * VCR_INIT: depend on product generation, refer to section 35.4.6.1 for detail

35.4.5 Connected Module

Table 35.21 shows modules related to the FCPF.

Table 35.3 Connected Module

| Module | Related Module | Description |
|--------|----------------|---------------|
| FCPF | FDP | I/P convertor |

35.4.6 Register Descriptions

35.4.6.1 FCP_VCR: FCP Version Control Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

FCP_VCR is a version control register of the FCPF. The value read is fixed.

| | | | | | | | | | | | | | | | | |
|----------------|----------|----|----|----|----|----|----|----|----------|----|----|----|-----|-----|-----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CATEGORY | | | | | | | | REVISION | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0/1 | 0/1 | 0/1 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|------------------------------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 8 | CATEGORY | H'01 | R | FCP use case category. 1: RZ/G |
| 7 to 0 | REVISION | H'04 or H'02 or H'06 or H'08 | R | Version of LSI product H'04: FCP revision is RZ/G2H H'02: FCP revision is RZ/G2M V1.3, RZ/G2M V3.0 H'06: FCP revision is RZ/G2N H'08: FCP revision is RZ/G2E |

35.4.6.2 FCP_RST: FCP Reset Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

FCP_RST controls the FCPF reset operation. The reset operation of the FCPF highly depends on the reset sequence of FDP. For detailed information, refer to the section 35.4.7.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|-------------|-------------|-------------|----|----|----|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | RIIFRS T | RSIFR ST | DCMP RST | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SOFT RST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 23 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 22 | RIIFRST | B'0 | R/W | RIIF ReSeT 0: NOP 1: Initialize the module RIIF. After the active period of RIIFRST is finished, it is switched back to 0 automatically. |
| 21 | RSIFRST | B'0 | R/W | RSIF ReSeT 0: NOP 1: Initialize the module RSIF. After the active period of RSIFRST is finished, it is switched back to 0 automatically. |
| 20 | DCMPRST | B'0 | R/W | DCMP ReSeT 0: NOP 1: Initialize the module DCMP. After the active period of DCMPRST is finished, it is switched back to 0 automatically. |
| 19 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | SOFRST | B'0 | R/W | FCPF SOFT ReSeT This reset stops the FCPF module after finishing active bus transaction. To operate the FCPF after this reset, reconfigure all FCPF registers. After the active period of SOFRST is finished, it is switched back to 0 automatically. 0: NOP 1: Terminates the FCPF operation. |

35.4.6.3 FCP_STA: FCP Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

FCP_STA indicates the FCPF status.

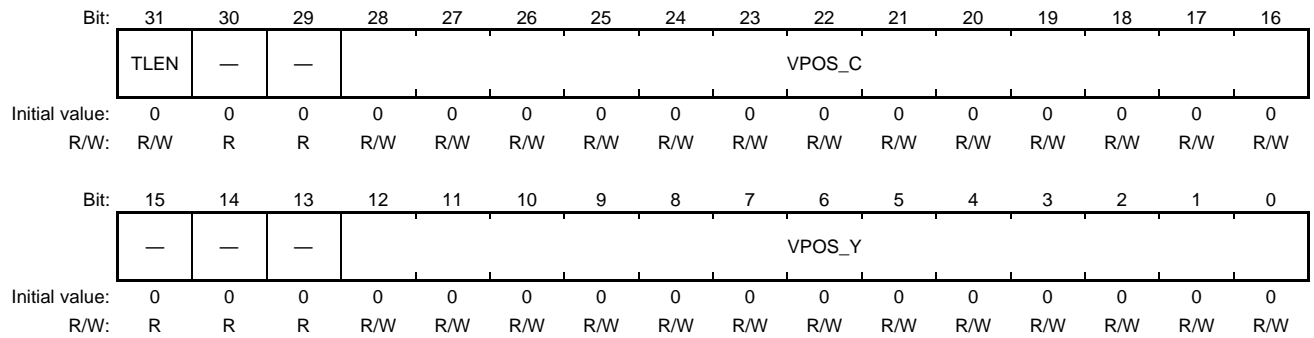
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ACT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0 or 1. Do not care about this value. The write value should always be 0. |
| 0 | ACT | B'0 | R | FCPF Active Flag 0: Indicates the FCPF is not active. 1: Indicates the FCPF is active. |

35.4.6.4 FCP_TL_CTRL: FCP Tile/Linear Control Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

FCP_TL_CTRL specifies the control of Tile/Linear address conversion on FCPF. Refer to section 35.4.7.4(3) to grasp how to set registers when reading a tile addressing image.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | TLEN | B'0 | R/W | Tile Linear Enable 0: Disable Tile/Linear address conversion 1: Enable Tile/Linear address conversion |
| 30 to 29 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 16 | VPOS_C | All 0 | R/W | Vertical POSition of Chroma Vertical position of tile address chroma plane which FDP start to read. This register must be set value from 0 to H'FFE. |
| 15 to 13 | — | B'000 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 0 | VPOS_Y | All 0 | R/W | Vertical POSition of Y plane Vertical position of tile address luma plane which FDP start to read. This register must be set value from 0 to H'1FFD. |

35.4.6.5 FCP_PICINFO1: FCP Picture Information 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

FCP_PICINFO1 specifies the picture information of the tile address image. Refer to section 35.4.7.4(3) to grasp how to set registers when reading a tile addressing image.

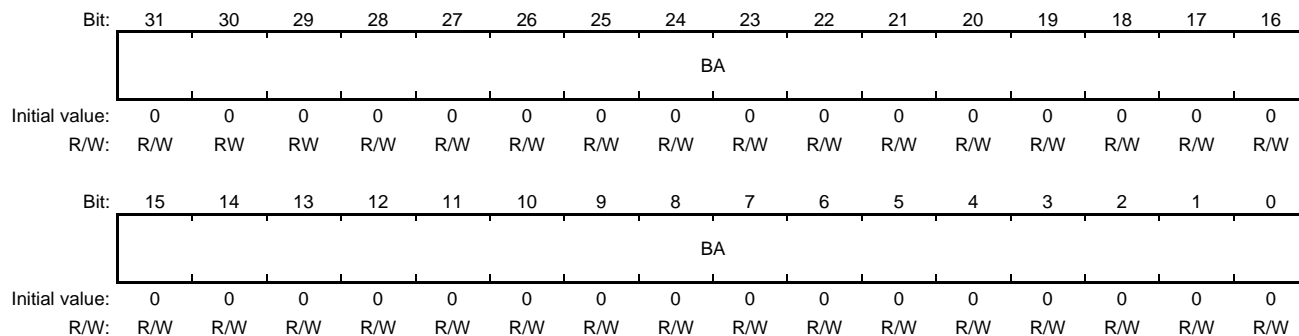
| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|--------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | STRIDE_DIV16 | | | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 11 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 to 0 | STRIDE_DIV16 | All 0 | R/W | STRIDE DIVided by 16 Memory stride of the tile addressing image. Common for all luma and chroma planes Y0, Y1, Y2 and C. It must be specified so that STRIDE_DIV16*16 is exponent of 2 and larger than or equal to 128. STRIDE_DIV16*16 must be the same with memory stride of Y plane, C plane set by FDP registers. This bit must get value from H'8 to H'200. |

35.4.6.6 FCP_BA_ANC_m {m: Y0, Y1, Y2, C}: FCP Base Address Ancillary information of plane m

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

FCP_BA_ANC_m specifies the base address of the ancillary information of the plane Y0, Y1, Y2, C. The ancillary information is explained at the section 35.4.7.4.

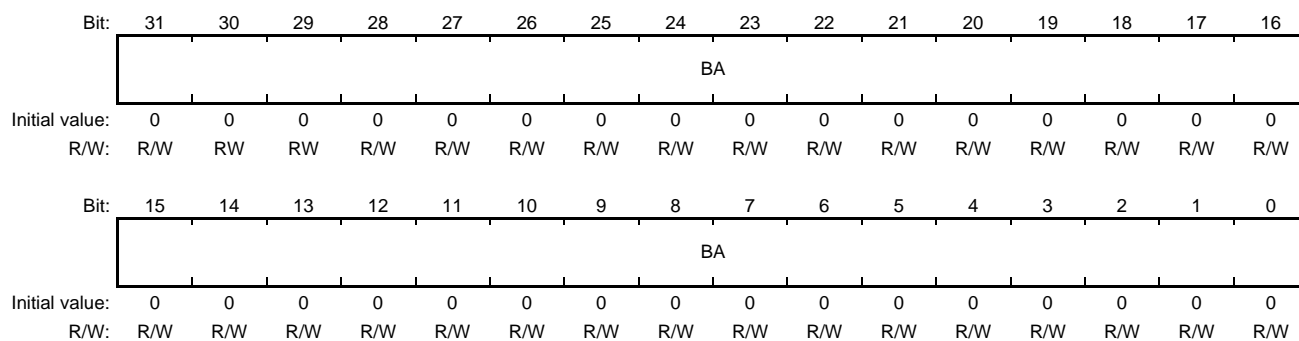


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | BA | All 0 | R/W | <p>Base Address of ancillary data</p> <p>The base address of the ancillary information of the plane Y0, Y1, Y2, and C. It must be specified in 128 bytes unit.</p> <p>In case of FDP connection Y0, Y1, Y2 are corresponding to the luma plane of previous, current, next field (frame in case of progressive input) respectively. C is corresponding to the chroma plane of the current field (frame).</p> <p>Be sure to set 0 to this register in one of following cases</p> <ol style="list-style-type: none"> 1. Tile/Linear address conversion is off (FCP_TL_CTRL.TLEN = 0) 2. Tile/Linear address conversion is on (FCP_TL_CTRL.TLEN = 1) but the corresponding plane is unused. <p>This bit must be set value from 0 to H'FFFF_FF80</p> |

35.4.6.7 FCP_BA_REF_m {m: Y0, Y1, Y2, C}: FCP Base Address Reference picture of plane m

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

FCP_BA_REF_m specifies the base address of the reference picture of the plane Y0, Y1, Y2, C which is located in tile addressing. Those registers are different from the source address registers set in FDP. Refer to the section 35.4.7.4(2) to grasp the way of setting address registers.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | BA | All 0 | R/W | <p>Base Address of reference picture</p> <p>The base address of the tile addressing image of the plane Y0, Y1, Y2, C. It must be specified in 16384 bytes unit.</p> <p>In case of FDP connection Y0, Y1, Y2 are corresponding to the luma plane of previous, current, next field (frame in case of progressive input) respectively. C is corresponding to the chroma plane of the current field (frame).</p> <p>This bit must be set value from 0 to H'FFFF_4000</p> |

35.4.7 Operation

This section explains how to control operation of an IP set which includes FCPF.

35.4.7.1 Operation Control Setting for the IP set FCPF+FDP

(1) The case FDP in Interrupt Mode or Best Effort Mode

(1a) to start the first frame, set registers of FDP, FCPF as following order.

1. Set value 1 to register FD1_CTL_CLKCTRL.CSTP_N
2. Set FDP registers other than FD1_CTL_CMD, FD1_CTL_SGCMD, and FD1_CTL_REGEND. FDP registers must follow restriction at 35.4.7.3.
In case of writing FCNL data set 1 to FD1_WPF_FORMAT.FCNL.
3. Set FCPF registers (V-update registers and other registers if necessary).
In case of reading tile addressing image set 1 to FCP_TL_CTRL.TLEN.
4. Set FD1_CTL_CMD.STRCMD to the value 1. Set FD1_CTL_REGEND.REGEND to the value 1.
5. Set FD1_CTL_SGCMD.SGEN to the value 1, then FDP starts its process. FD1_CTL_SGCMD.SGEN must be not cleared if continue to the second or later frames.

(1b) to start the second or later frame, set registers in following order.

1. After V-interruption of the previous frame start timing occurred, set all FDP registers other than FD1_CTL_CMD and FD1_CTL_REGEND. FDP registers must follow restriction at 35.4.7.3.
In case of writing FCNL data set 1 to FD1_WPF_FORMAT.FCNL.
2. Set FCPF V-update registers.
In case of reading tile addressing image set 1 to FCP_TL_CTRL.TLEN.
3. Set FD1_CTL_CMD.STRCMD to the value 1.
4. Set FD1_CTL_REGEND.REGEND to the value 1.
5. At the V-interruption of the previous frame end timing, FDP starts the target frame. The V-interruption of the previous frame end timing and the one of the target frame start timing are the same.

(1c) to stop FDP, FCPF operation after current frame finished, set registers as the following.

1. After V-interruption of the current frame start timing occurred, set FD1_CTL_CMD.STRCMD to the value 0.
2. Set FD1_CTL_REGEND.REGEND to the value 1. If this register was completed before the V-Interruption of frame end timing, FDP stops its operation and current frame becomes the last frame of the image sequence. If these register setting could not be completed before the V-Interruption of frame end timing, FDP stops its operation in the next V-interruption. In the latter case, V-update status registers are not updated at V-interruption of frame end but at the next V-interruption at which FD1_CTL_REGEND.REGEND is set as 1.
3. Set FD1_CTL_SGCMD.SGEN to the value 0, then FDP will stop generating V-interruption.

(1d) to stop FDP, FCPF operation immediately, set registers as the following.

1. Set FD1_CTL_SGCMD.SGEN to the value 0, then FDP will stop generating V-interruption.
2. Set FD1_CTL_SRESET.SRST to the value 1. FDP will invoke termination process immediately.
3. Wait frame end interrupt from FDP or until the register bit FD1_CTL_IRQSTA.FRE is set to 1. After it occurs, both FDP and FCPF have finished its processing.
4. Set FCP_RST.SOFTRST to the value 1. FCPF will invoke termination process immediately.
5. Wait until ACT bit of the FCP_STA register become 0. After it occurs, FCPF has finished its processing.

With this operation, FDP can stop its process quickly, but the output frame of the last frame is corrupted.

(2) The case FDP in No interrupt mode

(2a) to start the first frame, set registers in following order.

Same as (1a).

(2b) to start the second or later frame, set registers in following order.

Same as (1b).

(2c) to stop FDP operation after current frame finished, set registers as the following.

1. After V-interruption of the current frame start timing occurred, set FD1_CTL_CMD.STRCMD to the value 0.
2. Set FD1_CTL_REGEND.REGEND to the value 1. At the V-Interruption of frame end timing, FDP stops its operation and current frame becomes the last frame of the image sequence.
3. Set FD1_CTL_SGCMD.SGEN to the value 0, then FDP will stop generating V-interruption.

(2d) to stop FDP operation immediately, set registers as the following.

Same as (1d).

35.4.7.2 Reset Operation

FCPF reset can be conducted by software reset and power on reset. The software reset which is controlled by software, is used to stop the FCPF operation. The software reset operation is not executed immediately; it is executed on completion of the bus transaction. The other is power on reset which initialize the whole FCPF logic immediately.

(1) Software Reset

The procedure of software reset of the FCPF is as following:

- 1) Activate software reset sequence

Set FCP_RST.SOFTRST to the value 1.

- 2) Confirm software reset sequence finished:

FCP_STA.ACT becomes 0. All remain transactions with the bus system finished.

The procedure to stop operation of an IP set including FCPF is described at the sections 35.4.7.1.

(2) Power-on Reset

The power-on reset is controlled by an externally input signal called hardware reset. This reset initializes the whole internal logic of FCPF.

35.4.7.3 Register settings and restrictions on connected IP (FDP)

The connected FDP has to follow the setting and restrictions in following tables.

(1) Register setting on connected IP (FDP)

Table 35.22 Setting restriction when writing FCNL data

| Register | Register value | Use cases |
|---------------------|----------------|---------------------|
| FD1_WPF_FORMAT.FCNL | 0 | Write non FCNL data |
| | 1 | Write FCNL data |

(2) Restrictions on FDP read side

In case of reading tile addressing image (FCP_TL_CTRL.TLEN = 1), the setting of input data swap (FD1_RPF_SWAP.ISWAP [3:0]), line memory pixel number (FD1_IPC_LMEM.PNUM [11:0]), read color format (FD1_RPF_FORMAT.RD T [6:0]), read memory stride (FD1_RPF_PSTRIDE.SRC_STRD_Y [15:0]), FD1_RPF_PSTRIDE.SRC_STRD_C [15:0]) must follow the Table 35.23.

Table 35.23 Setting restriction on FDP read side

| Register | Value |
|--------------------------------------|--------------------------------|
| FD1_RPF_SWAP.ISWAP[3:0] | H'F |
| Max value of FD1_IPC_LMEM.PNUM[11:0] | 960 |
| FD1_RPF_FORMAT.RD T[6:0] | H'42 (YCbCr420 semi-planar) |
| FD1_RPF_PSTRIDE.SRC_STRD_Y[15:0] | STRIDE |
| FD1_RPF_PSTRIDE.SRC_STRD_C[15:0] | STRIDE |

STRIDE = FCP_PICINFO1.STRIDE_DIV16 [10:0]*16

(3) Restrictions on FDP write side

The connected FDP have to follow the setting and restrictions in following tables.

In case of reading tile addressing image (FCP_TL_CTRL.TLEN = 1), the setting of still mask write data swap must follow Table 35.24.

Table 35.24 Setting restriction about still mask write data swap

| Register | Register Value |
|-------------------------|----------------|
| FD1_WPF_SWAP.SSWAP[3:0] | H'F |

In case of writing FCNL data (FD1_WPF_FORMAT.FCNL = 1), the setting of image write data swap (FD1_WPF_SWAP.OSWAP [3:0]), output color format (FD1_WPF_FORMAT.WR T [6:0]), output memory stride (FD1_WPF_PSTRIDE.DST_STRD_Y [15:0], FD1_WPF_PSTRIDE.DST_STRD_C [15:0]) and output memory address (FD1_WPF_ADDR_m.DST_ADDR_m [31:0] (m = Y, C0, C1)) must follow Table 35.25.

Table 35.25 Setting restriction on FDP write side

| Register | Register Value |
|--|---|
| FD1_WPF_SWAP.OSWAP[3:0] | H'8 |
| FD1_WPF_FORMAT. WR T[6:0] | H'13 (ARGB8888) H'47 (YCbCr 422 Interleave YUY2) (*) H'4A (YCbCr 444 Planar) H'4B (YCbCr 422 Planar) H'4C (YCbCr 420 Planar) |
| FD1_WPF_PSTRIDE.DST_STRD_Y[15:0] | 256n |
| FD1_WPF_PSTRIDE.DST_STRD_C[15:0] | (multiple of 256 bytes) |
| FD1_WPF_ADDR_m.DST_ADDR_m[31:0] (m = Y, C0, C1) | |

Note: * In this case, FD1_WPF_FORMAT.WSPYCS must be set to 1 and FD1_WPF_FORMAT.WSPUVS must be set to 0.

Data Stored in External RAM and Its Format

35.4.7.4 Decode data

Decode data consists of 2 separated data, one is ancillary information data and the other is tile addressing image data.

(1) Ancillary information data

The ancillary information data is an additional data to access to tile addressing image and to decompress the lossless compressed picture data. It is required for each Luminance plane and Chrominance plane. The information mention about whether decompress or not (the data is lossless compressed or not) is embedded in this ancillary data and automatically analyzed by FCPF.

The access to ancillary information data memory area of FCPF is shown in Table 35.26.

Table 35.26 Access to Ancillary Information memory area

| Write / Read | Access trigger | Remarks |
|--------------|---|-----------------------------|
| Write | None (FCPF does not conduct Write access) | |
| Read | Always if set the FCP_BA_ANC_ <i>n</i> (<i>n</i> : Y0/Y1/Y2/C) register | Pre-fetch the next 128bytes |

The size of ancillary information data is shown in Table 35.27. And the ancillary information data should be aligned on a 128-byte boundary.

Table 35.27 Data Size of Ancillary Information

| Number of Buffer | Data Size per Buffer |
|---|--|
| Same as maximum number of frames/fields | $\text{CEIL}_{128}(2 + ((2 \times \text{the number of GROUP in a picture}) + 7) / 8) + 384$ bytes (*) <p>[The number of GROUP in a picture]</p> <p>Luminance: $(\text{STRIDE} \times \text{CEIL}_{32}(\text{BLOCK_HEIGHT} + 4)) / 256$ (*)</p> <p>Chrominance: $(\text{STRIDE} \times \text{CEIL}_{32}(\text{BLOCK_HEIGHT}/2 + 4)) / 256$</p> <p>where</p> <p>$\text{STRIDE} = \text{FCP_PICINFO1}.\text{STRIDE_DIV16} * 16$</p> <p>$\text{BLOCK_HEIGHT} = \text{pixel height of picture in the size of MB or CTB boundary}$</p> <p>[pixel height of picture in the size of MB or CTB boundary]</p> <p>MB: $\text{CEIL}_{16}(\text{pixel height})$</p> <p>CtbSizeY = 16: $\text{CEIL}_{16}(\text{pixel height})$</p> <p>CtbSizeY = 32: $\text{CEIL}_{32}(\text{pixel height})$</p> <p>CtbSizeY = 64: $\text{CEIL}_{64}(\text{pixel height})$</p> <p>Ex) Picture size: 1920 × 1080, STRIDE: 2048, CtbSizeY: 16</p> <p>Luminance:</p> <p>The number of GROUP in a picture</p> $= (2048 \times \text{CEIL}_{32}(\text{CEIL}_{16}(1080) + 4)) / 256$ $= 8960$ <p>Data size of Ancillary information</p> $= \text{CEIL}_{128}(2 + ((2 \times 8960) + 7) / 8) + 384$ $= 2688 \text{ bytes}$ <p>Chrominance:</p> <p>The number of GROUP in a picture</p> $= (2048 \times \text{CEIL}_{32}(\text{CEIL}_{16}(1080)/2 + 4)) / 256$ $= 4608$ <p>Data size of Ancillary information</p> $= \text{CEIL}_{128}(2 + ((2 \times 4608) + 7) / 8) + 384$ $= 1664 \text{ bytes}$ |

Note: * $\text{CEIL}_n(a) = ((a + (n - 1)) / n) * n$

(2) Tile addressing image data

Image data is located on memory in tile addressing as shown in Figure 35.6. Although the valid image data exists in the green area only, the tile addressing area is extended to horizontal and vertical direction. The tile size is 128bytes x 32lines. The stride of allocation memory area is exponent of 2.

The pixels in the first line of the picture are stored in 2 lines shift down in the case of H.264 frame picture in interlace sequence (not MBAFF). Note that reading access to an address is converted to the address of 2lines below. The Figure 35.7 shows examples of the stored 2 line shift down.

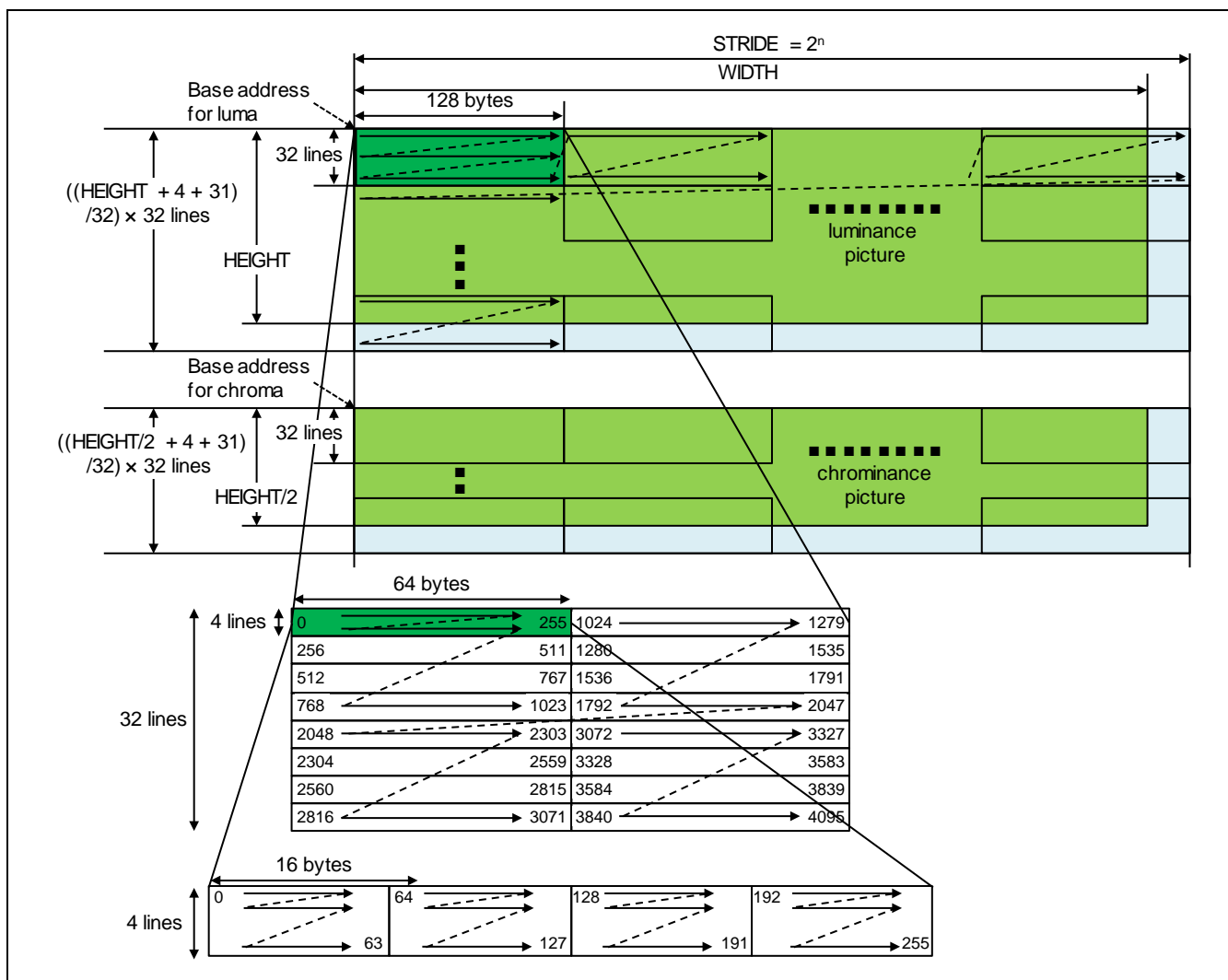


Figure 35.6 Memory allocation of tile addressing image

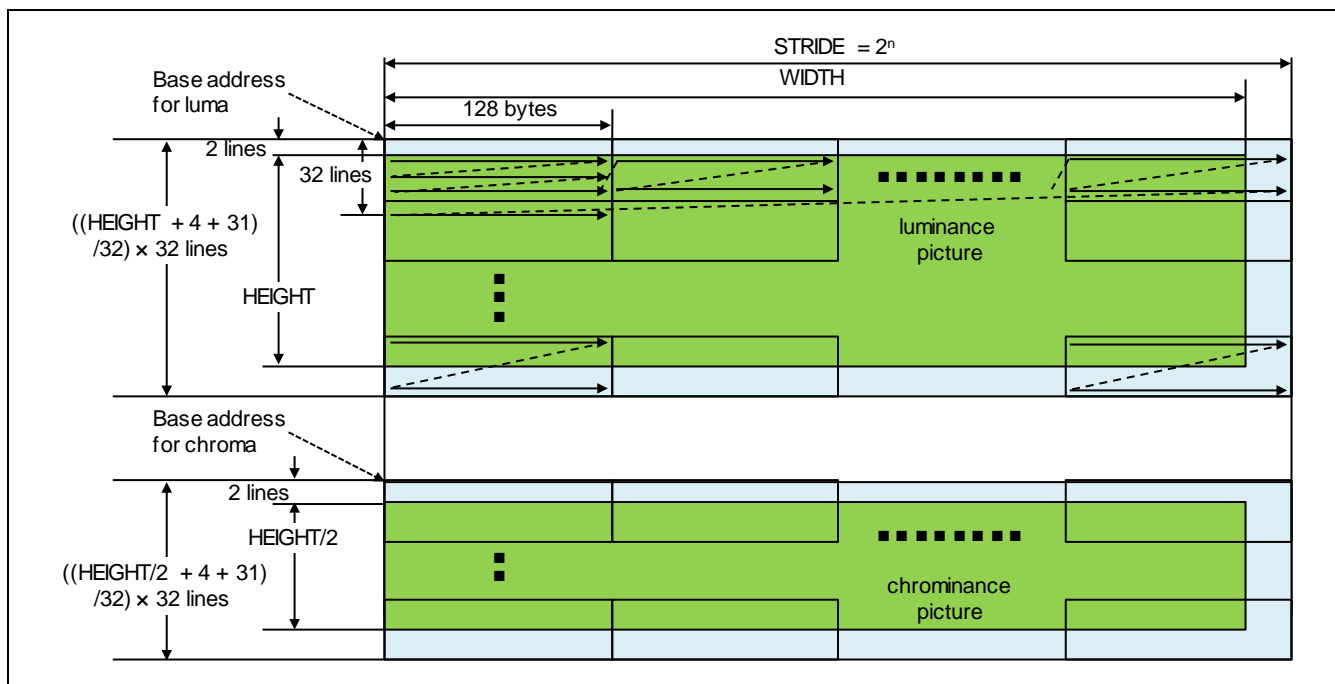


Figure 35.7 An Example of 2 line shift down

Parameters in Figure 35.6 and Figure 35.7:

- STRIDE = FCP_PICINFO1. STRIDE_DIV16 * 16
- HEIGHT, WIDTH: the height and width of the tile addressing image
- Base address for luma: value of register FCP_BA_REF_{Y0,Y1,Y2}.BA[31:0]
- Base address for chroma: value of register FCP_BA_REF_C.BA[31:0]

The buffer size needed for a tile addressing image is shown in Table 35.28

:

Table 35.28 Data Size of a tile addressing image

| Number of Pic | Data Size per Picture |
|---------------------------------|--|
| Maximum number of frames/fields | Luminance: STRIDE x CEIL32(BLOCK_HEIGHT + 4) bytes (*) Chrominance: STRIDE x CEIL32(BLOCK_HEIGHT/2 + 4) bytes where STRIDE = FCP_PICINFO1. STRIDE_DIV16 * 16 BLOCK_HEIGHT = pixel height of picture in the size of MB or CTB boundary [pixel height of picture in the size of MB or CTB boundary] MB: CEIL16(pixel height) CtbSizeY = 16: CEIL16 (pixel height) CtbSizeY = 32: CEIL32 (pixel height) CtbSizeY = 64: CEIL64 (pixel height) |

Note: * CEILn(a) = ((a + (n - 1)) / n) * n

(3) Tile/Linear addressing conversion

In order to enable the FDP which supports only linear addressing to access the tile addressing image data, the software needs to set the registers following to Figure 35.8. Note that the value of VPOS_{Y/C} is set basing on base address regardless of 2 lines of line-shift because reading access to an address is converted to the address of 2lines below.

In order to enable tile/linear addressing conversion, set 1 to the register FCP_TL_CTRL.TLEN and set values other than 0 to base address of ancillary information (FCP_BA_REF_m {m: Y0, Y1, Y2, C}) of the in-used plane.

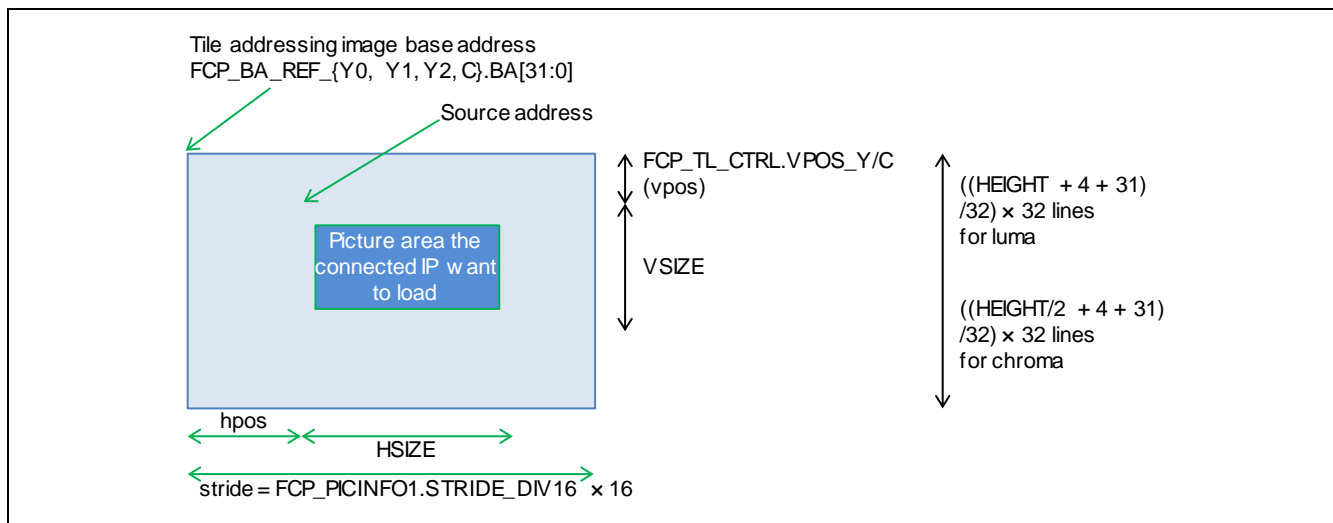


Figure 35.8 Reading tile addressing image by linear access

Parameters in Figure 35.8:

- “Source address”: $FD1_RPF\{n\}_ADDR_{Y/C0}.PSRC_ADDR_{Y/C0}[31:0]$, $n = 0/1/2$
 “Source address” = “Tile addressing image base address” + $vpos * stride + hpos$
- hpos: the start reading byte position for each line in linear addressing. This value is automatically calculated by FCP and not needed to be set as register.
- VSIZE: $FD1_RPF_SIZE.VSIZE[12:0]$
- HSIZE: $FD1_RPF_SIZE.HSIZE[12:0]$

Registers of FDP must satisfy the restrictions on 35.4.7.3.

35.4.7.5 FCNL image data

Figure 35.9 shows the memory layout of a FCNL image. Although the output memory area, which is specified by a destination address and a stride, is a rectangle linear address area, the FCNL data is located on the shadow address area of that. That is, in each 256 bytes area starting from 256 bytes boundary, FCNL data is located on 128 bytes area of the lower side address.

The destination start address and memory stride which is specified by VSP registers must satisfy the restriction on section 35.4.7.3.

The buffer size for a FCNL image is equal to $stride * height$. Here, height is half of the picture height if the plane is Cb (U), Cr (V) for format YCbCr 420 planar and equal to the picture height for other cases.

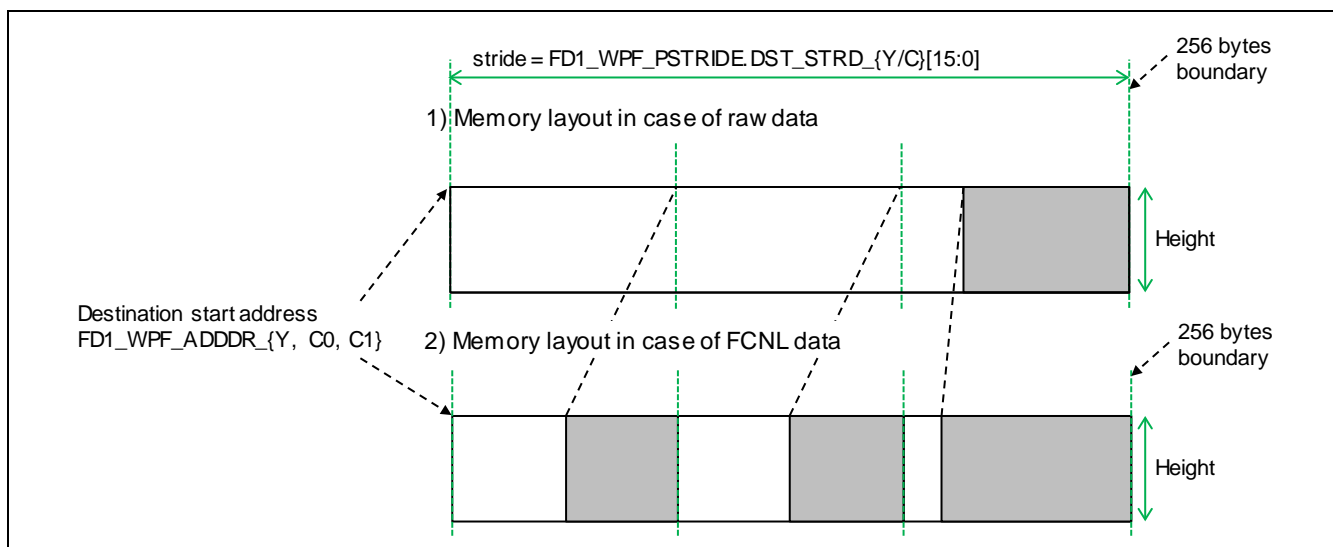


Figure 35.9 Memory layout of the FCNL image

Parameters in Figure 35.9:

- Height: the height of output image.

35.4.7.6 Operation status

Operation status of FDP, FCPF is defined for the whole IP set as follows.

- 1) If both register bits $FD1_CTL_STATUS.BSY$ and $FCP_STA.ACT$ are 0, both of modules FDP, FCPF are stopping.
- 2) If either register bit $FD1_CTL_STATUS.BSY$ or $FCP_STA.ACT$ is 1, both of modules FDP, FCPF are operating.

35.4.8 Usage Notes

35.4.8.1 Manipulating FCPF+FDP with CPG module stop, software reset

The following notes 1) to 2) are to prevent FCPF, FDP from the unstable status which may cause hang or wrong operation.

Note 1) In the case of stopping FCPF and FDP using the corresponding register bit of module stop registers (MSTPSR1, MSTPSR6), realtime module stop registers (RMSTPCR1, RMSTPCR6), system module stop registers (SMSTPCR1, SMSTPCR6) and software reset registers (SRCR1, SRCR6) of the module CPG, do following steps. Note that it is needed to stop both FCPF and FDP.

Step1) Stop FCPF + FDP with the software reset sequence following to the guidance in the section 35.4.7.1.

Step2) Stop FDP using the corresponding register bit of CPG register (MSTPSR1 or RMSTPCR1 or SMSTPCR1 or SRCR1) and stop FCPF using the corresponding register bit of CPG register (MSTPSR6 or RMSTPCR6 or SMSTPCR6 or SRCR6). Either FDP or FCPF can be stopped first.

Note 2) In the case of re-starting FCPF + FDP after stopping those modules with CPG module stop, CPG software reset, be sure to run FDP start process as defined in the section 35.4.7.1.

36. Display Unit (DU)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

36.1 Overview

Display Unit (hereinafter it is called DU) is a module to input an image data from VSP* and to output the data to LVDS-IF, HDMI-IF, and digital RGB etc. in accordance with optionally settable display timing.

Note: In this section, VSP means VSP2 (Video Signal Processor 2), the notation of VSP0, VSP1 and VSP2 in this section means channel-0 of VSP2, channel-1 of VSP2 and channel-2 of VSP2.

When using DU, set the desired value in VSP2's registers. For details of VSP2, refer to section 33 in this manual.

36.1.1 Features

(1) Display Channel

Display functions are individually controllable, and equip plural channels. For the channel number of each products, refer to Table 36.1.

(2) Plane

Each of the display surfaces configured to display an image on the screen is called plane in this section. For the plane number of each products, refer to Table 36.1. Parameters for each plane can be set independently through the settings of an internal register. The internal register settings can also be used to set the display priority order.

- Display size
- Display position
- Display data format (32-bit/pixel or YC)
- Plane superimposing

(3) Synchronization Method

Internal register settings can be used to select any of three synchronization modes for the display output timing. RZ/G2E support only master mode (internal sync mode).

- Master mode (internal sync mode)
- TV sync mode (external sync mode)
- Sync method switching mode

(4) CRT Scan Mode (CRT Scan Method)

Internal register settings can be used to select from among three scan modes. RZ/G2E support only non-interlaced mode.

- Non-interlaced mode
- Interlaced sync mode
- Interlaced sync & video mode

(5) YC → RGB Color Space Conversion Functions

Image data stored in YC format can be converted into the RGB color space and displayed in a window. The conversion coefficients can be set in a register.

(6) Register Access Control

The module has internal control registers; these are writable/readable by the APB protocol over the APB. The unit of access is fixed to 32 bits.

(7) VSP Connection

VSP output provides RGB or YCbCr444 data format for the DU. The DU is not connected with external memory. For VSP connection of each products, refer to Table 36.1.

(8) LVDS-IF Connection

Connection with the LVDS-IF allows the output of image data via the LVDS-IF. In this case, data format is RGB. For LVDS-IF connection of each products, refer to Table 36.1.

(9) DPAD Connection

Connection with the DPAD allows the output of the digital RGB data and the YC format data via a pin. In this case, data format is RGB or YCbCr422. For DPAD connection of each products, refer to Table 36.1.

(10) HDMI-IF Connection [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Connection with the HDMI-IF allows the output of image data via the HDMI-IF. In this case, data format is RGB or YCbCr444. For HDMI-IF connection of each products, refer to Table 36.1.

Table 36.1 Configuration of channel, plane and connection for each product

| | | | | | | | Second Generation RZ/G Series Products | | | |
|----------|-------|---------------------|------|------------|----------|------------|---|----------------------------|--------|--------|
| Channel | Abbr. | Plane | VSP | Connection | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| | | | | LVDS-IF | HDMI-IF | DPAD | | | | |
| channel0 | DU0 | plane1 and plane3*1 | VSP0 | LVDS-IF | — | — | √ | √ | √ | — |
| | | | | LVDS0-IF | — | DPAD *2 | — | — | — | √ |
| channel1 | DU1 | plane1 and plane3*1 | VSP1 | — | HDMI-IF0 | — | √ | √ | √ | — |
| | | | | LVDS1-IF | — | DPAD *2 | — | — | — | √ |
| channel2 | DU2 | plane1 | VSP2 | — | — | DPAD | — | √ | — | — |
| channel3 | DU3 | plane1 and plane3*1 | VSP0 | — | — | DPAD | √ | — | — | — |
| | | plane3 | | | | | — | — | √ | — |

- Notes: 1. Up to two display planes can be superimposed on each other. Correspondences between the DU0 and DU1, the DU2 and DU3 planes are not fixed but can be selected as desired through register settings.
2. DPAD can be connected with the DU0 or DU1. [RZ/G2E]

Table 36.2 Corresponding data format for each product

| Product | Implementation | | | | Corresponding data format | | | |
|-----------------------------|----------------|-----|-----|-----|---------------------------|--------------|--------------|--------------|
| | DU0 | DU1 | DU2 | DU3 | DU0 | DU1 | DU2 | DU3 |
| RZ/G2H | √ | √ | — | √ | RGB | RGB/YCbCr444 | — | RGB/YCbCr422 |
| RZ/G2M V1.3, RZ/G2M V3.0 | √ | √ | √ | — | RGB | RGB/YCbCr444 | RGB/YCbCr422 | — |
| RZ/G2N | √ | √ | — | √ | RGB | RGB/YCbCr444 | — | RGB/YCbCr422 |
| RZ/G2E | √ | √ | — | — | RGB/YCbCr422 | RGB/YCbCr422 | — | — |

36.1.2 Block Diagram

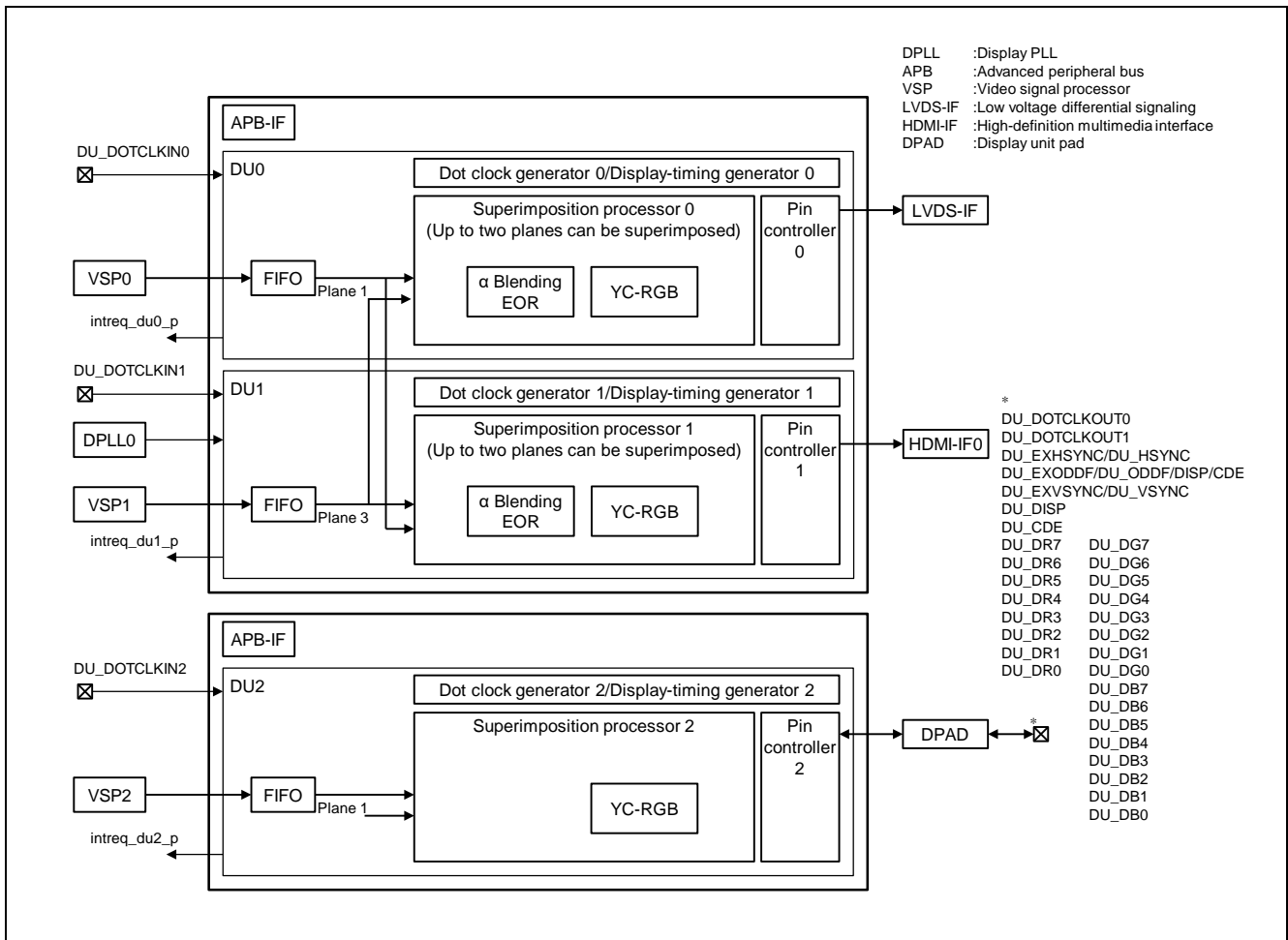


Figure 36.1 Block Diagram of the DU [RZ/G2M V1.3, RZ/G2M V3.0]

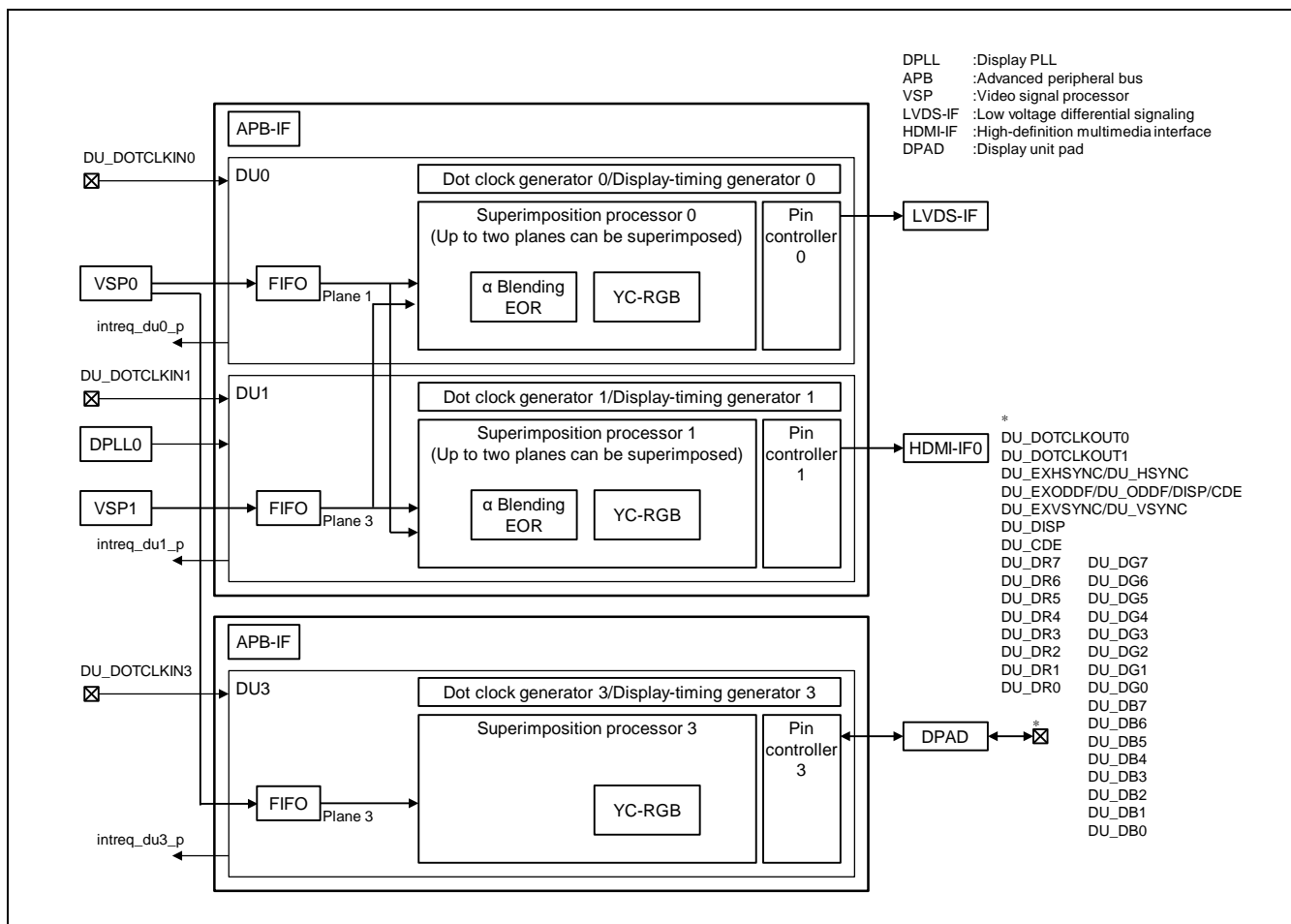


Figure 36.2 Block Diagram of the DU [RZ/G2H, RZ/G2N]

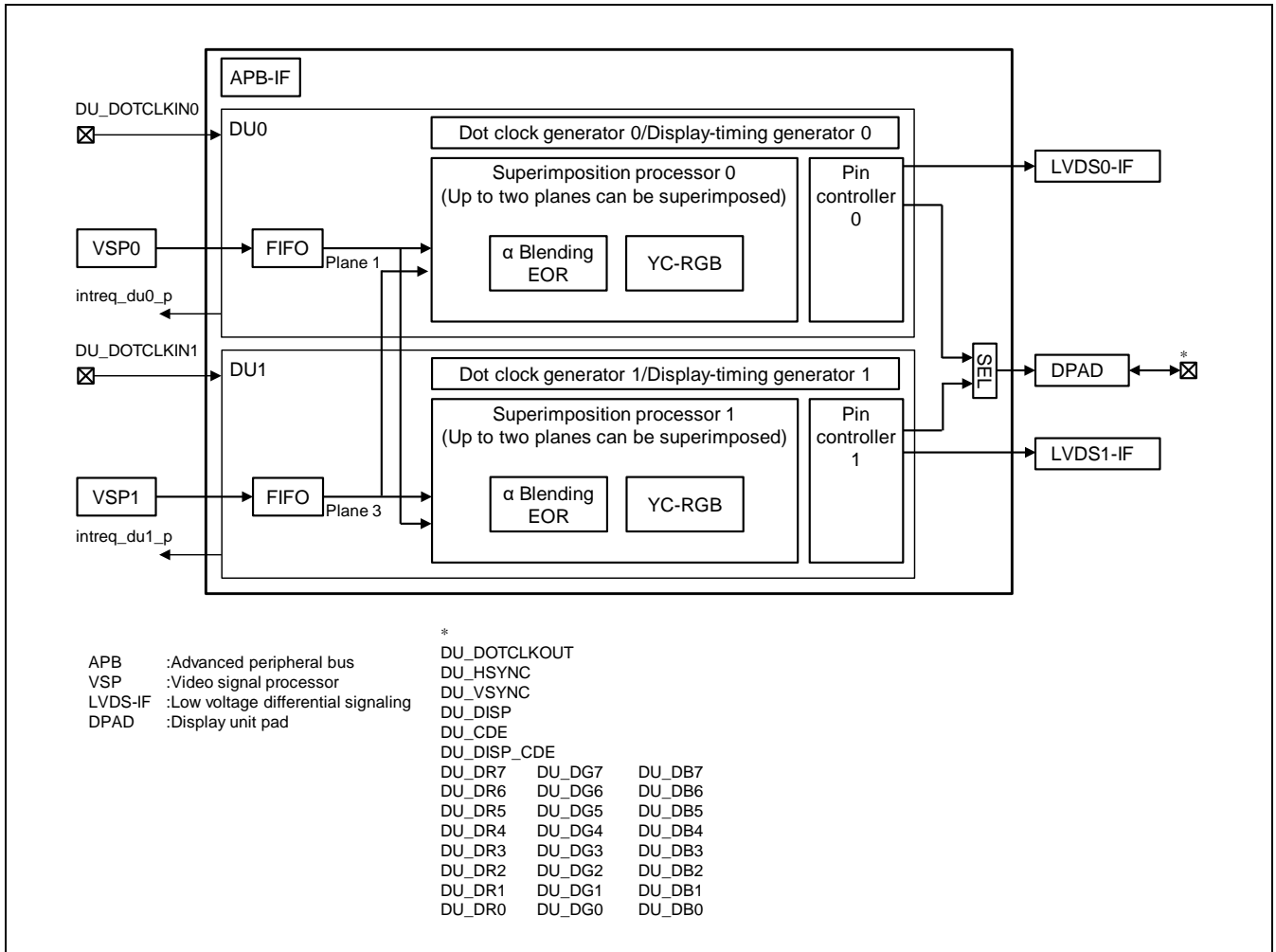


Figure 36.3 Block Diagram of the DU [RZ/G2E]

36.1.3 External Pins

Supplementary explanation of the signals listed in column “Signal Name Used in This Section” are as follows.

DISP is a signal activated during the period “Display area” in figure 36.14 and is activated during output display data valid period.

CLAMP is a signal that determines the activation period with CLAMPSRn and CLAMPWRn registers. Refer to these two registers.

DE is a signal that determines the activation period with DESRn and DEWRn registers. Refer to these two registers.

CDE is a signal activated when output display data and CDERn register match. Refer to section 36.3.13.

Table 36.3 Pin Functions (DU0/DU1/DU3 [RZ/G2H, RZ/G2N])

| Name | Pin Name | I/O | Function | Signal Name Used in This Section |
|---|--|--------|---|----------------------------------|
| DU0 input dot clock | DU_DOTCLKIN0 | Input | DU0 input dot clock | DCLKIN |
| DU1 input dot clock | DU_DOTCLKIN1 | Input | DU1 input dot clock | DCLKIN |
| DU3 input dot clock | DU_DOTCLKIN3 | Input | DU3 input dot clock | DCLKIN |
| DU output dot clock 0 | DU_DOTCLKOUT0 | Output | DU3 output dot clock (normal phase) | DCLKOUT |
| DU output dot clock 1 | DU_DOTCLKOUT1 | Output | DU3 output dot clock (counter phase) | DCLKOUTB |
| DU external horizontal synchronous output/DU horizontal synchronous input | DU_EXHSYNC/ DU_HSYNC | I/O | DU3 composite synchronous output signal (Initial value) | CSYNC |
| | | | DU3 horizontal synchronous output/DU3 external horizontal synchronous input | HSYNC or EXHSYNC*1 |
| DU external vertical synchronous output/DU vertical synchronous output | DU_EXVSYNC/ DU_VSYNC | I/O | DU3 vertical synchronous output/DU3 external vertical synchronous input (Initial value) | VSYNC or EXVSYNC*1 |
| | | | DU3 composite synchronous output signal | CSYNC |
| DU odd/even field | DU_EXODDF/ DU_ODDF/ DISP/ CDE | I/O | DU3 odd/even field (Initial value) | ODDF or EXODDF*1 |
| | | | DU3 CLAMP output signal | CLAMP |
| | | | DU3 display interval | DISP |
| | | | DU3 color detection | CDE |
| DU display interval | DU_DISP | Output | DU3 display interval (Initial value) | DISP |
| | | | DU3 composite synchronous output signal | CSYNC |
| | | | DU3 DE output signal | DE |
| DU color detection | DU_CDE | Output | DU3 color detection (Initial value) | CDE |

| Name | Pin Name | I/O | Function | Signal Name Used in This Section |
|-----------------|----------|--------|----------------------------|----------------------------------|
| DU display data | DU_DR0 | Output | DU3 digital red 0/green 4 | Digital RGB or YC*3 |
| | DU_DR1 | Output | DU3 digital red 1/green 5 | |
| | DU_DR2 | Output | DU3 digital red 2/green 6 | |
| | DU_DR3 | Output | DU3 digital red 3/green 7 | |
| | DU_DR4 | Output | DU3 digital red 4/blue 0 | |
| | DU_DR5 | Output | DU3 digital red 5/blue 1 | |
| | DU_DR6 | Output | DU3 digital red 6/blue 2 | |
| | DU_DR7 | Output | DU3 digital red 7/blue 3 | |
| | DU_DG0 | Output | DU3 digital green 0/blue 4 | |
| | DU_DG1 | Output | DU3 digital green 1/blue 5 | |
| | DU_DG2 | Output | DU3 digital green 2/blue 6 | |
| | DU_DG3 | Output | DU3 digital green 3/blue 7 | |
| | DU_DG4 | Output | DU3 digital green 4 | |
| | DU_DG5 | Output | DU3 digital green 5 | |
| | DU_DG6 | Output | DU3 digital green 6 | |
| | DU_DG7 | Output | DU3 digital green 7 | |
| | DU_DB0 | Output | DU3 digital blue 0 | |
| | DU_DB1 | Output | DU3 digital blue 1 | |
| | DU_DB2 | Output | DU3 digital blue 2 | |
| | DU_DB3 | Output | DU3 digital blue 3 | |
| | DU_DB4 | Output | DU3 digital blue 4 | |
| | DU_DB5 | Output | DU3 digital blue 5 | |
| | DU_DB6 | Output | DU3 digital blue 6 | |
| | DU_DB7 | Output | DU3 digital blue 7 | |

- Notes:
1. There are expressed as EXHSYNC, EXVSYNC, and EXODDF in explanations of the functions as input signals and as HSYNC, VSYNC and ODDF otherwise.
 2. In this section, unless otherwise noted, "dot clock" refers to the output dot clock.
 3. In YC format display, refer to Table 36.32.

Table 36.4 Pin Functions (DU0/DU1/DU2 [RZ/G2M V1.3, RZ/G2M V3.0])

| Name | Pin Name | I/O | Function | Signal Name Used in This Section |
|---|--|--------|---|----------------------------------|
| DU0 input dot clock | DU_DOTCLKIN0 | Input | DU0 input dot clock | DCLKIN |
| DU1 input dot clock | DU_DOTCLKIN1 | Input | DU1 input dot clock | DCLKIN |
| DU2 input dot clock | DU_DOTCLKIN2 | Input | DU2 input dot clock | DCLKIN |
| DU output dot clock 0 | DU_DOTCLKOUT0 | Output | DU2 output dot clock (normal phase) | DCLKOUT |
| DU output dot clock 1 | DU_DOTCLKOUT1 | Output | DU2 output dot clock (counter phase) | DCLKOUTB |
| DU external horizontal synchronous output/DU horizontal synchronous input | DU_EXHSYNC/ DU_HSYNC | I/O | DU2 composite synchronous output signal (Initial value) | CSYNC |
| | | | DU2 horizontal synchronous output/DU2 external horizontal synchronous input | HSYNC or EXHSYNC*1 |
| DU external vertical synchronous output/DU vertical synchronous output | DU_EXVSYNC/ DU_VSYNC | I/O | DU2 vertical synchronous output/DU2 external vertical synchronous input (Initial value) | VSYNC or EXVSYNC*1 |
| | | | DU2 composite synchronous output signal | CSYNC |
| DU odd/even field | DU_EXODDF/ DU_ODDF/ DISP/ CDE | I/O | DU2 odd/even field (Initial value) | ODDF or EXODDF*1 |
| | | | DU2 CLAMP output signal | CLAMP |
| | | | DU2 display interval | DISP |
| | | | DU2 color detection | CDE |
| DU display interval | DU_DISP | Output | DU2 display interval (Initial value) | DISP |
| | | | DU2 composite synchronous output signal | CSYNC |
| | | | DU2 DE output signal | DE |
| DU color detection | DU_CDE | Output | DU2 color detection (Initial value) | CDE |

| Name | Pin Name | I/O | Function | Signal Name Used in This Section |
|-----------------|----------|--------|---------------------|----------------------------------|
| DU display data | DU_DR0 | Output | DU2 digital red 0 | Digital RGB or YC* ³ |
| | DU_DR1 | Output | DU2 digital red 1 | |
| | DU_DR2 | Output | DU2 digital red 2 | |
| | DU_DR3 | Output | DU2 digital red 3 | |
| | DU_DR4 | Output | DU2 digital red 4 | |
| | DU_DR5 | Output | DU2 digital red 5 | |
| | DU_DR6 | Output | DU2 digital red 6 | |
| | DU_DR7 | Output | DU2 digital red 7 | |
| | DU_DG0 | Output | DU2 digital green 0 | |
| | DU_DG1 | Output | DU2 digital green 1 | |
| | DU_DG2 | Output | DU2 digital green 2 | |
| | DU_DG3 | Output | DU2 digital green 3 | |
| | DU_DG4 | Output | DU2 digital green 4 | |
| | DU_DG5 | Output | DU2 digital green 5 | |
| | DU_DG6 | Output | DU2 digital green 6 | |
| | DU_DG7 | Output | DU2 digital green 7 | |
| | DU_DB0 | Output | DU2 digital blue 0 | |
| | DU_DB1 | Output | DU2 digital blue 1 | |
| | DU_DB2 | Output | DU2 digital blue 2 | |
| | DU_DB3 | Output | DU2 digital blue 3 | |
| | DU_DB4 | Output | DU2 digital blue 4 | |
| | DU_DB5 | Output | DU2 digital blue 5 | |
| | DU_DB6 | Output | DU2 digital blue 6 | |
| | DU_DB7 | Output | DU2 digital blue 7 | |

- Notes:
1. There are expressed as EXHSYNC, EXVSYNC, and EXODDF in explanations of the functions as input signals and as HSYNC, VSYNC and ODDF otherwise.
 2. In this section, unless otherwise noted, "dot clock" refers to the output dot clock.
 3. In YC format display, refer to Table 36.32.

Table 36.5 Pin Functions (DU0/DU1 [RZ/G2E])

| Name | Pin Name | I/O | Function | Signal Name Used in This Section |
|--|-----------------|------------|---|---|
| DU0 input dot clock | DU_DOTCLKIN0 | Input | DU0 input dot clock | DCLKIN |
| DU1 input dot clock | DU_DOTCLKIN1 | Input | DU1 input dot clock | DCLKIN |
| DU output dot clock 0 | DU_DOTCLKOUT0 | Output | DU0 or DU1*4 output dot clock (normal phase) | DCLKOUT |
| DU horizontal synchronous output | DU_HSYNC | Output | DU0 or DU1*4 horizontal synchronous output | HSYNC |
| DU vertical synchronous output | DU_VSYNC | Output | DU0 or DU1*4 vertical synchronous output | VSYNC |
| DU display interval/ DU color detection | DU_DISP_CDE*1 | Output | DU0 or DU1*4 CLAMP output signal | CLAMP |
| | | | DU0 or DU1*4 display interval | DISP |
| | | | DU0 or DU1*4 color detection | CDE |
| DU display interval | DU_DISP | Output | DU0 or DU1*4 display interval (Initial value) | DISP |
| | | | DU0 or DU1*4 DE output signal | DE |
| DU color detection | DU_CDE | Output | DU0 or DU1*4 color detection (Initial value) | CDE |

| Name | Pin Name | I/O | Function | Signal Name Used in This Section |
|-----------------|----------|--------|---|----------------------------------|
| DU display data | DU_DR0 | Output | DU0 or DU1* ⁴ digital red 0 green 4* ⁵ | Digital RGB or YC* ² |
| | DU_DR1 | Output | DU0 or DU1* ⁴ digital red 1 green 5* ⁵ | |
| | DU_DR2 | Output | DU0 or DU1* ⁴ digital red 2 green 6* ⁵ | |
| | DU_DR3 | Output | DU0 or DU1* ⁴ digital red 3 green 7* ⁵ | |
| | DU_DR4 | Output | DU0 or DU1* ⁴ digital red 4 blue 0* ⁵ | |
| | DU_DR5 | Output | DU0 or DU1* ⁴ digital red 5 blue 1* ⁵ | |
| | DU_DR6 | Output | DU0 or DU1* ⁴ digital red 6 blue 2 ⁵ | |
| | DU_DR7 | Output | DU0 or DU1* ⁴ digital red 7 blue 3* ⁵ | |
| | DU_DG0 | Output | DU0 or DU1* ⁴ digital green 0 blue 4* ⁵ | |
| | DU_DG1 | Output | DU0 or DU1* ⁴ digital green 1 blue 5* ⁵ | |
| | DU_DG2 | Output | DU0 or DU1* ⁴ digital green 2 blue 6* ⁵ | |
| | DU_DG3 | Output | DU0 or DU1* ⁴ digital green 3 blue 7* ⁵ | |
| | DU_DG4 | Output | DU0 or DU1* ⁴ digital green 4 | |
| | DU_DG5 | Output | DU0 or DU1* ⁴ digital green 5 | |
| | DU_DG6 | Output | DU0 or DU1* ⁴ digital green 6 | |
| | DU_DG7 | Output | DU0 or DU1* ⁴ digital green 7 | |
| | DU_DB0 | Output | DU0 or DU1* ⁴ digital blue 0 | |
| | DU_DB1 | Output | DU0 or DU1* ⁴ digital blue 1 | |
| | DU_DB2 | Output | DU0 or DU1* ⁴ digital blue 2 | |
| | DU_DB3 | Output | DU0 or DU1* ⁴ digital blue 3 | |
| | DU_DB4 | Output | DU0 or DU1* ⁴ digital blue 4 | |
| | DU_DB5 | Output | DU0 or DU1* ⁴ digital blue 5 | |
| | DU_DB6 | Output | DU0 or DU1* ⁴ digital blue 6 | |
| | DU_DB7 | Output | DU0 or DU1* ⁴ digital blue 7 | |

- Notes:
1. Initial value of the DU_DISP_CDE pin is fixed to 0. For the DU_DISP_CDE pin functions as CLAMP or DISP or CDE output, it is necessary for setting the ODPM bit in DSMR0 and the ODPM02 bit in DEF6R0 for the DU0, or the ODPM bit in DSMR1 and the ODPM12 bit in DEF6R0 for the DU1.
 2. In this section, unless otherwise noted, "dot clock" refers to the output dot clock.
 3. In YC format display, refer to Table 36.32.
 4. Digital RGB signal from the DU0 or DU1 can be selected by setting the DRGBS bit in DEF8R0.
 5. RZ/G2E supports multiple output display.

36.1.4 Register Configuration

Table 36.6 shows the meaning of the suffix associated with register name and register abbreviation in this manual. The suffix, means different from Table 36.6, show as Notes.

Table 36.6 The meaning of suffix associated with register name and register abbreviation

| Suffix | Significance | Cond. | Value | Target | | | | Second Generation RZ/G Series Products | | | |
|--------|------------------------------|-------|-------|---------------|-------------------------------------|------------|-------------|---|----------------------------|--------|--------|
| | | | | channel/plane | component (Figure 36.1, 36.2, 36.3) | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| m | common channel number | — | 0 | DU0/DU1 | — | — | — | √ | √ | √ | √ |
| | | | 2 | DU2/DU3 | — | — | — | √ | √ | √ | — |
| n | channel and component number | — | 0 | DU0 | processer0 | generator0 | controller0 | √ | √ | √ | √ |
| | | | 1 | DU1 | processer1 | generator1 | controller1 | √ | √ | √ | √ |
| | | | 2 | DU2 | processer2 | generator2 | controller2 | — | √ | — | — |
| | | | 3 | DU3 | processer3 | generator3 | controller3 | √ | — | √ | — |
| p | plane number | m = 0 | 1 | Plane1 | — | — | — | √ | √ | √ | √ |
| | | | 3 | Plane3 | — | — | — | √ | √ | √ | √ |
| | | m = 2 | 1 | Plane1 | — | — | — | — | √ | — | — |
| | | | 3 | Plane3 | — | — | — | √ | — | √ | — |

For DU, there are two register update methods; the external update and the internal update. The internal update has two reflection timings. Refer to Table 36.7 for the update methods and the reflecting timings in DU registers.

Table 36.7 Register update methods

| Update Methods | Reflection timings | Definitions of the Internal Updates in the Bit Description Table | Remarks |
|-----------------|---|--|--|
| External Update | After the end of CPU access | Not available | — |
| Internal Update | When the DRES bit in DSYSRn is 1. | DRES | — |
| | When the DRES bit in DSYSRn is 1 and at the beginning of each frame, which is the VSYNC falling edge if the VSL bit in DSMRn is 0 and the VSYNC rising edge if the VSL bit in DSMRn is 1. | Available | Disabled when the IUPD bit in DSYSRn is 1. |

(1) External Update

An "external update" is an update which reflects the address-mapped register settings made by the CPU after the end of CPU access. Registers related to display control (for example, DSYSRn) and the settings of which are updated through external updates can be overwritten without display flicker by using bits 11 and 14 in DSSRn indicating the start position of the vertical blanking interval.

(2) Internal Update

An "internal update" is an update that reflects the address-mapped register settings with the internal update timing of the DU. Hence in the case of a register with an internal update function, even when the CPU overwrites address-mapped registers related to display operation without being aware of the display timing, display flicker can be prevented.

There are two types of internal update timings. One reflects the register value when the DRES bit is set to 1. The other reflects the register value in the following cases: (a) When the DRES bit is set to 1, and (b) when the VSYNC falling edge occurs if the VSL bit in DSMRn is 0 and when the VSYNC rising edge occurs if the VSL bit in DSMRn is 1. The type of internal update timings is defined by registers and that appears in the bit description table.

Bits which are internally updated in response to setting of the DRES bit in DSYSRn are listed in the column headed "Bit with Internal Update Function" in tables on the following pages.

Updates are performed at the falling edge of VSYNC output when the sync method of DSYSRn is master mode (bit 7 = 0, bit 6 = 0), or at the falling edge of EXVSYNC detected in TV sync mode (bit 7 = 1, bit 6 = 0). In sync transition mode (bit 7 = 0, bit 6 = 1), internal updates are not performed. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

For RZ/G2E, Updates are performed at the falling edge of VSYNC output.

The address-mapped registers with an internal update function are shown in Table 36.8 to Table 36.21. The initial settings for these registers should be made during the interval in which the DRES bit in DSYSRn is 1.

(3) Register Configurations**(a) Display Unit System Control Register Configuration****Table 36.8 Display Unit System Control Register Configuration (1)**

Base address: DU0: H'FEB0_0000 (suffix 0)

DU1: H'FEB3_0000 (suffix 1)

DU2: H'FEB4_0000 (suffix 2) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

DU3: H'FEB7_0000 (suffix 3) [RZ/G2H, RZ/G2N]

| Register Name | Abbr. | R/W | Offset Address | Access Size | Bit with Internal Update Function* | Second Generation RZ/G Series Products | | | | |
|---|--------|-----|----------------|-------------|--|--|-------------|-------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Display unit system control register n | DSYSR0 | R/W | H'0000 | 32 | DEN (bit 8) | √ | √ | √ | √ | |
| | DSYSR1 | | | | None | √ | √ | √ | √ | |
| | DSYSR2 | | | | DEN (bit 8) | √ | √ | √ | — | |
| | DSYSR3 | | | | None | √ | — | √ | — | |
| Display unit mode register n | DSMRn | R/W | H'0004 | 32 | All bits However, the following bits are updated with DRES: VSPM (bit 28) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] ODPM (bit 27) DIPM (bits 26, 25) CSPM (bit 24) DIL (bit 19) VSL (bit 18) HSL (bit 17) | √ | √ | √ | √ | |
| Display unit status register n | DSSRn | R | H'0008 | 32 | None | √ | √ | √ | √ | |
| Display unit status register clear register n | DSRCRn | —/W | H'000C | 32 | None | √ | √ | √ | √ | |
| Display unit interrupt enable register n | DIERn | R/W | H'0010 | 32 | None | √ | √ | √ | √ | |
| Display plane priority register m | DPPRm | R/W | H'0018 | 32 | All bits | √ | √ | √ | √ | |

| | | | | | | Second Generation RZ/G Series Products | | | | |
|--|--------|-----|-------------------|----------------|--|---|-------------|-------------|--------|--------|
| Register Name | Abbr. | R/W | Offset Address | Access Size | Bit with Internal Update Function* | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| | | | | | | √ | √ | √ | √ | √ |
| Display unit extensional function control register n | DEFR0 | R/W | H'0020 | 32 | The following bits are updated with DRES: EXSL (bit 12) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] DODF (bits 9, 8) [RZ/G2E] EXUP (bit 5) VCUP (bit 4) DEFE (bit 0) | √ | √ | √ | √ | √ |
| | DEFR1 | | | | The following bits are updated with DRES: EXSL1 (bit 12) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] DODF1 (bits 9, 8) [RZ/G2E] VCUP1 (bit 4) | √ | √ | √ | √ | √ |
| | DEFR2 | | | | The following bits are updated with DRES: EXSL (bit 12) [RZ/G2M V1.3, RZ/G2M V3.0] DODF (bits 9, 8) [RZ/G2M V1.3, RZ/G2M V3.0] EXUP (bit 5) VCUP (bit 4) [RZ/G2M V1.3, RZ/G2M V3.0] DEFE (bit 0) | √ | √ | √ | — | — |
| | DEFR3 | | | | The following bits are updated with DRES: EXSL1 (bit 12) DODF1 (bits 9, 8) VCUP1 (bit 4) | √ | — | √ | — | — |
| Display unit extensional function control 5 register m | DEF5Rm | R/W | H'00E0 | 32 | All bits However, the following bit is updated with DRES: DEFE5 (bit 0) | √ | √ | √ | √ | √ |
| Display unit data latency adjustment register 0 | DDLTR0 | R/W | H'00E4 | 32 | All bits Updated with DRES | √ | √ | √ | √ | √ |
| Display unit extensional function control 6 register m | DEF6Rm | R/W | H'00E8 | 32 | All bits Updated with DRES | √ | √ | √ | √ | √ |
| Display unit extensional function control 7 register m | DEF7Rm | R/W | H'00EC | 32 | All bits Updated with DRES | √ | √ | √ | √ | √ |

Base address: DU0: H'FEB2_0000 (suffix 0)
 DU1: H'FEB2_0000 (suffix 1)
 DU2: H'FEB6_0000 (suffix 2) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 DU3: H'FEB6_0000 (suffix 3) [RZ/G2H, RZ/G2N]

| Register Name | Abbr. | R/W | Offset Address | Access Size | Bit with Internal Update Function* | Second Generation RZ/G Series Products | | | | |
|---|----------|-----|----------------|-------------|---|--|-------------|-------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Display unit domain 1 status register n | DD1SSR0 | R | H'0008 | 32 | None | √ | √ | √ | √ | √ |
| | DD1SSR1 | | H'8008 | | | √ | √ | √ | √ | |
| | DD1SSR2 | | H'0008 | | | — | √ | — | — | |
| | DD1SSR3 | | H'8008 | | | √ | — | √ | — | |
| Display unit domain 1 status register clear register n | DD1SRCR0 | —/W | H'000C | 32 | None | √ | √ | √ | √ | √ |
| | DD1SRCR1 | | H'800C | | | √ | √ | √ | √ | |
| | DD1SRCR2 | | H'000C | | | — | √ | — | — | |
| | DD1SRCR3 | | H'800C | | | √ | — | √ | — | |
| Display unit domain 1 interrupt enable register n | DD1IER0 | R/W | H'0010 | 32 | None | √ | √ | √ | √ | √ |
| | DD1IER1 | | H'8010 | | | √ | √ | √ | √ | |
| | DD1IER2 | | H'0010 | | | — | √ | — | — | |
| | DD1IER3 | | H'8010 | | | √ | — | √ | — | |
| Display unit extensional function control 8 register m | DEF8Rm | R/W | H'0020 | 32 | All bits Updated with DRES | √ | √ | √ | √ | √ |
| Display unit output signal fixed level register m | DOFLR0 | R/W | H'0024 | 32 | None | — | — | — | √ | √ |
| | DOFLR2 | | | | | √ | √ | √ | — | |
| Display unit input dot clock select register m | DIDSR0 | R/W | H'0028 | 32 | All bits Updated with DRES | √ | √ | √ | √ | √ |
| Display unit extensional function control 10 register m | DEF10R0 | R/W | H'0038 | 32 | All bits However, the following bit is updated with DRES: DEFE10(bit 0) | √ | √ | √ | √ | √ |
| | DEF10R2 | | | | All bits However, the following bit is updated with DRES: DEFE10(bit 0) | √ | √ | √ | — | |
| Display unit dither control register m | DDTHCRm | R/W | H'003C | 32 | All bits | √ | √ | √ | √ | √ |
| Display unit PLL control register m | DPLLCR0 | R/W | H'0044 | 32 | All bits Updated with DRES | √ | √ | √ | — | — |
| Display unit PLL control 2 register m | DPLLC2R0 | R/W | H'0048 | 32 | All bits Updated with DRES | √ | √ | √ | — | — |

Note: The register available code is excluded. There is no internal update function for the register available code.

Table 36.9 Display Unit System Control Register Configuration (2)

| Register Name | Abbr. | Power-on Reset | Second Generation RZ/G Series Products | | | | |
|---|----------|----------------|--|----------------------------|--------|--------|--|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Display unit system control register n | DSYSRn | H'****_**0* | √ | √ | √ | √ | |
| Display unit mode register n | DSMRn | H'*0*0_0*** | √ | √ | √ | √ | |
| Display unit status register n | DSSRn | H'****_**** | √ | √ | √ | √ | |
| Display unit status register clear register n | DSRCRn | H'****_**** | √ | √ | √ | √ | |
| Display unit interrupt enable register n | DIERn | H'****_**** | √ | √ | √ | √ | |
| Display plane priority register m | DPPRm | H'****_**10 | √ | √ | √ | √ | |
| Display unit extensional function control register n | DEFRn | H'****_**** | √ | √ | √ | √ | |
| Display unit extensional function control 5 register m | DEF5Rm | H'****_0*** | √ | √ | √ | √ | |
| Display unit data latency adjustment register 0 | DDLTR0 | H'****_**** | √ | √ | √ | √ | |
| Display unit extensional function control 6 register m | DEF6Rm | H'****_0** | √ | √ | √ | √ | |
| Display unit extensional function control 7 register m | DEF7Rm | H'****_**** | √ | √ | √ | √ | |
| Display unit domain 1 status register n | DD1SSRn | H'****_**** | √ | √ | √ | √ | |
| Display unit domain 1 status register clear register n | DD1SRCRn | H'****_**** | √ | √ | √ | √ | |
| Display unit domain 1 interrupt enable register n | DD1IERn | H'****_**** | √ | √ | √ | √ | |
| Display unit extensional function control 8 register m | DEF8Rm | H'****_**** | √ | √ | √ | √ | |
| Display unit output signal fixed level register m | DOFLR0 | H'****_***0 | — | — | — | √ | |
| | DOFLR2 | H'****_0** | √ | √ | √ | — | |
| Display unit input dot clock select register m | DIDSR0 | H'****_**** | √ | √ | √ | √ | |
| Display unit extensional function control 10 register m | DEF10Rm | H'****_**** | √ | √ | √ | √ | |
| Display unit dither control register m | DDTHCRm | H'****_**** | √ | √ | √ | √ | |
| Display unit PLL control register m | DPLLCR0 | H'****_0000 | √ | √ | √ | — | |
| Display unit PLL control 2 register m | DPLLC2R0 | H'****_***0 | √ | √ | √ | — | |

(b) Display Timing Generation Register Configuration**Table 36.10 Display Timing Generation Register Configuration (1)**

Base address: DU0: H'FEB0_0000 (suffix 0)
 DU1: H'FEB3_0000 (suffix 1)
 DU2: H'FEB4_0000 (suffix 2) [RZ/G2M V1.3, RZ/G2M V3.0]
 DU3: H'FEB7_0000 (suffix 3) [RZ/G2H, RZ/G2N]

| | | | | | | Second Generation RZ/G Series Products | | | |
|-------------------------------------|----------|-----|----------------|-------------|-----------------------------------|--|----------------------------|--------|--------|
| Register Name | Abbr. | R/W | Offset Address | Access Size | Bit with Internal Update Function | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Horizontal display start register n | HDSRn | R/W | H'0040 | 32 | All bits | √ | √ | √ | √ |
| Horizontal display end register n | HDERN | R/W | H'0044 | 32 | All bits | √ | √ | √ | √ |
| Vertical display start register n | VDSRn | R/W | H'0048 | 32 | All bits | √ | √ | √ | √ |
| Vertical display end register n | VDERN | R/W | H'004C | 32 | All bits | √ | √ | √ | √ |
| Horizontal cycle register n | HCRn | R/W | H'0050 | 32 | All bits | √ | √ | √ | √ |
| Horizontal sync width register n | HSWRn | R/W | H'0054 | 32 | All bits | √ | √ | √ | √ |
| Vertical cycle register n | VCRn | R/W | H'0058 | 32 | All bits | √ | √ | √ | √ |
| Vertical sync point register n | VSPRn | R/W | H'005C | 32 | All bits | √ | √ | √ | √ |
| Equal pulse width register n | EQWRn | R/W | H'0060 | 32 | All bits | √ | √ | √ | — |
| Serration width register n | SPWRn | R/W | H'0064 | 32 | All bits | √ | √ | √ | — |
| CLAMP signal start register n | CLAMPSRn | R/W | H'0070 | 32 | All bits | √ | √ | √ | √ |
| CLAMP signal width register n | CLAMPWRn | R/W | H'0074 | 32 | All bits | √ | √ | √ | √ |
| DE signal start register n | DESRn | R/W | H'0078 | 32 | All bits | √ | √ | √ | √ |
| DE signal width register n | DEWRn | R/W | H'007C | 32 | All bits | √ | √ | √ | √ |

Table 36.11 Display Timing Generation Register Configuration (2)

| Register Name | Abbr. | Power-on Reset | Second Generation RZ/G Series Products | | | | |
|-------------------------------------|----------|----------------|---|----------------------------|--------|--------|--|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Horizontal display start register n | HDSRn | H'****_**** | √ | √ | √ | √ | |
| Horizontal display end register n | HDERn | H'****_**** | √ | √ | √ | √ | |
| Vertical display start register n | VDSRn | H'****_**** | √ | √ | √ | √ | |
| Vertical display end register n | VDERn | H'****_**** | √ | √ | √ | √ | |
| Horizontal cycle register n | HCRn | H'****_**** | √ | √ | √ | √ | |
| Horizontal sync width register n | HSWRn | H'****_**** | √ | √ | √ | √ | |
| Vertical cycle register n | VCRn | H'****_**** | √ | √ | √ | √ | |
| Vertical sync point register n | VSPRn | H'****_**** | √ | √ | √ | √ | |
| Equal pulse width register n | EQWRn | H'****_**** | √ | √ | √ | — | |
| Serration width register n | SPWRn | H'****_**** | √ | √ | √ | — | |
| CLAMP signal start register n | CLAMPSRn | H'****_**** | √ | √ | √ | √ | |
| CLAMP signal width register n | CLAMPWRn | H'****_**** | √ | √ | √ | √ | |
| DE signal start register n | DESRn | H'****_**** | √ | √ | √ | √ | |
| DE signal width register n | DEWRn | H'****_**** | √ | √ | √ | √ | |

(c) Display Attribute Register Configuration**Table 36.12 Display Attribute Register Configuration (1)**

Base address: DU0: H'FEB0_0000 (suffix 0)
 DU1: H'FEB3_0000 (suffix 1)
 DU2: H'FEB4_0000 (suffix 2) [RZ/G2M V1.3, RZ/G2M V3.0]
 DU3: H'FEB7_0000 (suffix 3) [RZ/G2H, RZ/G2N]

| Register Name | Abbr. | R/W | Offset Address | Access Size | Bit with Internal Update Function* | Second Generation RZ/G Series Products | | | |
|------------------------------------|-----------|-----|----------------|-------------|------------------------------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Display off mode output register n | DOORn | R/W | H'0090 | 32 | All bits | √ | √ | √ | √ |
| Color detection register n | CDERn | R/W | H'0094 | 32 | All bits | √ | √ | √ | √ |
| Background plane output register n | BPORn | R/W | H'0098 | 32 | All bits | √ | √ | √ | √ |
| Raster interrupt offset register n | RINTOFSRn | R/W | H'009C | 32 | All bits | √ | √ | √ | √ |

Note: The register available code is excluded. There is no internal update function for the register available code.

Table 36.13 Display Attribute Register Configuration (2)

| Register Name | Abbr. | Power-on Reset | Second Generation RZ/G Series Products | | | |
|------------------------------------|-----------|----------------|--|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Display off mode output register n | DOORn | H'****_**** | √ | √ | √ | √ |
| Color detection register n | CDERn | H'****_**** | √ | √ | √ | √ |
| Background plane output register n | BPORn | H'****_**** | √ | √ | √ | √ |
| Raster interrupt offset register n | RINTOFSRn | H'****_**** | √ | √ | √ | √ |

(d) Display Plane Register Configuration**Table 36.14 Display Plane Register Configuration (1)**

Base address: DU0/DU1: H'FEB0_0000 (suffix 0) [DU1 is for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

DU2/DU3: H'FEB4_0000 (suffix 2) [DU2 is for RZ/G2M V1.3, RZ/G2M V3.0, DU3 is only for RZ/G2H, RZ/G2N]

| | | | | | | Second Generation RZ/G Series Products | | | |
|---|----------|-----|----------------|-------------|------------------------------------|---|----------------------------|--------|--------|
| Register Name | Abbr. | R/W | Offset Address | Access Size | Bit with Internal Update Function* | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Plane 1 mode register m | P1MRm | R/W | H'0100 | 32 | All bits | √ | √ | √ | √ |
| Plane 1 display size X register m | P1DSXRm | R/W | H'0110 | 32 | All bits | √ | √ | √ | √ |
| Plane 1 display size Y register m | P1DSYRm | R/W | H'0114 | 32 | All bits | √ | √ | √ | √ |
| Plane 1 display position X register m | P1DPXRm | R/W | H'0118 | 32 | All bits | √ | √ | √ | √ |
| Plane 1 display position Y register m | P1DPYRm | R/W | H'011C | 32 | All bits | √ | √ | √ | √ |
| Plane 1 display data control 4 register m | P1DDC4Rm | R/W | H'0190 | 32 | All bits | √ | √ | √ | √ |
| Plane 3 mode register m | P3MR0 | R/W | H'0300 | 32 | All bits | √ | √ | √ | √ |
| | P3MR2 | | | | | √ | — | √ | — |
| Plane 3 display size X register m | P3DSXR0 | R/W | H'0310 | 32 | All bits | √ | √ | √ | √ |
| | P3DSXR2 | | | | | √ | — | √ | — |
| Plane 3 display size Y register m | P3DSYR0 | R/W | H'0314 | 32 | All bits | √ | √ | √ | √ |
| | P3DSYR2 | | | | | √ | — | √ | — |
| Plane 3 display position X register m | P3DPXR0 | R/W | H'0318 | 32 | All bits | √ | √ | √ | √ |
| | P3DPXR2 | | | | | √ | — | √ | — |
| Plane 3 display position Y register m | P3DPYR0 | R/W | H'031C | 32 | All bits | √ | √ | √ | √ |
| | P3DPYR2 | | | | | √ | — | √ | — |
| Plane 3 display data control 4 register m | P3DDC4R0 | R/W | H'0390 | 32 | All bits | √ | √ | √ | √ |
| | P3DDC4R2 | | | | | √ | — | √ | — |

Note: * The register available code is excluded. There is no internal update function for the register available code.

Table 36.15 Display Plane Register Configuration (2)

| Register Name | Abbr. | Power-on Reset | Second Generation RZ/G Series Products | | | |
|---|----------|----------------|---|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Plane 1 mode register m | P1MRm | H'****_**** | √ | √ | √ | √ |
| Plane 1 display size X register m | P1DSXRm | H'****_**** | √ | √ | √ | √ |
| Plane 1 display size Y register m | P1DSYRm | H'****_**** | √ | √ | √ | √ |
| Plane 1 display position X register m | P1DPXRm | H'****_**** | √ | √ | √ | √ |
| Plane 1 display position Y register m | P1DPYRm | H'****_**** | √ | √ | √ | √ |
| Plane 1 display data control 4 register m | P1DDC4Rm | H'****_**** | √ | √ | √ | √ |
| Plane 3 mode register m | P3MR0 | H'****_**** | √ | √ | √ | √ |
| | P3MR2 | | √ | — | √ | — |
| Plane 3 display size X register m | P3DSXR0 | H'****_**** | √ | √ | √ | √ |
| | P3DSXR2 | | √ | — | √ | — |
| Plane 3 display size Y register m | P3DSYR0 | H'****_**** | √ | √ | √ | √ |
| | P3DSYR2 | | √ | — | √ | — |
| Plane 3 display position X register m | P3DPXR0 | H'****_**** | √ | √ | √ | √ |
| | P3DPXR2 | | √ | — | √ | — |
| Plane 3 display position Y register m | P3DPYR0 | H'****_**** | √ | √ | √ | √ |
| | P3DPYR2 | | √ | — | √ | — |
| Plane 3 display data control 4 register m | P3DDC4R0 | H'****_**** | √ | √ | √ | √ |
| | P3DDC4R2 | | √ | — | √ | — |

(e) External Synchronization Control Register Configuration**Table 36.16 External Synchronization Control Register Configuration (1)**

Base address: DU0: H'FEB1_0000 (suffix 0)
 DU1: H'FEB3_1000 (suffix 1)
 DU2: H'FEB5_0000 (suffix 2) [RZ/G2M V1.3, RZ/G2M V3.0]
 DU3: H'FEB7_1000 (suffix 3) [RZ/G2H, RZ/G2N]

| Register Name | Abbr. | R/W | Offset Address | Access Size | Bit with Internal Update Function* | Second Generation RZ/G Series Products | | | |
|---|-------|-----|----------------|-------------|------------------------------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| External synchronization control register n | ESCR0 | R/W | H'0000 | 32 | None | √ | √ | √ | √ |
| | ESCR1 | | | | | — | — | — | √ |
| | ESCR2 | | | | | — | √ | — | — |
| | ESCR3 | | | | | √ | — | √ | — |
| Output signal timing adjustment register n | OTAR0 | R/W | H'0004 | 32 | All bits updated with DRES | — | — | — | √ |
| | OTAR1 | | | | | — | — | — | √ |
| | OTAR2 | | | | | — | √ | — | — |
| | OTAR3 | | | | | √ | — | √ | — |

Table 36.17 External Synchronization Control Register Configuration (2)

| Register Name | Abbr. | Power-on Reset | Second Generation RZ/G Series Products | | | |
|---|-------|----------------|--|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| External synchronization control register n | ESCR0 | H'****_***0 | √ | √ | √ | √ |
| | ESCR1 | | — | — | — | √ |
| | ESCR2 | | — | √ | — | — |
| | ESCR3 | | √ | — | √ | — |
| Output signal timing adjustment register n | OTAR0 | H'****_**** | — | — | — | √ |
| | OTAR1 | | — | — | — | √ |
| | OTAR2 | | — | √ | — | — |
| | OTAR3 | | √ | — | √ | — |

(f) Dual Display Output Control Register Configuration**Table 36.18 Dual Display Output Control Register Configuration (1)**

Base address: DU0/DU1: H'FEB1_0000 (suffix 0)

DU2/DU3: H'FEB5_0000 (suffix 2) [DU2 is for RZ/G2M V1.3, RZ/G2M V3.0, DU3 is only for RZ/G2H, RZ/G2N]

| Register Name | Abbr. | R/W | Offset Address | Access Size | Bit with Internal Update Function | Second Generation RZ/G Series Products | | | |
|--|--------|-----|----------------|-------------|-----------------------------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Display unit output route control register m | DORCRm | R/W | H'1000 | 32 | All bits updated with DRES | √ | √ | √ | √ |
| Display unit plane timing select register m | DPTSR0 | R/W | H'1004 | 32 | All bits updated with DRES | √ | √ | √ | √ |
| | DPTSR2 | | | | | √ | — | √ | — |
| Display Superimpose Priority Register 0 | DSPR0 | R/W | H'1020 | 32 | All bits | √ | √ | √ | √ |
| Display Superimpose Priority Register 1 | DSPR1 | R/W | H'1024 | 32 | All bits | √ | √ | √ | √ |
| Display Superimpose Priority Register 2 | DSPR2 | R/W | H'1020 | 32 | All bits | — | √ | — | — |
| Display Superimpose Priority Register 3 | DSPR3 | R/W | H'1024 | 32 | All bits | √ | — | √ | — |

Table 36.19 Dual Display Output Control Register Configuration (2)

| Register Name | Abbr. | Power-on Reset | Second Generation RZ/G Series Products | | | |
|--|--------|----------------|---|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Display unit output route control register m | DORCRm | H'****_**** | √ | √ | √ | √ |
| Display unit plane timing select register m | DPTSR0 | H'****_**** | √ | √ | √ | √ |
| | DPTSR2 | | √ | — | √ | — |
| Display Superimpose Priority Register 0 | DSPR0 | H'****_**00 | √ | √ | √ | √ |
| Display Superimpose Priority Register 1 | DSPR1 | H'****_**00 | √ | √ | √ | √ |
| Display Superimpose Priority Register 2 | DSPR2 | H'****_***0 | — | √ | — | — |
| Display Superimpose Priority Register 3 | DSPR3 | H'****_***0 | √ | — | √ | — |

(g) YC-RGB Conversion Coefficient Register Configuration**Table 36.20 YC-RGB Conversion Coefficient Register Configuration (1)**

Base address: DU0/DU1: H'FEB1_4000 (suffix 0)

DU2/DU3: H'FEB5_4000 (suffix 2) [DU2 is for RZ/G2M V1.3, RZ/G2M V3.0, DU3 is only for RZ/G2H, RZ/G2N]

| Register Name | Abbr. | R/W | Offset Address | Access Size | Bit with Internal Update Function | Second Generation RZ/G Series Products | | | |
|--|--------|-----|----------------|-------------|-----------------------------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Y normalization coefficient register m | YNCRm | R/W | H'80 | 32 | All bits | √ | √ | √ | √ |
| Y normalization offset register m | YNORm | R/W | H'84 | 32 | All bits | √ | √ | √ | √ |
| Cr normalization offset register m | CRNORm | R/W | H'88 | 32 | All bits | √ | √ | √ | √ |
| Cb normalization offset register m | CBNORm | R/W | H'8C | 32 | All bits | √ | √ | √ | √ |
| Red Cr coefficient register m | RCRCRm | R/W | H'90 | 32 | All bits | √ | √ | √ | √ |
| Green Cr coefficient register m | GCRCRm | R/W | H'94 | 32 | All bits | √ | √ | √ | √ |
| Green Cb coefficient register m | GCBCRm | R/W | H'98 | 32 | All bits | √ | √ | √ | √ |
| Blue Cb coefficient register m | BCBCRm | R/W | H'9C | 32 | All bits | √ | √ | √ | √ |

Table 36.21 YC-RGB Conversion Coefficient Register Configuration (2)

| Register Name | Abbr. | Power-on Reset | Second Generation RZ/G Series Products | | | |
|--|--------|----------------|---|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Y normalization coefficient register m | YNCRm | H'*800_*800 | √ | √ | √ | √ |
| Y normalization offset register m | YNORm | H'**00_**00 | √ | √ | √ | √ |
| Cr normalization offset register m | CRNORm | H'**80_**80 | √ | √ | √ | √ |
| Cb normalization offset register m | CBNORm | H'**80_**80 | √ | √ | √ | √ |
| Red Cr coefficient register m | RCRCRm | H'*AF0_*AF0 | √ | √ | √ | √ |
| Green Cr coefficient register m | GCRCRm | H'*590_*590 | √ | √ | √ | √ |
| Green Cb coefficient register m | GCBCRm | H'*2B0_*2B0 | √ | √ | √ | √ |
| Blue Cb coefficient register m | BCBCRm | H'*DE0_*DE0 | √ | √ | √ | √ |

36.1.5 Connected Module

Table 36.22 Connected Module

| Module name | Connected module name | Function of connected module | Second Generation RZ/G Series Products | | | | |
|-------------|-----------------------|--|--|----------------------------|--------|--------|--|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| DU | DPLL | Output dot clock | √ | √ | √ | — | |
| | APB | Control to register access | √ | √ | √ | √ | |
| | VSP | Output image data | √ | √ | √ | √ | |
| | LVDS-IF | Input display data and control signals, after that output LVDS signals | √ | √ | √ | √ | |
| | HDMI-IF | Input display data and control signals, after that output HDMI signals | √ | √ | √ | — | |
| | DPAD | Input display data and control signals, after that output digital RGB and YC format data | √ | √ | √ | √ | |
| | CPG | Output system clocks and dot clock. | √ | √ | √ | √ | |
| | PFC | Select the external pins | √ | √ | √ | √ | |
| | Module Standby | Control to stop clocks | √ | √ | √ | √ | |
| | Software Reset | Execute software reset | √ | √ | √ | √ | |

36.2 Register Description

Legend for Register Description

Initial value: Register value after a reset.

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R: Read-only. The write value should always be 0.

R1: Read-only. The write value should always be 1.

—/W: Write-only. The read value is undefined.

Plane p: Indicates plane 1 and plane 3

Note: Do not access locations that appear empty in the address map. Operation is not guaranteed in case of access to change the value at any location other than those of the registers listed in Table 36.8 to Table 36.21.

36.2.1 Display Unit System Control Registers

(1) Display Unit System Control Register n (DSYSRn)

Note: n = 0 and 2 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 n = 0 [RZ/G2E]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0: H'FEB0_0000, DU2: H'FEB4_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|--------------|----|----|----|------------|-----------|-----|-----|-----|-----|----|----|----|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | ILTS/ —*1 | — | — | — | — | — | — | — | — | — | — | — | — | IUPD *2 |
| Initial value: | — | — | 0/—*1 | — | — | — | — | — | — | — | — | — | — | — | — | 0 |
| R/W: | R | R | R/W*1 | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | DRES *2 | DEN *2 | TVM | | SCM | | — | — | — | — |
| Initial value: | — | — | — | — | — | — | 1 | 0 | 1 | 0 | 0 | 0 | — | — | — | — |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R |

- Notes: 1. For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.
 2. Bits 16, 9, and 8 in DSYSR0 are shared by the DU0 and DU1. For RZ/G2H, bits 16, 9, and 8 in DSYSR2 are shared by the DU2 and DU3. For RZ/G2H, RZ/G2N, bits 16, 9, and 8 in DSYSR2 are shared by the DU3.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31, 30 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 29 | ILTS | B'0 | R/W | Not available | Input Pad Latch Timing Select [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] To enable this bit, the DEFE bit in DEFRn should be set to 1. In the initial state, this bit is fixed to 0. 0: A signal of the input pad is latched at the DCLKIN rising edge. 1: A signal of the input pad is latched at the DCLKIN falling edge. Electrical characteristics do not apply. |
| | — | — | R | — | Reserved [RZ/G2E] The read value is undefined. The write value should always be 0. |
| 28 to 17 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 16 | IUPD | B'0 | R/W | Not available | Internal Updating Disable When DRES = 1, internal register updating occurs regardless of this bit setting. For details on internal updating, see section 36.1.4(2), Internal Update. 0: Internal register updating occurs for each vertical synchronous signal (VSYNC) assertion. 1: If this bit is set to 1, internal register updating does not occur. <ul style="list-style-type: none"> If this bit is set to 0, register updating occurs according to the next vertical synchronous signal (VSYNC). This bit is shared by the DU0 and DU1. For , this bit is shared by the DU2 and DU3. For RZ/G2N, RZ/G2H, this bit is shared by the DU3. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 15 to 10 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 9 | DRES | B'1 | R/W | Not available | Display Reset |
| 8 | DEN | B'0 | R/W | Available | Display Enable B'00: Display synchronous operation starts. If any register has not been set yet, the DU may perform unexpected operation, thus set DRES to 0 after setting all registers in the DU. For DEN = 0, the display data becomes values set with the DOORn. B'01: Display synchronous operation starts. If any register has not been set yet, the DU may perform unexpected operation, thus set DRES = 0, DEN = 1 after setting all registers in the DU. To re-set DRES to 1 following the start of operations for synchronization of the display with this setting, proceed with steps 1 and 2 in section 36.4.2, Transition to Module Standby Mode before doing so. For DEN = 1, the display data becomes values stored on the unified memory from the next frame. B'10: Display operation stops.[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Display operation and synchronization operation stop. The set values except for the following bits in DSSRn are retained. In this setting, the following operations occur: 1. All 0s are output as display data. 2. The following bits of DSSRn are cleared to 0: TV synchronization signal error flag (TVR) Frame flag (FRM) Vertical blanking flag (VBK) Raster interrupt flag (RINT) Horizontal blanking flag (HBK) 3. HSYNC, VSYNC, and ODDF pins function as inputs. However, the ODDF pin functions as a CLAMP output when the ODPM bit in DSMRn is 1. B'10: Display operation stops. [RZ/G2E] Display operation and synchronization operation stop. The set values except for the following bits in DSSRn are retained. In this setting, the following operations occur: 1. All 0s are output as display data. 2. The following bits of DSSRn are cleared to 0: Vertical blanking flag (VBK) Raster interrupt flag (RINT) Horizontal blanking flag (HBK) B'11: Setting prohibited <ul style="list-style-type: none"> This bit is shared by the DU0 and DU1. For , this bit is shared by the DU2 and DU3. For RZ/G2N, RZ/G2H, this bit is shared by the DU3. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|---|
| 7, 6 | TVM | B'10 | R/W | Not available | <p>TV Synchronized Mode [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>B'00: Master Mode HSYNC, VSYNC, and CSYNC are output.</p> <p>B'01: Sync Method Switching Mode. Use this mode if necessary when switching from TV synchronized mode to master mode or vice versa is performed. In this mode, display operation forcibly stops and the DISP pin outputs the low level. Moreover, clock supply to DCLKIN can be stopped (input invalid) (the inside of the LSI is fixed to the high level). The EXHSYNC, EXVSYNC, and EXODDF pins are used as input pins.</p> <p>B'10: TV Sync Mode EXHSYNC, EXVSYNC, and EXODDF are input. When the ODPM bit in DSMRn* is set to 1, the ODDF pin functions as an output pin.</p> <p>B'11: Setting prohibited</p> <hr/> <p>TV Synchronized Mode [RZ/G2E]</p> <p>B'00: Master Mode others: Setting prohibited</p> |
| 5, 4 | SCM | B'00 | R/W | Not available | <p>Scan Mode [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>B'00: Non-interlaced mode B'01: Setting prohibited B'10: Interlace sync mode B'11: Interlace sync & video mode</p> <hr/> <p>Scan Mode [RZ/G2E]</p> <p>B'00: Non-interlaced mode others: Setting prohibited</p> |
| 3 to 0 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |

Note: * n = 0 and 2 [RZ/G2M V1.3, RZ/G2M V3.0]
n = 0 [RZ/G2H, RZ/G2N, RZ/G2E]

(2) Display Unit System Control Register n (DSYSRn)

Note: n = 1 and 3 [RZ/G2H, RZ/G2N]
 n = 1 [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU1: H'FEB3_0000, DU3: H'FEB7_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|--------------|----|----|----|----|----|-----|------|------|-----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | ILTS1/ —* | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | 0/—* | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R/W* | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | TVM1 | SCM1 | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | 1 | 0 | 0 | 0 | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R |

Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|---------|----------|---------------|-----|-----------------|--|
| 31, 30 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 29 | ILTS1 | B'0 | R/W | Not available | Input Pad Latch Timing Select 1 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] To enable this bit, the DEFE bit in DEFRn should be set to 1. In the initial state, this bit is fixed to 0. 0: A signal of the input pad is latched at the DCLKIN rising edge. 1: A signal of the input pad is latched at the DCLKIN falling edge. Electrical characteristics do not apply. |
| | — | — | R | — | Reserved [RZ/G2E] The read value is undefined. The write value should always be 0. |
| 28 to 8 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|--|
| 7, 6 | TVM1 | B'10 | R/W | Not available | <p>TV Synchronized Mode 1 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>B'00: Master Mode HSYNC, VSYNC, and CSYNC are output.</p> <p>B'01: Sync Method Switching Mode. Use this mode if necessary when switching from TV synchronized mode to master mode or vice versa is performed.</p> <p>In this mode, display operation forcibly stops and the DISP pin outputs the low level. Moreover, clock supply to DCLKIN can be stopped (input invalid) (the inside of the LSI is fixed to the high level). The EXHSYNC, EXVSYNC, and EXODDF pins are used as input pins.</p> <p>B'10: TV Sync Mode EXHSYNC, EXVSYNC, and EXODDF are input. When the ODPM bit in DSMRn* is set to 1, the ODDF pin functions as an output pin.</p> <p>B'11: Setting prohibited</p> <hr/> <p>TV Synchronized Mode 1[RZ/G2E]</p> <p>00: Master Mode other: Setting prohibited</p> |
| 5, 4 | SCM1 | B'00 | R/W | Not available | <p>Scan Mode 1[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>B'00: Non-interlaced mode B'01: Setting prohibited B'10: Interlace sync mode B'11: Interlace sync & video mode</p> <hr/> <p>Scan Mode 1 [RZ/G2E]</p> <p>00: Non-interlaced mode other: Setting prohibited</p> |
| 3 to 0 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |

Note: * n = 1 and 3 [RZ/G2H, RZ/G2N]
n = 1 [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]

(3) Display Unit Mode Register n (DSMRn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0: H'FEB0_0004, DU1: H'FEB3_0004, DU2: H'FEB4_0004, DU3: H'FEB7_0004

| | | | | | | | | | | | | | | | | |
|----------------|------|------|-----|-------------|------|------|-----|-------------|------------|------|----|----|-----|-----|-----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | VSPM/ —* | ODPM | DIPM | | CSPM | — | — | — | — | DIL | VSL | HSL | DDIS |
| Initial value: | — | — | — | 0/—* | 0 | 0 | 0 | 0 | — | — | — | — | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W* | R/W | R/W | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CDEL | CDEM | | CDED | — | — | — | ODEV/ —* | CSY/ —* | | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | — | — | — | 0/—* | 0/—* | 0/—* | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R | R | R | R/W* | R/W* | R/W* | R | R | R | R | R | R |

Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.

For RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, all bit in this register setting is also available for the display data output via the LVDS-IF. Bit 24, 18 and 17 in this register setting is also available for the display data output via the HDMI-IF.

For RZ/G2E, all bit in this register setting is also available for the display data output via the LVDS-IF.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 29 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 28 | VSPM | B'0 | R/W | DRES | VSYNC Pin Mode [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: The VSYNC signal is output to the VSYNC pin. 1: The CSYNC signal is output to the VSYNC pin. In case of the output via the DPAD, "VSYNC pin" refers to the DU_VSYNC/DU_EXVSYNC pin indicated in Table 36.3 and Table 36.4. In case of the output via the LVDS-IF, "VSYNC pin" connected with the VSYNC signal of the LVDS-IF. |
| | — | — | R | — | Reserved [RZ/G2E] The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|---|
| 27 | ODPM | B'0 | R/W | DRES | <p>ODDF Pin Mode [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>0: The ODDF signal is output to the ODDF pin. 1: The CLAMP signal is output to the ODDF pin. Even if bits 7 and 6 in DSYSRn indicate TV synchronized mode, the ODDF pin becomes an output. In case of the output via the DPAD, "ODDF pin" refers to the DU_ODDF/DU_EXODDF pin indicated in Table 36.3 and Table 36.4. In case of the output via the LVDS-IF, "ODDF pin" connected with the Odd/even signal of the LVDS-IF.</p> <hr/> <p>DU_DISP_CDE Pin Mode [RZ/G2E]</p> <p>0: The DU_DISP_CDE pin is fixed to 0. 1: The CLAMP signal is output to the DU_DISP_CDE pin. In case of the output via the DPAD, "DU_DISP_CDE pin" refers to the DU_DISP_CDE pin indicated in Table 36.5. In case of the output via the LVDS-IF, "DU_DISP_CDE pin" connected with the Odd/even signal of LVDS-IF.</p> |
| 26, 25 | DIPM | B'00 | R/W | DRES | <p>DISP Pin Mode[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>B'00: The DISP signal is output to the DISP pin. B'01: The CSYNC signal is output to the DISP pin. B'10: Setting prohibited. (Fixed to 0.) B'11: The DE signal is output to the DISP pin. In case of the output via the DPAD, "DISP pin" refers to the DU_DISP pin indicated in Table 36.3 and Table 36.4. In case of the output via the LVDS-IF, "DISP pin" connected with the DISP signal of the LVDS-IF.</p> <hr/> <p>DISP Pin Mode [RZ/G2E]</p> <p>B'00: The DISP signal is output to the DISP pin. B'01: Setting prohibited. B'10: Setting prohibited. (Fixed to 0.) B'11: The DE signal is output to the DISP pin. In case of the output via the DPAD, "DISP pin" refers to the DU_DISP pin indicated in Table 36.5. In case of the output via the LVDS-IF, "DISP pin" connected with the DISP signal of the LVDS-IF.</p> |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 24 | CSPM | B'0 | R/W | DRES | <p>CSYNC Pin Mode [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>This description is the specification in case of the output via the DPAD and the LVDS-IF.</p> <p>0: The CSYNC signal is output to the CSYNC pin. 1: The HSYNC signal is output to the CSYNC pin.</p> <p>In case of the output via the DPAD, "CSYNC pin" refers to the DU_HSYNC/DU_EXHSYNC pin indicated in Table 36.3 and Table 36.4.</p> <p>In case of the output via the LVDS-IF, "CSYNC pin" connected with the HSYNC signal of the LVDS-IF.</p> <hr/> <p>CSYNC Pin Mode [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>This description is the specification in case of the output via the HDMI-IF.</p> <p>0: Setting prohibited. (Initial value) 1: The HSYNC signal is output to the HSYNC pin.</p> <p>"HSYNC pin" connected with the HSYNC signal of the HDMI-IF.</p> <hr/> <p>HSYNC Pin Mode [RZ/G2E]</p> <p>0: Setting prohibited. (Initial value) 1: The HSYNC signal is output to the HSYNC pin.</p> <p>In case of the output via the DPAD, "HSYNC pin" refers to the DU_HSYNC pin indicated in Table 36.5.</p> <p>In case of the output via the LVDS-IF, "HSYNC pin" connected with the HSYNC signal of the LVDS-IF.</p> |
| 23 to 20 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 19 | DIL | B'0 | R/W | DRES | <p>DISP Polarity Select</p> <p>0: The DISP signal becomes a high level during display period. 1: The polarity of the DISP signal is inverted.</p> |
| 18 | VSL | B'0 | R/W | DRES | <p>VSYNC Polarity Select</p> <p>0: The VSYNC signal becomes active low. 1: The polarity of VSYNC is inverted.</p> |
| 17 | HSL | B'0 | R/W | DRES | <p>HSYNC Polarity Select</p> <p>0: The HSYNC signal becomes active low. 1: The polarity of the HSYNC signal is inverted.</p> |
| 16 | DDIS | B'0 | R/W | Available | <p>DISP Disable</p> <p>0: The DISP signal is output. 1: The DISP signal is not output.</p> |
| 15 | CDEL | B'0 | R/W | Available | <p>CDE Polarity Select</p> <p>0: The CDE signal becomes high when the CDERn match. 1: The polarity of the CDE signal is inverted.</p> |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|---------|----------|---------------|-----|-----------------|---|
| 14, 13 | CDEM | B'00 | R/W | Available | CDE Output Mode B'00: The CDE signal is output as is. B'01: The CDE signal is output as is. B'10: The low level is output outside the display period. B'11: The high level is output outside the display period. |
| 12 | CDED | B'0 | R/W | Available | CDE Disable 0: The CDE signal is output. 1: The CDE signal output is prohibited. |
| 11 to 9 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 8 | ODEV | B'0 | R/W | Available | Odd Even Location for ODDF Signal [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: The odd field (ODDF = low) is located in the first half of the same frame of the interlace display. 1: The odd field (ODDF = low) is located in the second half of the same frame of the interlace display. |
| | — | — | R | — | Reserved [RZ/G2E] The read value is undefined. The write value should always be 0. |
| 7, 6 | CSY | B'00 | R/W | Available | CSYNC Mode [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] B'00: A waveform obtained by performing exclusive logical OR for VSYNC and HSYNC is output as CSYNC. B'01: Setting prohibited B'10: During intervals of 3 rasters from the VSYNC falling edge, the equalizing pulse is output. After that, during intervals of 3 rasters, the serration pulse is output. After that, during intervals of 3 rasters, the equalizing pulse is output. During other intervals, the HSYNC waveform is output as CSYNC. B'11: After 1/2 rasters from the VSYNC falling edge, the equalizing pulse is output during intervals of 2.5 rasters. After that, during intervals of 2.5 rasters, the serration pulse is output. After that, during intervals of 2.5 rasters, the equalizing pulse is output. During other intervals, the HSYNC waveform is output as CSYNC. |
| | — | — | R | — | Reserved [RZ/G2E] The read value is undefined. The write value should always be 0. |
| 5 to 0 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

(4) Display Unit Status Register n (DSSRn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0: H'FEB0_0008, DU1: H'FEB3_0008, DU2: H'FEB4_0008, DU3: H'FEB7_0008

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|----|----|-----|----|------|-----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TVR/* | FRM/* | — | — | VBK | — | RINT | HBK | — | — | — | — | — | — | — | — |
| Initial value: | 0/* | 0/* | — | — | 0 | — | 0 | 0 | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 16 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 15 | TVR | B'0 | R | Not available | TV Synchronization Error Flag [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: Indicates that, after the TVR bit is cleared with the DRES bit in DSYSRn or the TVCL bit in DSRCRn, the EXVSYNC rising edge is detected every time within the vertical cycle determined with the setting of the VCRn. 1: Indicates that, in TV synchronized mode, the EXVSYNC rising edge was not detected within the vertical cycle determined with the setting of the VCRn. The TVR bit retains its status until it is cleared with the DRES or TVCL bit. |
| | — | — | R | — | Reserved [RZ/G2E] The read value is undefined. The write value should always be 0. |
| 14 | FRM | B'0 | R | Not available | Frame Flag [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: Indicates the period from the time when the FRM bit is cleared by the DRES bit in DSYSRn or by the FRCL bit in DSRCRn to the time when display of the next field is completed in the non-interlaced mode, or when display of the next even field is completed in interlace sync mode or interlace sync & video mode. 1: Indicates the period (in units of frames) from the start of the vertical blanking period of the first even field after the FRM bit is cleared by the DRES or FRCL bit, to the time when the FRM bit is cleared again. |
| | — | — | R | — | Reserved [RZ/G2E] The read value is undefined. The write value should always be 0. |
| 13, 12 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|---|
| 11 | VBK | B'0 | R | Not available | <p>Vertical Blanking Flag</p> <p>0: Indicates the period from the time when the VBK bit is cleared by the DRES bit in DSYSRn or by the VBCL bit in DSRCRn to the time when display of the next field is completed.</p> <p>1: Indicates the period (in units of fields) from the start of the first vertical blanking period after the VBK bit is cleared by the DRES or VBCL bit, to the time when the VBK bit is cleared again.</p> |
| 10 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 9 | RINT | B'0 | R | Not available | <p>Raster Interrupt Flag</p> <p>0: Indicates the period from the time when the RINT bit is cleared by the DRES bit in DSYSRn or by the RICL bit in DSRCRn to the time when the period of the rasters set in the RINTOFSRn has elapsed after the beginning of the next field display.</p> <p>1: Indicates the period from the time when the period of the RINTOFSRn has elapsed after the beginning of the next field display after the RINT bit is cleared by the DRES or RICL bit, to the time when the RINT bit is cleared again.</p> |
| 8 | HBK | B'0 | R | Not available | <p>Horizontal Blanking Flag</p> <p>0: Indicates the period from the time when the HBK bit is cleared by the DRES bit in DSYSRn or by the HBCL bit in DSRCRn to the time of the next HSYNC assertion.</p> <p>1: Indicates the period from the start of the first horizontal blanking period after the HBK bit is cleared by the DRES or HBCL bit, to the time when the HBK bit is cleared again.</p> |
| 7 to 0 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |

(5) Display Unit Status Register Clear Register n (DSRCRn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0: H'FEB0_000C, DU1: H'FEB3_000C, DU2: H'FEB4_000C, DU3: H'FEB7_000C

| | | | | | | | | | | | | | | | | |
|----------------|-------------|-------------|----|----|------|----|------|------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TVCL/ —* | FRCL/ —* | — | — | VBCL | — | RICL | HBCL | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | —/W | —/W | R | R | —/W | R | —/W | —/W | R | R | R | R | R | R | R | R |

Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 16 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 15 | TVCL | — | —/W | Not available | TV Synchronization Signal Error Flag Clear [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: Does not change the TVR flag in DSSRn. 1: Clears the TVR flag in DSSRn. |
| | — | — | R | — | Reserved [RZ/G2E] The read value is undefined. The write value should always be 0. |
| 14 | FRCL | — | —/W | Not available | Frame Flag Clear [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: Does not change the FRM flag in DSSRn. 1: Clears the FRM flag in DSSRn. |
| | — | — | R | — | Reserved [RZ/G2E] The read value is undefined. The write value should always be 0. |
| 13, 12 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 11 | VBCL | — | —/W | Not available | Vertical Blanking Flag Clear 0: Does not change the VBK flag in DSSRn. 1: Clears the VBK flag in DSSRn. |
| 10 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 9 | RICL | — | —/W | Not available | Raster Interrupt flag clear 0: Does not change the RINT flag in DSSRn. 1: Clears the RINT flag in DSSRn. |
| 8 | HBCL | — | —/W | Not available | HBK flag clear 0: Does not change the HBK flag in DSSRn. 1: Clears the HBK flag in DSSRn. |
| 7 to 0 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

(6) Display Unit Interrupt Enable Register n (DIERn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0: H'FEB0_0010, DU1: H'FEB3_0010, DU2: H'FEB4_0010, DU3: H'FEB7_0010

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|----|----|-----|----|-----|-----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TVE/— * | FRE/— * | — | — | VBE | — | RIE | HBE | — | — | — | — | — | — | — | — |
| Initial value: | 0/—* | 0/—* | — | — | 0 | — | 0 | 0 | — | — | — | — | — | — | — | — |
| R/W: | R/W* | R/W* | R | R | R/W | R | R/W | R/W | R | R | R | R | R | R | R | R |

Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.

The DIERn allows the generation of interrupts for the CPU by aspects of the internal state of the DU as reflected in bits of DSSRn as the sources. When a bit of this register is set, setting of the bit in the same position within DSSRn will lead to the generation of an interrupt for the CPU.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 16 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 15 | TVE | B'0 | R/W | Not available | TV Synchronization Signal Error Interrupt Enable [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: Disables the TVR flag interrupt of DSSRn. 1: Enables the TVR flag interrupt of DSSRn. |
| | — | — | R | — | Reserved [RZ/G2E] The read value is undefined. The write value should always be 0. |
| 14 | FRE | B'0 | R/W | Not available | Frame Flag Interrupt Enable [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: Disables the FRM flag interrupt of DSSRn. 1: Enables the FRM flag interrupt of DSSRn. |
| | — | — | R | — | Reserved [RZ/G2E] The read value is undefined. The write value should always be 0. |
| 13, 12 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 11 | VBE | B'0 | R/W | Not available | Vertical Blanking Flag Interrupt Enable 0: Disables the VBK flag interrupt of DSSRn. 1: Enables the VBK flag interrupt of DSSRn. |
| 10 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|---|
| 9 | RIE | B'0 | R/W | Not available | Raster Interrupt Flag Interrupt Enable 0: Disables the RINT flag interrupt of DSSRn. 1: Enables the RINT flag interrupt of DSSRn. |
| 8 | HBE | B'0 | R/W | Not available | HBK Flag Interrupt Enable 0: Disables the HBK flag interrupt of DSSRn. 1: Enables the HBK flag interrupt of DSSRn. |
| 7 to 0 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

The following are conditions, based on DSSRn and DIERN, for issuing an interrupt to the CPU from the DUn.

Conditions for issuing an interrupt from each DU = a + b + c + d + e [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Conditions for issuing an interrupt from each DU = c + d + e [RZ/G2E]

- a = TVR & TVE
- b = FRM & FRE
- c = VBK & VBE
- d = RINT & RIE
- e = HBK & HBE

(7) Display Unit Domain 1 Status Register n (DD1SSRn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0: H'FEB2_0008, DU1: H'FEB2_8008, DU2: H'FEB6_0008, DU3: H'FEB6_8008

The specifications shown in section (4), Display Unit Status Register n (DSSRn), are also applied to this register.

(8) Display Unit Domain 1 Status Register Clear Register n (DD1SRCRn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0: H'FEB2_000C, DU1: H'FEB2_800C, DU2: H'FEB6_000C, DU3: H'FEB6_800C

The specifications shown in section (5), Display Unit Status Register Clear Register n (DSRCRn), are also applied to this register.

(9) Display Unit Domain 1 Interrupt Enable Register n (DD1IERn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0: H'FEB2_0010, DU1: H'FEB2_8010, DU2: H'FEB6_0010, DU3: H'FEB6_8010

The specifications shown in section (6), Display Unit Interrupt Enable Register n (DIERn), are also applied to this register.

(10) Display Plane Priority Register 0 (DPPR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB0_0018

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------------|---------|------|------|------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | DPE2/ —* | DPS2/—* | | DPE1 | DPS1 | | | |
| Initial value: | — | — | — | — | — | — | — | — | 0/—* | 0/—* | 0/—* | 1/—* | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W* | R/W* | R/W* | R/W* | R/W | R/W | R/W | R/W |

Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.

- Defines the order of planes in image superimposition and turns the display of planes on or off
- When the DPRS bit in DORCR0 is 0, this register is enabled, all plane display can be turn on.
- This register can only be used to set the priority order for superimposition processor 0. To use superimposition processor 1, set the DPRS bit in DORCR0 to 1 and make other settings as required in the DSPR1.
- After setting a desired value in a register listed in section 36.2.4 Display Plane Registers, set the corresponding bit in DPPR0 (bit 7 or 3) to 1.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|---------|----------|---------------|-----|-----------------|---|
| 31 to 8 | — | — | R | — | Reserved. The read value is undefined. The write value should always be 0. |
| 7 | DPE2 | B'0 | R/W | Available | Display Plane Priority 2 Enable |
| 6 to 4 | DPS2 | B'001 | R/W | Available | Display Plane Priority 2 Select B'1000: Assigns priority 2 to plane 1 and displays plane 1. B'1001: Setting prohibited. B'1010: Assigns priority 2 to plane 3 and displays plane 3. B'1011: Setting prohibited. B'1100: Setting prohibited. B'1101: Setting prohibited. B'1110: Setting prohibited. B'1111: Setting prohibited. Others: Priority 2 is not displayed. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|---|
| 3 | DPE1 | B'0 | R/W | Available | Display Plane Priority 1 Enable |
| 2 to 0 | DPS1 | B'000 | R/W | Available | Display Plane Priority 1 Select B'1000: Assigns priority 1 to plane 1 and displays plane 1. B'1001: Setting prohibited. B'1010: Assigns priority 1 to plane 3 and displays plane 3. B'1011: Setting prohibited. B'1100: Setting prohibited. B'1101: Setting prohibited. B'1110: Setting prohibited. B'1111: Setting prohibited. Others: Priority 1 is not displayed. |

(11) Display Plane Priority Register 2 (DPPR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | — | — |

Address: DU2/DU3: H'FEB4_0018

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|------|------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | DPE1 | DPS1 | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.

- Defines the order of planes in image superimposition and turns the display of planes on or off
- When the DPRS bit in DORCR2 is 0, this register is enabled, all plane display can be turn on.
- This register can only be used to set the priority order for superimposition processor 2. To use superimposition processor 3, set the DPRS bit in DORCR2 to 1 and make other settings as required in the DSPR3.
- After setting a desired value in a register listed in section 36.2.4 Display Plane Registers, set the corresponding bit in DPPR2 (bit 7 or 3) to 1.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|---------|----------|---------------|-----|-----------------|--|
| 31 to 8 | — | — | R | — | Reserved. The read value is undefined. The write value should always be 0. |
| 7 to 4 | — | — | R | — | Reserved. [RZ/G2M V1.3, RZ/G2M V3.0] The read value is undefined. The write value should always be 0. |

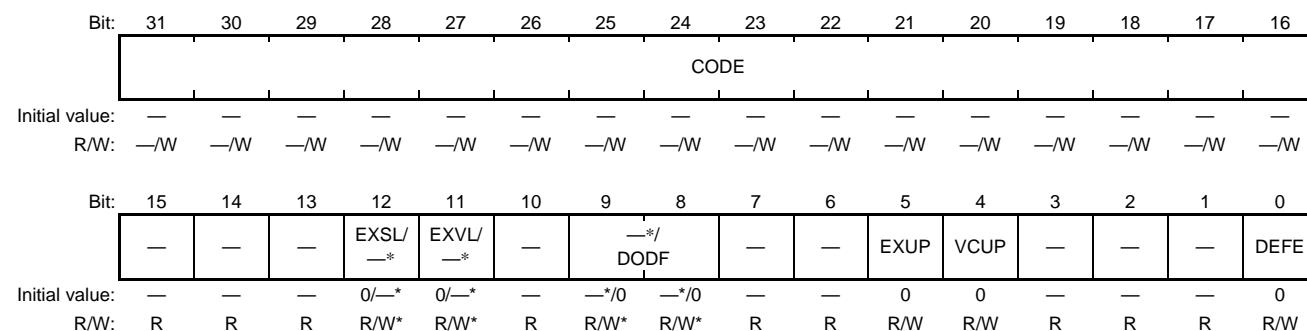
| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|---|
| 3 | DPE1 | B'0 | R/W | Available | Display Plane Priority 1 Enable [RZ/G2M V1.3, RZ/G2M V3.0] |
| 2 to 0 | DPS1 | B'000 | R/W | Available | Display Plane Priority 1 Select [RZ/G2M V1.3, RZ/G2M V3.0] B'1000: Assigns priority 1 to plane 1 and displays plane 1. B'1001: Setting prohibited. B'1010: Setting prohibited. B'1011: Setting prohibited. B'1100: Setting prohibited. B'1101: Setting prohibited. B'1110: Setting prohibited. B'1111: Setting prohibited. Others : Priority 1 is not displayed. |

(12) Display Unit Extensional Function Control Register n (DEFRn)

Note: n = 0 and 2 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 n = 0 [RZ/G2E]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0: H'FEB0_0020, DU2: H'FEB4_0020



Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 16 | CODE | — | —/W | Not available | DEFRn* Enabling Code (register available code) For a value written to DEFRn* to be effective, the value must include H'7773 in these bits. |
| 15 to 13 | — | — | R | — | Reserved. The read value is undefined. The write value should always be 0. |
| 12 | EXSL | B'0 | R/W | DRES | External Sync Signal Select [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: External SYNC signals (EXVSYNC, EXHSYNC) allow signals from the pins to be directly read in terms of post-division clocks. 1: External SYNC signals (EXVSYNC, EXHSYNC) allow the signals that were read in terms of pre-division clock to be read again as post-division clocks. For RZ/G2H, RZ/G2N, these bits are not provided in DEFR2. |
| | — | — | R | — | Reserved.[RZ/G2E] The read value is undefined. The write value should always be 0. |
| 11 | EXVL | B'0 | R/W | Not available | External Vsync Latch Select [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: External VSYNC signal (EXVSYNC) allows the signal from the pin to be latched every clock cycle. 1: External VSYNC signal (EXVSYNC) allows the signal from the pin to be latched at the rising edge of the external HSYNC signal. For RZ/G2H, RZ/G2N, these bits are not provided in DEFR2. |
| | — | — | R | — | Reserved.[RZ/G2E] The read value is undefined. The write value should always be 0. |
| 10 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|---|
| 9, 8 | — | — | R | — | Reserved [RZ/G2H, RZ/G2N] The read value is undefined. The write value should always be 0. |
| | DODF | B'00 | R/W | DRES | Display Output Data Format. [RZ/G2M V1.3, RZ/G2M V3.0] B'00: Sets the DU2 display output to RGB data format. B'01: Setting prohibited B'10: Sets the DU2 display output to non-multiplexed YC data format. Y (luminance) and C (chrominance) are output in parallel. B'11: Sets the DU2 display output to multiplexed YC data format. Y (luminance) and C (chrominance) are output in multiplexed format. For YC data format, set to 1 the YCDF0 and VSPF0 bits in DEF10R2. When YC format is displayed by timing as shown in Figure 36.12 and Figure 36.15, set to 1 bits DCKOSEL and FRQSEL (division by 2) in ESCR2. Non-multiplexed YC format can be displayed in Figure 36.13 and Figure 36.14, this setting is not required. These bits are not provided in DEFR0. |
| | | | | | Display Output Data Format. [RZ/G2E] B'00: Sets the DU0 display output to RGB data format. B'01: Setting prohibited B'10: Sets the DU0 display output to non-multiplexed YC data format. Y (luminance) and C (chrominance) are output in parallel. B'11: Sets the DU0 display output to multiplexed YC data format. Y (luminance) and C (chrominance) are output in multiplexed format. For YC data format, set to 1 the YCDF0 and VSPF0 bits in DEF10R0. When YC format is displayed by timing as shown in Figure 36.12 and Figure 36.15, set to 1 bits DCKOSEL and FRQSEL (division by 2) in ESCR0. Non-multiplexed YC format can be displayed in Figure 36.13 and Figure 36.14, this setting is not required. |
| 7, 6 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 5 | EXUP | B'0 | R/W | DRES | External Updating Mode 0: Internally updates the internal update function bit. 1: Externally updates the internal update function bit without updating it internally. This bit takes precedence over the IUPD bit in DSYSRn. |
| 4 | VCUP | B'0 | R/W | DRES | Vertical Cycle Register Update Timing Select 0: The internal updating is based on the falling VSYNC. 1: The internal updating is based on the rising VSYNC. By setting the internal updating of the vertical scanning cycle register as a VSYNC rise, any disturbance of VSYNC signals during vertical scanning cycle register switching can be prevented. For RZ/G2H, RZ/G2N, this bit is not provided in DEF2R. |
| 3 to 1 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|-----|----------|---------------|-----|-----------------|--|
| 0 | DEFE | B'0 | R/W | DRES | Display Unit Extensional Function Enable 0: Disables the extensional functions 1: Enables the following extensional functions: Enables bits 27 and 26 in PpMRm. Enables bits 25 and 5 in ESCRn. Enables bit 29 in DSYSRn. |

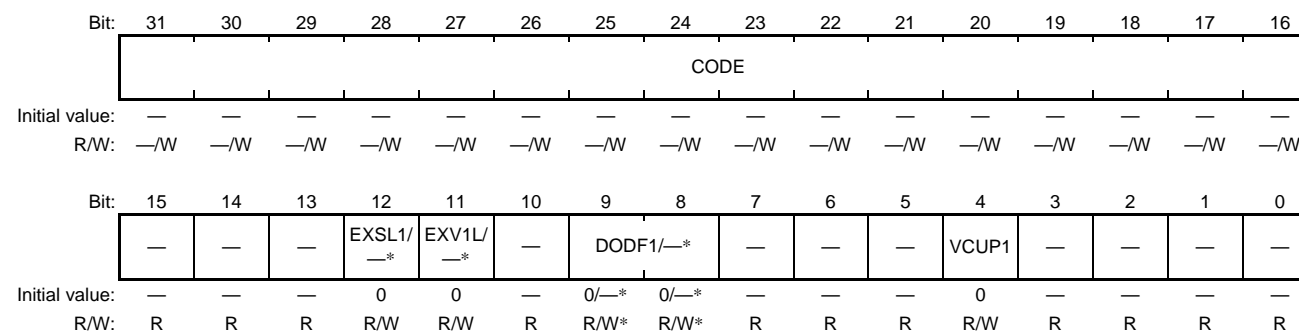
Note: * n = 0 and 2 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 n = 0 [RZ/G2E]

(13) Display Unit Extensional Function Control Register n (DEFRn)

Note: n = 1 and 3 [RZ/G2H, RZ/G2N]
 n = 1 [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU1: H'FEB3_0020, DU3: H'FEB7_0020



Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 16 | CODE | — | —/W | Not available | DEFRn* Enabling Code (register available code) For a value written to DEFRn* to be effective, the value must include H'7773 in these bits. |
| 15 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 | EXSL1 | B'0 | R/W | DRES | External Sync Signal Select 1[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: External SYNC signals (EXVSYNC, EXHSYNC) allow signals from the pins to be directly read in terms of post-division clocks. 1: External SYNC signals (EXVSYNC, EXHSYNC) allow the signals that were read in terms of pre-division clock to be read again as post-division clocks. |
| | — | — | R | — | Reserved [RZ/G2E] The read value is undefined. The write value should always be 0. |
| 11 | EXV1L | B'0 | R/W | Not available | External Vsync Latch Select 1[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: External VSYNC signal (EXVSYNC) allows the signal from the pin to be latched every clock cycle. 1: External VSYNC signal (EXVSYNC) allows the signal from the pin to be latched at the rising edge of the external HSYNC signal. |
| | — | — | R | — | Reserved [RZ/G2E] The read value is undefined. The write value should always be 0. |
| 10 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|--|
| 9, 8 | DODF1 | B'00 | R/W | DRES | <p>Display Output Data Format. [RZ/G2H, RZ/G2N]</p> <p>B'00: Sets the DU3 display output to RGB data format.</p> <p>B'01: Setting prohibited</p> <p>B'10: Sets the DU3 display output to non-multiplexed YC data format. Y (luminance) and C (chrominance) are output in parallel.</p> <p>B'11: Sets the DU3 display output to multiplexed YC data format. Y (luminance) and C (chrominance) are output in multiplexed format.</p> <p>For YC data format, set to 1 the YCDF1 and VSPF1 bits in DEF10R2.</p> <p>When YC format is displayed by timing as shown in Figure 36.12 and Figure 36.15, set to 1 bits DCKOSEL and FRQSEL (division by 2) in ESCR3. Non-multiplexed YC format can be displayed in Figure 36.13 and Figure 36.14, this setting is not required.</p> <p>These bits are not provided in DEFR1.</p> <hr/> <p>Display Output Data Format. [RZ/G2E]</p> <p>B'00: Sets the DU1 display output to RGB data format.</p> <p>B'01: Setting prohibited</p> <p>B'10: Sets the DU1 display output to non-multiplexed YC data format. Y (luminance) and C (chrominance) are output in parallel.</p> <p>B'11: Sets the DU1 display output to multiplexed YC data format. Y (luminance) and C (chrominance) are output in multiplexed format.</p> <p>For YC data format, set to 1 the YCDF1 and VSPF1 bits in DEF10R0</p> <p>When YC format is displayed by timing as shown in Figure 36.12 and Figure 36.15, set to 1 bits DCKOSEL and FRQSEL (division by 2) in ESCR1 Non-multiplexed YC format can be displayed in Figure 36.13 and Figure 36.14, this setting is not required.</p> |
| — | — | — | R | — | <p>Reserved [RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 7 to 5 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 4 | VCUP1 | B'0 | R/W | DRES | <p>Vertical cycle register update timing select</p> <p>0: The internal updating is based on the falling VSYNC.</p> <p>1: The internal updating is based on the rising VSYNC.</p> <p>By setting the internal updating of the vertical scanning cycle register as a VSYNC rise, any disturbance of VSYNC signals during vertical scanning cycle register switching can be prevented.</p> |
| 3 to 0 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |

Note: * n = 1 and 3 [RZ/G2H, RZ/G2N]
n = 1 [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]

(14) Display Unit Extensional Function Control 5 Register m (DEF5Rm)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB0_00E0, DU2/DU3: H'FEB4_00E0

| | | | | | | | | | | | | | | | | |
|----------------|--------|------|--------|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CODE | | | | | | | | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | YCRGB1 | | YCRGB0 | | — | — | — | — | — | — | — | — | — | — | — | DEF5 |
| Initial value: | 0/—* | 0/—* | 0 | 0 | — | — | — | — | — | — | — | — | — | — | — | 0 |
| R/W: | R/W* | R/W* | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 24 | CODE | — | —/W | Not available | DEF5Rm Enabling Code (register available code) For a value written to DEF5Rm to be effective, the value must include H'66 in these bits. |
| 23 to 16 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 15, 14 | YCRGB1 | B'00 | R/W | Available | YC-RGB Select 1 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E] B'00: YC-RGB conversion is not performed after superimposing in superimposition processor s*1. B'01: YC-RGB conversion is performed for the superimposed result of priority levels 1 and 2 in superimposition processor s*1. B'10: Setting prohibited. B'11: Setting prohibited. When YC-RGB conversion is performed, the functions of color detection in section 36.3.13 Color Detection cannot be used. For RZ/G2M V1.3, RZ/G2M V3.0, these bits are not provided in DEF5R2. |
| 13, 12 | YCRGB0 | B'00 | R/W | Available | YC-RGB Select 0 B'00: YC-RGB conversion is not performed after superimposing in superimposition processor s*2. B'01: YC-RGB conversion is performed for the superimposed result of priority levels 1 and 2 in superimposition processor s*2. B'10: Setting prohibited. B'11: Setting prohibited. When YC-RGB conversion is performed, the functions of color detection in section 36.3.13 Color Detection cannot be used. For RZ/G2H, RZ/G2N, these bits are not provided in DEF5R2. |
| 11 to 1 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

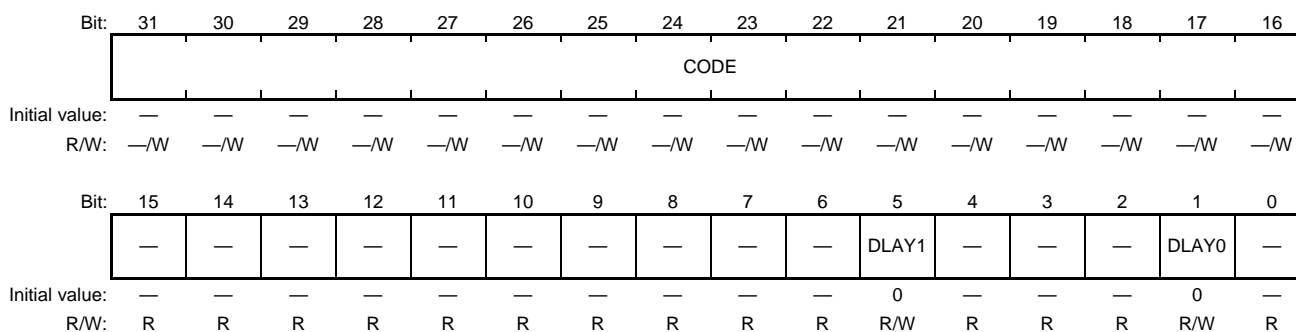
| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|-----|----------|---------------|-----|-----------------|---|
| 0 | DEFE5 | B'0 | R/W | DRES | <p>Display Unit Extensional Function Enable 5</p> <p>0: Extensional functions are disabled. 1: Extensional functions are enabled.</p> <p>The following extensional function is enabled.</p> <p>The number of bits is increased by one for the following registers:</p> <p>Display timing generation registers HDSR_n, HDER_n, VDER_n, HCR_n, VCR_n, VSPR_n, CLAMP_{SRn}, CLAMP_{PWRn}, DESR_n, and DEWR_n</p> <p>Display attribute registers RINTOFSR_n</p> <p>Display plane registers PpDSXR_m, PpDSYR_m, PpDPXR_m, and PpDPYR_m</p> |

- Notes:
- s = 1 and 3 [RZ/G2H, RZ/G2N]
s = 1 [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]
 - s = 0 and 2 [RZ/G2M V1.3, RZ/G2M V3.0]
s = 0 [RZ/G2H, RZ/G2N, RZ/G2E]

(15) Display Unit Data Latency Adjustment Register 0 (DDLTR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0: H'FEB0_00E4



This register is used to adjust the latency of two display data items output from the DU0 and DU1 when two image data items are output from the LVDS-IF on both edges (rising and falling edges) of the output dot clock. This register should not be changed from its initial value when the DR0D bit in DORCR0 is 0. When the DR0D bit is set to 1 (when two image data items are output from the LVDS-IF on both edges of the output dot clock), set the necessary bits in this register to 1.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 16 | CODE | — | —/W | Not available | DDLTR Enabling Code (register available code) For a value written to DDLTR to be effective, the value must include H'7766 in these bits. |
| 15 to 6 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 5 | DLAY1 | B'0 | R/W | DRES | Display Data Latency Adjustment YCRGB1 Set this bit to 1 in order to adjust the latency on the DU1 side when the values of the YCRGB0 and YCRGB1 bits in DEF5R0 are set to B'01 or B'10 (YC-RGB conversion is performed) and B'00 (YC-RGB conversion is not performed), respectively. 0: The DU1 display data is output without delay. 1: The DU1 display data is output with a delay corresponding to the latency of YC-RGB conversion on the DU0 side. |
| 4 to 2 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 1 | DLAY0 | B'0 | R/W | DRES | Display Data Latency Adjustment YCRGB0 Set this bit to 1 in order to adjust the latency on the DU0 side when the values of the YCRGB1 and YCRGB0 bits in DEF5R0 are B'01, B'10, or B'11 (YC-RGB conversion is performed) and B'00 (YC-RGB conversion is not performed), respectively. 0: The DU0 display data is output without delay. 1: The DU0 display data is output with a delay corresponding to the latency of YC-RGB conversion on the DU1 side. |
| 0 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

(16) Display Unit Extensional Function Control 6 Register m (DEF6Rm)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB0_00E8, DU2/DU3: H'FEB4_00E8

| | | | | | | | | | | | | | | | | |
|----------------|------|-----|-----|-----|-----------|--------|-----|-----|-----|-----|-----|-----|------|------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CODE | | | | | | | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | ODPM12/—* | ODPM02 | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | 0/—* | 0/—* | 0 | 0 | — | — | — | — | 0/—* | —*/0 | — | — |
| R/W: | R | R | R | R | R/W* | R/W* | R/W | R/W | R | R/W | R | R/W | R/W* | R/W* | R | R |

Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.

Bit 11 to 8 in this register setting is also available for the display data output via the LVDS-IF.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 16 | CODE | — | —/W | Not available | DEF6Rm Enabling Code (register available code) For a value written to DEF6Rm to be effective, the value must include H'7778 in these bits. |
| 15 to 12 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|---|
| 11, 10 | ODPM12 | B'00 | R/W | DRES | <p>ODDF Pin Mode 12 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>B'00: The ODDF pin function is determined by the ODPM bit in DSMRn*1.</p> <p>B'01: Setting prohibited.</p> <p>B'10: The DISP signal is output to the ODDF pin, which is a DU1 and DU3 pin. Even if the TVM1 bits in DSYSRn*1 indicate TV synchronized mode, the ODDF pin becomes an output.</p> <p>B'11: The CDE signal is output to the ODDF pin, which is a DU1 and DU3 pin. Even if the TVM1 bits in DSYSRn*1 indicate TV synchronized mode, the ODDF pin becomes an output.</p> <p>In case of the output via the DPAD, "ODDF pin" refers to the DU_EXODDF/DU_ODDF pin indicated in Table 36.3 and Table 36.4.</p> <p>In case of the output via the LVDS-IF, "ODDF pin" connected with the Odd/even signal of the LVDS-IF.</p> <p>For RZ/G2M V1.3, RZ/G2M V3.0, these bits are not provided in DEF6R2.</p> <hr/> <p>DU_DISP_CDE Pin Mode 12 [RZ/G2E]</p> <p>B'00: The DU_DISP_CDE pin function is determined by the ODPM bit in DSMRn*1.</p> <p>B'01: Setting prohibited.</p> <p>B'10: The DISP signal is output to the DU_DISP_CDE pin, which is a DU1 pins.</p> <p>B'11: The CDE signal is output to the DU_DISP_CDE pin, which is a DU1 pins.</p> <p>When the output as the digital RGB signal, "DU_DISP_CDE pin" refers to Table 36.5.</p> <p>When the output as the LVDS1-IF, "DU_DISP_CDE pin" connected with the odd/even signal of LVDS1-IF.</p> |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|------|----------|---------------|-----|-----------------|---|
| 9, 8 | ODPM02 | B'00 | R/W | DRES | <p>ODDF Pin Mode 02 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>B'00: The ODDF pin function is determined by the ODPM bit in DSMRn*2.</p> <p>B'01: Setting prohibited.</p> <p>B'10: The DISP signal is output to the ODDF pin, which are DU0 and DU2 pins. Even if the TVM bits in DSYSRn*2 indicate TV synchronized mode, the ODDF pin becomes an output.</p> <p>B'11: The CDE signal is output to the ODDF pin, which are DU0 and DU2 pins. Even if the TVM bits in DSYSRn*2 indicate TV synchronized mode, the ODDF pin becomes an output.</p> <p>In case of the output via the DPAD, "ODDF pin" refers to the DU_EXODDF/DU_ODDF pin indicated in Table 36.3 and Table 36.4.</p> <p>In case of the output via the LVDS-IF, "ODDF pin" connected with the Odd/even signal of the LVDS-IF.</p> <p>For RZ/G2H, RZ/G2N, these bits are not provided in DEF6R2.</p> <hr/> <p>DU_DISP_CDE Pin Mode 02 [RZ/G2E]</p> <p>B'00: The DU_DISP_CDE pin function is determined by the ODPM bit in DSMRn*2.</p> <p>B'01: Setting prohibited.</p> <p>B'10: The DISP signal is output to the DU_DISP_CDE pin, which is a DU0 pins.</p> <p>B'11: The CDE signal is output to the DU_DISP_CDE pin, which is a DU0 pins.</p> <p>When the output as the digital RGB signal, "DU_DISP_CDE pin" refers to Table 36.5.</p> <p>When the output as the LVDS0-IF, "DU_DISP_CDE pin" connected with the odd/even signal of LVDS0-IF.</p> |
| 7 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 6 | — | — | R/W | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 5 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 4 | — | — | R/W | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 3 | — | B'0 | R/W | DRES | <p>Reserved [RZ/G2H, RZ/G2N]</p> <p>The write value should always be 0.</p> <p>This bit is not provided in DEF6R0.</p> <hr/> <p>Reserved [RZ/G2E]</p> <p>The write value should always be 0.</p> <hr/> <p>— — R —</p> <p>Reserved [RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>The read value is undefined. The write value should always be 0.</p> |

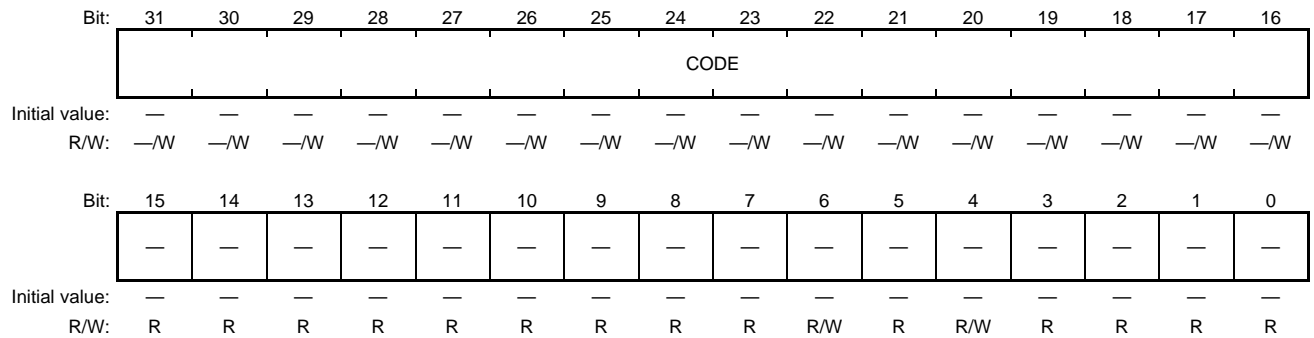
| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|------|----------|---------------|-----|-----------------|--|
| 2 | — | — | R | — | Reserved [RZ/G2H, RZ/G2N] The read value is undefined. The write value should always be 0. |
| | — | B'0 | R/W | DRES | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] The write value should always be 0. This bit is not provided in DEF6R0. |
| | | | | | Reserved [RZ/G2E] The write value should always be 0. |
| 1, 0 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

- Notes:
- n = 1 and 3 [RZ/G2H, RZ/G2N]
n = 1 [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]
 - n = 0 and 2 [RZ/G2M V1.3, RZ/G2M V3.0]
n = 0 [RZ/G2H, RZ/G2N, RZ/G2E]

(17) Display Unit Extensional Function Control 7 Register m (DEF7Rm)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB0_00EC, DU2/DU3: H'FEB4_00EC

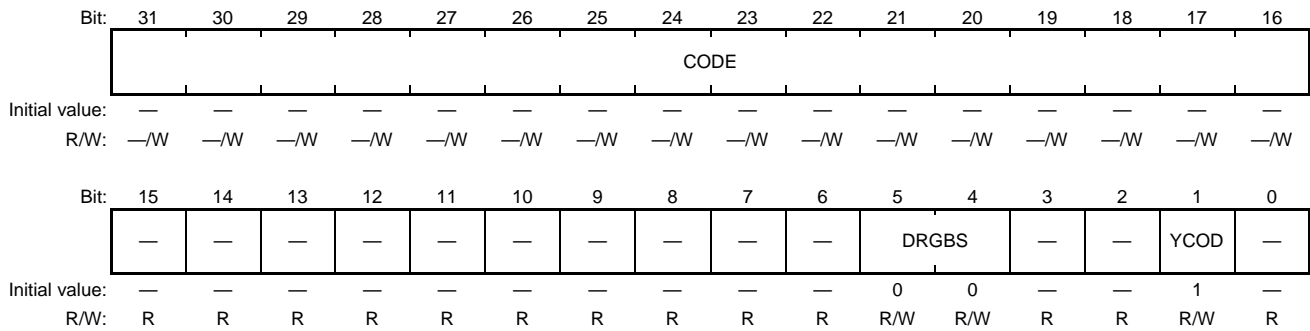


| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 16 | CODE | — | —/W | Not available | DEF7Rm Enabling Code (register available code) For a value written to DEF7Rm to be effective, the value must include H'7779 in these bits. |
| 15 to 7 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 6 | — | — | R/W | — | Reserved The read value is undefined. The write value should always be 0. |
| 5 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 4 | — | — | R/W | — | Reserved The read value is undefined. The write value should always be 0. |
| 3 to 0 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

(18) Display Unit Extensional Function Control 8 Register m (DEF8Rm)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB2_0020, DU2/DU3: H'FEB6_0020



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 16 | CODE | — | —/W | Not available | DEF8Rm Enabling Code (register available code) For a value written to DEF8Rm to be effective, the value must include H'7790 in these bits. |
| 15 to 6 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|------|----------|---------------|-----|-----------------|---|
| 5, 4 | DRGBS | B'00 | R/W | DRES | <p>Digital RGB Output Select [RZ/G2H, RZ/G2N]</p> <p>These bits are used to select the digital RGB and YC signal output from the DPAD (shown in Figure 36.2 [RZ/G2N, RZ/G2H]).</p> <p>B'00: Setting prohibited. B'01: Setting prohibited. B'10: Setting prohibited. B'11: Digital RGB and YC signal from the DU3 is selected.</p> <p>These bits are not provided in DEF8R0.</p> <p>For selection of signals output from the LVDS-IF, refer to the target specifications of the LVDS-IF.</p> <hr/> <p>Digital RGB Output Select [RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>These bits are used to select the digital RGB and YC signal output from the DPAD (shown in Figure 36.1).</p> <p>B'00: Setting prohibited. B'01: Setting prohibited. B'10: Digital RGB and YC signal from the DU2 is selected. B'11: Setting prohibited.</p> <p>These bits are not provided in DEF8R0.</p> <p>For selection of signals output from the LVDS-IF, refer to the target specifications of the LVDS-IF.</p> <hr/> <p>Digital RGB Output Select [RZ/G2E]</p> <p>These bits are used to select the digital RGB and YC signal output from the DPAD (shown in Figure 36.3).</p> <p>B'00: Digital RGB and YC signal from the DU0 is selected. B'01: Digital RGB and YC signal from the DU1 is selected. B'10: Setting prohibited. B'11: Setting prohibited.</p> <p>For selection of signals output from the LVDS-IF, refer to the target specifications of the LVDS-IF.</p> |
| 3, 2 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 1 | YCOD | B'1 | R/W | DRES | <p>YC Off mode Output Data</p> <p>0: When data is displayed in YC format, UV data in the non-display period is H'00. 1: When data is displayed in YC format, UV data in the non-display period is H'80.</p> <p>For RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, this bit is not provided in DEF8R0.</p> |
| 0 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |

(19) Display Unit Output Signal Fixed Level Register m (DOFLRm)

Note: m = 2 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
m = 0 [RZ/G2E]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB2_0024, DU2/DU3: H'FEB6_0024

| | | | | | | | | | | | | | | | | |
|----------------|------|-----|----------------|----------------|---------------|--------------|---------------|---------------|-----|-----|----------------|----------------|---------------|----------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CODE | | | | | | | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | HSYCF L1/—* | VSYCF L1/—* | ODDFL 1/—* | DISP FL1/ | CDEFL 1/—* | RGBF L1/—* | — | — | —*/HS YCFL0 | —*/VS YCFL0 | —*/OD DFL0 | —*/DI SPFL0 | —*/CD EFL0 | —*/RG BFL0 |
| Initial value: | — | — | 0/—* | 0/—* | 0/—* | 0/—* | 0/—* | 0/—* | — | — | —*/0 | —*/0 | —*/0 | —*/0 | —*/0 | —*/0 |
| R/W: | R | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* |

Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 16 | CODE | — | —/W | Not available | DOFLRm* Enabling Code (register available code) For a value written to DOFLRm* to be effective, the value must include H'7790 in these bits. |
| 15, 14 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 13 | HSYCFL1 | B'0 | R/W | Not available | HSYNC (DU3) Signal Fixed Low Level. [RZ/G2H, RZ/G2N] 0: HSYNC (DU3) output is normal. 1: HSYNC (DU3) output is fixed low. This bit setting is effective only in master mode. |
| | — | — | R | — | HSYNC (DU1) Signal Fixed Low Level. [RZ/G2E] 0: HSYNC (DU1) output is normal. 1: HSYNC (DU1) output is fixed low. |
| | — | — | R | — | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] The read value is undefined. The write value should always be 0. |
| 12 | VSYCFL1 | B'0 | R/W | Not available | VSYNC (DU3) Signal Fixed Low Level. [RZ/G2H, RZ/G2N] 0: VSYNC (DU3) output is normal. 1: VSYNC (DU3) output is fixed low. This bit setting is effective only in master mode. |
| | — | — | R | — | VSYNC (DU1) Signal Fixed Low Level. [RZ/G2E] 0: VSYNC (DU1) output is normal. 1: VSYNC (DU1) output is fixed low. |
| | — | — | R | — | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|------|----------|---------------|-----|-----------------|--|
| 11 | ODDFL1 | B'0 | R/W | Not available | ODDF (DU3) Signal Fixed Low Level. [RZ/G2H, RZ/G2N] 0: ODDF (DU3) output is normal. 1: ODDF (DU3) output is fixed low. This bit setting is effective only in master mode. |
| | — | — | R | — | Reserved [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E] The read value is undefined. The write value should always be 0. |
| 10 | DISPFL1 | B'0 | R/W | Not available | DISP (DU3) Signal Fixed Low Level. [RZ/G2H, RZ/G2N] 0: DISP (DU3) output is normal. 1: DISP (DU3) output is fixed low. DISP (DU1) Signal Fixed Low Level. [RZ/G2E] 0: DISP (DU1) output is normal. 1: DISP (DU1) output is fixed low. |
| | — | — | R | — | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] The read value is undefined. The write value should always be 0. |
| 9 | CDEFL1 | B'0 | R/W | Not available | CDE (DU3) Signal Fixed Low Level. [RZ/G2H, RZ/G2N] 0: CDE (DU3) output is normal. 1: CDE (DU3) output is fixed low. CDE (DU1) Signal Fixed Low Level. [RZ/G2E] 0: CDE (DU1) output is normal. 1: CDE (DU1) output is fixed low. |
| | — | — | R | — | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] The read value is undefined. The write value should always be 0. |
| 8 | RGBFL1 | B'0 | R/W | Not available | RGB (DU3) Signal Fixed Low Level. [RZ/G2H, RZ/G2N] 0: RGB (DU3) output is normal. 1: RGB (DU3) output is fixed low. RGB (DU1) Signal Fixed Low Level. [RZ/G2E] 0: RGB (DU1) output is normal. 1: RGB (DU1) output is fixed low. |
| | — | — | R | — | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] The read value is undefined. The write value should always be 0. |
| 7, 6 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 5 | — | — | R | — | Reserved [RZ/G2H, RZ/G2N] The read value is undefined. The write value should always be 0. |
| | HSYCFL0 | B'0 | R/W | Not available | HSYNC (DU2) Signal Fixed Low Level. [RZ/G2M V1.3, RZ/G2M V3.0] 0: HSYNC (DU2) output is normal. 1: HSYNC (DU2) output is fixed low. This bit setting is effective only in master mode. HSYNC (DU0) Signal Fixed Low Level. [RZ/G2E] 0: HSYNC (DU0) output is normal. 1: HSYNC (DU0) output is fixed low. This bit setting is effective only in master mode. |

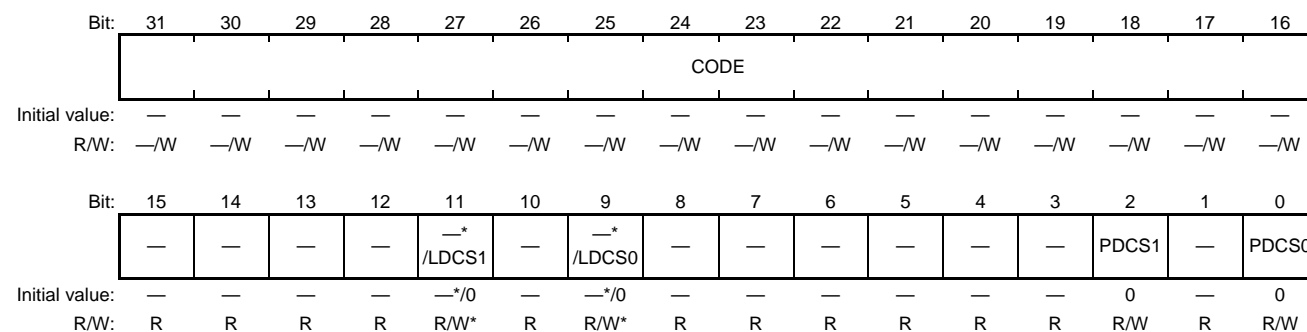
| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|-----|----------|---------------|-----|-----------------|--|
| 4 | — | — | R | — | Reserved [RZ/G2H, RZ/G2N] The read value is undefined. The write value should always be 0. |
| | VSYCFLO | B'0 | R/W | Not available | VSYNC (DU2) Signal Fixed Low Level. [RZ/G2M V1.3, RZ/G2M V3.0] 0: VSYNC (DU2) output is normal. 1: VSYNC (DU2) output is fixed low. This bit setting is effective only in master mode. VSYNC (DU0) Signal Fixed Low Level. [RZ/G2E] 0: VSYNC (DU0) output is normal. 1: VSYNC (DU0) output is fixed low. This bit setting is effective only in master mode. |
| 3 | — | — | R | — | Reserved [RZ/G2H, RZ/G2N, RZ/G2E] The read value is undefined. The write value should always be 0. |
| | ODDFLO | B'0 | R/W | Not available | ODDF (DU2) Signal Fixed Low Level. [RZ/G2M V1.3, RZ/G2M V3.0] 0: ODDF (DU2) output is normal. 1: ODDF (DU2) signal at fixed low level. This bit setting is effective only in master mode. |
| 2 | — | — | R | — | Reserved [RZ/G2H, RZ/G2N] The read value is undefined. The write value should always be 0. |
| | DISPFLO | B'0 | R/W | Not available | DISP (DU2) Signal Fixed Low Level. [RZ/G2M V1.3, RZ/G2M V3.0] 0: DISP (DU2) output is normal. 1: DISP (DU2) output is fixed low. DISP (DU0) Signal Fixed Low Level. [RZ/G2E] 0: DISP (DU0) output is normal. 1: DISP (DU0) output is fixed low. |
| 1 | — | — | R | — | Reserved [RZ/G2H, RZ/G2N] The read value is undefined. The write value should always be 0. |
| | CDEFLO | B'0 | R/W | Not available | CDE (DU2) Signal Fixed Low Level. [RZ/G2M V1.3, RZ/G2M V3.0] 0: CDE (DU2) output is normal. 1: CDE (DU2) output is fixed low. CDE (DU0) Signal Fixed Low Level. [RZ/G2E] 0: CDE (DU0) output is normal. 1: CDE (DU0) output is fixed low. |
| 0 | — | — | R | — | Reserved [RZ/G2H, RZ/G2N] The read value is undefined. The write value should always be 0. |
| | RGBFLO | B'0 | R/W | Not available | RGB (DU2) Signal Fixed Low Level. [RZ/G2M V1.3, RZ/G2M V3.0] 0: RGB (DU2) output is normal. 1: RGB (DU2) output is fixed low. RGB (DU0) Signal Fixed Low Level. [RZ/G2E] 0: RGB (DU0) output is normal. 1: RGB (DU0) output is fixed low. |

Note: * m = 2 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
m = 0 [RZ/G2E]

(20) Display Unit Input Dot Clock Select Register 0 (DIDSR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB2_0028



Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 16 | CODE | — | —/W | Not available | DIDSR0 Enabling Code (register available code) For a value written to DIDSR0 to be effective, the value must include H'7790 in these bits. |
| 15 to 12 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 11 | — | — | R | — | Reserved[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] The read value is undefined. The write value should always be 0. |
| | LDCS1 | B'0 | R/W | DRES | DU0 LVDS-IF Dot Clock Select [RZ/G2E] 0: The DU1 input dot clock source is the DCLKIN pin. 1: The DU1 input dot clock source is the LVDS-IF pin. LVDS-IF pin mean the equivalent clock of the internal CLK clock from the LVDS-IF PLL. When setting B'1 to this bit, the LVDS must be enabled. |
| 10 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 9 | — | — | R | — | Reserved [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] The read value is undefined. The write value should always be 0. |
| | LDCS0 | B'0 | R/W | DRES | DU0 LVDS-IF Dot Clock Select [RZ/G2E] 0: The DU0 input dot clock source is the DCLKIN pin. 1: The DU0 input dot clock source is the LVDS-IF pin. LVDS-IF pin mean the equivalent clock of the internal CLK clock from the LVDS-IF PLL. When setting B'1 to this bit, the LVDS must be enabled. |
| 8 to 3 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 2 | PDCS1 | B'0 | R/W | DRES | DU1 Pad Dot Clock Select 0: The DU1 input dot clock source is the DU_DOTCLKIN1 pin. 1: The DU1 input dot clock source is the DU_DOTCLKIN0 pin. |
| 1 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|-----|----------|---------------|-----|-----------------|--|
| 0 | PDCS0 | B'0 | R/W | DRES | DU0 Pad Dot Clock Select 0: The DU0 input dot clock source is the DU_DOTCLKIN0 pin. 1: The DU0 input dot clock source is the DU_DOTCLKIN1 pin. |

(21) Display Unit Input Dot Clock Select Register 2 (DIDSR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | — |

Address: DU2/DU3: H'FEB6_0028

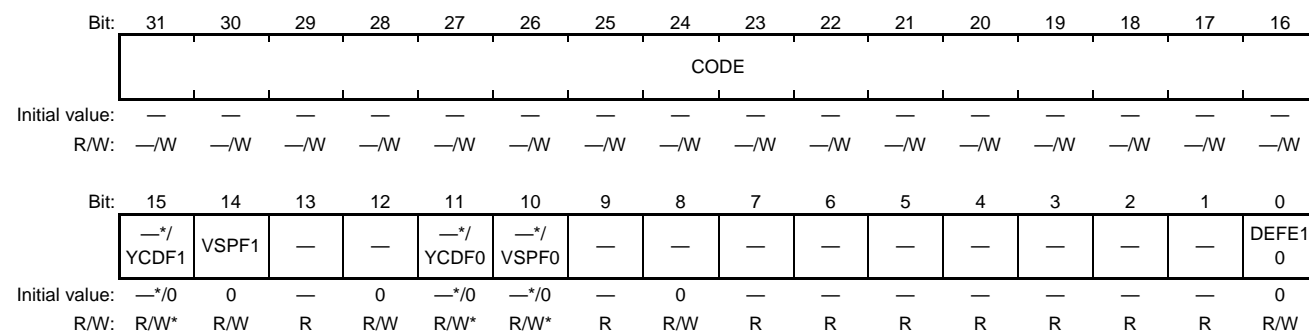
| | | | | | | | | | | | | | | | | |
|----------------|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|--------|-----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CODE | | | | | | | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W | —/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | PDC S1 | — | PDC S0 |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | 0 | — | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 16 | CODE | — | —/W | Not available | DIDSR2 Enabling Code (register available code) For a value written to DIDSR2 to be effective, the value must include H'7790 in these bits. |
| 15 to 3 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 2 | PDCS1 | B'0 | R/W | DRES | DU3 Pad Dot Clock Select 0: The DU3 input dot clock source is the DU_DOTCLKIN3 pin. 1: The DU3 input dot clock source is the DU_DOTCLKIN2 pin. |
| 1 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 0 | PDCS0 | B'0 | R/W | DRES | DU2 Pad Dot Clock Select 0: The DU2 input dot clock source is the DU_DOTCLKIN2 pin. 1: The DU2 input dot clock source is the DU_DOTCLKIN3 pin. |

(22) Display Unit Extensional Function Control 10 Register 0 (DEF10R0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB2_0038



Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 16 | CODE | — | —/W | Not available | DEF10R0 Enabling Code (register available code) For a value written to DEF10R0 to be effective, the value must include H'7795 in these bits. |
| 15 | — | — | R | — | Reserved[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] The read value is undefined. The write value should always be 0. |
| | YCDF1*1 | B'0 | R/W | Available | YC data format [RZ/G2E] This bit defines YC data format of DU1. (If DU1's data format is not YC, any setting is invalid.) 0: YCbCr444 1: YCbCr422 For RZ/G2E, YCbCr422 is only available. |
| 14 | VSPF1*1 | B'0 | R/W | Available | VSP1 data format This bit defines data format of VSP1 connected with DU1. 0: VSP1 data format is RGB 1: VSP1 data format is YC |
| 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 | — | B'0 | R/W | Available | Reserved The read value is undefined. The write value should always be 0. |
| 11 | YCDF0*2 | B'0 | R/W | Available | YC data format [RZ/G2E] This bit defines YC data format of DU0. (If DU0's data format is not YC, any setting is invalid.) 0: YCbCr444 1: YCbCr422 For RZ/G2E, YCbCr422 is only available. |
| | — | — | R | — | Reserved [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] The read value is undefined. The write value should always be 0. |

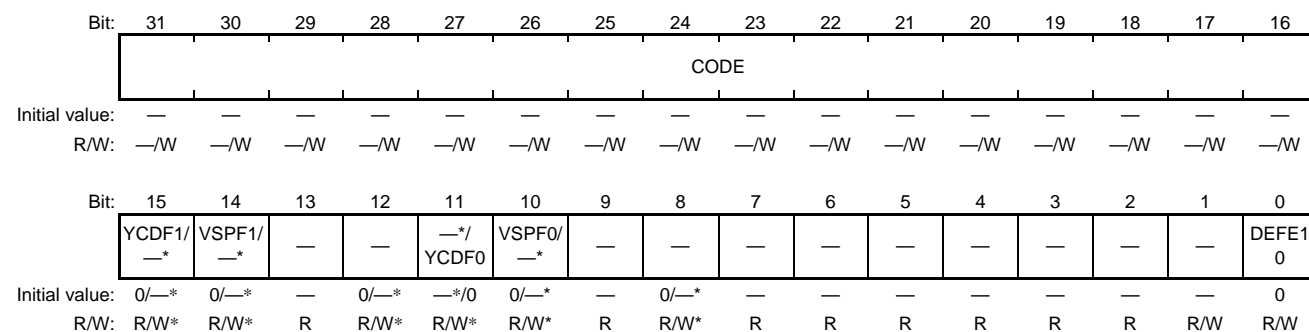
| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|--|
| 10 | VSPF0*2 | B'0 | R/W | Available | VSP0 for DU0 data format [RZ/G2E] This bit defines data format of VSP0 connected with DU0. 0: VSP0 for DU0 data format is RGB 1: VSP0 for DU0 data format is YC |
| — | — | — | R | — | Reserved [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] The read value is undefined. The write value should always be 0. |
| 9 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 8 | — | B'0 | R/W | Available | Reserved The read value is undefined. The write value should always be 0. |
| 7 to 1 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 0 | DEFE10 | B'0 | R/W | DRES | Display Unit Extensional Function Enable 10 0: Extensional functions are disabled. 1: Extensional functions are enabled. The following extensional function is enabled. The number of bits is increased for the following registers. Refer to each register chapter for the number of increase. Display timing generation registers HDSRn*3, HDERn*3, VDERn*3, HCRn*3, VCRn*3, VSPRn*3, CLAMPSRn*3, CLAMPWRn*3, DESRn*3, and DEWRn*3 Display attribute registers RINTOFSRn*3 Display plane registers PpDSXR0, PpDSYR0, PpDPXR0, and PpDPYR0. |

- Notes:
- When DU1's data format is YC, set the corresponding registers YC data format as follows.
For corresponding data format of each product, refer to Table 36.2.
— DEF10R0.VSPF1 = B'1
— DEFR1.DODF1 = B'10 or B'11
— P3DDC4R0.P3EDF = B'100
 - When DU0's data format is YC, set the corresponding registers YC data format as follows.
For corresponding data format of each product, refer to Table 36.2.
— DEF10R0.VSPF0 = B'1
— DEFR0.DODF = B'10 or B'11
— P1DDC4R0.P1EDF = B'100
 - n = 0 and 1

(23) Display Unit Extensional Function Control 10 Register 2 (DEF10R2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Address: DU2/DU3: H'FEB6_0038



Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details of bit description, refer to following table.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 16 | CODE | — | —/W | Not available | DEF10R2 Enabling Code (register available code) For a value written to DEF10R2 to be effective, the value must include H'7795 in these bits. |
| 15 | YCDF1*1 | B'0 | R/W | Available | YC data format [RZ/G2H, RZ/G2N] This bit defines YC data format of DU3. (If DU3's data format is not YC, any setting is invalid.) 0: YCbCr444 1: YCbCr422. For RZ/G2H and RZ/G2N, YCbCr422 is only available. |
| | — | — | R | — | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] The read value is undefined. The write value should always be 0. |
| 14 | VSPF1*1 | B'0 | R/W | Available | VSP0 for DU3 data format [RZ/G2H, RZ/G2N] This bit defines data format of VSP0 connected with DU3. 0: VSP0 data format is RGB 1: VSP0 data format is YC |
| | — | — | R | — | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] The read value is undefined. The write value should always be 0. |
| 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|---|
| 12 | — | B'0 | R/W | Available | Reserved [RZ/G2H] [RZ/G2N] The read value is undefined. The write value should always be 0. |
| | — | — | R | — | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] The read value is undefined. The write value should always be 0. |
| 11 | — | — | R | — | Reserved [RZ/G2H, RZ/G2N] The read value is undefined. The write value should always be 0. |
| | YCDF0*2 | B'0 | R/W | Available | YC data format [RZ/G2M V1.3, RZ/G2M V3.0] This bit defines YC data format of DU2. (If DU2's data format is not YC, any setting is invalid.) 0: YCbCr444 1: YCbCr422 For RZ/G2M V1.3, RZ/G2M V3.0, YCbCr422 is only available. |
| 10 | VSPF0*2 | B'0 | R/W | Available | VSP2 data format [RZ/G2M V1.3, RZ/G2M V3.0] This bit defines data format of VSP2 connected with DU2. 0: VSP2 data format is RGB 1: VSP2 data format is YC |
| | — | — | R | — | Reserved [RZ/G2N, RZ/G2H] The read value is undefined. The write value should always be 0. |
| 9 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 8 | — | B'0 | R/W | Available | Reserved [RZ/G2M V1.3, RZ/G2M V3.0] The read value is undefined. The write value should always be 0. |
| | — | — | R | — | Reserved [RZ/G2N, RZ/G2H] The read value is undefined. The write value should always be 0. |
| 7 to 2 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 1 | — | — | R/W | — | Reserved [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 RZ/G2N] The read value is undefined. The write value should always be 0.. |
| 0 | DEFE10 | B'0 | R/W | DRES | Display Unit Extensional Function Enable 10 0: Extensional functions are disabled. 1: Extensional functions are enabled. The following extensional function is enabled. The number of bits is increased for the following registers. Refer to each register chapter for the number of increase. Display timing generation registers HDSRn*3, HDERn*3, VDERn*3, HCRn*3, VCRn*3, VSPRn*3, CLAMPSRn*3, CLAMPWRn*3, DESRn*3, and DEWRn*3* Display attribute registers RINTOFSRn*3 Display plane registers PpDSXR2, PpDSYR2, PpDPXR2, and PpDPYR2. |

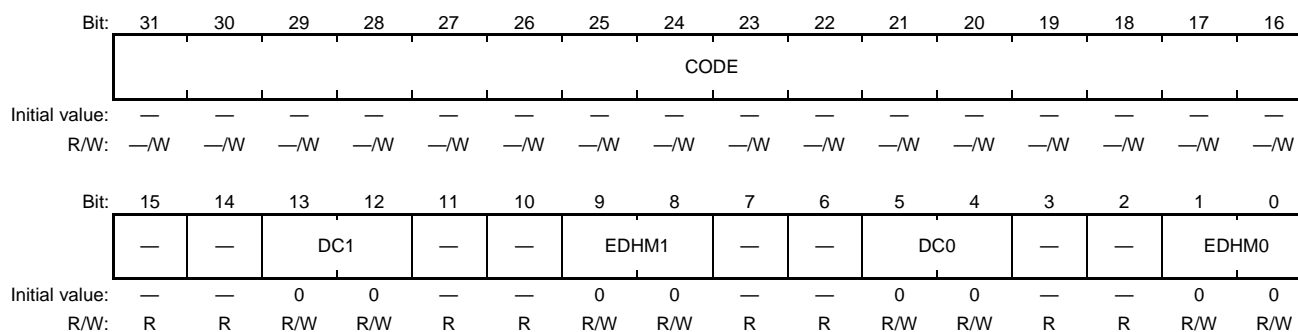
- Notes: 1. When DU3's data format is YC, set the corresponding registers YC data format as follows.
For corresponding data format of each product, refer to Table 36.2.
— DEF10R2.VSPF1 = B'1
— DEFR3.DODF1 = B'10 or B'11
— P3DDC4R2.P3EDF = B'100
2. When DU2's data format is YC, set the corresponding registers YC data format as follows.
For corresponding data format of each product, refer to Table 36.2.

- DEF10R2.VSPF0 = B'1
 - DEFR2.DODF = B'10 or B'11
 - P1DDC4R2.P1EDF = B'100
3. n = 2 [RZ/G2M V1.3, RZ/G2M V3.0]
n = 3 [RZ/G2H, RZ/G2N]

(24) Display Unit Dither Control Register m (DDTHCRm)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB2_003C, DU2/DU3: H'FEB6_003C



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 16 | CODE | — | —/W | Not available | DDTHCRm Enabling Code (register available code) For a value written to DDTHCRm to be effective, the value must include H'7795 in these bits. |
| 15, 14 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 13, 12 | DC1 | B'00 | R/W | Available | Dithering Mode Control 1 This bit select the dithering mode for conversion from RGB888 to RGB666 in superimposition processor s*1. B'00: Dithering with cumulative addition B'01: Ordered dithering B'10: Setting prohibited B'11: Setting prohibited For RZ/G2M V1.3, RZ/G2M V3.0, this bit is not provided in DDTHCR2. |
| 11, 10 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 9, 8 | EDHM1 | B'00 | R/W | Available | Extensional Dither Mode 1 B'00: Dither conversion is not performed by the setting of the DC1 bits in this register in superimposition processor s*1. B'01: Dither conversion is performed by the setting of the DC1 bits in this register in superimposition processor s*1. B'10: Setting prohibited B'11: Setting prohibited For RZ/G2M V1.3, RZ/G2M V3.0, this bit is not provided in DDTHCR2. |
| 7, 6 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|------|----------|---------------|-----|-----------------|--|
| 5, 4 | DC0 | B'00 | R/W | Available | <p>Dithering Mode Control 0</p> <p>These bits select the dithering mode for conversion from RGB888 to RGB666 in superimposition processor s*²</p> <p>B'00: Dithering with cumulative addition</p> <p>B'01: Ordered dithering</p> <p>B'10: Setting prohibited</p> <p>B'11: Setting prohibited</p> <p>For RZ/G2N, RZ/G2H, this bit is not provided in DDTHCR2.</p> |
| 3, 2 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 1, 0 | EDHM0 | B'00 | R/W | Available | <p>Extensional Dither Mode 0</p> <p>B'00: Dither conversion is not performed by the setting of the DC0 bits in this register in superimposition processor s*².</p> <p>B'01: Dither conversion is performed by the setting of the DC0 bits in this register in superimposition processor s*².</p> <p>B'10: Setting prohibited</p> <p>B'11: Setting prohibited</p> <p>For RZ/G2N, RZ/G2H, this bit is not provided in DDTHCR2.</p> |

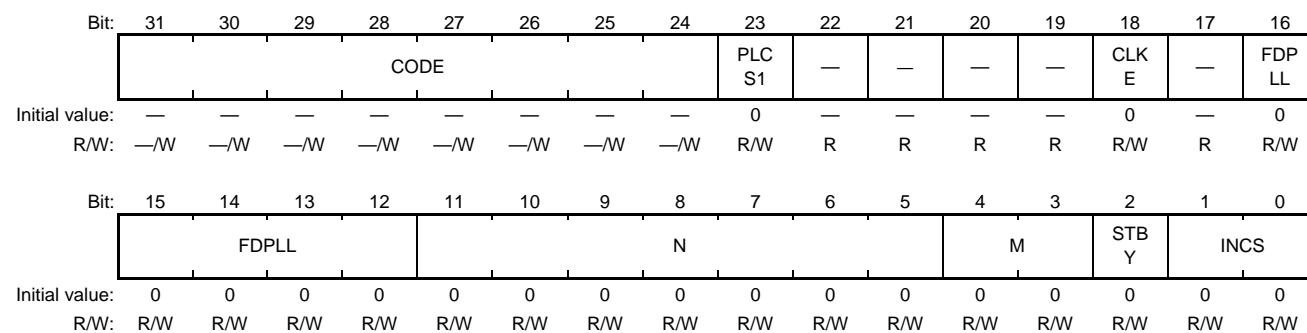
- Notes: 1. s = 1 and 3 [RZ/G2H, RZ/G2N].
s = 1 [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E].
2. s = 0 and 2 [RZ/G2M V1.3, RZ/G2M V3.0].
s = 0 [RZ/G2H, RZ/G2N, RZ/G2E]

(25) Display Unit PLL Control Register m (DPLLCRm)

Note: m = 0 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Address: DU1: H'FEB2_0044, DU2: H'FEB6_0044



Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified. For details, refer to following table.

For RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, the DPLLCR0 control DPLL0. There is no DPLL1.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 24 | CODE | — | —/W | Not available | DPLLCRm* Enabling Code (register available code) For a value written to DPLLCRm* to be effective, the value must include H'95 in these bits. |
| 23 | PLCS1 | B'0 | R/W | DRES | DU1 DPLL Clock Select 0: The DU1 input dot clock source is the DCLKIN pin. 1: The DU1 input dot clock source is the DPLL0. For , this bit is not provided in DPLLCR2. |
| 22 to 19 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 18 | CLKE | B'0 | R/W | DRES | DPLL0/DPLL1 Clock Output Enable 0: The DPLL0/DPLL1 clock is not output. 1: The DPLL0/DPLL1 clock is output. |
| 17 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 16 to 12 | FDPLL | H'00 | R/W | DRES | <p>Frequency division of the output DPLL0/DPLL1 clock.</p> <p>These bits set the frequency division ratio of output the DPLL0/DPLL1 clock.</p> <p>The clock source is divided by the division ratio of $1/(\text{setting} + 1)$.</p> <p>H'00: Setting prohibited H'01: x1/2 H'02: x1/3 : H'1F: x1/32</p> <p>The frequency of the divided clock should be less than 400 MHz.</p> <p>There is the frequency division ratio that the duty cycle of the divided clock is not 50%.</p> <p>FDPLL can also be set by the DPLLC2Rm*. In case of setting x 1/33 and over, set by the DPLLC2Rm*.</p> |
| 11 to 5 | N | H'00 | R/W | DRES | <p>DPLL0/DPLL1 Multiplier Setting of Feedback clock</p> <p>H'00 to H'26: Setting prohibited. H'27: x40 H'28: x41 : H'77: x120 H'78 to H'7F: Setting prohibited.</p> <p>$\text{DPLL0/DPLL1 Output Frequency} = (\text{DPLL0/DPLL1 Input Frequency} \times (\text{N}+1)) / (\text{M}+1) / (\text{FDPLL}+1)$.</p> |
| 4, 3 | M | B'00 | R/W | DRES | <p>DPLL0/DPLL1 Divider Setting of Clock Input</p> <p>B'00: x1 B'01: x1/2 B'10: x1/3 B'11: x1/4</p> <p>M can also be set by the DPLLC2Rm*. In case of setting x 1/5 and over, set by the DPLLC2Rm*.</p> |
| 2 | STBY | B'0 | R/W | DRES | <p>DPLL0/DPLL1 Enable</p> <p>Controls the DPLL0/DPLL1 operation.</p> <p>0: The DPLL0/DPLL1 stopped. 1: The DPLL0/DPLL1 operates.</p> |
| 1, 0 | INCS | B'00 | R/W | DRES | <p>DPLL0 Input Clock Select [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>B'00: The DPLL0 input clock source is the DU_DOTCLKIN0 pin. B'01: Setting prohibited. B'10: The DPLL0 input clock source is the DU_DOTCLKIN1 pin. B'11: Setting prohibited.</p> |

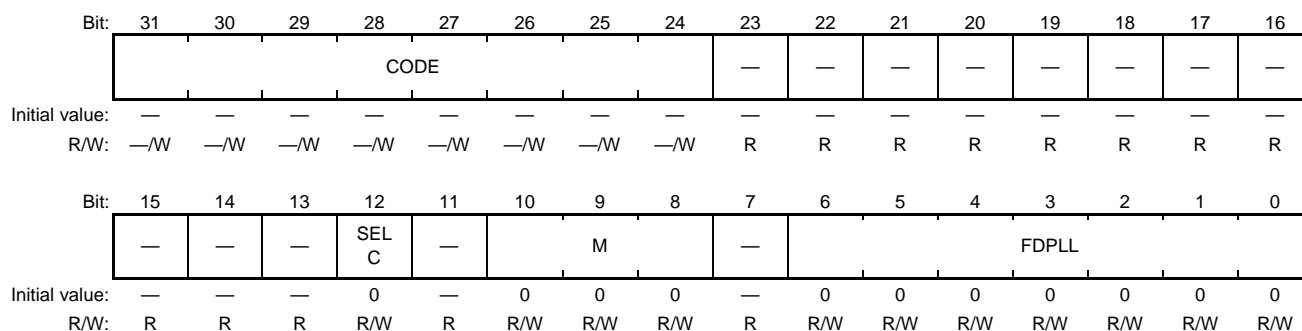
Note: * m = 0 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

(26) Display Unit PLL Control 2 Register m (DPLLC2Rm)

Note: m = 0 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Address: DU1: H'FEB2_0048, DU2: H'FEB6_0048



For RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, the DDTHC2R0 control DPLL0. There is no DPLL1.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 24 | CODE | — | —/W | Not available | DPLLC2Rm* Enabling Code (register available code) For a value written to DPLLC2Rm* to be effective, the value must include H'95 in these bits. |
| 23 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 | SELC | B'0 | R/W | DRES | Selection of register 0: The frequency of the divided clock (FDPLL) and divider setting of clock input (M) is set by the DPLLCRm*. 1: The frequency of the divided clock (FDPLL) and divider setting of clock input (M) is set by this register. |
| 11 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 10 to 8 | M | B'000 | R/W | DRES | DPLL0/DPLL1 Divider Setting of Clock Input B'000: x1 B'001: x1/2 B'010: x1/3 : B'111: x1/8 To enable this bit, set the SELC bit in this register to 1. |
| 7 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|---|
| 6 to 0 | FDPLL | H'00 | R/W | DRES | <p>Frequency division of the output DPLL0/DPLL1 clock.</p> <p>These bits set the frequency division ratio of output DPLL0/DPLL1 clock.</p> <p>The clock source is divided by the division ratio of $1/(\text{setting} + 1)$.</p> <p>H'00: Setting prohibited</p> <p>H'01: x1/2</p> <p>H'02: x1/3</p> <p>:</p> <p>H'7F: x1/128</p> <p>The frequency of the divided clock should be less than 400MHz.</p> <p>There is the frequency division ratio that the duty cycle of the divided clock is not 50%.</p> <p>To enable this bit, set the SELC bit in this register to 1.</p> |

Note: * m = 0 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

36.2.2 Display Timing Generation Registers

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: All registers in section 36.2.2 are common to RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N.
For RZ/G2E, there are not EQWRn and SPWRn registers, other registers are common.

The sets of display timing generation registers are for the respective channels and have the same functions; they are described as one here.

(1) Horizontal Display Start Register n (HDSRn)

Address: DU0: H'FEB0_0040, DU1: H'FEB3_0040, DU2: H'FEB4_0040, DU3: H'FEB7_0040

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | — | HDS | | | | | | | | | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 10 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 9 to 0 | HDS | — | R/W | Available | Horizontal Display Start To enable bit 9, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 9 cannot be written to. These bits are used to set the horizontal display start position in dot clock units. The set value is retained at a reset. HDS should be set to 1 or greater. |

(2) Horizontal Display End Register n (HDERn)

Address: DU0: H'FEB0_0044, DU1: H'FEB3_0044, DU2: H'FEB4_0044, DU3: H'FEB7_0044

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | HDE | | | | | | | | | | | | | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 to 0 | HDE | — | R/W | Available | Horizontal Display End To enable bit 11, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 11 cannot be written to. To enable bit 12, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12 cannot be written to. These bits are used to set the horizontal display end position in dot clock units. The set value is retained at a reset. |

(3) Vertical Display Start Register n (VDSRn)

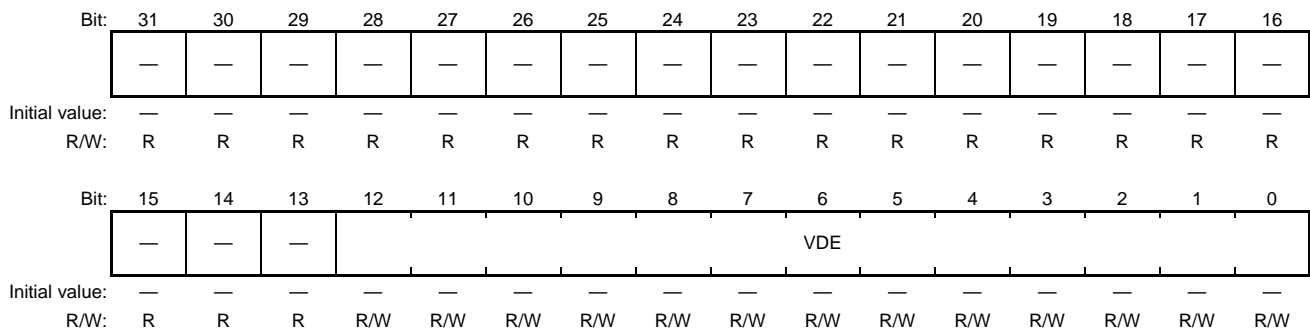
Address: DU0: H'FEB0_0048, DU1: H'FEB3_0048, DU2: H'FEB4_0048, DU3: H'FEB7_0048

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | — | — | VDS | | | | | | | | | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|---------|----------|---------------|-----|-----------------|---|
| 31 to 9 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 8 to 0 | VDS | — | R/W | Available | Vertical Display Start These bits are used to set the vertical display start position in raster line units. The set value is retained at a reset. VDS should be set to 1 or greater. |

(4) Vertical Display End Register n (VDERn)

Address: DU0: H'FEB0_004C, DU1: H'FEB3_004C, DU2: H'FEB4_004C, DU3: H'FEB7_004C



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 to 0 | VDE | — | R/W | Available | Vertical Display End To enable bit 10, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 10 cannot be written to. To enable bit 12, 11, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12, 11 cannot be written to. These bits are used to set the vertical display end position in raster line units. The set value is retained at a reset. |

(5) Horizontal Cycle Register n (HCRn)

Address: DU0: H'FEB0_0050, DU1: H'FEB3_0050, DU2: H'FEB4_0050, DU3: H'FEB7_0050

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | HC | | | | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 to 0 | HC | — | R/W | Available | Horizontal Cycle [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] To enable bit 11, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 11 cannot be written to. To enable bit 12, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12 cannot be written to. These bits are used to set one horizontal scan cycle including the horizontal blanking period in dot clock units. In TV synchronized mode, set this register so that the HSYNC cycle set with this register is equal to or greater than the EXHSYNC cycle. The set value is retained at a reset. Horizontal Cycle [RZ/G2E] To enable bit 11, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 11 cannot be written to. To enable bit 12, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12 cannot be written to. These bits are used to set one horizontal scan cycle including the horizontal blanking period in dot clock units. The set value is retained at a reset. |

(6) Horizontal Sync Width Register n (HSWRn)

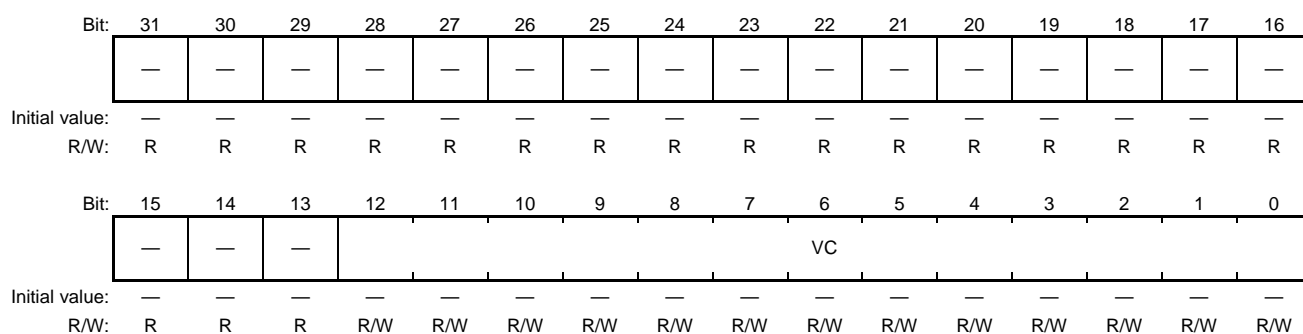
Address: DU0: H'FEB0_0054, DU1: H'FEB3_0054, DU2: H'FEB4_0054, DU3: H'FEB7_0054

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | HSW | | | | | | | | | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|---------|----------|---------------|-----|-----------------|---|
| 31 to 9 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 8 to 0 | HSW | — | R/W | Available | Horizontal Sync Width These bits are used to set the low-level pulse width of the horizontal synchronous signal in dot clock units. The set value is retained at a reset. |

(7) Vertical Cycle Register n (VCRn)

Address: DU0: H'FEB0_0058, DU1: H'FEB3_0058, DU2: H'FEB4_0058, DU3: H'FEB7_0058



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 to 0 | VC | — | R/W | Available | Vertical Cycle [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] To enable bit 10, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 10 cannot be written to. To enable bit 12, 11, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12, 11 cannot be written to. These bits are used to set the vertical scan cycle including the vertical blanking period in raster line units. In TV synchronized mode, set the time limit of the EXVSYNC rising edge detection. If the EXVSYNC rising edge is not detected within the time limit, the result is reflected in bit 15 in DSSRn. The set value is retained at a reset. |
| | | | | | Vertical Cycle [RZ/G2E] To enable bit 10, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 10 cannot be written to. To enable bit 12, 11, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12, 11 cannot be written to. These bits are used to set the vertical scan cycle including the vertical blanking period in raster line units. The set value is retained at a reset. |

(8) Vertical Sync Point Register n (VSPRn)

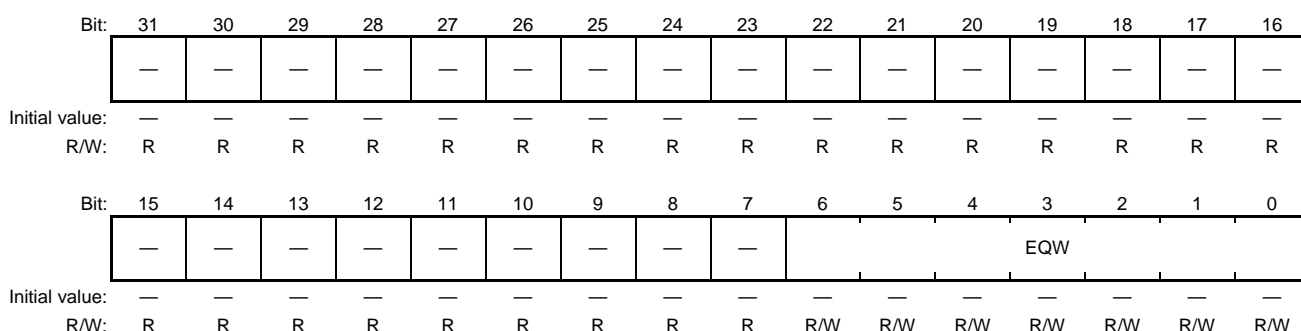
Address: DU0: H'FEB0_005C, DU1: H'FEB3_005C, DU2: H'FEB4_005C, DU3: H'FEB7_005C

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | VSP | | | | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 to 0 | VSP | — | R/W | Available | Vertical Sync Point [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] To enable bit 10, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 10 cannot be written to. To enable bit 12, 11, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12, 11 cannot be written to. These bits are used to set the vertical synchronous signal start position in raster line units. In TV synchronized mode, set this register so that the VSYNC falling edge setting position set with this register is the same as or comes after that of the EXVSYNC falling edge. The set value is retained at a reset. <hr/> Vertical Sync Point [RZ/G2E] To enable bit 10, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 10 cannot be written to. To enable bit 12, 11, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12, 11 cannot be written to. These bits are used to set the vertical synchronous signal start position in raster line units. The set value is retained at a reset. |

(9) Equal Pulse Width Register n (EQWRn)

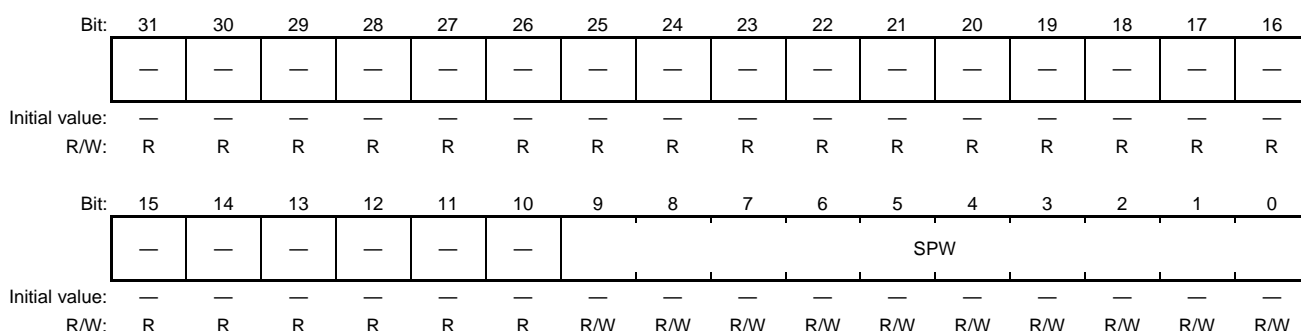
Address: DU0: H'FEB0_0060, DU1: H'FEB3_0060, DU2: H'FEB4_0060, DU3: H'FEB7_0060



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|---------|----------|---------------|-----|-----------------|--|
| 31 to 7 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 6 to 0 | EQW | — | R/W | Available | Equal Pulse Width [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] These bits are used to set the low-level equalizing pulse width of the CSYNC signal in dot clock units. To validate this setting, set bit 7 in DSMRn to 1. The set value is retained at a reset. |

(10) Serration Width Register n (SPWRn)

Address: DU0: H'FEB0_0064, DU1: H'FEB3_0064, DU2: H'FEB4_0064, DU3: H'FEB7_0064



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 10 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 9 to 0 | SPW | — | R/W | Available | Serration Width [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] These bits are used to set the low-level serration pulse width of the CSYNC signal in dot clock units. Set a value smaller than 1/2 of HC. To validate this setting, set bit 7 in DSMRn to 1. The set value is retained at a reset. |

(11) CLAMP Signal Start Register n (CLAMPSRn)

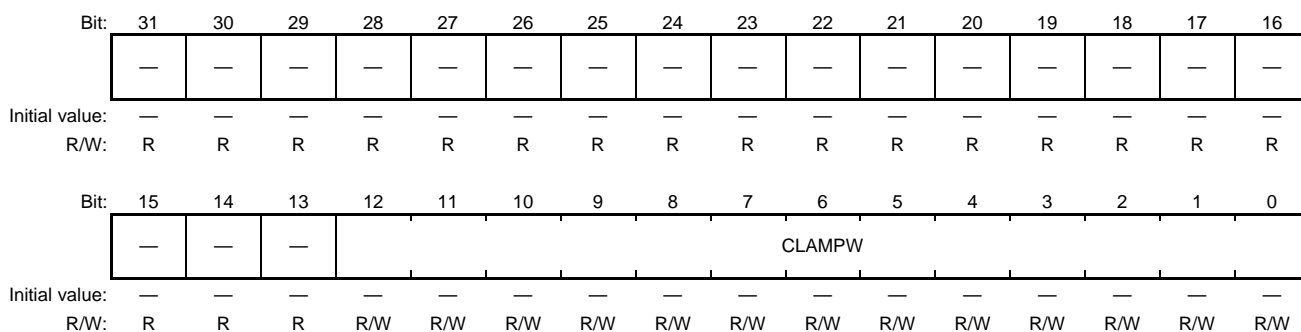
Address: DU0: H'FEB0_0070, DU1: H'FEB3_0070, DU2: H'FEB4_0070, DU3: H'FEB7_0070

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|--------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | CLAMPS | | | | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 to 0 | CLAMPS | — | R/W | Available | <p>CLAMP Signal Start</p> <p>To enable bit 11, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 11 cannot be written to.</p> <p>To enable bit 12, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12 cannot be written to.</p> <p>These bits are used to set the CLAMP signal rising edge position in dot clock units, using as the reference the HSYNC signal falling edge.</p> <p>The CLAMP signal rises (setting value + 1) cycle after the HSYNC signal falls. Therefore, the CLAMP signal cannot rise in the same cycle as the HSYNC signal falling edge.</p> <p>The set value is retained at a reset.</p> |

(12) CLAMP Signal Width Register n (CLAMPWRn)

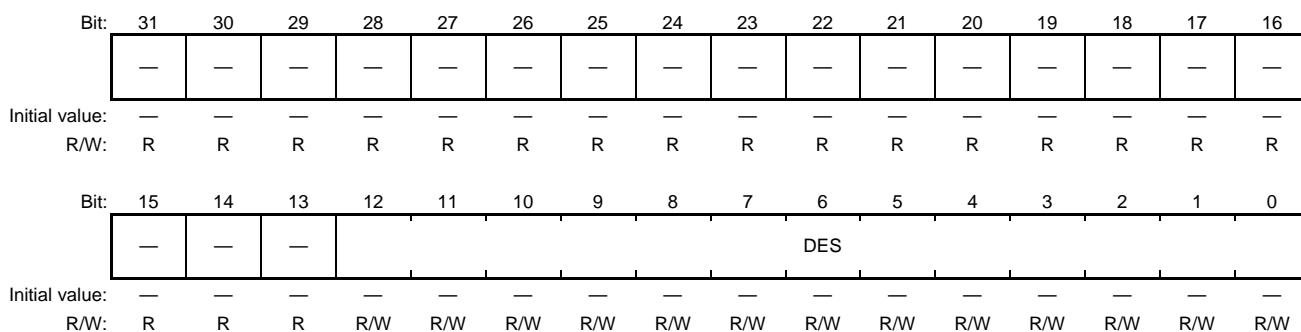
Address: DU0: H'FEB0_0074, DU1: H'FEB3_0074, DU2: H'FEB4_0074, DU3: H'FEB7_0074



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 to 0 | CLAMPW | — | R/W | Available | CLAMP Signal Width To enable bit 11, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 11 cannot be written to. To enable bit 12, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12 cannot be written to. These bits are used to set the high-level width of the CLAMP signal in dot clock units. When the CLAMP signal is high, if the HSYNC signal falls, the CLAMP signal falls. The set value is retained at a reset. |

(13) DE Signal Start Register n (DESRn)

Address: DU0: H'FEB0_0078, DU1: H'FEB3_0078, DU2: H'FEB4_0078, DU3: H'FEB7_0078



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 to 0 | DES | — | R/W | Available | DE Signal Start To enable bit 11, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 11 cannot be written to. To enable bit 12, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12 cannot be written to. These bits are used to set the DE signal rising edge position in dot clock units, using as the reference the HSYNC signal falling edge. The DE signal rises (setting value + 1) cycle after the HSYNC signal falls. Therefore, the DE signal cannot rise in the same cycle as the HSYNC signal falling edge. During a vertical blanking period, the DE signal is fixed to the low level. The set value is retained at a reset. |

(14) DE Signal Width Register n (DEWRn)

Address: DU0: H'FEB0_007C, DU1: H'FEB3_007C, DU2: H'FEB4_007C, DU3: H'FEB7_007C

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | DEW | | | | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 to 0 | DEW | — | R/W | Available | DE Signal Width To enable bit 11, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 11 cannot be written to. To enable bit 12, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12 cannot be written to. These bits are used to set the high-level width of the DE signal in dot clock units. When the DE signal is high, if the HSYNC signal falls, the DE signal falls. The set value is retained at a reset. |

36.2.3 Display Attribute Registers

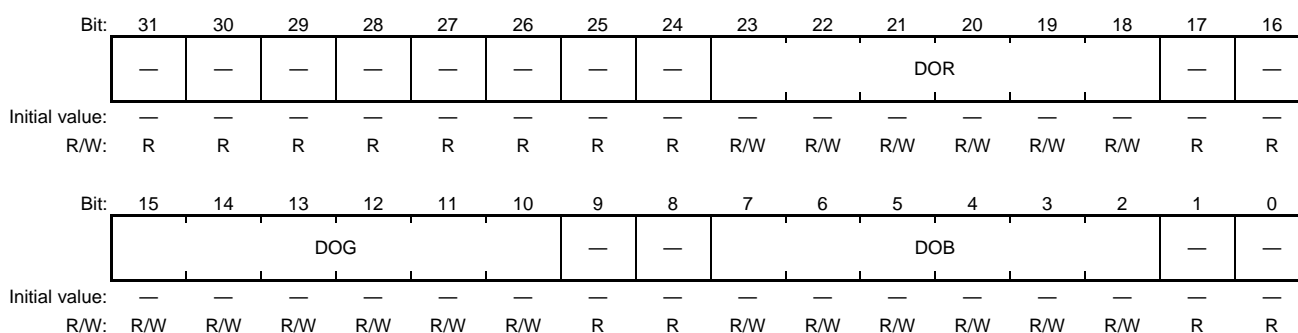
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: All registers in section 36.2.3 are common to RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E.

The sets of display attribute registers are for the respective channels and have the same functions; they are described as one here.

(1) Display Off Mode Output Register n (DOORn)

Address: DU0: H'FEB0_0090, DU1: H'FEB3_0090, DU2: H'FEB4_0090, DU3: H'FEB7_0090

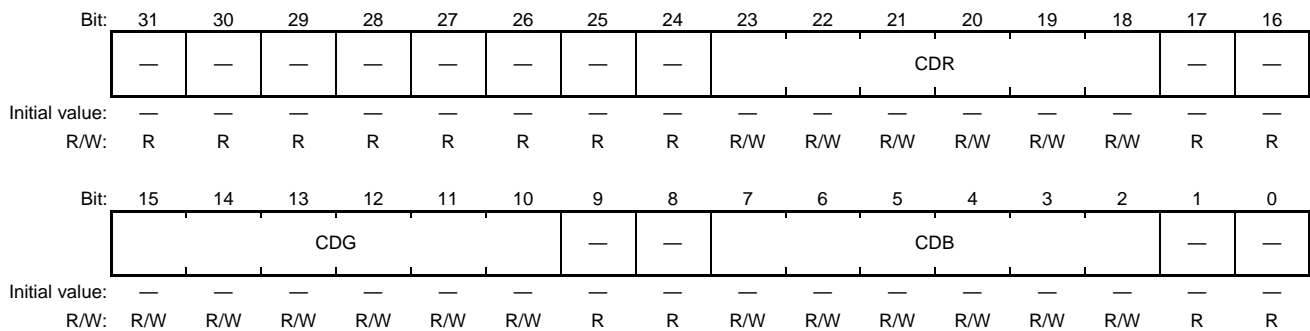


| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 24 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 23 to 18 | DOR | — | R/W | Available | Display Off Mode Output Red These bits are used to set the red display data to be output when the display is off (DRES and DEN bits in DSYSRn are B'00). The set value is retained at a reset. |
| 17, 16 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 15 to 10 | DOG | — | R/W | Available | Display Off Mode Output Green These bits are used to set the green display data to be output when the display is off (DRES and DEN bits in DSYSRn are B'00). The set value is retained at a reset. |
| 9, 8 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 7 to 2 | DOB | — | R/W | Available | Display Off Mode Output Blue These bits are used to set the blue display data to be output when the display is off (DRES and DEN bits in DSYSRn are B'00). The set value is retained at a reset. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|------|----------|---------------|-----|-----------------|--|
| 1, 0 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

(2) Color Detection Register n (CDERn)

Address: DU0: H'FEB0_0094, DU1: H'FEB3_0094, DU2: H'FEB4_0094, DU3: H'FEB7_0094

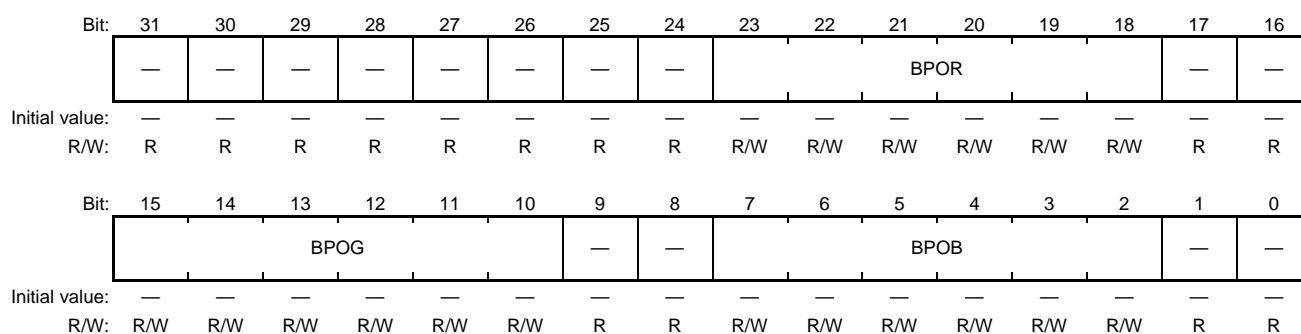


| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 24 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 23 to 18 | CDR | — | R/W | Available | Color Detection Red These bits are used to set the red data for color detection. The set value is retained at a reset. |
| 17, 16 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 15 to 10 | CDG | — | R/W | Available | Color Detection Green These bits are used to set the green data for color detection. The set value is retained at a reset. |
| 9, 8 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 7 to 2 | CDB | — | R/W | Available | Color Detection Blue These bits are used to set the blue data for color detection. The set value is retained at a reset. |
| 1, 0 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

Note: When output data matches the setting value of this register, the high level is output from the CDE pin. For details about the output color data format, see section 36.3.5 Data Formats for Output.

(3) Background Plane Output Register n (BPORn)

Address: DU0: H'FEB0_0098, DU1: H'FEB3_0098, DU2: H'FEB4_0098, DU3: H'FEB7_0098



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 24 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 23 to 18 | BPOR | — | R/W | Available | Background Plane Output Red These bits are used to set the red color to be displayed if no plane to be displayed exists due to a display size, transparent color, and so on. The set value is retained at a reset. |
| 17, 16 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 15 to 10 | BPOG | — | R/W | Available | Background Plane Output Green These bits are used to set the green color to be displayed if no plane to be displayed exists due to a display size, transparent color, and so on. The set value is retained at a reset. |
| 9, 8 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 7 to 2 | BPOB | — | R/W | Available | Background Plane Output Blue These bits are used to set the blue color to be displayed if no plane to be displayed exists due to a display size, transparent color, and so on. The set value is retained at a reset. |
| 1, 0 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

(4) Raster Interrupt Offset Register n (RINTOFSRn)

Address: DU0: H'FEB0_009C, DU1: H'FEB3_009C, DU2: H'FEB4_009C, DU3: H'FEB7_009C

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|---------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | RINTOFS | | | | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 to 0 | RINTOFS | — | R/W | Available | Raster Interrupt Offset To enable bit 10, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 10 cannot be written to. To enable bit 12, 11, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12, 11 cannot be written to. These bits are used to set the raster offset value (number of Hs) that is based on the number of rasters set with the vertical display start register n (VDSRn). If the offset value is assumed to be n, bit 9 in DSSRn is set to 1 at the HSYNC falling edge after the horizontal display period of (VDS + n-the raster). The set value is retained at a reset. |

36.2.4 Display Plane Registers

In descriptions of registers that are common to planes 1 and 3, the planes are generically referred to as plane p. The meanings of characters # is given below.

#: Replaces p (in hexadecimal) in addresses. For example, address H'FEB0_0#00 for the plane 3 mode register corresponds to H'FEB0_0300.

(1) Plane p Mode Register m (PpMRm)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB0_0#00, DU2/DU3: H'FEB4_0#00

| | | | | | | | | | | | | | | | | |
|----------------|----|-----|--------|-----|----|----|----|----|----|----|----|----|----|----|-----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | | PpSPIM | | — | — | — | — | — | — | — | — | — | — | | PpDDF |
| Initial value: | — | 0 | 0 | 0 | — | — | — | — | — | — | — | — | — | — | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 15 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 14 to 12 | PpSPIM | B'000 | R/W | Available | <p>Plane p Super Impose Mode [(DU0/DU1)]</p> <p>B'000: Setting prohibited.</p> <p>B'001: Setting prohibited.</p> <p>B'010: Setting prohibited.</p> <p>B'011: Setting prohibited.</p> <p>B'100: Transparent color processing is not performed for plane p. Plane p is displayed.</p> <p>B'101: α blending of plane p and the lower plane is performed. The transparent color specification for plane p is ignored, and α blending is performed between all the pixels of plane p and the lower plane.</p> <p>B'110: An EOR operation is performed on plane p and the lower plane. The transparent color specification for plane p is ignored, and EOR operation is performed on all the pixels of plane p and the lower plane.</p> <p>B'111: Setting prohibited</p> <hr/> <p>Plane 1 Super Impose Mode [RZ/G2M V1.3, RZ/G2M V3.0(DU2)]</p> <p>B'000: Setting prohibited.</p> <p>B'001: Setting prohibited.</p> <p>B'010: Setting prohibited.</p> <p>B'011: Setting prohibited.</p> <p>B'100: Transparent color processing is not performed for plane 1. Plane 1 is displayed.</p> <p>B'101: Setting prohibited.</p> <p>B'110: Setting prohibited.</p> <p>B'111: Setting prohibited</p> <hr/> <p>Plane 3 Super Impose Mode [RZ/G2H, RZ/G2N(DU3)]</p> <p>B'000: Setting prohibited.</p> <p>B'001: Setting prohibited.</p> <p>B'010: Setting prohibited.</p> <p>B'011: Setting prohibited.</p> <p>B'100: Transparent color processing is not performed for plane 3. Plane 3 is displayed.</p> <p>B'101: Setting prohibited.</p> <p>B'110: Setting prohibited.</p> <p>B'111: Setting prohibited</p> <hr/> |
| 11 to 2 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|------|----------|---------------|-----|-----------------|--|
| 1, 0 | PpDDF | B'00 | R/W | Available | <p>Plane p Display Data Format</p> <p>B'00: Setting prohibited.</p> <p>B'01: 16-bit/pixel</p> <p>B'10: Setting prohibited.</p> <p>B'11: YC.</p> <p>For RZ/G2H, RZ/G2N, bits 9 and 8 in DEFR3 are set to B'10 or B'11 and bit 15 and bit 14 in DEF10R2 set to 1.</p> <p>For RZ/G2M V1.3, RZ/G2M V3.0, bits 9 and 8 in DEFR2 are set to B'10 or B'11 and bit 11 and bit 10 in DEF10R2 set to 1.</p> <p>For RZ/G2E, when YC is output from DU0, Bits 9 and 8 in DEFR0 are set to B'10 or B'11 and bit 11 and bit 10 in DEF10R0 set to 1. Alternatively, when YC is output from DU1, bits 9 and 8 in DEFR1 are set to B'10 or B'11 and bit 15 and bit 14 in DEF10R0 set to 1.</p> <p>In the case of 32-bit/pixel data, set these bits to B'01.</p> |

(2) Plane p Display Size X Register m (PpDSXRm)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB0_0#10, DU2/DU3: H'FEB4_0#10

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | PpDSX | | | | | | | | | | | | | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 to 0 | PpDSX | — | R/W | Available | Plane p Display Size X To enable bit 11, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 11 cannot be written to. To enable bit 12, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12 cannot be written to. The horizontal-direction display size of plane p should be set in dot clock units. Note: When YC has been selected by the PpDDF bits in PpMRm, this value should be set to an even number. The value is retained during a reset. |

(3) Plane p Display Size Y Register m (PpDSYRm)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB0_0#14, DU2/DU3: H'FEB4_0#14

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | PpDSY | | | | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 to 0 | PpDSY | — | R/W | Available | Plane p Display Size Y To enable bit 10, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 10 cannot be written to. To enable bit 12, 11, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12, 11 cannot be written to. The vertical-direction display size of plane p should be set in raster line units. The value is retained during a reset. |

(4) Plane p Display Position X Register m (PpDPXRm)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB0_0#18, DU2/DU3: H'FEB4_0#18

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | PpDPX | | | | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 12 to 0 | PpDPX | — | R/W | Available | Plane p Display Position X To enable bit 11, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 11 cannot be written to. To enable bit 12, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12 cannot be written to. The horizontal start position on the display monitor of plane p should be set in dot clock units, taking as the origin the upper-left corner of the display monitor. Notes: 1. For RZ/G2H, RZ/G2N, when YC has been selected by the DODF1 bit in DEFR3 during display output from DU3, this value should be set to an even number. 2. For RZ/G2M V1.3, RZ/G2M V3.0, when YC has been selected by the DODF bit in DEFR2 during display output from DU2, this value should be set to an even number. 3. For RZ/G2E, when YC has been selected by the DODF bit in DEFR0 during display output from DU0, alternatively, by the DODF bit in DEFR1 during display output from DU1, this value should be set to an even number. |

(5) Plane p Display Position Y Register m (PpDPYRm)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB0_0#1C, DU2/DU3: H'FEB4_0#1C

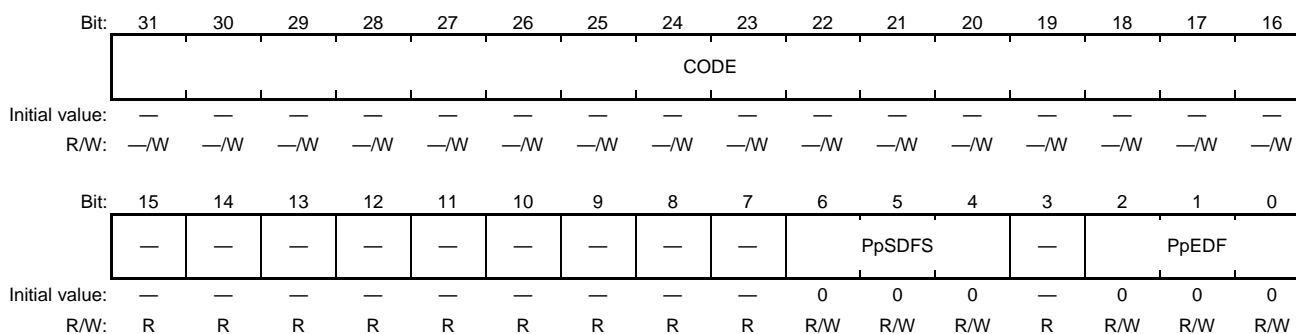
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | PpDPY | | | | | | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description | | | | | | | | | | | | |
|----------|-------------------------|---------------|-----|-----------------|--|-------|-------------------------|---|---|---|---|---|---|---|---|---|---|
| 31 to 13 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. | | | | | | | | | | | | |
| 12 to 0 | PpDPY | — | R/W | Available | <p>Plane p Display Position Y [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>To enable bit 10, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 10 cannot be written to.</p> <p>To enable bit 12, 11, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12, 11 cannot be written to.</p> <p>The vertical start position on the display monitor of plane p should be set in raster line units, taking as the origin the upper-left corner of the display monitor.</p> <p>For interlaced sync & video display, the display starts at the position one bit shifted from the PpDPY bit setting value.</p> <table border="0"> <tr> <td>PpDPY</td> <td>Vertical start position</td> </tr> <tr> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> </tr> <tr> <td>2</td> <td>1</td> </tr> <tr> <td>3</td> <td>1</td> </tr> <tr> <td>4</td> <td>2</td> </tr> </table> <p>For non-interlaced and interlaced sync display, the display starts at the position specified by the PpDPY bit.</p> <p>The value is retained during a reset.</p> <hr/> <p>Plane p Display Position Y [RZ/G2E]</p> <p>To enable bit 10, set the DEFE5 bit in DEF5Rm to 1. In the initial state, bit 10 cannot be written to.</p> <p>To enable bit 12, 11, set the DEFE10 bit in DEF10Rm to 1. In the initial state, bit 12, 11 cannot be written to.</p> <p>The vertical start position on the display monitor of plane p should be set in raster line units, taking as the origin the upper-left corner of the display monitor.</p> <p>For non-interlaced, the display starts at the position specified by the PpDPY bit.</p> <p>The value is retained during a reset.</p> | PpDPY | Vertical start position | 0 | 0 | 1 | 0 | 2 | 1 | 3 | 1 | 4 | 2 |
| PpDPY | Vertical start position | | | | | | | | | | | | | | | | |
| 0 | 0 | | | | | | | | | | | | | | | | |
| 1 | 0 | | | | | | | | | | | | | | | | |
| 2 | 1 | | | | | | | | | | | | | | | | |
| 3 | 1 | | | | | | | | | | | | | | | | |
| 4 | 2 | | | | | | | | | | | | | | | | |

(6) Plane p Display Data Control 4 Register m (PpDDC4Rm)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB0_0#90, DU2/DU3: H'FEB4_0#90



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 16 | CODE | — | —/W | Not available | PpDDC4Rm Enabling Code (register available code) For a value written to PpDDC4Rm to be effective, the value must include H'7766 in these bits. |
| 15 to 7 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 6 to 4 | PpSDFS | B'000 | R/W | Available | Plane p Superimpose Data Format Select B'100: Setting prohibited. B'101: YC composite. After YC composite. YC-RGB conversion is performed. B'110: Setting prohibited. B'111: Setting prohibited. Others: RGB composite. When performing YC-RGB conversion, set desired values in the YCRGB1 and YCRGB0 bits in DEF5Rm. YC-RGB conversion cannot be performed with only the settings of this register. When YC composite has been set, YC composite should also be set for the superimposed lower plane. |
| 3 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 2 to 0 | PpEDF | B'000 | R/W | Available | Plane p Extensional Data Format B'000: The data format will be determined by the PpDDF bits in PpMRm B'001: ARGB8888 B'010: RGB888 B'011: RGB666 B'100: YCbCr444 B'101: Setting prohibited B'110: Setting prohibited B'111: Setting prohibited |

36.2.5 Dual Display Output Control Registers

(1) Display Unit Output Route Control Register 0 (DORCR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB1_1000

| | | | | | | | | | | | | | | | | |
|----------------|----|------|----|------|----|----|------|-----|----|------|-----|----|----|----|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | PG1T | — | DK1S | — | — | PG1D | — | — | DR0D | — | — | — | — | PG0D | — |
| Initial value: | — | 1 | — | 1 | — | — | 0 | 1 | — | — | 0 | — | — | — | 0 | 0 |
| R/W: | R | R/W | R | R/W | R | R | R/W | R/W | R | R | R/W | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | DPRS |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|---|
| 31 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 30 | PG1T | B'1 | R/W | DRES | Pin Generate 1 Timing Select Selects the source of timing for pin controller 1. 0: Display-timing generator 0. 1: Display-timing generator 1 |
| 29 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 28 | DK1S | B'1 | R/W | DRES | Dot Clock Select 1 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 0: Dot-clock generator 0 supplies the dot clock for display-timing generator 1 and pin controller 1. 1: Dot-clock generator 1 supplies the dot clock for display-timing generator 1 and pin controller 1. When the TVM1 bits in DSYSR1 are set to TV synchronized mode, do not set this bit to 0. Dot Clock Select 1 [RZ/G2E] 0: Dot-clock generator 0 supplies the dot clock for display-timing generator 1 and pin controller 1. 1: Dot-clock generator 1 supplies the dot clock for display-timing generator 1 and pin controller 1. |
| 27, 26 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 25, 24 | PG1D | B'01 | R/W | DRES | <p>Pin Generate 1 Input Data Select</p> <p>Selects the source of input data for pin controller 1.</p> <p>B'00: Data from superimposition processor 0 are input to pin controller 1. For the timing of superimposition processor 0, refer to section 36.3.8(3), Combinations of Blocks for Dual Display Output.</p> <p>B'01: Data from superimposition processor 1 are input to pin controller 1. For the timing of superimposition processor 1, refer to section 36.3.8(3), Combinations of Blocks for Dual Display Output.</p> <p>B'10: Input to pin controller 1 is fixed to 0. The value of the CDE pin is fixed to 0.</p> <p>B'11: The value of DOOR1 is input to pin controller 1. The value of the CDE pin is fixed to 0.</p> <p>The combination of the DRES and DEN bits in DSYSR0 determines the data to be superimposed.</p> <p>DRES/DEN = 00: The value of DOOR1</p> <p>DRES/DEN = 01: The data in unified memory</p> <p>DRES/DEN = 10: 0</p> <p>DRES/DEN = 11: Setting prohibited (data: 0)</p> |
| 23, 22 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 21 | DR0D | B'0 | R/W | DRES | <p>Display Output Route 0 Data Select</p> <p>0: Data from pin controller 0 are output from the LVDS-IF data.</p> <p>1: On the LVDS-IF data, data from pin controller 0 are output on rising edges of the output dot clock and data from pin controller 1 are output on falling edges of the output dot clock. If this setting is made, the frequency of the output dot clock has to be at least half of the frequency of the input dot clock.</p> |
| 20 to 18 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 17, 16 | PG0D | B'00 | R/W | DRES | <p>Pin Generate 0 Input Data Select</p> <p>Selects the source of input data for pin controller 0.</p> <p>B'00: Data from superimposition processor 0 are input to pin controller 0. For the timing of superimposition processor 0, refer to section 36.3.8(3), Combinations of Blocks for Dual Display Output.</p> <p>B'01: Data from superimposition processor 1 are input to pin controller 0. For the timing of superimposition processor 1, refer to section 36.3.8(3), Combinations of Blocks for Dual Display Output.</p> <p>B'10: Input to pin controller 0 is fixed to 0. The value of the CDE pin is fixed to 0.</p> <p>B'11: The value of DOOR0 is input to pin controller 0. The value of the CDE pin is fixed to 0.</p> <p>The combination of the DRES and DEN bits in DSYSR0 determines the data to be superimposed.</p> <p>DRES/DEN = 00: The value of DOOR0</p> <p>DRES/DEN = 01: The data in unified memory</p> <p>DRES/DEN = 10: 0</p> <p>DRES/DEN = 11: Setting prohibited (data: 0)</p> |
| 15 to 1 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|-----|----------|---------------|-----|-----------------|--|
| 0 | DPRS | B'0 | R/W | DRES | Display Priority Register Select 0: The order of priority for the planes is set in DPPR0. Superimposition processor 0 can be used to superimpose planes 1 and 3. Superimposition processor 1 is not available. 1: The order of priority for the planes is set in DSPR0 or DSPR1. Superimposition processors 0 and 1 can be used to superimpose planes 1 and 3. |

(2) Display Unit Output Route Control Register 2 (DORCR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Address: DU2/DU3: H'FEB5_1000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R1 | R | R1 | R | R | R | R1 | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | DPRS |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 30 | — | — | R1 | — | Reserved The read value is undefined. The write value should always be 1. |
| 29 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 28 | — | — | R1 | — | Reserved The read value is undefined. The write value should always be 1. |
| 27 to 25 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 24 | — | — | R1 | — | Reserved The read value is undefined. The write value should always be 1. |
| 23 to 1 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 0 | DPRS | B'0 | R/W | DRES | Display Priority Register Select 0: For RZ/G2M V1.3, RZ/G2M V3.0, the order of priority for the planes is set in DPPR2. Superimposition processor 2 can be used to superimpose planes 1. For RZ/G2H, RZ/G2N, superimposition processor 3 is not available. 1: For RZ/G2M V1.3, RZ/G2M V3.0, the order of priority for the planes is set in DSPR2. Superimposition processor 2 can be used to superimpose planes 1. For RZ/G2H, RZ/G2N, the order of priority for the planes is set in DSPR3. Superimposition processors 3 can be used to superimpose planes 3. |

(3) Display Unit Plane Timing Select Register 0 (DPTSR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0/DU1: H'FEB1_1004

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|------|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | P3DK | — | P1DK |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | 0 | — | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | P3TS | — | P1TS |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | 0 | — | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 20 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 19 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 18 | P3DK | B'0 | R/W | DRES | Plane 3 Dot Clock Select 0: Dot-clock generator 0 1: Dot-clock generator 1 |
| 17 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 16 | P1DK | B'0 | R/W | DRES | Plane 1 Dot Clock Select 0: Dot-clock generator 0 1: Dot-clock generator 1 |
| 15 to 4 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 3 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 2 | P3TS | B'0 | R/W | DRES | Plane 3 Timing Select 0: Display-timing generator 0 1: Display-timing generator 1 |
| 1 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 0 | P1TS | B'0 | R/W | DRES | Plane 1 Timing Select 0: Display-timing generator 0 1: Display-timing generator 1 |

(4) Display Unit Plane Timing Select Register 2 (DPTSR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

Address: DU2/DU3: H'FEB5_1004

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|------|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | P3DK | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | 0 | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | P3TS | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | 0 | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 19 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 18 | P3DK | B'0 | R/W | DRES | Plane 3 Dot Clock Select 0: Setting prohibited 1: Dot-clock generator 3 |
| 17 to 16 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 15 to 4 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 3 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 2 | P3TS | B'0 | R/W | DRES | Plane 3 Timing Select 0: Setting prohibited 1: Display-timing generator 3 |
| 1 to 0 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

(5) Display Superimpose Priority Register 0 (DSPR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU0: H'FEB1_1020

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------|------|------|------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | S0S2 | | | S0S1 | | | | |
| Initial value: | — | — | — | — | — | — | — | — | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W* | R/W* | R/W* | R/W* | R/W | R/W | R/W | R/W |

- The setting of this register is valid when the DPRS bit in DORCR0 is 1.
- After setting a desired value in a register listed in section 36.2.4 Display Plane Registers, set DSPR0 to the value indicating the plane number (B'0001 and B'0011).

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|---------|----------|---------------|-----|-----------------|--|
| 31 to 8 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 7 to 4 | S0S2 | B'0000 | R/W | Available | Display Superimposition 0 Priority 2 Select B'0000: Priority level 2 is not used in display generation by superimposition processor 0. B'0001: Plane 1 has priority level 2 in display generation by superimposition processor 0. B'0010: Setting prohibited. B'0011: Plane 3 has priority level 2 in display generation by superimposition processor 0. Others: Setting prohibited |
| 3 to 0 | S0S1 | B'0000 | R/W | Available | Display Superimposition 0 Priority 1 Select B'0000: Priority level 1 is not used in display generation by superimposition processor 0. B'0001: Plane 1 has priority level 1 in display generation by superimposition processor 0. B'0010: Setting prohibited. B'0011: Plane 3 has priority level 1 in display generation by superimposition processor 0. Others: Setting prohibited |

(6) Display Superimpose Priority Register 2 (DSPR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | — | — |

Address: DU2: H'FEB5_1020

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | S0S1 | | | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

- The setting of this register is valid when the DPRS bit in DORCR2 is 1.
- After setting a desired value in a register listed in section 36.2.4 Display Plane Registers, set DSPR2 to the value indicating the plane number (B'0001).

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|---------|----------|---------------|-----|-----------------|---|
| 31 to 8 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 7 to 4 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 3 to 0 | S0S1 | B'0000 | R/W | Available | Display Superimposition 0 Priority 1 Select B'0000: Priority level 1 is not used in display generation by superimposition processor 2. B'0001: Plane 1 has priority level 1 in display generation by superimposition processor 2. Others: Setting prohibited |

(7) Display Superimpose Priority Register 1 (DSPR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address: DU1: H'FEB1_1024

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------|-----|-----|------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | S1S2 | | | S1S1 | | | | |
| Initial value: | — | — | — | — | — | — | — | — | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

- The setting of this register is valid when the DPRS bit in DORCR0 is 1.
- After setting a desired value in a register listed in section 36.2.4 Display Plane Registers, set DSPR1 to the value indicating the plane number (B'0001 and B'0011).

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|---------|----------|---------------|-----|-----------------|--|
| 31 to 8 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 7 to 4 | S1S2 | B'0000 | R/W | Available | Display Superimposition 1 Priority 2 Select B'0000: Priority level 2 is not used in display generation by superimposition processor 1. B'0001: Plane 1 has priority level 2 in display generation by superimposition processor 1. B'0010: Setting prohibited. B'0011: Plane 3 has priority level 2 in display generation by superimposition processor 1. Others: Setting prohibited |
| 3 to 0 | S1S1 | B'0000 | R/W | Available | Display Superimposition 1 Priority 1 Select B'0000: Priority level 1 is not used in display generation by superimposition processor 1. B'0001: Plane 1 has priority level 1 in display generation by superimposition processor 1. B'0010: Setting prohibited. B'0011: Plane 3 has priority level 1 in display generation by superimposition processor 1. Others: Setting prohibited |

(8) Display Superimpose Priority Register 3 (DSPR3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

Address: DU3: H'FEB5_1024

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | S1S1 | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

- The setting of this register is valid when the DPRS bit in DORCR2 is 1.
- After setting a desired value in a register listed in section 36.2.4 Display Plane Registers, set DSPR3 to the value indicating the plane number (B'0011[RZ/G2H, RZ/G2N]).

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|---------|----------|---------------|-----|-----------------|---|
| 31 to 4 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 3 to 0 | S1S1 | B'0000 | R/W | Available | Display Superimposition 1 Priority 1 Select B'0000: Priority level 1 is not used in display generation by superimposition processor 3. B'0001: Setting prohibited. B'0010: Setting prohibited. B'0011: Plane 3 has priority level 1 in display generation by superimposition processor 3. Others: Setting prohibited |

36.2.6 External Synchronization Control Registers

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: All registers in section 36.2.6 are common to RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E

The sets of display timing generation registers are for the respective channels and have the same functions; they are described as one here.

(1) External Synchronization Control Register n (ESCRn)

Note: n = 0 and 3 [RZ/G2H, RZ/G2N]
n = 0 and 2 [RZ/G2M V1.3, RZ/G2M V3.0]
n = 0 and 1 [RZ/G2E]

Address: DU0: H'FEB1_0000, DU1: H'FEB3_1000, DU2: H'FEB5_0000, DU3: H'FEB7_1000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-------------|-------------|----|----|-----|-------------|-----|-----|-----|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | DCKOIN V | DCKOS EL | — | — | — | DCLKS EL | — | — | — | DCLKD IS |
| Initial value: | — | — | — | — | — | — | 0 | 0 | — | — | — | 0 | — | — | — | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R/W | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | SYNCSEL/—* | — | — | — | — | FRQSEL | | | | |
| Initial value: | — | — | — | — | — | — | 0/—* | 0/—* | — | — | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W* | R/W* | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * For reserved bit (bit name: —), initial value is undefined, writing is not available and writing value should be specified.
For details of bit description, refer to following table.

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 26 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 25 | DCKOINV | B'0 | R/W | Not available | DCLKOUT Invert To enable this bit, set the DEFE bit in DEFERn to 1. In the initial state, this bit is fixed to 0. 0: The DCLKOUT is output in normal phase. 1: The DCLKOUT is output in counter phase. For RZ/G2H, RZ/G2N, this bit is provided in ESCR3. For RZ/G2M V1.3, RZ/G2M V3.0, this bit is provided in ESCR2. For RZ/G2E, this bit is provided in ESCR1 or ESCR0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 24 | DCKOSEL | B'0 | R/W | Not available | <p>Output Dot Clock Select (DCLKOUT Select)</p> <p>To enable this bit, set DEFE to 1 in DEFRn. In the initial state, this bit is fixed at 0.</p> <p>Set this bit to 1 to specify display output in YC data format.</p> <p>0: The DCLKOUT division ratio is determined by bits 5 to 0 in this register.</p> <p>1: The DCLKIN is used as the DCLKOUT, regardless of the division ratio.</p> <p>For RZ/G2H, RZ/G2N, this bit is provided in ESCR3.</p> <p>For RZ/G2M V1.3, RZ/G2M V3.0, this bit is provided in ESCR2.</p> <p>For RZ/G2E, this bit is provided in ESCR1 or ESCR0.</p> |
| 23 to 21 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 20 | DCLKSEL | B'0 | R/W | Not available | <p>DCLKIN Select [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>0: The input dot clock source is the DCLKIN pin</p> <p>1: The input dot clock source is the S2D1ϕ clock.</p> <p>If this setting is made, ensure that the frequency of the input dot clock is divided by two or a greater value (so that the result of dividing the frequency is more than or equal to half the S2D1ϕ frequency).</p> <hr/> <p>DCLKIN Select [RZ/G2E]</p> <p>0: The input dot clock source is the DCLKIN pin</p> <p>1: The input dot clock source is the S1D1ϕ clock.</p> <p>If this setting is made, ensure that the frequency of the input dot clock is divided by two or a greater value (so that the result of dividing the frequency is more than or equal to half the S1D1ϕ frequency).</p> |
| 19 to 17 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 16 | DCLKDIS | B'0 | R/W | Not available | <p>DCLKOUT Disable</p> <p>0: The DCLKOUT is output.</p> <p>1: The DCLKOUT is not output.</p> <p>The DCLKOUT is fixed to low level.</p> <p>For RZ/G2H, RZ/G2N, this bit is provided in ESCR3.</p> <p>For RZ/G2M V1.3, RZ/G2M V3.0, this bit is provided in ESCR2.</p> <p>For RZ/G2E, this bit is provided in ESCR1 or ESCR0.</p> |
| 15 to 10 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|--|
| 9, 8 | SYNCSEL | B'00 | R/W | Not available | <p>SYNC Select t [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>B'00: Phases are not synchronized</p> <p>B'01: Phases are not synchronized</p> <p>B'10: Phases are synchronized by using the EXVSYNC signal</p> <p>B'11: Phases are synchronized by using the EXHSYNC signal</p> |
| | — | — | R | — | <p>Reserved [RZ/G2E]</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 7, 6 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 5 to 0 | FRQSEL | B'00_0000 | R/W | Not available | <p>Frequency Select</p> <p>To enable bit 5, the DEFE bit in DEFRn must be set to 1. In the initial state, bit 5 is fixed to 0.</p> <p>If frequency division is by an odd number, the duty cycle of the frequency divided dot clock will be below 50%.</p> <p>For RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, these bits are not provided in ESCR1.</p> <p>B'00_0000: Frequency division of the input dot clock (clock for division) is not performed. Dot clock is the DCLKIN pin.</p> <p>B'00_0001: Division by 2 of the input dot clock (clock for division)</p> <p>B'00_0010: Division by 3 of the input dot clock (clock for division)</p> <p> : Division by (FRQSEL+1) of the input dot clock</p> <p> : (clock for division)</p> <p>B'11_1111: Division by 64 of the input dot clock (clock for division)</p> |

(2) Output Signal Timing Adjustment Register n (OTARn)

Note: n = 3 [RZ/G2H, RZ/G2N]
 n = 2 [RZ/G2M V1.3, RZ/G2M V3.0]
 n = 0 and 1 [RZ/G2E]

Address: DU0: H'FEB1_0004, DU1: H'FEB3_1004, DU2: H'FEB5_0004, DU3: H'FEB7_1004

| | | | | | | | | | | | | | | | | |
|----------------|----|-----|-----|-----|----|--------|-----|-----|----|-------|-----|-----|----|-------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | DEA | | | — | CLAMPA | | | — | DRGBA | | | — | — | — | — |
| Initial value: | — | 0 | 0 | 0 | — | 0 | 0 | 0 | — | 0 | 0 | 0 | — | — | — | — |
| R/W: | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | CDEA | | | — | DISPA | | | — | SYNCA | | |
| Initial value: | — | — | — | — | — | 0 | 0 | 0 | — | 0 | 0 | 0 | — | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 30 to 28 | DEA | B'000 | R/W | DRES | DE Output Timing Adjustment B'000: Adjustment of output timing is not performed. The DE signal is output on the rising edge of the dot clock, with the reference timing. B'001: The DE signal is output on the rising edge, delayed one dot clock cycle relative to the reference timing. B'010: The DE signal is output on the rising edge, delayed two dot clock cycles relative to the reference timing. B'011: The DE signal is output on the rising edge, delayed three dot clock cycles relative to the reference timing. B'100: The DE signal is output on the falling edge, preceding the reference timing by 1/2 dot clock cycle. B'101: The DE signal is output on the falling edge, delayed 1/2 dot clock cycle relative to the reference timing. B'110: The DE signal is output on the falling edge, delayed (1+1/2) dot clock cycles relative to the reference timing. B'111: The DE signal is output on the falling edge, delayed (2+1/2) dot clock cycles relative to the reference timing. |
| 27 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 26 to 24 | CLAMPA | B'000 | R/W | DRES | <p>CLAMP Output Timing Adjustment</p> <p>B'000: Adjustment of output timing is not performed. The CLAMP signal is output on the rising edge of the dot clock, with the reference timing.</p> <p>B'001: The CLAMP signal is output on the rising edge, delayed one dot clock cycle relative to the reference timing.</p> <p>B'010: The CLAMP signal is output on the rising edge, delayed two dot clock cycles relative to the reference timing.</p> <p>B'011: The CLAMP signal is output on the rising edge, delayed three dot clock cycles relative to the reference timing.</p> <p>B'100: The CLAMP signal is output on the falling edge, preceding the reference timing by 1/2 dot clock cycle.</p> <p>B'101: The CLAMP signal is output on the falling edge, delayed 1/2 dot clock cycle relative to the reference timing.</p> <p>B'110: The CLAMP signal is output on the falling edge, delayed (1+1/2) dot clock cycles relative to the reference timing.</p> <p>B'111: The CLAMP signal is output on the falling edge, delayed (2+1/2) dot clock cycles relative to the reference timing.</p> |
| 23 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 22 to 20 | DRGBA | B'000 | R/W | DRES | <p>Digital RGB Output Timing Adjustment</p> <p>B'000: Adjustment of output timing is not performed. The digital RGB signal is output on the rising edge of the dot clock, with the reference timing.</p> <p>B'001: The digital RGB signal is output on the rising edge, delayed one dot clock cycle relative to the reference timing.</p> <p>B'010: The digital RGB signal is output on the rising edge, delayed two dot clock cycles relative to the reference timing.</p> <p>B'011: The digital RGB signal is output on the rising edge, delayed three dot clock cycles relative to the reference timing.</p> <p>B'100: The digital RGB signal is output on the falling edge, preceding the reference timing by 1/2 dot clock cycle.</p> <p>B'101: The digital RGB signal is output on the falling edge, delayed 1/2 dot clock cycle relative to the reference timing.</p> <p>B'110: The digital RGB signal is output on the falling edge, delayed (1+1/2) dot clock cycles relative to the reference timing.</p> <p>B'111: The digital RGB signal is output on the falling edge, delayed (2+1/2) dot clock cycles relative to the reference timing.</p> |
| 19 to 11 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|---------|----------|---------------|-----|-----------------|---|
| 10 to 8 | CDEA | B'000 | R/W | DRES | <p>CDE Output Timing Adjustment</p> <p>B'000: Adjustment of output timing is not performed. The CDE signal is output on the rising edge of the dot clock, with the reference timing.</p> <p>B'001: The CDE signal is output on the rising edge, delayed one dot clock cycle relative to the reference timing.</p> <p>B'010: The CDE signal is output on the rising edge, delayed two dot clock cycles relative to the reference timing.</p> <p>B'011: The CDE signal is output on the rising edge, delayed three dot clock cycles relative to the reference timing.</p> <p>B'100: The CDE signal is output on the falling edge, preceding the reference timing by 1/2 dot clock cycle.</p> <p>B'101: The CDE signal is output on the falling edge, delayed 1/2 dot clock cycle relative to the reference timing.</p> <p>B'110: The CDE signal is output on the falling edge, delayed (1+1/2) dot clock cycles relative to the reference timing.</p> <p>B'111: The CDE signal is output on the falling edge, delayed (2+1/2) dot clock cycles relative to the reference timing.</p> |
| 7 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |
| 6 to 4 | DISPA | B'000 | R/W | DRES | <p>DISP Output Timing Adjustment</p> <p>B'000: Adjustment of output timing is not performed. The DISP signal is output on the rising edge of the dot clock, with the reference timing.</p> <p>B'001: The DISP signal is output on the rising edge, delayed one dot clock cycle relative to the reference timing.</p> <p>B'010: The DISP signal is output on the rising edge, delayed two dot clock cycles relative to the reference timing.</p> <p>B'011: The DISP signal is output on the rising edge, delayed three dot clock cycles relative to the reference timing.</p> <p>B'100: The DISP signal is output on the falling edge, preceding the reference timing by 1/2 dot clock cycle.</p> <p>B'101: The DISP signal is output on the falling edge, delayed 1/2 dot clock cycle relative to the reference timing.</p> <p>B'110: The DISP signal is output on the falling edge, delayed (1+1/2) dot clock cycles relative to the reference timing.</p> <p>B'111: The DISP signal is output on the falling edge, delayed (2+1/2) dot clock cycles relative to the reference timing.</p> |
| 3 | — | — | R | — | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|--------|----------|---------------|-----|-----------------|--|
| 2 to 0 | SYNCA | B'000 | R/W | DRES | <p>SYNC* Output Timing Adjustment</p> <p>B'000: Adjustment of output timing is not performed. The SYNC* signal is output on the rising edge of the dot clock, with the reference timing.</p> <p>B'001: The SYNC* signal is output on the rising edge, delayed one dot clock cycle relative to the reference timing.</p> <p>B'010: The SYNC* signal is output on the rising edge, delayed two dot clock cycles relative to the reference timing.</p> <p>B'011: The SYNC* signal is output on the rising edge, delayed three dot clock cycles relative to the reference timing.</p> <p>B'100: The SYNC* signal is output on the falling edge, preceding the reference timing by 1/2 dot clock cycle.</p> <p>B'101: The SYNC* signal is output on the falling edge, delayed 1/2 dot clock cycle relative to the reference timing.</p> <p>B'110: The SYNC* signal is output on the falling edge, delayed (1+1/2) dot clock cycles relative to the reference timing.</p> <p>B'111: The SYNC* signal is output on the falling edge, delayed (2+1/2) dot clock cycles relative to the reference timing.</p> <p>Note: * HSYNC, VSYNC, CSYNC, and ODDF signals. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>Note: * HSYNC, VSYNC, signals. [RZ/G2E]</p> |

Note: For RZ/G2E, when signals are to be output on the falling edge, the electrical characteristics do not apply.

36.2.7 YC-RGB Conversion Coefficient Registers

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

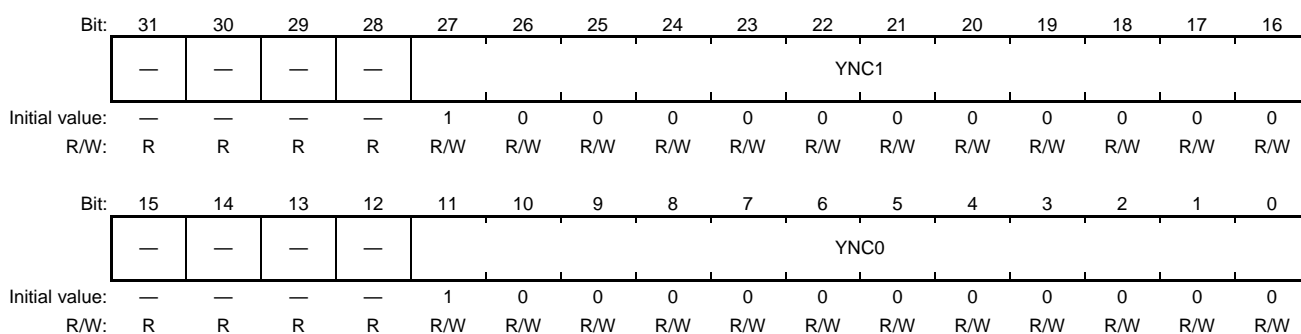
The sets of YC-RGB conversion coefficient registers are for the respective channels and have the same functions; they are described as one here. However, the usable bits by each products are as Table 36.23. The unusable bits are changed to the reserved bits.

Table 36.23 Configuration of YC-RGB Conversion Coefficient Registers

| Second Generation RZ/G Series Products | | | | | | |
|---|---------------|--|--------|----------------------------|--------|--------|
| Suffix m of register | Register bits | Suffix m of superimposition processors | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 0 | 31 to 16 | 1 | √ | √ | √ | √ |
| | 15 to 0 | 0 | √ | √ | √ | √ |
| 2 | 31 to 16 | 3 | √ | — | √ | — |
| | 15 to 0 | 2 | — | √ | — | — |

(1) Y Normalization Coefficient Register m (YNCRm)

Address: DU0/DU1: H'FEB1_4080, DU2/DU3: H'FEB5_4080



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 28 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 27 to 16 | YNC1 | H'800 | R/W | Available | Y Normalization Coefficient 1 This coefficient is for normalization of Y values in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 1. |
| 15 to 12 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 11 to 0 | YNC0 | H'800 | R/W | Available | Y Normalization Coefficient 0 This coefficient is for normalization of Y values in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 1. |

(2) Y Normalization Offset Register m (YNORm)

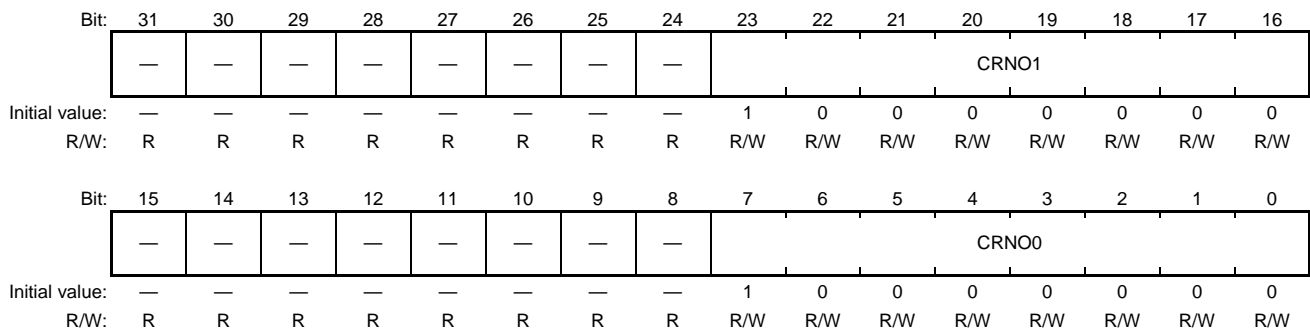
Address: DU0/DU1: H'FEB1_4084, DU2/DU3: H'FEB5_4084

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | YNO1 | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | YNO0 | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 24 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 23 to 16 | YNO1 | H'00 | R/W | Available | Y Normalization Offset 1 This offset is to be subtracted from Y values before they are normalized in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The offset is an unsigned 8-bit integer. Its default value is 0. |
| 15 to 8 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 7 to 0 | YNO0 | H'00 | R/W | Available | Y Normalization Offset 0 This offset is to be subtracted from Y values before they are normalized in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The offset is an unsigned 8-bit integer. Its default value is 0. |

(3) Cr Normalization Offset Register m (CRNORm)

Address: DU0/DU1: H'FEB1_4088, DU2/DU3: H'FEB5_4088



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 24 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 23 to 16 | CRNO1 | H'80 | R/W | Available | Cr Normalization Offset 1 This offset is to be subtracted from Cr values before they are normalized in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The offset is an unsigned 8-bit integer. Its default value is 128. |
| 15 to 8 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 7 to 0 | CRNO0 | H'80 | R/W | Available | Cr Normalization Offset 0 This offset is to be subtracted from Cr values before they are normalized in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The offset is an unsigned 8-bit integer. Its default value is 128. |

(4) Cb Normalization Offset Register m (CBNORM)

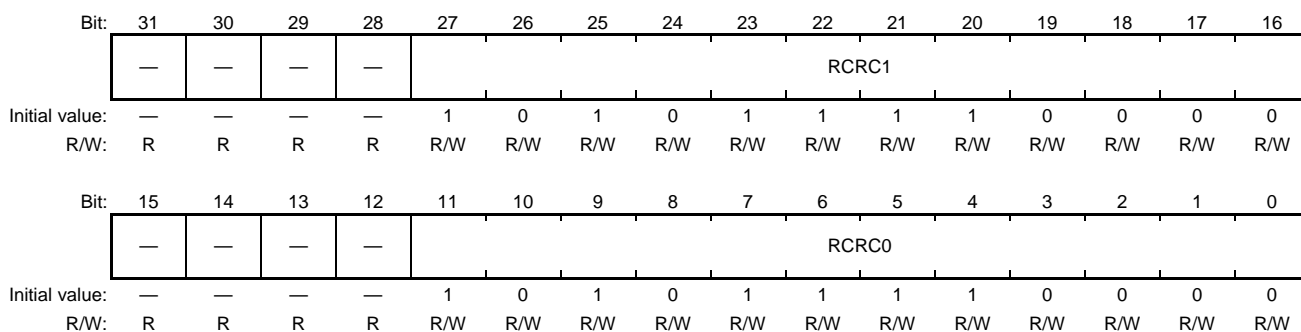
Address: DU0/DU1: H'FEB1_408C, DU2/DU3: H'FEB5_408C

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | CBNO1 | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | CBNO0 | | | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 24 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 23 to 16 | CBNO1 | H'80 | R/W | Available | Cb Normalization Offset 1 This offset is to be subtracted from Cb values before they are normalized in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The offset is an unsigned 8-bit integer. Its default value is 128. |
| 15 to 8 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 7 to 0 | CBNO0 | H'80 | R/W | Available | Cb Normalization Offset 0 This offset is to be subtracted from Cb values before they are normalized in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The offset is an unsigned 8-bit integer. Its default value is 128. |

(5) Red Cr Coefficient Register m (RCRCRm)

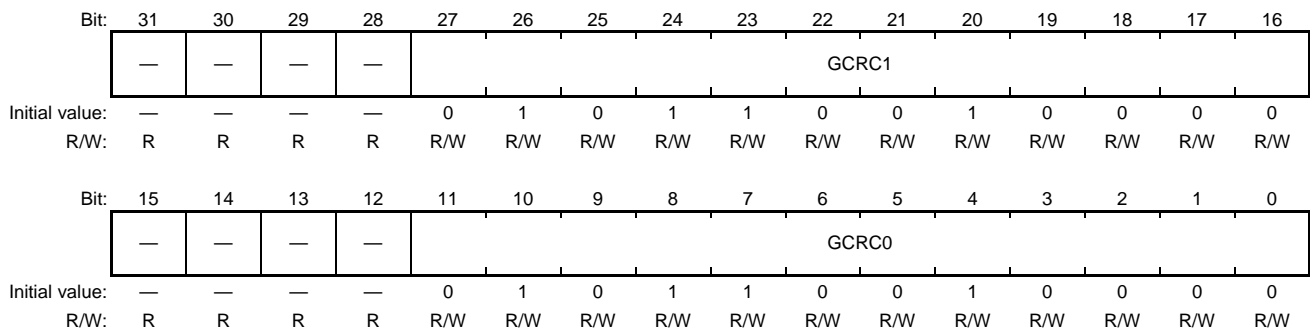
Address: DU0/DU1: H'FEB1_4090, DU2/DU3: H'FEB5_4090



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 28 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 27 to 16 | RCRC1 | H'AF0 | R/W | Available | Red Cr Coefficient 1 Cr is multiplied by this coefficient to create the red component in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 1.37. |
| 15 to 12 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 11 to 0 | RCRC0 | H'AF0 | R/W | Available | Red Cr Coefficient 0 Cr is multiplied by this coefficient to create the red component in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 1.37. |

(6) Green Cr Coefficient Register m (GCRCRm)

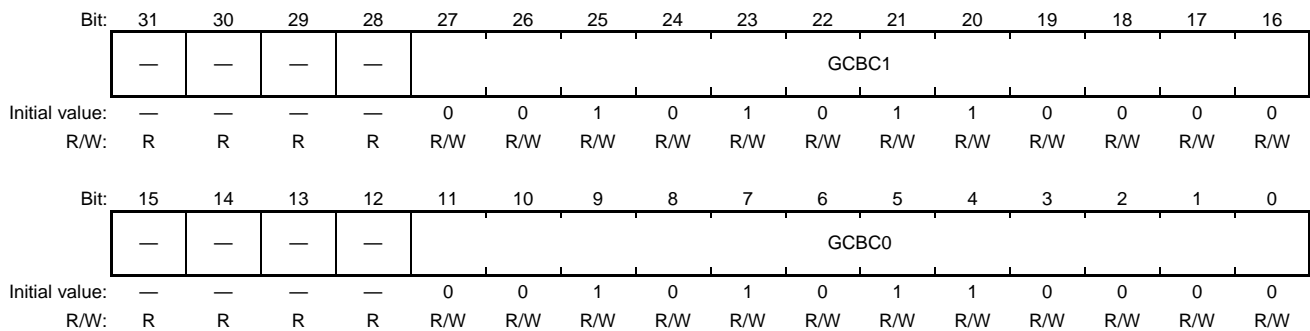
Address: DU0/DU1: H'FEB1_4094, DU2/DU3: H'FEB5_4094



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 28 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 27 to 16 | GCRC1 | H'590 | R/W | Available | Green Cr Coefficient 1 Cr is multiplied by this coefficient to create the green component in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 0.698. |
| 15 to 12 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 11 to 0 | GCRC0 | H'590 | R/W | Available | Green Cr Coefficient 0 Cr is multiplied by this coefficient to create the green component in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 0.698. |

(7) Green Cb Coefficient Register m (GCBCRm)

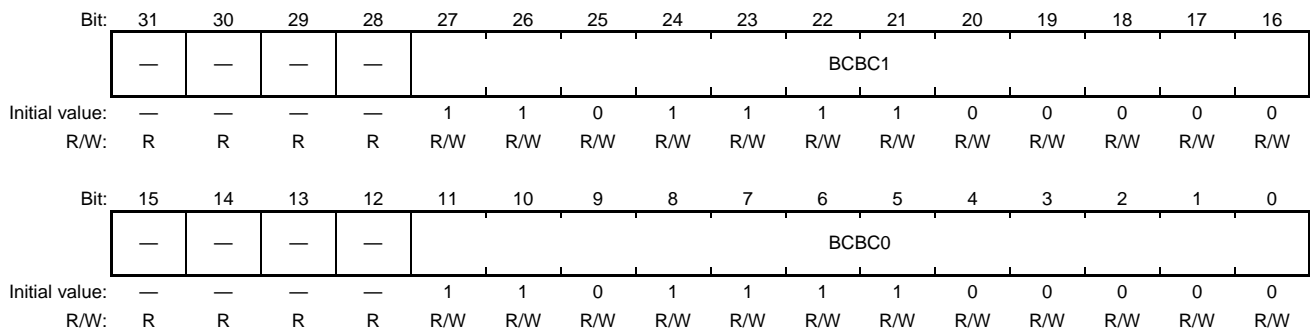
Address: DU0/DU1: H'FEB1_4098, DU2/DU3: H'FEB5_4098



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|--|
| 31 to 28 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 27 to 16 | GCBC1 | H'2B0 | R/W | Available | Green Cb Coefficient 1 Cb is multiplied by this coefficient to create the green component in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 0.336. |
| 15 to 12 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 11 to 0 | GCBC0 | H'2B0 | R/W | Available | Green Cb Coefficient 0 Cb is multiplied by this coefficient to create the green component in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 0.336. |

(8) Blue Cb Coefficient Register m (BCBCRm)

Address: DU0/DU1: H'FEB1_409C, DU2/DU3: H'FEB5_409C



| Bit | Bit Name | Initial Value | R/W | Internal Update | Description |
|----------|----------|---------------|-----|-----------------|---|
| 31 to 28 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 27 to 16 | BCBC1 | H'DE0 | R/W | Available | Blue Cb Coefficient 1 Cb is multiplied by this coefficient to create the blue component in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 1.73. |
| 15 to 12 | — | — | R | — | Reserved The read value is undefined. The write value should always be 0. |
| 11 to 0 | BCBC0 | H'DE0 | R/W | Available | Blue Cb Coefficient 0 Cb is multiplied by this coefficient to create the blue component in YC-RGB conversion by the YC-RGB conversion generation by superimposition processors m. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 1.73. |

36.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

36.3.1 Configuration of Output Screen

The DU executes window displays with up to a maximum of two window layers. Each of these windows is called a "plane", and the order of stacking of the planes can be set arbitrarily. For each plane, display can be turned on and off, and the display data format (32-bit/pixel, YC), blending functions, and other settings can be changed independently. For RZ/G2M V1.3, RZ/G2M V3.0, plane 3 in the DU2 cannot be used and plane 1 superimposing in DU2 cannot be operated. For RZ/G2H, RZ/G2N, plane 1 in the DU3 cannot be used and plane 3 superimposing in DU3 cannot be operated.

Table 36.24 Display Functions of Planes

| | Display On/Off | Display Data Format | | Superimposing | Size |
|---------|----------------|---------------------|--------------|--------------------------|-------------------|
| | | YC | 32-bit/pixel | | |
| Plane 1 | ✓ | ✓ | ✓ | α blending/EOR operation | X, Y (as desired) |
| Plane 3 | ✓ | ✓ | ✓ | Same as above | Same as above |

36.3.2 Display On/Off

All plane display can be turned on and off using the DEN bit in DSYSRn. When the DEN bit is 0, the display data set in DOORn is displayed.

When the value of the DPRS bit in DORCRm is 0, DPPRm is used to turn the display of planes 1 and 3 on and off. When the value of the DPRS bit in DORCRm is 1, on the other hand, DSPRn are used to turn the display of planes 1 and 3 on and off. Under the following display conditions, display data set in BPORn is displayed. RZ/G2M V1.3, RZ/G2M V3.0, plane 3 in the DU2 cannot be used. For RZ/G2H, RZ/G2N, plane 1 in the DU3 cannot be used.

1. When display of all planes 1 and 3 is turned off
2. In an area with no plane for display, due to the display size and display position
3. When the pixels in a plane for display are all a transparent color

Table 36.25 Turning On and Off the Display of Planes 1 and 3 (when the DPRS bit in DORCRm is 0)

| Display Plane | Display Plane Priority Register m (DPPRm) |
|---------------|---|
| Plane 1 | Plane 1 is selected in one among priority positions 1 and 2, and the corresponding enable bit is set to 1 |
| Plane 3 | Plane 3 is selected in one among priority positions 1 and 2, and the corresponding enable bit is set to 1 |

Table 36.26 Turning On and Off the Display of Planes 1 and 3 (when the DPRS bit in DORCRm is 1)

| Display Plane | Display Superimpose Priority Register n (DSPRn) |
|---------------|---|
| Plane 1 | Plane 1 is selected in one among priority positions 1 and 2 (setting to select the plane: B'0001) |
| Plane 3 | Plane 3 is selected in one among priority positions 1 and 2 (setting to select the plane: B'0011) |

Notes: Even if display on is set using DPPRm and DSPRn under the following conditions, the setting is handled as display off, and the corresponding plane is not displayed.

- Planes for which the value set in PpDPXRm is greater than the screen size (horizontal display end (HDE) - horizontal display start (HDS))
- Planes for which the value set in PpDPYRm is greater than the screen size (vertical display end (VDE) - vertical display start (VDS))
- Planes for which the value set in PpDSXRm is 0
- Planes for which the value set in PpDSYRm is 0

36.3.3 Plane Parameter

For each plane, a display area start position, memory width, display start position, and display size are set using registers.

The followings are the schematic diagram of start positions and sizes related to planes and the registers used for setting start positions and sizes.

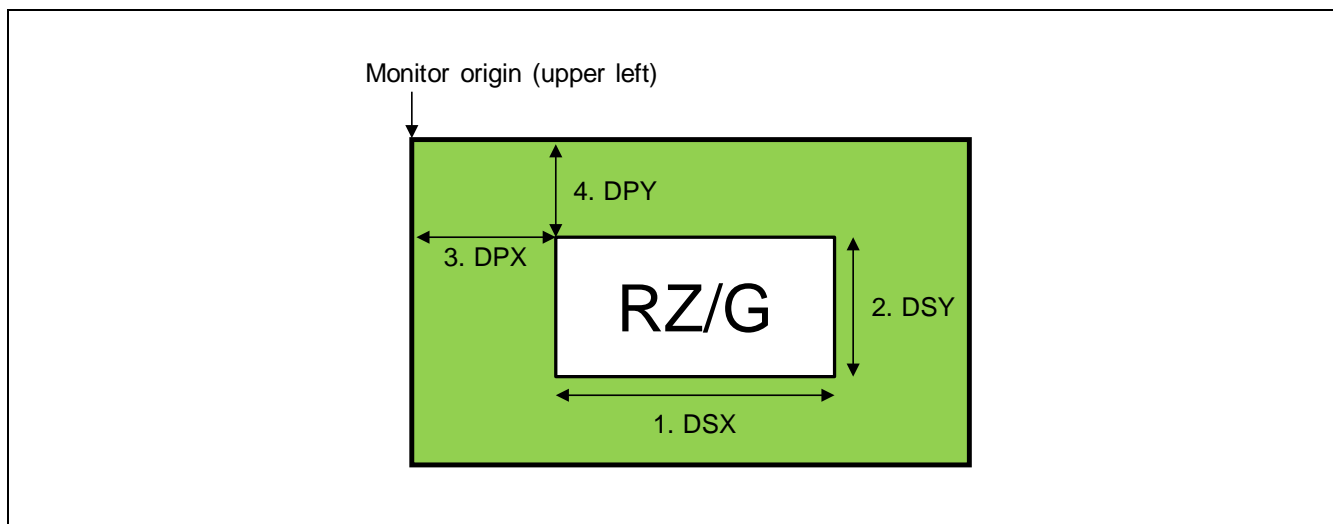


Figure 36.4 Parameters

Table 36.27 Monitor Parameter Setting Registers

| No. | Name Used in the Figure | Register | Description |
|-----|-----------------------------|----------|--|
| 1 | DSX (Display size X) | PpDSXRm | The X-direction display size of plane is set in dot-clock units. |
| 2 | DSY (Display size Y) | PpDSYRm | The Y-direction display size of plane is set in raster line units. |
| 3 | DPX (Display position X) | PpDPXRm | The X-direction distance to the display position is set in dot-clock units, taking the upper-left corner of the monitor as the origin. |
| 4 | DPY (Display position Y) | PpDPYRm | The Y-direction distance to the display position is set in raster line units, taking the upper-left corner of the monitor as the origin. |

36.3.4 Data Format for Input

The following format is used for color data used in display. A data configuration is shown in which data is allocated to the unified memory in little endian.

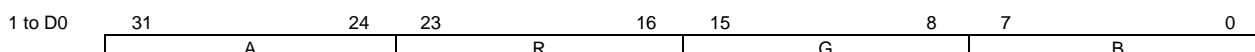
(1) 32-bit/pixel: ARGB8888

Levels for A, R, G, and B are represented by 8 bits each (ARGB8888). Note that the format includes a setting for alpha ratio in addition to the R, G, and B values. In this section, "32-bit/pixel" refers to the ARGB8888 format unless otherwise specified.

A 32-bit/pixel (ARGB8888) display should be specified with the PpEDF bits in PpDDC4Rm.

Only the value A is applied as the alpha value in ARGB8888.

32-bit/pixel data (ARGB8888 data) format

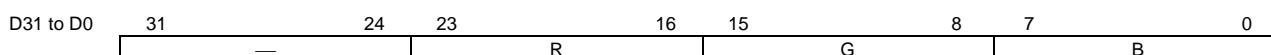


(2) 32-bit/pixel: RGB888

Levels for R, G, and B are represented by 8 bits each (RGB888). "—" Indicates any desired value.

A 32-bit/pixel (RGB888) display should be specified with the PpEDF bits in PpDDC4Rm.

32-bit/pixel data (RGB888 data) format

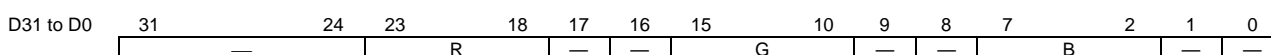


(3) 32-bit/pixel: RGB666

Levels for R, G, and B are represented by 6 bits each (RGB666). "—" Indicates any desired value.

A 32-bit/pixel (RGB666) display should be specified with the PpEDF bits in PpDDC4Rm.

32-bit/pixel data (RGB666 data) format

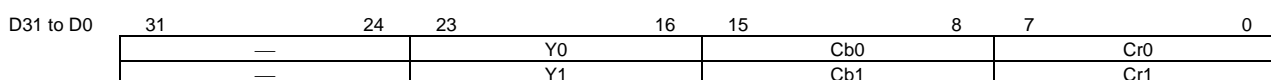


(4) YC: YCbCr444

Image data has the format YC (YCbCr) = 4:4:4. "—" Indicates any desired value.

A YC(YCbCr444) display should be specified with the PpEDF bits in PpDDC4Rm and value other than B'11 is specified in the PpDDF bit in PpMRm.

YC(YCbCr444) format



The formulae for YC-RGB conversion are given below. The underlined coefficients are defined by the settings in the corresponding registers.

$$R = YNC * (Y - YNO) + RCRC * (Cr - CRNO)$$

$$G = YNC * (Y - YNO) - GCRCR * (Cr - CRNO) - GCBC * (Cb - CBNO)$$

$$B = YNC * (Y - YNO) + BCBC * (Cb - CBNO)$$

The formulae for YC-RGB conversion in the default state are thus as follows.

$$R = Y + 1.37 * (Cr - 128)$$

$$G = Y - 0.698 * (Cr - 128) - 0.336 * (Cb - 128)$$

$$B = Y + 1.73 * (Cb - 128)$$

36.3.5 Data Formats for Output

In the case of digital RGB output from the DU, superimposing in the form of α blending and EOR operations is performed after the display data format has been expanded into the RGB888 format, and then the data are output. The supplementary formats in the case of expansion to RGB888 and output data formats are as indicated in the following table. For YCbCr422 output formats, refer to section 36.3.9 YC Format Display.

Table 36.28 Output Data Format

| | | Red | | | | | | | | Green | | | | | | | | Blue | | | | | | | |
|--|-------------------------|------------|---|---|---|---|---|---|---|-------------|---|---|---|---|---|---|---|-------------|---|---|---|---|---|---|---|
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Expanded data | 32-bit/pixel (ARGB8888) | R (8 bits) | | | | | | | | G (8 bits) | | | | | | | | B (8 bits) | | | | | | | |
| | 32-bit/pixel (RGB888) | R (8 bits) | | | | | | | | G (8 bits) | | | | | | | | B (8 bits) | | | | | | | |
| | 32-bit/pixel (RGB666) | R (6 bits) | | | | | | 0 | 0 | G (6 bits) | | | | | | 0 | 0 | B (6 bits) | | | | | | 0 | 0 |
| After superimposing | | R (8 bits) | | | | | | | | G (8 bits) | | | | | | | | B (8 bits) | | | | | | | |
| Output Data Format | | R (8 bits) | | | | | | | | G (8 bits) | | | | | | | | B (8 bits) | | | | | | | |
| | | Y | | | | | | | | Cb | | | | | | | | Cr | | | | | | | |
| | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Expanded data | | Y (8 bits) | | | | | | | | Cb (8 bits) | | | | | | | | Cr (8 bits) | | | | | | | |
| After superimposing | | Y (8 bits) | | | | | | | | Cb (8 bits) | | | | | | | | Cr (8 bits) | | | | | | | |
| Output Data Format (YCbCr444) | | Y (8 bits) | | | | | | | | Cb (8 bits) | | | | | | | | Cr (8 bits) | | | | | | | |
| Output Data Format (After YC-RGB conversion) | | R (8 bits) | | | | | | | | G (8 bits) | | | | | | | | B (8 bits) | | | | | | | |

36.3.6 Superimposing of Planes

For each plane, two types of combined superimposing are possible: α blending and EOR operations. By setting the PpSPIM bits in PpMRm, the super positioned display type can be selected.

However, α blending and EOR operation cannot be performed simultaneously on the same plane.

The alpha ratio of the 32-bit/pixel data (ARGB8888) can only be the higher-order eight bits (A value).

Table 36.29 Superimposing

| PpSPIM | Superimposing |
|--------|--|
| B'000 | Setting prohibited. (Default) |
| B'001 | Setting prohibited. |
| B'010 | Setting prohibited. |
| B'011 | Setting prohibited |
| B'100 | The specified plane is displayed. |
| B'101 | Blending of the specified plane with the lower plane is performed. |
| B'110 | EOR operation of the specified plane and the lower plane is performed. |
| B'111 | Setting prohibited |

After the image data format has been expanded to RGB888, α blending or EOR operation is performed. The complementary format of each display data format is shown in Table 36.28. α blending and EOR operation are performed in the sequence of the lower plane to the upper plane. The block diagram is shown below.

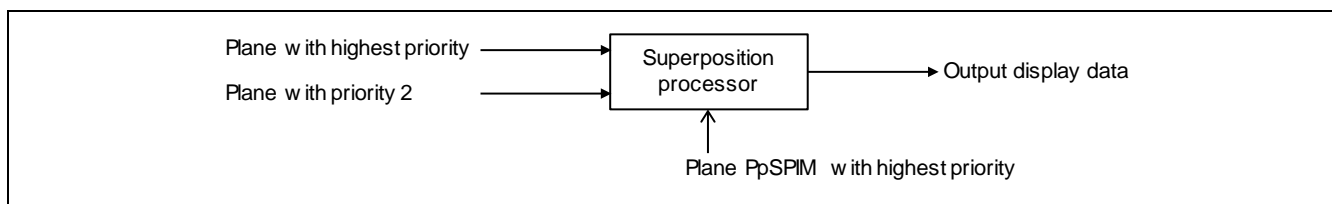


Figure 36.5 Plane Processing Sequence in α Blending and EOR Operation

When display of all lower planes is turned off, the specified plane is displayed. That is, α blending or EOR operation of the specified plane with the image data specified in BPORn is not performed.

(1) α Blending

The alpha ratio can only be the higher-order eight bits (A value) of the 32-bit/pixel data (ARGB8888).

$$\text{Result of blending} \approx (\text{specified plane} \times \alpha/255 + \text{lower plane} \times (1 - \alpha/255)) \quad (\text{Approximation})$$

In the above formula, the blending result, α , the specified plane, and the lower plane are all given as 8-bit data.

(2) EOR Operation

EOR operation of the specified plane with the lower plane is performed.

36.3.7 Sync Mode

In order to facilitate synchronization with external equipment, in addition to master mode, a TV synchronization function is provided. Selection of master mode and TV sync mode is performed using bits 7 and 6 in DSYSRn. In master mode (internal sync mode), the position of the falling edge of the vertical sync signal (VSYNC) set by VSPRn is detected. In TV sync mode (external sync mode), the position of the falling edge of the EXVSYNC signal is detected. The results are then reflected in the bits 14 and 11 in DSSRn.

Master Mode (Internal Sync Mode): By setting the period and pulse width of the horizontal and vertical sync signals (HSYNC, VSYNC) in the display timing generation registers, the corresponding waveforms are output. Also, display data is output in sync with these signals.

In interlaced sync mode and interlaced sync & video mode, a signal is output to the ODDF pin indicating odd/even fields.

TV Sync Mode (External Sync Mode): In TV sync mode, display data is output in sync with a horizontal sync signal and vertical sync signal (EXHSYNC, EXVSYNC) input from a TV, video, or other external sync signal generation circuit. Display data is output with reference to the falling edge of the EXHSYNC signal and the rising edge of the EXVSYNC signal.

The horizontal sync signal, vertical sync signal, and clock signal from the external sync signal generation circuit are input to the EXHSYNC, EXVSYNC, and DCLKIN pins, respectively. CSYNC is at high level. In interlaced sync mode and interlaced sync & video mode, a signal should be input to the EXODDF pin indicating odd/even fields. In non-interlaced mode, the input to the EXODDF pin should be fixed at low level or at high level.

When operating the DU in TV sync mode also, values must be set in HSWRn, HCRn, VSPRn, and VCRn in

Display Timing Generation Registers

When the EXVSYNC signal is input, either before or after completion of display of the display size portion set in the DU, the DU performs vertical display completion operation and transitions to control for the next screen. When the EXVSYNC signal is not input, the unit continues to wait for the EXVSYNC signal while remaining in the vertical blanking interval (auto-control is not performed). Similarly, when the EXHSYNC signal is input the DU performs horizontal display completion operation and transitions to control for the next raster line; but if the EXHSYNC signal is not input, the unit continues to wait for the EXHSYNC signal while remaining in the horizontal blanking interval (auto-control is not performed).

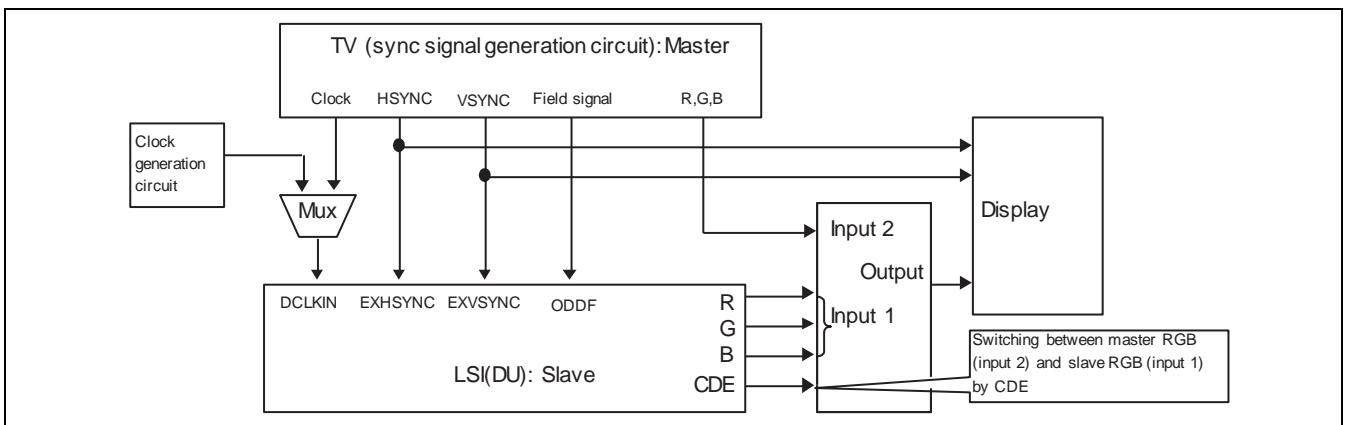


Figure 36.6 Signal Flow in TV Sync Mode [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Sync Method Switching Mode: When switching from master mode into TV sync mode, or from TV sync mode into master mode, when necessary this mode should be switched into first. Even if a transition to this mode is not made first, switching of the synchronization method is possible.

In this mode, input/output pins connected to the DU are for input, and so collision of pin signals can be avoided. Also, in this mode the internal dot clock is stopped, so that disorder in the input dot clock has no effect on display operation.

36.3.8 Dual Display Output

The DU has superimposition processors, display-timing generators, pin controllers, and dot-clock generators, each of which is independently controllable. The DU0 and DU1 realize various types of display because two channels are available for selecting from among the two display planes.

(1) Independent Display

Different images can be displayed with different sizes by specifying different superimposition processors, display-timing generators, pin controllers, and dot-clock generators.

However, separate output images cannot be produced from a single plane. To display a single set of image data in separate output images, two planes must be used.

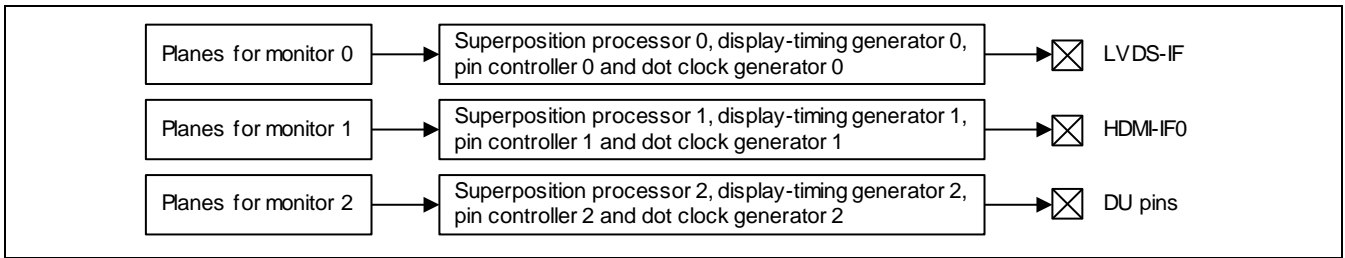


Figure 36.7 Independent Display [RZ/G2M V1.3, RZ/G2M V3.0]

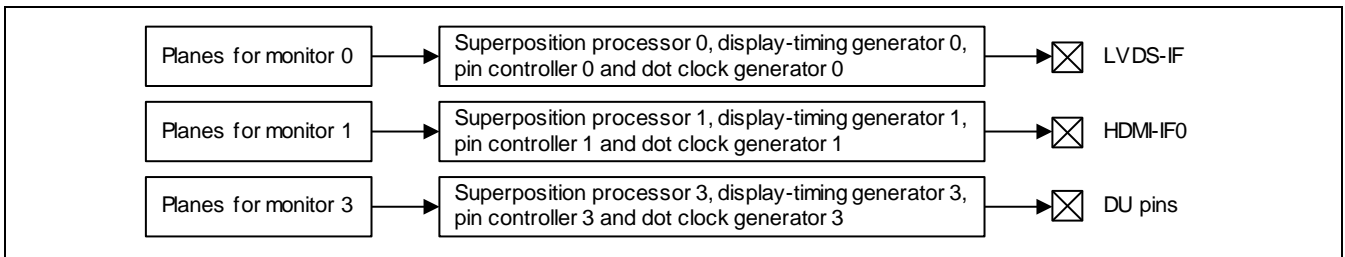


Figure 36.8 Independent Display [RZ/G2H, RZ/G2N]

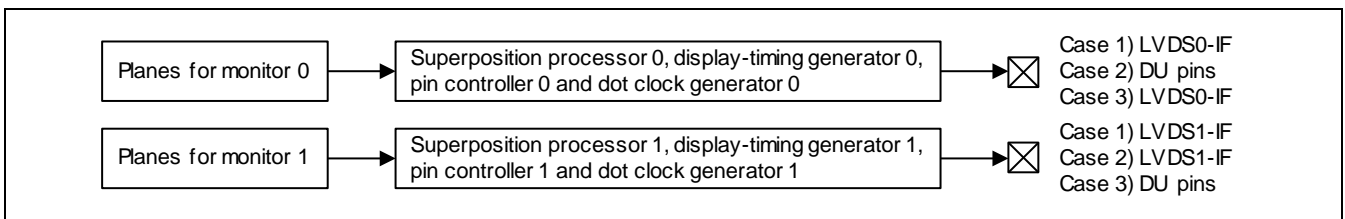


Figure 36.9 Independent Display [RZ/G2E]

(2) Display of Different Images on Monitors of the Same Size

Different images can be displayed on monitors of the same size when the same display-timing generator and dot-clock generator have been specified but the superimposition processors and pin controllers are different.

Images can be output to two monitors through a single set of pins by switching the display data in synchronization with the output dot clock.

When images are output to two monitors through a single set of pins, the image data can be output only via the LVDS-IF.

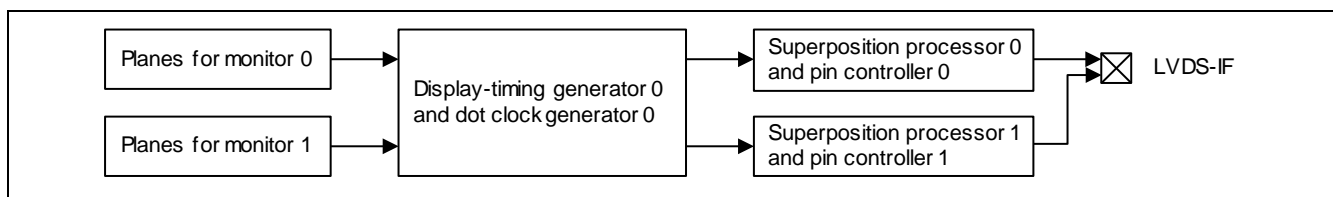


Figure 36.10 Display of Different Images on Monitors of the Same Size

The timing for switching of display data in synchronization with the output dot clock is shown below. This feature is only supported by the LVDS-IF. For switching display data in synchronization with the output dot clock, the output dot clock should be set to be division by two or more of the input dot clock.

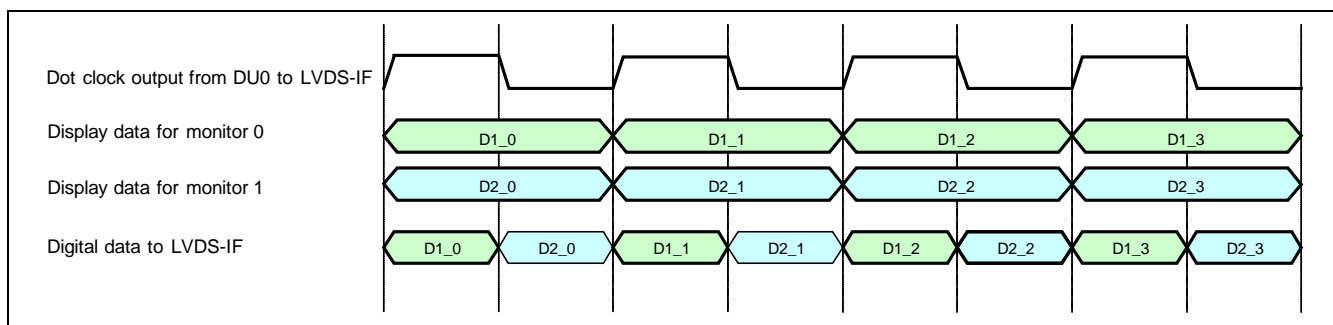


Figure 36.11 Switching Display Data in Synchronization with the Output Dot Clock

(3) Combinations of Blocks for Dual Display Output

For dual display output, the dot clock to be supplied to blocks in the DU, timing signals including SYNC, and display data are specifiable by setting the corresponding registers. Table 36.30 is a matrix of the combinations of blocks, and Table 36.31 gives the meanings of the symbols used in Table 36.30.

When the input block is superimposition processor 1, for example, the information indicated in the table is as follows.

- The dot clock is that from dot-clock generator 0 or 1 as selected by the DR0D and DK1S bits in DORCR0 (No. 5 in the table).
- The timing signal is that from display-timing generator 0 or 1 as selected by the DR0D and PG1T bits in DORCR0 (No. 3 in the table).
- The planes for superimposing of data are selected in DSPR0 (No. 12 in the table).

Take care with register settings because timing signals such as the dot clock, sync signals and so on are freely specifiable, but if the dot clock or other timing signals differ from block to block, a normal display will not be possible.

Table 36.30 Combinations of Blocks

| | Output Block | | | | | | | | | | | | |
|---|-------------------------------|---------------------------------------|-------------------------------|---------------------------------------|----------------------------|----------------------------|-----------------------------|-----------------------------|---|---|----------------|----------------|----------------|
| | Dot-clock generator 0 (clock) | Dot-clock generator 0 (external SYNC) | Dot-clock generator 1 (clock) | Dot-clock generator 1 (external SYNC) | Display-timing generator 0 | Display-timing generator 1 | Superimposition processor 0 | Superimposition processor 1 | Pin controller (output timing adjustment) 0 | Pin controller (output timing adjustment) 1 | Planes 1 and 3 | DU0 input pins | DU1 input pins |
| Dot-clock generator 0 (clock) | — | — | — | — | — | — | — | — | — | — | — | ✓ | — |
| Dot-clock generator 0 (external SYNC) | — | — | — | — | — | — | — | — | — | — | — | ✓ | — |
| Dot-clock generator 1 (clock) | — | — | — | — | — | — | — | — | — | — | — | — | ✓ |
| Dot-clock generator 1 (external SYNC) | — | — | — | — | — | — | — | — | — | — | — | — | ✓ |
| Display-timing generator 0 | ✓ | ✓ | — | — | — | — | — | — | — | — | — | — | — |
| Display-timing generator 1 | 1 | — | 1 | ✓ | — | — | — | — | — | — | — | — | — |
| Superimposition processor 0 | 4 | — | 4 | — | 2 | 2 | — | — | — | — | 11 | — | — |
| Superimposition processor 1 | 5 | — | 5 | — | 3 | 3 | — | — | — | — | 12 | — | — |
| Pin controller (output timing adjustment) 0 | ✓ | — | — | — | ✓ | — | 6 | 6 | — | — | — | — | — |
| Pin controller (output timing adjustment) 1 | 1 | — | 1 | — | 8 | 8 | 7 | 7 | — | — | — | — | — |
| Planes 1 and 3 | 9 | — | 9 | — | 10 | 10 | — | — | — | — | — | — | — |
| LVDS-IF | — | — | — | — | — | — | — | — | ✓ | — | — | — | — |
| HDMI-IF | — | — | — | — | — | — | — | — | — | ✓ | — | — | — |

Table 36.31 Meanings of Symbols and Numbers

| Symbol or No. | Register Name (Abbr.) | Bit Name | Description |
|---------------|-----------------------|---------------|--|
| ✓ | — | — | One-to-one combination |
| — | — | — | No combination |
| 1 | DORCR0 | DK1S | 0: Dot-clock generator 0 (clock) 1: Dot-clock generator 1 (clock) |
| 2 | DORCR0 | PG0D PG1T | Other than 01_1: Display-timing generator 0 01_1: Display-timing generator 1 |
| 3 | DORCR0 | PG0D PG1T | Other than 00_1: Display-timing generator 0 00_1: Display-timing generator 1 |
| 4 | DORCR0 | PG0D DK1S | Other than 01_1: Dot-clock generator 0 (clock) 01_1: Dot-clock generator 1 (clock) |
| 5 | DORCR0 | PG0D DK1S | Other than 00_1: Dot-clock generator 0 (clock) 00_1: Dot-clock generator 1 (clock) |
| 6 | DORCR0 | PG0D | 00: Superimposition processor 0 01: Superimposition processor 1 10: Fixed to 0 11: The value of DOOR0 |
| 7 | DORCR0 | PG1D | 00: Superimposition processor 0 01: Superimposition processor 1 10: Fixed to 0 11: The value of DOOR1 |
| 8 | DORCR0 | PG1T | 0: Display-timing generator 0 1: Display-timing generator 1 |
| 9 | DPTSR0 | P1DK and P3DK | 0: Dot-clock generator 0 (clock) 1: Dot-clock generator 1 (clock) |
| 10 | DPTSR0 | P1TS and P3TS | 0: Display-timing generator 0 1: Display-timing generator 1 |
| 11 | DSPR0 | — | See the description of DSPR0 |
| 12 | DSPR1 | — | See the description of DSPR1. |

36.3.9 YC Format Display

In addition to RGB format display, YC format can be displayed. YCbCr444 data from VSP convert into YCbCr422 format and display as YC format. For RZ/G2H, RZ/G2N, YC(YCbCr422) format can be displayed from the DU3 to external pins. YC format display cannot be superimposed. For RZ/G2M V1.3, RZ/G2M V3.0, YC (YCbCr422) format can be displayed from the DU2 to external pins. For RZ/G2E, YC (YCbCr422) format can be displayed from the DU0 or DU1 to external pins.

RGB format display and YC format display is selected by setting bits 9 and 8 in DEFRn*. Bits 9 and 8 setting of B'10 selects 16-bit YC format display (non-multiplexed), and a setting of B'11 selects 8-bit YC format display (multiplexed). A setting of B'00 selects RGB format.

When YC format is displayed by timing as shown in Figure 36.12 and Figure 36.15, input a 2× clock to DCLKIN and specify division by 2. Also output a 2× clock from DCLKOUT. Both 2× clock output and division by 2 are specified by settings in ESCRn*. Non-multiplexed YC format can be displayed in Figure 36.13 and Figure 36.14, this setting is not required.

Note: * n = 3 [RZ/G2H, RZ/G2N], n = 2 [RZ/G2M V1.3, RZ/G2M V3.0], n = 0, 1 [RZ/G2E]

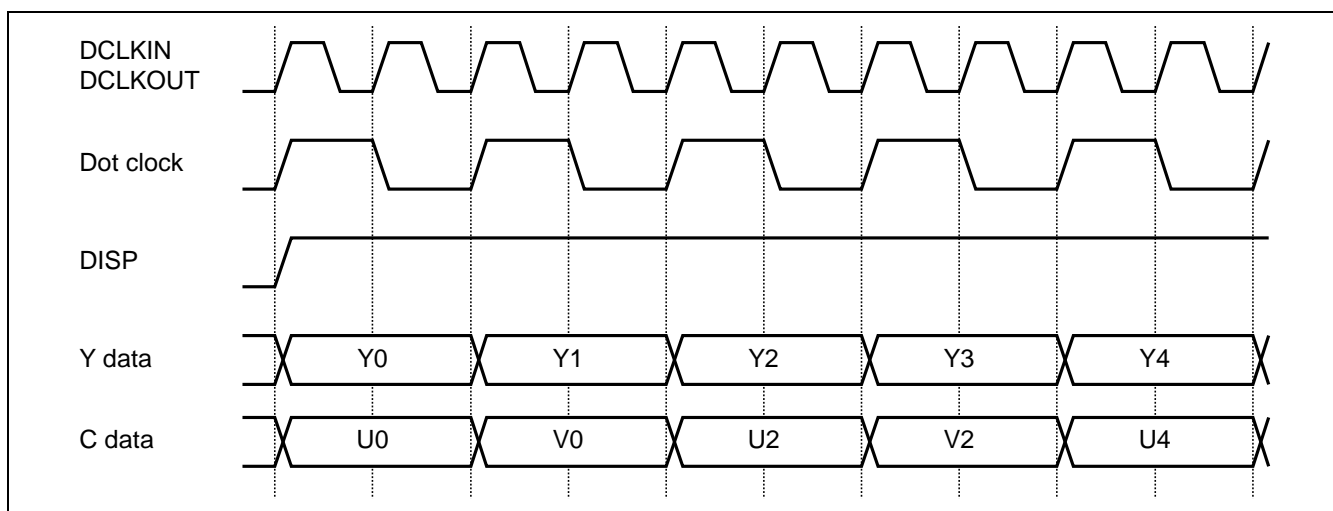


Figure 36.12 YC Non-Multiplexed

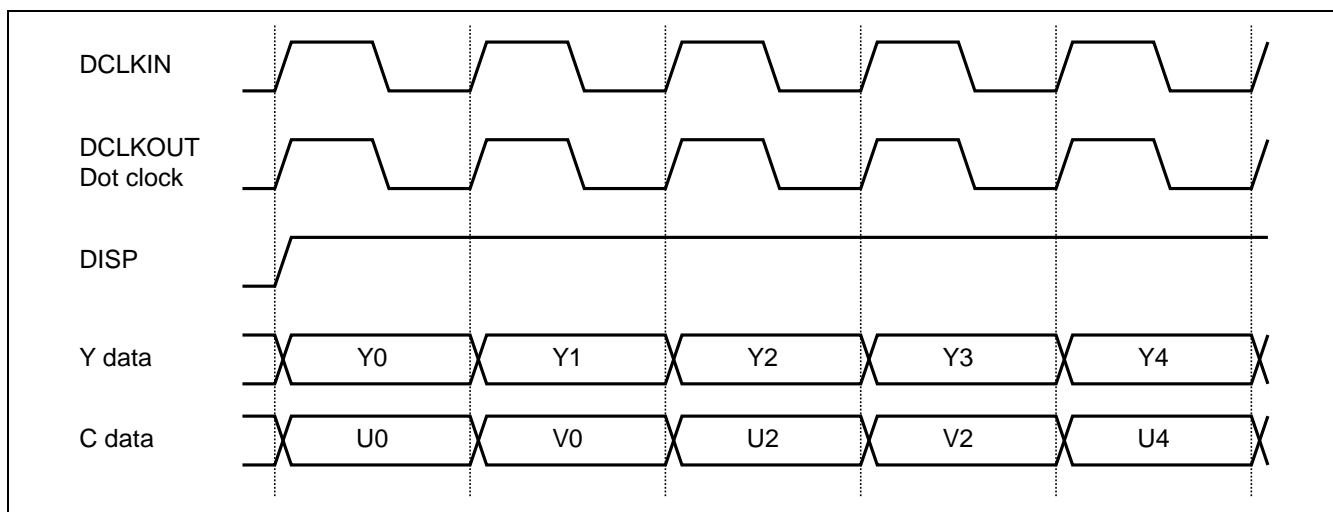


Figure 36.13 Non-Multiplexed (no division)

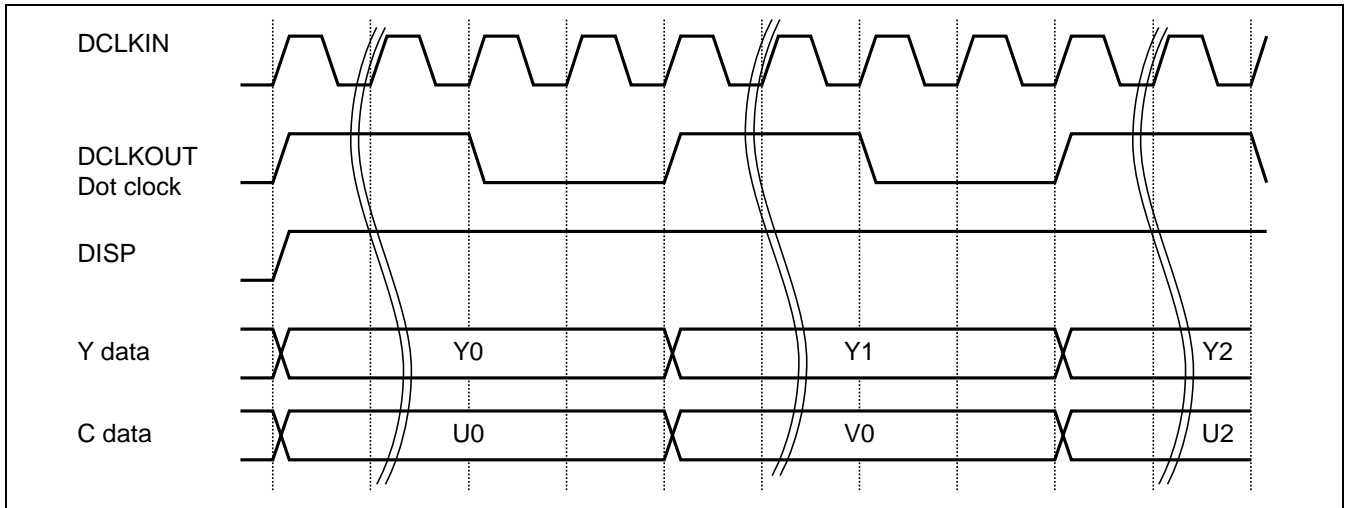


Figure 36.14 Non-Multiplexed (division by x)

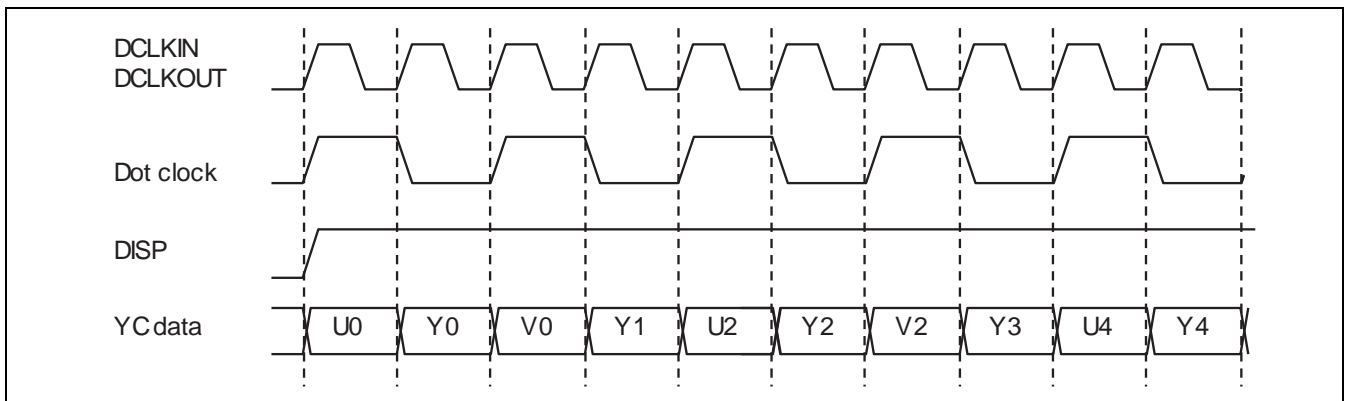


Figure 36.15 YC Multiplexed

Table 36.32 Pin Names and YC Data for YC Format Display

| Pin Name | YC Format Display | |
|----------|-------------------|-------------|
| | Non-Multiplexed | Multiplexed |
| DU_DR2 | Y2 | Y2, C2 |
| DU_DR3 | Y3 | Y3, C3 |
| DU_DR4 | Y4 | Y4, C4 |
| DU_DR5 | Y5 | Y5, C5 |
| DU_DR6 | Y6 | Y6, C6 |
| DU_DR7 | Y7 | Y7, C7 |
| DU_DG2 | C6 | — |
| DU_DG3 | C7 | — |
| DU_DG4 | — | — |
| DU_DG5 | — | — |
| DU_DG6 | Y0 | Y0, C0 |
| DU_DG7 | Y1 | Y1, C1 |
| DU_DB2 | C0 | — |
| DU_DB3 | C1 | — |
| DU_DB4 | C2 | — |
| DU_DB5 | C3 | — |
| DU_DB6 | C4 | — |
| DU_DB7 | C5 | — |

36.3.10 Display Timing Generation

In the DU, display timing is generated for the horizontal direction and vertical direction of the display screen. Display timing is set by using display timing generation registers in section 36.2.2. Figure 36.16 shows the display timing in non-interlaced mode. Here, the display screen is defined in terms of the variables of Table 36.33.

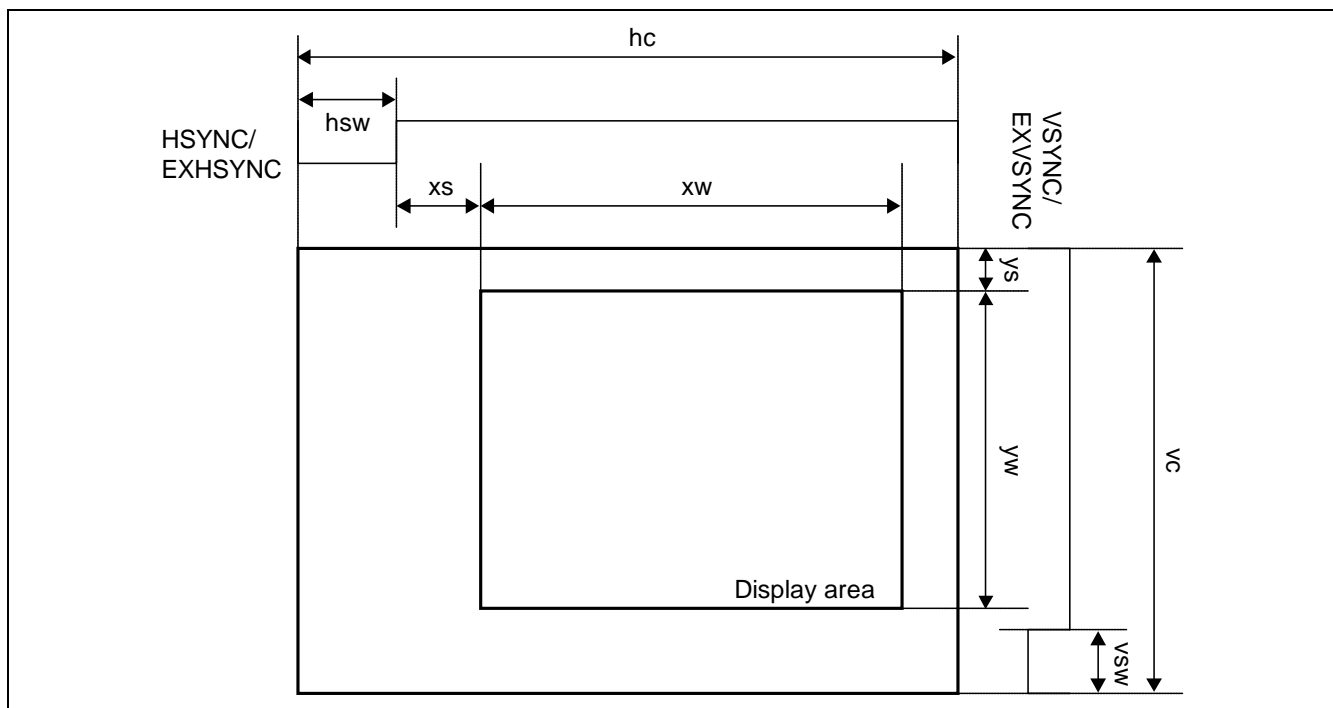


Figure 36.16 Display Timing Generation for Horizontal Direction and Vertical Direction of Display Screen

Table 36.33 Variables Defined in Display Screen

| Variables | Contents | Units |
|-----------|--|-------------|
| hc*1 | Horizontal scan period | Dot clock |
| hsw | Horizontal sync pulse width | Dot clock |
| xs | From rise of HSYNC to display start position in the horizontal direction of the display screen | Dot clock |
| xw | Display width per 1 raster of display screen | Dot clock |
| vc*2 | Vertical scan period | Raster line |
| vsw | Vertical sync pulse width | Raster line |
| ys | From rise of VSYNC to display start position in the vertical direction of the display screen | Raster line |
| yw | Vertical display period of display screen | Raster line |

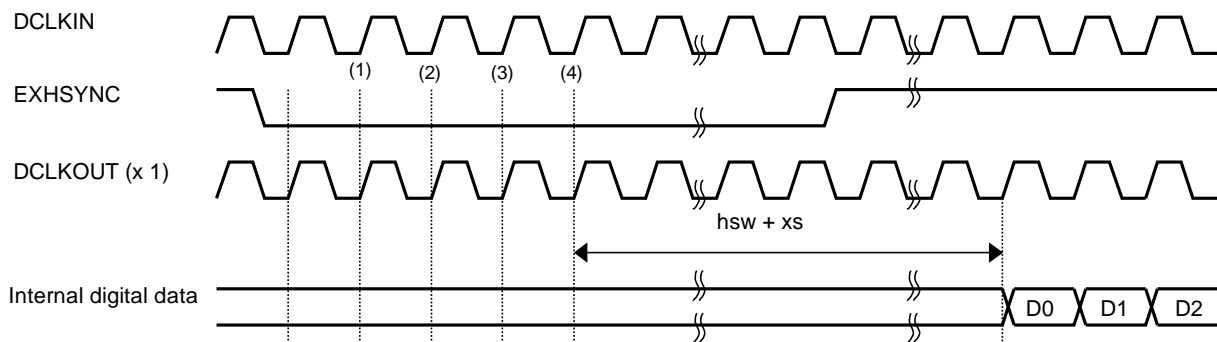
Notes: 1. Should be set such that $hsw + xs + xw < hc$
 2. Should be set such that $vsw + ys + yw < vc$

The display timing generation register settings are different depending on the scan method and synchronization method. Hence the value of the display timing generation registers should be set after performing calculations like those indicated in Table 36.34.

Table 36.34 Correspondence Table of Settings of Display Timing Generation Registers

| Register Name | Bit Name | Synchronization Method | |
|---|-------------------|------------------------|--------------------------|
| | | Master Mode | TV Sync Mode |
| Horizontal display start register n (HDSRn) | HDS* ⁶ | $hsw + xs - 19*5$ | $hsw + xs - 25*2*5$ |
| Horizontal display end register n (HDERn) | HDE | $hsw + xs - 19*5 + xw$ | $hsw + xs - 25 + xw*2*5$ |
| Vertical display start register n (VDSRn) | VDS | $ys - 2*3$ | $ys - 2*3$ |
| Vertical display end register n (VDERn) | VDE | $ys - 2 + yw$ | $ys - 2 + yw$ |
| Horizontal synch width register n (HSWRn) | HSW | $hsw - 1$ | $hsw - 1$ |
| Horizontal cycle register n (HCRn) | HC | $hc - 1$ | $hc - 1$ |
| Vertical synch point register n (VSPRn) | VSP | $vc - vsw - 1$ | $vc - vsw - 1$ |
| Vertical cycle register n (VCRn) | VC | $vc - 1$ | $vc - 1$ |

Notes: 1. In all scan modes, VDS, VDE, VSP, VC settings are in single-field units.
 2. The values of HDS and HDE are from the fourth rising edge of DCLKOUT after detection of the falling edge of EXHSYNC through the rising edge of DCLKOUT



- 3. VDS should be set to 1 or greater.
 - 4. HC should be set so as to satisfy $HC > HDE$.
 - 5. If the function below is used, the following correction value is subtracted from both HDS and HDE in Table 36.34.
 - 6. HDS should be set to 1 or greater.
- When using the YC-RGB conversion function (YCRGB0 or YCRGB1 bits in DEF5Rm are set to 1), 3 should be subtracted.

36.3.11 CSYNC [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

In master mode, a CSYNC (composite sync) signal is output. EQWRn is used to set the low-level pulse width of the CSYNC equivalent pulse. SPWRn is used to set the low-level pulse width of the CSYNC serration pulse.

The CSYNC waveform is selected using the CSY bit in DSMRn.

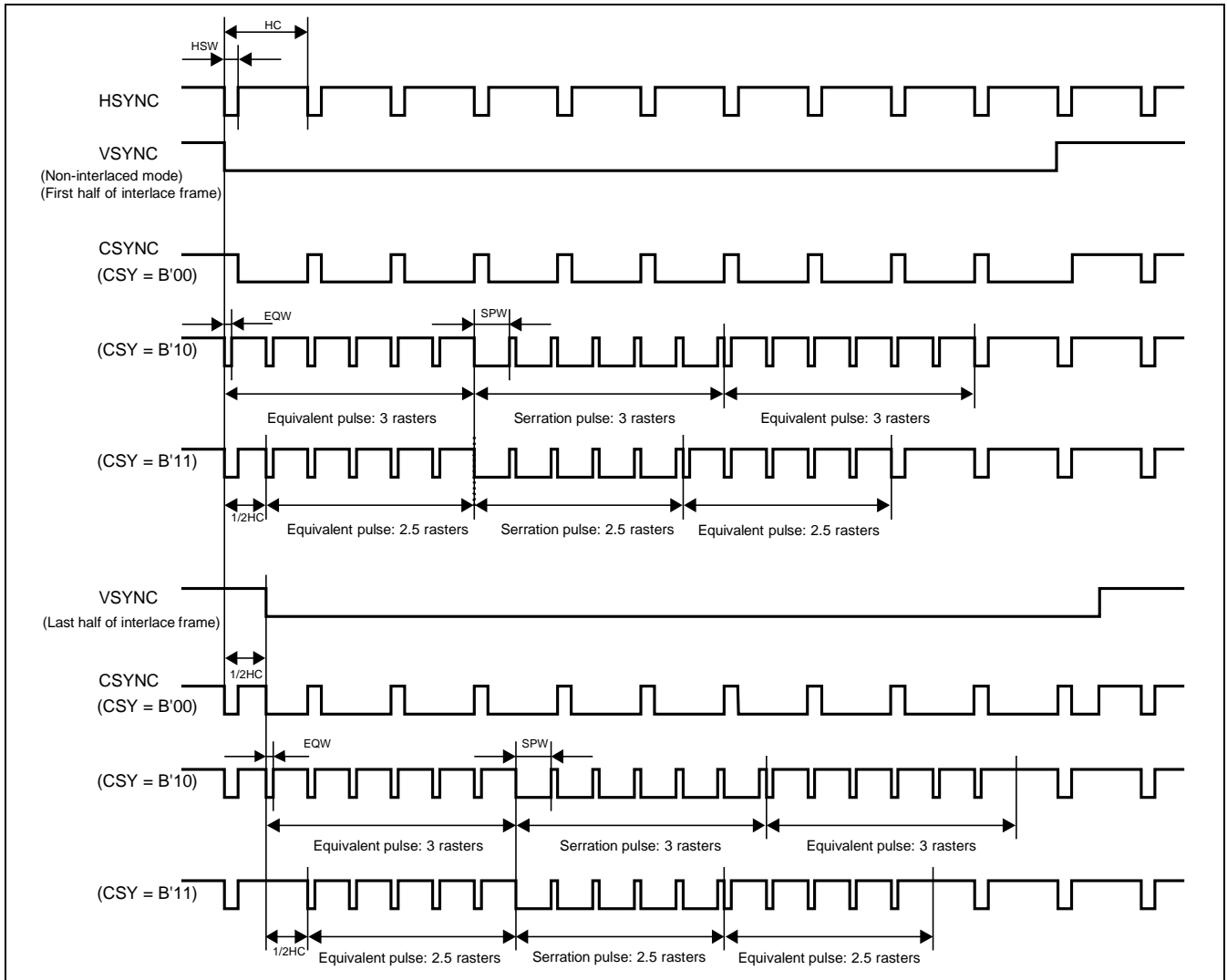


Figure 36.17 CSYNC Timing Chart

36.3.12 Scan Method

The scan method can be selected from among non-interlaced mode, interlaced sync mode, and interlaced sync & video mode. The mode is selected using bit 5 and bit 4 in DSYSRn.

RZ/G2E support only non-interlaced mode, and does not support Interlaced sync mode and Interlaced sync & video mode.

- Non-interlaced mode

In this scan method, one frame consists of a single field.

- Interlaced sync mode

In this scan method, one frame consists of two fields. The two fields are an even field and an odd field, displaying the same data.

- Interlaced sync & video mode

In this scan method, one frame consists of two fields. The two fields are an even field and an odd field, displaying different data.

The ODEV bit in DSMRn is used to set the display order of fields in interlaced sync mode and in interlaced sync & video mode. When the ODEV bit is 0, the display order for one frame is odd field, then even field; when the ODEV bit is 1, the order for one frame is even field, then odd field.

In master mode, high level is output from the ODDF pin during even field display, and low level is output during odd field display. In TV sync mode, high level is input to the EXODDF pin to cause display of the even field, and low level is input to cause display of the odd field.

Note: When non-interlaced mode is selected in TV sync mode, the EXODDF pin should be fixed at low level or high level.

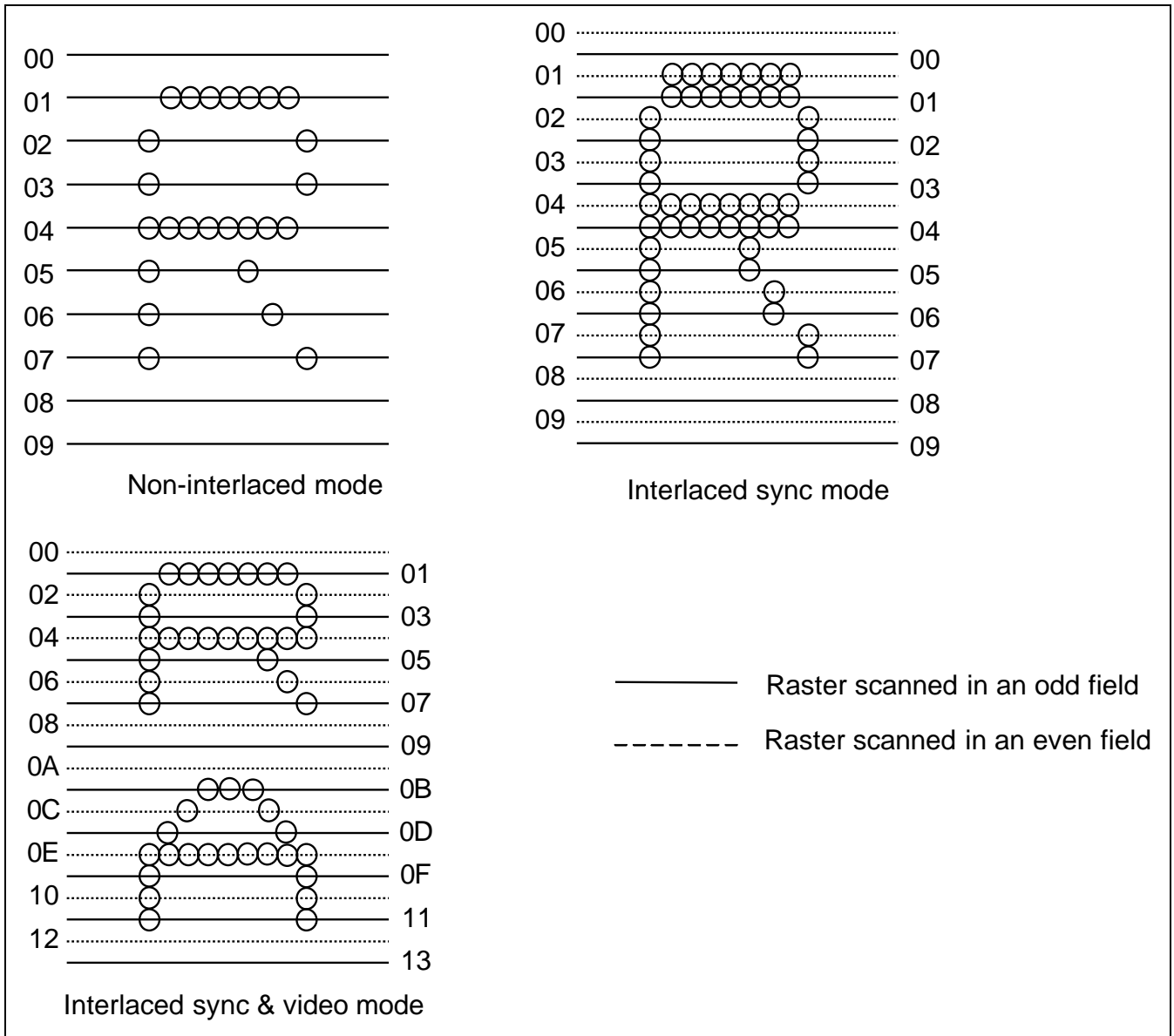


Figure 36.18 Example of Display in Each Scan Mode

- Example of vertical scan period
 Non-interlaced mode: 1/60 second/field, 1/30 second/field
 Interlaced sync mode: 1/30 second/frame
 Interlaced sync & video mode: 1/30 second/frame
- Display in non-interlaced method
 In this method, all lines are displayed at once without providing intervals between input video signals.
 This input method is for monitors capable of high-resolution display.

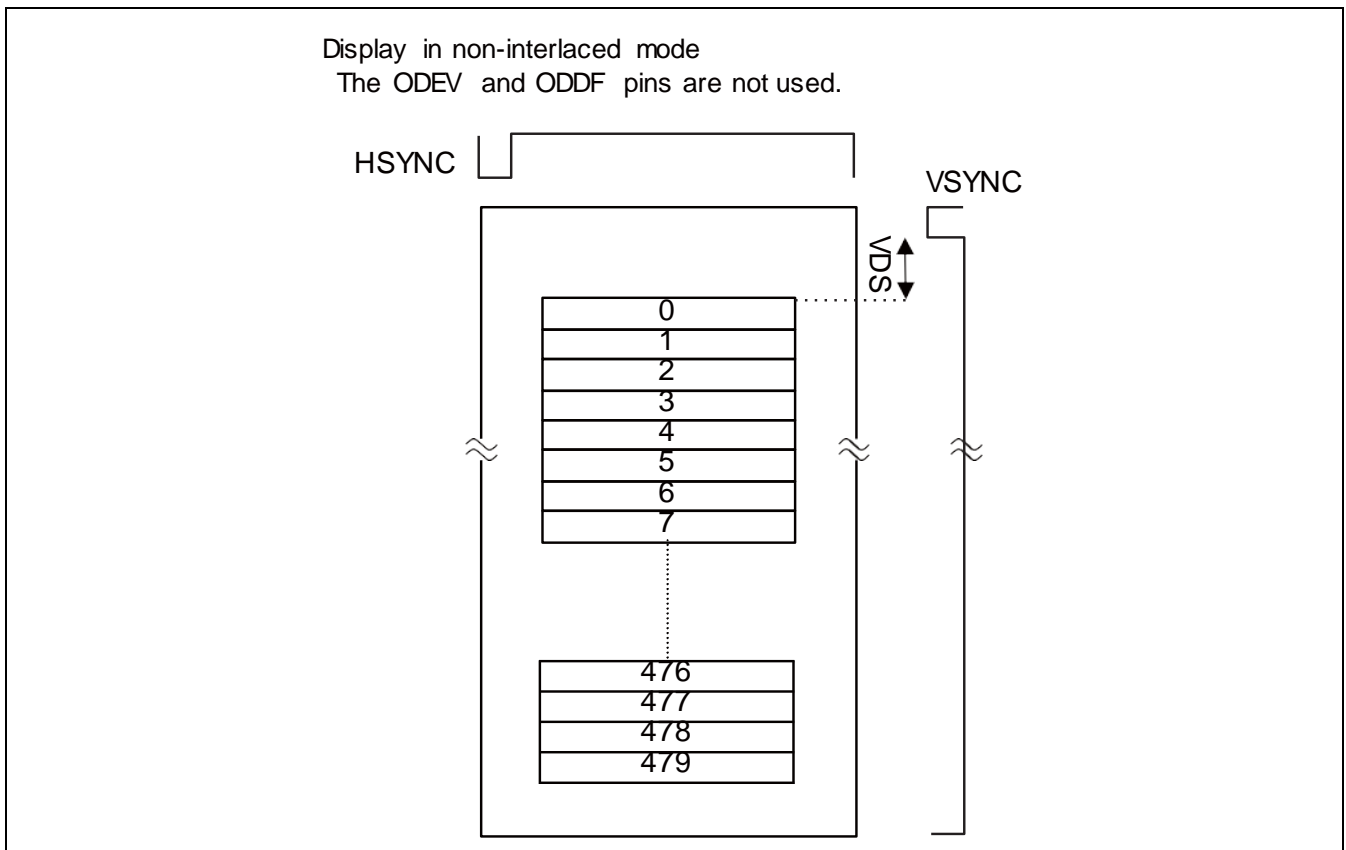


Figure 36.19 Display in Non-Interlaced Method

• Display in interlaced method

At every scan period VC of the input video signal, even lines and odd lines are switched and displayed in alternation, and a single screen (one frame) is combined and displayed (with the afterimage of the preceding VC) with a period of 2VC. This is a basic TV input method.

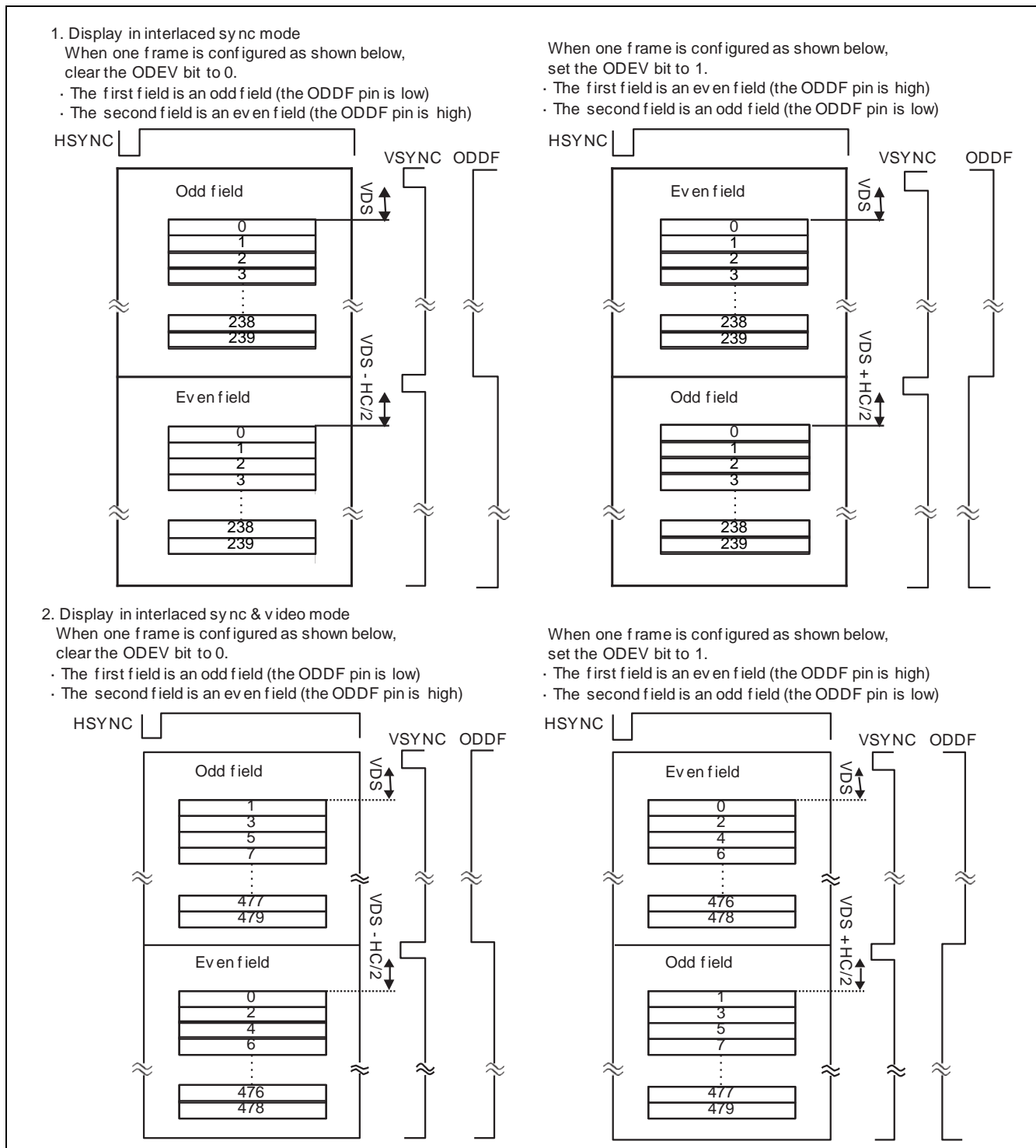


Figure 36.20 Display in Interlaced Method

36.3.13 Color Detection

When output display data matches a color set in CDERn, high level is output from the CDE pin. The CDEM bit in DSMRn can be used to fix the level outside display intervals. Also, the CDEL bit in DSMRn can be used to select the polarity of the output level. See Table 36.36 for the bits to be compared.

When the following display functions are set, the color detection function cannot be used.

- YC-RGB conversion performed by setting the YCRGB0 or YCRGB1 bits in DEF5Rm

Table 36.35 Output Level of the CDE Pin

| CDEL | CDEM | The CDE pin in display intervals | | The CDE pin outside display intervals | |
|------|------|---|------------|---------------------------------------|--------------|
| | | Result of Comparison of Output Display Data and CDERn | | Value of CDERn* | |
| | | Same | Different | 0 | Other than 0 |
| 0 | B'00 | High level | Low level | High level | Low level |
| 0 | B'01 | High level | Low level | High level | Low level |
| 0 | B'10 | High level | Low level | Low level | Low level |
| 0 | B'11 | High level | Low level | High level | High level |
| 1 | B'00 | Low level | High level | Low level | High level |
| 1 | B'01 | Low level | High level | Low level | High level |
| 1 | B'10 | Low level | High level | High level | High level |
| 1 | B'11 | Low level | High level | Low level | Low level |

Note: * Output display data is 0 outside display intervals.

Table 36.36 Data Comparison by CDERn

| Bits | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
|---------------------|-----|----|----|----|----|----|----|----|----|----|-----|----|----|----|----|----|----|----|-----|----|----|----|----|----|---|---|
| CDERn | CDR | | | | | | | | X | X | CDG | | | | | | X | X | CDB | | | | | | X | X |
| Output display data | R7 | R6 | R5 | R4 | R3 | R2 | R1 | R0 | G7 | G6 | G5 | G4 | G3 | G2 | G1 | G0 | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | | |

Note: "X" is not compared.

36.3.14 External Sync Control [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

In TV-sync mode, the DU is capable of using an externally input dot clock (clock signal for frequency multiplication: DCLKIN) to generate a dot clock (output dot clock: DCLKOUT) that is in accord with external synchronizing signals (EXHSYNC, EXVSYNC). Supply the externally input dot clock (multiplication clock: DCLKIN) and set the following parameters in ESCRn.

Table 36.37 External Sync Control Parameters

| Variable | Function |
|---------------|---|
| ESCRn/SYNCSEL | Selects the sync signal (EXHSYNC or EXVSYNC) to use in phase matching of the dot clock. |
| ESCRn/FRQSEL | Selects the dot-clock division ratio. |

1. Use the SYNCSEL bits in ESCRn to set the sync timing of the dot clock (output dot clock: DCLKOUT) generated from the internal dot clock.
2. Use the FRQSEL bits in ESCRn to set the division ratio for generation of the internal dot clock. The following figure shows the internal dot clock timing where the input dot clock has been synchronized with EXHSYNC and then divided by four.

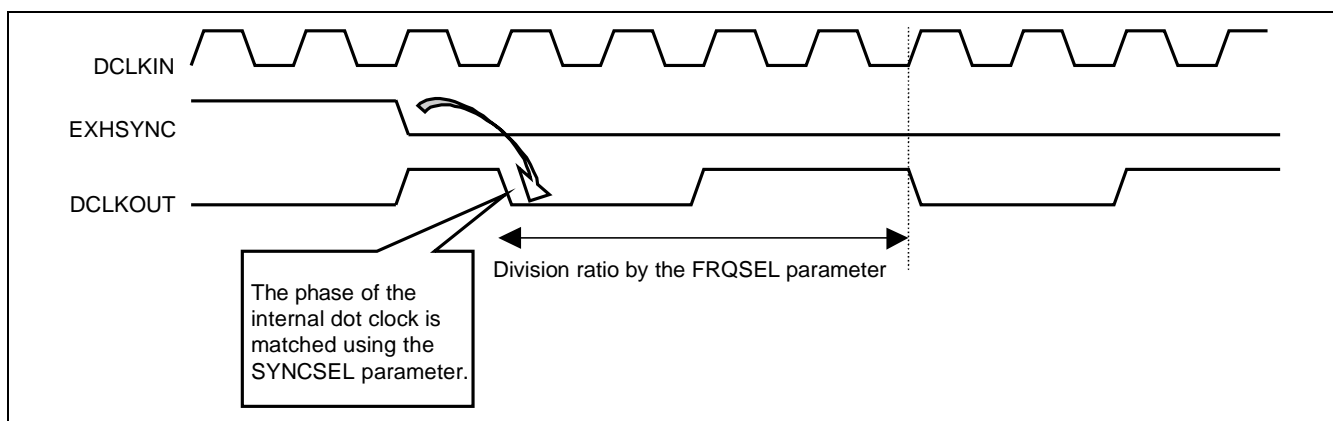


Figure 36.21 DCLKOUT Timing Chart where DCLKIN Synchronized with EXHSYNC is divided by four

36.3.15 Output Signal Timing Adjustment

The DU enables selection of output timing, with respect to the output dot clock, of the various output signals (the four sync signals HSYNC, VSYNC, CSYNC*¹ and ODDF*¹, as well as DISP, CDE, CLAMP, DE and digital RGB signals). Timing is selected by setting OTAR_n*².

- Notes: 1. RZ/G2E does not support CSYNC and ODDF signal.
 2. n = 3 [RZ/G2N]
 n = 2 [RZ/G2M V1.3, RZ/G2M V3.0]
 n = 0 and 1 [RZ/G2E]

Table 36.38 Output Signal Timing Setting Parameters

| Variable | Function |
|----------|--|
| SYNCA | Sets output timing of the HSYNC, VSYNC, CSYNC and ODDF signals. [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| | Sets output timing of the HSYNC and VSYNC signals. [RZ/G2E] |
| DISPA | Sets output timing of the DISP signal |
| CDEA | Sets output timing of the CDE signal |
| DRGBA | Sets output timing of digital RGB signal |
| CLAMPA | Sets output timing of the CLAMP signal |
| DEA | Sets output timing of the DE signal |

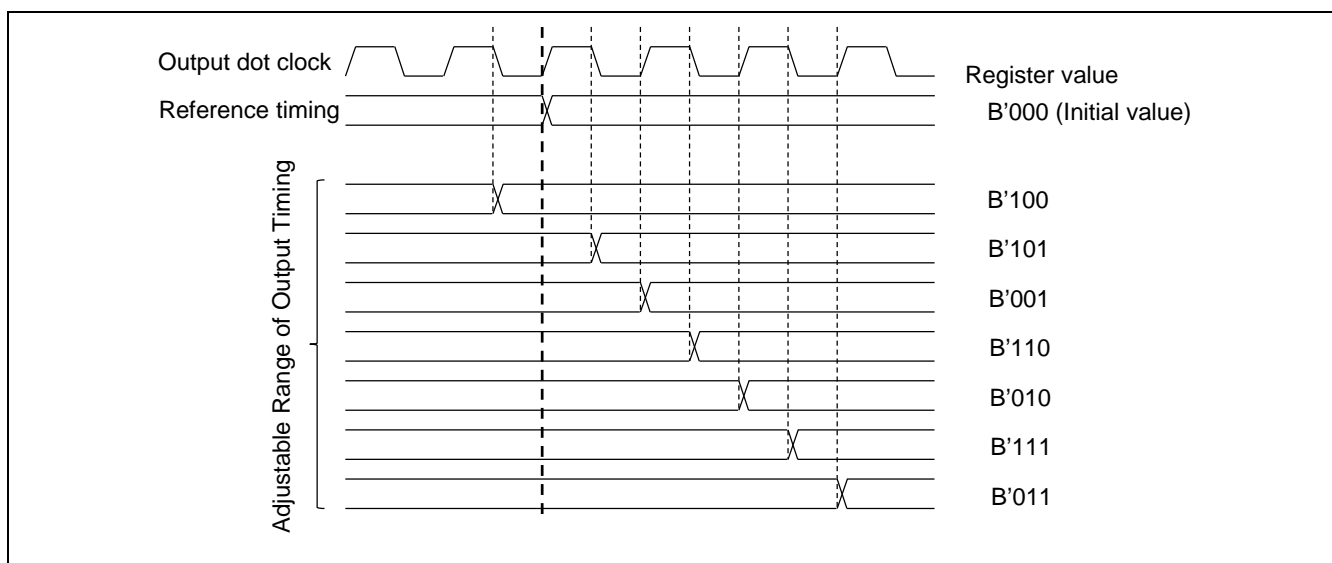


Figure 36.22 Adjustable Range of Output Timing

36.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

36.4.1 Module Standby Mode

Module Standby mode, in which supply of the clock signal to the DU is stopped, is supported.

Even when the DU enters the standby mode, the register values are retained. Do not access the DU during periods in standby mode. For RZ/G2N, when running the DU3, release module standby mode of the DU2 and DU3.

36.4.2 Transition to Module Standby Mode

- Set both the DEN and DRES bits in DSYSRn to 0 to stop access to the VSP.
- Test bit 11 in DSSRn to confirm the next VBK flag. The setting made in step 1 becomes effective with the timing of VBK and access to the VSP is stopped. The value of the display data becomes the value set in DOORn
 For RZ/G2M V1.3, RZ/G2M V3.0, when the DU0, DU1 and DU2 are running, test bit 11 of all of DSSR0, DSSR1, and DSSR2.
 For RZ/G2H, RZ/G2N, when the DU0, DU1 and DU3 are running, test bit 11 of all of DSSR0, DSSR1, and DSSR3.
 For RZ/G2E, when the DU0 and DU1 are running, test bit 11 of all of DSSR0 and DSSR1.
- Stop the clock.

36.4.3 Release from Module Standby Mode and Restarting Display

- Start the clock.
- Make settings to turn the display on by setting the DEN and DRES bits in DSYSRn to 1 and 0 respectively.

36.4.4 Acquisition of External Sync Signal [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

The following three ways of acquiring the external SYNC signal are available.

- The SYNC signal is acquired on rising edges of DCLKIN (or the frequency-divided clock signal derived from this if division has been set up).
- The SYNC signal is acquired on falling edges of DCLKIN (or the frequency-divided clock signal derived from this if division has been set up).
- Assuming that division has been set up, the SYNC signal that is acquired by the pre-division clock signal is latched on edges of the frequency-divided clock signal.

The electrical characteristics (AC spec.) are only guaranteed for case (a) above. There is no guarantee of electrical characteristics (AC spec.) for cases (b) and (c).

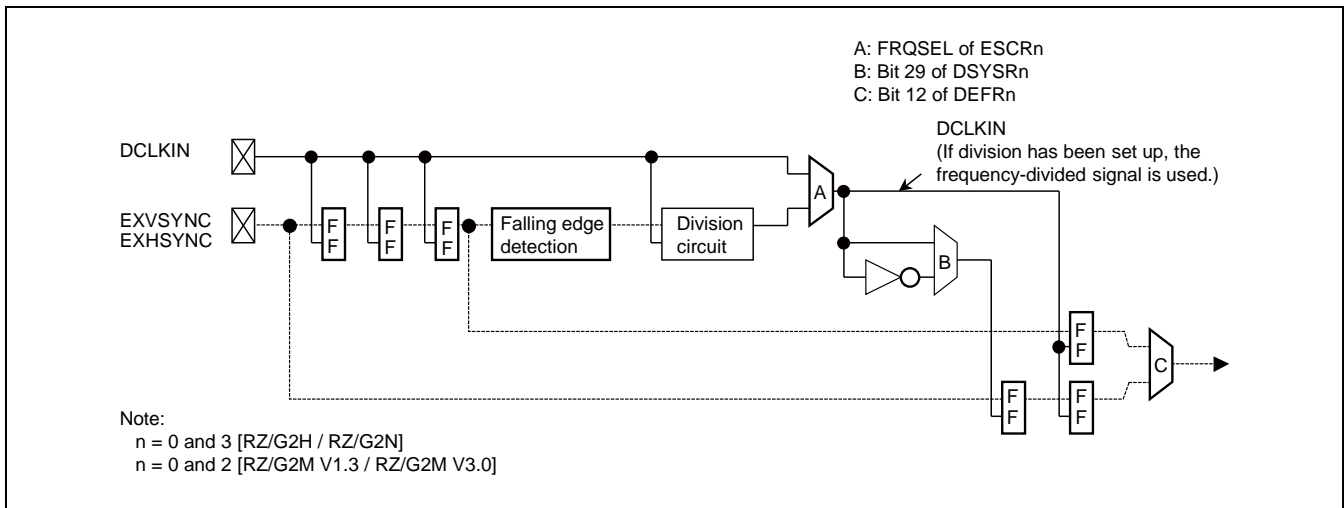


Figure 36.23 Diagram of Circuit that Generates the Display Timing from an External Sync Signal

36.4.5 Restrictions on Changing the Synchronization of the External SYNC Signal [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

When electrical characteristics (AC spec.) related to the acquisition of an external signal are not satisfied, ensure an interval of at least two cycles between changes of EXHSYNC or EXVSYNC, i.e. the external SYNC signals.

The frequency-divided dot clock is used as the basis of the cycle period when bit 12 in DEFRn is 0. The pre-division dot clock is used when bit 12 is 1.

36.4.6 Note on Register Settings

At the time of a reset, some registers of the DU have fixed initial values but others do not. The initial values of the latter when the power for the LSI is turned on can be either 0 or 1. From the initial state after power is switched on, be sure to set all of the registers in the DU to the desired values before starting synchronization of the display*¹.

If turning on the display*² is attempted while values in registers are unknown, operation may be incorrect due to access by the DU to areas other than those intended (the DU is capable of access to any area).

- Notes: 1. Setting (a) or (b) starts synchronization of the display.
- (a) The DRES and DEN bits in DSYSRn are B'00.
 - (b) The DRES and DEN bits in DSYSRn are B'01.
2. Setting (a) or (b) turns the display on.
- (a) When the DPRS bit in DORCRm is 0:
With setting (b) in note 1, setting any of bits DPE2 and DPE1 in DPPRm to 1.
 - (b) When the DPRS bit in DORCRm is 1:
With setting (b) in note 1, setting bits S0S2 and S0S1 in DSPR0 and DSPR2 or bits S1S2 and S1S1 in DSPR1 and DSPR3 to any value from B'0001 and B'0011.

36.4.7 Transition to Display operation stop by DRES bit in DSYSRn

(1) Display screen is not disturbed

1. Execute the procedure of 1. and 2. of section 36.4.2.
2. Set the DRES and DEN bits in DSYSRn to B'10.

(2) Display screen is disturbed

1. Set the DRES and DEN bits in DSYSRn to B'10.

2. Execute software reset to VSP module.

37. LVDS Interface

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

37.1 Overview

The LVDS-IF (low voltage differential signaling) module converts an RGB signal output by the DU module to the LVDS format and outputs those signals.

The LVDS-IF interface supports 8 output data formats with the conversion formats selected by register settings. The output control signals can also be selected freely.

The LVDS-IF supports the Dual-link output by using vertical stripe output function. [RZ/G2E]

37.1.1 Features

This module has the following features.

- Output pins: Five differential output pairs (4 data and 1 clock) that conform to the TIA/EIA-644 standard.
- Number of channels:
RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 1ch
RZ/G2E: 2ch
- Operating frequency: dot clock frequency is from 31MHz to 148.5 MHz. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
Operating frequency: dot clock frequency is from 5 MHz to 148.5 MHz. [RZ/G2E]
- Supports eight output data formats
- Supports the Dual-link output by using vertical stripe output function. [RZ/G2E]

37.1.2 Block Diagram

Figure 37.1 and Figure 37.2 show the block diagrams of the LVDS-IF module.

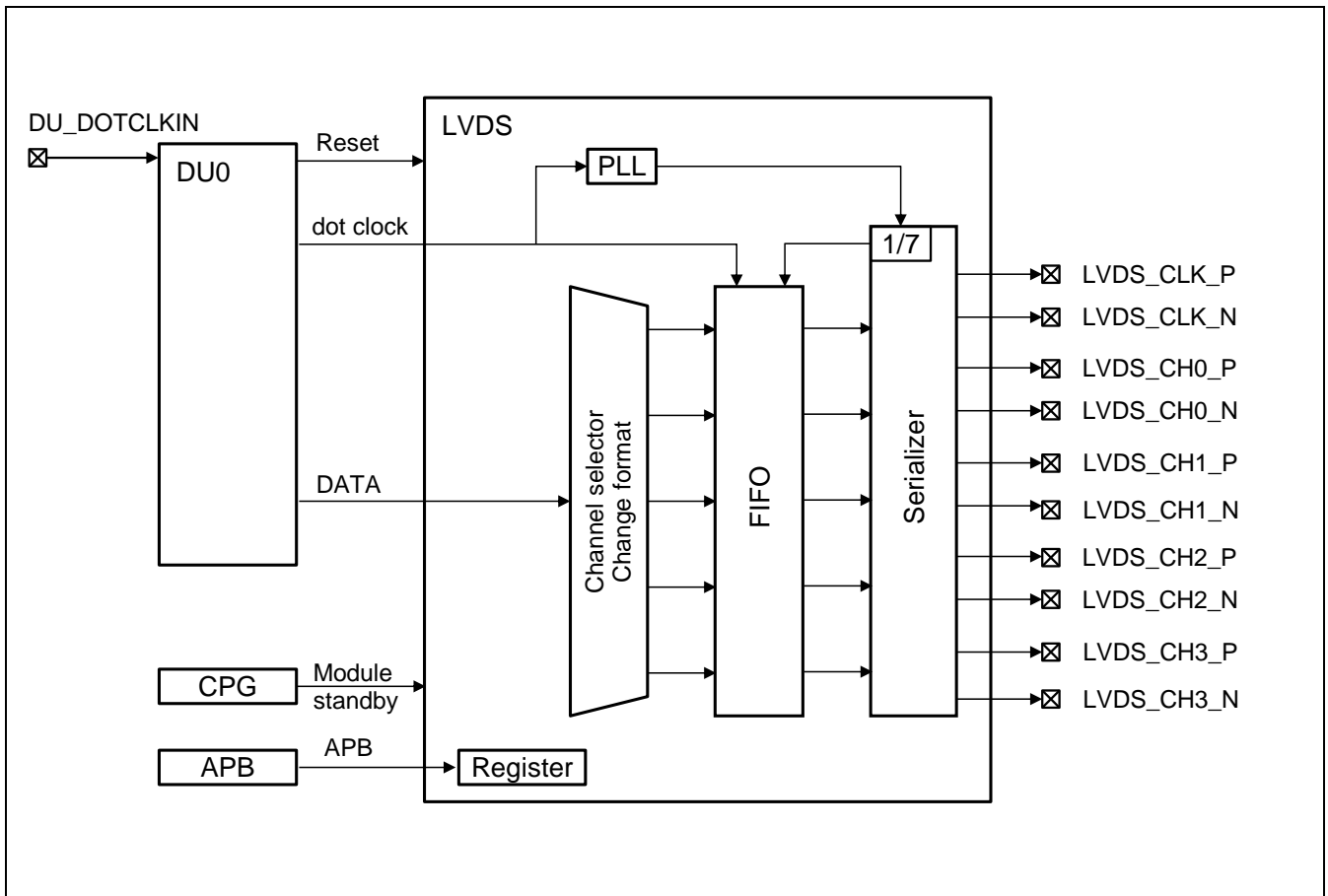


Figure 37.1 Block Diagram of LVDS-IF module for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N

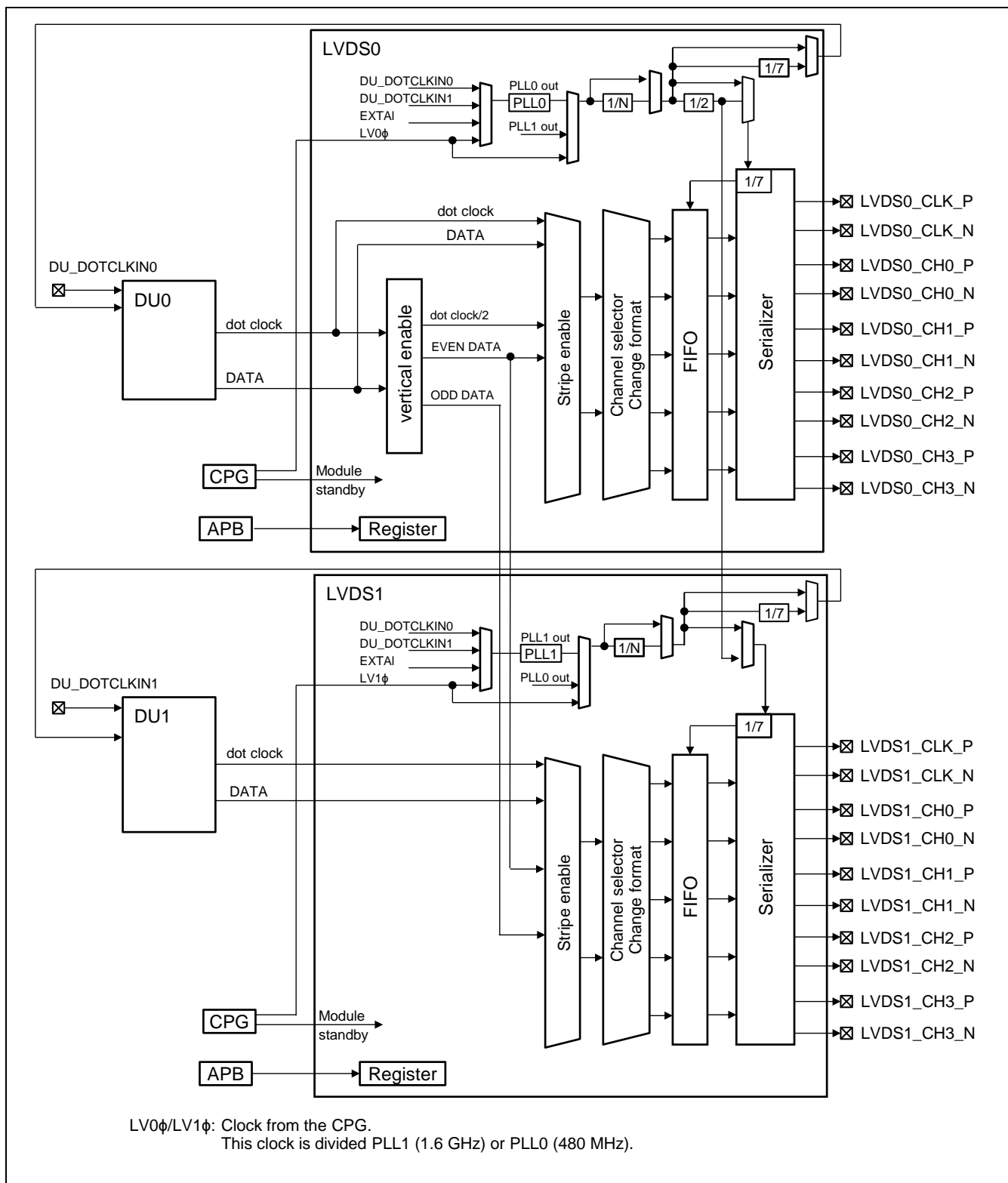


Figure 37.2 Block Diagram of LVDS-IF module for RZ/G2E

37.1.3 External Pins

Table 37.1 Pin Configuration

| Name | Pin name | I/O | Function | Second Generation RZ/G Series Products | | | | |
|-------------|-------------|--------|----------------------------------|--|----------------------------|--------|--------|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| LVDS output | LVDS0_CLK_P | Output | LVDS differential clock (pos) | √ | √ | √ | √ | |
| LVDS output | LVDS0_CLK_N | Output | LVDS differential clock (neg) | √ | √ | √ | √ | |
| LVDS output | LVDS0_CH0_P | Output | LVDS differential ch0 data (pos) | √ | √ | √ | √ | |
| LVDS output | LVDS0_CH0_N | Output | LVDS differential ch0 data (neg) | √ | √ | √ | √ | |
| LVDS output | LVDS0_CH1_P | Output | LVDS differential ch1 data (pos) | √ | √ | √ | √ | |
| LVDS output | LVDS0_CH1_N | Output | LVDS differential ch1 data (neg) | √ | √ | √ | √ | |
| LVDS output | LVDS0_CH2_P | Output | LVDS differential ch2 data (pos) | √ | √ | √ | √ | |
| LVDS output | LVDS0_CH2_N | Output | LVDS differential ch2 data (neg) | √ | √ | √ | √ | |
| LVDS output | LVDS0_CH3_P | Output | LVDS differential ch3 data (pos) | √ | √ | √ | √ | |
| LVDS output | LVDS0_CH3_N | Output | LVDS differential ch3 data (neg) | √ | √ | √ | √ | |
| LVDS output | LVDS1_CLK_P | Output | LVDS differential clock (pos) | — | — | — | √ | |
| LVDS output | LVDS1_CLK_N | Output | LVDS differential clock (neg) | — | — | — | √ | |
| LVDS output | LVDS1_CH0_P | Output | LVDS differential ch0 data (pos) | — | — | — | √ | |
| LVDS output | LVDS1_CH0_N | Output | LVDS differential ch0 data (neg) | — | — | — | √ | |
| LVDS output | LVDS1_CH1_P | Output | LVDS differential ch1 data (pos) | — | — | — | √ | |
| LVDS output | LVDS1_CH1_N | Output | LVDS differential ch1 data (neg) | — | — | — | √ | |
| LVDS output | LVDS1_CH2_P | Output | LVDS differential ch2 data (pos) | — | — | — | √ | |
| LVDS output | LVDS1_CH2_N | Output | LVDS differential ch2 data (neg) | — | — | — | √ | |
| LVDS output | LVDS1_CH3_P | Output | LVDS differential ch3 data (pos) | — | — | — | √ | |
| LVDS output | LVDS1_CH3_N | Output | LVDS differential ch3 data (neg) | — | — | — | √ | |

Note: All LVDS output pins are HiZ state before startup.

37.1.4 Register Configuration

Table 37.2 Address Configuration

| Ch* | Name of Register | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|-----|----------------------------------|---------------------------|-----|-------------|---------------|-------------|--|----------------------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 0 | LVDS-IF Control Register 0 | LVDCR0* ¹ | R/W | H'FEB9_0000 | H'0000_0000 | 32 | √ | √ | √ | √ |
| 0 | LVDS-IF Control Register 1 | LVDCR1* ¹ | R/W | H'FEB9_0004 | H'0000_0000 | 32 | √ | √ | √ | √ |
| 0 | PLL Control Register | LVDPLLCR* ³ | R/W | H'FEB9_0008 | H'0000_0000 | 32 | √ | √ | √ | √ |
| 0 | CTR Control Register | LVDCRTR* ¹ | R/W | H'FEB9_000C | H'0000_0000 | 32 | √ | √ | √ | √ |
| 0 | CH Control Register | LVDCRCH* ¹ | R/W | H'FEB9_0010 | H'0000_0000 | 32 | √ | √ | √ | √ |
| 0 | Vertical Stripe Control Register | LVDCRSTR* ¹ | R/W | H'FEB9_0014 | H'0000_0000 | 32 | — | — | — | √ |
| 0 | SSCG Control Register | LVDCRSCR* ³ | R/W | H'FEB9_0018 | H'0000_0000 | 32 | — | — | — | √ |
| 0 | DIVIDER Register | LVDCRDIV* ³ | R/W | H'FEB9_001C | H'0000_0000 | 32 | — | — | — | √ |
| 1 | LVDS-IF Control Register 0 | LVDCR0* ² | R/W | H'FEB9_0100 | H'0000_0000 | 32 | — | — | — | √ |
| 1 | LVDS-IF Control Register 1 | LVDCR1* ² | R/W | H'FEB9_0104 | H'0000_0000 | 32 | — | — | — | √ |
| 1 | PLL Control Register | LVDCR1PLLCR* ³ | R/W | H'FEB9_0108 | H'0000_0000 | 32 | — | — | — | √ |
| 1 | CTR Control Register | LVDCR1CTR* ² | R/W | H'FEB9_010C | H'0000_0000 | 32 | — | — | — | √ |
| 1 | CH Control Register | LVDCR1CH* ² | R/W | H'FEB9_0110 | H'0000_0000 | 32 | — | — | — | √ |
| 1 | Vertical Stripe Control Register | LVDCR1STR* ² | R/W | H'FEB9_0114 | H'0000_0000 | 32 | — | — | — | √ |
| 1 | SSCG Control Register | LVDCR1SCR* ³ | R/W | H'FEB9_0118 | H'0000_0000 | 32 | — | — | — | √ |
| 1 | DIVIDER Register | LVDCR1DIV* ³ | R/W | H'FEB9_011C | H'0000_0000 | 32 | — | — | — | √ |

Notes: [RZ/G2E]

1. This register in the LVDS0-IF is controlled by software reset register SRCR7[27] and SRSTCLR7[27].
2. This register in the LVDS1-IF is controlled by software reset register SRCR7[26] and SRSTCLR7[26].
3. This register for clock in the LVDS0-IF and the LVDS1-IF is controlled by software reset register SRCR7[25] and SRSTCLR7[25].

37.1.5 Connected Module**Table 37.3 Connected module**

| Module name | Connected module name | Function of connected module |
|--------------------|------------------------------|-------------------------------------|
| LVDS-IF | CPG | Clock Pulse Generator |
| | DU | Display Unit |

37.2 Register Description

Explanation of abbreviation of register

- Initial value: Value of the register after power-on reset
- : Undefined value
- R/W: The bit or field is readable and writable.
- R/WC0: The bit or field is readable and writable. Writing 0 to the bit initializes the bit.
Writing 1 to the bit is ignored.
- R: The bit or field is readable only. When writing to the register, write 0 to it.
- /WB: The bit or field is writable. Note that values read from write-only bits are not guaranteed.

37.2.1 LVDS-IF Control Register 0 (LVDCR0 / LVD1CR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

LVDCR0 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

LVD1CR0 [RZ/G2E]

| | | | | | | | | | | | | | | | | |
|----------------|-----|----|----|----|------|-----|-----|-----|----|----|----|-------------|----|---------|------------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | LVMD | | | | — | — | — | PLLON/ — | — | PWD/— | —/ LVEN | LVRES |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R/W / R | R | R/W / R | R / R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | — | B'0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 14 to 12 | — | B'000 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 11 to 8 | LVMD | H'0 | R/W | LVDS-IF Mode Selects the LVDS-IF module output data format (see Figure 37.3). B'0000: MODE0 B'0001: MODE1 B'0010: MODE2 B'0011: MODE3 B'0100: MODE4 B'0101: MODE5 B'0110: MODE6 B'0111: MODE7 All other values: Setting prohibited |
| 7 to 5 | — | B'000 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | PLLON | B'0 | R/W | PLL Enable [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Controls the PLL operation. 0: PLL stopped. 1: PLL operates. |
| | — | B'0 | R | Reserved [RZ/G2E] This bit is always read as 0. The write value should always be 0. |
| 3 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 2 | PWD | B'0 | R/W | Power Down mode [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Controls the LVDS Power Down circuit operation. 0: Power Down mode 1: Normal mode |
| | — | B'0 | R | Reserved [RZ/G2E] This bit is always read as 0. The write value should always be 0. |
| 1 | — | B'0 | R | Reserved [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] This bit is always read as 0. The write value should always be 0. |
| | LVEN | B'0 | R/W | LVDS-IF PHY Enable Bit [RZ/G2E] Controls the LVDS PHY. 0: PHY off 1: PHY on |
| 0 | LVRES | B'0 | R/W | LVDS-IF Reset Bit Controls the LVDS output. 0: Output off 1: Output on |

37.2.2 LVDS-IF Control Register 1 (LVDCR1 / LVD1CR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

LVDCR1 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

LVD1CR1 [RZ/G2E]

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---------|---------|---------|---------|---------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | CH3STBY | CH2STBY | CH1STBY | CH0STBY | CLKSTBY | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9, 8 | CH3STBY | B'00 | R/W | CH3 Control Controls the CH3 pin. B'00: Standby mode B'11: Operating mode Other settings are prohibited. |
| 7, 6 | CH2STBY | B'00 | R/W | CH2 Control Controls the CH2 pin. B'00: Standby mode B'11: Operating mode Other settings are prohibited. |
| 5, 4 | CH1STBY | B'00 | R/W | CH1 Control Controls the CH1 pin. B'00: Standby mode B'11: Operating mode Other settings are prohibited. |
| 3, 2 | CH0STBY | B'00 | R/W | CH0 Control Controls the CH0 pin. B'00: Standby mode B'11: Operating mode Other settings are prohibited. |
| 1, 0 | CLKSTBY | B'00 | R/W | CLK Control Controls the CLK pin. B'00: Standby mode B'11: Operating mode Other settings are prohibited. |

37.2.3 PLL Control Register (LVDPLLCR / LVD1PLLCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

LVDPLLCR [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

LVD1PLLCR [RZ/G2E]

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| | | | | | | | | | | | | | | | | |
|----------------|------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | PLLDIVCNT[18:16] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PLLDIVCNT [15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 19 | — | All 0 | R | Reserved [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] These bits are always read as 0. The write value should always be 0. |
| 18 to 0 | PLLDIVCNT | All 0 | R/W | PLL Setting [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Sets the PLL multiplication rate. H'0 14CB: 31 MHz <= dot clock < 42 MHz H'0 0A45: 42 MHz <= dot clock < 85 MHz H'0 06C3: 85 MHz <= dot clock < 128 MHz H'0 46C1: 128 MHz <= dot clock <= 148.5 MHz Other settings are prohibited. |

[RZ/G2E]

| | | | | | | | | | | | | | | | | |
|----------------|----|-------------|----|------------|---------|------|------|-----|-----|-------|--------|-----|-------|------|-----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | PLLON | PLLSEL | | CKSEL | | | OCKSEL |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | STP_CLKOUTE | — | OUT_CLKSEL | CLK_OUT | PLLE | PLLN | | | | | | | PLLM | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 23 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|---|
| 22 | PLLON | B'0 | R/W | PLL0 Enable Controls the PLL operation. 0: PLL stopped. 1: PLL operates. |
| 21, 20 | PLLSEL | All 0 | R/W | Operating clock (LVDPLLCCR register in the LVDS0-IF) B'00: PLL0 B'10: PLL1 *1: LV0 ϕ from CPG module Operating clock (LVD1PLLCCR register in the LVDS1-IF) B'00: PLL1 B'10: PLL0 *1: LV1 ϕ from CPG module |
| 19 to 17 | CKSEL | All 0 | R/W | Input clock (LVDPLLCCR register in the LVDS0-IF) B'001: LV0 ϕ clock from CPG module B'011: EXTAL B'101: DU_DOTCLKIN0 B'111: DU_DOTCLKIN1 Others: Setting prohibited Input clock (LVD1PLLCCR register in the LVDS1-IF) B'001: LV1 ϕ clock from CPG module B'011: EXTAL B'101: DU_DOTCLKIN0 B'111: DU_DOTCLKIN1 Others: Setting prohibited |
| 16 | OCKSEL | B'0 | R/W | Output clock Selector 0: 7 divided Select 7 divided when the LVDS-IF enable. 1: not divided Select not divided when the digital output from pin (DPAD) |
| 15 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 14 | STP_ CLKOUTE | B'0 | R/W | Output Enable of E-Divider 0: Disable 1: Enable Setting the same value in OUTCLKSEL. (see section 37.3.6 PLL Setting) |
| 13 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 12 | OUTCLKSEL | B'0 | R/W | Out clock Selector 0: Clock before E-Divider 1: Clock after E-Divider (see section 37.3.6 PLL Setting) |
| 11 | CLKOUT | B'0 | R/W | Output Enable of the 7 divided. 0: Disable Select Disable when the digital output from pin (PAD) 1: Enable Select Enable, when the LVDS-IF enable. |
| 10 | PLLE | B'0 | R/W | E-Divider Setting (output divider) (see section 37.3.6 PLL Setting) |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 9 to 3 | PLLN | All 0 | R/W | N-Divider Setting (feedback divider) (see section 37.3.6 PLL Setting) |
| 2 to 0 | PLLM | All 0 | R/W | M-Divider Setting (input divider) (see section 37.3.6 PLL Setting) |

37.2.4 CTR Control Register (LVDCTRCR / LVD1CTRCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

LVDCTRCR [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

LVD1CTRCR [RZ/G2E]

| | | | | | | | | | | | | | | | | |
|----------------|----|---------|-----|-----|----|---------|-----|-----|----|---------|-----|-----|----|---------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | CTR3SEL | | | — | CTR2SEL | | | — | CTR1SEL | | | — | CTR0SEL | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 15 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 14 to 12 | CTR3SEL | B'000 | R/W | Ctrl3 Select Selects data to be output to Ctrl3 (see section 37.3.2 Ctrl Signal Selection). B'000: 0 B'001: Odd/even B'010: CDE Other settings are prohibited. |
| 11 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 10 to 8 | CTR2SEL | B'000 | R/W | Ctrl2 Select Selects data to be output to Ctrl2 (see section 37.3.2 Ctrl Signal Selection). B'000: DISP B'001: Odd/even B'010: CDE B'011: HSYNC B'100: VSYNC Other settings are prohibited. |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 to 4 | CTR1SEL | B'000 | R/W | Ctrl1 Select Selects data to be output to Ctrl1 (see section 37.3.2 Ctrl Signal Selection). B'000: VSYNC B'001: DISP B'010: Odd/even B'011: CDE B'100: HSYNC Other settings are prohibited. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 3 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 2 to 0 | CTR0SEL | B'000 | R/W | Ctrl0 Select Selects data to be output to Ctrl0 (see section 37.3.2 Ctrl Signal Selection). B'000: HSYNC B'001: VSYNC B'010: DISP B'011: Odd/even B'100: CDE Other settings are prohibited. |

37.2.5 CH Control Register (LVDCHCR / LVD1CHCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

LVDCHCR [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

LVD1CHCR [RZ/G2E]

| | | | | | | | | | | | | | | | | |
|----------------|----|----|--------|-----|----|--------|-----|-----|--------|----|-----|--------|----|----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | CH3SEL | — | — | CH2SEL | — | — | CH1SEL | — | — | CH0SEL | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R | R | R/W | R/W | R | R | R/W | R/W | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13, 12 | CH3SEL | B'00 | R/W | CH3 Select Selects data to be output to LVDS*_C_P/N (see section 37.3.3 CH Selection). B'00: CH3 B'01: CH0 B'10: CH1 B'11: CH2 |
| 11, 10 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9, 8 | CH2SEL | B'00 | R/W | CH2 Select Selects data to be output to LVDS*_CH2_P/N (see section 37.3.3 CH Selection). B'00: CH2 B'01: CH3 B'10: CH0 B'11: CH1 |
| 7, 6 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5, 4 | CH1SEL | B'00 | R/W | CH1 Select Selects data to be output to LVDS*_CH1_P/N (see section 37.3.3 CH Selection). B'00: CH1 B'01: CH2 B'10: CH3 B'11: CH0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 3, 2 | — | B'00 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1, 0 | CH0SEL | B'00 | R/W | CH0 Select Selects data to be output to LVDS*_CH0_P/N (see section 37.3.3 CH Selection). B'00: CH0 B'01: CH1 B'10: CH2 B'11: CH3 |

37.2.6 Vertical Stripe Control Register (LVDSTRIPE / LVD1STRIPE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|-------------|---------|-----------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | ST_TRGSEL/— | — | ST_SWAP/— | ST_ON |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W / R | R/W / R | R/W / R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3, 2 | ST_TRGSEL | B'00 | R/W | Trigger selection for EVEN DATA and ODD DATA separated (LVDSTRIPE register in the LVDS0-IF) B'00: DISP rising edge (default) B'01: HSYNC rising edge B'10: HSYNC falling edge B'11: Setting prohibited |
| | — | B'00 | R | Reserved (LVD1STRIPE register in the LVDS1-IF) These bits are always read as 0. The write value should always be 0. |
| 1 | ST_SWAP | B'0 | R/W | Odd/even swap mode (LVDSTRIPE register in the LVDS0-IF) 0: Normal LVDS0_* ports output EVEN DATA (It corresponds to the first pixel) LVDS1_* ports output ODD DATA (It corresponds to the second pixel) 1: swap LVDS0_* ports output ODD DATA (It corresponds to the second pixel) LVDS1_* ports output EVEN DATA (It corresponds to the first pixel) |
| | — | B'0 | R | Reserved (LVD1STRIPE register in the LVDS1-IF) This bit is always read as 0. The write value should always be 0. |
| 0 | ST_ON | B'0 | R/W | Vertical stripe output 0: Disable 1: Enable |

37.2.7 SSCG Control Register (LVDSR / LVD1SCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|-------|-----|----------|--------|-----|-----|------|------|------|-----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | DEPTH | | BAND SET | TWGCNT | | | SDIV | MODE | RSTN | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 30, 29 | DEPTH | All 0 | R/W | PLL SSCG modulation depth control parameter Dp B'00: Dp = 1 B'01: Dp = 2 B'10: Dp = 3 B'11: Setting prohibited (see section 37.3.6 PLL Setting). |
| 28 | BANDSET | B'0 | R/W | PLL SSCG modulation frequency band control 0: Fmod > 10 kHz 1: Fmod ≤ 10 kHz Fmod: Modulation Frequency (see section 37.3.6 PLL Setting). |
| 27 to 24 | TWGCNT | H'0 | R/W | PLL SSCG modulation frequency control parameter Cv Cv = 256 + 16 × TWGCNT[3:0] B'0000: Cv = 256 B'0001: Cv = 272 .. B'1110: Cv = 480 B'1111: Cv = 496 (see section 37.3.6 PLL Setting). |
| 23, 22 | SDIV | All 0 | R/W | PLL SSCG modulation frequency control parameter Sr Sr = 2 ^{SDIV[1:0]} B'00: Sr = 1 B'01: Sr = 2 B'10: Sr = 4 B'11: Sr = 8 (see section 37.3.6 PLL Setting). |
| 21 | MODE | B'0 | R/W | PLL SSCG frequency dithering mode selection 0: Center spread 1: Down spread (see section 37.3.6 PLL Setting). |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 20 | RSTN | B'0 | R/W | PLL SSCG mode control 0: Disabled 1: Enable |
| 19 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

37.2.8 LVDS DIVIDER Register (LVDDIV/ LVD1DIV)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|------------|--------------|------------|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | DIV SEL | DIV RESET | DIV STP | DIV | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | DIVSEL | B'0 | R/W | Division Selector 0: Bypass 1: Division |
| 7 | DIVRESET | B'0 | R/W | Division RESET 0: Reset 1: Clear |
| 6 | DIVSTP | B'0 | R/W | Division Stop Control 0: Enable 1: Stop |
| 5 to 0 | DIV | All 0 | R/W | Frequency division of the output 0: Setting prohibited 1: Division by 2 63: Division by 64 |

37.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The LVDS-IF module converts the RGB signals output by DU to the LVDS format and outputs those signals. Since the output data format can be selected by register settings, this module can convert to any of the LVDS formats. The output data format is determined by (1) the mode selection, (2) the Ctrl signal selection, and (3) the CH selection.

The registers concerned with the output data format must be set before the LVDS-IF module is started and must not be changed during module operation. (See section 37.3.7 Setting Procedure, for details.)

37.3.1 Mode Selection

The mode is selected by the LDMD bits in the LVDS-IF control register 0. Figure 37.3 shows the modes that can be set.

Here, R0 to R7, G0 to G7, and B0 to B7 are the RGB signals and Ctrl0 to Ctrl3 are control signals (such as HSYNC and VSYNC). The Ctrl signals can be set by the CTR control register.

CH0, CH1, CH2, CLK and CH3 are buffers that hold data temporarily. The default is for the CH0, CH1, CH2, CLK and CH3 data to be output directly without change to LVDS0_CH0_P/N, LVDS0_CH1_P/N, LVDS0_CH2_P/N, LVDS0_CLK_P/N and LVDS0_CH3_P/N. The CH assignment can be switched with the CH control register settings.

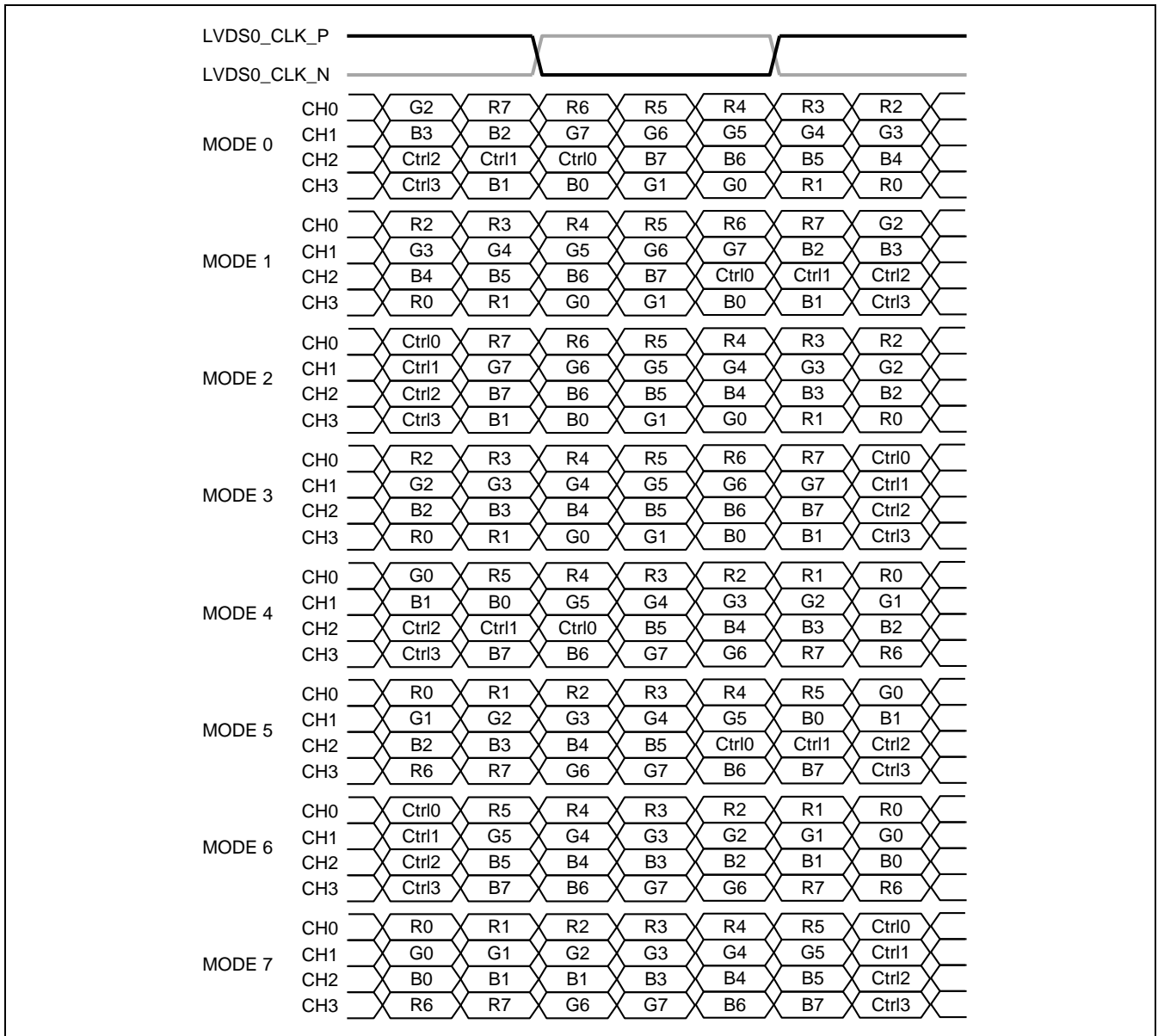


Figure 37.3 Output Data Format

37.3.2 Ctrl Signal Selection

The two stage settings shown below are required to select the Ctrl signals.

1. DU Output Signal Selection

The DU output ports are multiplexed. The signals that you want to output are set up with the corresponding register. The specified signals are input to the LVDS-IF module ports.

2. LVDS Port Selection

The LVDS ports used to output to the Ctrl signals are selected with the CTR control register (LVDCTRCR). This results in the signals input to the LVDS ports being output as Ctrl signals.

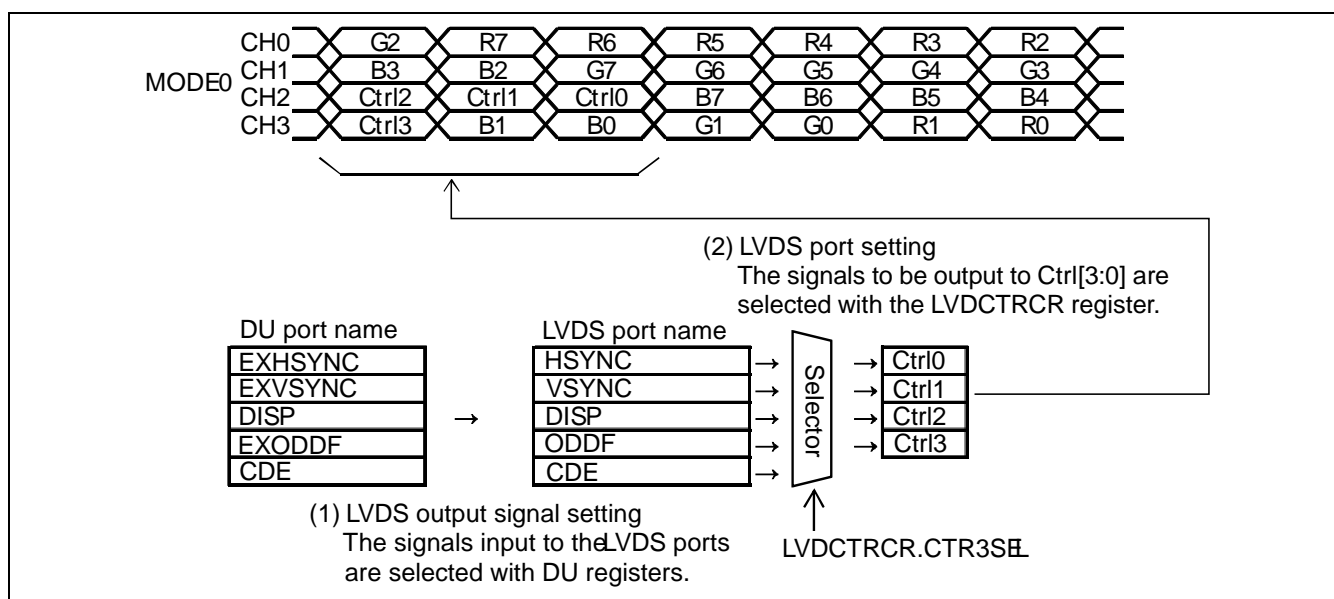


Figure 37.4 Ctrl Signal Selection

Set the desired Ctrl signal characteristics with the DU registers.

For example, to reverse the polarity of the HSYNC signal, the polarity must be set with DU register settings. (The LVDS-IF module only converts the signals output from DU to the LVDS format.)

An example of Ctrl signal settings is shown below.

- Ctrl0 = HSYNC, Ctrl1 = VSYNC, and Ctrl2 = DISP

Set the DU registers so that HSYNC, VSYNC, and DISP are output.

Then set the CTR control register (LVDCTRCR) so that CTR0SEL = B'000, CTR1SEL = B'000, and CTR2SEL = B'000.

- Ctrl0 = CSYNC, Ctrl1 = ODDF, and Ctrl2 = CDE

Set the DU registers so that CSYNC, ODDF, and CDE are output.

Then set the LVDS-IF register (LVDCTRCR) so that CTR0SEL = B'000, CTR1SEL = B'010, and CTR2SEL = B'010.

(Note that since CSYNC can also be output from other ports, CTR0SEL must be set to match the port from which DU outputs CSYNC.)

37.3.3 CH Selection

The LVDS-IF module stores the RGB signal data in CH0, CH1, CH2, and CH3 according to the mode selection and Ctrl signal selection registers.

The CH0, CH1, CH2, and CH3 data is then stored in the CH0, CH1, CH2, and CH3 FIFOs according to the CH selection register setting. The data stored in the CH0, CH1, CH2, and CH3 FIFOs is output from the external pins after passing through the LVDS buffers.

37.3.4 Vertical stripe output [RZ/G2E]

The LVDS-IF supports the vertical stripe output separated the even data and the odd data by setting the Vertical Stripe Control Register (LVDSTRIPE.ST_ON = B'1 and LVD1STRIPE.ST_ON = B'1). Figure 37.5 shows the vertical stripe output. The LVDS0_* ports output the even data and the LVDS1_* ports output the odd data. The even data corresponds to the first pixel, and the odd data corresponds to the second pixel. Also, the even data and the odd data can swap by setting LVDSTRIPE.ST_SWAP = B'1.

The vertical stripe output is possible only from DU0 input.

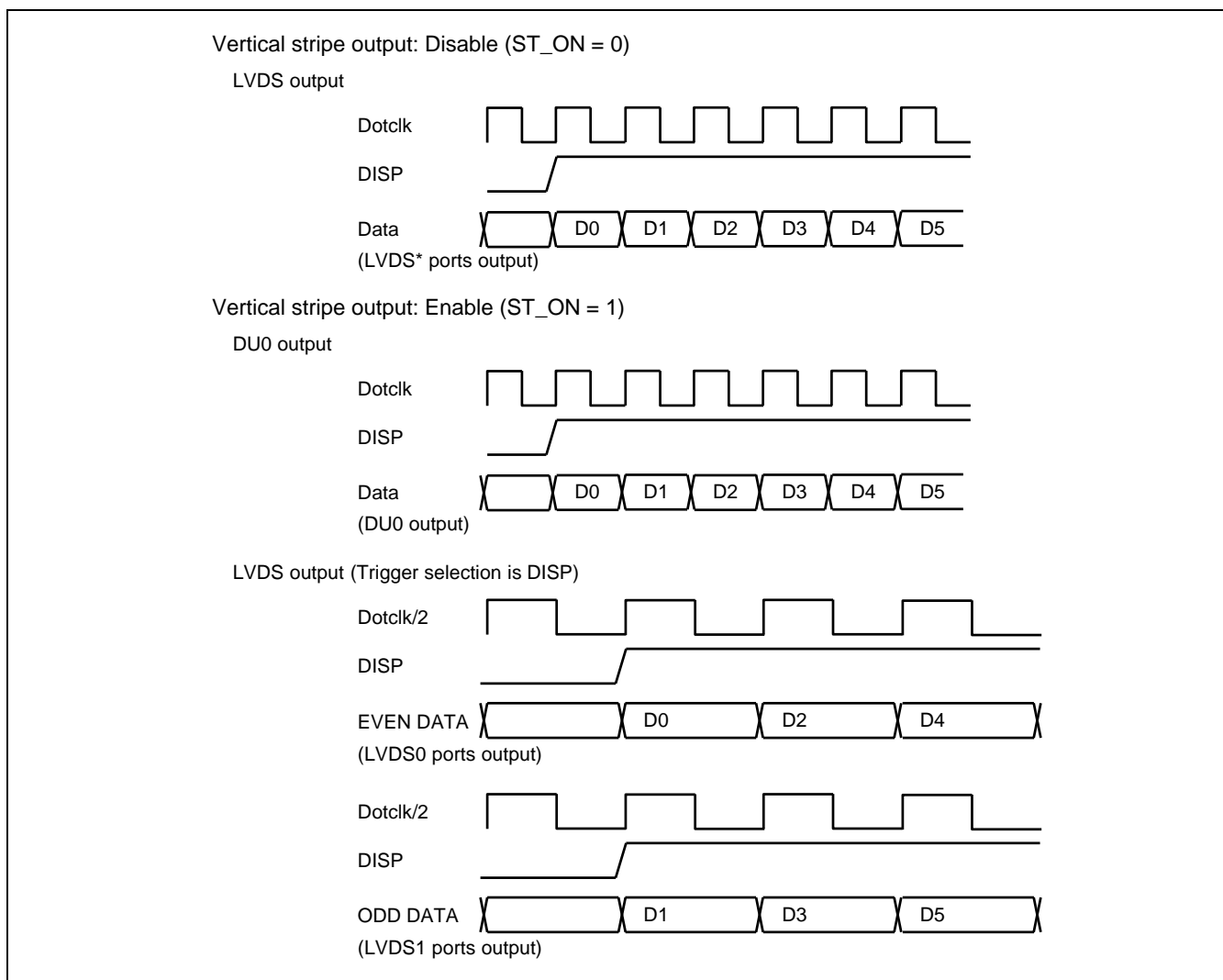


Figure 37.5 Vertical Stripe output for DUAL-Link

Note: The EVEN DATA corresponds to the first pixel, and the ODD DATA corresponds to the second pixel in Figure 37.5

37.3.5 Dot clock Settings

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

The LVDS-IF module dot clock frequency is determined by the clock source selected and the divisor settings. This section describes how this frequency is set.

The individual settings are made with the DU (Display Unit) registers. Refer to the corresponding sections for details on the registers, notes, and other information on these settings.

- Clock source selection
The clock source can be selected to be either the internal clock or an external input (DU_DOTCLKIN*). This is set with the DCLKSEL bit in the DU external synchronization control register (ESCR0).
- Divisor setting
This setting specifies the divisor applied to the clock source.

This is set with the FRQSEL field in the DU external synchronization control register (ESCR0).

[RZ/G2E]

This section describes how this frequency of the LVDS-IF module and the digital output from pin (DPAD) set. Figure 37.6 shows the path for dot clock.

Setting output clock from the LVDS-IF to the DU:

- Clock source of the PLL selection
The clock source of the PLL can be selected to be either the EXTAL, an external input (DU_DOTCLKIN0/1) or LV0 ϕ /1 ϕ clock from CPG. (LVDPLLCR.CKSEL / LVD1PLLCR.CKSEL)
- Frequency of the PLL setting (See Section 37.3.6 PLL Setting)
- 1/N divisor setting (LVDDIV / LVD1DIV)

When the LVDS-IF enable:

- Select 7 divider in the LVDS-IF.
(LVDS0: LVDPLLCR.OCKSEL = B'0 and LVDPLLCR.CLKOUT = B'1LVDS1: LVD1PLLCR.OCKSEL = B'0 and LVD1PLLCR.CLKOUT = B'1)
- Select the LVDS output clock in the DU0/1.
(DIDSR0.LVCS0/1 = B'1, ESCR0/1.DCLKSEL = B'0 and ESCR0/1.FRQSEL = B'0*)

Note: * However, when outputting from the LVDS0-IF as section (2) Display of Different Images on Monitors of the Same Size of 35.3.8 Dual Display Output, the output dot clock should be set to be division by two of the input dot clock. (ESCR0.FRQSEL = B'1)

When the digital output from pin (DPAD) enable:

- Clock source selection
The clock source can be selected to be either the internal clock or an external input (DU_DOTCLKIN0/1) or the LVDS-IF output clock. When select the LVDS-IF output clock, setting bypass 7 divider in the LVDS-IF. (LVDS0: LVDPLLCR.OCKSEL = B'1 and LVDPLLCR.CLKOUT = B'0, LVDS1: LVD1PLLCR.OCKSEL = B'1 and LVD1PLLCR.CLKOUT = B'0)
- Divisor setting in the DU0/1.
This setting specifies the divisor applied to the clock source.

This is set with the FRQSEL field in the DU external synchronization control register (ESCR0/1).

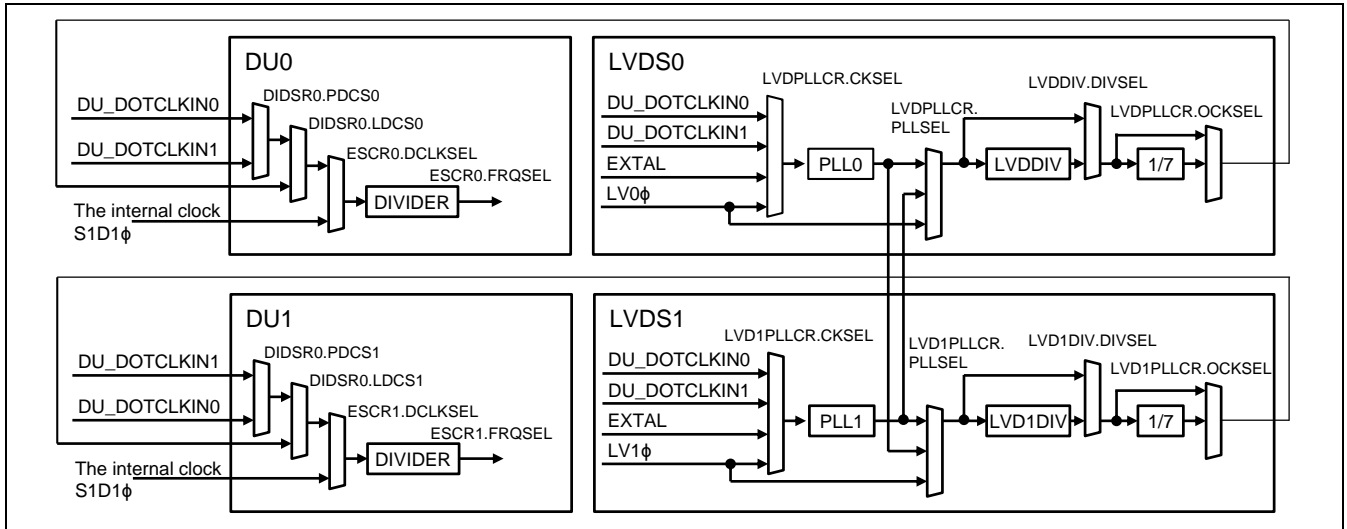


Figure 37.6 Path for dot clock [RZ/G2E]

37.3.6 PLL Setting [RZ/G2E]

1. PLL Frequency is calculated as the following.

- Not use E-Divide, Setting OUTCLKSEL = B'0 and STP_CLKOUTE = B'0

$$F_{out} = F_{in} (\text{Input Clock Frequency}) * (PLL_N+1) / (PLL_M+1)$$

- Use E-Divide, Setting OUTCLKSEL = B'1 and STP_CLKOUTE = B'1

$$F_{out} = F_{in} (\text{Input Clock Frequency}) * (PLL_N+1) / ((PLL_M+1) * (PLLE+1)^2)$$

- Limited Setting:

| | |
|---|------------------------------------|
| Input Clock Frequency Range: F_{in} | $12 \leq F_{in} \leq 192$ (MHz) |
| Comparison Frequency Range: $F_{pfd} = F_{in}/(PLL_M+1)$ | $12 \leq F_{pfd} \leq 24$ (MHz) |
| VCO operating range: $F_{vco} = F_{in} * (PLL_N+1)/(PLL_M+1)$ | $900 \leq F_{vco} \leq 1800$ (MHz) |
| Output Frequency Range: F_{out} | $F_{out} \leq 1039.5$ (MHz) |
| N-Divider Setting: PLLN | $60 \leq PLL_N+1 \leq 120$ |

2. Modulation frequency of SSCG Function is calculated as the following.

- Modulation Frequency(F_{mod})

$$F_{mod} = F_{pfd} / (C_v * S_r)$$

$$C_v = 256 + 16 * TWGCNT[3:0]$$

$$S_r = 2^{SDIV[1:0]}$$

$$\text{Comparison Frequency Range: } F_{pfd} = F_{in} / (PLL_M+1)$$

- F_{mod} Limited

$$F_{mod} \leq F_{pfd} / 450, 3\text{kHz} \leq F_{mod} \leq 30\text{kHz}$$

- BANDSET Setting

When $F_{mod} > 10\text{kHz}$, Setting BANDSET = B'0

When $F_{mod} \leq 10\text{kHz}$, Setting BANDSET = B'1

- Mode and Modulation Depth Setting (%)

$$\text{MODE} = B'0 (\text{Center spread}) \text{DEPTH}[1:0] = B'00 \quad \pm 0.5\%$$

$$\text{MODE} = B'0 (\text{Center spread}) \text{DEPTH}[1:0] = B'01 \quad \pm 1.0\%$$

$$\text{MODE} = B'0 (\text{Center spread}) \text{DEPTH}[1:0] = B'10 \quad \pm 1.5\%$$

$$\text{MODE} = B'1 (\text{Down spread}) \text{DEPTH}[1:0] = B'00 \quad \text{PLL}_N = 59-78 \quad -1.17\% \sim 0.17\% *^1$$

$$\text{MODE} = B'1 (\text{Down spread}) \text{DEPTH}[1:0] = B'01 \quad \text{PLL}_N = 79-119 \quad -1.0\%$$

$$\text{MODE} = B'1 (\text{Down spread}) \text{DEPTH}[1:0] = B'01 \quad -2.0\%$$

$$\text{MODE} = B'1 (\text{Down spread}) \text{DEPTH}[1:0] = B'10 \quad -3.0\%$$

Note 1. In this region, the modulation depth increase. The unwished effect results from the limitation of circuit architecture.

3. Setup sequence

- When not using SSC function, setup sequence is constraint free. After the parameter setting, the PLL output stable after 200 μs .
- Figure 37.7 shows setup sequence, when using SSC Function. The parameter cannot be changed after releasing LVDSCR / LVD1SCR.RSTN.

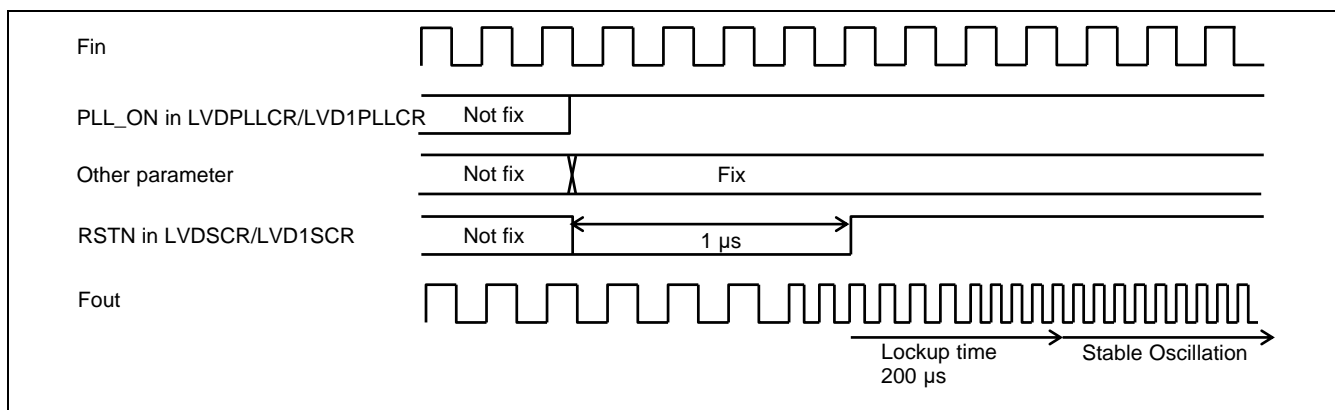


Figure 37.7 Setup sequence when using SSC Function

37.3.7 Setting Procedure

The procedure for using the LVDS-IF module is shown below.

Startup Procedure: Example of ch0

1. Clear the DU and LVDS-IF module standby states using CPG module stop control register.*¹
2. Set the DU registers.*²
3. Set the LVDS-IF registers other than LVDCR0.PLLON [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N], PWD, LVRES, and LVDCR1.CHnSTBY.*³
4. Set LVDCR1.CHnSTBY to B'11 to turn on the LVDS IO.
5. Set LVDCR0.PLLON to B'1 to turn on the PLL. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
6. Set LVDCR0.PWD to B'1 to turn on the LVDS Normal mode. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
7. Set LVDCR0.LVEN to B'1 to turn on the LVDS PHY. [RZ/G2E]
8. When all of the following conditions have been met, set LVDCR0.LVRES to B'1.
At least 100 μ s has elapsed since LVDCR0.PLLON and LVDCR0.PWD were set to 1. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
9. Signals will be output from the LVDS pins after LVDCR0.LVRES is set to B'1. *⁴

Turning the Display On or Off

Display Off

1. Set LVDCR0.LVRES to B'0 to turn the display off.
2. Set LVDCR0.LVEN to B'0 to turn the LVDS PHY. [RZ/G2E]
3. Set LVDCR0.PWD to B'0 to turn the LVDS Power Down mode.
4. Set LVDCR0.PLLON to B'0 to turn the LVDS-IF PLL circuit off.
5. Set LVDCR1.CHnSTBY to B'0 to turn the LVDS IO off.
6. Set CPG software reset register *⁵to B'1 to reset LVDS-IF module.

Display On

6. Set CPG software reset clearing register *⁵to B'1 to clear reset of LVDS-IF module.
7. Re-perform the above procedure [Startup Procedure]

- Notes:
1. For the DU and LVDS-IF modules, the initial state is the module standby state. When starting the module immediately after a reset, the module standby state must be cleared.
 2. Here, it is possible to proceed as long as the dot clock signal is output. (The LVDS-IF PLL must be started.) Other items may be set while waiting for the conditions of step 6 to be met. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
The Clock including the PLL must set before setting the DU registers. (See setup sequence in 37.3.6 PLL Setting.) Also, It is necessary to set the divider (LVDDIV/ LVD1DIV) after the PLL output stable. [RZ/G2E]
 3. This refers to settings other than those that are concerned with LVDS-IF startup. These items may be set while waiting for the conditions of step 6 to be met. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 4. When the vertical stripe output, set LVDCR0.LVRES after set LVD1CR0.LVRES. [RZ/G2E]
 5. When display on in the LVDS0-IF, it is necessary to reset (SRCR7[27]) and reset clearing (SRSTCLR7[27]) LVDS0-IF module.
When display on in the LVDS1-IF, it is necessary to reset (SRCR7[26]) and reset clearing (SRSTCLR7[26]) LVDS0-IF module.
When display on in the dual output operates, it is necessary to reset (SRCR7[27] and SRCR7[26]) and reset clearing (SRSTCLR7 [27] and SRSTCLR7 [26]) LVDS0-IF and LVDS1-IF module. [RZ/G2E]

Table 37.4 shows example of PLL setting. [RZ/G2E].

Figure 37.9 shows example of dual output and register setting. [RZ/G2E].

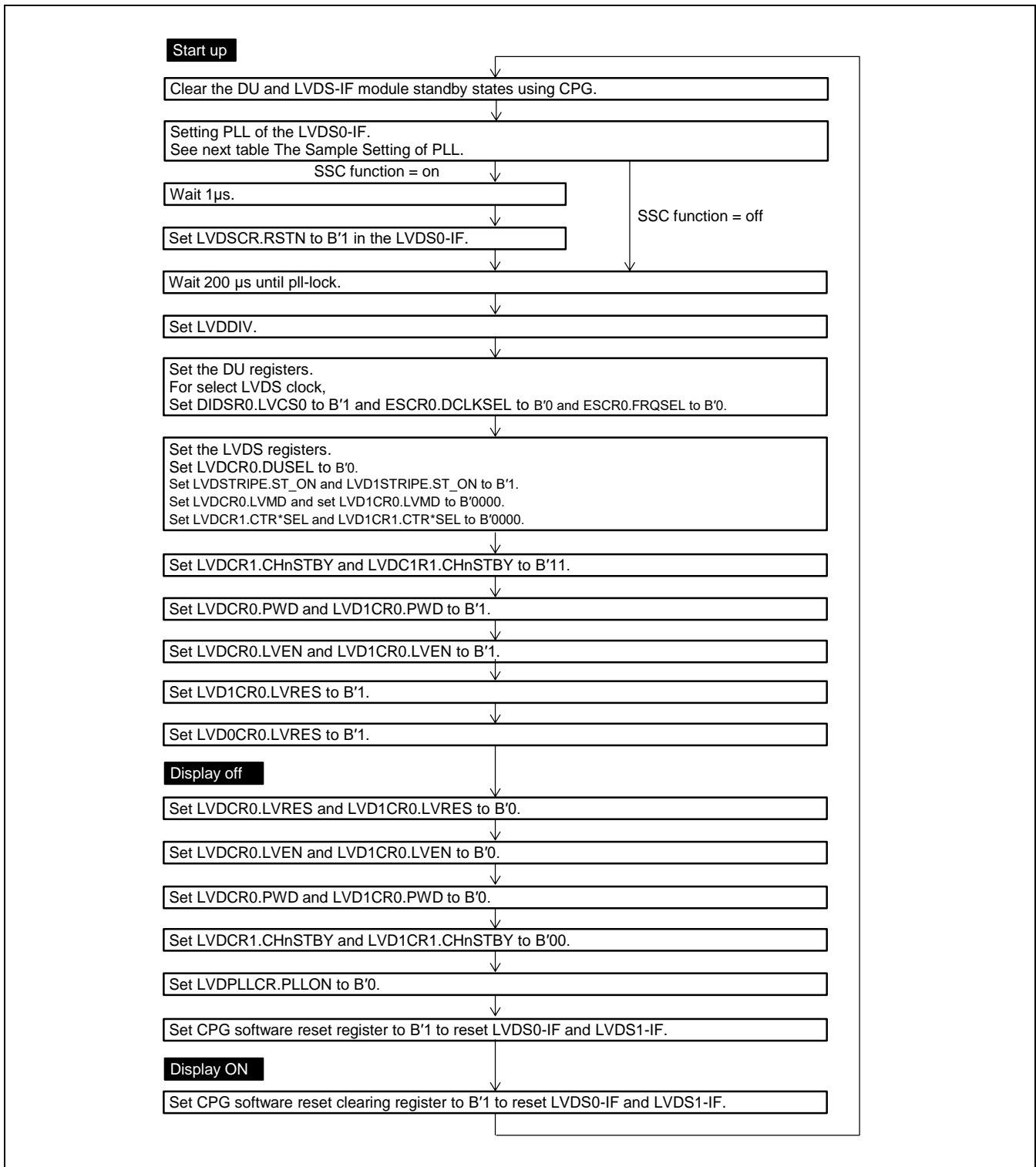


Figure 37.8 The sample setting of the vertical stripe output [RZ/G2E]

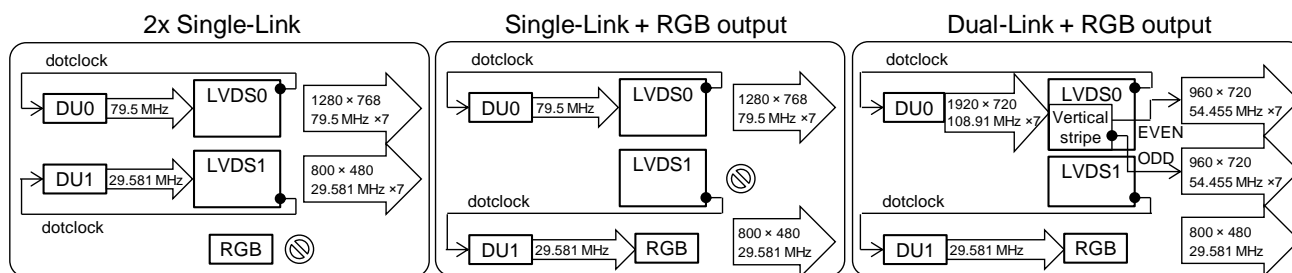
Table 37.4 Example of PLL setting [RZ/G2E]

| Display Panel (dotclock) | Select of Source clock | Fin (MHz) | Register Setting | | | RZ/G2E | |
|---|------------------------|-----------|-------------------------|----------------------|-------------|-----------|-------|
| | | | LVDPLLCR | LVDS ^{CR} 1 | LVDDIV | dot clock | fmod |
| | | | LVD1PLLCR | LVD1SCR ¹ | LVD1DIV | (MHz) | (MHz) |
| When LVDS-IF enable, Setting PLL | | | | | | | |
| 640x480:60 Hz (pixel clock 25.175 MHz) | EXTAL | 48.000 | H'0046_5A0A | H'4170_0000 | H'0000_0182 | 25.143 | 0.029 |
| 800x480:60 Hz (pixel clock 29.581 MHz) | EXTAL | 48.000 | H'0046_5E21 | H'4970_0000 | H'0000_0181 | 29.571 | 0.030 |
| 800x600:60 Hz (pixel clock 40.0 MHz) | EXTAL | 48.000 | H'0046_5A29 | H'4970_0000 | H'0000_0182 | 40.000 | 0.030 |
| 1024x768:60 Hz (pixel clock 65.0 MHz) | EXTAL | 48.000 | H'0046_5A5B | H'4D30_0000 | H'0000_0000 | 65.143 | 0.026 |
| 1280x768:60 Hz (pixel clock 79.5 MHz) | EXTAL | 48.000 | H'0046_5AE3 | H'4D30_0000 | H'0000_0000 | 79.714 | 0.026 |
| 1280x960:60 Hz (pixel clock 102.1 MHz) | EXTAL | 48.000 | H'0046_5BB3 | H'4D30_0000 | H'0000_0000 | 102.000 | 0.026 |
| 1280x1024:60 Hz (pixel clock 108.0 MHz) | EXTAL | 48.000 | H'0046_59F1 | H'4970_0000 | H'0000_0000 | 108.000 | 0.030 |
| 1920x720:60 Hz (pixel clock 108.910 MHz) | EXTAL | 48.000 | H'0046_5AF2 | H'4170_0000 | H'0000_0000 | 108.571 | 0.029 |
| 1920x1080:60 Hz (pixel clock 148.5MHz) | EXTAL | 48.000 | H'0046_0AAB | H'4D30_0000 | H'0000_0000 | 147.429 | 0.026 |
| 640x480:60 Hz (pixel clock 25.175 MHz) | DU_DOTCLKIN0/1 | 25.175 | H'004A_5F79 | H'4D30_0000 | H'0000_0181 | 25.175 | 0.027 |
| 800x480:60 Hz (pixel clock 29.581 MHz) | DU_DOTCLKIN0/1 | 29.581 | H'004A_5F79 | H'4070_0000 | H'0000_0181 | 29.581 | 0.029 |
| 800x600:60 Hz (pixel clock 40.0 MHz) | DU_DOTCLKIN0/1 | 40.000 | H'004A_5A99 | H'4570_0000 | H'0000_0182 | 40.000 | 0.030 |
| 1024x768:60 Hz (pixel clock 65.0 MHz) | DU_DOTCLKIN0/1 | 65.000 | H'004A_5A2C | H'4D30_0000 | H'0000_0000 | 65.000 | 0.028 |
| 1280x768:60 Hz (pixel clock 79.5 MHz) | DU_DOTCLKIN0/1 | 79.500 | H'004A_5A2C | H'4170_0000 | H'0000_0000 | 79.500 | 0.029 |
| 1280x960:60 Hz (pixel clock 102.1 MHz) | DU_DOTCLKIN0/1 | 102.100 | H'004A_5A9D | H'4270_0000 | H'0000_0000 | 102.100 | 0.030 |
| 1280x1024:60 Hz (pixel clock 108.0 MHz) | DU_DOTCLKIN0/1 | 108.000 | H'004A_5A9D | H'4370_0000 | H'0000_0000 | 108.000 | 0.030 |
| 1920x720:60 Hz (pixel clock 108.91.0 MHz) | DU_DOTCLKIN0/1 | 108.910 | H'004A_5A9D | H'4370_0000 | H'0000_0000 | 108.910 | 0.030 |
| 1920x1080:60 Hz (pixel clock 148.5MHz) | DU_DOTCLKIN0/1 | 74.250 | H'004A_0A2C | H'4070_0000 | H'0000_0000 | 148.500 | 0.029 |
| 1920x1080:60 Hz (pixel clock 148.5MHz) | DU_DOTCLKIN0/1 | 148.500 | It cannot be generated. | | | | |
| When the digital output from pin (DPAD) enable, Setting PLL | | | | | | | |
| 640x480:60 Hz (pixel clock 25.175 MHz) | EXTAL | 48.000 | H'0047_5752 | H'4170_0000 | H'0000_0190 | 25.176 | 0.029 |
| 800x480:60 Hz (pixel clock 29.581 MHz) | EXTAL | 48.000 | H'0047_01E2 | H'4170_0000 | H'0000_01A0 | 29.576 | 0.029 |
| 800x600:60 Hz (pixel clock 40.0 MHz) | EXTAL | 48.000 | H'0047_51D9 | H'4970_0000 | H'0000_0191 | 40.000 | 0.030 |

Notes: 1. Example of SSC function setting. Fmod = 0.03 MHz / Mode = Down spread (-3.0%)

When SSCG is no used, set LVDS^{CR} or LVD1SCR to H'0000_0040 or H'0000_0000.

2. It is necessary to select the frequency of dot clock within the bus bandwidth.



Notes:

- In this case, the dot clock of DU0/1 is generated by the PLL0/1 in the LVDS0/1-IF. Also, EXTAL is selected as the source clock of the PLL0/1 in the LDVSO/1-IF
- During Dual-Link operation, the PLL1 in the LVDS1-IF can output the dot clock to DU1.

| 2xSingle-Link | | | | Single-Link+RGB output | | | | Dual-Link+RGB output | | | |
|---|-------------|---|-------------|---|-------------|---|-------------|--|-------------|---|-------------|
| Single-Link(LVDS0) | | Single-Link(LVDS1) | | Single-Link(LVDS0) | | The dotclock of DU1 is generated by the PLL1 in the LVDS1-IF. | | Dual-Link(LVDS0/1) | | The dotclock of DU1 is generated by the PLL1 in the LVDS1-IF. | |
| (dotclk: 79.714 MHz sscg:0.026 MHz Down spread -3%) | | (dotclk: 29.571 MHz sscg:0.030 MHz Down spread -3%) | | (dotclk: 79.714 MHz sscg:0.026 MHz Down spread -3%) | | (dotclk: 29.576 MHz sscg:0.029 MHz Down spread -3%) | | (dotclk: 108.571 MHz sscg:0.029 MHz Down spread -3%) | | (dotclk: 29.576 MHz sscg:0.029 MHz Down spread -3%) | |
| LVDPLLCR | H'0046_5AE3 | LVD1PLLCR | H'0046_5E21 | LVDPLLCR | H'0046_5AE3 | LVD1PLLCR | H'0047_01E2 | LVDPLLCR | H'0046_5AF2 | LVD1PLLCR | H'004701E2 |
| LVDS0SCR | H'4D20_0000 | LVD1SCR | H'4960_0000 | LVDS0SCR | H'4D20_0000 | LVD1SCR | H'4160_0000 | LVDS0SCR | H'4160_0000 | LVD1SCR | H'4160_0000 |
| ↓ 1 μs Wait | | ↓ 1 μs Wait | | ↓ 1 μs Wait | | ↓ 1 μs Wait | | ↓ 1 μs Wait | | ↓ 1 μs Wait | |
| LVDS0SCR | H'4D30_0000 | LVD1SCR | H'4970_0000 | LVDS0SCR | H'4D30_0000 | LVD1SCR | H'4170_0000 | LVDS0SCR | H'4170_0000 | LVD1SCR | H'4170_0000 |
| ↓ 200 μs Wait | | ↓ 200 μs Wait | | ↓ 200 μs Wait | | ↓ 200 μs Wait | | ↓ 200 μs Wait | | ↓ 200 μs Wait | |
| LVDDIV | H'0000_0000 | LVD1DIV | H'0000_0181 | LVDDIV | H'0000_0000 | LVD1DIV | H'0000_01A0 | LVDDIV | H'0000_0000 | LVD1DIV | H'0000_01A0 |
| Set DU register | | Set DU register | | Set DU register | | Set DU register | | Set DU register | | Set DU register | |
| LVDCR0 | H'0000_0000 | LVD1CR0 | H'0000_0000 | LVDCR0 | H'0000_0000 | | | LVDCR0 | H'0000_0000 | | |
| LVDCR1 | H'0000_0000 | LVD1CR1 | H'0000_0000 | LVDCR1 | H'0000_0000 | | | LVDCR1 | H'0000_0000 | | |
| LVDCR2 | H'0000_0000 | LVD1CR2 | H'0000_0000 | LVDCR2 | H'0000_0000 | | | LVDCR2 | H'0000_0001 | | |
| | | | | | | | | LVD1CTR0 | H'0000_0000 | | |
| LVDCR3 | H'0000_03FF | LVD1CR3 | H'0000_03FF | LVDCR3 | H'0000_03FF | | | LVD1CHR0 | H'0000_0000 | | |
| | | | | | | | | LVD1STR0 | H'0000_0001 | | |
| LVDCR4 | H'0000_0002 | LVD1CR4 | H'0000_0002 | LVDCR4 | H'0000_0002 | | | | | | |
| | | | | | | | | LVD1CR1 | H'0000_0000 | | |
| LVDCR5 | H'0000_0003 | LVD1CR5 | H'0000_0003 | LVDCR5 | H'0000_0003 | | | LVD1STR1 | H'0000_0001 | | |
| | | | | | | | | | | | |
| | | | | | | | | LVD1CR2 | H'0000_03FF | | |
| | | | | | | | | LVD1CR3 | H'0000_03FF | | |
| | | | | | | | | | | | |
| | | | | | | | | LVD1CR4 | H'0000_0002 | | |
| | | | | | | | | LVD1CR5 | H'0000_0002 | | |
| | | | | | | | | | | | |
| | | | | | | | | LVD1CR6 | H'0000_0003 | | |
| | | | | | | | | | | | |
| | | | | | | | | LVD1CR7 | H'0000_0003 | | |
| | | | | | | | | | | | |

Note *: LVRES and LVEN is set to B'11 at the same time.

Figure 37.9 Example of dual output [RZ/G2E]

37.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The following notes must be observed when using the LVDS-IF module.

Incorrect operation and damage to the device itself may occur if these notes are not followed.

- Since the LVDS-IF module includes logic that operates from the dot clock signal output by the DU module, incorrect operation may occur if the DU registers are not set appropriately. Also, do not change any DU register values during LVDS-IF operation. (If dot clock output is not stable, the LVDS-IF PLL is unlocked, and thus operation cannot be guaranteed.) There are two DU registers related to dot clock: Display unit System Control Register (DSYSR0) and External Synchronization Control Register (ESCR0). These registers must not be changed during LVDS-IF operation. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
- Set the dot clock signal input to the LVDS-IF module to a frequency within the LVDS-IF module's guaranteed operating range (5[RZ/G2E]/ 31[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] to 148.5 MHz).
- Application systems should be implemented in a failsafe manner, such as by connecting the LVDS outputs to a failsafe receiver or by connecting terminators, so that no problems in or damage to the application can occur if the LVDS pins become unstable (for example, the differential outputs going to the same level).

38. HDMI

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

38.1 Overview

38.1.1 Features

HDMI TX supports the following features.

(1) Number of channels

- RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2H and RZ/G2N: 1ch (HDMI0)

(2) Video formats

- CEA-861-E video formats up to 1080p at 60 Hz and 720p/1080i
- Optional HDMI 1.4b video formats
 - CEA-861-E video formats up to 1080p
 - HDMI 1.4b 4K x 2K@30Hz video formats
 - HDMI 1.4b 3D video modes with up to 297 MHz (TMDS clock)

(3) Input Colorimetry

- 24-bit RGB 4:4:4/YCbCr 4:4:4

(4) Color space converter (CSC)

- RGB (4:4:4) to YCbCr (4:4:4 or 4:2:2)
- YCbCr (4:4:4) to RGB (4:4:4) /YCbCr (4:2:2)

(5) HDMI 1.4b supported Infoframes

- Audio InfoFrame packet extension to support LFE playback level information
- AVI infoFrame packet extension to support YCC Quantization range (Limited Range, Full Range)
- AVI infoFrame packet extension to support Content type (Graphics, Photo, Cinema, Game)

(6) Audio formats

- Up to 192 kHz IEC60958 audio sampling rate
- I2S 2 x 4 Ch L-PCM Input

(7) Pixel clock from 25MHz up to 297MHz

(8) I2C DDC, EDID block read mode

(9) Single-channel DVI 1.0 backward compatibility (dual-link DVI not supported)

39. AUDIO

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

39.1 Overview

This section describes the data path between audio modules, list of routings, and transfer flow.

39.1.1 Features

Audio module is a module connecting Audio-DMAC, Audio-DMACpp, SCU, SSIU, and ADG.

Refer to the section of each module for the detailed function.

39.1.2 Block Diagram

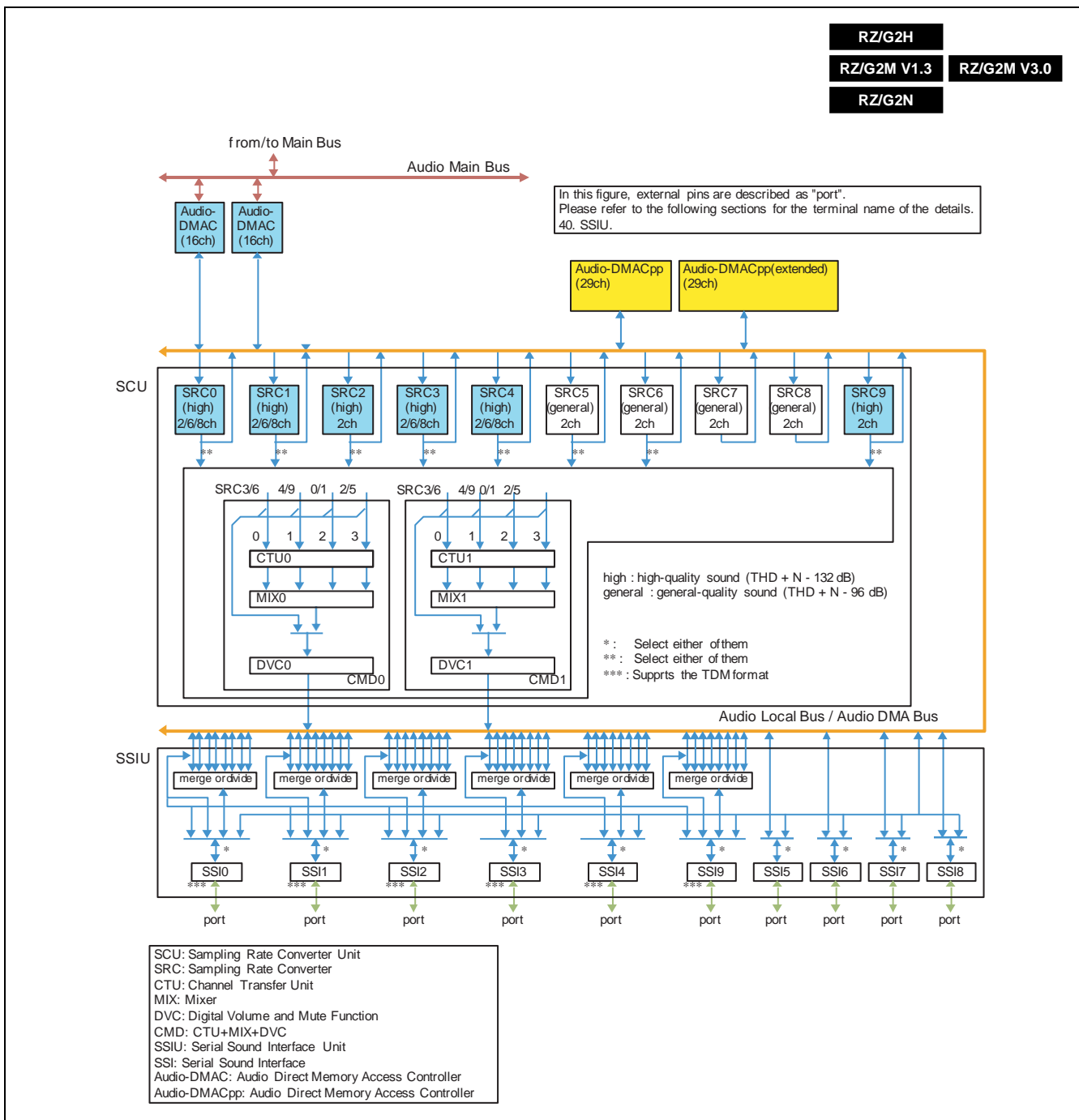


Figure 39.1 Data Paths between Audio Modules [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

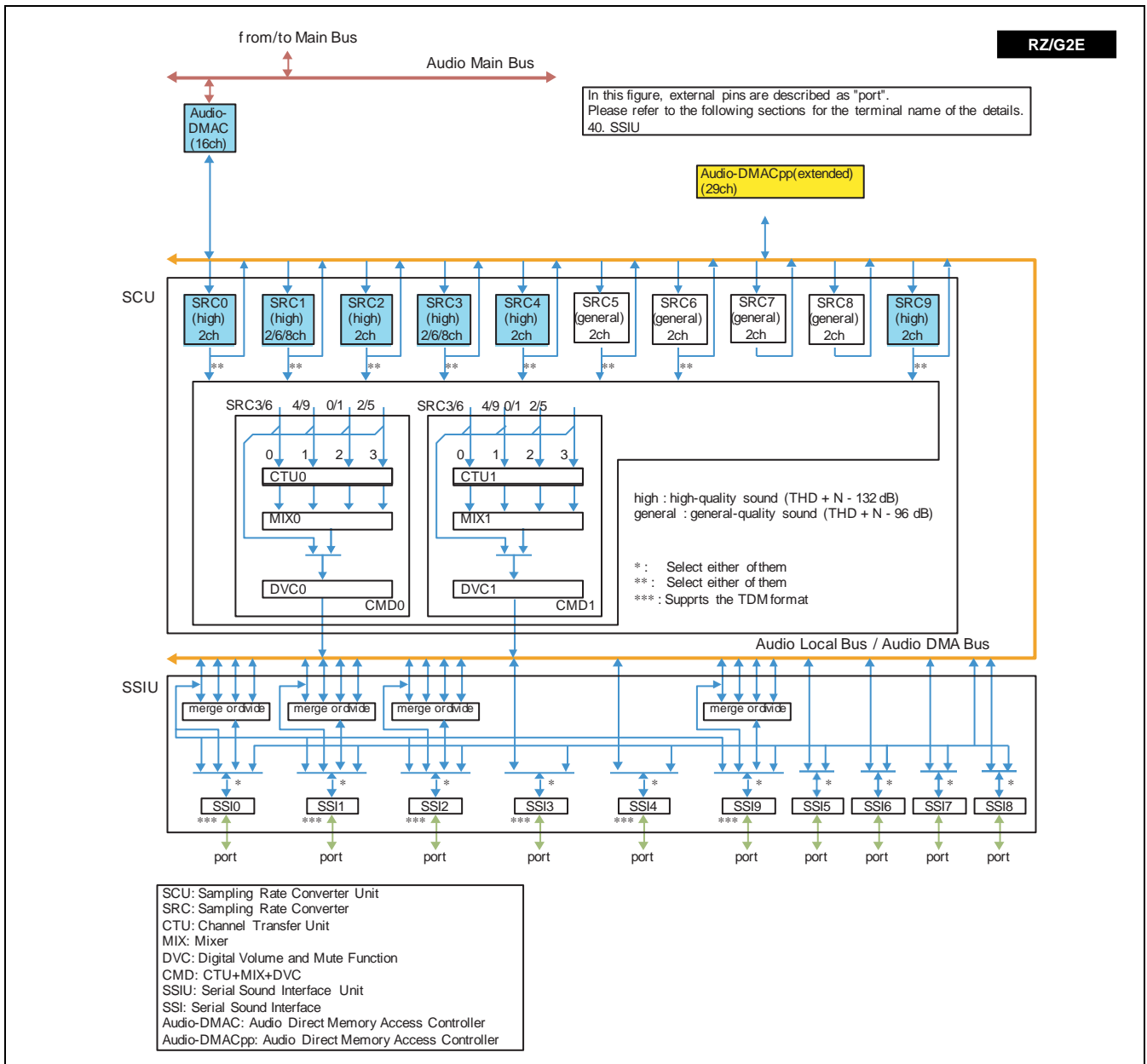


Figure 39.2 Data Paths between Audio Modules [RZ/G2E]

39.1.3 External Pins

Refer to the section 41(SSIU) for more information about terminal information.

39.1.4 Register Configuration

The Audio module does not have a register.

39.1.5 Connected Module

Table 39.1 Connected Modules

| Module name | | Connected module name | Function of connected module |
|-------------|-----------|-----------------------|--|
| Audio | Related | APMU | Access the Register |
| | | PFC | Select External pins |
| | | CPG | Output Clocks |
| | | Module Standby | Control to stop clocks |
| | | Software Reset | Execute software reset |
| | | INTC-AP | Control to interrupt |
| Including | Including | SSIU | Serial Sound Interface Unit |
| | | SSI | Serial Sound Interface |
| | | ADG | Output Clocks for Audio module |
| | | SCU | Sampling Rate Converter Unit |
| | | Audio-DMAC | Control Direct Memory Access for Audio module |
| | | Audio-DMACpp | Control Direct Memory Access for Audio module connected to the audio local bus |

39.2 Register Description

The Audio module does not have a register.

39.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

39.3.1 List of Routings

Data transmission paths in the Audio module are shown in Figure 39.3.

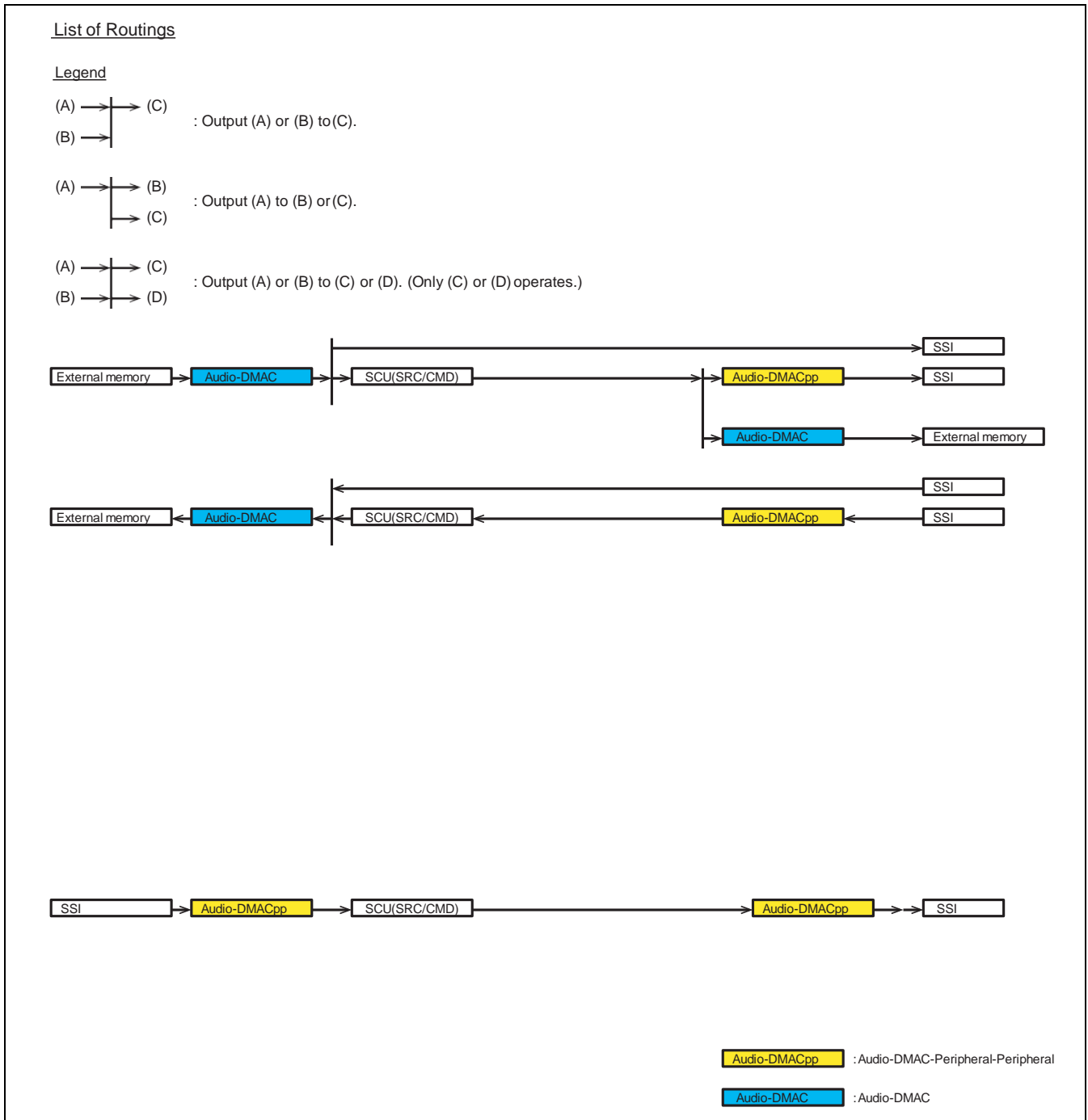


Figure 39.3 List of Routings

39.3.2 Audio-DMAC

The Audio-DMAC is used by data transfer between the external memory and audio modules. The setting values of DMASAR and DMADAR of the Audio-DMAC are shown in Table 39.2.

Table 39.2 Setting Values of DMASAR and DMADAR of Audio-DMAC

| Name of Register | DMASAR/DMADAR | Second Generation RZ/G Series Products | | | |
|------------------|---------------|---|----------------------------|--------|--------|
| | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI0-0_BUSIF | H'EC10 0000 | √ | √ | √ | √ |
| SSI0-1_BUSIF | 0400 | √ | √ | √ | √ |
| SSI0-2_BUSIF | 0800 | √ | √ | √ | √ |
| SSI0-3_BUSIF | 0C00 | √ | √ | √ | √ |
| SSI1-0_BUSIF | 1000 | √ | √ | √ | √ |
| SSI1-1_BUSIF | 1400 | √ | √ | √ | √ |
| SSI1-2_BUSIF | 1800 | √ | √ | √ | √ |
| SSI1-3_BUSIF | 1C00 | √ | √ | √ | √ |
| SSI2-0_BUSIF | 2000 | √ | √ | √ | √ |
| SSI2-1_BUSIF | 2400 | √ | √ | √ | √ |
| SSI2-2_BUSIF | 2800 | √ | √ | √ | √ |
| SSI2-3_BUSIF | 2C00 | √ | √ | √ | √ |
| SSI3-0_BUSIF | 3000 | √ | √ | √ | √ |
| SSI3-1_BUSIF | 3400 | √ | √ | √ | √ |
| SSI3-2_BUSIF | 3800 | √ | √ | √ | √ |
| SSI3-3_BUSIF | 3C00 | √ | √ | √ | √ |
| SSI4-0_BUSIF | 4000 | √ | √ | √ | √ |
| SSI4-1_BUSIF | 4400 | √ | √ | √ | √ |
| SSI4-2_BUSIF | 4800 | √ | √ | √ | √ |
| SSI4-3_BUSIF | 4C00 | √ | √ | √ | √ |
| SSI5_BUSIF | 5000 | √ | √ | √ | √ |
| SSI6_BUSIF | 6000 | √ | √ | √ | √ |
| SSI7_BUSIF | 7000 | √ | √ | √ | √ |
| SSI8_BUSIF | 8000 | √ | √ | √ | √ |
| SSI9-0_BUSIF | 9000 | √ | √ | √ | √ |
| SSI9-1_BUSIF | 9400 | √ | √ | √ | √ |
| SSI9-2_BUSIF | 9800 | √ | √ | √ | √ |
| SSI9-3_BUSIF | 9C00 | √ | √ | √ | √ |
| SSI0-4_BUSIF | A000 | √ | √ | √ | — |
| SSI0-5_BUSIF | A400 | √ | √ | √ | — |
| SSI0-6_BUSIF | A800 | √ | √ | √ | — |
| SSI0-7_BUSIF | AC00 | √ | √ | √ | — |
| SSI1-4_BUSIF | B000 | √ | √ | √ | — |

**Second Generation
RZ/G Series Products**

| Name of Register | DMASAR/DMADAR | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|------------------|---------------|------|--------|----------------------------|--------|--------|
| | | | | | | |
| SSI1-5_BUSIF | H'EC10 | B400 | √ | √ | √ | — |
| SSI1-6_BUSIF | | B800 | √ | √ | √ | — |
| SSI1-7_BUSIF | | BC00 | √ | √ | √ | — |
| SSI2-4_BUSIF | | C000 | √ | √ | √ | — |
| SSI2-5_BUSIF | | C400 | √ | √ | √ | — |
| SSI2-6_BUSIF | | C800 | √ | √ | √ | — |
| SSI2-7_BUSIF | | CC00 | √ | √ | √ | — |
| SSI3-4_BUSIF | | D000 | √ | √ | √ | — |
| SSI3-5_BUSIF | | D400 | √ | √ | √ | — |
| SSI3-6_BUSIF | | D800 | √ | √ | √ | — |
| SSI3-7_BUSIF | | DC00 | √ | √ | √ | — |
| SSI4-4_BUSIF | | E000 | √ | √ | √ | — |
| SSI4-5_BUSIF | | E400 | √ | √ | √ | — |
| SSI4-6_BUSIF | | E800 | √ | √ | √ | — |
| SSI4-7_BUSIF | | EC00 | √ | √ | √ | — |
| SSI9-4_BUSIF | | F000 | √ | √ | √ | — |
| SSI9-5_BUSIF | | F400 | √ | √ | √ | — |
| SSI9-6_BUSIF | | F800 | √ | √ | √ | — |
| SSI9-7_BUSIF | | FC00 | √ | √ | √ | — |
| SSITDR0*2 | H'EC24 | 1008 | √ | √ | √ | √ |
| SSIRDR0*1 | | 100C | √ | √ | √ | √ |
| SSITDR1*2 | | 1048 | √ | √ | √ | √ |
| SSIRDR1*1 | | 104C | √ | √ | √ | √ |
| SSITDR2*2 | | 1088 | √ | √ | √ | √ |
| SSIRDR2*1 | | 108C | √ | √ | √ | √ |
| SSITDR3*2 | | 10C8 | √ | √ | √ | √ |
| SSIRDR3*1 | | 10CC | √ | √ | √ | √ |
| SSITDR4*2 | | 1108 | √ | √ | √ | √ |
| SSIRDR4*1 | | 110C | √ | √ | √ | √ |
| SSITDR5*2 | | 1148 | √ | √ | √ | √ |
| SSIRDR5*1 | | 114C | √ | √ | √ | √ |
| SSITDR6*2 | | 1188 | √ | √ | √ | √ |
| SSIRDR6*1 | | 118C | √ | √ | √ | √ |
| SSITDR7*2 | | 11C8 | √ | √ | √ | √ |
| SSIRDR7*1 | | 11CC | √ | √ | √ | √ |
| SSITDR8*2 | | 1208 | √ | √ | √ | √ |
| SSIRDR8*1 | | 120C | √ | √ | √ | √ |
| SSITDR9*2 | | 1248 | √ | √ | √ | √ |
| SSIRDR9*1 | | 124C | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Name of Register | DMASAR/DMADAR | | Second Generation RZ/G Series Products | | | |
|------------------|---------------|------|---|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SRC0in_BUSIF*2 | H'EC00 | 0000 | √ | √ | √ | √ |
| SRC1in_BUSIF*2 | | 0400 | √ | √ | √ | √ |
| SRC2in_BUSIF*2 | | 0800 | √ | √ | √ | √ |
| SRC3in_BUSIF*2 | | 0C00 | √ | √ | √ | √ |
| SRC4in_BUSIF*2 | | 1000 | √ | √ | √ | √ |
| SRC5in_BUSIF*2 | | 1400 | √ | √ | √ | √ |
| SRC6in_BUSIF*2 | | 1800 | √ | √ | √ | √ |
| SRC7in_BUSIF*2 | | 1C00 | √ | √ | √ | √ |
| SRC8in_BUSIF*2 | | 2000 | √ | √ | √ | √ |
| SRC9in_BUSIF*2 | | 2400 | √ | √ | √ | √ |
| SRC0out_BUSIF*1 | | 4000 | √ | √ | √ | √ |
| SRC1out_BUSIF*1 | | 4400 | √ | √ | √ | √ |
| SRC2out_BUSIF*1 | | 4800 | √ | √ | √ | √ |
| SRC3out_BUSIF*1 | | 4C00 | √ | √ | √ | √ |
| SRC4out_BUSIF*1 | | 5000 | √ | √ | √ | √ |
| SRC5out_BUSIF*1 | | 5400 | √ | √ | √ | √ |
| SRC6out_BUSIF*1 | | 5800 | √ | √ | √ | √ |
| SRC7out_BUSIF*1 | | 5C00 | √ | √ | √ | √ |
| SRC8out_BUSIF*1 | | 6000 | √ | √ | √ | √ |
| SRC9out_BUSIF*1 | | 6400 | √ | √ | √ | √ |
| CMD0out_BUSIF*1 | | 8000 | √ | √ | √ | √ |
| CMD1out_BUSIF*1 | | 8400 | √ | √ | √ | √ |

Notes: 1. Used only by DMASAR.
2. Used only by DMADAR.

39.3.3 Data Format on Audio Local Bus

Figure 39.4 shows the data formats handled in the audio local bus. When writing or reading data through the Audio-DMAC, align data to match the appropriate data format from those in Figure 39.4.

| | |
|--|--|
| 24-bit stereo data or multichannel data | |
| 17- to 23-bit stereo data or multichannel data | |
| 16-bit stereo data or multichannel data | |
| 9- to 15-bit stereo data or multichannel data | |
| 8-bit stereo data | |
| 16-bit monaural data | |
| 8-bit monaural data | |

Figure 39.4 Data Format

- Notes: 1 Write 0 to the * bits in the figure when writing data. When reading data, ignore the values read from these bits.
2. In the 8-bit stereo data format, (Lch/1) and (Rch/1) indicate the data pair to be processed first and (Lch/2) and (Rch/2) indicate the next data pair to be processed.
3. Only the MSB-first data formats can be used.
4. In Data Format of "16-bit stereo data or multichannel data" and "9- to 15-bit stereo data or multichannel data", set SSIU.SSI_MODE0.ind_word_swap>(* = 0-9) in the setup of independent SSI transmission (SSIU.SSI_MODE0.ind(* = 0-9) = 1).

39.3.4 Rearranging the Order of Data

For the audio modules (SSI, SCU), rearranging the order of data is possible in the channel unit. Places of data in each data format are defined as in Table 39.3 to 39.6.

Table 39.3 Definition of Data Places in Each Data Format (1)




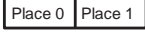
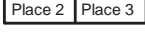
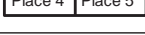

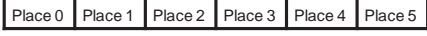

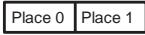
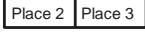
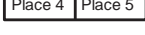
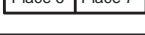

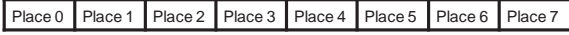
| | |
|-------------------------------------|--|
| SSI · Stereo (2 channels) | SSI_WS01239  SSI_SDATA0  |
| SSI · Stereo x 3 (6 channels) | SSI_WS01239  SSI_SDATA0  SSI_SDATA1  SSI_SDATA2  |
| SSI · TDM (6 channels) | SSI_WS01239  SSI_SDATA0  |
| SSI · Stereo x 4 (8 channels) | SSI_WS01239  SSI_SDATA0  SSI_SDATA1  SSI_SDATA2  SSI_SDATA9  |
| SSI · TDM (8 channels) | SSI_WS01239  SSI_SDATA0  |


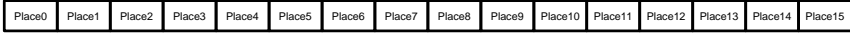
Table 39.4 Definition of Data Places in Each Data Format (4)

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--|--|---------|---------|---------|------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|------|---------|-----|------|---------|---|------|---------|---|------|---------|---|-----|-----|---|
| <p>BUSIF · Stereo (2 channels) · 24 bits</p> | <p>External memory image</p> <table style="margin-left: 20px;"> <tr> <td></td> <td style="text-align: right;">31</td> <td style="text-align: right;">0</td> </tr> <tr> <td>H'00</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'04</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'08</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'0C</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> </table> | | 31 | 0 | H'00 | Place 0 | * | H'04 | Place 1 | * | H'08 | Place 0 | * | H'0C | Place 1 | * | ... | ... | * | | | | | | | | | | | | |
| | 31 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 1 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 0 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 1 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF · Stereo (2 channels) · 16 bits</p> | <p>External memory image</p> <table style="margin-left: 20px;"> <tr> <td></td> <td style="text-align: right;">31</td> <td style="text-align: right;">0</td> </tr> <tr> <td>H'00</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> </tr> <tr> <td>H'04</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> </tr> <tr> <td>H'08</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> </tr> <tr> <td>H'0C</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> </tr> <tr> <td>...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;">...</td> </tr> </table> | | 31 | 0 | H'00 | Place 0 | Place 1 | H'04 | Place 0 | Place 1 | H'08 | Place 0 | Place 1 | H'0C | Place 0 | Place 1 | ... | ... | ... | | | | | | | | | | | | |
| | 31 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF · Stereo (2 channels) · 8 bits</p> | <p>External memory image</p> <table style="margin-left: 20px;"> <tr> <td></td> <td style="text-align: right;">31</td> <td style="text-align: right;">0</td> </tr> <tr> <td>H'00</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> </tr> <tr> <td>H'04</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> </tr> <tr> <td>...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;">...</td> </tr> </table> | | 31 | 0 | H'00 | Place 1 | Place 0 | Place 1 | Place 0 | H'04 | Place 1 | Place 0 | Place 1 | Place 0 | ... | ... | ... | ... | ... | | | | | | | | | | | | |
| | 31 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 1 | Place 0 | Place 1 | Place 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 1 | Place 0 | Place 1 | Place 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | ... | ... | ... | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF · Monaural (1 channel) · 8/16 bits</p> | <p>External memory image</p> <table style="margin-left: 20px;"> <tr> <td></td> <td style="text-align: right;">31</td> <td style="text-align: right;">0</td> </tr> <tr> <td>H'00</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'04</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'08</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'0C</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> </table> | | 31 | 0 | H'00 | Place 0 | * | H'04 | Place 0 | * | H'08 | Place 0 | * | H'0C | Place 0 | * | ... | ... | * | | | | | | | | | | | | |
| | 31 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 0 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 0 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 0 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF · Multichannel (4 channels) · 24 bits</p> | <p>External memory image</p> <table style="margin-left: 20px;"> <tr> <td></td> <td style="text-align: right;">31</td> <td style="text-align: right;">0</td> </tr> <tr> <td>H'00</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'04</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'08</td> <td style="border: 1px solid black; padding: 2px;">Place 2</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'0C</td> <td style="border: 1px solid black; padding: 2px;">Place 3</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'10</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'14</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'18</td> <td style="border: 1px solid black; padding: 2px;">Place 2</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'1C</td> <td style="border: 1px solid black; padding: 2px;">Place 3</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> </table> | | 31 | 0 | H'00 | Place 0 | * | H'04 | Place 1 | * | H'08 | Place 2 | * | H'0C | Place 3 | * | H'10 | Place 0 | * | H'14 | Place 1 | * | H'18 | Place 2 | * | H'1C | Place 3 | * | ... | ... | * |
| | 31 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 1 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 2 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 3 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'10 | Place 0 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'14 | Place 1 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'18 | Place 2 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'1C | Place 3 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF · Multichannel (4 channels) · 16 bits</p> | <p>External memory image</p> <table style="margin-left: 20px;"> <tr> <td></td> <td style="text-align: right;">31</td> <td style="text-align: right;">0</td> </tr> <tr> <td>H'00</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> </tr> <tr> <td>H'04</td> <td style="border: 1px solid black; padding: 2px;">Place 2</td> <td style="border: 1px solid black; padding: 2px;">Place 3</td> </tr> <tr> <td>H'08</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> </tr> <tr> <td>H'0C</td> <td style="border: 1px solid black; padding: 2px;">Place 2</td> <td style="border: 1px solid black; padding: 2px;">Place 3</td> </tr> <tr> <td>...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;">...</td> </tr> </table> | | 31 | 0 | H'00 | Place 0 | Place 1 | H'04 | Place 2 | Place 3 | H'08 | Place 0 | Place 1 | H'0C | Place 2 | Place 3 | ... | ... | ... | | | | | | | | | | | | |
| | 31 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 2 | Place 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 2 | Place 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Table 39.5 Definition of Data Places in Each Data Format (5)

| | | | | | | | | | | | | | | | | | | | | | | | |
|--|--|----|---|------|-----------------|------|-----------------|------|-----------------|------|-----------------|------|-----------------|------|-----------|------|-----------|------|-----------|------|-----------|-----|-------|
| <p>BUSIF · Multichannel (6 channels) · 24 bits</p> | <p>External memory image</p> <table border="1" style="margin-left: 40px;"> <tr> <td style="text-align: right;">31</td> <td style="text-align: right;">0</td> </tr> <tr> <td>H'00</td> <td>Place 0 *</td> </tr> <tr> <td>H'04</td> <td>Place 1 *</td> </tr> <tr> <td>H'08</td> <td>Place 2 *</td> </tr> <tr> <td>H'0C</td> <td>Place 3 *</td> </tr> <tr> <td>H'10</td> <td>Place 4 *</td> </tr> <tr> <td>H'14</td> <td>Place 5 *</td> </tr> <tr> <td>H'18</td> <td>Place 0 *</td> </tr> <tr> <td>H'1C</td> <td>Place 1 *</td> </tr> <tr> <td>...</td> <td>... *</td> </tr> </table> | 31 | 0 | H'00 | Place 0 * | H'04 | Place 1 * | H'08 | Place 2 * | H'0C | Place 3 * | H'10 | Place 4 * | H'14 | Place 5 * | H'18 | Place 0 * | H'1C | Place 1 * | ... | ... * | | |
| 31 | 0 | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 * | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 1 * | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 2 * | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 3 * | | | | | | | | | | | | | | | | | | | | | | |
| H'10 | Place 4 * | | | | | | | | | | | | | | | | | | | | | | |
| H'14 | Place 5 * | | | | | | | | | | | | | | | | | | | | | | |
| H'18 | Place 0 * | | | | | | | | | | | | | | | | | | | | | | |
| H'1C | Place 1 * | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... * | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF · Multichannel (6 channels) · 16 bits</p> | <p>External memory image</p> <table border="1" style="margin-left: 40px;"> <tr> <td style="text-align: right;">31</td> <td style="text-align: right;">0</td> </tr> <tr> <td>H'00</td> <td>Place 0 Place 1</td> </tr> <tr> <td>H'04</td> <td>Place 2 Place 3</td> </tr> <tr> <td>H'08</td> <td>Place 4 Place 5</td> </tr> <tr> <td>H'0C</td> <td>Place 0 Place 1</td> </tr> <tr> <td>...</td> <td>... ..</td> </tr> </table> | 31 | 0 | H'00 | Place 0 Place 1 | H'04 | Place 2 Place 3 | H'08 | Place 4 Place 5 | H'0C | Place 0 Place 1 | ... | | | | | | | | | | | |
| 31 | 0 | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 Place 1 | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 2 Place 3 | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 4 Place 5 | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 0 Place 1 | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF · Multichannel (8 channels) · 24 bits</p> | <p>External memory image</p> <table border="1" style="margin-left: 40px;"> <tr> <td style="text-align: right;">31</td> <td style="text-align: right;">0</td> </tr> <tr> <td>H'00</td> <td>Place 0 *</td> </tr> <tr> <td>H'04</td> <td>Place 1 *</td> </tr> <tr> <td>H'08</td> <td>Place 2 *</td> </tr> <tr> <td>H'0C</td> <td>Place 3 *</td> </tr> <tr> <td>H'10</td> <td>Place 4 *</td> </tr> <tr> <td>H'14</td> <td>Place 5 *</td> </tr> <tr> <td>H'18</td> <td>Place 6 *</td> </tr> <tr> <td>H'1C</td> <td>Place 7 *</td> </tr> <tr> <td>H'20</td> <td>Place 0 *</td> </tr> <tr> <td>...</td> <td>... *</td> </tr> </table> | 31 | 0 | H'00 | Place 0 * | H'04 | Place 1 * | H'08 | Place 2 * | H'0C | Place 3 * | H'10 | Place 4 * | H'14 | Place 5 * | H'18 | Place 6 * | H'1C | Place 7 * | H'20 | Place 0 * | ... | ... * |
| 31 | 0 | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 * | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 1 * | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 2 * | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 3 * | | | | | | | | | | | | | | | | | | | | | | |
| H'10 | Place 4 * | | | | | | | | | | | | | | | | | | | | | | |
| H'14 | Place 5 * | | | | | | | | | | | | | | | | | | | | | | |
| H'18 | Place 6 * | | | | | | | | | | | | | | | | | | | | | | |
| H'1C | Place 7 * | | | | | | | | | | | | | | | | | | | | | | |
| H'20 | Place 0 * | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... * | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF · Multichannel (8 channels) · 16 bits</p> | <p>External memory image</p> <table border="1" style="margin-left: 40px;"> <tr> <td style="text-align: right;">31</td> <td style="text-align: right;">0</td> </tr> <tr> <td>H'00</td> <td>Place 0 Place 1</td> </tr> <tr> <td>H'04</td> <td>Place 2 Place 3</td> </tr> <tr> <td>H'08</td> <td>Place 4 Place 5</td> </tr> <tr> <td>H'0C</td> <td>Place 6 Place 7</td> </tr> <tr> <td>H'10</td> <td>Place 0 Place 1</td> </tr> <tr> <td>...</td> <td>... ..</td> </tr> </table> | 31 | 0 | H'00 | Place 0 Place 1 | H'04 | Place 2 Place 3 | H'08 | Place 4 Place 5 | H'0C | Place 6 Place 7 | H'10 | Place 0 Place 1 | ... | | | | | | | | | |
| 31 | 0 | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 Place 1 | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 2 Place 3 | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 4 Place 5 | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 6 Place 7 | | | | | | | | | | | | | | | | | | | | | | |
| H'10 | Place 0 Place 1 | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | |

Table 39.6 Definition of Data Places in Each Data Format (6) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---------|--------|--------|------|--------|--------|------|--------|--------|------|--------|--------|------|--------|--------|------|---------|---------|------|---------|---------|------|---------|---------|------|--------|--------|-----|-----|-----|
| SSI • TDM (16 channels) | SSL_WS01239  SSL_SDATA0  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| BUSIF • Multichannel (16 channels) • 16 bits | External memory image <div style="text-align: center;"> 31 0 </div> <table border="1" style="margin-left: 20px;"> <tr> <td>H'00</td> <td>Place0</td> <td>Place1</td> </tr> <tr> <td>H'04</td> <td>Place2</td> <td>Place3</td> </tr> <tr> <td>H'08</td> <td>Place4</td> <td>Place5</td> </tr> <tr> <td>H'0C</td> <td>Place6</td> <td>Place7</td> </tr> <tr> <td>H'10</td> <td>Place8</td> <td>Place9</td> </tr> <tr> <td>H'14</td> <td>Place10</td> <td>Place11</td> </tr> <tr> <td>H'18</td> <td>Place12</td> <td>Place13</td> </tr> <tr> <td>H'1C</td> <td>Place14</td> <td>Place15</td> </tr> <tr> <td>H'20</td> <td>Place0</td> <td>Place1</td> </tr> <tr> <td>...</td> <td>...</td> <td>...</td> </tr> </table> | H'00 | Place0 | Place1 | H'04 | Place2 | Place3 | H'08 | Place4 | Place5 | H'0C | Place6 | Place7 | H'10 | Place8 | Place9 | H'14 | Place10 | Place11 | H'18 | Place12 | Place13 | H'1C | Place14 | Place15 | H'20 | Place0 | Place1 | ... | ... | ... |
| H'00 | Place0 | Place1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place2 | Place3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place4 | Place5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place6 | Place7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'10 | Place8 | Place9 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'14 | Place10 | Place11 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'18 | Place12 | Place13 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'1C | Place14 | Place15 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'20 | Place0 | Place1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

39.3.5 Transfer Flow

When data transfer is performed between the audio modules, use the Audio-DMAC and Audio-DMACpp to perform settings in accord with the transfer flow shown in Figure 39.5.

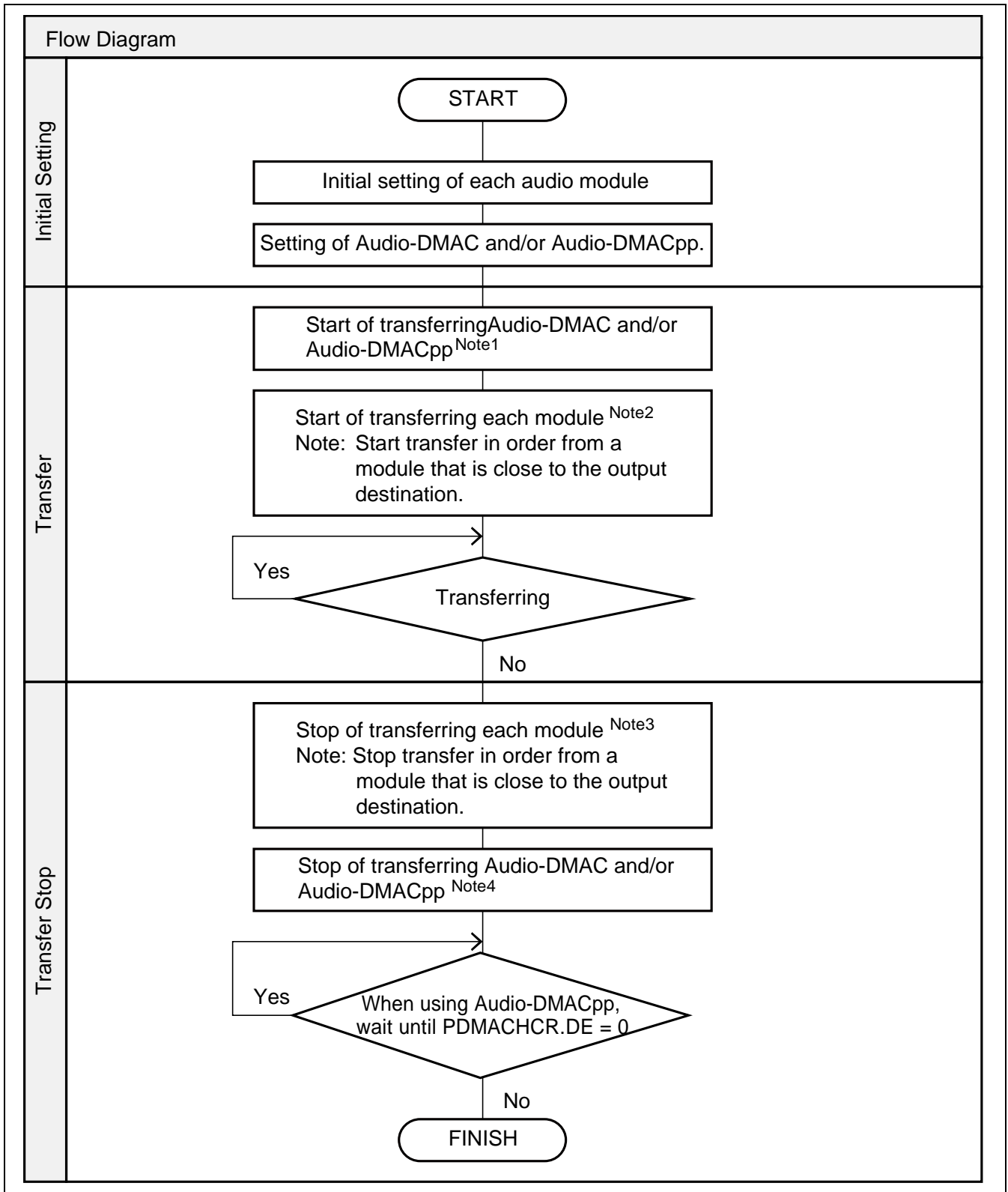


Figure 39.5 Transfer Flow

Notes: 1 For Audio-DMAC, set DMACHCR.DE.

For Audio-DMACpp, set PDMACHCR.DE.

Set up registers except the above in “setting of Audio-DMAC and/or Audio-DMACpp.”

2. For SSI (except for the case when TDM split mode, SSI independent transfer, SSI0, SSI1, SSI2, or SSI9, and SSI3 or SSI4 are used at the same time), when transmitting (from SSI to the external device), set SSIq/r_CONTROL.start after setting SSICR.en/dmen. When receiving (from the external device to SSI), set SSICR.en/dmen after setting SSIq/r_CONTROL.start.

For SSI (In the case of TDM split mode) regardless of transmission and reception, set SSIq/r_CONTROL.start after setting SSICR.en/dmen.

For SSI (In the case of SSI independent transfer), set SSICR.en/dmen.

For SSI (When using SSI0, SSI1, SSI2, or SSI9 and SSI3 or SSI4 at the same time), when transmitting (from SSI to the external device) set SSIq/r_CONTROL.start after setting SSI_CONTROL. When receiving (from the external device to SSI), set SSI_CONTROL after setting SSIq/r_CONTROL.start. (Set SSICR.dmen at the time of the initial setting.)

For SCU, Set SRCm_CONTROL and CMDn_CONTROL.

Set up registers except the above in “initial setting of each audio module”.

3. For SSI (except for the case when TDM split mode, SSI independent transfer, SSI0, SSI1, SSI2, or SSI9, and SSI3 or SSI4 are used at the same time), when transmitting (from SSI to the external device), clear SSIq/r_CONTROL.start after clearing SSICR.en/dmen. When receiving (from the external device to SSI), clear SSICR.en/dmen after clearing SSIq/r_CONTROL.start.

For SSI (In the case of TDM split mode), regardless of transmission and reception, clear all SSIq/r_CONTROL.start of the relevant channels to clear SSICR.en/dmen.

For SSI (In the case of SSI independent transfer), clear SSICR.en/dmen.

For SSI (When using SSI0, SSI1, SSI2, or SSI9 and SSI3 or SSI4 at the same time), when transmitting (from SSI to the external device) clear SSIq/r_CONTROL.start after clearing SSICR.dmen of each module and SSI_CONTROL. When receiving (from the external device to SSI), clear SSICR.dmen of each module and SSI_CONTROL after clearing SSIq/r_CONTROL.start.

For SCU, clear SRCm_CONTROL and CMDn_CONTROL.

Because transfer is stopped in order from a module that is close to the output destination, overflow may occur in a preceding module. Be careful.

4. For Audio-DMACpp, clear PDMACHCR.DE. At this time, Other bits must not change.
For Audio-DMAC, clear DMACHCR.DE. Clear processing of Audio-DMAC can be cleared anywhere in a TransferStop sequence.

39.3.6 Module Standby Function

For the register settings to make a transition to or from module standby mode, refer to section 8A, Module Standby and Software Reset.

(1) Transition to Module Standby Mode

To make a transition to module standby mode by the module standby function, refer to the following transition procedure for each module.

(a) SSIU/SSI

1. Check the following registers' settings.

[SSIU]

- All bits in the SSIq control register (SSIq_CONTROL) (q = 0-0, 1-0, 2-0, 3-0, 4-0, or 9-0) are cleared to 0 (data transfer stopped).
- All bits in the SSIr control register (SSIr_CONTROL) (r = 5 to 8) are cleared to 0 (data transfer stopped).
- All bits in the SSIq interrupt enable register (SSIq_INT_ENABLE_MAIN) (q = 0-0, 1-0, 2-0, 3-0, 4-0, or 9-0) are cleared to 0 (interrupt disabled).
- All bits in the SSIr interrupt enable register (SSIr_INT_ENABLE_MAIN) (r = 5 to 8) are cleared to 0 (interrupt disabled).
- All bits in the SSI control register (SSI_CONTROL) are cleared to 0 (data transfer stopped)

[SSI]

- The DMEN and EN bits in the control register (SSICRn) (n = 0 to 9) are cleared to 0 (disabled).
- The IDST bit in the status register (SSISRn) (n = 0 to 9) is set to 1.
- The CONT bit in the WS mode register (SSIWSRn) (n = 0 to 9) is cleared to 0 (disabled).

2. Set the MSTP1005 to MSTP1015 bits in the module stop control register (RMSTPCR10/SMSTPCR10)

Note: The above bits should be set while the module operation has been completed and is placed in the idle state in which the module cannot be activated by external pins or other modules.

(b) SCU

1. Check the following registers' settings.
 - All bits in the SRCm control register (SRCm_CONTROL) (m = 0 to 9) are cleared to 0 (data transfer stopped).
 - All bits in the CMDn control register (CMDn_CONTROL) (n = 0, 1) are cleared to 0 (data transfer stopped).
 - All bits in the SRCm interrupt enable register 0 (SRCm_INT_ENABLE0) (m = 0 to 6, 9) are cleared to 0 (interrupt disabled).
 - All bits in the SRCn interrupt enable register 0 (SRCn_INT_ENABLE0) (n = 7, 8) are cleared to 0 (interrupt disabled).
2. Set the MSTP1017 to MSTP1031 bits in the module stop control register (RMSTPCR10/SMSTPCR10)

Note: The above bits should be set while the module operation has been completed and is placed in the idle state in which the module cannot be activated by external pins or other modules.

(c) ADG

1. Set the MSTP922 bit in the module stop control register (RMSTPCR9/SMSTPCR9) to 1.

Note: The above bit is for clock divider(brgclk/avb_counter8).

The above bit should be set while the module cannot be activated by other modules

(2) Releasing and Restarting from Module Standby Mode

After the transition to module standby mode, modules can be released from module standby mode by a power-on reset or the appropriate procedure from the list below.

(a) SSIU/SSI

1. Supply the clock signal to SSIU/SSI modules.
2. Clear the MSTP1005 to MSTP1015 bits in the module stop control register (RMSTPCR10/SMSTPCR10).

(b) SCU

1. Clear the MSTP1017 to MSTP1031 bits in the module stop control register (RMSTPCR10/SMSTPCR10).

(c) ADG

1. Supply the clock signal to ADG modules.
2. Clear the MSTP922 bit in the module stop control register (RMSTPCR9/SMSTPCR9).

39.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

39.4.1 Notes on Route Switching and Setting Changes

When switching the transfer route, stop the current transfer, specify the new transfer route, and then start transfer. Also, when changing the quantization bit count or the formats, once stop transfer, and then start transfer again after resetting. If this procedure is not followed, correct operation is not guaranteed.

39.4.2 Use of SSI3 and SSI4

Table 39.7 shows the combinations of SSI3 and SSI4 usage and register settings.

Table 39.7 Combinations of SSI3 and SSI4 Usage and Register Settings

| | SSI_MODE1 Register Settings | | Bit Used to Control Start or Stop Operation |
|---|-----------------------------|----------|---|
| | ssi34_sync | ssi4_pin | |
| Operating SSI3 and SSI4 independently | 0 | 00 | SSICRn.EN (n = 3 and 4) |
| Operating SSI4 with the SCK and WS signals of SSI3 being shared by both SSI3 and SSI4 | 0 | 01 or 10 | SSICRn.EN (n = 3 and 4) |
| Synchronizing serial data of SSI3 and SSI4 | 1 | 01 or 10 | SSI_CONTROL.ssi34 |

39.4.3 Notes on CMD Block Usage

When a single sound source is split into two routes, CMD0 and CMD1, and output to SSI3 and SSI4, use the settings for "Synchronizing serial data of SSI3 and SSI4" in Table 39.7, Combinations of SSI3 and SSI4 Usage and Register Settings. The quantization bit count and sampling frequency settings should be the same between SSI3 and SSI4. Note that only the stereo format can be specified for SSI3 and SSI4. For example, it is not possible to output TDM-format data from SSI3 and stereo-format data from SSI4. The reverse combination is not possible either (stereo-format data from SSI3 or TDM-format data from SSI4).

When a single sound source is split into two routes, CMD0 and CMD1, the sampling frequency settings should be the same.

When a single sound source is split into two routes, CMD0 and CMD1, transfer to different types of destination is impossible. For example, outputting CMD0 to SSI is not allowed.

39.4.4 Note when SSI0, SSI1, and SSI2, or SSI0, SSI1, SSI2, and SSI9 are Used for Single-Source Audio Data

When SSI0, SSI1, and SSI2, or SSI0, SSI1, SSI2, and SSI9 are used for single-source audio data, specifying eight-bit quantization and six- or eight-channel operation at the same time is not possible.

39.4.5 Note on Transfer

When an underflow or an overflow occurs in any audio module, stop the transfer.

39.4.6 Note on Usage of the SCU Module

Before starting data transfer, the SCU module needs certain "input data timing" and "output data timing". When "input data timing" and "output data timing" are not inputted before the transmission start, it does not operate correctly.

In addition, if the SSI module operates in master mode, and if the WS signal is used as "input/output timing signal for the SRC", set SSIWSRn.CONT to 1.

(The WS signal inputted into the ADG module is the same signal as the SSI_WS terminal.)

39.4.7 Note on Usage of the SSIU Module [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

When an SSIU module is used in "TDM-16ch mode", only the transfer with the external memory is available.

39.4.8 Note on software reset during data transfer

If software reset (Software Reset.SRCR) is performed during data transfer, abnormal operation may occur. At this time, normal values can not be guaranteed for the transferred data.

40. Serial Sound Interface Unit (SSIU)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

40.1 Overview

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

The SSIU, incorporating ten SSI modules, allows independent operation of SSI modules, operation of multiple SSI modules sharing the same serial clock, and connection or split of multichannel data. Refer to section 41 to understand single SSI module in detail.

Note: Configure the system so that connected external devices can operate on the same clock source.

40.1.1 Features

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

- Incorporates ten SSI modules.
- Supports 6 channels/1 sound source with three SSI modules (SSI0, SSI1, and SSI2).
- Supports 8 channels/1 sound source with four SSI modules (SSI0, SSI1, SSI2, and SSI9).
- Operation of multiple modules on the same serial clock (SSI0/SSI1/SSI2/SSI3/SSI9, SSI3/SSI4/SSI9, or SSI7/SSI8).
- TDM format (Basic Configuration) corresponds to 4, 6 or 8-channel data.
- TDM format corresponds to 16-channel data (TDM-16ch mode).
- Handles 8-channel data on the serial bus and 6-channel data in the LSI (TDM extend mode).
- Handles 6-channel data on the serial bus and 8-channel data in the LSI (TDM extend mode).
- Connects monaural or stereo data to output TDM format data (TDM split mode).
- Splits TDM format input data into monaural or stereo data (TDM split mode).
- Connects stereo or multi data to output TDM format data (TDM ex-split mode).
- Splits TDM format input data into stereo or multi data (TDM ex-split mode).
- The frequency range of SCK signal is from 297.3 kHz to 12.5MHz at master mode, and from 297.3 kHz to 15.1 MHz at slave mode.
- Selects SSI signals that is transmitted to the 1-channel of HDMI-IF (at master transmitter mode only). [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

[RZ/G2E]

- Incorporates ten SSI modules.
- Supports 6 channels/1 sound source with three SSI modules (SSI0, SSI1, and SSI2).
- Supports 8 channels/1 sound source with four SSI modules (SSI0, SSI1, SSI2, and SSI9).
- Operation of multiple modules on the same serial clock (SSI0/SSI1/SSI2/SSI3/SSI9, SSI3/SSI4/SSI9, or SSI7/SSI8).
- TDM format (Basic Configuration) corresponds to 4, 6 or 8-channel data.
- Handles 8-channel data on the serial bus and 6-channel data in the LSI (TDM extend mode).
- Handles 6-channel data on the serial bus and 8-channel data in the LSI (TDM extend mode).
- Connects monaural or stereo data to output TDM format data (TDM split mode).
- Splits TDM format input data into monaural or stereo data (TDM split mode).
- Connects stereo or multi data to output TDM format data (TDM ex-split mode).
- Splits TDM format input data into stereo or multi data (TDM ex-split mode).

- The frequency range of SCK signal is from 297.3 kHz to 12.5MHz at master mode, and from 297.3 kHz to 15.1 MHz at slave mode.

40.1.2 Block Diagram

Figure 40.1 to Figure 40.4 show the SSIU block diagrams.

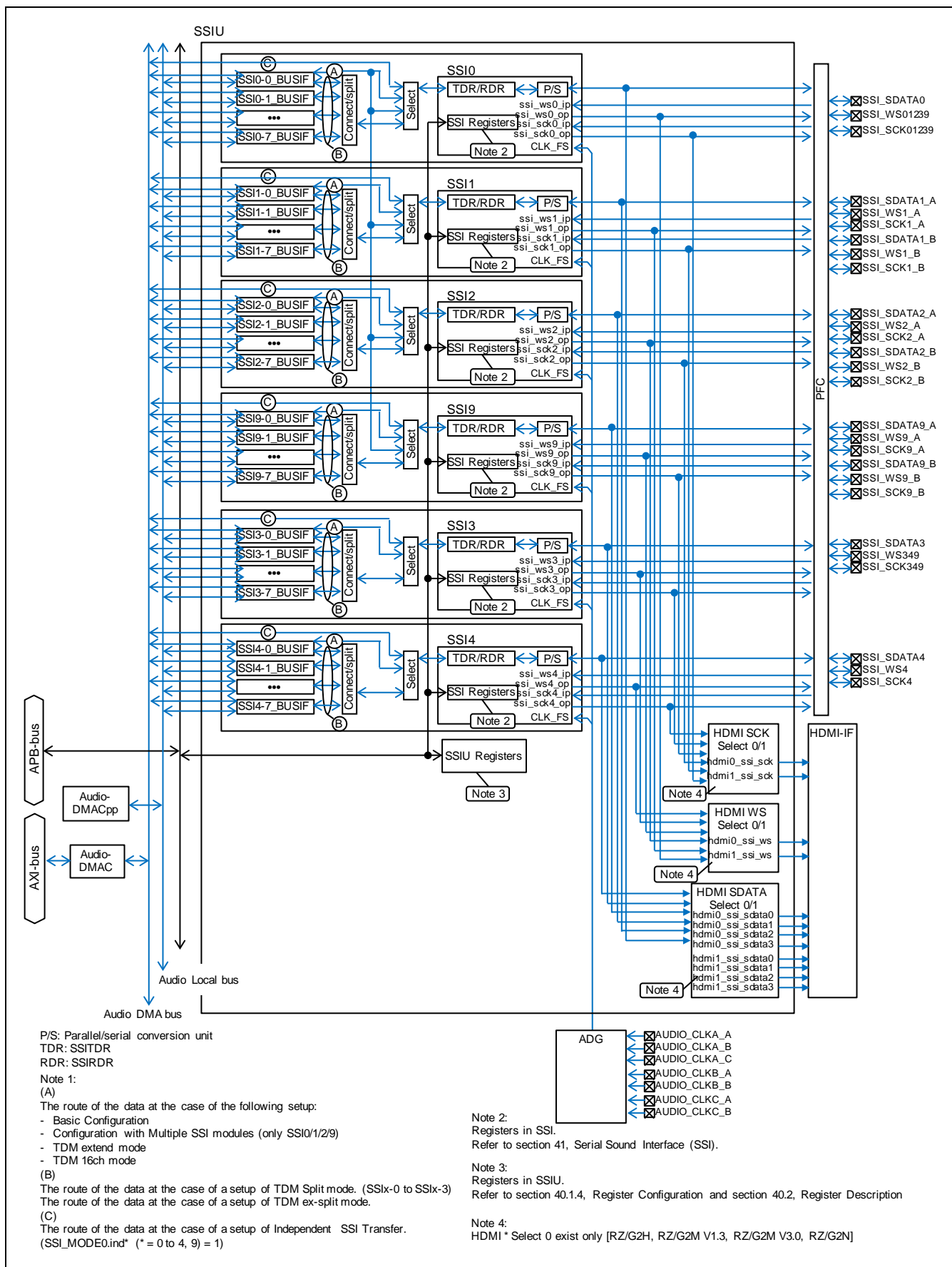


Figure 40.1 Block Diagram of SSIU (SSI0, SSI1, SSI2, SSI3, SSI4, and SSI9) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

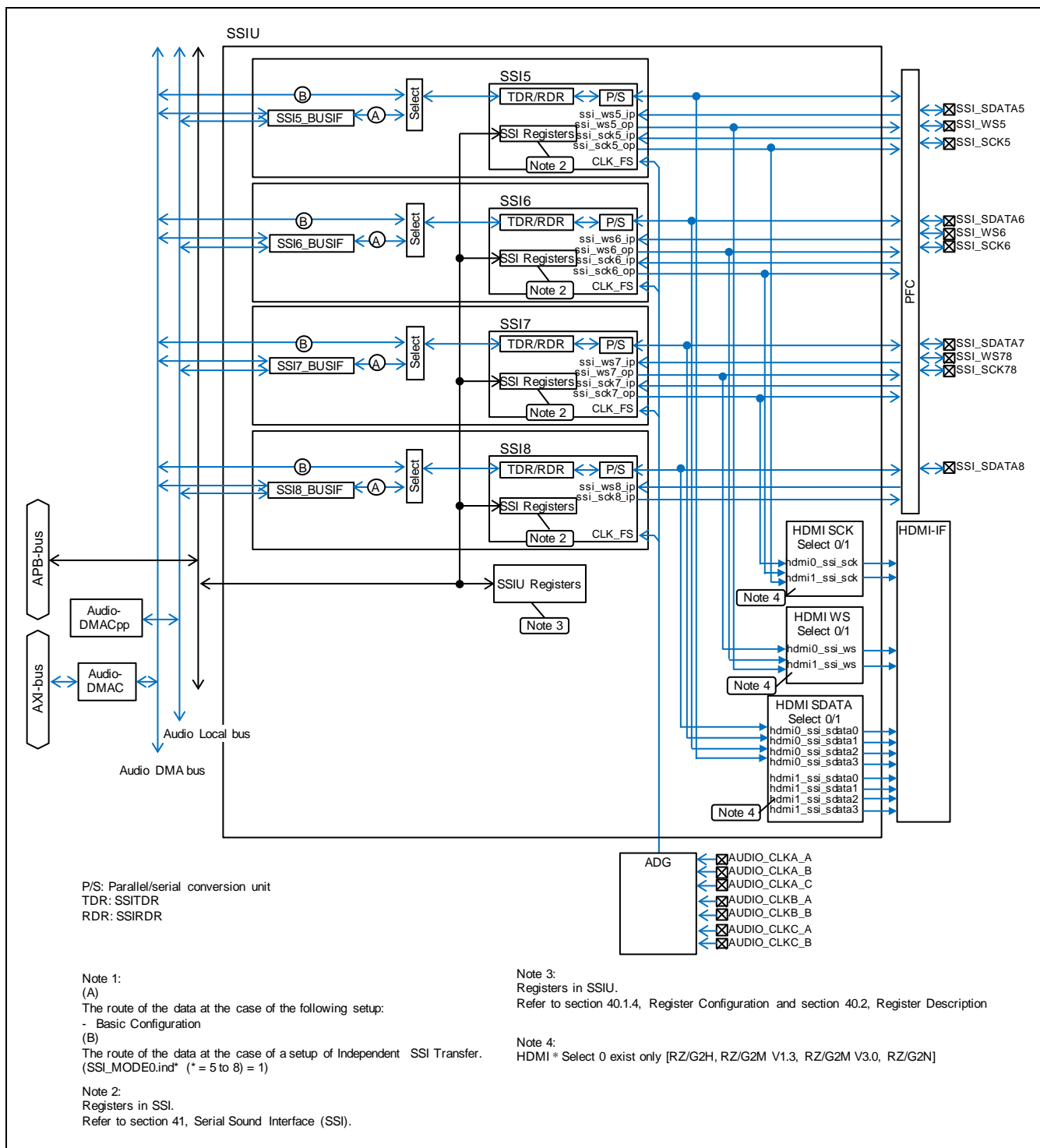


Figure 40.2 Block Diagram of SSIU (SSI5, SSI6, SSI7, and SSI8)
[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

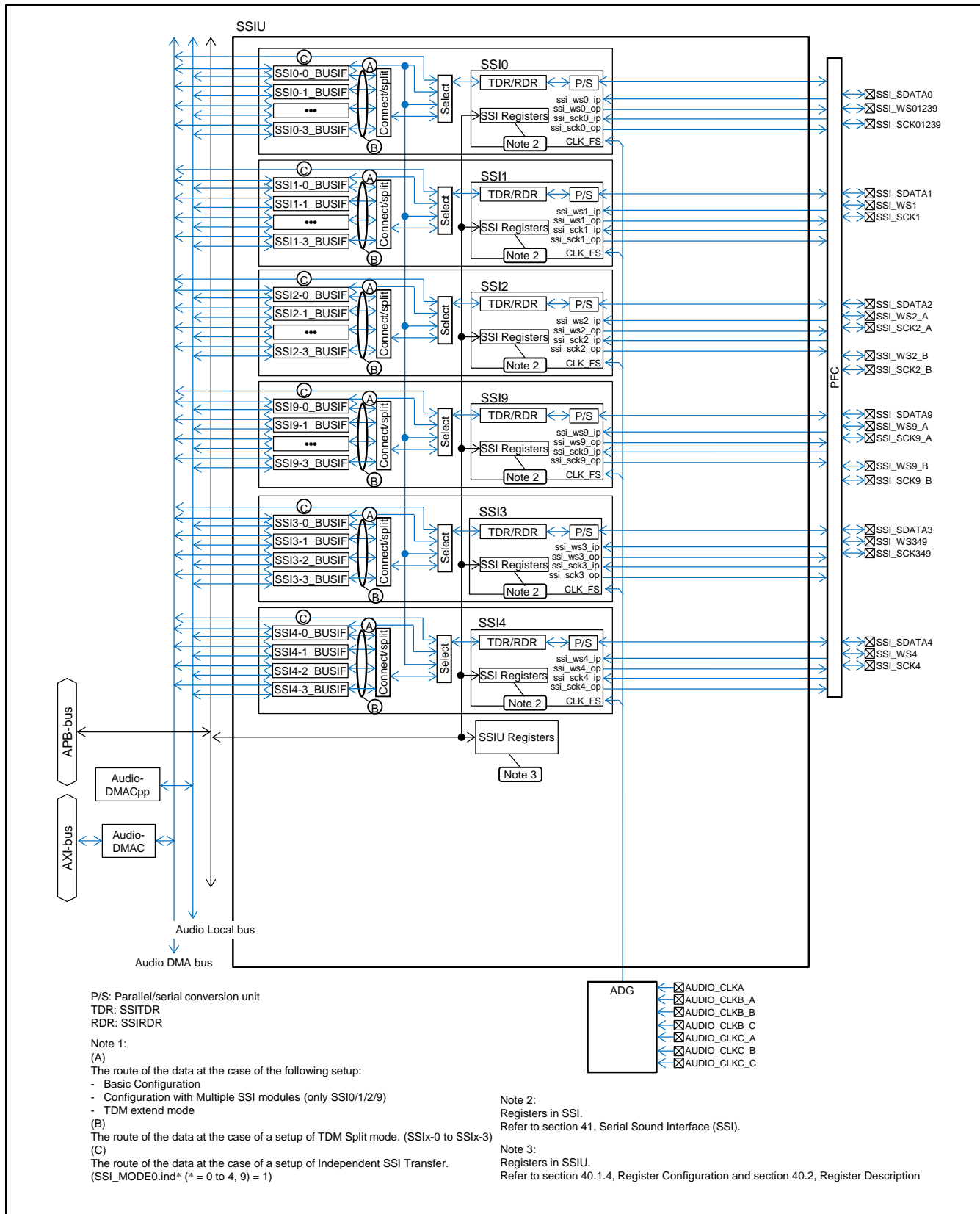


Figure 40.3 Block Diagram of SSIU (SSI0, SSI1, SSI2, SSI3, SSI4, and SSI9) [RZ/G2E]

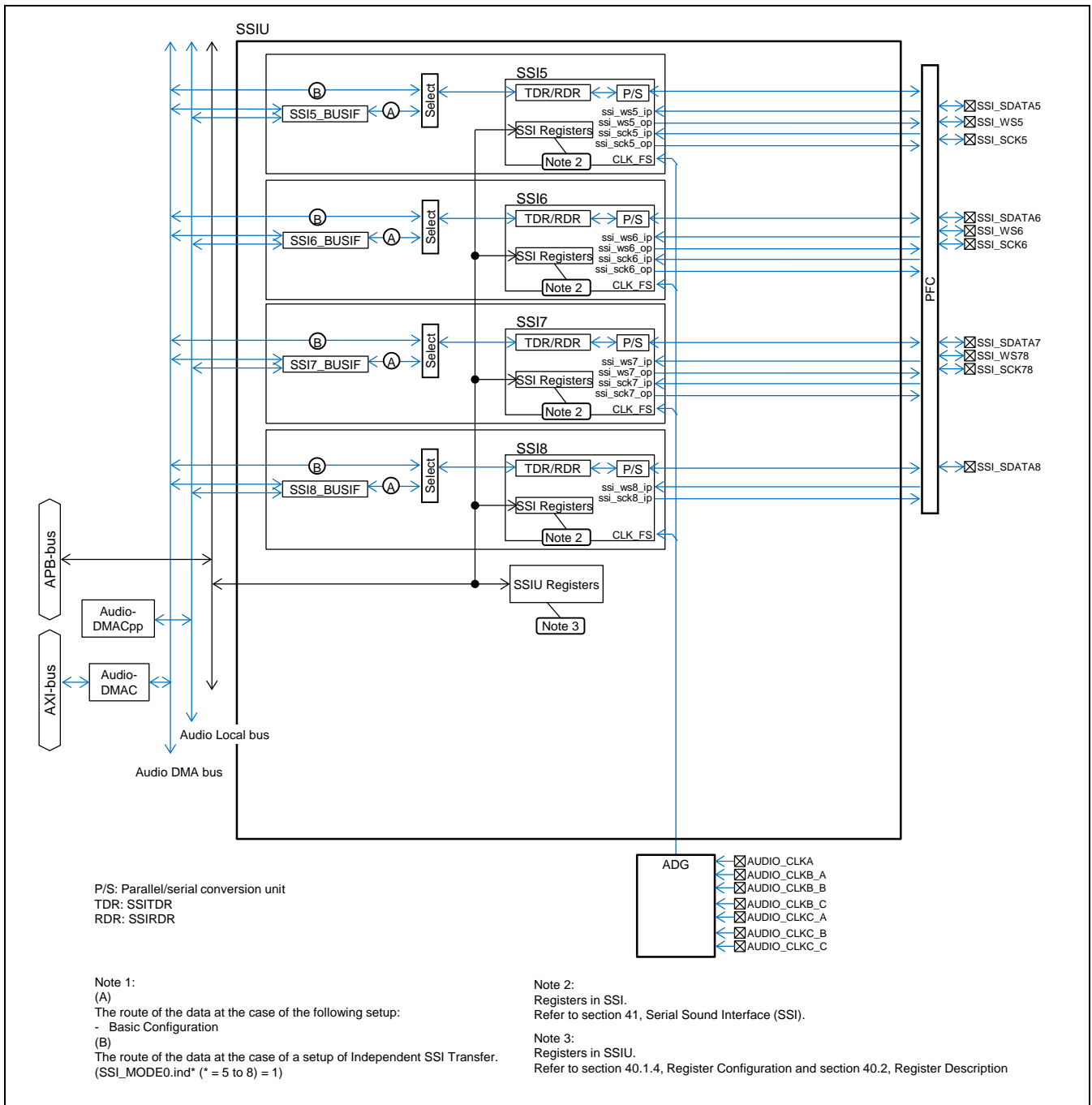


Figure 40.4 Block Diagram of SSIU (SSI5, SSI6, SSI7, and SSI8) [RZ/G2E]

40.1.3 External Pins

Table 40.1 shows the pin configuration.

Table 40.1 Pin Configuration

| | | | | Second Generation RZ/G Series Products | | | | |
|--|--------------|-----|---|---|----------------------------|--------|--------|--|
| Name | Pin Name | I/O | Function | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Serial clock pin for SSI0, SSI1, SSI2, SSI3 and SSI9 | SSI_SCK01239 | I/O | Serial clock (SSI0 only or combination of SSI0, SSI1, SSI2, SSI3, and SSI9) | √ | √ | √ | √ | |
| Word select pin for SSI0, SSI1, SSI2, SSI3 and SSI9 | SSI_WS01239 | I/O | Word select (SSI0 only or combination of SSI0, SSI1, SSI2, SSI3, and SSI9) | √ | √ | √ | √ | |
| Serial data pin for SSI0 | SSI_SDATA0 | I/O | Serial data | √ | √ | √ | √ | |
| Serial clock pin for SSI1 | SSI_SCK1 | I/O | Serial clock | — | — | — | √ | |
| Serial clock pin for SSI1 (Group A) | SSI_SCK1_A | I/O | Serial clock | √ | √ | √ | — | |
| Serial clock pin for SSI1 (Group B) | SSI_SCK1_B | I/O | Serial clock | √ | √ | √ | — | |
| Word select pin for SSI1 | SSI_WS1 | I/O | Word select | — | — | — | √ | |
| Word select pin for SSI1 (Group A) | SSI_WS1_A | I/O | Word select | √ | √ | √ | — | |
| Word select pin for SSI1 (Group B) | SSI_WS1_B | I/O | Word select | √ | √ | √ | — | |
| Serial data pin for SSI1 | SSI_SDATA1 | I/O | Serial data | — | — | — | √ | |
| Serial data pin for SSI1 (Group A) | SSI_SDATA1_A | I/O | Serial data | √ | √ | √ | — | |
| Serial data pin for SSI1 (Group B) | SSI_SDATA1_B | I/O | Serial data | √ | √ | √ | — | |
| Serial clock pin for SSI2 (Group A) | SSI_SCK2_A | I/O | Serial clock | √ | √ | √ | √ | |
| Serial clock pin for SSI2 (Group B) | SSI_SCK2_B | I/O | Serial clock | √ | √ | √ | √ | |
| Word select pin for SSI2 (Group A) | SSI_WS2_A | I/O | Word select | √ | √ | √ | √ | |
| Word select pin for SSI2 (Group B) | SSI_WS2_B | I/O | Word select | √ | √ | √ | √ | |
| Serial data pin for SSI2 | SSI_SDATA2 | I/O | Serial data | — | — | — | √ | |
| Serial data pin for SSI2 (Group A) | SSI_SDATA2_A | I/O | Serial data | √ | √ | √ | — | |
| Serial data pin for SSI2 (Group B) | SSI_SDATA2_B | I/O | Serial data | √ | √ | √ | — | |
| Serial clock pin for SSI3, SSI4 and SSI9 | SSI_SCK349 | I/O | Serial clock (SSI3 only, common to SSI3 and SSI4, or combination of SSI3, SSI4, and SSI9) | √ | √ | √ | √ | |
| Word select pin for SSI3, SSI4 and SSI9 | SSI_WS349 | I/O | Word select (SSI3 only, common to SSI3 and SSI4, or combination of SSI3, SSI4, and SSI9) | √ | √ | √ | √ | |
| Serial data pin for SSI3 | SSI_SDATA3 | I/O | Serial data | √ | √ | √ | √ | |
| Serial clock pin for SSI4 | SSI_SCK4 | I/O | Serial clock | √ | √ | √ | √ | |
| Word select pin for SSI4 | SSI_WS4 | I/O | Word select | √ | √ | √ | √ | |
| Serial data pin for SSI4 | SSI_SDATA4 | I/O | Serial data | √ | √ | √ | √ | |
| Serial clock pin for SSI5 | SSI_SCK5 | I/O | Serial clock | √ | √ | √ | √ | |
| Word select pin for SSI5 | SSI_WS5 | I/O | Word select | √ | √ | √ | √ | |

| | | | | Second Generation RZ/G Series Products | | | |
|-------------------------------------|--------------|-----|--|---|----------------------------|--------|--------|
| Name | Pin Name | I/O | Function | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Serial data pin for SSI5 | SSI_SDATA5 | I/O | Serial data | √ | √ | √ | √ |
| Serial clock pin for SSI6 | SSI_SCK6 | I/O | Serial clock | √ | √ | √ | √ |
| Word select pin for SSI6 | SSI_WS6 | I/O | Word select | √ | √ | √ | √ |
| Serial data pin for SSI6 | SSI_SDATA6 | I/O | Serial data | √ | √ | √ | √ |
| Serial clock pin for SSI7 and SSI8 | SSI_SCK78 | I/O | Serial clock (common to SSI7 and SSI8) | √ | √ | √ | √ |
| Word select pin for SSI7 and SSI8 | SSI_WS78 | I/O | Word select (common to SSI7 and SSI8) | √ | √ | √ | √ |
| Serial data pin for SSI7 | SSI_SDATA7 | I/O | Serial data | √ | √ | √ | √ |
| Serial data pin for SSI8 | SSI_SDATA8 | I/O | Serial data | √ | √ | √ | √ |
| Serial clock pin for SSI9 (Group A) | SSI_SCK9_A | I/O | Serial clock | √ | √ | √ | √ |
| Serial clock pin for SSI9 (Group B) | SSI_SCK9_B | I/O | Serial clock | √ | √ | √ | √ |
| Word select pin for SSI9 (Group A) | SSI_WS9_A | I/O | Word select | √ | √ | √ | √ |
| Word select pin for SSI9 (Group B) | SSI_WS9_B | I/O | Word select | √ | √ | √ | √ |
| Serial data pin for SSI9 | SSI_SDATA9 | I/O | Serial data | — | — | — | √ |
| Serial data pin for SSI9 (Group A) | SSI_SDATA9_A | I/O | Serial data | √ | √ | √ | — |
| Serial data pin for SSI9 (Group B) | SSI_SDATA9_B | I/O | Serial data | √ | √ | √ | — |

Note: Although each SSI is multiplexed on multiple LSI pins, the combinations of pins in use must be from the same group. For Example, the use of combinations such as SSI_SCK1_A, SSI_WS1_B, and SSI_SDATA1_A is prohibited. Refer to section 8, Pin Function Controller (PFC) for details.

40.1.4 Register Configuration

Table 40.2 shows the register configuration. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined. Access the register as a longword (32 bits). Operation cannot be guaranteed if the register is not accessed as a longword.

Table 40.2 Register Configuration

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|------------------------|-----|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI0_0 BUSIF Mode Register | SSI0-0_BUSIF_MODE | R/W | H'EC54_0000 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI0_0 BUSIF Audio Information Register | SSI0-0_BUSIF_ADINR | R/W | H'EC54_0004 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI0_0 BUSIF Data Align Register | SSI0-0_BUSIF_DALIGN | R/W | H'EC54_0008 | H'7654_3210 | 32 | √ | √ | √ | √ |
| SSI0_0 Mode Register | SSI0-0_MODE | R/W | H'EC54_000C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI0_0 Control Register | SSI0-0_CONTROL | R/W | H'EC54_0010 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI0_0 Status Register | SSI0-0_STATUS | R | H'EC54_0014 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI0_0 Interrupt Enable Register | SSI0-0_INT_ENABLE_MAIN | R/W | H'EC54_0018 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI0_1 BUSIF Mode Register | SSI0-1_BUSIF_MODE | R/W | H'EC54_0020 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI0_1 BUSIF Audio Information Register | SSI0-1_BUSIF_ADINR | R/W | H'EC54_0024 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI0_1 BUSIF Data Align Register | SSI0-1_BUSIF_DALIGN | R/W | H'EC54_0028 | H'0000_0032 | 32 | √ | √ | √ | √ |
| SSI0_2 BUSIF Mode Register | SSI0-2_BUSIF_MODE | R/W | H'EC54_0040 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI0_2 BUSIF Audio Information Register | SSI0-2_BUSIF_ADINR | R/W | H'EC54_0044 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI0_2 BUSIF Data Align Register | SSI0-2_BUSIF_DALIGN | R/W | H'EC54_0048 | H'0000_7654 | 32 | √ | √ | √ | √ |
| SSI0_3 BUSIF Mode Register | SSI0-3_BUSIF_MODE | R/W | H'EC54_0060 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI0_3 BUSIF Audio Information Register | SSI0-3_BUSIF_ADINR | R/W | H'EC54_0064 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI0_3 BUSIF Data Align Register | SSI0-3_BUSIF_DALIGN | R/W | H'EC54_0068 | H'0000_0076 | 32 | √ | √ | √ | √ |
| SSI1_0 BUSIF Mode Register | SSI1-0_BUSIF_MODE | R/W | H'EC54_0080 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI1_0 BUSIF Audio Information Register | SSI1-0_BUSIF_ADINR | R/W | H'EC54_0084 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI1_0 BUSIF Data Align Register | SSI1-0_BUSIF_DALIGN | R/W | H'EC54_0088 | H'7654_3210 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|------------------------|-----|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI1_0 Mode Register | SSI1-0_MODE | R/W | H'EC54_008C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI1_0 Control Register | SSI1-0_CONTROL | R/W | H'EC54_0090 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI1_0 Status Register | SSI1-0_STATUS | R | H'EC54_0094 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI1_0 Interrupt Enable Register | SSI1-0_INT_ENABLE_MAIN | R/W | H'EC54_0098 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI1_1 BUSIF Mode Register | SSI1-1_BUSIF_MODE | R/W | H'EC54_00A0 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI1_1 BUSIF Audio Information Register | SSI1-1_BUSIF_ADINR | R/W | H'EC54_00A4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI1_1 BUSIF Data Align Register | SSI1-1_BUSIF_DALIGN | R/W | H'EC54_00A8 | H'0000_0032 | 32 | √ | √ | √ | √ |
| SSI1_2 BUSIF Mode Register | SSI1-2_BUSIF_MODE | R/W | H'EC54_00C0 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI1_2 BUSIF Audio Information Register | SSI1-2_BUSIF_ADINR | R/W | H'EC54_00C4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI1_2 BUSIF Data Align Register | SSI1-2_BUSIF_DALIGN | R/W | H'EC54_00C8 | H'0000_7654 | 32 | √ | √ | √ | √ |
| SSI1_3 BUSIF Mode Register | SSI1-3_BUSIF_MODE | R/W | H'EC54_00E0 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI1_3 BUSIF Audio Information Register | SSI1-3_BUSIF_ADINR | R/W | H'EC54_00E4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI1_3 BUSIF Data Align Register | SSI1-3_BUSIF_DALIGN | R/W | H'EC54_00E8 | H'0000_0076 | 32 | √ | √ | √ | √ |
| SSI2_0 BUSIF Mode Register | SSI2-0_BUSIF_MODE | R/W | H'EC54_0100 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI2_0 BUSIF Audio Information Register | SSI2-0_BUSIF_ADINR | R/W | H'EC54_0104 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI2_0 BUSIF Data Align Register | SSI2-0_BUSIF_DALIGN | R/W | H'EC54_0108 | H'7654_3210 | 32 | √ | √ | √ | √ |
| SSI2_0 Mode Register | SSI2-0_MODE | R/W | H'EC54_010C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI2_0 Control Register | SSI2-0_CONTROL | R/W | H'EC54_0110 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI2_0 Status Register | SSI2-0_STATUS | R | H'EC54_0114 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI2_0 Interrupt Enable Register | SSI2-0_INT_ENABLE_MAIN | R/W | H'EC54_0118 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI2_1 BUSIF Mode Register | SSI2-1_BUSIF_MODE | R/W | H'EC54_0120 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI2_1 BUSIF Audio Information Register | SSI2-1_BUSIF_ADINR | R/W | H'EC54_0124 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI2_1 BUSIF Data Align Register | SSI2-1_BUSIF_DALIGN | R/W | H'EC54_0128 | H'0000_0032 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|------------------------|-----|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI2_2 BUSIF Mode Register | SSI2-2_BUSIF_MODE | R/W | H'EC54_0140 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI2_2 BUSIF Audio Information Register | SSI2-2_BUSIF_ADINR | R/W | H'EC54_0144 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI2_2 BUSIF Data Align Register | SSI2-2_BUSIF_DALIGN | R/W | H'EC54_0148 | H'0000_7654 | 32 | √ | √ | √ | √ |
| SSI2_3 BUSIF Mode Register | SSI2-3_BUSIF_MODE | R/W | H'EC54_0160 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI2_3 BUSIF Audio Information Register | SSI2-3_BUSIF_ADINR | R/W | H'EC54_0164 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI2_3 BUSIF Data Align Register | SSI2-3_BUSIF_DALIGN | R/W | H'EC54_0168 | H'0000_0076 | 32 | √ | √ | √ | √ |
| SSI3_0 BUSIF Mode Register | SSI3-0_BUSIF_MODE | R/W | H'EC54_0180 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI3_0 BUSIF Audio Information Register | SSI3-0_BUSIF_ADINR | R/W | H'EC54_0184 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI3_0 BUSIF Data Align Register | SSI3-0_BUSIF_DALIGN | R/W | H'EC54_0188 | H'7654_3210 | 32 | √ | √ | √ | √ |
| SSI3_0 Mode Register | SSI3-0_MODE | R/W | H'EC54_018C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI3_0 Control Register | SSI3-0_CONTROL | R/W | H'EC54_0190 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI3_0 Status Register | SSI3-0_STATUS | R | H'EC54_0194 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI3_0 Interrupt Enable Register | SSI3-0_INT_ENABLE_MAIN | R/W | H'EC54_0198 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI3_1 BUSIF Mode Register | SSI3-1_BUSIF_MODE | R/W | H'EC54_01A0 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI3_1 BUSIF Audio Information Register | SSI3-1_BUSIF_ADINR | R/W | H'EC54_01A4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI3_1 BUSIF Data Align Register | SSI3-1_BUSIF_DALIGN | R/W | H'EC54_01A8 | H'0000_0032 | 32 | √ | √ | √ | √ |
| SSI3_2 BUSIF Mode Register | SSI3-2_BUSIF_MODE | R/W | H'EC54_01C0 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI3_2 BUSIF Audio Information Register | SSI3-2_BUSIF_ADINR | R/W | H'EC54_01C4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI3_2 BUSIF Data Align Register | SSI3-2_BUSIF_DALIGN | R/W | H'EC54_01C8 | H'0000_7654 | 32 | √ | √ | √ | √ |
| SSI3_3 BUSIF Mode Register | SSI3-3_BUSIF_MODE | R/W | H'EC54_01E0 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI3_3 BUSIF Audio Information Register | SSI3-3_BUSIF_ADINR | R/W | H'EC54_01E4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI3_3 BUSIF Data Align Register | SSI3-3_BUSIF_DALIGN | R/W | H'EC54_01E8 | H'0000_0076 | 32 | √ | √ | √ | √ |
| SSI4_0 BUSIF Mode Register | SSI4-0_BUSIF_MODE | R/W | H'EC54_0200 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI4_0 BUSIF Audio Information Register | SSI4-0_BUSIF_ADINR | R/W | H'EC54_0204 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI4_0 BUSIF Data Align Register | SSI4-0_BUSIF_DALIGN | R/W | H'EC54_0208 | H'7654_3210 | 32 | √ | √ | √ | √ |
| SSI4_0 Mode Register | SSI4-0_MODE | R/W | H'EC54_020C | H'0000_0000 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|------------------------|-----|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI4_0 Control Register | SSI4-0_CONTROL | R/W | H'EC54_0210 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI4_0 Status Register | SSI4-0_STATUS | R | H'EC54_0214 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI4_0 Interrupt Enable Register | SSI4-0_INT_ENABLE_MAIN | R/W | H'EC54_0218 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI4_1 BUSIF Mode Register | SSI4-1_BUSIF_MODE | R/W | H'EC54_0220 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI4_1 BUSIF Audio Information Register | SSI4-1_BUSIF_ADINR | R/W | H'EC54_0224 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI4_1 BUSIF Data Align Register | SSI4-1_BUSIF_DALIGN | R/W | H'EC54_0228 | H'0000_0032 | 32 | √ | √ | √ | √ |
| SSI4_2 BUSIF Mode Register | SSI4-2_BUSIF_MODE | R/W | H'EC54_0240 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI4_2 BUSIF Audio Information Register | SSI4-2_BUSIF_ADINR | R/W | H'EC54_0244 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI4_2 BUSIF Data Align Register | SSI4-2_BUSIF_DALIGN | R/W | H'EC54_0248 | H'0000_7654 | 32 | √ | √ | √ | √ |
| SSI4_3 BUSIF Mode Register | SSI4-3_BUSIF_MODE | R/W | H'EC54_0260 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI4_3 BUSIF Audio Information Register | SSI4-3_BUSIF_ADINR | R/W | H'EC54_0264 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI4_3 BUSIF Data Align Register | SSI4-3_BUSIF_DALIGN | R/W | H'EC54_0268 | H'0000_0076 | 32 | √ | √ | √ | √ |
| SSI5 BUSIF Mode Register | SSI5_BUSIF_MODE | R/W | H'EC54_0280 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI5 BUSIF Audio Information Register | SSI5_BUSIF_ADINR | R/W | H'EC54_0284 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI5 BUSIF Data Align Register | SSI5_BUSIF_DALIGN | R/W | H'EC54_0288 | H'7654_3210 | 32 | √ | √ | √ | √ |
| SSI5 Control Register | SSI5_CONTROL | R/W | H'EC54_0290 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI5 Status Register | SSI5_STATUS | R | H'EC54_0294 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI5 Interrupt Enable Register | SSI5_INT_ENABLE_MAIN | R/W | H'EC54_0298 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI6 BUSIF Mode Register | SSI6_BUSIF_MODE | R/W | H'EC54_0300 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI6 BUSIF Audio Information Register | SSI6_BUSIF_ADINR | R/W | H'EC54_0304 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI6 BUSIF Data Align Register | SSI6_BUSIF_DALIGN | R/W | H'EC54_0308 | H'7654_3210 | 32 | √ | √ | √ | √ |
| SSI6 Control Register | SSI6_CONTROL | R/W | H'EC54_0310 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI6 Status Register | SSI6_STATUS | R | H'EC54_0314 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI6 Interrupt Enable Register | SSI6_INT_ENABLE_MAIN | R/W | H'EC54_0318 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI7 BUSIF Mode Register | SSI7_BUSIF_MODE | R/W | H'EC54_0380 | H'0000_0001 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|------------------------|-----|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI7 BUSIF Audio Information Register | SSI7_BUSIF_ADINR | R/W | H'EC54_0384 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI7 BUSIF Data Align Register | SSI7_BUSIF_DALIGN | R/W | H'EC54_0388 | H'7654_3210 | 32 | √ | √ | √ | √ |
| SSI7 Control Register | SSI7_CONTROL | R/W | H'EC54_0390 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI7 Status Register | SSI7_STATUS | R | H'EC54_0394 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI7 Interrupt Enable Register | SSI7_INT_ENABLE_MAIN | R/W | H'EC54_0398 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI8 BUSIF Mode Register | SSI8_BUSIF_MODE | R/W | H'EC54_0400 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI8 BUSIF Audio Information Register | SSI8_BUSIF_ADINR | R/W | H'EC54_0404 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI8 BUSIF Data Align Register | SSI8_BUSIF_DALIGN | R/W | H'EC54_0408 | H'7654_3210 | 32 | √ | √ | √ | √ |
| SSI8 Control Register | SSI8_CONTROL | R/W | H'EC54_0410 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI8 Status Register | SSI8_STATUS | R | H'EC54_0414 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI8 Interrupt Enable Register | SSI8_INT_ENABLE_MAIN | R/W | H'EC54_0418 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI9_0 BUSIF Mode Register | SSI9-0_BUSIF_MODE | R/W | H'EC54_0480 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI9_0 BUSIF Audio Information Register | SSI9-0_BUSIF_ADINR | R/W | H'EC54_0484 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI9_0 BUSIF Data Align Register | SSI9-0_BUSIF_DALIGN | R/W | H'EC54_0488 | H'7654_3210 | 32 | √ | √ | √ | √ |
| SSI9_0 Mode Register | SSI9-0_MODE | R/W | H'EC54_048C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI9_0 Control Register | SSI9-0_CONTROL | R/W | H'EC54_0490 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI9_0 Status Register | SSI9-0_STATUS | R | H'EC54_0494 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI9_0 Interrupt Enable Register | SSI9-0_INT_ENABLE_MAIN | R/W | H'EC54_0498 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI9_1 BUSIF Mode Register | SSI9-1_BUSIF_MODE | R/W | H'EC54_04A0 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI9_1 BUSIF Audio Information Register | SSI9-1_BUSIF_ADINR | R/W | H'EC54_04A4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI9_1 BUSIF Data Align Register | SSI9-1_BUSIF_DALIGN | R/W | H'EC54_04A8 | H'0000_0032 | 32 | √ | √ | √ | √ |
| SSI9_2 BUSIF Mode Register | SSI9-2_BUSIF_MODE | R/W | H'EC54_04C0 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI9_2 BUSIF Audio Information Register | SSI9-2_BUSIF_ADINR | R/W | H'EC54_04C4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI9_2 BUSIF Data Align Register | SSI9-2_BUSIF_DALIGN | R/W | H'EC54_04C8 | H'0000_7654 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|---------------------|-----|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI9_3 BUSIF Mode Register | SSI9-3_BUSIF_MODE | R/W | H'EC54_04E0 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SSI9_3 BUSIF Audio Information Register | SSI9-3_BUSIF_ADINR | R/W | H'EC54_04E4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI9_3 BUSIF Data Align Register | SSI9-3_BUSIF_DALIGN | R/W | H'EC54_04E8 | H'0000_0076 | 32 | √ | √ | √ | √ |
| SSI0_4 BUSIF Mode Register | SSI0-4_BUSIF_MODE | R/W | H'EC54_0500 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI0_4 BUSIF Audio Information Register | SSI0-4_BUSIF_ADINR | R/W | H'EC54_0504 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI0_4 BUSIF Data Align Register | SSI0-4_BUSIF_DALIGN | R/W | H'EC54_0508 | H'FEDC_BA98 | 32 | √ | √ | √ | — |
| SSI0_5 BUSIF Mode Register | SSI0-5_BUSIF_MODE | R/W | H'EC54_0520 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI0_5 BUSIF Audio Information Register | SSI0-5_BUSIF_ADINR | R/W | H'EC54_0524 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI0_5 BUSIF Data Align Register | SSI0-5_BUSIF_DALIGN | R/W | H'EC54_0528 | H'0000_00BA | 32 | √ | √ | √ | — |
| SSI0_6 BUSIF Mode Register | SSI0-6_BUSIF_MODE | R/W | H'EC54_0540 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI0_6 BUSIF Audio Information Register | SSI0-6_BUSIF_ADINR | R/W | H'EC54_0544 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI0_6 BUSIF Data Align Register | SSI0-6_BUSIF_DALIGN | R/W | H'EC54_0548 | H'0000_FEDC | 32 | √ | √ | √ | — |
| SSI0_7 BUSIF Mode Register | SSI0-7_BUSIF_MODE | R/W | H'EC54_0560 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI0_7 BUSIF Audio Information Register | SSI0-7_BUSIF_ADINR | R/W | H'EC54_0564 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI0_7 BUSIF Data Align Register | SSI0-7_BUSIF_DALIGN | R/W | H'EC54_0568 | H'0000_00FE | 32 | √ | √ | √ | — |
| SSI1_4 BUSIF Mode Register | SSI1-4_BUSIF_MODE | R/W | H'EC54_0580 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI1_4 BUSIF Audio Information Register | SSI1-4_BUSIF_ADINR | R/W | H'EC54_0584 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI1_4 BUSIF Data Align Register | SSI1-4_BUSIF_DALIGN | R/W | H'EC54_0588 | H'FEDC_BA98 | 32 | √ | √ | √ | — |
| SSI1_5 BUSIF Mode Register | SSI1-5_BUSIF_MODE | R/W | H'EC54_05A0 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI1_5 BUSIF Audio Information Register | SSI1-5_BUSIF_ADINR | R/W | H'EC54_05A4 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI1_5 BUSIF Data Align Register | SSI1-5_BUSIF_DALIGN | R/W | H'EC54_05A8 | H'0000_00BA | 32 | √ | √ | √ | — |
| SSI1_6 BUSIF Mode Register | SSI1-6_BUSIF_MODE | R/W | H'EC54_05C0 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI1_6 BUSIF Audio Information Register | SSI1-6_BUSIF_ADINR | R/W | H'EC54_05C4 | H'0000_0000 | 32 | √ | √ | √ | — |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|---------------------|-----|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI1_6 BUSIF Data Align Register | SSI1-6_BUSIF_DALIGN | R/W | H'EC54_05C8 | H'0000_FEDC | 32 | √ | √ | √ | — |
| SSI1_7 BUSIF Mode Register | SSI1-7_BUSIF_MODE | R/W | H'EC54_05E0 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI1_7 BUSIF Audio Information Register | SSI1-7_BUSIF_ADINR | R/W | H'EC54_05E4 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI1_7 BUSIF Data Align Register | SSI1-7_BUSIF_DALIGN | R/W | H'EC54_05E8 | H'0000_00FE | 32 | √ | √ | √ | — |
| SSI2_4 BUSIF Mode Register | SSI2-4_BUSIF_MODE | R/W | H'EC54_0600 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI2_4 BUSIF Audio Information Register | SSI2-4_BUSIF_ADINR | R/W | H'EC54_0604 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI2_4 BUSIF Data Align Register | SSI2-4_BUSIF_DALIGN | R/W | H'EC54_0608 | H'FEDC_BA98 | 32 | √ | √ | √ | — |
| SSI2_5 BUSIF Mode Register | SSI2-5_BUSIF_MODE | R/W | H'EC54_0620 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI2_5 BUSIF Audio Information Register | SSI2-5_BUSIF_ADINR | R/W | H'EC54_0624 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI2_5 BUSIF Data Align Register | SSI2-5_BUSIF_DALIGN | R/W | H'EC54_0628 | H'0000_00BA | 32 | √ | √ | √ | — |
| SSI2_6 BUSIF Mode Register | SSI2-6_BUSIF_MODE | R/W | H'EC54_0640 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI2_6 BUSIF Audio Information Register | SSI2-6_BUSIF_ADINR | R/W | H'EC54_0644 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI2_6 BUSIF Data Align Register | SSI2-6_BUSIF_DALIGN | R/W | H'EC54_0648 | H'0000_FEDC | 32 | √ | √ | √ | — |
| SSI2_7 BUSIF Mode Register | SSI2-7_BUSIF_MODE | R/W | H'EC54_0660 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI2_7 BUSIF Audio Information Register | SSI2-7_BUSIF_ADINR | R/W | H'EC54_0664 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI2_7 BUSIF Data Align Register | SSI2-7_BUSIF_DALIGN | R/W | H'EC54_0668 | H'0000_00FE | 32 | √ | √ | √ | — |
| SSI3_4 BUSIF Mode Register | SSI3-4_BUSIF_MODE | R/W | H'EC54_0680 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI3_4 BUSIF Audio Information Register | SSI3-4_BUSIF_ADINR | R/W | H'EC54_0684 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI3_4 BUSIF Data Align Register | SSI3-4_BUSIF_DALIGN | R/W | H'EC54_0688 | H'FEDC_BA98 | 32 | √ | √ | √ | — |
| SSI3_5 BUSIF Mode Register | SSI3-5_BUSIF_MODE | R/W | H'EC54_06A0 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI3_5 BUSIF Audio Information Register | SSI3-5_BUSIF_ADINR | R/W | H'EC54_06A4 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI3_5 BUSIF Data Align Register | SSI3-5_BUSIF_DALIGN | R/W | H'EC54_06A8 | H'0000_00BA | 32 | √ | √ | √ | — |
| SSI3_6 BUSIF Mode Register | SSI3-6_BUSIF_MODE | R/W | H'EC54_06C0 | H'0000_0001 | 32 | √ | √ | √ | — |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|---------------------|-----------|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI3_6 BUSIF Audio Information Register | SSI3-6_BUSIF_ADINR | R/W | H'EC54_06C4 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI3_6 BUSIF Data Align Register | SSI3-6_BUSIF_DALIGN | R/W | H'EC54_06C8 | H'0000_FEDC | 32 | √ | √ | √ | — |
| SSI3_7 BUSIF Mode Register | SSI3-7_BUSIF_MODE | R/W | H'EC54_06E0 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI3_7 BUSIF Audio Information Register | SSI3-7_BUSIF_ADINR | R/W | H'EC54_06E4 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI3_7 BUSIF Data Align Register | SSI3-7_BUSIF_DALIGN | R/W | H'EC54_06E8 | H'0000_00FE | 32 | √ | √ | √ | — |
| SSI4_4 BUSIF Mode Register | SSI4-4_BUSIF_MODE | R/W | H'EC54_0700 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI4_4 BUSIF Audio Information Register | SSI4-4_BUSIF_ADINR | R/W | H'EC54_0704 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI4_4 BUSIF Data Align Register | SSI4-4_BUSIF_DALIGN | R/W | H'EC54_0708 | H'FEDC_BA98 | 32 | √ | √ | √ | — |
| SSI4_5 BUSIF Mode Register | SSI4-5_BUSIF_MODE | R/W | H'EC54_0720 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI4_5 BUSIF Audio Information Register | SSI4-5_BUSIF_ADINR | R/W | H'EC54_0724 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI4_5 BUSIF Data Align Register | SSI4-5_BUSIF_DALIGN | R/W | H'EC54_0728 | H'0000_00BA | 32 | √ | √ | √ | — |
| SSI4_6 BUSIF Mode Register | SSI4-6_BUSIF_MODE | R/W | H'EC54_0740 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI4_6 BUSIF Audio Information Register | SSI4-6_BUSIF_ADINR | R/W | H'EC54_0744 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI4_6 BUSIF Data Align Register | SSI4-6_BUSIF_DALIGN | R/W | H'EC54_0748 | H'0000_FEDC | 32 | √ | √ | √ | — |
| SSI4_7 BUSIF Mode Register | SSI4-7_BUSIF_MODE | R/W | H'EC54_0760 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI4_7 BUSIF Audio Information Register | SSI4-7_BUSIF_ADINR | R/W | H'EC54_0764 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI4_7 BUSIF Data Align Register | SSI4-7_BUSIF_DALIGN | R/W | H'EC54_0768 | H'0000_00FE | 32 | √ | √ | √ | — |
| SSI Mode Register 0 | SSI_MODE0 | R/W | H'EC54_0800 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI Mode Register 1 | SSI_MODE1 | R/W | H'EC54_0804 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI Mode Register 2 | SSI_MODE2 | R/W | H'EC54_0808 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI Mode Register 3 | SSI_MODE3 | R/W | H'EC54_080C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI Control Register | SSI_CONTROL | R/W | H'EC54_0810 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI Control Register2 | SSI_CONTROL2 | R/W | H'EC54_0814 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI System Status Register 0 | SSI_SYSTEM_STATUS0 | R/W C1 | H'EC54_0840 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI System Status Register 1 | SSI_SYSTEM_STATUS1 | R/W C1 | H'EC54_0844 | H'0000_0000 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|--|-------------------------|-----------|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI System Status Register 2 | SSI_SYSTEM_STATUS2 | R/W C1 | H'EC54_0848 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI System Status Register 3 | SSI_SYSTEM_STATUS3 | R/W C1 | H'EC54_084C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI System Interrupt Enable Register 0 | SSI_SYSTEM_INT_ENABLE0 | R/W | H'EC54_0850 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI System Interrupt Enable Register 1 | SSI_SYSTEM_INT_ENABLE1 | R/W | H'EC54_0854 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI System Interrupt Enable Register 2 | SSI_SYSTEM_INT_ENABLE2 | R/W | H'EC54_0858 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI System Interrupt Enable Register 3 | SSI_SYSTEM_INT_ENABLE3 | R/W | H'EC54_085C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI System Status Register 4 | SSI_SYSTEM_STATUS4 | R/W C1 | H'EC54_0880 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI System Status Register 5 | SSI_SYSTEM_STATUS5 | R/W C1 | H'EC54_0884 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI System Status Register 6 | SSI_SYSTEM_STATUS6 | R/W C1 | H'EC54_0888 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI System Status Register 7 | SSI_SYSTEM_STATUS7 | R/W C1 | H'EC54_088C | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI System Interrupt Enable Register 4 | SSI_SYSTEM_INT_ENABLE4 | R/W | H'EC54_0890 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI System Interrupt Enable Register 5 | SSI_SYSTEM_INT_ENABLE5 | R/W | H'EC54_0894 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI System Interrupt Enable Register 6 | SSI_SYSTEM_INT_ENABLE6 | R/W | H'EC54_0898 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI System Interrupt Enable Register 7 | SSI_SYSTEM_INT_ENABLE7 | R/W | H'EC54_089C | H'0000_0000 | 32 | √ | √ | √ | — |
| HDMI0 Select Register | HDMI0_SEL | R/W | H'EC54_09E0 | H'FFFF_000F | 32 | √ | √ | √ | — |
| HDMI1 Select Register | HDMI1_SEL | R/W | H'EC54_09E4 | H'FFFF_000F | 32 | — | — | — | — |
| SSI0_0 BUSIF Data Align2 Register | SSI0-0_BUSIF_DALIGN2 | R/W | H'EC54_0A08 | H'FEDC_BA98 | 32 | √ | √ | √ | — |
| SSI0_0 Mode2 Register | SSI0-0_MODE2 | R/W | H'EC54_0A0C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI0_0 Status2 Register | SSI0-0_STATUS2 | R | H'EC54_0A14 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI0_0 Interrupt Enable2 Register | SSI0-0_INT_ENABLE_MAIN2 | R/W | H'EC54_0A18 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI1_0 BUSIF Data Align2 Register | SSI1-0_BUSIF_DALIGN2 | R/W | H'EC54_0A28 | H'FEDC_BA98 | 32 | √ | √ | √ | — |
| SSI1_0 Mode2 Register | SSI1-0_MODE2 | R/W | H'EC54_0A2C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI1_0 Status2 Register | SSI1-0_STATUS2 | R | H'EC54_0A34 | H'0000_0000 | 32 | √ | √ | √ | — |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|-------------------------|-----|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI1_0 Interrupt Enable2 Register | SSI1-0_INT_ENABLE_MAIN2 | R/W | H'EC54_0A38 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI2_0 BUSIF Data Align2 Register | SSI2-0_BUSIF_DALIGN2 | R/W | H'EC54_0A48 | H'FEDC_BA98 | 32 | √ | √ | √ | — |
| SSI2_0 Mode2 Register | SSI2-0_MODE2 | R/W | H'EC54_0A4C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI2_0 Status2 Register | SSI2-0_STATUS2 | R | H'EC54_0A54 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI2_0 Interrupt Enable2 Register | SSI2-0_INT_ENABLE_MAIN2 | R/W | H'EC54_0A58 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI3_0 BUSIF Data Align2 Register | SSI3-0_BUSIF_DALIGN2 | R/W | H'EC54_0A68 | H'FEDC_BA98 | 32 | √ | √ | √ | — |
| SSI3_0 Mode2 Register | SSI3-0_MODE2 | R/W | H'EC54_0A6C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI3_0 Status2 Register | SSI3-0_STATUS2 | R | H'EC54_0A74 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI3_0 Interrupt Enable2 Register | SSI3-0_INT_ENABLE_MAIN2 | R/W | H'EC54_0A78 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI4_0 BUSIF Data Align2 Register | SSI4-0_BUSIF_DALIGN2 | R/W | H'EC54_0A88 | H'FEDC_BA98 | 32 | √ | √ | √ | — |
| SSI4_0 Mode2 Register | SSI4-0_MODE2 | R/W | H'EC54_0A8C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI4_0 Status2 Register | SSI4-0_STATUS2 | R | H'EC54_0A94 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI4_0 Interrupt Enable2 Register | SSI4-0_INT_ENABLE_MAIN2 | R/W | H'EC54_0A98 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI9_0 BUSIF Data Align2 Register | SSI9-0_BUSIF_DALIGN2 | R/W | H'EC54_0B28 | H'FEDC_BA98 | 32 | √ | √ | √ | — |
| SSI9_0 Mode2 Register | SSI9-0_MODE2 | R/W | H'EC54_0B2C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI9_0 Status2 Register | SSI9-0_STATUS2 | R | H'EC54_0B34 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI9_0 Interrupt Enable2 Register | SSI9-0_INT_ENABLE_MAIN2 | R/W | H'EC54_0B38 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI9_4 BUSIF Mode Register | SSI9-4_BUSIF_MODE | R/W | H'EC54_0D80 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI9_4 BUSIF Audio Information Register | SSI9-4_BUSIF_ADINR | R/W | H'EC54_0D84 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI9_4 BUSIF Data Align Register | SSI9-4_BUSIF_DALIGN | R/W | H'EC54_0D88 | H'FEDC_BA98 | 32 | √ | √ | √ | — |
| SSI9_5 BUSIF Mode Register | SSI9-5_BUSIF_MODE | R/W | H'EC54_0DA0 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI9_5 BUSIF Audio Information Register | SSI9-5_BUSIF_ADINR | R/W | H'EC54_0DA4 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI9_5 BUSIF Data Align Register | SSI9-5_BUSIF_DALIGN | R/W | H'EC54_0DA8 | H'0000_00BA | 32 | √ | √ | √ | — |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|---------------------|-----|------------------------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI9_6 BUSIF Mode Register | SSI9-6_BUSIF_MODE | R/W | H'EC54_0DC0 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI9_6 BUSIF Audio Information Register | SSI9-6_BUSIF_ADINR | R/W | H'EC54_0DC4 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI9_6 BUSIF Data Align Register | SSI9-6_BUSIF_DALIGN | R/W | H'EC54_0DC8 | H'0000_FEDC | 32 | √ | √ | √ | — |
| SSI9_7 BUSIF Mode Register | SSI9-7_BUSIF_MODE | R/W | H'EC54_0DE0 | H'0000_0001 | 32 | √ | √ | √ | — |
| SSI9_7 BUSIF Audio Information Register | SSI9-7_BUSIF_ADINR | R/W | H'EC54_0DE4 | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI9_7 BUSIF Data Align Register | SSI9-7_BUSIF_DALIGN | R/W | H'EC54_0DE8 | H'0000_00FE | 32 | √ | √ | √ | — |
| SSI0-0_BUSIF Data Register | SSI0-0_BUSIF | R/W | H'EC10_0000/ H'EC40_0000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI0-1_BUSIF Data Register | SSI0-1_BUSIF | R/W | H'EC10_0400/ H'EC40_0400* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI0-2_BUSIF Data Register | SSI0-2_BUSIF | R/W | H'EC10_0800/ H'EC40_0800* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI0-3_BUSIF Data Register | SSI0-3_BUSIF | R/W | H'EC10_0C00/ H'EC40_0C00* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI1-0_BUSIF Data Register | SSI1-0_BUSIF | R/W | H'EC10_1000/ H'EC40_1000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI1-1_BUSIF Data Register | SSI1-1_BUSIF | R/W | H'EC10_1400/ H'EC40_1400* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI1-2_BUSIF Data Register | SSI1-2_BUSIF | R/W | H'EC10_1800/ H'EC40_1800* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI1-3_BUSIF Data Register | SSI1-3_BUSIF | R/W | H'EC10_1C00/ H'EC40_1C00* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI2-0_BUSIF Data Register | SSI2-0_BUSIF | R/W | H'EC10_2000/ H'EC40_2000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI2-1_BUSIF Data Register | SSI2-1_BUSIF | R/W | H'EC10_2400/ H'EC40_2400* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI2-2_BUSIF Data Register | SSI2-2_BUSIF | R/W | H'EC10_2800/ H'EC40_2800* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI2-3_BUSIF Data Register | SSI2-3_BUSIF | R/W | H'EC10_2C00/ H'EC40_2C00* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI3_0_BUSIF Data Register | SSI3-0_BUSIF | R/W | H'EC10_3000/ H'EC40_3000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI3_1_BUSIF Data Register | SSI3-1_BUSIF | R/W | H'EC10_3400/ H'EC40_3400* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI3_2_BUSIF Data Register | SSI3-2_BUSIF | R/W | H'EC10_3800/ H'EC40_3800* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI3_3_BUSIF Data Register | SSI3-3_BUSIF | R/W | H'EC10_3C00/ H'EC40_3C00* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI4_0_BUSIF Data Register | SSI4-0_BUSIF | R/W | H'EC10_4000/ H'EC40_4000* | H'0000_0000 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|----------------------------|--------------|-----|------------------------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI4_1_BUSIF Data Register | SSI4-1_BUSIF | R/W | H'EC10_4400/ H'EC40_4400* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI4_2_BUSIF Data Register | SSI4-2_BUSIF | R/W | H'EC10_4800/ H'EC40_4800* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI4_3_BUSIF Data Register | SSI4-3_BUSIF | R/W | H'EC10_4C00/ H'EC40_4C00* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI5_BUSIF Data Register | SSI5_BUSIF | R/W | H'EC10_5000/ H'EC40_5000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI6_BUSIF Data Register | SSI6_BUSIF | R/W | H'EC10_6000/ H'EC40_6000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI7_BUSIF Data Register | SSI7_BUSIF | R/W | H'EC10_7000/ H'EC40_7000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI8_BUSIF Data Register | SSI8_BUSIF | R/W | H'EC10_8000/ H'EC40_8000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI9-0_BUSIF Data Register | SSI9-0_BUSIF | R/W | H'EC10_9000/ H'EC40_9000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI9-1_BUSIF Data Register | SSI9-1_BUSIF | R/W | H'EC10_9400/ H'EC40_9400* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI9-2_BUSIF Data Register | SSI9-2_BUSIF | R/W | H'EC10_9800/ H'EC40_9800* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI9-3_BUSIF Data Register | SSI9-3_BUSIF | R/W | H'EC10_9C00/ H'EC40_9C00* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSI0_4_BUSIF Data Register | SSI0-4_BUSIF | R/W | H'EC10_A000/ H'EC40_A000* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI0_5_BUSIF Data Register | SSI0-5_BUSIF | R/W | H'EC10_A400/ H'EC40_A400* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI0_6_BUSIF Data Register | SSI0-6_BUSIF | R/W | H'EC10_A800/ H'EC40_A800* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI0_7_BUSIF Data Register | SSI0-7_BUSIF | R/W | H'EC10_AC00/ H'EC40_AC00* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI1_4_BUSIF Data Register | SSI1-4_BUSIF | R/W | H'EC10_B000/ H'EC40_B000* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI1_5_BUSIF Data Register | SSI1-5_BUSIF | R/W | H'EC10_B400/ H'EC40_B400* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI1_6_BUSIF Data Register | SSI1-6_BUSIF | R/W | H'EC10_B800/ H'EC40_B800* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI1_7_BUSIF Data Register | SSI1-7_BUSIF | R/W | H'EC10_BC00/ H'EC40_BC00* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI2_4_BUSIF Data Register | SSI2-4_BUSIF | R/W | H'EC10_C000/ H'EC40_C000* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI2_5_BUSIF Data Register | SSI2-5_BUSIF | R/W | H'EC10_C400/ H'EC40_C400* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI2_6_BUSIF Data Register | SSI2-6_BUSIF | R/W | H'EC10_C800/ H'EC40_C800* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI2_7_BUSIF Data Register | SSI2-7_BUSIF | R/W | H'EC10_CC00/ H'EC40_CC00* | H'0000_0000 | 32 | √ | √ | √ | — |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|----------------------------|--------------|-----|------------------------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI3_4_BUSIF Data Register | SSI3-4_BUSIF | R/W | H'EC10_D000/ H'EC40_D000* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI3_5_BUSIF Data Register | SSI3-5_BUSIF | R/W | H'EC10_D400/ H'EC40_D400* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI3_6_BUSIF Data Register | SSI3-6_BUSIF | R/W | H'EC10_D800/ H'EC40_D800* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI3_7_BUSIF Data Register | SSI3-7_BUSIF | R/W | H'EC10_DC00/ H'EC40_DC00* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI4_4_BUSIF Data Register | SSI4-4_BUSIF | R/W | H'EC10_E000/ H'EC40_E000* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI4_5_BUSIF Data Register | SSI4-5_BUSIF | R/W | H'EC10_E400/ H'EC40_E400* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI4_6_BUSIF Data Register | SSI4-6_BUSIF | R/W | H'EC10_E800/ H'EC40_E800* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI4_7_BUSIF Data Register | SSI4-7_BUSIF | R/W | H'EC10_EC00/ H'EC40_EC00* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI9_4_BUSIF Data Register | SSI9-4_BUSIF | R/W | H'EC10_F000/ H'EC40_F000* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI9_5_BUSIF Data Register | SSI9-5_BUSIF | R/W | H'EC10_F400/ H'EC40_F400* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI9_6_BUSIF Data Register | SSI9-6_BUSIF | R/W | H'EC10_F800/ H'EC40_F800* | H'0000_0000 | 32 | √ | √ | √ | — |
| SSI9_7_BUSIF Data Register | SSI9-7_BUSIF | R/W | H'EC10_FC00/ H'EC40_FC00* | H'0000_0000 | 32 | √ | √ | √ | — |

Note: * Address H'EC10_xxxx is used for data transfer with the Audio-DMAC. Address H'EC40_xxxx is used for data transfer with the Audio-DMACpp.

40.1.5 Connected Module**Table 40.3 Connected Module**

| Module name | Connected module name | Function of connected module |
|--------------------|------------------------------|---|
| SSIU | APMU | Access the Register |
| | PFC | Select External pins |
| | CPG | Output Clocks |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |
| | INTC-AP | Control to interrupt |
| | SSI | Serial Sound Interface |
| | ADG | Output Clocks for Audio module |
| | SCU | Sampling Rate Converter Unit |
| | Audio-DMAC | Control Direct Memory Access for Audio module |
| | Audio-DMACpp | Control Direct Memory Access for Audio modules connected to the audio local bus |

40.2 Register Description

Legend for Register Description

Initial value: Register value after a reset. H'xxxx represents a hexadecimal number. Others are represented in binary numbers.

—: Undefined value

R/W: Readable/writable. The written value can be read.

R: Read-only. The write value should always be 0.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

W: Write-only. Reading this bit is prohibited. When the bit is reserved, the write value should always be 0.

—/W: Write-only. The read value is undefined.

All access to registers is made in longword units.

40.2.1 SSIn BUSIF Mode Register (SSIn_BUSIF_MODE)

Note: n = 0-0 to 0-7, 1-0 to 1-7, 2-0 to 2-7, 3-0 to 3-7, 4-0 to 4-7, 5 to 8, or 9-0 to 9-7 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 n = 0-0 to 0-3, 1-0 to 1-3, 2-0 to 2-3, 3-0 to 3-3, 4-0 to 4-3, 5 to 8, or 9-0 to 9-3 [RZ/G2E]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSIn_BUSIF_MODE determines the initial settings of the bus interface.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----------|----|----|----|---------|---------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | sft_dir | sft_num | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | word_swap | — | — | — | — | — | — | — | dma |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 | sft_dir | B'0 | R/W | ssin_busif_shift_dir Selects the bit-shift direction for valid bit position adjustment in the SSIn_BUSIF input and output data. 0: Shift to left. 1: Shift to right. |
| 19 to 16 | sft_num | H'0 | R/W | ssin_busif_shift_num Selects the bit-shift count for valid bit position adjustment in the SSIn_BUSIF input and output data. B'0000: 0 bit B'0001: 1 bit B'0010: 2 bits B'0011: 3 bits B'0100: 4 bits B'0101: 5 bits B'0110: 6 bits B'0111: 7 bits B'1000: 8 bits B'1001: 9 bits B'1010: 10 bits B'1011: 11 bits B'1100: 12 bits B'1101: 13 bits B'1110: 14 bits B'1111: 15 bits |
| 15 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|---|
| 8 | word_swap | B'0 | R/W | word_swap_en Swaps the word order in SSI n _BUSIF. 0: Word order is not swapped. 1: Word order is swapped. Note: Select this function only when the data length is 16 bits. |
| 7 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | dma | B'1 | R/W | ssi n _dma Selects the access type for SSI n _BUSIF. 0: PIO access (setting prohibited) 1: DMA access Be sure to specify the DMA access. |

40.2.2 SSIm BUSIF Audio Information Register (SSIm_BUSIF_ADINR)

Note: m = 0-0 to 0-7, 1-0 to 1-7, 2-0 to 2-7, 3-0 to 3-7, 4-0 to 4-7, or 9-0 to 9-7 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 m = 0-0 to 0-3, 1-0 to 1-3, 2-0 to 2-3, 3-0 to 3-3, 4-0 to 4-3, or 9-0 to 9-3 [RZ/G2E]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSIm_BUSIF_ADINR is a 32-bit readable/writable register that selects channel number and bit length of output audio data.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|------------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | OTBL[4:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | CHNUM[4:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 to 16 | OTBL[4:0] | All 0 | R/W | Bit Length of Output Audio Data. These bits set the bit length of output audio data. B'0_0000: 24 bits B'0_0001: Reserved B'0_0010: 22 bits B'0_0011: Reserved B'0_0100: 20 bits B'0_0101: Reserved B'0_0110: 18 bits B'0_0111: Reserved B'0_1000: 16 bits B'0_1001 to B'0_1111: Reserved B'1_0000: 8 bits B'1_0001 to B'1_1111: Reversed |
| 15 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | CHNUM[4:0] | All 0 | R/W | Channel Number These bits set the channel number. Refer to Table 40.4 CHNUM Setting |

Table 40.4 CHNUM Setting

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| Format | | | | | |
|--------------------|--|--|---|---|--|
| SSIm_x | Basic Configuration (ex_func = 0, tdm_ext = 0, tdm_split = 0) | TDM Extend Mode (ex_func = 0, tdm_ext = 1, tdm_split = 0) | TDM Split Mode (ex_func = 0, tdm_ext = 0, tdm_split = 1) | TDM 16Ch Mode (ex_func = 1, tdm_ext = 0, tdm_split = 0) | TDM Ex-Split Mode (ex_func = 1, tdm_ext = 0, tdm_split = 1) |
| x = 0 | B'0_0000: 0 (none) | B'0_0000: 0 (none) | B'0_0000: 0 (none) | B'0_0000: 0 (none) | B'0_0000: 0 (none) |
| | B'0_0001: 1 channel | B'0_0001 to | B'0_0001: 1 channel | B'0_0001 to | B'0_0001: Reserved |
| | B'0_0010: 2 channels | B'0_0101: Reserved | B'0_0010: 2 channels | B'0_1111: Reserved | B'0_0010: 2 channels |
| | B'0_0011: Reserved | B'0_0110: 6 channels | B'0_0011 to | B'1_0000: 16 | B'0_0011: Reserved |
| | B'0_0100: 4 channels | B'0_0111: Reserved | B'1_1111: Reserved | channels | B'0_0100: 4 channels |
| | B'0_0101: Reserved | B'0_1000: 8 channels | | B'1_0001 to | B'0_0101: Reserved |
| | B'0_0110: 6 channels | B'0_1001 to | | B'1_1111: Reserved | B'0_0110: 6 channels |
| | B'0_0111: Reserved | B'1_1111: Reserved | | | B'0_0111: Reserved |
| | B'0_1000: 8 channels | | | | B'0_1000: 8 channels |
| | B'0_1001 to | | | | B'0_1001: Reserved |
| B'1_1111: Reserved | | | | B'0_1010: 10 channels B'0_1011 to B'1_1111: Reserved *2 *3 | |
| x = 1 | *1 | *1 | B'0_0000: 0 (none) | *1 | B'0_0000: 0 (none) |
| | | | B'0_0001: 1 channel | | B'0_0001: Reserved |
| | | | B'0_0010: 2 channels | | B'0_0010: 2 channels |
| | | | B'0_0011 to | | B'0_0011 to |
| | | | B'1_1111: Reserved | | B'1_1111: Reserved *3 |
| x = 2 | *1 | *1 | B'0_0000: 0 (none) | *1 | B'0_0000: 0 (none) |
| | | | B'0_0001: 1 channel | | B'0_0001: Reserved |
| | | | B'0_0010: 2 channels | | B'0_0010: 2 channels |
| | | | B'0_0011 to | | B'0_0011: Reserved |
| | | | B'1_1111: Reserved | | B'0_0100: 4 channels B'0_0101 to B'1_1111: Reserved *3 |
| x = 3 | *1 | *1 | B'0_0000: 0 (none) | *1 | B'0_0000: 0 (none) |
| | | | B'0_0001: 1 channel | | B'0_0001: Reserved |
| | | | B'0_0010: 2 channels | | B'0_0010: 2 channels |
| | | | B'0_0011 to | | B'0_0011 to |
| | | | B'1_1111: Reserved | | B'1_1111: Reserved *3 |

| Format | | | | | |
|--------|--|--|---|--|---|
| SSIm_x | Basic Configuration (ex_func = 0, tdm_ext = 0, tdm_split = 0) | TDM Extend Mode (ex_func = 0, tdm_ext = 1, tdm_split = 0) | TDM Split Mode (ex_func = 0, tdm_ext = 0, tdm_split = 1) | TDM 16Ch Mode (ex_func = 1, tdm_ext = 0, tdm_split = 0) | TDM Ex-Split Mode (ex_func = 1, tdm_ext = 0, tdm_split = 1) |
| x = 4 | *1 | *1 | *1 | *1 | B'0_0000: 0 (none) B'0_0001: Reserved B'0_0010: 2 channels B'0_0011: Reserved B'0_0100: 4 channels B'0_0101: Reserved B'0_0110: 6 channels B'0_0111: Reserved B'0_1000: 8 channels B'0_1001 to B'1_1111: Reserved *3 *4 |
| x = 5 | *1 | *1 | *1 | *1 | B'0_0000: 0 (none) B'0_0001: Reserved B'0_0010: 2 channels B'0_0011 to B'1_1111: Reserved *4 |
| x = 6 | *1 | *1 | *1 | *1 | B'0_0000: 0 (none) B'0_0001: Reserved B'0_0010: 2 channels B'0_0011: Reserved B'0_0100: 4 channels B'0_0101 to B'1_1111: Reserved *4 |
| x = 7 | *1 | *1 | *1 | *1 | B'0_0000: 0 (none) B'0_0001: Reserved B'0_0010: 2 channels B'0_0011 to B'1_1111: Reserved *4 |

- Notes:
1. The write value should always be 0.
 2. When use TDM-8ch, B'0_0111 to B'1_1111 is reserved.
 3. Setting to share place in the stream data with multiple BUSIF channels is prohibited. For example, when x = 0 is 8channel, x = 1 to 3 cannot be used and write value should always be 0
 4. When use TDM-8ch, the write value should always be 0.

[RZ/G2E]

| Format | | | | |
|--------|--|---|--|---|
| SSIm_x | Basic Configuration (ex_func = 0, tdm_ext = 0, tdm_split = 0) | TDM Extend Mode (ex_func = 0, tdm_ext = 1, tdm_split = 0) | TDM Split Mode (ex_func = 0, tdm_ext = 0, tdm_split = 1) | TDM Ex-Split Mode (ex_func = 1, tdm_ext = 0, tdm_split = 1) |
| x = 0 | B'0_0000: 0 (none) B'0_0001: 1 channel B'0_0010: 2 channels B'0_0011: Reserved B'0_0100: 4 channels B'0_0101: Reserved B'0_0110: 6 channels B'0_0111: Reserved B'0_1000: 8 channels B'0_1001 to B'1_1111: Reserved | B'0_0000: 0 (none) B'0_0001 to B'0_0101: Reserved B'0_0110: 6 channels B'0_0111: Reserved B'01000: 8 channels B'0_1001 to B'1_1111: Reserved | B'0_0000: 0 (none) B'0_0001: 1 channel B'0_0010: 2 channels B'0_0011 to B'1_1111: Reserved | B'0_0000: 0 (none) B'0_0001: Reserved B'0_0010: 2 channels B'0_0011: Reserved B'0_0100: 4 channels B'0_0101: Reserved B'0_0110: 6 channels B'0_0111 to B'1_1111: Reserved *2 |
| x = 1 | *1 | *1 | B'0_0000: 0 (none) B'0_0001: 1 channel B'0_0010: 2 channels B'0_0011 to B'1_1111: Reserved | B'0_0000: 0 (none) B'0_0001: Reserved B'0_0010: 2 channels B'0_0011 to B'1_1111: Reserved *2 |
| x = 2 | *1 | *1 | B'0_0000: 0 (none) B'0_0001: 1 channel B'0_0010: 2 channels B'0_0011 to B'1_1111: Reserved | B'0_0000: 0 (none) B'0_0001: Reserved B'0_0010: 2 channels B'0_0011: Reserved B'0_0100: 4 channels B'0_0101 to B'1_1111: Reserved *2 |
| x = 3 | *1 | *1 | B'0_0000: 0 (none) B'0_0001: 1 channel B'0_0010: 2 channels B'0_0011 to B'1_1111: Reserved | B'0_0000: 0 (none) B'0_0001: Reserved B'0_0010: 2 channels B'0_0011 to B'1_1111: Reserved *2 |

Notes: 1. The write value should always be 0.

2. Setting to share place in the stream data with multiple BUSIF channels is prohibited. For example, when x = 0 is 6channel, x = 1 to 2 cannot be used and write value should always be 0.

40.2.3 SSIR BUSIF Audio Information Register (SSIR_BUSIF_ADINR)

Note: r = 5 to 8

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSIR_BUSIF_ADINR is a 32-bit readable/writable register that selects channel number and bit length of output audio data.

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|------------|-----|-----|-----|-----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | — | OTBL[4:0] | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | — | — | — | — | — | — | CHNUM[3:0] | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | |

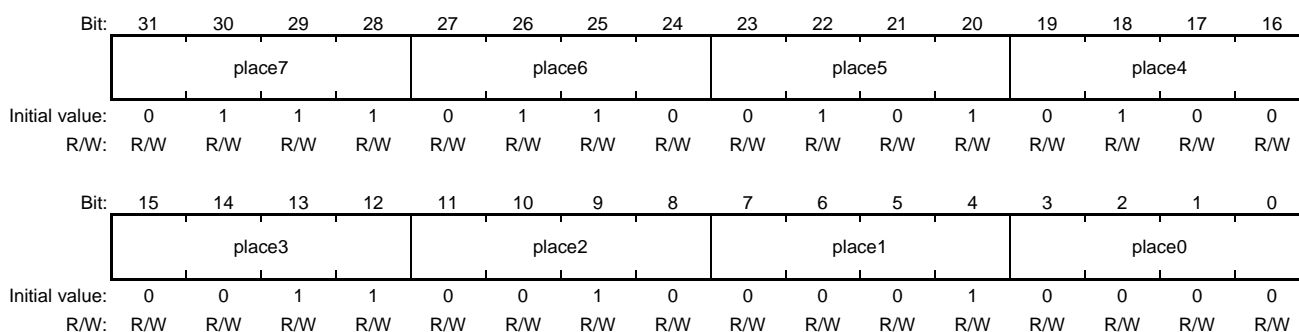
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 to 16 | OTBL[4:0] | All 0 | R/W | Bit Length of Output Audio Data. These bits set the bit length of output audio data. B'0_0000: 24 bits B'0_0001: Reserved B'0_0010: 22 bits B'0_0011: Reserved B'0_0100: 20 bits B'0_0101: Reserved B'0_0110: 18 bits B'0_0111: Reserved B'0_1000: 16 bits B'0_1001 to B'0_1111: Reserved B'1_0000: 8 bits B'1_0001 to B'1_1111: Reversed |
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | CHNUM[3:0] | H'0 | R/W | Channel Number These bits set the channel number. B'0000: 0 (none) B'0001: 1 channel B'0010: 2 channels B'0011 to B'1111: Reserved |

40.2.4 SSIq BUSIF Data Align Register (SSIq_BUSIF_DALIGN)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: q = 0-0, 1-0, 2-0, 3-0, 4-0 or 9-0

Function: SSIq_BUSIF_DALIGN determines the initial settings of the SSIq route.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 28 | place7 | B'0111 | R/W | <p>Changes the stream data order. These bits are used for the 8- or more channel setting. For the 6- or less channel setting or TDM split mode setting (tdm_split = 1, ex_func = 0), the initial value should not be changed.</p> <p>If SSIq are used for the 8channel, B'1000 to B'1111 is reserved, the initial value should not be changed.</p> <p>If SSIq are used for the 10channel, B'1010 to B'1111 is reserved, the initial value should not be changed.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 7 on the output side.</p> <p>B'0000: Data at input-side place 0 is sent to output-side place 7. B'0001: Data at input-side place 1 is sent to output-side place 7. B'0010: Data at input-side place 2 is sent to output-side place 7. B'0011: Data at input-side place 3 is sent to output-side place 7. B'0100: Data at input-side place 4 is sent to output-side place 7. B'0101: Data at input-side place 5 is sent to output-side place 7. B'0110: Data at input-side place 6 is sent to output-side place 7. B'0111: Data at input-side place 7 is sent to output-side place 7. B'1000: Data at input-side place 8 is sent to output-side place 7. B'1001: Data at input-side place 9 is sent to output-side place 7. B'1010: Data at input-side place 10 is sent to output-side place 7. B'1011: Data at input-side place 11 is sent to output-side place 7. B'1100: Data at input-side place 12 is sent to output-side place 7. B'1101: Data at input-side place 13 is sent to output-side place 7. B'1110: Data at input-side place 14 is sent to output-side place 7. B'1111: Data at input-side place 15 is sent to output-side place 7.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 27 to 24 | place6 | B'0110 | R/W | <p>Changes the stream data order. These bits are used for the 8- or more channel setting. For the 6- or less channel setting or TDM split mode setting (tdm_split = 1, ex_func = 0), the initial value should not be changed.</p> <p>When SSIq are used for the 8channel, B'1000 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 10channel, B'1010 to B'1111 is reserved, the initial value should not be changed.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 6 on the output side.</p> <p>B'0000: Data at input-side place 0 is sent to output-side place 6. B'0001: Data at input-side place 1 is sent to output-side place 6. B'0010: Data at input-side place 2 is sent to output-side place 6. B'0011: Data at input-side place 3 is sent to output-side place 6. B'0100: Data at input-side place 4 is sent to output-side place 6. B'0101: Data at input-side place 5 is sent to output-side place 6. B'0110: Data at input-side place 6 is sent to output-side place 6. B'0111: Data at input-side place 7 is sent to output-side place 6. B'1000: Data at input-side place 8 is sent to output-side place 6. B'1001: Data at input-side place 9 is sent to output-side place 6. B'1010: Data at input-side place 10 is sent to output-side place 6. B'1011: Data at input-side place 11 is sent to output-side place 6. B'1100: Data at input-side place 12 is sent to output-side place 6. B'1101: Data at input-side place 13 is sent to output-side place 6. B'1110: Data at input-side place 14 is sent to output-side place 6. B'1111: Data at input-side place 15 is sent to output-side place 6.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 23 to 20 | place5 | B'0101 | R/W | <p>Changes the stream data order. These bits are used for the 6- or more channel setting. For the 4- or less channel setting or TDM split mode setting (tdm_split = 1, ex_func = 0), the initial value should not be changed.</p> <p>When SSIq are used for the 6channel, B'0110 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 8channel, B'1000 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 10channel, B'1010 to B'1111 is reserved, the initial value should not be changed.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 5 on the output side.</p> <p>B'0000: Data at input-side place 0 is sent to output-side place 5.</p> <p>B'0001: Data at input-side place 1 is sent to output-side place 5.</p> <p>B'0010: Data at input-side place 2 is sent to output-side place 5.</p> <p>B'0011: Data at input-side place 3 is sent to output-side place 5.</p> <p>B'0100: Data at input-side place 4 is sent to output-side place 5.</p> <p>B'0101: Data at input-side place 5 is sent to output-side place 5.</p> <p>B'0110: Data at input-side place 6 is sent to output-side place 5.</p> <p>B'0111: Data at input-side place 7 is sent to output-side place 5.</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 5.</p> <p>B'1001: Data at input-side place 9 is sent to output-side place 5.</p> <p>B'1010: Data at input-side place 10 is sent to output-side place 5.</p> <p>B'1011: Data at input-side place 11 is sent to output-side place 5.</p> <p>B'1100: Data at input-side place 12 is sent to output-side place 5.</p> <p>B'1101: Data at input-side place 13 is sent to output-side place 5.</p> <p>B'1110: Data at input-side place 14 is sent to output-side place 5.</p> <p>B'1111: Data at input-side place 15 is sent to output-side place 5.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 19 to 16 | place4 | B'0100 | R/W | <p>Changes the stream data order. These bits are used for the 6- or more channel setting. For the 4- or less channel setting or TDM split mode setting (tdm_split = 1, ex_func = 0), the initial value should not be changed.</p> <p>When SSIq are used for the 6channel, B'0110 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 8channel, B'1000 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 10channel, B'1010 to B'1111 is reserved, the initial value should not be changed.</p> <p>The data order is changed between the SSI and bus interface. Selects the input-side data to be output to place 4 on the output side. B'0000: Data at input-side place 0 is sent to output-side place 4. B'0001: Data at input-side place 1 is sent to output-side place 4. B'0010: Data at input-side place 2 is sent to output-side place 4. B'0011: Data at input-side place 3 is sent to output-side place 4. B'0100: Data at input-side place 4 is sent to output-side place 4. B'0101: Data at input-side place 5 is sent to output-side place 4. B'0110: Data at input-side place 6 is sent to output-side place 4. B'0111: Data at input-side place 7 is sent to output-side place 4. B'1000: Data at input-side place 8 is sent to output-side place 4. B'1001: Data at input-side place 9 is sent to output-side place 4. B'1010: Data at input-side place 10 is sent to output-side place 4. B'1011: Data at input-side place 11 is sent to output-side place 4. B'1100: Data at input-side place 12 is sent to output-side place 4. B'1101: Data at input-side place 13 is sent to output-side place 4. B'1110: Data at input-side place 14 is sent to output-side place 4. B'1111: Data at input-side place 15 is sent to output-side place 4.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 15 to 12 | place3 | B'0011 | R/W | <p>Changes the stream data order. These bits are used for the 4- or more channel setting. For the 2- or less channel setting or TDM split mode setting (tdm_split = 1, ex_func = 0), the initial value should not be changed.</p> <p>When SSIq are used for the 4channel, B'0100 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 6channel, B'0110 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 8channel, B'1000 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 10channel, B'1010 to B'1111 is reserved, the initial value should not be changed.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 3 on the output side.</p> <p>B'0000: Data at input-side place 0 is sent to output-side place 3. B'0001: Data at input-side place 1 is sent to output-side place 3. B'0010: Data at input-side place 2 is sent to output-side place 3. B'0011: Data at input-side place 3 is sent to output-side place 3. B'0100: Data at input-side place 4 is sent to output-side place 3. B'0101: Data at input-side place 5 is sent to output-side place 3. B'0110: Data at input-side place 6 is sent to output-side place 3. B'0111: Data at input-side place 7 is sent to output-side place 3. B'1000: Data at input-side place 8 is sent to output-side place 3. B'1001: Data at input-side place 9 is sent to output-side place 3. B'1010: Data at input-side place 10 is sent to output-side place 3. B'1011: Data at input-side place 11 is sent to output-side place 3. B'1100: Data at input-side place 12 is sent to output-side place 3. B'1101: Data at input-side place 13 is sent to output-side place 3. B'1110: Data at input-side place 14 is sent to output-side place 3. B'1111: Data at input-side place 15 is sent to output-side place 3.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 11 to 8 | place2 | B'0010 | R/W | <p>Changes the stream data order. These bits are used for the 4- or more channel setting. For the 2- or less channel setting or TDM split mode setting (tdm_split = 1, ex_func = 0), the initial value should not be changed.</p> <p>When SSIq are used for the 4channel, B'0100 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 6channel, B'0110 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 8channel, B'1000 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 10channel, B'1010 to B'1111 is reserved, the initial value should not be changed.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 2 on the output side.</p> <p>B'0000: Data at input-side place 0 is sent to output-side place 2.</p> <p>B'0001: Data at input-side place 1 is sent to output-side place 2.</p> <p>B'0010: Data at input-side place 2 is sent to output-side place 2.</p> <p>B'0011: Data at input-side place 3 is sent to output-side place 2.</p> <p>B'0100: Data at input-side place 4 is sent to output-side place 2.</p> <p>B'0101: Data at input-side place 5 is sent to output-side place 2.</p> <p>B'0110: Data at input-side place 6 is sent to output-side place 2.</p> <p>B'0111: Data at input-side place 7 is sent to output-side place 2.</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 2.</p> <p>B'1001: Data at input-side place 9 is sent to output-side place 2.</p> <p>B'1010: Data at input-side place 10 is sent to output-side place 2.</p> <p>B'1011: Data at input-side place 11 is sent to output-side place 2.</p> <p>B'1100: Data at input-side place 12 is sent to output-side place 2.</p> <p>B'1101: Data at input-side place 13 is sent to output-side place 2.</p> <p>B'1110: Data at input-side place 14 is sent to output-side place 2.</p> <p>B'1111: Data at input-side place 15 is sent to output-side place 2.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 to 4 | place1 | B'0001 | R/W | <p>Changes the stream data order. The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 1 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) <ul style="list-style-type: none"> B'xxx0: Data at input-side place 0 is sent to output-side place 1. B'xxx1: Data at input-side place 1 is sent to output-side place 1. * x is don't care. When other mode is used (exclude "TDM split mode") <ul style="list-style-type: none"> B'0000: Data at input-side place 0 is sent to output-side place 1. B'0001: Data at input-side place 1 is sent to output-side place 1. B'0010: Data at input-side place 2 is sent to output-side place 1. B'0011: Data at input-side place 3 is sent to output-side place 1. B'0100: Data at input-side place 4 is sent to output-side place 1. B'0101: Data at input-side place 5 is sent to output-side place 1. B'0110: Data at input-side place 6 is sent to output-side place 1. B'0111: Data at input-side place 7 is sent to output-side place 1. B'1000: Data at input-side place 8 is sent to output-side place 1. B'1001: Data at input-side place 9 is sent to output-side place 1. B'1010: Data at input-side place 10 is sent to output-side place 1. B'1011: Data at input-side place 11 is sent to output-side place 1. B'1100: Data at input-side place 12 is sent to output-side place 1. B'1101: Data at input-side place 13 is sent to output-side place 1. B'1110: Data at input-side place 14 is sent to output-side place 1. B'1111: Data at input-side place 15 is sent to output-side place 1. <p>And, when TDM Ex-Split mode is used (tdm_split = 1, ex_func = 1), add the following condition.</p> <p>When SSIq are used for the 2channel, B'0010 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 4channel, B'0100 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 6channel, B'0110 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 8channel, B'1000 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SSIq are used for the 10channel, B'1010 to B'1111 is reserved, the initial value should not be changed.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 3 to 0 | place0 | B'0000 | R/W | <p>Changes the stream data order. The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 0 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) <ul style="list-style-type: none"> B'xxx0: Data at input-side place 0 is sent to output-side place 0. B'xxx1: Data at input-side place 1 is sent to output-side place 0. * x is don't care. When other mode is used (exclude "TDM split mode") <ul style="list-style-type: none"> B'0000: Data at input-side place 0 is sent to output-side place 0. B'0001: Data at input-side place 1 is sent to output-side place 0. B'0010: Data at input-side place 2 is sent to output-side place 0. B'0011: Data at input-side place 3 is sent to output-side place 0. B'0100: Data at input-side place 4 is sent to output-side place 0. B'0101: Data at input-side place 5 is sent to output-side place 0. B'0110: Data at input-side place 6 is sent to output-side place 0. B'0111: Data at input-side place 7 is sent to output-side place 0. B'1000: Data at input-side place 8 is sent to output-side place 0. B'1001: Data at input-side place 9 is sent to output-side place 0. B'1010: Data at input-side place 10 is sent to output-side place 0. B'1011: Data at input-side place 11 is sent to output-side place 0. B'1100: Data at input-side place 12 is sent to output-side place 0. B'1101: Data at input-side place 13 is sent to output-side place 0. B'1110: Data at input-side place 14 is sent to output-side place 0. B'1111: Data at input-side place 15 is sent to output-side place 0. <p>And, when TDM Ex-Split mode is used (tdm_split = 1, ex_func = 1), add the following condition.</p> <p>When SS1q are used for the 2channel, B'0010 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SS1q are used for the 4channel, B'0100 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SS1q are used for the 6channel, B'0110 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SS1q are used for the 8channel, B'1000 to B'1111 is reserved, the initial value should not be changed.</p> <p>When SS1q are used for the 10channel, B'1010 to B'1111 is reserved, the initial value should not be changed.</p> |

40.2.5 SSIr BUSIF Data Align Register (SSIr_BUSIF_DALIGN)

Note: r = 5 to 8

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSIr_BUSIF_DALIGN determines the initial settings of the SSIr route.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|------------|----|----|----|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | place 1 | — | — | — | place 0 |
| Initial value: | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 30 to 28 | — | B'111 | R | These bits are always read as B'111. The write value should always be B'111. |
| 27 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 26 to 24 | — | B'110 | R | These bits are always read as B'110. The write value should always be B'110. |
| 23 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 22 to 20 | — | B'101 | R | These bits are always read as B'101. The write value should always be B'101. |
| 19 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 18 to 16 | — | B'100 | R | These bits are always read as B'100. The write value should always be B'100. |
| 15 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 14 to 12 | — | B'011 | R | These bits are always read as B'011. The write value should always be B'011. |
| 11 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 10 to 8 | — | B'010 | R | These bits are always read as B'010. The write value should always be B'010. |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 to 5 | — | B'00 | R | These bits are always read as B'00. The write value should always be B'00. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 4 | place1 | B'1 | R/W | Changes the stream data order. The data order is changed between the SSI and bus interface. Selects the input-side data to be output to place 1 on the output side. 0: Data at input-side place 0 is sent to output-side place 1. 1: Data at input-side place 1 is sent to output-side place 1. |
| 3 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 2 to 1 | — | B'00 | R | These bits are always read as B'00. The write value should always be B'00. |
| 0 | place0 | B'0 | R/W | Changes the stream data order. The data order is changed between the SSI and bus interface. Selects the input-side data to be output to place 0 on the output side. 0: Data at input-side place 0 is sent to output-side place 0. 1: Data at input-side place 1 is sent to output-side place 0. |

40.2.6 SSIs BUSIF Data Align Register (SSIs_BUSIF_DALIGN)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Note: s = 0-1, 1-1, 2-1, 3-1, 4-1, 9-1

Function: SSIs_BUSIF_DALIGN determines the initial settings of the SSIs route.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|--------|-----|-----|--------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | place3 | | | place2 | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 4 | place3 | B'0011 | R/W | Changes the stream data order. The data order is changed between the SSI and bus interface. Selects the input-side data to be output to place 3 on the output side. <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) B'xxx0: Data at input-side place 2 is sent to output-side place 3. B'xxx1: Data at input-side place 3 is sent to output-side place 3. * x is don't care. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) When SSIs are used for the 2 channel setting B'0010: Data at input-side place 2 is sent to output-side place 3. B'0011: Data at input-side place 3 is sent to output-side place 3. Other settings are prohibited. Other Setting (Include SSIq are 4 to 16 channel) All bit is reserved. The initial value should not be changed. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 3 to 0 | place2 | B'0010 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 2 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) B'xxx0: Data at input-side place 2 is sent to output-side place 2. B'xxx1: Data at input-side place 3 is sent to output-side place 2. * x is don't care. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) When SSIs are used for the 2 channel setting B'0010: Data at input-side place 2 is sent to output-side place 2. B'0011: Data at input-side place 3 is sent to output-side place 2. Other settings are prohibited. Other Setting (Include SSIq are 4 to 16 channel) All bit is reserved. The initial value should not be changed |

[RZ/G2E]

Note: s = 0-1, 1-1, 2-1, 3-1, 4-1, 9-1

Function: SSIs_BUSIF_DALIGN determines the initial settings of the SSIs route.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|-----|--------|-----|----|-----|--------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | | place3 | | — | | place2 | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 7 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 6 to 4 | place3 | B'011 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 3 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) B'xx0: Data at input-side place 2 is sent to output-side place 3. B'xx1: Data at input-side place 3 is sent to output-side place 3. * x is don't care. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSIs are used for the 2 channel setting</p> <p>B'010: Data at input-side place 2 is sent to output-side place 3. B'011: Data at input-side place 3 is sent to output-side place 3. Other settings are prohibited.</p> <p>Other Setting (Include SSIq are 4 to 8 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |
| 3 | — | 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 2 to 0 | place2 | B'010 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 2 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) B'xx0: Data at input-side place 2 is sent to output-side place 2. B'xx1: Data at input-side place 3 is sent to output-side place 2. * x is don't care. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSIs are used for the 2 channel setting</p> <p>B'010: Data at input-side place 2 is sent to output-side place 2. B'011: Data at input-side place 3 is sent to output-side place 2. Other settings are prohibited.</p> <p>Other Setting (Include SSIq are 4 to 8 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |

40.2.7 SSIt BUSIF Data Align Register (SSIt_BUSIF_DALIGN)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Note: t = 0-2, 1-2, 2-2, 3-2, 4-2, 9-2

Function: SSIt_BUSIF_DALIGN determines the initial settings of the SSIt route.

| | | | | | | | | | | | | | | | | |
|----------------|--------|-----|-----|-----|--------|-----|-----|-----|--------|-----|-----|-----|--------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | place7 | | | | place6 | | | | place5 | | | | place4 | | | |
| Initial value: | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 12 | place7 | B'0111 | R/W | Changes the stream data order. The data order is changed between the SSI and bus interface. Selects the input-side data to be output to place 7 on the output side. <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) All bit is reserved. The initial value should not be changed. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) When SSIt are used for the 4 channel setting B'0100: Data at input-side place 4 is sent to output-side place 7. B'0101: Data at input-side place 5 is sent to output-side place 7. B'0110: Data at input-side place 6 is sent to output-side place 7. B'0111: Data at input-side place 7 is sent to output-side place 7. Other settings are prohibited. Other Setting (Include SSIt are 8 to 16 channel) All bit is reserved. The initial value should not be changed. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 11 to 8 | place6 | B'0110 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 6 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) All bit is reserved. The initial value should not be changed. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSI are used for the 4 channel setting</p> <p>B'0100: Data at input-side place 4 is sent to output-side place 6. B'0101: Data at input-side place 5 is sent to output-side place 6. B'0110: Data at input-side place 6 is sent to output-side place 6. B'0111: Data at input-side place 7 is sent to output-side place 6. Other settings are prohibited.</p> <p>Other Setting (Include SSI are 8 to 16 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |
| 7 to 4 | place5 | B'0101 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 5 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) B'xxx0: Data at input-side place 4 is sent to output-side place 5. B'xxx1: Data at input-side place 5 is sent to output-side place 5. * x is don't care. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSI are used for the 4 channel setting</p> <p>B'0100: Data at input-side place 4 is sent to output-side place 5. B'0101: Data at input-side place 5 is sent to output-side place 5. B'0110: Data at input-side place 6 is sent to output-side place 5. B'0111: Data at input-side place 7 is sent to output-side place 5. Other settings are prohibited.</p> <p>When SSI are used for the 2 channel setting</p> <p>B'0100: Data at input-side place 4 is sent to output-side place 5. B'0101: Data at input-side place 5 is sent to output-side place 5. Other settings are prohibited.</p> <p>Other Setting (Include SSI are 6 to 16 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 3 to 0 | place4 | B'0100 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 4 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) B'xxx0: Data at input-side place 4 is sent to output-side place 4. B'xxx1: Data at input-side place 5 is sent to output-side place 4. * x is don't care. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) When SSI are used for the 4 channel setting B'0100: Data at input-side place 4 is sent to output-side place 4. B'0101: Data at input-side place 5 is sent to output-side place 4. B'0110: Data at input-side place 6 is sent to output-side place 4. B'0111: Data at input-side place 7 is sent to output-side place 4. Other settings are prohibited. When SSI are used for the 2 channel setting B'0100: Data at input-side place 4 is sent to output-side place 4. B'0101: Data at input-side place 5 is sent to output-side place 4. Other settings are prohibited. Other Setting (Include SSI are 6 to 16 channel) All bit is reserved. The initial value should not be changed. |

[RZ/G2E]

Note: t = 0-2, 1-2, 2-2, 3-2, 4-2, 9-2

Function: SSIt_BUSIF_DALIGN determines the initial settings of the SSIt route.

| | | | | | | | | | | | | | | | | |
|----------------|----|--------|-----|-----|----|--------|-----|-----|----|--------|-----|-----|----|--------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | place7 | | | — | place6 | | | — | place5 | | | — | place4 | | |
| Initial value: | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 15 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 14 to 12 | place7 | B'111 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 7 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) <p>All bit is reserved. The initial value should not be changed.</p> <ul style="list-style-type: none"> When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSIt are used for the 4 channel setting</p> <p>B'100: Data at input-side place 4 is sent to output-side place 7. B'101: Data at input-side place 5 is sent to output-side place 7. B'110: Data at input-side place 6 is sent to output-side place 7. B'111: Data at input-side place 7 is sent to output-side place 7.</p> <p>Other settings are prohibited.</p> <p>Other Setting (Include SSIt are 8 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |
| 11 | — | B'0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 10 to 8 | place6 | B'0110 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 6 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) <p>All bit is reserved. The initial value should not be changed.</p> <ul style="list-style-type: none"> When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSIt are used for the 4 channel setting</p> <p>B'100: Data at input-side place 4 is sent to output-side place 6. B'101: Data at input-side place 5 is sent to output-side place 6. B'110: Data at input-side place 6 is sent to output-side place 6. B'111: Data at input-side place 7 is sent to output-side place 6.</p> <p>Other settings are prohibited.</p> <p>Other Setting (Include SSIt are 8 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |
| 7 | — | B'0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 6 to 4 | place5 | B'0101 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 5 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) B'xx0: Data at input-side place 4 is sent to output-side place 5. B'xx1: Data at input-side place 5 is sent to output-side place 5. * x is don't care. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSI are used for the 4 channel setting</p> <p>B'100: Data at input-side place 4 is sent to output-side place 5. B'101: Data at input-side place 5 is sent to output-side place 5. B'110: Data at input-side place 6 is sent to output-side place 5. B'111: Data at input-side place 7 is sent to output-side place 5. Other settings are prohibited.</p> <p>When SSI are used for the 2 channel setting</p> <p>B'100: Data at input-side place 4 is sent to output-side place 5. B'101: Data at input-side place 5 is sent to output-side place 5. Other settings are prohibited.</p> <p>Other Setting (Include SSI are 6 to 8 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |
| 3 | — | B'0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 2 to 0 | place4 | B'0100 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 4 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) B'xx0: Data at input-side place 4 is sent to output-side place 4. B'xx1: Data at input-side place 5 is sent to output-side place 4. * x is don't care. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSI are used for the 4 channel setting</p> <p>B'100: Data at input-side place 4 is sent to output-side place 4. B'101: Data at input-side place 5 is sent to output-side place 4. B'110: Data at input-side place 6 is sent to output-side place 4. B'111: Data at input-side place 7 is sent to output-side place 4. Other settings are prohibited.</p> <p>When SSI are used for the 2 channel setting</p> <p>B'100: Data at input-side place 4 is sent to output-side place 4. B'101: Data at input-side place 5 is sent to output-side place 4. Other settings are prohibited.</p> <p>Other Setting (Include SSI are 6 to 8 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |

40.2.8 SSlu BUSIF Data Align Register (SSlu_BUSIF_DALIGN)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Note: u = 0-3, 1-3, 2-3, 3-3, 4-3, 9-3

Function: SSlu_BUSIF_DALIGN determines the initial settings of the SSIu route.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|--------|-----|-----|--------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | place7 | | | place6 | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 4 | place7 | B'0111 | R/W | Changes the stream data order. The data order is changed between the SSI and bus interface. Selects the input-side data to be output to place 7 on the output side. <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) B'xxx0: Data at input-side place 6 is sent to output-side place 7. B'xxx1: Data at input-side place 7 is sent to output-side place 7. * x is don't care. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) if SSlu are used for the 2 channel setting B'0110: Data at input-side place 6 is sent to output-side place 7. B'0111: Data at input-side place 7 is sent to output-side place 7. Other settings are prohibited. Other Setting (Include SSli are 8 to 16 channel) All bit is reserved. The initial value should not be changed. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 3 to 0 | place6 | B'0110 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 6 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) B'xxx0: Data at input-side place 6 is sent to output-side place 6. B'xxx1: Data at input-side place 7 is sent to output-side place 6. * x is don't care. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) if SSIu are used for the 2 channel setting B'0110: Data at input-side place 6 is sent to output-side place 6. B'0111: Data at input-side place 7 is sent to output-side place 6. Other settings are prohibited. Other Setting (Include SSIq are 8 to 16 channel) All bit is reserved. The initial value should not be changed. |

[RZ/G2E]

Note: u = 0-3, 1-3, 2-3, 3-3, 4-3, 9-3

Function: SSIu_BUSIF_DALIGN determines the initial settings of the SSIu route.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|--------|-----|-----|----|--------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | place7 | | | — | place6 | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 7 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

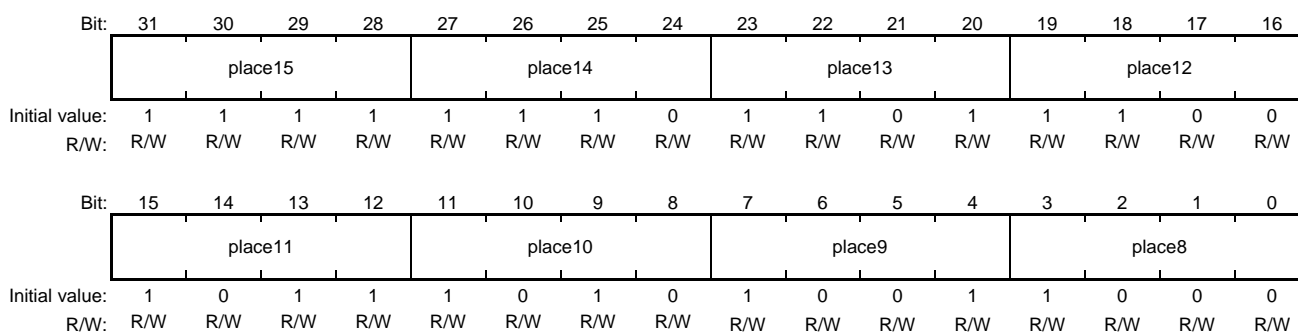
| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 6 to 4 | place7 | B'111 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 7 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) B'xx0: Data at input-side place 6 is sent to output-side place 7. B'xx1: Data at input-side place 7 is sent to output-side place 7. * x is don't care. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) if SSIu are used for the 2 channel setting B'110: Data at input-side place 6 is sent to output-side place 7. B'111: Data at input-side place 7 is sent to output-side place 7. Other settings are prohibited. Other Setting (Include SSIq are 8 channel) All bit is reserved. The initial value should not be changed. |
| 3 | — | B'0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 2 to 0 | place6 | B'110 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 6 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) B'xx0: Data at input-side place 6 is sent to output-side place 6. B'xx1: Data at input-side place 7 is sent to output-side place 6. * x is don't care. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) if SSIu are used for the 2 channel setting B'110: Data at input-side place 6 is sent to output-side place 6. B'111: Data at input-side place 7 is sent to output-side place 6. Other settings are prohibited. Other Setting (Include SSIq are 8 channel) All bit is reserved. The initial value should not be changed. |

40.2.9 SSIv BUSIF Data Align Register (SSIv_BUSIF_DALIGN)

Note: v = 0-4, 1-4, 2-4, 3-4, 4-4, 9-4

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSIv_BUSIF_DALIGN determines the initial settings of the SSIv route.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 28 | place15 | B'1111 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 15 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) <p>All bit is reserved. The initial value should not be changed.</p> <ul style="list-style-type: none"> When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSIv are used for the 8 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 15.</p> <p>B'1001: Data at input-side place 9 is sent to output-side place 15.</p> <p>B'1010: Data at input-side place 10 is sent to output-side place 15.</p> <p>B'1011: Data at input-side place 11 is sent to output-side place 15.</p> <p>B'1100: Data at input-side place 12 is sent to output-side place 15.</p> <p>B'1101: Data at input-side place 13 is sent to output-side place 15.</p> <p>B'1110: Data at input-side place 14 is sent to output-side place 15.</p> <p>B'1111: Data at input-side place 15 is sent to output-side place 15.</p> <p>Other settings are prohibited.</p> <p>Other Setting (Include SSIq are 16 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 27 to 24 | place14 | B'1110 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 14 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) <p>All bit is reserved. The initial value should not be changed.</p> <ul style="list-style-type: none"> When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSIv are used for the 8 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 14. B'1001: Data at input-side place 9 is sent to output-side place 14. B'1010: Data at input-side place 10 is sent to output-side place 14. B'1011: Data at input-side place 11 is sent to output-side place 14. B'1100: Data at input-side place 12 is sent to output-side place 14. B'1101: Data at input-side place 13 is sent to output-side place 14. B'1110: Data at input-side place 14 is sent to output-side place 14. B'1111: Data at input-side place 15 is sent to output-side place 14.</p> <p>Other settings are prohibited.</p> <p>Other Setting (Include SSIq are 16 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |
| 23 to 20 | place13 | B'1101 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 13 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) <p>All bit is reserved. The initial value should not be changed.</p> <ul style="list-style-type: none"> When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSIv are used for the 8 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 13. B'1001: Data at input-side place 9 is sent to output-side place 13. B'1010: Data at input-side place 10 is sent to output-side place 13. B'1011: Data at input-side place 11 is sent to output-side place 13. B'1100: Data at input-side place 12 is sent to output-side place 13. B'1101: Data at input-side place 13 is sent to output-side place 13. B'1110: Data at input-side place 14 is sent to output-side place 13. B'1111: Data at input-side place 15 is sent to output-side place 13.</p> <p>When SSIv are used for the 6 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 13. B'1001: Data at input-side place 9 is sent to output-side place 13. B'1010: Data at input-side place 10 is sent to output-side place 13. B'1011: Data at input-side place 11 is sent to output-side place 13. B'1100: Data at input-side place 12 is sent to output-side place 13. B'1101: Data at input-side place 13 is sent to output-side place 13.</p> <p>Other settings are prohibited.</p> <p>Other Setting (Include SSIq are 16 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 19 to 16 | place12 | B'1100 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 12 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) <p>All bit is reserved. The initial value should not be changed.</p> <ul style="list-style-type: none"> When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSIv are used for the 8 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 12. B'1001: Data at input-side place 9 is sent to output-side place 12. B'1010: Data at input-side place 10 is sent to output-side place 12. B'1011: Data at input-side place 11 is sent to output-side place 12. B'1100: Data at input-side place 12 is sent to output-side place 12. B'1101: Data at input-side place 13 is sent to output-side place 12. B'1110: Data at input-side place 14 is sent to output-side place 12. B'1111: Data at input-side place 15 is sent to output-side place 12.</p> <p>When SSIv are used for the 6 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 12. B'1001: Data at input-side place 9 is sent to output-side place 12. B'1010: Data at input-side place 10 is sent to output-side place 12. B'1011: Data at input-side place 11 is sent to output-side place 12. B'1100: Data at input-side place 12 is sent to output-side place 12. B'1101: Data at input-side place 13 is sent to output-side place 12.</p> <p>Other settings are prohibited.</p> <p>Other Setting (Include SSIq are 16 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 15 to 12 | place11 | B'1011 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 11 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) <p>All bit is reserved. The initial value should not be changed.</p> <ul style="list-style-type: none"> When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSIv are used for the 8 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 11. B'1001: Data at input-side place 9 is sent to output-side place 11. B'1010: Data at input-side place 10 is sent to output-side place 11. B'1011: Data at input-side place 11 is sent to output-side place 11. B'1100: Data at input-side place 12 is sent to output-side place 11. B'1101: Data at input-side place 13 is sent to output-side place 11. B'1110: Data at input-side place 14 is sent to output-side place 11. B'1111: Data at input-side place 15 is sent to output-side place 11.</p> <p>When SSIv are used for the 6 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 11. B'1001: Data at input-side place 9 is sent to output-side place 11. B'1010: Data at input-side place 10 is sent to output-side place 11. B'1011: Data at input-side place 11 is sent to output-side place 11. B'1100: Data at input-side place 12 is sent to output-side place 11. B'1101: Data at input-side place 13 is sent to output-side place 11. Other settings are prohibited.</p> <p>When SSIv are used for the 4 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 11. B'1001: Data at input-side place 9 is sent to output-side place 11. B'1010: Data at input-side place 10 is sent to output-side place 11. B'1011: Data at input-side place 11 is sent to output-side place 11. Other settings are prohibited.</p> <p>Other Setting (Include SSIq are 16 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 11 to 8 | place10 | B'1010 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 10 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) <p>All bit is reserved. The initial value should not be changed.</p> <ul style="list-style-type: none"> When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSIv are used for the 8 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 10. B'1001: Data at input-side place 9 is sent to output-side place 10. B'1010: Data at input-side place 10 is sent to output-side place 10. B'1011: Data at input-side place 11 is sent to output-side place 10. B'1100: Data at input-side place 12 is sent to output-side place 10. B'1101: Data at input-side place 13 is sent to output-side place 10. B'1110: Data at input-side place 14 is sent to output-side place 10. B'1111: Data at input-side place 15 is sent to output-side place 10.</p> <p>When SSIv are used for the 6 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 10. B'1001: Data at input-side place 9 is sent to output-side place 10. B'1010: Data at input-side place 10 is sent to output-side place 10. B'1011: Data at input-side place 11 is sent to output-side place 10. B'1100: Data at input-side place 12 is sent to output-side place 10. B'1101: Data at input-side place 13 is sent to output-side place 10. Other settings are prohibited.</p> <p>When SSIv are used for the 4 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 10. B'1001: Data at input-side place 9 is sent to output-side place 10. B'1010: Data at input-side place 10 is sent to output-side place 10. B'1011: Data at input-side place 11 is sent to output-side place 10. Other settings are prohibited.</p> <p>Other Setting (Include SSIq are 16 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 to 4 | place9 | B'1001 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 9 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) <p>All bit is reserved. The initial value should not be changed.</p> <ul style="list-style-type: none"> When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSIv are used for the 8 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 9. B'1001: Data at input-side place 9 is sent to output-side place 9. B'1010: Data at input-side place 10 is sent to output-side place 9. B'1011: Data at input-side place 11 is sent to output-side place 9. B'1100: Data at input-side place 12 is sent to output-side place 9. B'1101: Data at input-side place 13 is sent to output-side place 9. B'1110: Data at input-side place 14 is sent to output-side place 9. B'1111: Data at input-side place 15 is sent to output-side place 9.</p> <p>When SSIv are used for the 6 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 9. B'1001: Data at input-side place 9 is sent to output-side place 9. B'1010: Data at input-side place 10 is sent to output-side place 9. B'1011: Data at input-side place 11 is sent to output-side place 9. B'1100: Data at input-side place 12 is sent to output-side place 9. B'1101: Data at input-side place 13 is sent to output-side place 9. Other settings are prohibited.</p> <p>When SSIv are used for the 4 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 9. B'1001: Data at input-side place 9 is sent to output-side place 9. B'1010: Data at input-side place 10 is sent to output-side place 9. B'1011: Data at input-side place 11 is sent to output-side place 9. Other settings are prohibited.</p> <p>When SSIv are used for the 2 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 9. B'1001: Data at input-side place 9 is sent to output-side place 9. Other settings are prohibited.</p> <p>Other Setting (Include SSIq are 10 to 16 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 3 to 0 | place8 | B'1000 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 8 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) <p>All bit is reserved. The initial value should not be changed.</p> <ul style="list-style-type: none"> When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SSIv are used for the 8 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 8. B'1001: Data at input-side place 9 is sent to output-side place 8. B'1010: Data at input-side place 10 is sent to output-side place 8. B'1011: Data at input-side place 11 is sent to output-side place 8. B'1100: Data at input-side place 12 is sent to output-side place 8. B'1101: Data at input-side place 13 is sent to output-side place 8. B'1110: Data at input-side place 14 is sent to output-side place 8. B'1111: Data at input-side place 15 is sent to output-side place 8.</p> <p>When SSIv are used for the 6 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 8. B'1001: Data at input-side place 9 is sent to output-side place 8. B'1010: Data at input-side place 10 is sent to output-side place 8. B'1011: Data at input-side place 11 is sent to output-side place 8. B'1100: Data at input-side place 12 is sent to output-side place 8. B'1101: Data at input-side place 13 is sent to output-side place 8. Other settings are prohibited.</p> <p>When SSIv are used for the 4 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 8. B'1001: Data at input-side place 9 is sent to output-side place 8. B'1010: Data at input-side place 10 is sent to output-side place 8. B'1011: Data at input-side place 11 is sent to output-side place 8. Other settings are prohibited.</p> <p>When SSIv are used for the 2 channel setting</p> <p>B'1000: Data at input-side place 8 is sent to output-side place 8. B'1001: Data at input-side place 9 is sent to output-side place 8. Other settings are prohibited.</p> <p>Other Setting (Include SSIq are 10 to 16 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |

40.2.10 SSIw BUSIF Data Align Register (SSIw_BUSIF_DALIGN)

Note: w = 0-5, 1-5, 2-5, 3-5, 4-5, 9-5

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSIw_BUSIF_DALIGN determines the initial settings of the SSIw route.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------|-----|-----|---------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | place11 | | | place10 | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

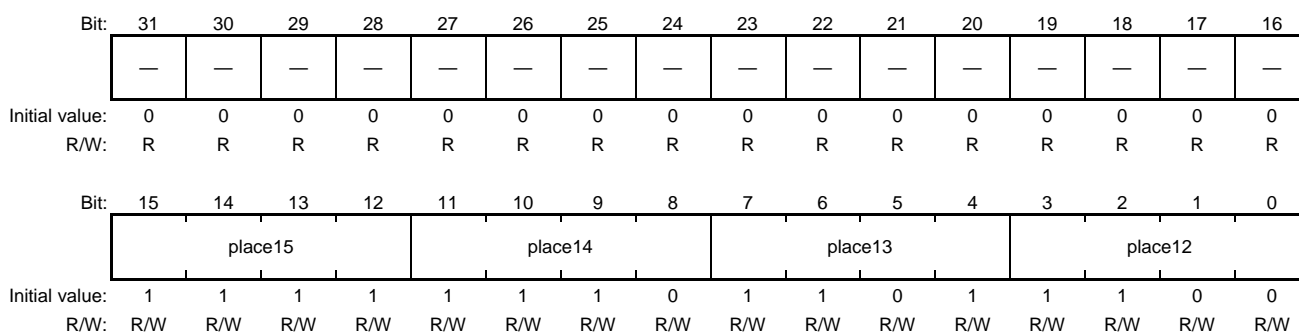
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 4 | place11 | B'1011 | R/W | Changes the stream data order. The data order is changed between the SSI and bus interface. Selects the input-side data to be output to place 11 on the output side. <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) All bit is reserved. The initial value should not be changed. When TDM Ex Func mode is used (tdm_split = 1, ex_func = 1) When SSIw are used for the 2 channel setting B'1011: Data at input-side place 11 is sent to output-side place 11. B'1010: Data at input-side place 10 is sent to output-side place 11. Other settings are prohibited. Other Setting (Include SSIq are 16 channel or SSIv are 4 to 8 channel) All bit is reserved. The initial value should not be changed. |
| 3 to 0 | place10 | B'1010 | R/W | Changes the stream data order. The data order is changed between the SSI and bus interface. Selects the input-side data to be output to place 10 on the output side. <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) All bit is reserved. The initial value should not be changed. When TDM Ex Func mode is used (tdm_split = 1, ex_func = 1) When SSIw are used for the 2 channel setting B'1011: Data at input-side place 11 is sent to output-side place 10. B'1010: Data at input-side place 10 is sent to output-side place 10. Other settings are prohibited. Other Setting (Include SSIq are 16 channel or SSIv are 4 to 8 channel) All bit is reserved. The initial value should not be changed. |

40.2.11 SSIx BUSIF Data Align Register (SSIx_BUSIF_DALIGN)

Note: x = 0-6, 1-6, 2-6, 3-6, 4-6, 9-6

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSIX_BUSIF_DALIGN determines the initial settings of the SSIX route.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 12 | place15 | B'1111 | R/W | Changes the stream data order. The data order is changed between the SSI and bus interface. Selects the input-side data to be output to place 15 on the output side. <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) All bit is reserved. The initial value should not be changed. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) When SSIX are used for the 4 channel setting B'1100: Data at input-side place 12 is sent to output-side place 15. B'1101: Data at input-side place 13 is sent to output-side place 15. B'1110: Data at input-side place 14 is sent to output-side place 15. B'1111: Data at input-side place 15 is sent to output-side place 15. Other settings are prohibited. Other Setting (Include SSIX are 16 channel or SSIX are 8 channel) All bit is reserved. The initial value should not be changed. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 11 to 8 | place14 | B'1110 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 14 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) All bit is reserved. The initial value should not be changed. When TDM Ex Func mode is used (tdm_split = 1, ex_func = 1) <p>When SS1x are used for the 4 channel setting</p> <p>B'1100: Data at input-side place 12 is sent to output-side place 14. B'1101: Data at input-side place 13 is sent to output-side place 14. B'1110: Data at input-side place 14 is sent to output-side place 14. B'1111: Data at input-side place 15 is sent to output-side place 14. Other settings are prohibited.</p> <p>Other Setting (Include SS1q are 16 channel or SS1v are 8 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |
| 7 to 4 | place13 | B'1101 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 13 on the output side.</p> <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) All bit is reserved. The initial value should not be changed. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SS1x are used for the 4 channel setting</p> <p>B'1100: Data at input-side place 12 is sent to output-side place 13. B'1101: Data at input-side place 13 is sent to output-side place 13. B'1110: Data at input-side place 14 is sent to output-side place 13. B'1111: Data at input-side place 15 is sent to output-side place 13. Other settings are prohibited.</p> <p>When SS1x are used for the 2 channel setting</p> <p>B'1100: Data at input-side place 12 is sent to output-side place 13. B'1101: Data at input-side place 13 is sent to output-side place 13. Other settings are prohibited.</p> <p>Other Setting (Include SS1q are 16 channel or SS1v are 6 to 8 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 3 to 0 | place12 | B'1100 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed between the SSI and bus interface.</p> <p>Selects the input-side data to be output to place 12 on the output side.</p> <ul style="list-style-type: none"> • When TDM split mode is used (tdm_split = 1, ex_func = 0) All bit is reserved. The initial value should not be changed. • When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) <p>When SS1x are used for the 4 channel setting</p> <p>B'1100: Data at input-side place 12 is sent to output-side place 12. B'1101: Data at input-side place 13 is sent to output-side place 12. B'1110: Data at input-side place 14 is sent to output-side place 12. B'1111: Data at input-side place 15 is sent to output-side place 12. Other settings are prohibited.</p> <p>When SS1x are used for the 2 channel setting</p> <p>B'1100: Data at input-side place 12 is sent to output-side place 12. B'1101: Data at input-side place 13 is sent to output-side place 12. Other settings are prohibited.</p> <p>Other Setting (Include SS1q are 16 channel or SS1v are 6 to 8 channel)</p> <p>All bit is reserved. The initial value should not be changed.</p> |

40.2.12 SSly BUSIF Data Align Register (SSly_BUSIF_DALIGN)

Note: y = 0-7, 1-7, 2-7, 3-7, 4-7, 9-7

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSly_BUSIF_DALIGN determines the initial settings of the SSly route.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------|-----|-----|-----|---------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | place15 | | | | place14 | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 4 | place15 | B'1111 | R/W | Changes the stream data order. The data order is changed between the SSI and bus interface. Selects the input-side data to be output to place 15 on the output side. <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) All bit is reserved. The initial value should not be changed. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) When SSly are used for the 2 channel setting B'1110: Data at input-side place 14 is sent to output-side place 15. B'1111: Data at input-side place 15 is sent to output-side place 15. Other settings are prohibited. Other Setting (Include SSly are 16 channel or SSly are 8 channel or SSly are 4 channel) All bit is reserved. The initial value should not be changed. |
| 3 to 0 | place14 | B'1110 | R/W | Changes the stream data order. The data order is changed between the SSI and bus interface. Selects the input-side data to be output to place 14 on the output side. <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) All bit is reserved. The initial value should not be changed. When TDM Ex Func mode is used (tdm_split = 1, ex_func = 1) When SSly are used for the 2 channel setting B'1110: Data at input-side place 14 is sent to output-side place 14. B'1111: Data at input-side place 15 is sent to output-side place 14. Other settings are prohibited. Other Setting (Include SSly are 16 channel or SSly are 8 channel or SSly are 4 channel) All bit is reserved. The initial value should not be changed. |

40.2.13 SSIq Mode Register (SSIq_MODE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: q = 0-0, 1-0, 2-0, 3-0, 4-0 or 9-0

Function: SSIq_MODE determines the initial settings of the SSIq route.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|---------|----|----|----|----|-----------|----|----|----|----|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | fs_mode | — | — | — | — | tdm_split | — | — | — | — | — | — | — | tdm_ext |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R | R | R | R | R/W | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 | fs_mode | B'0 | R/W | ssi i _fs_mode ($i = 0, 1, 2, 3, 4$ or 9) Selects fs to be used with TDM split mode for ssi i . <ul style="list-style-type: none"> When TDM split mode is used (tdm_split = 1, ex_func = 0) 0: ssii TDM split mode is used with 256 fs (stereo x 4). 1: ssii TDM split mode is used with 128 fs (monaural x 4). When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) The write value should always be 0, then this value should not be changed. |
| 12 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | tdm_split | B'0 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] ssi i _tdm_split ($i = 0, 1, 2, 3, 4$ or 9) Selects whether or not TDM split mode is used for ssi i . <ul style="list-style-type: none"> When TDM ex-split mode/TDM - 16ch mode Disable (ex_func = 0) 0: TDM split mode is not used for ssii (only ssii-0_BUSIF is used). 1: TDM split mode is used for ssii (ssii-1_BUSIF to ssii-3_BUSIF are also used). When TDM ex-split mode/TDM - 16ch mode Enable (ex_func = 1) 0: TDM split mode is not used for ssii (only ssii-0_BUSIF is used). 1: TDM split mode is used for ssii (ssii-1_BUSIF to ssii-7_BUSIF are also used). [RZ/G2E] ssi i _tdm_split ($i = 0, 1, 2, 3, 4$ or 9) Selects whether or not TDM split mode is used for ssi i . As for split mode and Ex Split mode as well. 0: TDM split mode is not used for ssi i (only ssi i -0_BUSIF is used). 1: TDM split mode is used for ssi i (ssi i -1_BUSIF to ssi i -3_BUSIF are also used). |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | tdm_ext | B'0 | R/W | ssi <i>i</i> _tdm_ext (<i>i</i> = 0, 1, 2, 3, 4 or 9) Selects whether or not TDM extend mode is used for ssi <i>i</i> . 0: ssi <i>i</i> TDM extend mode is not used. 1: ssi <i>i</i> TDM extend mode is used. |

Note: Bit 0 (tdm_ext) and bit 8 (tdm_split) cannot be set to 1 at the same time.

40.2.14 SSIq Control Register (SSIq_CONTROL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Note: q = 0-0, 1-0, 2-0, 3-0, 4-0 or 9-0

Function: SSIq_CONTROL controls the start and stop of data transfer.

Note: Refer to 40.4.1 Note on Transfer

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|---------|----|----|----|---------|----|----|----|---------|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | start_7 | — | — | — | start_6 | — | — | — | start_5 | — | — | — | start_4 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R/W | R | R | R | R/W | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | start_3 | — | — | — | start_2 | — | — | — | start_1 | — | — | — | start_0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R/W | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | start_7 | B'0 | R/W | SSI <i>i-7</i> _start_flag (i = 0 to 4, 9) Starts or stops data transfer through SSI <i>i-7</i> . 0: Transfer is stopped. 1: Transfer is started. When changing the corresponding bit, do not change other bits. |
| 27 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | start_6 | B'0 | R/W | SSI <i>i-6</i> _start_flag (i = 0 to 4, 9) Starts or stops data transfer through SSI <i>i-6</i> . 0: Transfer is stopped. 1: Transfer is started. When changing the corresponding bit, do not change other bits. |
| 23 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 | start_5 | B'0 | R/W | SSI <i>i-5</i> _start_flag (i = 0 to 4, 9) Starts or stops data transfer through SSI <i>i-5</i> . 0: Transfer is stopped. 1: Transfer is started. When changing the corresponding bit, do not change other bits. |
| 19 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 16 | start_4 | B'0 | R/W | SS <i>I</i> -4_start_flag (i = 0 to 4, 9) Starts or stops data transfer through SS <i>I</i> -4. 0: Transfer is stopped. 1: Transfer is started. When changing the corresponding bit, do not change other bits. |
| 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | start_3 | B'0 | R/W | SS <i>I</i> -3_start_flag (i = 0 to 4, 9) Starts or stops data transfer through SS <i>I</i> -3. 0: Transfer is stopped. 1: Transfer is started. When changing the corresponding bit, do not change other bits. |
| 11 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | start_2 | B'0 | R/W | SS <i>I</i> -2_start_flag (i = 0 to 4, 9) Starts or stops data transfer through SS <i>I</i> -2. 0: Transfer is stopped. 1: Transfer is started. When changing the corresponding bit, do not change other bits. |
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | start_1 | B'0 | R/W | SS <i>I</i> -1_start_flag (i = 0 to 4, 9) Starts or stops data transfer through SS <i>I</i> -1. 0: Transfer is stopped. 1: Transfer is started. When changing the corresponding bit, do not change other bits. |
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | start_0 | B'0 | R/W | SS <i>I</i> -0_start_flag (i = 0 to 4, 9) Starts or stops data transfer through SS <i>I</i> -0. 0: Transfer is stopped. 1: Transfer is started. When changing the corresponding bit, do not change other bits. |

Note: If only SS*I*-0 is used and TDM split mode is not used, SS*I*-1_start flag to SS*I*-7_start flag should be set to 0 (transfer stopped).

[RZ/G2E]

Note: q = 0-0, 1-0, 2-0, 3-0, 4-0 or 9-0

Function: SSIq_CONTROL controls the start and stop of data transfer.

Note: Refer to 40.4.1 Note on Transfer

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|---------|----|----|----|---------|----|----|----|---------|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | start_3 | — | — | — | start_2 | — | — | — | start_1 | — | — | — | start_0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R/W | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | start_3 | B'0 | R/W | SSI <i>i-3</i> _start_flag (i = 0 to 4, 9) Starts or stops data transfer through SSI <i>i-3</i> . 0: Transfer is stopped. 1: Transfer is started. When changing the corresponding bit, do not change other bits. |
| 11 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | start_2 | B'0 | R/W | SSI <i>i-2</i> _start_flag (i = 0 to 4, 9) Starts or stops data transfer through SSI <i>i-2</i> . 0: Transfer is stopped. 1: Transfer is started. When changing the corresponding bit, do not change other bits. |
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | start_1 | B'0 | R/W | SSI <i>i-1</i> _start_flag (i = 0 to 4, 9) Starts or stops data transfer through SSI <i>i-1</i> . 0: Transfer is stopped. 1: Transfer is started. When changing the corresponding bit, do not change other bits. |
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | start_0 | B'0 | R/W | SSI <i>i-0</i> _start_flag (i = 0 to 4, 9) Starts or stops data transfer through SSI <i>i-0</i> . 0: Transfer is stopped. 1: Transfer is started. When changing the corresponding bit, do not change other bits. |

Note: If only SSI*i-0* is used and TDM split mode is not used, SSI*i-1*_start flag to SSI*i-3*_start flag should be set to 0 (transfer stopped).

40.2.15 SSIr Control Register (SSIr_CONTROL)

Note: r = 5 to 8

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSIr_CONTROL controls the start and stop of data transfer.

Note: Refer to 40.4.1 Note on Transfer

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | start |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | start | B'0 | R/W | SSIr_start_flag Starts or stops data transfer through SSIr. 0: Transfer is stopped. 1: Transfer is started. |

40.2.16 SSIq Status Register (SSIq_STATUS)

Note: q = 0-0, 1-0, 2-0, 3-0, 4-0, 9-0

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSIq_STATUS indicates the internal buffer state. When a bit in this register is set, its status is indicated by the corresponding interrupt signal. However, when the interrupt output is masked by the SSIq Interrupt Enable Register, the interrupt signal is not output (refer to section 40.2.18 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | FCST | DTST | UIRQ | OIRQ | IIRQ | DIRQ | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | uf_3 | uf_2 | uf_1 | uf_0 | of_3 | of_2 | of_1 | of_0 | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 31, 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 | FCST | B'0 | R | SSI <i>i</i> _FCST (i = 0 to 4, 9) Indicates the state of SSIFSR <i>i</i> _FCST. 0: — 1: The WS signal has stopped. |
| 28 | DTST | B'0 | R | SSI <i>i</i> _DTST (i = 0 to 4, 9) Indicates the state of SSIFSR <i>i</i> _DTST. 0: — 1: The frequency switching has been detected. |
| 27 | UIRQ | B'0 | R | SSI <i>i</i> _UIRQ (i = 0 to 4, 9) Indicates the state of SSISR <i>i</i> _UIRQ. 0: — 1: An underflow has occurred. Note: Not indicated by the interrupt signal when SSICR <i>i</i> .UIEN = 0. |
| 26 | OIRQ | B'0 | R | SSI <i>i</i> _OIRQ (i = 0 to 4, 9) Indicates the state of SSISR <i>i</i> _OIRQ. 0: — 1: An overflow has occurred. Note: Not indicated by the interrupt signal when SSICR <i>i</i> .OIEN = 0. |
| 25 | IIRQ | B'0 | R | SSI <i>i</i> _IIRQ (i = 0 to 4, 9) Indicates the state of SSISR <i>i</i> _IIRQ. 0: — 1: Idle state Note: Not indicated by the interrupt signal when SSICR <i>i</i> .IIEN = 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 24 | DIRQ | B'0 | R | <p>SSIi_DIRQ (i = 0 to 4, 9)</p> <p>Indicates the state of SSISRi_DIRQ.</p> <p>0: —</p> <p>1: When SSICRi.TRMD is 0, DIRQ = 1 indicates that there are unread data in SSIRDR. When SSICRi.TRMD is 1, DIRQ = 1 indicates that data can be written to SSITDR.</p> <p>Note: Not indicated by the interrupt signal when SSICRi.DIEN = 0.</p> |
| 23 to 16 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 15 | uf_3 | B'0 | R | <p>buf_under_flow$i-3$ (i = 0 to 4, 9)</p> <p>Indicates the state of SSI_SYSTEM_STATUS2/3.uf$i-3$.</p> <p>0: —</p> <p>1: An underflow has occurred.</p> <p>Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE2/3.uf$i-3$_ie = 0.</p> |
| 14 | uf_2 | B'0 | R | <p>buf_under_flow$i-2$ (i = 0 to 4, 9)</p> <p>Indicates the state of SSI_SYSTEM_STATUS2/3.uf$i-2$.</p> <p>0: —</p> <p>1: An underflow has occurred.</p> <p>Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE2/3.uf$i-2$_ie = 0.</p> |
| 13 | uf_1 | B'0 | R | <p>buf_under_flow$i-1$ (i = 0 to 4, 9)</p> <p>Indicates the state of SSI_SYSTEM_STATUS2/3.uf$i-1$.</p> <p>0: —</p> <p>1: An underflow has occurred.</p> <p>Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE2/3.uf$i-1$_ie = 0.</p> |
| 12 | uf_0 | B'0 | R | <p>buf_under_flow$i-0$ (i = 0 to 4, 9)</p> <p>Indicates the state of SSI_SYSTEM_STATUS2/3.uf$i-0$.</p> <p>0: —</p> <p>1: An underflow has occurred.</p> <p>Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE2/3.uf$i-0$_ie = 0.</p> |
| 11 | of_3 | B'0 | R | <p>buf_over_flow$i-3$ (i = 0 to 4, 9)</p> <p>Indicates the state of SSI_SYSTEM_STATUS2/3.of$i-3$.</p> <p>0: —</p> <p>1: An overflow has occurred.</p> <p>Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE2/3.of$i-3$_ie = 0.</p> |
| 10 | of_2 | B'0 | R | <p>buf_over_flow$i-2$ (i = 0 to 4, 9)</p> <p>Indicates the state of SSI_SYSTEM_STATUS2/3.of$i-2$.</p> <p>0: —</p> <p>1: An overflow has occurred.</p> <p>Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE2/3.of$i-2$_ie = 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 9 | of_1 | B'0 | R | buf_over_flow <i>i</i> -1 (i = 0 to 4, 9) Indicates the state of SSI_SYSTEM_STATUS2/3.of <i>i</i> -1. 0: — 1: An overflow has occurred. Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE2/3.of <i>i</i> -1_ie = 0. |
| 8 | of_0 | B'0 | R | buf_over_flow <i>i</i> -0 (i = 0 to 4, 9) Indicates the state of SSI_SYSTEM_STATUS2/3.of <i>i</i> -0. 0: — 1: An overflow has occurred. Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE2/3.of <i>i</i> -0_ie = 0. |
| 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

40.2.17 SSIr Status Register (SSIr_STATUS)

Note: r = 5 to 8

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSIr_STATUS indicates the internal buffer state. When a bit in this register is set, its status is indicated by the corresponding interrupt signal. However, when the interrupt output is masked by the SSIr Interrupt Enable Register, the interrupt signal is not output (refer to section 40.2.19 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|------|------|------|------|------|------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | FCST | DTST | UIRQ | OIRQ | IIRQ | DIRQ | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 31, 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 | FCST | B'0 | R | SSIr _i _FCST (i = 5 to 8) Indicates the state of SSIFSR _i _FCST. 0: — 1: The WS signal has stopped. |
| 28 | DTST | B'0 | R | SSIr _i _DTST (i = 5 to 8) Indicates the state of SSIFSR _i _DTST. 0: — 1: The frequency switching has been detected. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 27 | UIRQ | B'0 | R | <p>SSI$_i$_UIRQ (i = 5 to 8)</p> <p>Indicates the state of SSISR$_i$_UIRQ.</p> <p>0: —</p> <p>1: An underflow has occurred.</p> <p>Note: Not indicated by the interrupt signal when SSICR$_i$.UIEN = 0.</p> |
| 26 | OIRQ | B'0 | R | <p>SSI$_i$_OIRQ (i = 5 to 8)</p> <p>Indicates the state of SSISR$_i$_OIRQ.</p> <p>0: —</p> <p>1: An overflow has occurred.</p> <p>Note: Not indicated by the interrupt signal when SSICR$_i$.OIEN = 0.</p> |
| 25 | IIRQ | B'0 | R | <p>SSI$_i$_IIRQ (i = 5 to 8)</p> <p>Indicates the state of SSISR$_i$_IIRQ.</p> <p>0: —</p> <p>1: Idle state</p> <p>Note: Not indicated by the interrupt signal when SSICR$_i$.IIEN = 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 24 | DIRQ | B'0 | R | <p>SSI$_i$_DIRQ (i = 5 to 8)</p> <p>Indicates the state of SSISR$_i$_DIRQ.</p> <p>0: —</p> <p>1: When SSICR$_i$.TRMD is 0, DIRQ = 1 indicates that there are unread data in SSIRD$_i$. When SSICR$_i$.TRMD is 1, DIRQ = 1 indicates that data can be written to SSITDR.</p> <p>Note: Not indicated by the interrupt signal when SSICR$_i$.DIEN = 0.</p> |
| 23 to 0 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

40.2.18 SSIIq Interrupt Enable Register (SSIIq_INT_ENABLE_MAIN)

Note: q = 0-0, 1-0, 2-0, 3-0, 4-0 or 9-0

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSIIq_INT_ENABLE_MAIN enables or disables output of interrupts corresponding to the states indicated in the SSIIq Status Register (refer to section 40.2.16 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | FCST_ ie | DTST_ ie | UIRQ_ ie | OIRQ_ ie | IIRQ_ ie | DIRQ_ ie | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | uf_3_ ie | uf_2_ ie | uf_1_ ie | uf_0_ ie | of_3_ ie | of_2_ ie | of_1_ ie | of_0_ ie | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31, 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 | FCST_ie | B'0 | R/W | SSII _i _FCST_int_enable (i = 0 to 4,9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 28 | DTST_ie | B'0 | R/W | SSII _i _DTST_int_enable (i = 0 to 4,9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 27 | UIRQ_ie | B'0 | R/W | SSII _i _UIRQ_int_enable (i = 0 to 4,9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 26 | OIRQ_ie | B'0 | R/W | SSII _i _OIRQ_int_enable (i = 0 to 4,9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 25 | IIRQ_ie | B'0 | R/W | SSII _i _IIRQ_int_enable (i = 0 to 4,9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 24 | DIRQ_ie | B'0 | R/W | SSII _i _DIRQ_int_enable (i = 0 to 4,9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 23 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | uf_3_ie | B'0 | R/W | buf_under_flow _{i-3} _int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 14 | uf_2_ie | B'0 | R/W | buf_under_flow <i>i</i> -2_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 13 | uf_1_ie | B'0 | R/W | buf_under_flow <i>i</i> -1_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 12 | uf_0_ie | B'0 | R/W | buf_under_flow <i>i</i> -0_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 11 | of_3_ie | B'0 | R/W | buf_over_flow <i>i</i> -3_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 10 | of_2_ie | B'0 | R/W | buf_over_flow <i>i</i> -2_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 9 | of_1_ie | B'0 | R/W | buf_over_flow <i>i</i> -1_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 8 | of_0_ie | B'0 | R/W | buf_over_flow <i>i</i> -0_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

40.2.19 SSIr Interrupt Enable Register (SSIr_INT_ENABLE_MAIN)

Note: r = 5 to 8

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSIr_INT_ENABLE_MAIN enables or disables output of interrupts corresponding to the states indicated in the SSIr Status Register (refer to section 40.2.17 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|-------------|-------------|-------------|-------------|-------------|-------------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | FCST_ ie | DTST_ ie | UIRQ_ ie | OIRQ_ ie | IIRQ_ ie | DIRQ_ ie | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 31, 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 | FCST_ie | B'0 | R/W | SSIr_FCST_int_enable (i = 5 to 8) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 28 | DTST_ie | B'0 | R/W | SSIr_DTST_int_enable (i = 5 to 8) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 27 | UIRQ_ie | B'0 | R/W | SSI <i>i</i> _UIRQ_int_enable (i = 5 to 8) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 26 | OIRQ_ie | B'0 | R/W | SSI <i>i</i> _OIRQ_int_enable (i = 5 to 8) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 25 | IIRQ_ie | B'0 | R/W | SSI <i>i</i> _IIRQ_int_enable (i = 5 to 8) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 24 | DIRQ_ie | B'0 | R/W | SSI <i>i</i> _DIRQ_int_enable (i = 5 to 8) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 23 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

40.2.20 SSI Mode Register 0 (SSI_MODE0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSI_MODE0 specifies the independent SSI transfer.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | ind_word_swap9 | ind_word_swap8 | ind_word_swap7 | ind_word_swap6 | ind_word_swap5 | ind_word_swap4 | ind_word_swap3 | ind_word_swap2 | ind_word_swap1 | ind_word_swap0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | ind9 | ind8 | ind7 | ind6 | ind5 | ind4 | ind3 | ind2 | ind1 | ind0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 26 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25 | ind_word_swap9 | B'0 | R/W | ind_word_swap_en9 Swaps the word order (only for access to SSITDR or SSIRDR). 0: Word order is not swapped. 1: Word order is swapped. Note: Select this function only when the data length is 16 bits. |
| 24 | ind_word_swap8 | B'0 | R/W | ind_word_swap_en8 Swaps the word order (only for access to SSITDR or SSIRDR). 0: Word order is not swapped. 1: Word order is swapped. Note: Select this function only when the data length is 16 bits. |
| 23 | ind_word_swap7 | B'0 | R/W | ind_word_swap_en7 Swaps the word order (only for access to SSITDR or SSIRDR). 0: Word order is not swapped. 1: Word order is swapped. Note: Select this function only when the data length is 16 bits. |
| 22 | ind_word_swap6 | B'0 | R/W | ind_word_swap_en6 Swaps the word order (only for access to SSITDR or SSIRDR). 0: Word order is not swapped. 1: Word order is swapped. Note: Select this function only when the data length is 16 bits. |
| 21 | ind_word_swap5 | B'0 | R/W | ind_word_swap_en5 Swaps the word order (only for access to SSITDR or SSIRDR). 0: Word order is not swapped. 1: Word order is swapped. Note: Select this function only when the data length is 16 bits. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 20 | ind_word_swap4 | B'0 | R/W | ind_word_swap_en4 Swaps the word order (only for access to SSITDR or SSIRDR). 0: Word order is not swapped. 1: Word order is swapped. Note: Select this function only when the data length is 16 bits. |
| 19 | ind_word_swap3 | B'0 | R/W | ind_word_swap_en3 Swaps the word order (only for access to SSITDR or SSIRDR). 0: Word order is not swapped. 1: Word order is swapped. Note: Select this function only when the data length is 16 bits. |
| 18 | ind_word_swap2 | B'0 | R/W | ind_word_swap_en2 Swaps the word order (only for access to SSITDR or SSIRDR). 0: Word order is not swapped. 1: Word order is swapped. Note: Select this function only when the data length is 16 bits. |
| 17 | ind_word_swap1 | B'0 | R/W | ind_word_swap_en1 Swaps the word order (only for access to SSITDR or SSIRDR). 0: Word order is not swapped. 1: Word order is swapped. Note: Select this function only when the data length is 16 bits. |
| 16 | ind_word_swap0 | B'0 | R/W | ind_word_swap_en0 Swaps the word order (only for access to SSITDR or SSIRDR). 0: Word order is not swapped. 1: Word order is swapped. Note: Select this function only when the data length is 16 bits. |
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | ind9 | B'0 | R/W | Independent SSI9 transfer setting 0: Independent SSI9 transfer is not performed. 1: Independent SSI9 transfer is performed. |
| 8 | ind8 | B'0 | R/W | Independent SSI8 transfer setting 0: Independent SSI8 transfer is not performed. 1: Independent SSI8 transfer is performed. |
| 7 | ind7 | B'0 | R/W | Independent SSI7 transfer setting 0: Independent SSI7 transfer is not performed. 1: Independent SSI7 transfer is performed. |
| 6 | ind6 | B'0 | R/W | Independent SSI6 transfer setting 0: Independent SSI6 transfer is not performed. 1: Independent SSI6 transfer is performed. |
| 5 | ind5 | B'0 | R/W | Independent SSI5 transfer setting 0: Independent SSI5 transfer is not performed. 1: Independent SSI5 transfer is performed. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 4 | ind4 | B'0 | R/W | Independent SSI4 transfer setting 0: Independent SSI4 transfer is not performed. 1: Independent SSI4 transfer is performed. |
| 3 | ind3 | B'0 | R/W | Independent SSI3 transfer setting 0: Independent SSI3 transfer is not performed. 1: Independent SSI3 transfer is performed. |
| 2 | ind2 | B'0 | R/W | Independent SSI2 transfer setting 0: Independent SSI2 transfer is not performed. 1: Independent SSI2 transfer is performed. |
| 1 | ind1 | B'0 | R/W | Independent SSI1 transfer setting 0: Independent SSI1 transfer is not performed. 1: Independent SSI1 transfer is performed. |
| 0 | ind0 | B'0 | R/W | Independent SSI0 transfer setting 0: Independent SSI0 transfer is not performed. 1: Independent SSI0 transfer is performed. |

40.2.21 SSI Mode Register 1 (SSI_MODE1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSI_MODE1 specifies the SSI pin modes.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|-------------|----------|-----|----------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | ssi34_sync | — | — | ssi4_pin | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | ssi012_3mod | ssi2_pin | | ssi1_pin | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 | ssi34_sync | B'0 | R/W | ssi34_sync_mode Selects whether to synchronize SSI3 and SSI4. 0: SSI3 and SSI4 are not synchronized. 1: SSI3 and SSI4 are synchronized. This bit can be set to 1 only when the ssi4_pin_mode bits are set to 01 or 10. |
| 19, 18 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 17, 16 | ssi4_pin | B'00 | R/W | ssi4_pin_mode Select the connections of the SSI_SCK4_A/ SSI_SCK4_B and SSI_WS4_A/ SSI_WS4_B pins. B'00: SSI3 and SSI4 use their own pins independently. B'01: The SSI3 pins are used in common by SSI3 and SSI4. Both modules work as slaves. The SSI_WS and SSI_SCK pins of SSI3 are used. B'10: The SSI3 pins are used in common by SSI3 and SSI4. SSI3 works as the master and SSI4 works as a slave. The SSI_WS and SSI_SCK pins of SSI3 are used. B'11: Setting prohibited |
| 15 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|-------------|---------------|-----|---|
| 4 | ssi012_3mod | B'0 | R/W | <p>ssi012_3module_mode</p> <p>Selects whether to use three modules (SSI0, SSI1, and SSI2) together as six channels.</p> <p>0: SSI0, SSI1, and SSI2 are not used as six channels.</p> <p>1: SSI0, SSI1, and SSI2 are used as six channels.</p> <p>This bit should be cleared to 0 when the ssi1_pin_mode or ssi2_pin_mode bits are set to 00.</p> <p>It can be set to 1 only when the ssi1_pin_mode and ssi2_pin_mode bits are set to 01 or when the ssi1_pin_mode and ssi2_pin_mode bits are set to 10.</p> <p>Either ssi012_3module_mode or ssi0129_4module_mode must be set to 1.</p> |
| 3, 2 | ssi2_pin | B'00 | R/W | <p>ssi2_pin_mode</p> <p>Select the connections of the SSI_SCK2 and SSI_WS2 pins.</p> <p>B'00: SSI0 and SSI2 use their own pins independently.</p> <p>B'01: The SSI0 pins are used in common by SSI0 and SSI2. Both modules work as slaves. The SSI_WS and SSI_SCK pins of SSI0 are used.</p> <p>B'10: The SSI0 pins are used in common by SSI0 and SSI2. SSI0 works as the master and SSI2 works as a slave. The SSI_WS and SSI_SCK pins of SSI0 are used.</p> <p>B'11: Setting prohibited</p> |
| 1, 0 | ssi1_pin | B'00 | R/W | <p>ssi1_pin_mode</p> <p>Select the connections of the SSI_SCK1 and SSI_WS1 pins.</p> <p>B'00: SSI0 and SSI1 use their own pins independently.</p> <p>B'01: The SSI0 pins are used in common by SSI0 and SSI1. Both modules work as slaves. The SSI_WS and SSI_SCK pins of SSI0 are used.</p> <p>B'10: The SSI0 pins are used in common by SSI0 and SSI1. SSI0 works as the master and SSI1 works as a slave. The SSI_WS and SSI_SCK pins of SSI0 are used.</p> <p>B'11: Setting prohibited</p> |

40.2.22 SSI Mode Register 2 (SSI_MODE2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSI_MODE2 specifies the SSI pin modes.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|--------------|----|----------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | ssi0129_4mod | — | ssi9_pin | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|--|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | ssi0129_4mod | B'0 | R/W | ssi0129_4module_mode Selects whether to use four modules (SSI0, SSI1, SSI2, and SSI9) together as eight channels. 0: SSI0, SSI1, SSI2, and SSI9 are not used as eight channels. 1: SSI0, SSI1, SSI2, and SSI9 are used as eight channels. This bit should be cleared to 0 when the ssi1_pin_mode or ssi2_pin_mode bits are set to B'00, or when ssi9_pin_mode bits are set to B'000. It can be set to 1 only when the ssi1_pin_mode and ssi2_pin_mode bits are set to 01 and the ssi9_pin_mode bits are set to B'001, or when the ssi1_pin_mode and ssi2_pin_mode bits are set to 10 and the ssi9_pin_mode bits are set to B'010. Either ssi0129_4module_mode or ssi012_3module_mode must be set to 1. |
| 3 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 2 to 0 | ssi9_pin | B'000 | R/W | <p>ssi9_pin_mode</p> <p>Selects the connections of the SSI_SCK9 and SSI_WS9 pins.</p> <p>B'000: SSI0 and SSI9 use their own pins independently.</p> <p>B'001: The SSI0 pins are used in common by SSI0 and SSI9. Both modules work as slaves. The SSI_WS and SSI_SCK pins of SSI0 are used.</p> <p>B'010: The SSI0 pins are used in common by SSI0 and SSI9. SSI0 works as the master and SSI9 works as a slave. The SSI_WS and SSI_SCK pins of SSI0 are used.</p> <p>B'011: Setting prohibited</p> <p>B'100: Setting prohibited</p> <p>B'101: The SSI3 pins are used in common by SSI3 and SSI9. Both modules work as slaves. The SSI_WS and SSI_SCK pins of SSI3 are used.</p> <p>B'110: The SSI3 pins are used in common by SSI3 and SSI9. SSI3 works as the master and SSI9 works as a slave. The SSI_WS and SSI_SCK pins of SSI3 are used.</p> <p>B'111: Setting prohibited</p> |

40.2.23 SSI Mode Register 3 (SSI_MODE3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSI_MODE3 specifies the SSI pin modes.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ssi3_pin | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1, 0 | ssi3_pin | B'00 | R/W | ssi3_pin_mode Select the connections of the SSI_SCK349 and SSI_WS349 pins. B'00: SSI0 and SSI3 use their own pins independently. B'01: The SSI0 pins are used in common by SSI0 and SSI3. Both modules work as slaves. The SSI_WS and SSI_SCK pins of SSI0 are used. B'10: The SSI0 pins are used in common by SSI0 and SSI3. SSI0 works as the master and SSI3 works as a slave. The SSI_WS and SSI_SCK pins of SSI0 are used. B'11: Setting prohibited When setting these bits to use the pins in common by SSI0 and SSI3, ssi4_pin_mode should be 00 and ssi9_pin_mode should be B'000, B'001, or B'010 (selection of the SSI3 pins is prohibited). |

40.2.24 SSI Control Register (SSI_CONTROL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: * Some restrictions apply.

Function: SSI_CONTROL controls the startup of the SSI modules when multiple SSI modules are used at the same time.

Note: Refer to 40.4.1 Note on Transfer

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|-------|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | ssi34 | — | — | — | ssi0129 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | ssi34 | B'0 | R/W | ssi34_enable Starts or stops data transfer through two SSI modules (SSI3 and SSI4) at the same time. 0: Transfer through SSI3 and SSI4 is stopped. 1: Transfer through SSI3 and SSI4 is started. When changing the corresponding bit, do not change other bits. |
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | ssi0129 | B'0 | R/W | ssi0129_enable <ul style="list-style-type: none"> When ssi012_3module_mode is 1 Starts or stops data transfer through three SSI modules (SSI0, SSI1, and SSI2) at the same time. 0: Transfer through SSI0, SSI1, and SSI2 is stopped. 1: Transfer through SSI0, SSI1, and SSI2 is started. When ssi0129_4module_mode is 1 Starts or stops data transfer through four SSI modules (SSI0, SSI1, SSI2, and SSI9) at the same time. 0: Transfer through SSI0, SSI1, SSI2, and SSI9 is stopped. 1: Transfer through SSI0, SSI1, SSI2, and SSI9 is started. When changing the corresponding bit, do not change other bits. |

Note: This function should be used only when data is transferred through multiple SSI modules in synchronization. When using each SSI module independently, use the EN bit of SSICRn (n = 0 to 4 and 9) for the SSI module to control the transfer.

40.2.25 SSI Control Register 2 (SSI_CONTROL2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSI_CONTROL2 is same function with SSI_CONTROL. This register is prepared for multi-CPU.

Note: Refer to 40.4.1 Note on Transfer

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|---------|----|----|----|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | ssi34_2 | — | — | — | ssi0129_2 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | ssi34_2 | B'0 | R/W | ssi34_enable Starts or stops data transfer through two SSI modules (SSI3 and SSI4) at the same time. 0: Transfer through SSI3 and SSI4 is stopped. 1: Transfer through SSI3 and SSI4 is started. When changing the corresponding bit, do not change other bits. |
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | ssi0129_2 | B'0 | R/W | ssi0129_enable <ul style="list-style-type: none"> When ssi012_3module_mode is 1 Starts or stops data transfer through three SSI modules (SSI0, SSI1, and SSI2) at the same time. 0: Transfer through SSI0, SSI1, and SSI2 is stopped. 1: Transfer through SSI0, SSI1, and SSI2 is started. When ssi0129_4module_mode is 1 Starts or stops data transfer through four SSI modules (SSI0, SSI1, SSI2, and SSI9) at the same time. 0: Transfer through SSI0, SSI1, SSI2, and SSI9 is stopped. 1: Transfer through SSI0, SSI1, SSI2, and SSI9 is started. When changing the corresponding bit, do not change other bits. |

Note: This function should be used only when data is transferred through multiple SSI modules in synchronization. When using each SSI module independently, use the EN bit of SSICRn (n = 0 to 4 and 9) for the SSI module to control the transfer.

40.2.26 SSI SYSTEM Status Register 0 (SSI_SYSTEM_STATUS0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSI_SYSTEM_STATUS0 indicates the internal buffer state. When a bit in this register is set, its status is indicated by the corresponding interrupt signal. However, when the interrupt output is masked by the SSI_SYSTEM Interrupt Enable Register 0, the interrupt signal is not output (refer to section 40.2.30 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | of4-3 | of4-2 | of4-1 | of4-0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 |

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | of3-3 | of3-2 | of3-1 | of3-0 | of2-3 | of2-2 | of2-1 | of2-0 | of1-3 | of1-2 | of1-1 | of1-0 | of0-3 | of0-2 | of0-1 | of0-0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|--|
| 31 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 | of4-3 | B'0 | R/WC1 | buf_over_flow4-3 Indicates the state of the SSI4-3_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 18 | of4-2 | B'0 | R/WC1 | buf_over_flow4-2 Indicates the state of the SSI4-2_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 17 | of4-1 | B'0 | R/WC1 | buf_over_flow4-1 Indicates the state of the SSI4-1_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 16 | of4-0 | B'0 | R/WC1 | buf_over_flow4-0 Indicates the state of the SSI4-0_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 15 | of3-3 | B'0 | R/WC1 | buf_over_flow3-3 Indicates the state of the SSI3-3_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 14 | of3-2 | B'0 | R/WC1 | buf_over_flow3-2 Indicates the state of the SSI3-2_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 13 | of3-1 | B'0 | R/WC1 | buf_over_flow3-1 Indicates the state of the SSI3-1_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 12 | of3-0 | B'0 | R/WC1 | buf_over_flow3-0 Indicates the state of the SSI3-0_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 11 | of2-3 | B'0 | R/WC1 | buf_over_flow2-3 Indicates the state of the SSI2-3_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 10 | of2-2 | B'0 | R/WC1 | buf_over_flow2-2 Indicates the state of the SSI2-2_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 9 | of2-1 | B'0 | R/WC1 | buf_over_flow2-1 Indicates the state of the SSI2-1_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 8 | of2-0 | B'0 | R/WC1 | buf_over_flow2-0 Indicates the state of the SSI2-0_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 7 | of1-3 | B'0 | R/WC1 | buf_over_flow1-3 Indicates the state of the SSI1-3_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 6 | of1-2 | B'0 | R/WC1 | buf_over_flow1-2 Indicates the state of the SSI1-2_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 5 | of1-1 | B'0 | R/WC1 | buf_over_flow1-1 Indicates the state of the SSI1-1_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 4 | of1-0 | B'0 | R/WC1 | buf_over_flow1-0 Indicates the state of the SSI1-0_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 3 | of0-3 | B'0 | R/WC1 | buf_over_flow0-3 Indicates the state of the SSI0-3_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 2 | of0-2 | B'0 | R/WC1 | buf_over_flow0-2 Indicates the state of the SSI0-2_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 1 | of0-1 | B'0 | R/WC1 | buf_over_flow0-1 Indicates the state of the SSI0-1_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 0 | of0-0 | B'0 | R/WC1 | buf_over_flow0-0 Indicates the state of the SSI0-0_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |

40.2.27 SSI SYSTEM Status Register 1 (SSI_SYSTEM_STATUS1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSI_SYSTEM_STATUS1 indicates the internal buffer state. When a bit in this register is set, its status is indicated by the corresponding interrupt signal. However, when the interrupt output is masked by the SSI_SYSTEM Interrupt Enable Register 1, the interrupt signal is not output (refer to section 40.2.31 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------|-------|-------|-------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | of9-3 | of9-2 | of9-1 | of9-0 | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-------|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | of9-3 | B'0 | R/WC1 | buf_over_flow9-3 Indicates the state of the SSI9-3_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 6 | of9-2 | B'0 | R/WC1 | buf_over_flow9-2 Indicates the state of the SSI9-2_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 5 | of9-1 | B'0 | R/WC1 | buf_over_flow9-1 Indicates the state of the SSI9-1_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 4 | of9-0 | B'0 | R/WC1 | buf_over_flow9-0 Indicates the state of the SSI9-0_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

40.2.28 SSI SYSTEM Status Register 2 (SSI_SYSTEM_STATUS2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSI_SYSTEM_STATUS2 indicates the internal buffer state. When a bit in this register is set, its status is indicated by the corresponding interrupt signal. However, when the interrupt output is masked by the SSI_SYSTEM Interrupt Enable Register 2, the interrupt signal is not output (refer to section 40.2.32 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | uf4-3 | uf4-2 | uf4-1 | uf4-0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | uf3-3 | uf3-2 | uf3-1 | uf3-0 | uf2-3 | uf2-2 | uf2-1 | uf2-0 | uf1-3 | uf1-2 | uf1-1 | uf1-0 | uf0-3 | uf0-2 | uf0-1 | uf0-0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|--|
| 31 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 | uf4-3 | B'0 | R/WC1 | buf_under_flow4-3 Indicates the state of the SSI4-3_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 18 | uf4-2 | B'0 | R/WC1 | buf_under_flow4-2 Indicates the state of the SSI4-2_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 17 | uf4-1 | B'0 | R/WC1 | buf_under_flow4-1 Indicates the state of the SSI4-1_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 16 | uf4-0 | B'0 | R/WC1 | buf_under_flow4-0 Indicates the state of the SSI4-0_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 15 | uf3-3 | B'0 | R/WC1 | buf_under_flow3-3 Indicates the state of the SSI3-3_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 14 | uf3-2 | B'0 | R/WC1 | buf_under_flow3-2 Indicates the state of the SSI3-2_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 13 | uf3-1 | B'0 | R/WC1 | buf_under_flow3-1 Indicates the state of the SSI3-1_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 12 | uf3-0 | B'0 | R/WC1 | buf_under_flow3-0 Indicates the state of the SSI3-0_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 11 | uf2-3 | B'0 | R/WC1 | buf_under_flow2-3 Indicates the state of the SSI2-3_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 10 | uf2-2 | B'0 | R/WC1 | buf_under_flow2-2 Indicates the state of the SSI2-2_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 9 | uf2-1 | B'0 | R/WC1 | buf_under_flow2-1 Indicates the state of the SSI2-1_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 8 | uf2-0 | B'0 | R/WC1 | buf_under_flow2-0 Indicates the state of the SSI2-0_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 7 | uf1-3 | B'0 | R/WC1 | buf_under_flow1-3 Indicates the state of the SSI1-3_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 6 | uf1-2 | B'0 | R/WC1 | buf_under_flow1-2 Indicates the state of the SSI1-2_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 5 | uf1-1 | B'0 | R/WC1 | buf_under_flow1-1 Indicates the state of the SSI1-1_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 4 | uf1-0 | B'0 | R/WC1 | buf_under_flow1-0 Indicates the state of the SSI1-0_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 3 | uf0-3 | B'0 | R/WC1 | buf_under_flow0-3 Indicates the state of the SSI0-3_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 2 | uf0-2 | B'0 | R/WC1 | buf_under_flow0-2 Indicates the state of the SSI0-2_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 1 | uf0-1 | B'0 | R/WC1 | buf_under_flow0-1 Indicates the state of the SSI0-1_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 0 | uf0-0 | B'0 | R/WC1 | buf_under_flow0-0 Indicates the state of the SSI0-0_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |

40.2.29 SSI SYSTEM Status Register 3 (SSI_SYSTEM_STATUS3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSI_SYSTEM_STATUS3 indicates the internal buffer state. When a bit in this register is set, its status is indicated by the corresponding interrupt signal. However, when the interrupt output is masked by the SSI_SYSTEM Interrupt Enable Register 3, the interrupt signal is not output (refer to section 40.2.33 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------|-------|-------|-------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | uf9-3 | uf9-2 | uf9-1 | uf9-0 | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-------|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | uf9-3 | B'0 | R/WC1 | buf_under_flow9-3 Indicates the state of the SSI9-3_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 6 | uf9-2 | B'0 | R/WC1 | buf_under_flow9-2 Indicates the state of the SSI9-2_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 5 | uf9-1 | B'0 | R/WC1 | buf_under_flow9-1 Indicates the state of the SSI9-1_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 4 | uf9-0 | B'0 | R/WC1 | buf_under_flow9-0 Indicates the state of the SSI9-0_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

40.2.30 SSI SYSTEM Interrupt Enable Register 0 (SSI_SYSTEM_INT_ENABLE0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSI_SYSTEM_INT_ENABLE0 enables or disables output of interrupts corresponding to the states indicated in the SSI SYSTEM Status Register 0 (refer to section 40.2.26 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | of4-3 _ie | of4-2 _ie | of4-1 _ie | of4-0 _ie |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | of3-3 _ie | of3-2 _ie | of3-1 _ie | of3-0 _ie | of2-3 _ie | of2-2 _ie | of2-1 _ie | of2-0 _ie | of1-3 _ie | of1-2 _ie | of1-1 _ie | of1-0 _ie | of0-3 _ie | of0-2 _ie | of0-1 _ie | of0-0 _ie |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 | of4-3_ie | B'0 | R/W | buf_over_flow4-3_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 18 | of4-2_ie | B'0 | R/W | buf_over_flow4-2_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 17 | of4-1_ie | B'0 | R/W | buf_over_flow4-1_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 16 | of4-0_ie | B'0 | R/W | buf_over_flow4-0_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 15 | of3-3_ie | B'0 | R/W | buf_over_flow3-3_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 14 | of3-2_ie | B'0 | R/W | buf_over_flow3-2_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 13 | of3-1_ie | B'0 | R/W | buf_over_flow3-1_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 12 | of3-0_ie | B'0 | R/W | buf_over_flow3-0_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 11 | of2-3_ie | B'0 | R/W | buf_over_flow2-3_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 10 | of2-2_ie | B'0 | R/W | buf_over_flow2-2_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 9 | of2-1_ie | B'0 | R/W | buf_over_flow2-1_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 8 | of2-0_ie | B'0 | R/W | buf_over_flow2-0_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 7 | of1-3_ie | B'0 | R/W | buf_over_flow1-3_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 6 | of1-2_ie | B'0 | R/W | buf_over_flow1-2_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 5 | of1-1_ie | B'0 | R/W | buf_over_flow1-1_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 4 | of1-0_ie | B'0 | R/W | buf_over_flow1-0_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 3 | of0-3_ie | B'0 | R/W | buf_over_flow0-3_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 2 | of0-2_ie | B'0 | R/W | buf_over_flow0-2_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 1 | of0-1_ie | B'0 | R/W | buf_over_flow0-1_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 0 | of0-0_ie | B'0 | R/W | buf_over_flow0-0_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

40.2.31 SSI SYSTEM Interrupt Enable Register 1 (SSI_SYSTEM_INT_ENABLE1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSI_SYSTEM_INT_ENABLE1 enables or disables output of interrupts corresponding to the states indicated in the SSI SYSTEM Status Register 1 (refer to section 40.2.27 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|--------------|--------------|--------------|--------------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | of9-3 _ie | of9-2 _ie | of9-1 _ie | of9-0 _ie | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | of9-3_ie | B'0 | R/W | buf_over_flow9-3_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 6 | of9-2_ie | B'0 | R/W | buf_over_flow9-2_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 5 | of9-1_ie | B'0 | R/W | buf_over_flow9-1_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 4 | of9-0_ie | B'0 | R/W | buf_over_flow9-0_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

40.2.32 SSI SYSTEM Interrupt Enable Register 2 (SSI_SYSTEM_INT_ENABLE2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSI_SYSTEM_INT_ENABLE2 enables or disables output of interrupts corresponding to the states indicated in the SSI SYSTEM Status Register 2 (refer to section 40.2.28 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | uf4-3_ie | uf4-2_ie | uf4-1_ie | uf4-0_ie |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | uf3-3_ie | uf3-2_ie | uf3-1_ie | uf3-0_ie | uf2-3_ie | uf2-2_ie | uf2-1_ie | uf2-0_ie | uf1-3_ie | uf1-2_ie | uf1-1_ie | uf1-0_ie | uf0-3_ie | uf0-2_ie | uf0-1_ie | uf0-0_ie |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 | uf4-3_ie | B'0 | R/W | buf_under_flow4-3_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 18 | uf4-2_ie | B'0 | R/W | buf_under_flow4-2_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 17 | uf4-1_ie | B'0 | R/W | buf_under_flow4-1_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 16 | uf4-0_ie | B'0 | R/W | buf_under_flow4-0_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 15 | uf3-3_ie | B'0 | R/W | buf_under_flow3-3_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 14 | uf3-2_ie | B'0 | R/W | buf_under_flow3-2_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 13 | uf3-1_ie | B'0 | R/W | buf_under_flow3-1_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 12 | uf3-0_ie | B'0 | R/W | buf_under_flow3-0_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 11 | uf2-3_ie | B'0 | R/W | buf_under_flow2-3_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 10 | uf2-2_ie | B'0 | R/W | buf_under_flow2-2_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 9 | uf2-1_ie | B'0 | R/W | buf_under_flow2-1_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 8 | uf2-0_ie | B'0 | R/W | buf_under_flow2-0_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 7 | uf1-3_ie | B'0 | R/W | buf_under_flow1-3_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 6 | uf1-2_ie | B'0 | R/W | buf_under_flow1-2_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 5 | uf1-1_ie | B'0 | R/W | buf_under_flow1-1_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 4 | uf1-0_ie | B'0 | R/W | buf_under_flow1-0_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 3 | uf0-3_ie | B'0 | R/W | buf_under_flow0-3_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 2 | uf0-2_ie | B'0 | R/W | buf_under_flow0-2_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 1 | uf0-1_ie | B'0 | R/W | buf_under_flow0-1_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 0 | uf0-0_ie | B'0 | R/W | buf_under_flow0-0_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

40.2.33 SSI SYSTEM Interrupt Enable Register 3 (SSI_SYSTEM_INT_ENABLE3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSI_SYSTEM_INT_ENABLE3 enables or disables output of interrupts corresponding to the states indicated in the SSI SYSTEM Status Register 3 (refer to section 40.2.29 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----------|----------|----------|----------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | uf9-3_ie | uf9-2_ie | uf9-1_ie | uf9-0_ie | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | uf9-3_ie | B'0 | R/W | buf_under_flow9-3_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 6 | uf9-2_ie | B'0 | R/W | buf_under_flow9-2_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 5 | uf9-1_ie | B'0 | R/W | buf_under_flow9-1_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 4 | uf9-0_ie | B'0 | R/W | buf_under_flow9-0_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

40.2.34 SSI SYSTEM Status Register 4 (SSI_SYSTEM_STATUS4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSI_SYSTEM_STATUS4 indicates the internal buffer state. When a bit in this register is set, its status is indicated by the corresponding interrupt signal. However, when the interrupt output is masked by the SSI_SYSTEM Interrupt Enable Register 4, the interrupt signal is not output (refer to section 40.2.38 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | of4-7 | of4-6 | of4-5 | of4-4 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | of3-7 | of3-6 | of3-5 | of3-4 | of2-7 | of2-6 | of2-5 | of2-4 | of1-7 | of1-6 | of1-5 | of1-4 | of0-7 | of0-6 | of0-5 | of0-4 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|--|
| 31 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 | of4-7 | B'0 | R/WC1 | buf_over_flow4-7 Indicates the state of the SSI4-7_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 18 | of4-6 | B'0 | R/WC1 | buf_over_flow4-6 Indicates the state of the SSI4-6_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 17 | of4-5 | B'0 | R/WC1 | buf_over_flow4-5 Indicates the state of the SSI4-5_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 16 | of4-4 | B'0 | R/WC1 | buf_over_flow4-4 Indicates the state of the SSI4-4_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 15 | of3-7 | B'0 | R/WC1 | buf_over_flow3-7 Indicates the state of the SSI3-7_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 14 | of3-6 | B'0 | R/WC1 | buf_over_flow3-6 Indicates the state of the SSI3-6_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 13 | of3-5 | B'0 | R/WC1 | buf_over_flow3-5 Indicates the state of the SSI3-5_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 12 | of3-4 | B'0 | R/WC1 | buf_over_flow3-4 Indicates the state of the SSI3-4_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 11 | of2-7 | B'0 | R/WC1 | buf_over_flow2-7 Indicates the state of the SSI2-7_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 10 | of2-6 | B'0 | R/WC1 | buf_over_flow2-6 Indicates the state of the SSI2-6_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 9 | of2-5 | B'0 | R/WC1 | buf_over_flow2-5 Indicates the state of the SSI2-5_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 8 | of2-4 | B'0 | R/WC1 | buf_over_flow2-4 Indicates the state of the SSI2-4_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 7 | of1-7 | B'0 | R/WC1 | buf_over_flow1-7 Indicates the state of the SSI1-7_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 6 | of1-6 | B'0 | R/WC1 | buf_over_flow1-6 Indicates the state of the SSI1-6_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 5 | of1-5 | B'0 | R/WC1 | buf_over_flow1-5 Indicates the state of the SSI1-5_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 4 | of1-4 | B'0 | R/WC1 | buf_over_flow1-4 Indicates the state of the SSI1-4_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 3 | of0-7 | B'0 | R/WC1 | buf_over_flow0-7 Indicates the state of the SSI0-7_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 2 | of0-6 | B'0 | R/WC1 | buf_over_flow0-6 Indicates the state of the SSI0-6_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 1 | of0-5 | B'0 | R/WC1 | buf_over_flow0-5 Indicates the state of the SSI0-5_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 0 | of0-4 | B'0 | R/WC1 | buf_over_flow0-4 Indicates the state of the SSI0-4_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |

40.2.35 SSI SYSTEM Status Register 5 (SSI_SYSTEM_STATUS5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSI_SYSTEM_STATUS5 indicates the internal buffer state. When a bit in this register is set, its status is indicated by the corresponding interrupt signal. However, when the interrupt output is masked by the SSI_SYSTEM Interrupt Enable Register 5, the interrupt signal is not output (refer to section 40.2.39 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------|-------|-------|-------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | of9-7 | of9-6 | of9-5 | of9-4 | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-------|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | of9-7 | B'0 | R/WC1 | buf_over_flow9-7 Indicates the state of the SSI9-7_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 6 | of9-6 | B'0 | R/WC1 | buf_over_flow9-6 Indicates the state of the SSI9-6_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 5 | of9-5 | B'0 | R/WC1 | buf_over_flow9-5 Indicates the state of the SSI9-5_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 4 | of9-4 | B'0 | R/WC1 | buf_over_flow9-4 Indicates the state of the SSI9-4_BUSIF buffer. 0: Normal operation 1: An overflow has occurred. |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

40.2.36 SSI SYSTEM Status Register 6 (SSI_SYSTEM_STATUS6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSI_SYSTEM_STATUS6 indicates the internal buffer state. When a bit in this register is set, its status is indicated by the corresponding interrupt signal. However, when the interrupt output is masked by the SSI_SYSTEM Interrupt Enable Register 6, the interrupt signal is not output (refer to section 40.2.40 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | uf4-7 | uf4-6 | uf4-5 | uf4-4 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 |

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | uf3-7 | uf3-6 | uf3-5 | uf3-4 | uf2-7 | uf2-6 | uf2-5 | uf2-4 | uf1-7 | uf1-6 | uf1-5 | uf1-4 | uf0-7 | uf0-6 | uf0-5 | uf0-4 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|--|
| 31 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 | uf4-7 | B'0 | R/WC1 | buf_under_flow4-7 Indicates the state of the SSI4-7_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 18 | uf4-6 | B'0 | R/WC1 | buf_under_flow4-6 Indicates the state of the SSI4-6_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 17 | uf4-5 | B'0 | R/WC1 | buf_under_flow4-5 Indicates the state of the SSI4-5_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 16 | uf4-4 | B'0 | R/WC1 | buf_under_flow4-4 Indicates the state of the SSI4-4_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 15 | uf3-7 | B'0 | R/WC1 | buf_under_flow3-7 Indicates the state of the SSI3-7_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 14 | uf3-6 | B'0 | R/WC1 | buf_under_flow3-6 Indicates the state of the SSI3-6_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 13 | uf3-5 | B'0 | R/WC1 | buf_under_flow3-5 Indicates the state of the SSI3-5_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 12 | uf3-4 | B'0 | R/WC1 | buf_under_flow3-4 Indicates the state of the SSI3-4_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 11 | uf2-7 | B'0 | R/WC1 | buf_under_flow2-7 Indicates the state of the SSI2-7_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 10 | uf2-6 | B'0 | R/WC1 | buf_under_flow2-6 Indicates the state of the SSI2-6_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 9 | uf2-5 | B'0 | R/WC1 | buf_under_flow2-5 Indicates the state of the SSI2-5_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 8 | uf2-4 | B'0 | R/WC1 | buf_under_flow2-4 Indicates the state of the SSI2-4_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 7 | uf1-7 | B'0 | R/WC1 | buf_under_flow1-7 Indicates the state of the SSI1-7_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 6 | uf1-6 | B'0 | R/WC1 | buf_under_flow1-6 Indicates the state of the SSI1-6_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 5 | uf1-5 | B'0 | R/WC1 | buf_under_flow1-5 Indicates the state of the SSI1-5_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 4 | uf1-4 | B'0 | R/WC1 | buf_under_flow1-4 Indicates the state of the SSI1-4_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 3 | uf0-7 | B'0 | R/WC1 | buf_under_flow0-7 Indicates the state of the SSI0-7_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 2 | uf0-6 | B'0 | R/WC1 | buf_under_flow0-6 Indicates the state of the SSI0-6_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 1 | uf0-5 | B'0 | R/WC1 | buf_under_flow0-5 Indicates the state of the SSI0-5_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 0 | uf0-4 | B'0 | R/WC1 | buf_under_flow0-4 Indicates the state of the SSI0-4_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |

40.2.37 SSI SYSTEM Status Register 7 (SSI_SYSTEM_STATUS7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSI_SYSTEM_STATUS7 indicates the internal buffer state. When a bit in this register is set, its status is indicated by the corresponding interrupt signal. However, when the interrupt output is masked by the SSI_SYSTEM Interrupt Enable Register 7, the interrupt signal is not output (refer to section 40.2.41 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------|-------|-------|-------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | uf9-7 | uf9-6 | uf9-5 | uf9-4 | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-------|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | uf9-7 | B'0 | R/WC1 | buf_under_flow9-7 Indicates the state of the SSI9-7_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 6 | uf9-6 | B'0 | R/WC1 | buf_under_flow9-6 Indicates the state of the SSI9-6_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 5 | uf9-5 | B'0 | R/WC1 | buf_under_flow9-5 Indicates the state of the SSI9-5_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 4 | uf9-4 | B'0 | R/WC1 | buf_under_flow9-4 Indicates the state of the SSI9-4_BUSIF buffer. 0: Normal operation 1: An underflow has occurred. |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

40.2.38 SSI SYSTEM Interrupt Enable Register 4 (SSI_SYSTEM_INT_ENABLE4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSI_SYSTEM_INT_ENABLE4 enables or disables output of interrupts corresponding to the states indicated in the SSI SYSTEM Status Register 4 (refer to section 40.2.34 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | of4-7 _ie | of4-6 _ie | of4-5 _ie | of4-4 _ie |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | of3-7 _ie | of3-6 _ie | of3-5 _ie | of3-4 _ie | of2-7 _ie | of2-6 _ie | of2-5 _ie | of2-4 _ie | of1-7 _ie | of1-6 _ie | of1-5 _ie | of1-4 _ie | of0-7 _ie | of0-6 _ie | of0-5 _ie | of0-4 _ie |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 | of4-7_ie | B'0 | R/W | buf_over_flow4-7_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 18 | of4-6_ie | B'0 | R/W | buf_over_flow4-6_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 17 | of4-5_ie | B'0 | R/W | buf_over_flow4-5_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 16 | of4-4_ie | B'0 | R/W | buf_over_flow4-4_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 15 | of3-7_ie | B'0 | R/W | buf_over_flow3-7_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 14 | of3-6_ie | B'0 | R/W | buf_over_flow3-6_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 13 | of3-5_ie | B'0 | R/W | buf_over_flow3-5_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 12 | of3-4_ie | B'0 | R/W | buf_over_flow3-4_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 11 | of2-7_ie | B'0 | R/W | buf_over_flow2-7_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 10 | of2-6_ie | B'0 | R/W | buf_over_flow2-6_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 9 | of2-5_ie | B'0 | R/W | buf_over_flow2-5_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 8 | of2-4_ie | B'0 | R/W | buf_over_flow2-4_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 7 | of1-7_ie | B'0 | R/W | buf_over_flow1-7_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 6 | of1-6_ie | B'0 | R/W | buf_over_flow1-6_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 5 | of1-5_ie | B'0 | R/W | buf_over_flow1-5_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 4 | of1-4_ie | B'0 | R/W | buf_over_flow1-4_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 3 | of0-7_ie | B'0 | R/W | buf_over_flow0-7_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 2 | of0-6_ie | B'0 | R/W | buf_over_flow0-6_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 1 | of0-5_ie | B'0 | R/W | buf_over_flow0-5_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 0 | of0-4_ie | B'0 | R/W | buf_over_flow0-4_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

40.2.39 SSI SYSTEM Interrupt Enable Register 5 (SSI_SYSTEM_INT_ENABLE5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSI_SYSTEM_INT_ENABLE5 enables or disables output of interrupts corresponding to the states indicated in the SSI SYSTEM Status Register 5 (refer to section 40.2.35 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|--------------|--------------|--------------|--------------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | of9-7 _ie | of9-6 _ie | of9-5 _ie | of9-4 _ie | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | of9-7_ie | B'0 | R/W | buf_over_flow9-7_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 6 | of9-6_ie | B'0 | R/W | buf_over_flow9-6_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 5 | of9-5_ie | B'0 | R/W | buf_over_flow9-5_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 4 | of9-4_ie | B'0 | R/W | buf_over_flow9-4_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

40.2.40 SSI SYSTEM Interrupt Enable Register 6 (SSI_SYSTEM_INT_ENABLE6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSI_SYSTEM_INT_ENABLE6 enables or disables output of interrupts corresponding to the states indicated in the SSI SYSTEM Status Register 6 (refer to section 40.2.36 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | uf4-7_ie | uf4-6_ie | uf4-5_ie | uf4-4_ie |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | uf3-7_ie | uf3-6_ie | uf3-5_ie | uf3-4_ie | uf2-7_ie | uf2-6_ie | uf2-5_ie | uf2-4_ie | uf1-7_ie | uf1-6_ie | uf1-5_ie | uf1-4_ie | uf0-7_ie | uf0-6_ie | uf0-5_ie | uf0-4_ie |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 | uf4-7_ie | B'0 | R/W | buf_under_flow4-7_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 18 | uf4-6_ie | B'0 | R/W | buf_under_flow4-6_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 17 | uf4-5_ie | B'0 | R/W | buf_under_flow4-5_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 16 | uf4-4_ie | B'0 | R/W | buf_under_flow4-4_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 15 | uf3-7_ie | B'0 | R/W | buf_under_flow3-7_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 14 | uf3-6_ie | B'0 | R/W | buf_under_flow3-6_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 13 | uf3-5_ie | B'0 | R/W | buf_under_flow3-5_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 12 | uf3-4_ie | B'0 | R/W | buf_under_flow3-4_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 11 | uf2-7_ie | B'0 | R/W | buf_under_flow2-7_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 10 | uf2-6_ie | B'0 | R/W | buf_under_flow2-6_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 9 | uf2-5_ie | B'0 | R/W | buf_under_flow2-5_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 8 | uf2-4_ie | B'0 | R/W | buf_under_flow2-4_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 7 | uf1-7_ie | B'0 | R/W | buf_under_flow1-7_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 6 | uf1-6_ie | B'0 | R/W | buf_under_flow1-6_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 5 | uf1-5_ie | B'0 | R/W | buf_under_flow1-5_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 4 | uf1-4_ie | B'0 | R/W | buf_under_flow1-4_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 3 | uf0-7_ie | B'0 | R/W | buf_under_flow0-7_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 2 | uf0-6_ie | B'0 | R/W | buf_under_flow0-6_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 1 | uf0-5_ie | B'0 | R/W | buf_under_flow0-5_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 0 | uf0-4_ie | B'0 | R/W | buf_under_flow0-4_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

40.2.41 SSI SYSTEM Interrupt Enable Register 7 (SSI_SYSTEM_INT_ENABLE7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSI_SYSTEM_INT_ENABLE7 enables or disables output of interrupts corresponding to the states indicated in the SSI SYSTEM Status Register 7 (refer to section 40.2.37 and section 40.3.14).

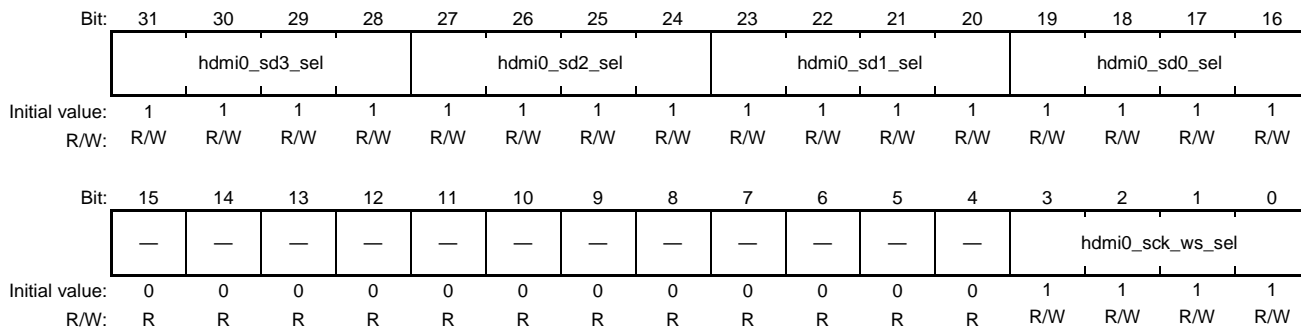
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|--------------|--------------|--------------|--------------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | uf9-7 _ie | uf9-6 _ie | uf9-5 _ie | uf9-4 _ie | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | uf9-7_ie | B'0 | R/W | buf_under_flow9-7_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 6 | uf9-6_ie | B'0 | R/W | buf_under_flow9-6_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 5 | uf9-5_ie | B'0 | R/W | buf_under_flow9-5_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 4 | uf9-4_ie | B'0 | R/W | buf_under_flow9-4_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

40.2.42 HDMI0 Select Register (HDMI0_SEL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: Select SSI signal to HDMI0. This function can use only SSI format is master/stereo transmit mode. If select slave SSI, it is necessary to connect to other master SSI by pin connect function (refer to section 40.3.12).



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 28 | hdmi0_sd3_sel | B'1111 | R/W | select sdata3 to HDMI0 B'0000: SSI0_SDATA (*1) B'0001: SSI1_SDATA (*2) B'0010: SSI2_SDATA (*3) B'0011: SSI3_SDATA (*4) B'0100: SSI4_SDATA (*5) B'0101: SSI5_SDATA (*6) B'0110: SSI6_SDATA (*7) B'0111: SSI7_SDATA (*8) B'1000: SSI8_SDATA (*9) B'1001: SSI9_SDATA (*10) B'1010 to B'1110: Reserved B'1111: Output Fix0 |
| 27 to 24 | hdmi0_sd2_sel | B'1111 | R/W | select sdata2 to HDMI0 B'0000: SSI0_SDATA (*1) B'0001: SSI1_SDATA (*2) B'0010: SSI2_SDATA (*3) B'0011: SSI3_SDATA (*4) B'0100: SSI4_SDATA (*5) B'0101: SSI5_SDATA (*6) B'0110: SSI6_SDATA (*7) B'0111: SSI7_SDATA (*8) B'1000: SSI8_SDATA (*9) B'1001: SSI9_SDATA (*10) B'1010 to B'1110: Reserved B'1111: Output Fix0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|---|
| 23 to 20 | hdmi0_sd1_sel | B'1111 | R/W | select sdata1 to HDMI0 B'0000: SSI0_SDATA (*1) B'0001: SSI1_SDATA (*2) B'0010: SSI2_SDATA (*3) B'0011: SSI3_SDATA (*4) B'0100: SSI4_SDATA (*5) B'0101: SSI5_SDATA (*6) B'0110: SSI6_SDATA (*7) B'0111: SSI7_SDATA (*8) B'1000: SSI8_SDATA (*9) B'1001: SSI9_SDATA (*10) B'1010 to B'1110: Reserved B'1111: Output Fix0 |
| 19 to 16 | hdmi0_sd0_sel | B'1111 | R/W | select sdata0 to HDMI0 B'0000: SSI0_SDATA (*1) B'0001: SSI1_SDATA (*2) B'0010: SSI2_SDATA (*3) B'0011: SSI3_SDATA (*4) B'0100: SSI4_SDATA (*5) B'0101: SSI5_SDATA (*6) B'0110: SSI6_SDATA (*7) B'0111: SSI7_SDATA (*8) B'1000: SSI8_SDATA (*9) B'1001: SSI9_SDATA (*10) B'1010 to B'1110: Reserved B'1111: Output Fix0 |
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | hdmi0_sck_ws_sel | B'1111 | R/W | select sck and ws to HDMI0 B'0000: SSI0_SCK/SSI0_WS B'0001: SSI1_SCK/SSI1_WS B'0010: SSI2_SCK/SSI2_WS B'0011: SSI3_SCK/SSI3_WS B'0100: SSI4_SCK/SSI4_WS B'0101: SSI5_SCK/SSI5_WS B'0110: SSI6_SCK/SSI6_WS B'0111: SSI7_SCK/SSI7_WS B'1000: Reserved B'1001: SSI9_SCK/SSI9_WS B'1010 to B'1110: Reserved B'1111: Output Fix0 |

- Notes:
1. In case that hdmi0_sck_ws_sel = B'0000 and SSI0 runs as master transmitter.
 2. In case that hdmi0_sck_ws_sel = B'0001 and SSI1 runs as master transmitter, or in case that hdmi0_sck_ws_sel = B'0000 and SSI0 runs as master transmitter and SSI1 runs as slave transmitter.
 3. In case that hdmi0_sck_ws_sel = B'0010 and SSI2 runs as master transmitter, or in case that hdmi0_sck_ws_sel = B'0000 and SSI0 runs as master transmitter and SSI2 runs as slave transmitter.
 4. In case that hdmi0_sck_ws_sel = B'0011 and SSI3 runs as master transmitter, or in case that hdmi0_sck_ws_sel = B'0000 and SSI0 runs as master transmitter and SSI3 runs as slave transmitter.

5. In case that `hdmi0_sck_ws_sel = B'0100` and SSI4 runs as master transmitter, or in case that `hdmi0_sck_ws_sel = B'0011` and SSI3 runs as master transmitter and SSI4 runs as slave transmitter.
6. In case that `hdmi0_sck_ws_sel = B'0101` and SSI5 runs as master transmitter.
7. In case that `hdmi0_sck_ws_sel = B'0110` and SSI6 runs as master transmitter.
8. In case that `hdmi0_sck_ws_sel = B'0111` and SSI7 runs as master transmitter.
9. In case that `hdmi0_sck_ws_sel = B'0111` and SSI7 runs as master transmitter and SSI8 runs as slave transmitter.
10. In case that `hdmi0_sck_ws_sel = B'1001` and SSI9 runs as master transmitter, or in case that `hdmi0_sck_ws_sel = B'0000` and SSI0 runs as master transmitter and SSI9 runs as slave transmitter or in case that `hdmi0_sck_ws_sel = B'0011` and SSI3 runs as master transmitter and SSI9 runs as slave transmitter.

40.2.43 HDMI1 Select Register (HDMI1_SEL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | — |

Function: Select SSI signal to HDMI1. This function can use only SSI format is master/stereo transmit mode. If select slave SSI, it is necessary to connect other master SSI by pin connect function (refer to section 40.3.12).

| | | | | | | | | | | | | | | | | |
|----------------|---------------|-----|-----|-----|---------------|-----|-----|-----|---------------|-----|-----|-----|------------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | hdmi1_sd3_sel | | | | hdmi1_sd2_sel | | | | hdmi1_sd1_sel | | | | hdmi1_sd0_sel | | | |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | | | | | | | | | | | | hdmi1_sck_ws_sel | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 28 | hdmi1_sd3_sel | B'1111 | R/W | select sdata3 to HDMI1 B'0000: SSI0_SDATA (*1) B'0001: SSI1_SDATA (*2) B'0010: SSI2_SDATA (*3) B'0011: SSI3_SDATA (*4) B'0100: SSI4_SDATA (*5) B'0101: SSI5_SDATA (*6) B'0110: SSI6_SDATA (*7) B'0111: SSI7_SDATA (*8) B'1000: SSI8_SDATA (*9) B'1001: SSI9_SDATA (*10) B'1010 to B'1110: Reserved B'1111: Output Fix0 |
| 27 to 24 | hdmi1_sd2_sel | B'1111 | R/W | select sdata2 to HDMI1 B'0000: SSI0_SDATA (*1) B'0001: SSI1_SDATA (*2) B'0010: SSI2_SDATA (*3) B'0011: SSI3_SDATA (*4) B'0100: SSI4_SDATA (*5) B'0101: SSI5_SDATA (*6) B'0110: SSI6_SDATA (*7) B'0111: SSI7_SDATA (*8) B'1000: SSI8_SDATA (*9) B'1001: SSI9_SDATA (*10) B'1010 to B'1110: Reserved B'1111: Output Fix0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|---|
| 23 to 20 | hdmi1_sd1_sel | B'1111 | R/W | select sdata1 to HDMI1 B'0000: SSI0_SDATA (*1) B'0001: SSI1_SDATA (*2) B'0010: SSI2_SDATA (*3) B'0011: SSI3_SDATA (*4) B'0100: SSI4_SDATA (*5) B'0101: SSI5_SDATA (*6) B'0110: SSI6_SDATA (*7) B'0111: SSI7_SDATA (*8) B'1000: SSI8_SDATA (*9) B'1001: SSI9_SDATA (*10) B'1010 to B'1110: Reserved B'1111: Output Fix0 |
| 19 to 16 | hdmi1_sd0_sel | B'1111 | R/W | select sdata0 to HDMI1 B'0000: SSI0_SDATA (*1) B'0001: SSI1_SDATA (*2) B'0010: SSI2_SDATA (*3) B'0011: SSI3_SDATA (*4) B'0100: SSI4_SDATA (*5) B'0101: SSI5_SDATA (*6) B'0110: SSI6_SDATA (*7) B'0111: SSI7_SDATA (*8) B'1000: SSI8_SDATA (*9) B'1001: SSI9_SDATA (*10) B'1010 to B'1110: Reserved B'1111: Output Fix0 |
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | hdmi1_sck_ws_sel | B'1111 | R/W | select sck and ws to HDMI1 B'0000: SSI0_SCK/SSI0_WS B'0001: SSI1_SCK/SSI1_WS B'0010: SSI2_SCK/SSI2_WS B'0011: SSI3_SCK/SSI3_WS B'0100: SSI4_SCK/SSI4_WS B'0101: SSI5_SCK/SSI5_WS B'0110: SSI6_SCK/SSI6_WS B'0111: SSI7_SCK/SSI7_WS B'1000: Reserved B'1001: SSI9_SCK/SSI9_WS B'1010 to B'1110: Reserved B'1111: Output Fix0 |

- Notes:
1. In case that hdmi0_sck_ws_sel = B'0000 and SSI0 runs as master transmitter.
 2. In case that hdmi0_sck_ws_sel = B'0001 and SSI1 runs as master transmitter, or in case that hdmi0_sck_ws_sel = B'0000 and SSI0 runs as master transmitter and SSI1 runs as slave transmitter.
 3. In case that hdmi0_sck_ws_sel = B'0010 and SSI2 runs as master transmitter, or in case that hdmi0_sck_ws_sel = B'0000 and SSI0 runs as master transmitter and SSI2 runs as slave transmitter.
 4. In case that hdmi0_sck_ws_sel = B'0011 and SSI3 runs as master transmitter, or in case that hdmi0_sck_ws_sel = B'0000 and SSI0 runs as master transmitter and SSI3 runs as slave transmitter.

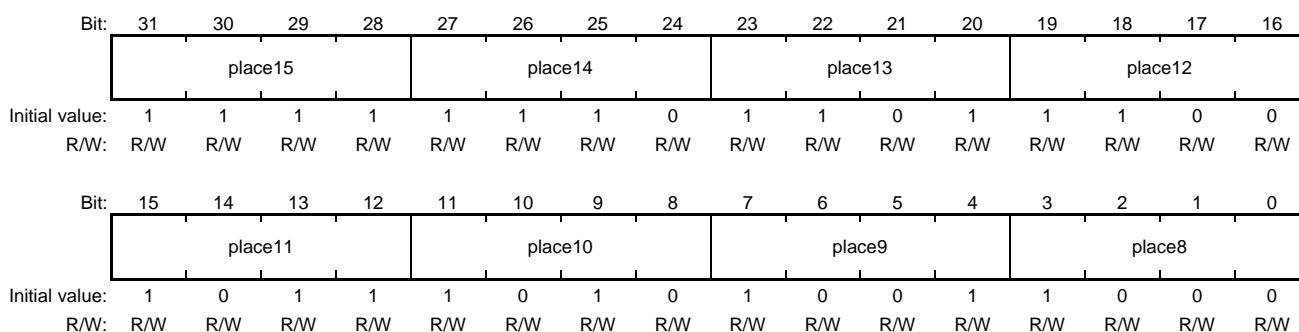
5. In case that `hdmi0_sck_ws_sel = B'0100` and SSI4 runs as master transmitter, or in case that `hdmi0_sck_ws_sel = B'0011` and SSI3 runs as master transmitter and SSI4 runs as slave transmitter.
6. In case that `hdmi0_sck_ws_sel = B'0101` and SSI5 runs as master transmitter.
7. In case that `hdmi0_sck_ws_sel = B'0110` and SSI6 runs as master transmitter.
8. In case that `hdmi0_sck_ws_sel = B'0111` and SSI7 runs as master transmitter.
9. In case that `hdmi0_sck_ws_sel = B'0111` and SSI7 runs as master transmitter and SSI8 runs as slave transmitter.
10. In case that `hdmi0_sck_ws_sel = B'1001` and SSI9 runs as master transmitter, or in case that `hdmi0_sck_ws_sel = B'0000` and SSI0 runs as master transmitter and SSI9 runs as slave transmitter or in case that `hdmi0_sck_ws_sel = B'0011` and SSI3 runs as master transmitter and SSI9 runs as slave transmitter.

40.2.44 SSIm BUSIF Data Align2 Register (SSIm_BUSIF_DALIGN2)

Note: m = 0-0, 1-0, 2-0, 3-0, 4-0 or 9-0,

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSIm_BUSIF_DALIGN2 determines the initial settings of the SSIm route



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 28 | place15 | B'1111 | R/W | <p>Changes the stream data order. These bits are used for the TDM-16ch mode (tdm_split = 0, ex_func = 1, chnum = 10000). Other setting, the initial value should not be changed.</p> <p>B'0000: Data at input-side place 0 is sent to output-side place 15. B'0001: Data at input-side place 1 is sent to output-side place 15. B'0010: Data at input-side place 2 is sent to output-side place 15. B'0011: Data at input-side place 3 is sent to output-side place 15. B'0100: Data at input-side place 4 is sent to output-side place 15. B'0101: Data at input-side place 5 is sent to output-side place 15. B'0110: Data at input-side place 6 is sent to output-side place 15. B'0111: Data at input-side place 7 is sent to output-side place 15. B'1000: Data at input-side place 8 is sent to output-side place 15. B'1001: Data at input-side place 9 is sent to output-side place 15. B'1010: Data at input-side place 10 is sent to output-side place 15. B'1011: Data at input-side place 11 is sent to output-side place 15. B'1100: Data at input-side place 12 is sent to output-side place 15. B'1101: Data at input-side place 13 is sent to output-side place 15. B'1110: Data at input-side place 14 is sent to output-side place 15. B'1111: Data at input-side place 15 is sent to output-side place 15.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 27 to 24 | place14 | B'1110 | R/W | <p>Changes the stream data order. These bits are used for the TDM-16ch mode (tdm_split = 0, ex_func = 1, chnum = 10000). Other setting, the initial value should not be changed.</p> <p>B'0000: Data at input-side place 0 is sent to output-side place 14. B'0001: Data at input-side place 1 is sent to output-side place 14. B'0010: Data at input-side place 2 is sent to output-side place 14. B'0011: Data at input-side place 3 is sent to output-side place 14. B'0100: Data at input-side place 4 is sent to output-side place 14. B'0101: Data at input-side place 5 is sent to output-side place 14. B'0110: Data at input-side place 6 is sent to output-side place 14. B'0111: Data at input-side place 7 is sent to output-side place 14. B'1000: Data at input-side place 8 is sent to output-side place 14. B'1001: Data at input-side place 9 is sent to output-side place 14. B'1010: Data at input-side place 10 is sent to output-side place 14. B'1011: Data at input-side place 11 is sent to output-side place 14. B'1100: Data at input-side place 12 is sent to output-side place 14. B'1101: Data at input-side place 13 is sent to output-side place 14. B'1110: Data at input-side place 14 is sent to output-side place 14. B'1111: Data at input-side place 15 is sent to output-side place 14.</p> |
| 23 to 20 | place13 | B'1101 | R/W | <p>Changes the stream data order. These bits are used for the TDM-16ch mode (tdm_split = 0, ex_func = 1, chnum = 10000). Other setting, the initial value should not be changed.</p> <p>B'0000: Data at input-side place 0 is sent to output-side place 13. B'0001: Data at input-side place 1 is sent to output-side place 13. B'0010: Data at input-side place 2 is sent to output-side place 13. B'0011: Data at input-side place 3 is sent to output-side place 13. B'0100: Data at input-side place 4 is sent to output-side place 13. B'0101: Data at input-side place 5 is sent to output-side place 13. B'0110: Data at input-side place 6 is sent to output-side place 13. B'0111: Data at input-side place 7 is sent to output-side place 13. B'1000: Data at input-side place 8 is sent to output-side place 13. B'1001: Data at input-side place 9 is sent to output-side place 13. B'1010: Data at input-side place 10 is sent to output-side place 13. B'1011: Data at input-side place 11 is sent to output-side place 13. B'1100: Data at input-side place 12 is sent to output-side place 13. B'1101: Data at input-side place 13 is sent to output-side place 13. B'1110: Data at input-side place 14 is sent to output-side place 13. B'1111: Data at input-side place 15 is sent to output-side place 13.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 19 to 16 | place12 | B'1100 | R/W | <p>Changes the stream data order. These bits are used for the TDM-16ch mode (tdm_split = 0, ex_func = 1, chnum = 10000). Other setting, the initial value should not be changed.</p> <p>B'0000: Data at input-side place 0 is sent to output-side place 12. B'0001: Data at input-side place 1 is sent to output-side place 12. B'0010: Data at input-side place 2 is sent to output-side place 12. B'0011: Data at input-side place 3 is sent to output-side place 12. B'0100: Data at input-side place 4 is sent to output-side place 12. B'0101: Data at input-side place 5 is sent to output-side place 12. B'0110: Data at input-side place 6 is sent to output-side place 12. B'0111: Data at input-side place 7 is sent to output-side place 12. B'1000: Data at input-side place 8 is sent to output-side place 12. B'1001: Data at input-side place 9 is sent to output-side place 12. B'1010: Data at input-side place 10 is sent to output-side place 12. B'1011: Data at input-side place 11 is sent to output-side place 12. B'1100: Data at input-side place 12 is sent to output-side place 12. B'1101: Data at input-side place 13 is sent to output-side place 12. B'1110: Data at input-side place 14 is sent to output-side place 12. B'1111: Data at input-side place 15 is sent to output-side place 12.</p> |
| 15 to 12 | place11 | B'1011 | R/W | <p>Changes the stream data order. These bits are used for the TDM-16ch mode (tdm_split = 0, ex_func = 1, chnum = 10000). Other setting, the initial value should not be changed.</p> <p>B'0000: Data at input-side place 0 is sent to output-side place 11. B'0001: Data at input-side place 1 is sent to output-side place 11. B'0010: Data at input-side place 2 is sent to output-side place 11. B'0011: Data at input-side place 3 is sent to output-side place 11. B'0100: Data at input-side place 4 is sent to output-side place 11. B'0101: Data at input-side place 5 is sent to output-side place 11. B'0110: Data at input-side place 6 is sent to output-side place 11. B'0111: Data at input-side place 7 is sent to output-side place 11. B'1000: Data at input-side place 8 is sent to output-side place 11. B'1001: Data at input-side place 9 is sent to output-side place 11. B'1010: Data at input-side place 10 is sent to output-side place 11. B'1011: Data at input-side place 11 is sent to output-side place 11. B'1100: Data at input-side place 12 is sent to output-side place 11. B'1101: Data at input-side place 13 is sent to output-side place 11. B'1110: Data at input-side place 14 is sent to output-side place 11. B'1111: Data at input-side place 15 is sent to output-side place 11.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 11 to 8 | place10 | B'1010 | R/W | <p>Changes the stream data order. These bits are used for the TDM-16ch mode (<code>tdm_split = 0</code>, <code>ex_func = 1</code>, <code>chnum = 10000</code>). Other setting, the initial value should not be changed.</p> <p>B'0000: Data at input-side place 0 is sent to output-side place 10. B'0001: Data at input-side place 1 is sent to output-side place 10. B'0010: Data at input-side place 2 is sent to output-side place 10. B'0011: Data at input-side place 3 is sent to output-side place 10. B'0100: Data at input-side place 4 is sent to output-side place 10. B'0101: Data at input-side place 5 is sent to output-side place 10. B'0110: Data at input-side place 6 is sent to output-side place 10. B'0111: Data at input-side place 7 is sent to output-side place 10. B'1000: Data at input-side place 8 is sent to output-side place 10. B'1001: Data at input-side place 9 is sent to output-side place 10. B'1010: Data at input-side place 10 is sent to output-side place 10. B'1011: Data at input-side place 11 is sent to output-side place 10. B'1100: Data at input-side place 12 is sent to output-side place 10. B'1101: Data at input-side place 13 is sent to output-side place 10. B'1110: Data at input-side place 14 is sent to output-side place 10. B'1111: Data at input-side place 15 is sent to output-side place 10.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 4 | place9 | B'1001 | R/W | <p>Changes the stream data order. These bits are used for the TDM-16ch mode (tdm_split = 0, ex_func = 1, chnum = 10000). or TDM Ex Split Mode (tdm_split = 0, ex_func = 1)</p> <p>Other setting, the initial value should not be changed.</p> <ul style="list-style-type: none"> When TDM 16Ch mode is used (tdm_split = 0, ex_func = 1, chnum = 10000). B'0000: Data at input-side place 0 is sent to output-side place 9. B'0001: Data at input-side place 1 is sent to output-side place 9. B'0010: Data at input-side place 2 is sent to output-side place 9. B'0011: Data at input-side place 3 is sent to output-side place 9. B'0100: Data at input-side place 4 is sent to output-side place 9. B'0101: Data at input-side place 5 is sent to output-side place 9. B'0110: Data at input-side place 6 is sent to output-side place 9. B'0111: Data at input-side place 7 is sent to output-side place 9. B'1000: Data at input-side place 8 is sent to output-side place 9. B'1001: Data at input-side place 9 is sent to output-side place 9. B'1010: Data at input-side place 10 is sent to output-side place 9. B'1011: Data at input-side place 11 is sent to output-side place 9. B'1100: Data at input-side place 12 is sent to output-side place 9. B'1101: Data at input-side place 13 is sent to output-side place 9. B'1110: Data at input-side place 14 is sent to output-side place 9. B'1111: Data at input-side place 15 is sent to output-side place 9. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) When SSIq_0 are used for the 10 channel setting B'0000: Data at input-side place 0 is sent to output-side place 9. B'0001: Data at input-side place 1 is sent to output-side place 9. B'0010: Data at input-side place 2 is sent to output-side place 9. B'0011: Data at input-side place 3 is sent to output-side place 9. B'0100: Data at input-side place 4 is sent to output-side place 9. B'0101: Data at input-side place 5 is sent to output-side place 9. B'0110: Data at input-side place 6 is sent to output-side place 9. B'0111: Data at input-side place 7 is sent to output-side place 9. B'1000: Data at input-side place 8 is sent to output-side place 9. B'1001: Data at input-side place 9 is sent to output-side place 9. <p>Other Setting All bit is reserved. The initial value should not be changed.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 3 to 0 | place8 | B'1000 | R/W | <p>Changes the stream data order. These bits are used for the TDM-16ch mode (tdm_split = 0, ex_func = 1, chnum = 10000). or TDM Ex Split Mode (tdm_split = 0, ex_func = 1)</p> <p>Other setting, the initial value should not be changed.</p> <ul style="list-style-type: none"> When TDM 16Ch mode is used (tdm_split = 0, ex_func = 1, chnum = 10000). B'0000: Data at input-side place 0 is sent to output-side place 8. B'0001: Data at input-side place 1 is sent to output-side place 8. B'0010: Data at input-side place 2 is sent to output-side place 8. B'0011: Data at input-side place 3 is sent to output-side place 8. B'0100: Data at input-side place 4 is sent to output-side place 8. B'0101: Data at input-side place 5 is sent to output-side place 8. B'0110: Data at input-side place 6 is sent to output-side place 8. B'0111: Data at input-side place 7 is sent to output-side place 8. B'1000: Data at input-side place 8 is sent to output-side place 8. B'1001: Data at input-side place 9 is sent to output-side place 8. B'1010: Data at input-side place 10 is sent to output-side place 8. B'1011: Data at input-side place 11 is sent to output-side place 8. B'1100: Data at input-side place 12 is sent to output-side place 8. B'1101: Data at input-side place 13 is sent to output-side place 8. B'1110: Data at input-side place 14 is sent to output-side place 8. B'1111: Data at input-side place 15 is sent to output-side place 8. When TDM Ex Split mode is used (tdm_split = 1, ex_func = 1) When SSIq_0 are used for the 10 channel setting B'0000: Data at input-side place 0 is sent to output-side place 8. B'0001: Data at input-side place 1 is sent to output-side place 8. B'0010: Data at input-side place 2 is sent to output-side place 8. B'0011: Data at input-side place 3 is sent to output-side place 8. B'0100: Data at input-side place 4 is sent to output-side place 8. B'0101: Data at input-side place 5 is sent to output-side place 8. B'0110: Data at input-side place 6 is sent to output-side place 8. B'0111: Data at input-side place 7 is sent to output-side place 8. B'1000: Data at input-side place 8 is sent to output-side place 8. B'1001: Data at input-side place 9 is sent to output-side place 8. <p>Other Setting All bit is reserved. The initial value should not be changed.</p> |

40.2.45 SSIq Mode2 Register (SSIq_MODE2)

Note: q = 0-0, 1-0, 2-0, 3-0, 4-0 or 9-0

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSIq_MODE2 determines the initial settings of the SSIq route

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ex_func |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | ex_func | B'0 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Enable TDM Ex Func Mode and TDM-16ch mode 0: TDM ex-split mode/TDM-16ch mode Disable. 1: TDM ex-split mode/TDM-16ch mode Enable [RZ/G2E] Enable TDM Ex Func Mode 0: TDM ex-split mode Disable. 1: TDM ex-split mode Enable |

40.2.46 SSIq Status2 Register (SSIq_STATUS2)

Note: q = 0-0, 1-0, 2-0, 3-0, 4-0, 9-0

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSIq_STATUS2 indicates the internal buffer state. When a bit in this register is set, its status is indicated by the corresponding interrupt signal. However, when the interrupt output is masked by the SSIq Interrupt Enable2 Register, the interrupt signal is not output (refer to section 40.2.47 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | uf_7 | uf_6 | uf_5 | uf_4 | of_7 | of_6 | of_5 | of_4 | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | uf_7 | B'0 | R | buf_under_flow <i>i</i> -7 (i = 0 to 4, 9) Indicates the state of SSI_SYSTEM_STATUS6/7.uf <i>i</i> -7. 0: — 1: An underflow has occurred. Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE6/7.uf <i>i</i> -7_ie = 0. |
| 14 | uf_6 | B'0 | R | buf_under_flow <i>i</i> -6 (i = 0 to 4, 9) Indicates the state of SSI_SYSTEM_STATUS6/7.uf <i>i</i> -6. 0: — 1: An underflow has occurred. Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE6/7.uf <i>i</i> -6_ie = 0. |
| 13 | uf_5 | B'0 | R | buf_under_flow <i>i</i> -5 (i = 0 to 4, 9) Indicates the state of SSI_SYSTEM_STATUS6/7.uf <i>i</i> -5. 0: — 1: An underflow has occurred. Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE6/7.uf <i>i</i> -5_ie = 0. |
| 12 | uf_4 | B'0 | R | buf_under_flow <i>i</i> -4 (i = 0 to 4, 9) Indicates the state of SSI_SYSTEM_STATUS6/7.uf <i>i</i> -4. 0: — 1: An underflow has occurred. Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE6/7.uf <i>i</i> -4_ie = 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 11 | of_7 | B'0 | R | buf_over_flow <i>i</i> -7 (i = 0 to 4, 9) Indicates the state of SSI_SYSTEM_STATUS4/5.of <i>i</i> -7. 0: — 1: An overflow has occurred. Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE4/5.of <i>i</i> -7_ie = 0. |
| 10 | of_6 | B'0 | R | buf_over_flow <i>i</i> -6 (i = 0 to 4, 9) Indicates the state of SSI_SYSTEM_STATUS4/5.of <i>i</i> -6. 0: — 1: An overflow has occurred. Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE4/5.of <i>i</i> -6_ie = 0. |
| 9 | of_5 | B'0 | R | buf_over_flow <i>i</i> -5 (i = 0 to 4, 9) Indicates the state of SSI_SYSTEM_STATUS4/5.of <i>i</i> -5. 0: — 1: An overflow has occurred. Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE4/5.of <i>i</i> -5_ie = 0. |
| 8 | of_4 | B'0 | R | buf_over_flow <i>i</i> -4 (i = 0 to 4, 9) Indicates the state of SSI_SYSTEM_STATUS4/5.of <i>i</i> -4. 0: — 1: An overflow has occurred. Note: Not indicated by the interrupt signal when SSI_SYSTEM_INT_ENABLE4/5.of <i>i</i> -4_ie = 0. |
| 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

40.2.47 SSIq Interrupt Enable2 Register (SSIq_INT_ENABLE_MAIN2)

Note: q = 0-0, 1-0, 2-0, 3-0, 4-0 or 9-0

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: SSIq_INT_ENABLE_MAIN2 enables or disables output of interrupts corresponding to the states indicated in the SSIq Status Register2 (refer to section 40.2.46 and section 40.3.14).

| | | | | | | | | | | | | | | | | |
|----------------|---------|---------|---------|---------|---------|---------|---------|---------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | uf_7_ie | uf_6_ie | uf_5_ie | uf_4_ie | of_7_ie | of_6_ie | of_5_ie | of_4_ie | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |

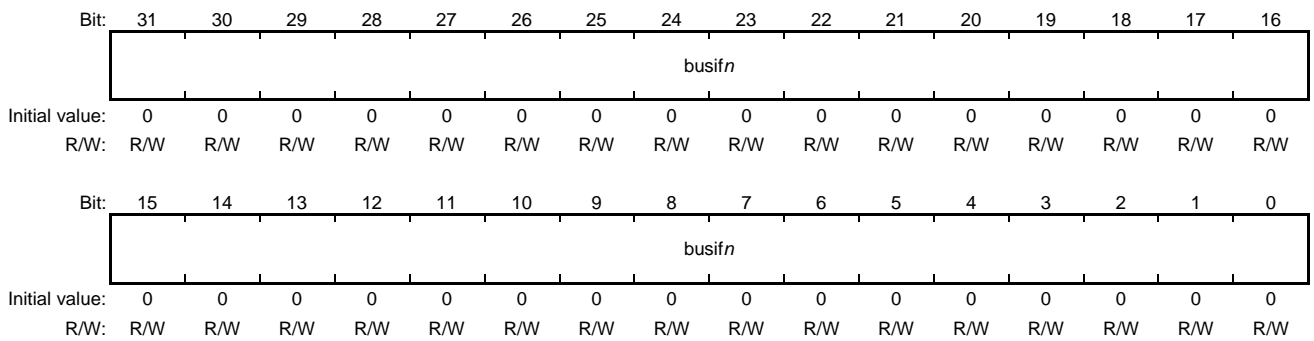
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | uf_7_ie | B'0 | R/W | buf_under_flow <i>i</i> -7_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 14 | uf_6_ie | B'0 | R/W | buf_under_flow <i>i</i> -6_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 13 | uf_5_ie | B'0 | R/W | buf_under_flow <i>i</i> -5_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 12 | uf_4_ie | B'0 | R/W | buf_under_flow <i>i</i> -4_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 11 | of_7_ie | B'0 | R/W | buf_over_flow <i>i</i> -7_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 10 | of_6_ie | B'0 | R/W | buf_over_flow <i>i</i> -6_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 9 | of_5_ie | B'0 | R/W | buf_over_flow <i>i</i> -5_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 8 | of_4_ie | B'0 | R/W | buf_over_flow <i>i</i> -4_int_enable (i = 0 to 4, 9) 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

40.2.48 SSIn_BUSIF Data Register (SSIn_BUSIF)

Note: n = 0-0 to 0-7, 1-0 to 1-7, 2-0 to 2-7, 3-0 to 3-7, 4-0 to 4-7, 5 to 8, or 9-0 to 9-7 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 n = 0-0 to 0-3, 1-0 to 1-3, 2-0 to 2-3, 3-0 to 3-3, 4-0 to 4-3, 5 to 8, or 9-0 to 9-3 [RZ/G2E]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SSIn_BUSIF is a window register in which data is stored during data transfer via SSIn_BUSIF. These registers are used for both transmission and reception.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | busifn | All 0 | R/W | These bits are used to hold the data during data transfer via SSIn_BUSIFn. This register is used for both transmission and reception. This register can only be written to during transmission and only be read from during reception. |

40.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

40.3.1 Module Specifications

Table 40.5 and Table 40.6 show the correspondences between the modules and functions.

Table 40.5 Function Correspondences [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| | | External Interface | Format | | TDM Split Mode/ TDM 16Ch Mode/ TDM Ex-Split Mode |
|---|--|---------------------------|---------------------|-----------------|--|
| | | | Basic Configuration | TDM Extend Mode | |
| 1 | SSI0 SSI1 SSI2 SSI3 SSI4 SSI9 | I2S | Supported | Not supported | Not supported |
| 2 | SSI0 SSI1 SSI2 SSI3 SSI4 SSI9 | TDM | Supported | Supported | Supported |
| 3 | SSI0 SSI1 SSI2 | I2S × 3 (multichannel) | Supported | Not supported | Not supported |
| 4 | SSI0 SSI1 SSI2 SSI9 | I2S × 4 (multichannel) | Supported | Not supported | Not supported |
| 5 | SSI5 SSI6 SSI7 SSI8 | I2S | Supported | Not supported | Not supported |
| 6 | SSI5 SSI6 SSI7 SSI8 | TDM | Not supported | Not supported | Not supported |

Table 40.6 Function Correspondences [RZ/G2E]

| | | External Interface | Format | | |
|---|--|---------------------------|---------------------|-----------------|--------------------------------------|
| | | | Basic Configuration | TDM Extend Mode | TDM Split Mode/ TDM Ex-Split Mode |
| 1 | SSI0 SSI1 SSI2 SSI3 SSI4 SSI9 | I2S | Supported | Not supported | Not supported |
| 2 | SSI0 SSI1 SSI2 SSI3 SSI4 SSI9 | TDM | Supported | Supported | Supported |
| 3 | SSI0 SSI1 SSI2 | I2S × 3 (multichannel) | Supported | Not supported | Not supported |
| 4 | SSI0 SSI1 SSI2 SSI9 | I2S × 4 (multichannel) | Supported | Not supported | Not supported |
| 5 | SSI5 SSI6 SSI7 SSI8 | I2S | Supported | Not supported | Not supported |
| 6 | SSI5 SSI6 SSI7 SSI8 | TDM | Not supported | Not supported | Not supported |

40.3.2 Basic Configuration

Data transfer is performed between the SSIU and other audio modules or external memories via BUSIF connected to the audio local bus and the audio DMA bus.

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Figure 40.5 shows the data transfer of SSI0. As shown in Figure 40.5, SSI0-1_BUSIF to SSI0-7_BUSIF are not used. This basic configuration is also applied to SSI1, SSI2, SSI3, SSI4 and SSI9. Figure 40.6 shows the data transfer of SSI5, which is also applied to SSI6 to SSI8.

[RZ/G2E]

Figure 40.7 shows the data transfer of SSI0. As shown in Figure 40.7, SSI0-1_BUSIF to SSI0-3_BUSIF are not used. This basic configuration is also applied to SSI1, SSI2, SSI3, SSI4 and SSI9. Figure 40.6 shows the data transfer of SSI5, which is also applied to SSI6 to SSI8.

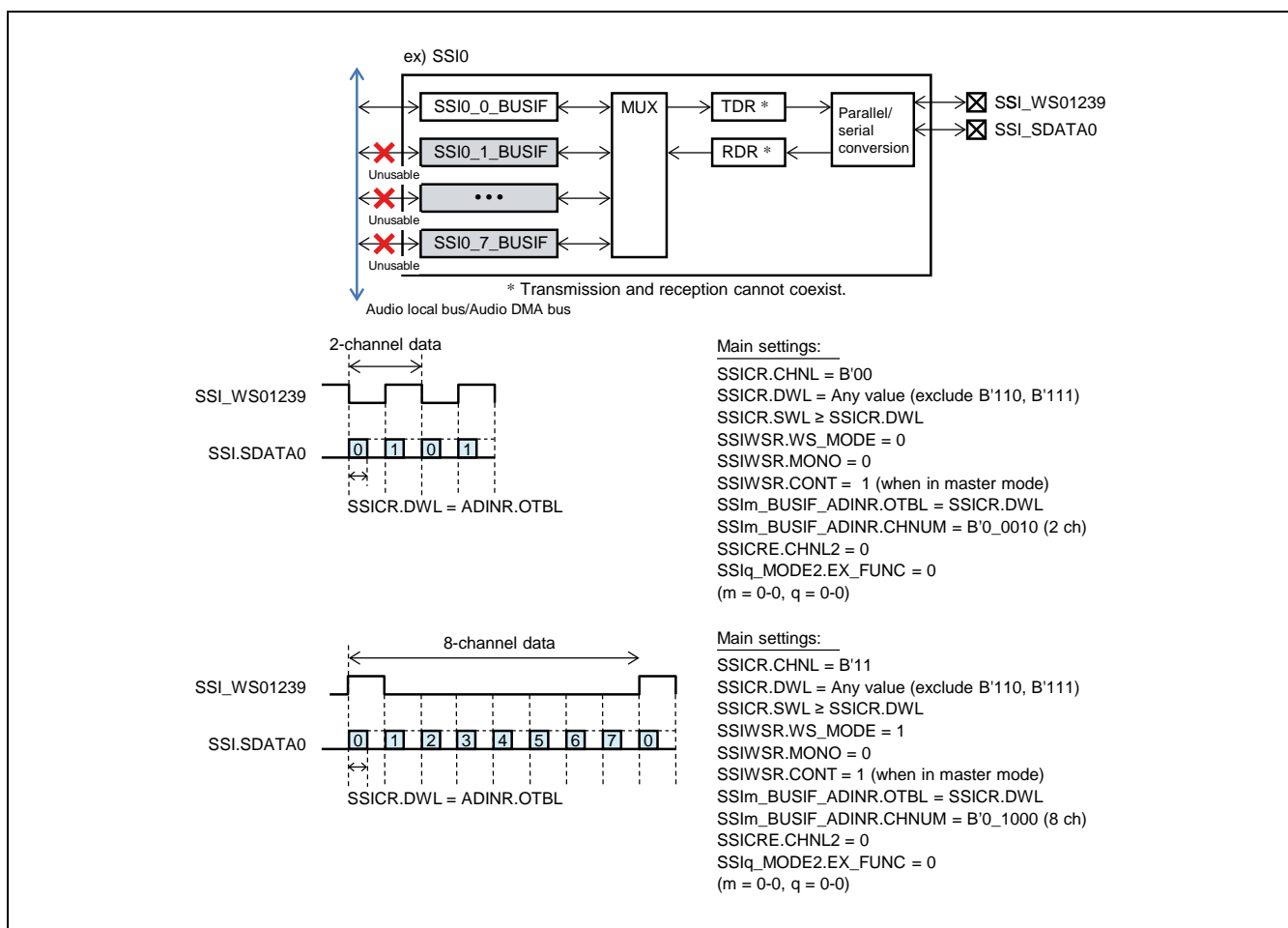


Figure 40.5 Basic Configuration (SSI0) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

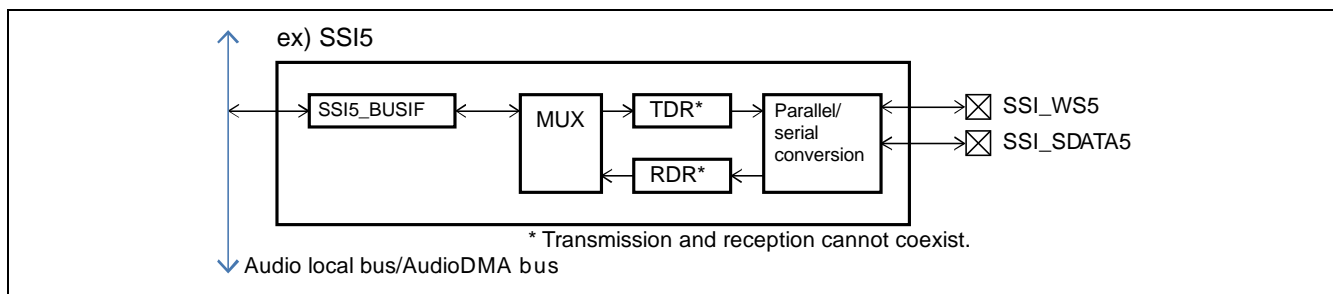


Figure 40.6 Basic Configuration (SSI5) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

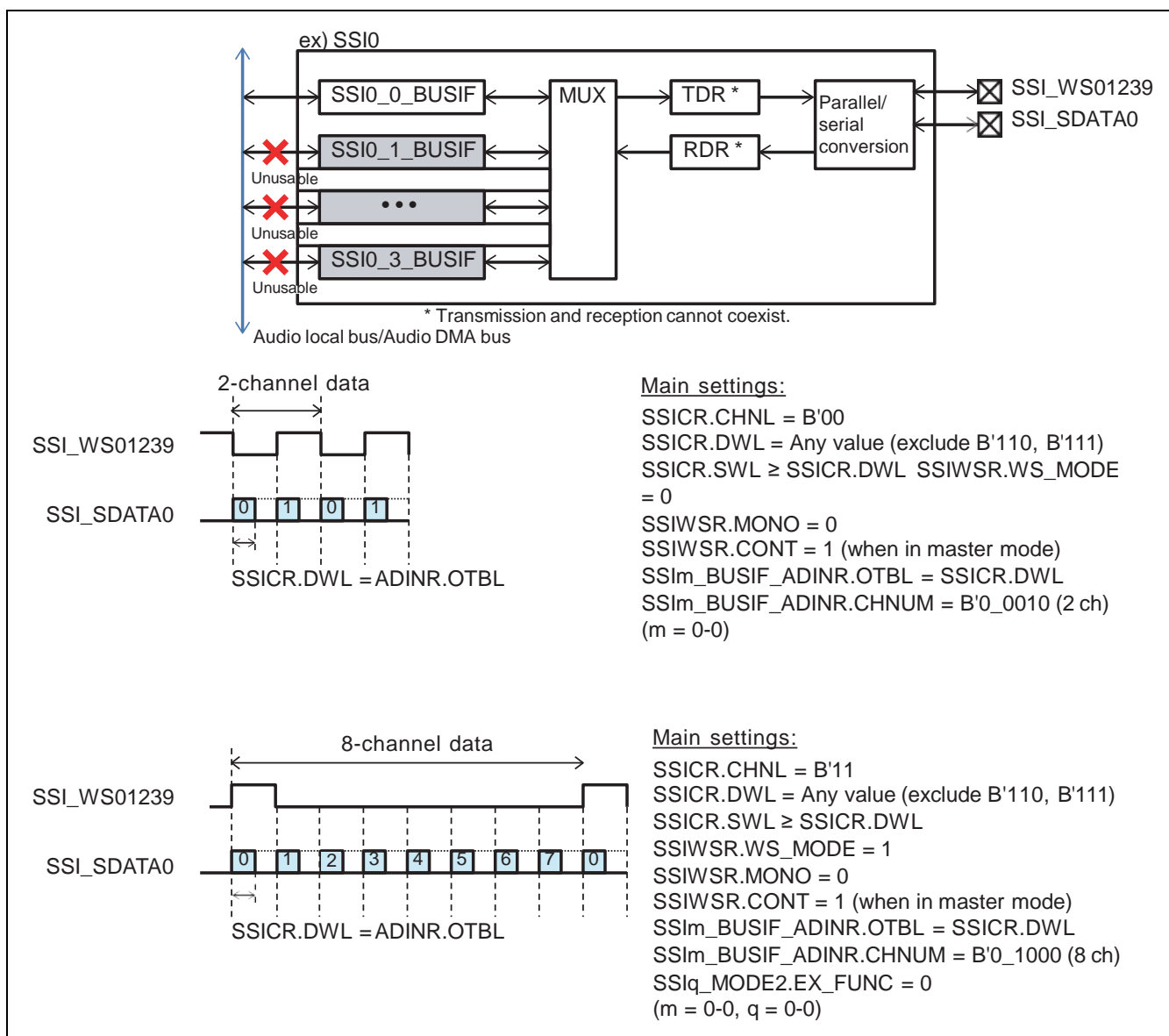


Figure 40.7 Basic Configuration (SSI0) [RZ/G2E]

40.3.3 Configuration with Multiple SSI Modules

Multichannel data transfer can be performed by using multiple SSI modules. The available combination of modules is SSI0, SSI1, and SSI2 or SSI0, SSI1, SSI2, and SSI9.

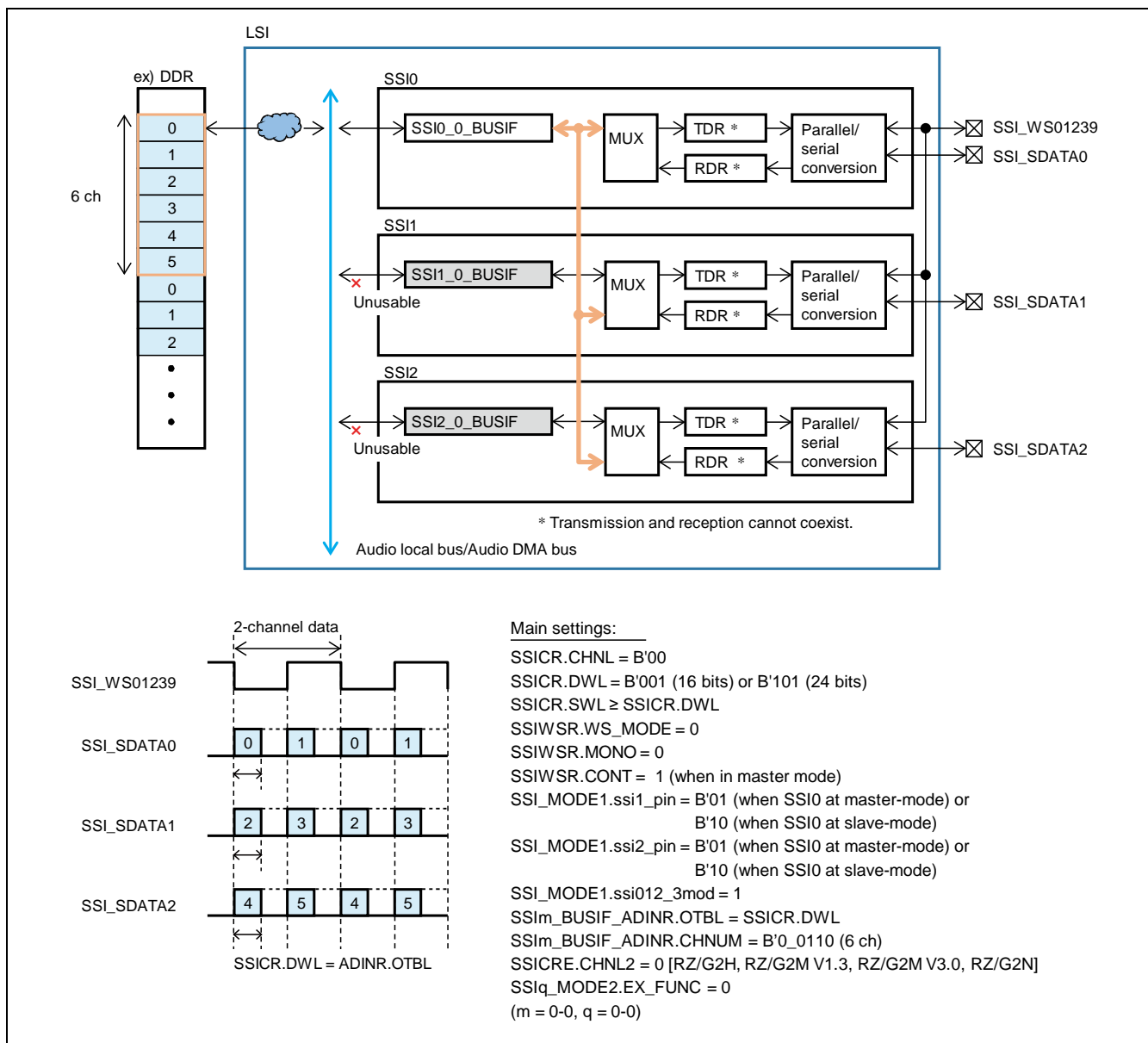


Figure 40.8 Multichannel Data Transfer (Using SSI0, SSI1 and SSI2)

Note: When the above configuration is used, SSI0-1_BUSIF to SSI0-7_BUSIF, all the BUSIFs of SSI1 and SSI2 cannot be used. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

When the above configuration is used, SSI0-1_BUSIF to SSI0-3_BUSIF, all the BUSIFs of SSI1 and SSI2 cannot be used. [RZ/G2E]

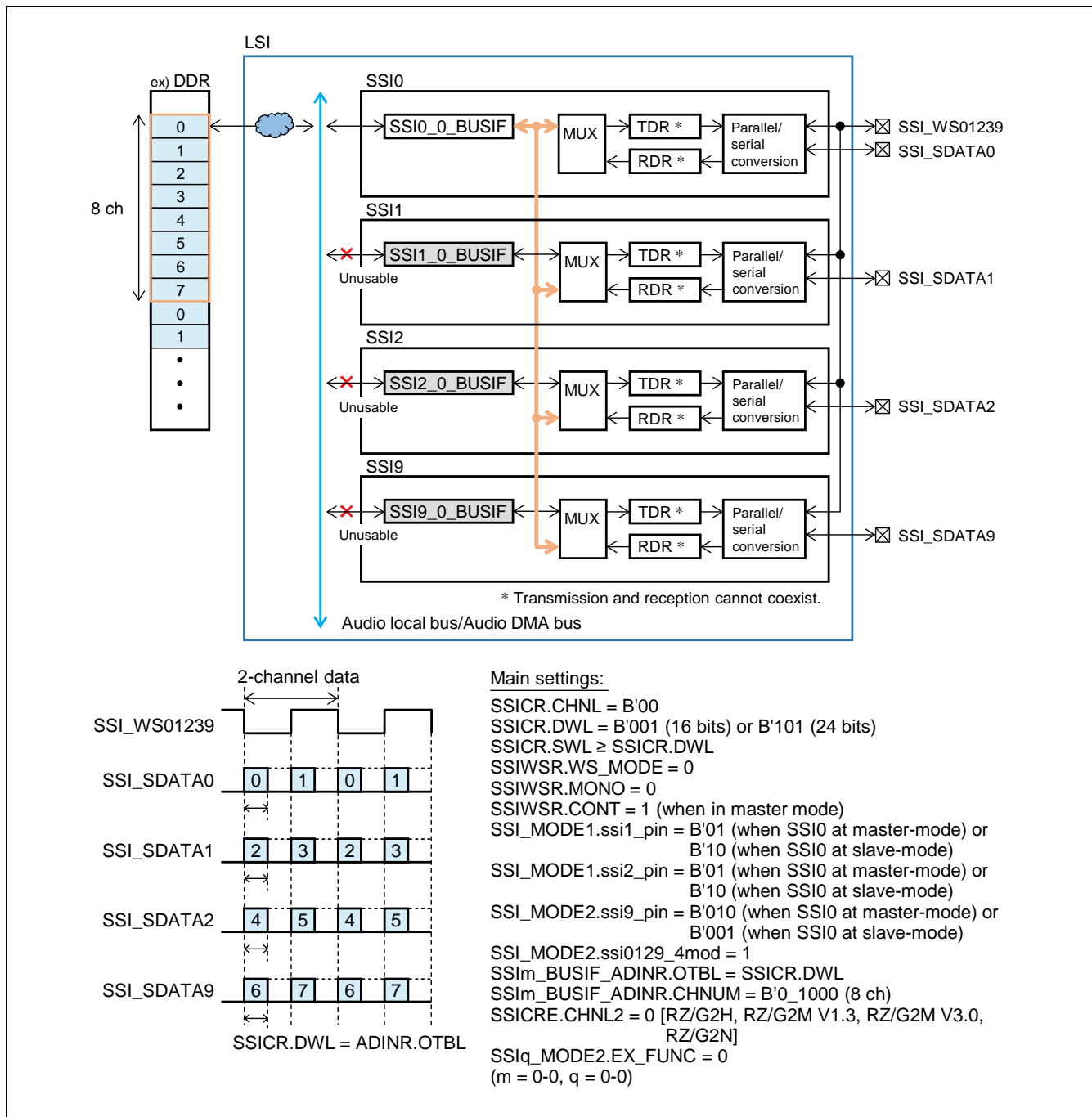


Figure 40.9 Multichannel Data Transfer (Using SSI0, SSI1, SSI2 and SSI9)

Note: When the above configuration is used, SSI0-1_BUSIF to SSI0-7_BUSIF, all the BUSIFs of SSI1, SSI2, and SSI9 cannot be used. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

When the above configuration is used, SSI0-1_BUSIF to SSI0-3_BUSIF, all the BUSIFs of SSI1, SSI2, and SSI9 cannot be used. [RZ/G2E]

40.3.4 TDM Format Extension Function (TDM Extend Mode) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

The TDM format extension function allows operation with 8-channel data in the serial bus and 6-channel data inside the LSI, or operation with 6-channel data in the serial bus and 8-channel data inside the LSI. This function is supported by SSI0, SSI1, SSI2, SSI3, SSI4 and SSI9.

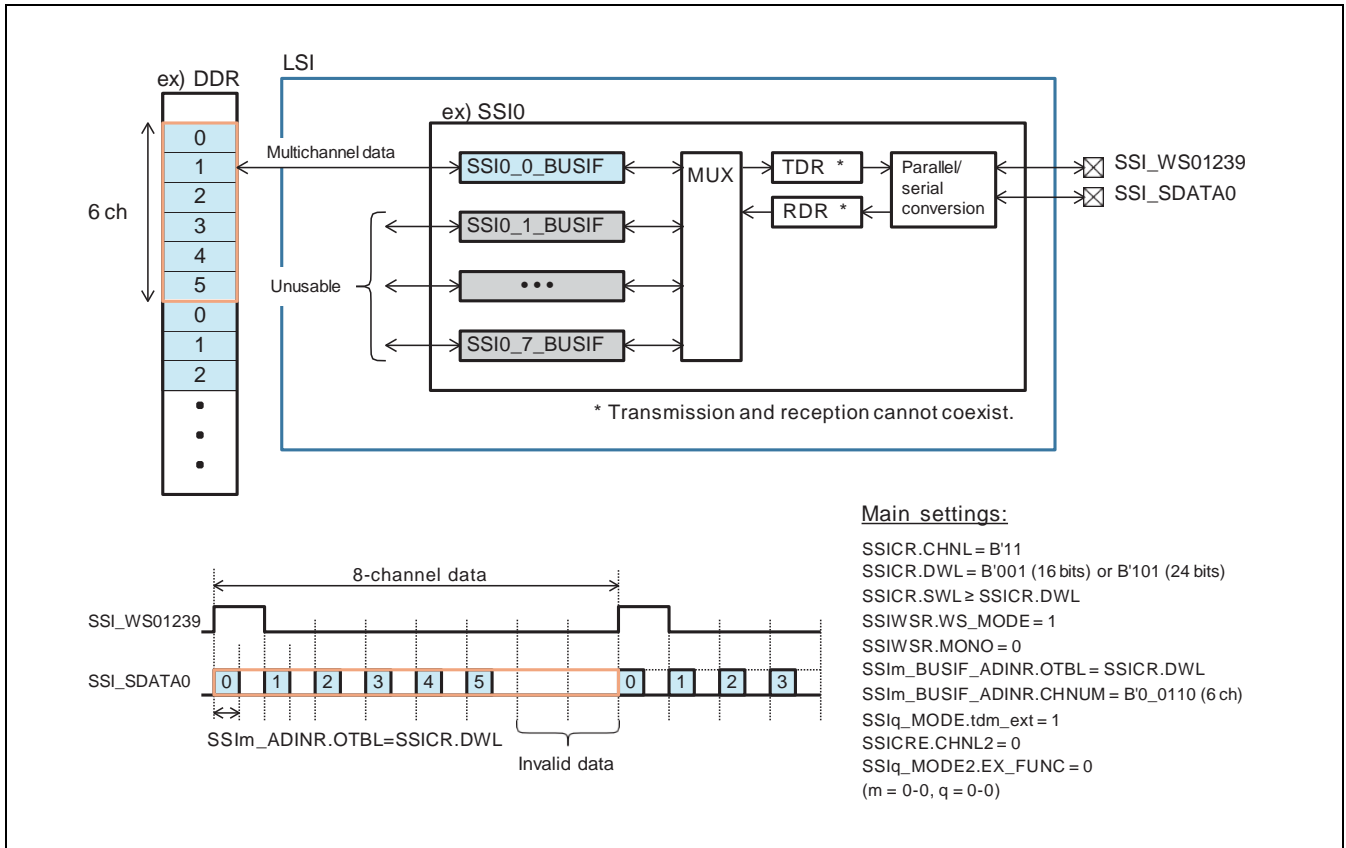


Figure 40.10 TDM Extend Mode Operation with 8-channel Data in Serial Bus and 6-channel Data Inside the LSI (SSI0)

Note: When the above configuration is used, SSI0-1_BUSIF to and SSI0-7_BUSIF cannot be used.

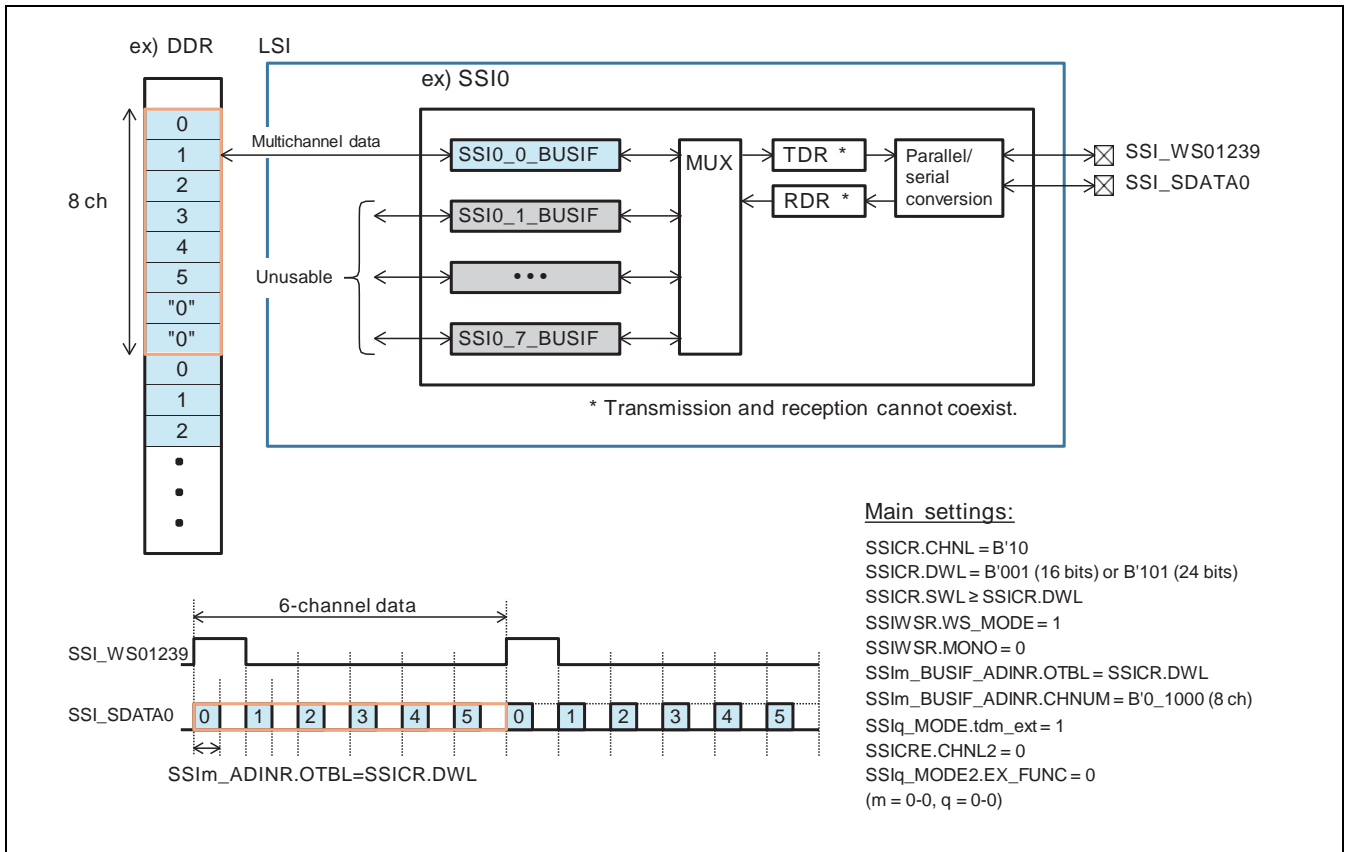


Figure 40.11 TDM Extend Mode Operation with 6-channel Data in Serial Bus and 8-channel Data Inside the LSI (SSI0)

Note: When the above configuration is used, SSIO-1_BUSIF to SSIO-7_BUSIF cannot be used.

40.3.5 TDM Format Extension Function (TDM Extend Mode) [RZ/G2E]

The TDM format extension function allows operation with 8-channel data in the serial bus and 6-channel data inside the LSI, or operation with 6-channel data in the serial bus and 8-channel data inside the LSI. This function is supported by SSI0, SSI1, SSI2, SSI3, SSI4 and SSI9.

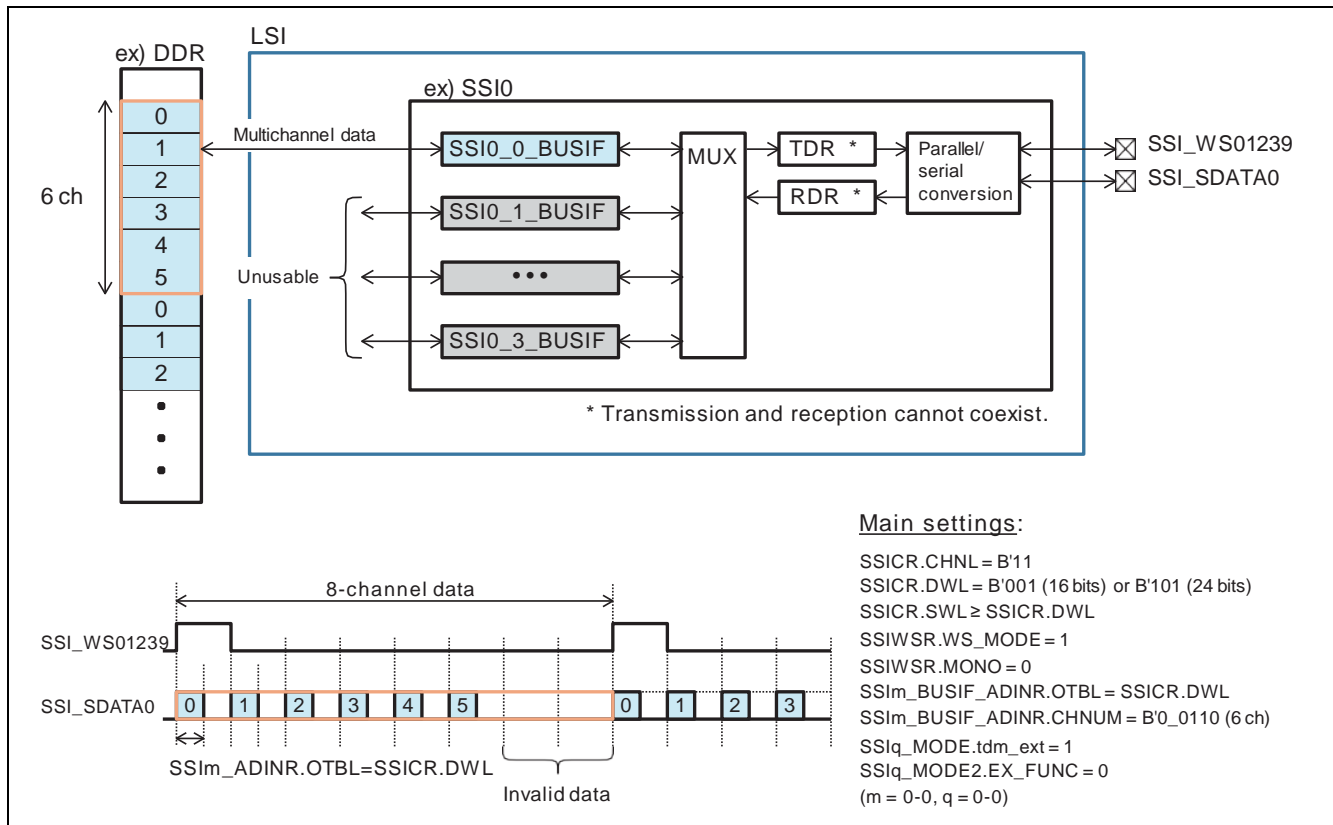


Figure 40.12 TDM Extend Mode Operation with 8-channel Data in Serial Bus and 6-channel Data Inside the LSI (SSI0)

Note: When the above configuration is used, SSI0-1_BUSIF to and SSI0-3_BUSIF cannot be used.

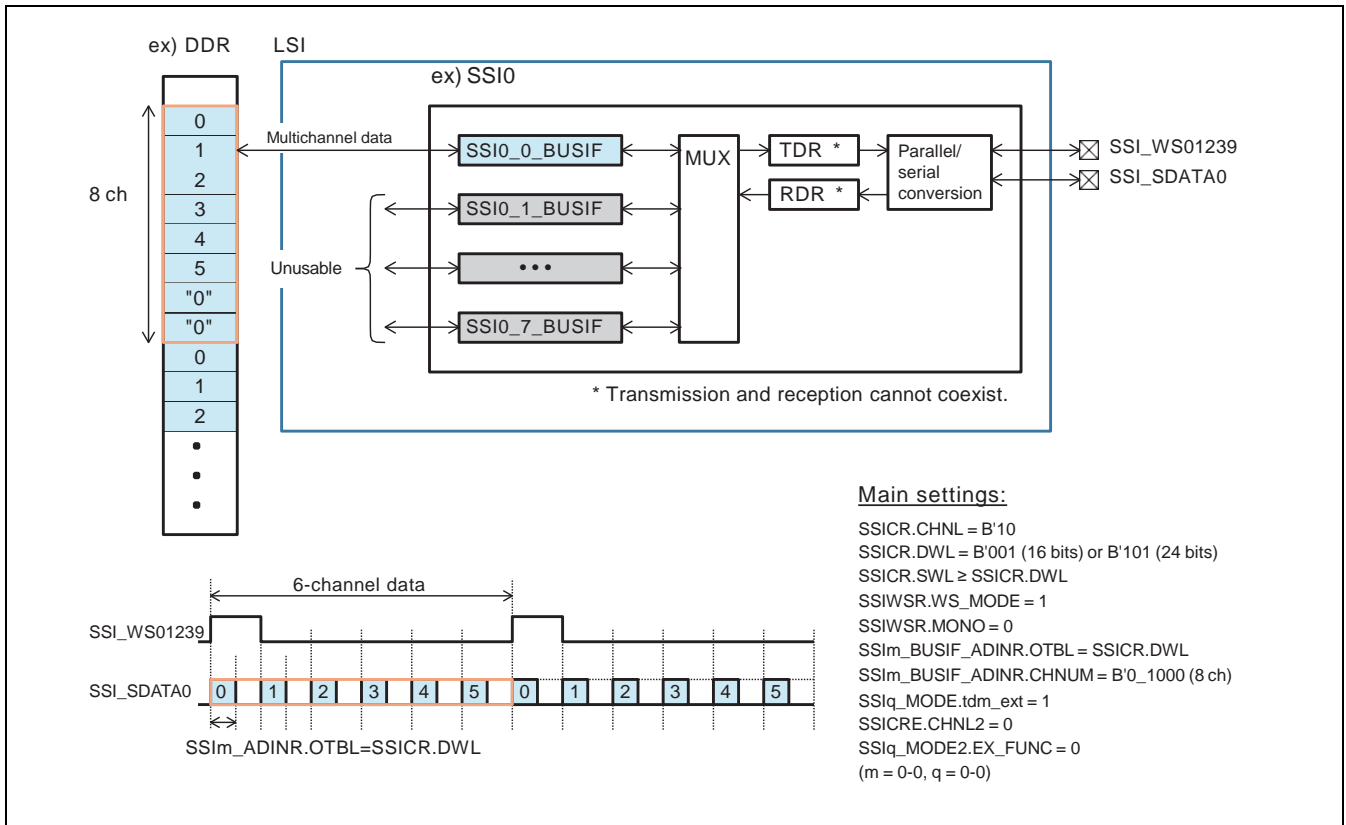


Figure 40.13 TDM Extend Mode Operation with 6-channel Data in Serial Bus and 8-channel Data Inside the LSI (SSI0)

Note: When the above configuration is used, SSI0-1_BUSIF to SSI0-3_BUSIF cannot be used.

40.3.6 TDM Format Split Function (TDM Split Mode) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

The TDM format split function (TDM split mode) allows connecting four independent monaural or stereo data inside the LSI and outputting the data in the TDM format, and splitting TDM format input data into four independent monaural or stereo data and distributing the data inside the LSI. This function allows TDM-8ch or TDM-4ch. This function is supported by SSI0, SSI1, SSI2, SSI3, SSI4 and SSI9.

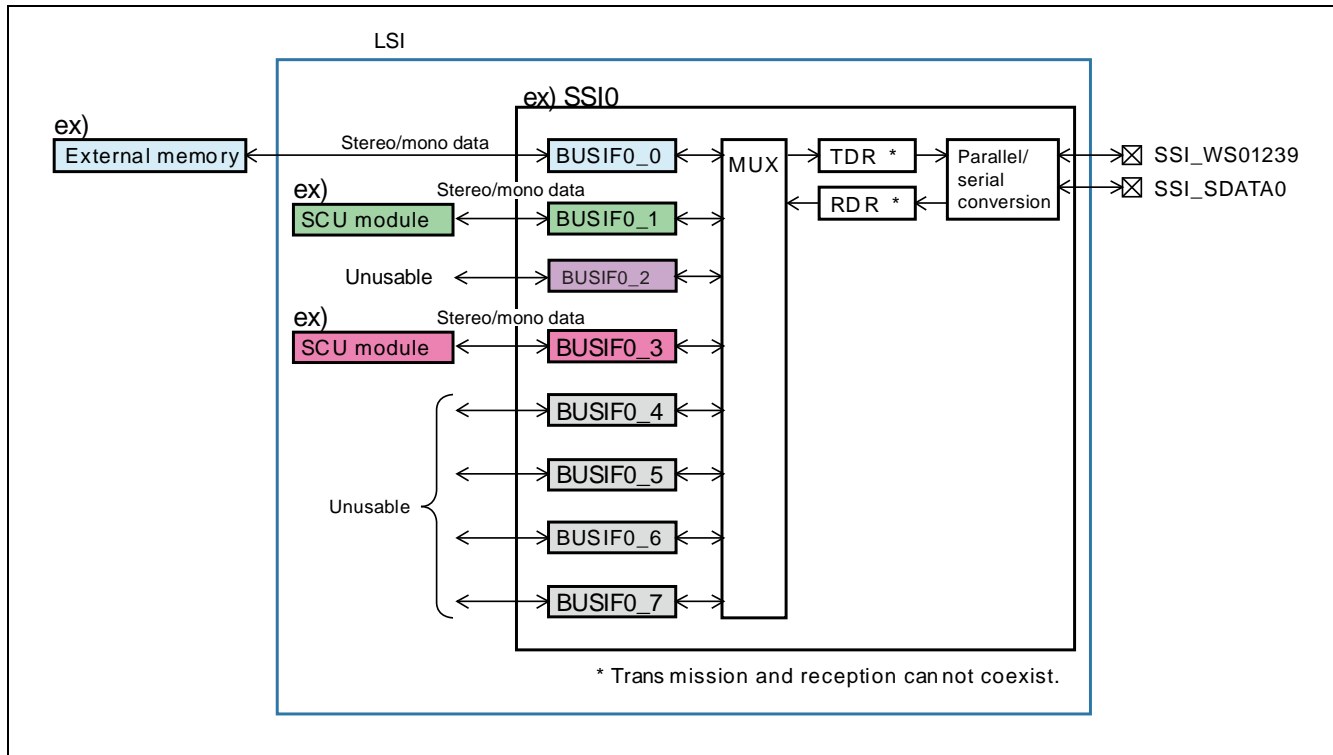


Figure 40.14 TDM Split Mode

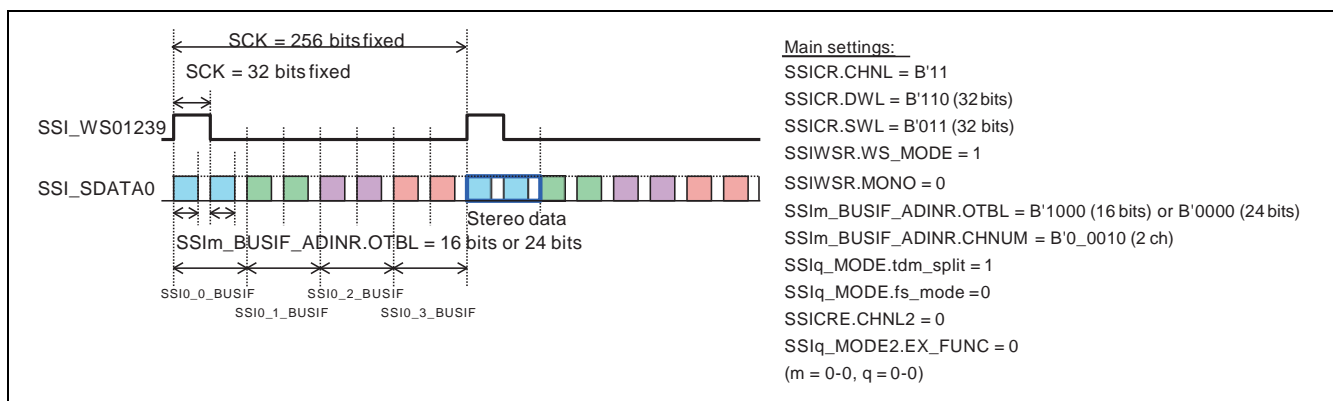


Figure 40.15 TDM Split Mode (Stereo x 4)

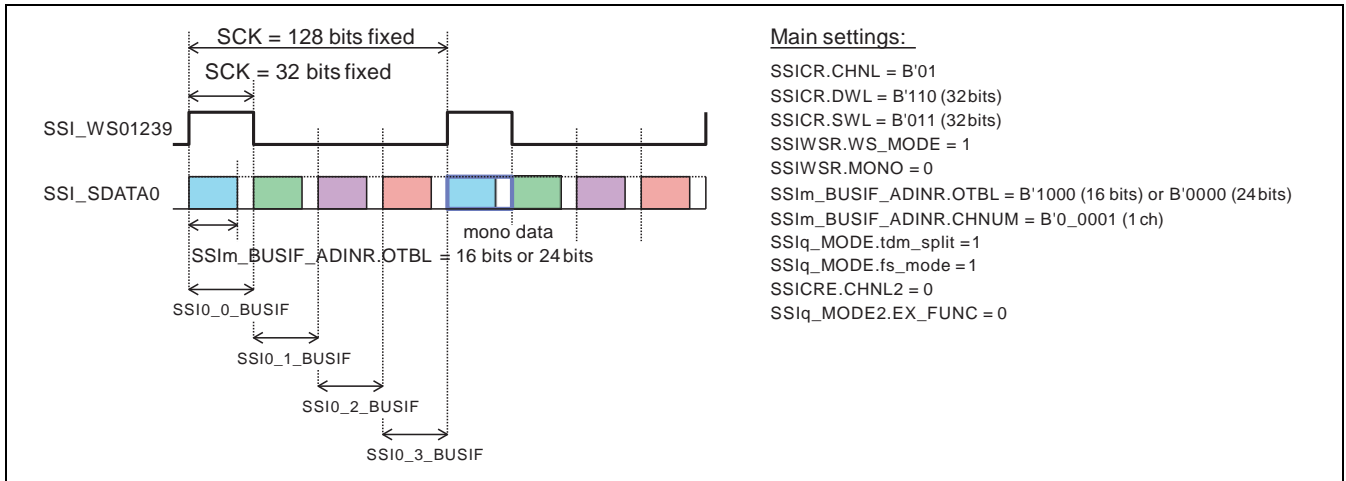


Figure 40.16 TDM Split Mode (Monaural x 4)

- Notes:
1. The SWL and DWL bits in SSICR should be fixed at 32-bit setting.
 2. Transmission and reception cannot coexist.
 3. Data input to BUSIFn 0 to 3 (n = 0, 1, 2, 3, 4, 9) should be operated synchronously with the SSI sampling frequency (WS signal cycle).
 4. Setting of SSIm_BUSIF_ADINR.OTBL = B'1000 (16bit) and SSIm_BUSIF_ADINR.OTBL = B'0000 (24bit) cannot coexist within the same SSI.

40.3.7 TDM Format Split Function (TDM Split Mode) [RZ/G2E]

The TDM format split function (TDM split mode) allows connecting four independent monaural or stereo data inside the LSI and outputting the data in the TDM format, and splitting TDM format input data into four independent monaural or stereo data and distributing the data inside the LSI. This function allows TDM-8ch or TDM-4ch. This function is supported by SSI0, SSI1, SSI2, SSI3, SSI4 and SSI9.

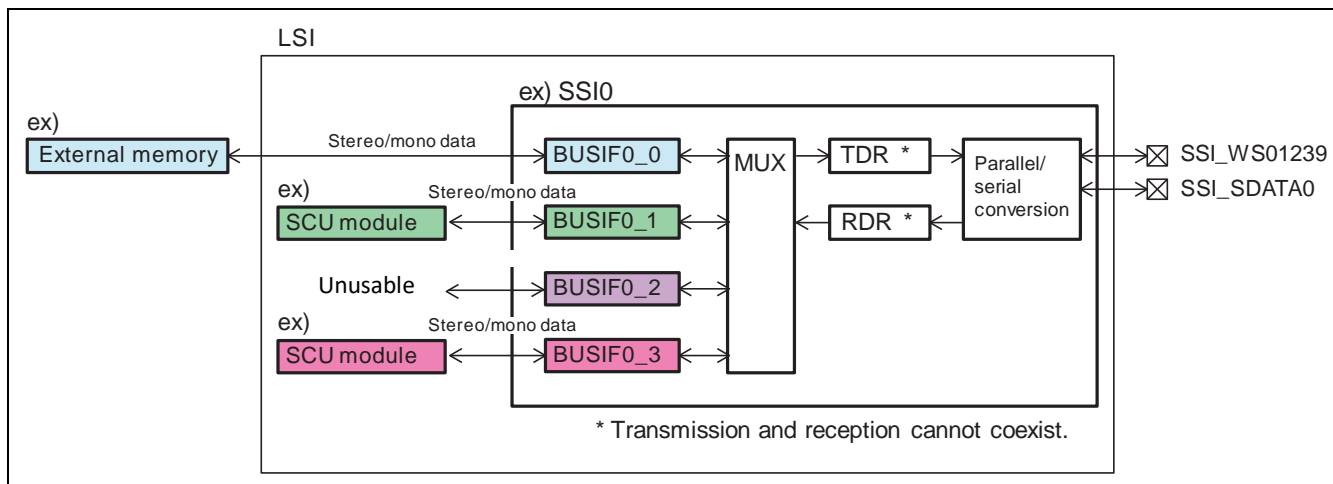


Figure 40.17 TDM Split Mode

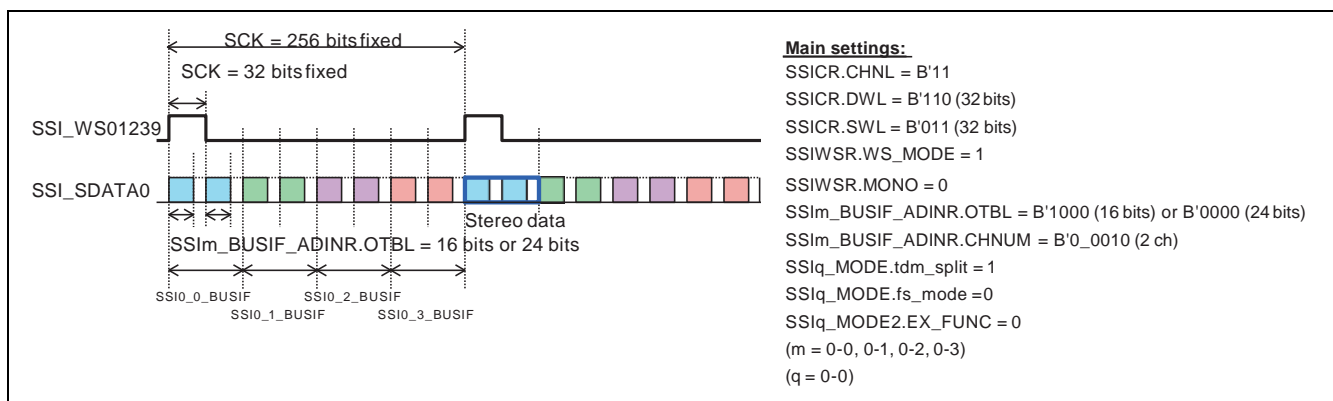


Figure 40.18 TDM Split Mode (Stereo x 4)

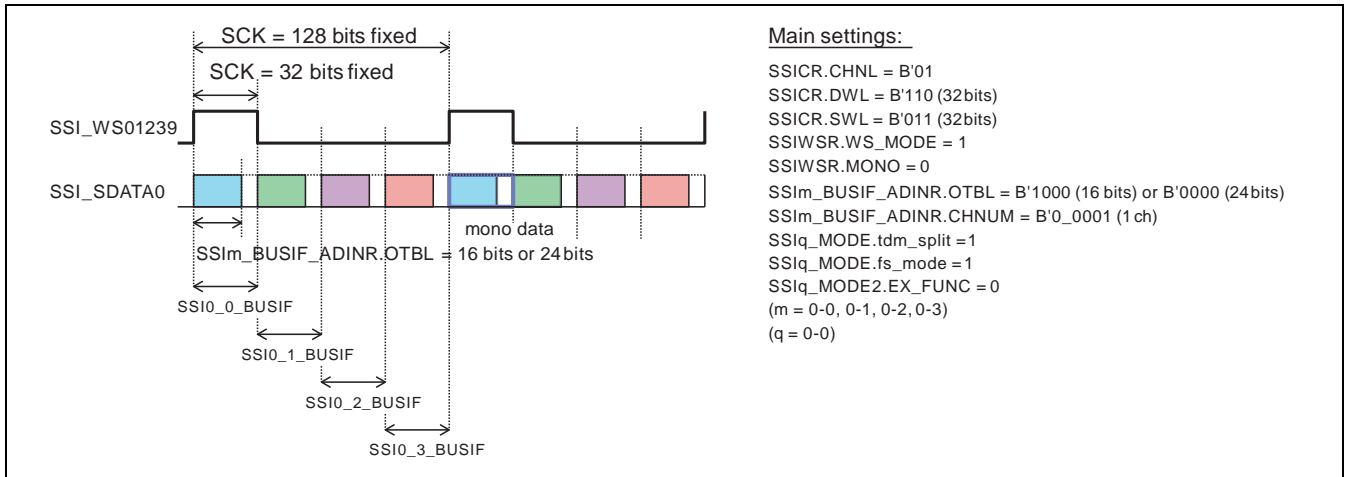


Figure 40.19 TDM Split Mode (Monaural x 4)

- Notes:
1. The SWL and DWL bits in SSICR should be fixed at 32-bit setting.
 2. Transmission and reception cannot coexist.
 3. Data input to BUSIFn 0 to 3 (n = 0, 1, 2, 3, 4, 9) should be operated synchronously with the SSI sampling frequency (WS signal cycle).
 4. Setting of SSIm_BUSIF_ADINR.OTBL = B'1000 (16bit) and SSIm_BUSIF_ADINR.OTBL = B'0000 (24bit) cannot coexist within the same SSI.

40.3.8 TDM Format 16ch Function (TDM 16ch Mode) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

The TDM format 16ch function (TDM 16ch mode) is similar to basic function. It is correspondence to TDM-16channel.

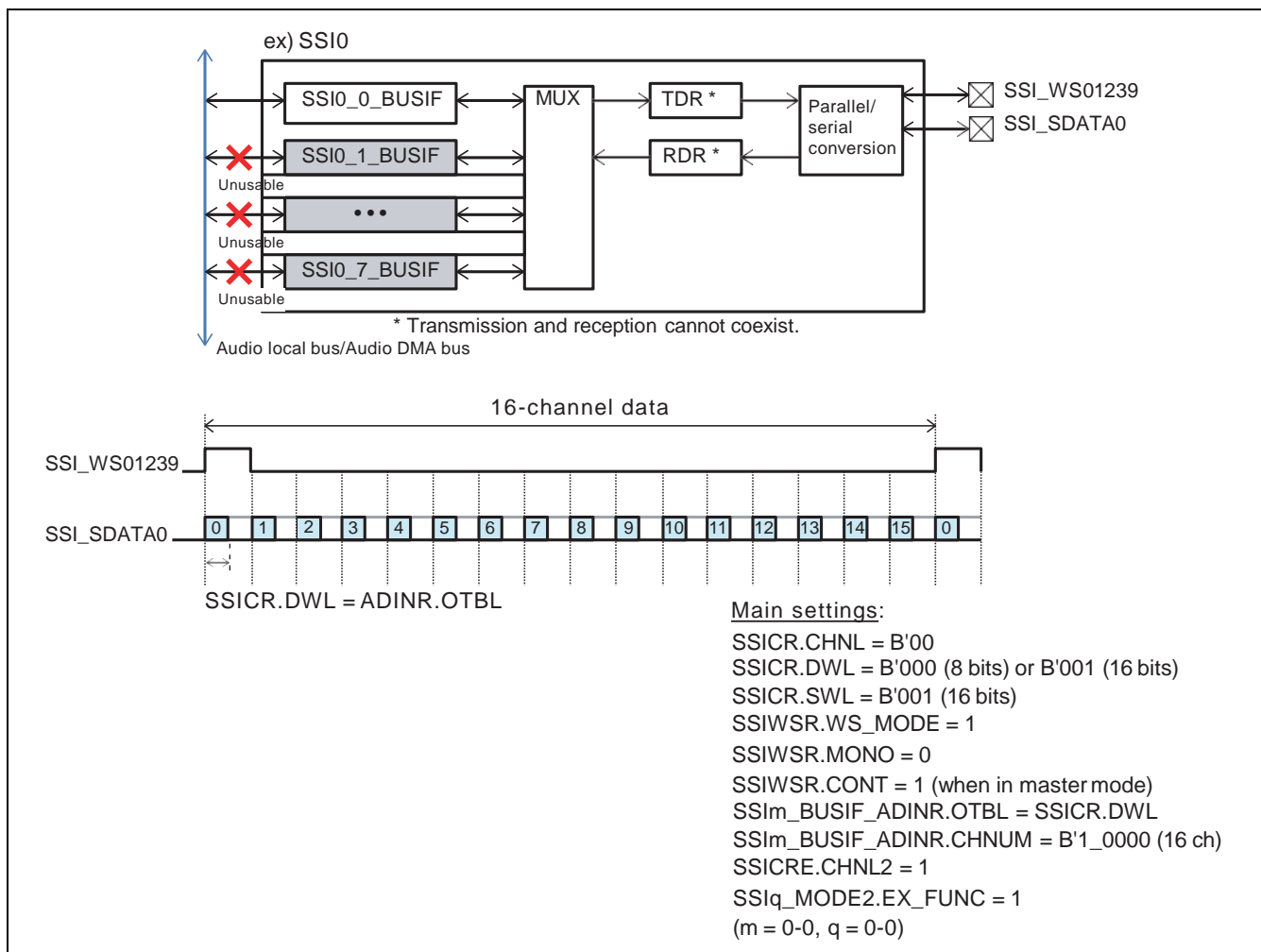


Figure 40.20 TDM 16ch Mode

40.3.9 TDM Format Extend Split Function (TDM Ex-Split Mode) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

The TDM format extend split function (TDM ex-split mode) allows connecting eight independent stereo or multi data inside the LSI and outputting the data in the TDM format, and splitting TDM input data into eight independent stereo or multi data and distributing the data inside the LSI. This function allows TDM-16ch or TDM-8ch (TDM-6ch, 4ch are not supported). This function is supported by SSI0, SSI1, SSI2, SSI3, SSI4 and SSI9. Setting to share place in the stream data with multiple BUSIF channels is prohibited. For example, when BUSIF0-0 used as 6channel, BUSIF0-1 and BUSIF0-2 cannot be used.

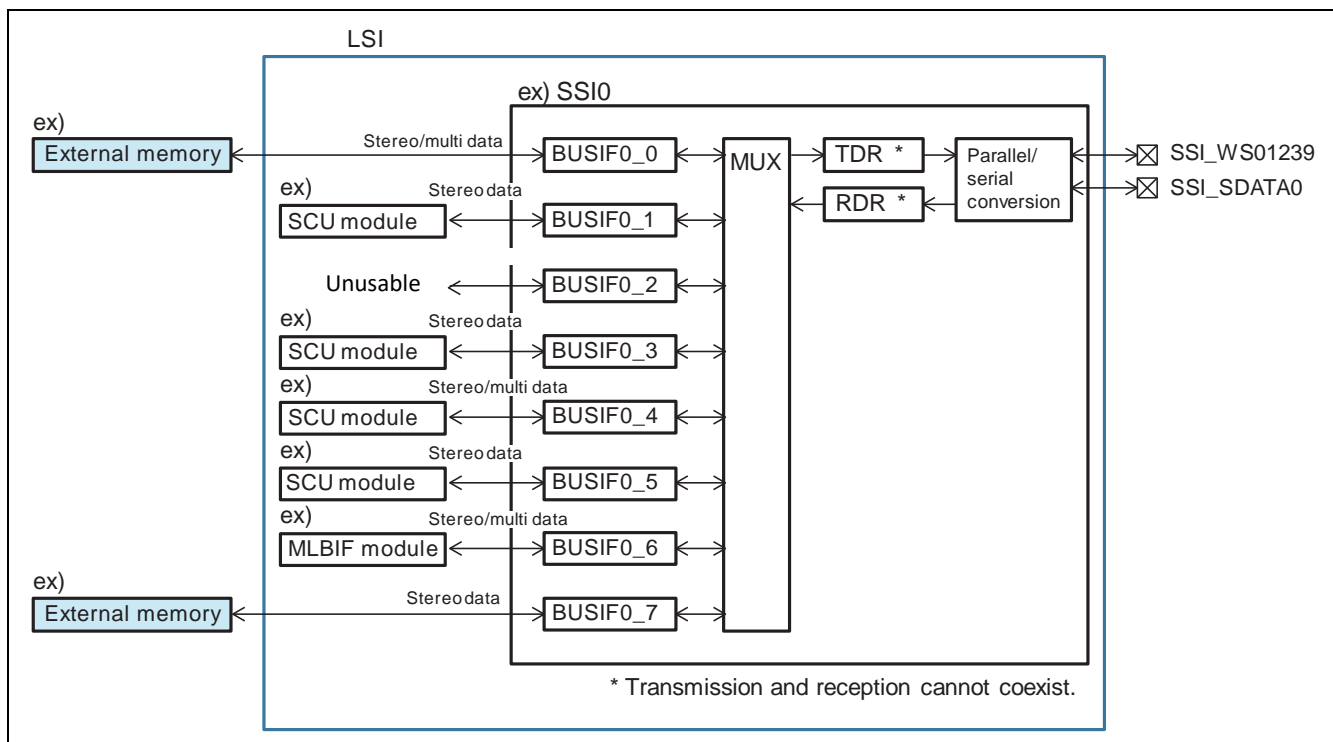
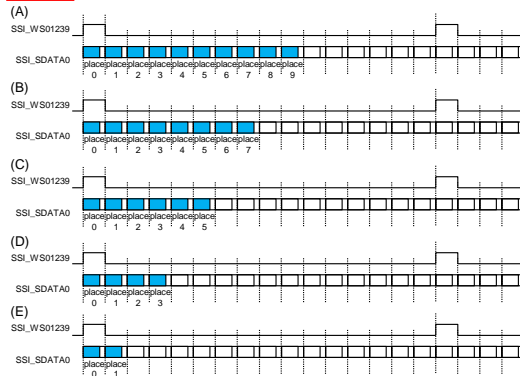


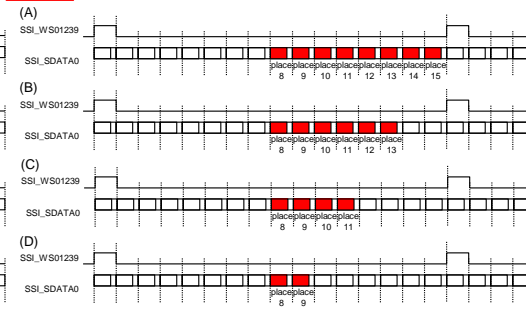
Figure 40.21 TDM Ex-Split Mode

TDM 16 ch ex-split mode (sound I/F format)

BUSIF0_0 : Correspond to 2/4/6/8/10 ch



BUSIF0_4 : Correspond to 2/4/6/8 ch



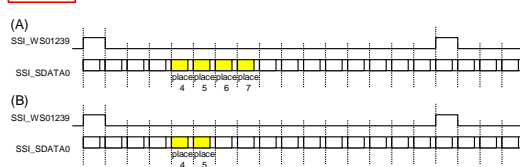
BUSIF0_1 : Correspond to 2 ch



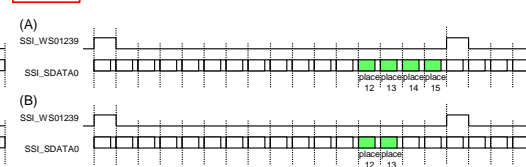
BUSIF0_5 : Correspond to 2 ch



BUSIF0_2 : Correspond to 2 ch/4 ch



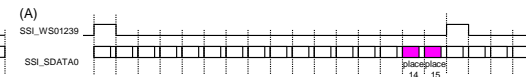
BUSIF0_6 : Correspond to 2 ch/4 ch



BUSIF0_3 : Correspond to 2 ch



BUSIF0_7 : Correspond to 2 ch



Main Setting
 SSICR.CHNL = B'00
 SSICR.DWL = B'000 (8-bit) or B'001 (16-bit)
 SSICR.SWL = B'001 (16-bit)
 SSIWSR.WS_MODE = 1
 SSIWSR.MONO = 0
 SSIm_BUSIF_ADINR.OTB_L= SSICR.DWL
 SSIm_BUSIF_ADINR.CHNUM = *
 SSIq_MODE.tdm_split = 1
 SSIq_MODE.fs_mode = 0
 SSICRE.CHNL2 = 1
 SSIq_MODE2.EX_FUN C= 1
 *: BUSIF0_0 = B'0_0010/B'0_0100/B'0_0110/B'0_1000/B'0_1010
 BUSIF0_1 = B'0_0010
 BUSIF0_2 = B'0_0010/B'0_0100
 BUSIF0_3 = B'0_0010
 BUSIF0_4 = B'0_0010/B'0_0100/B'0_0110/B'0_1000
 BUSIF0_5 = B'0_0010
 BUSIF0_6 = B'0_0010/B'0_0100
 BUSIF0_7 = B'0_0010

Figure 40.22 TDM 16ch ex-split mode (TDM 16ch ex-split)

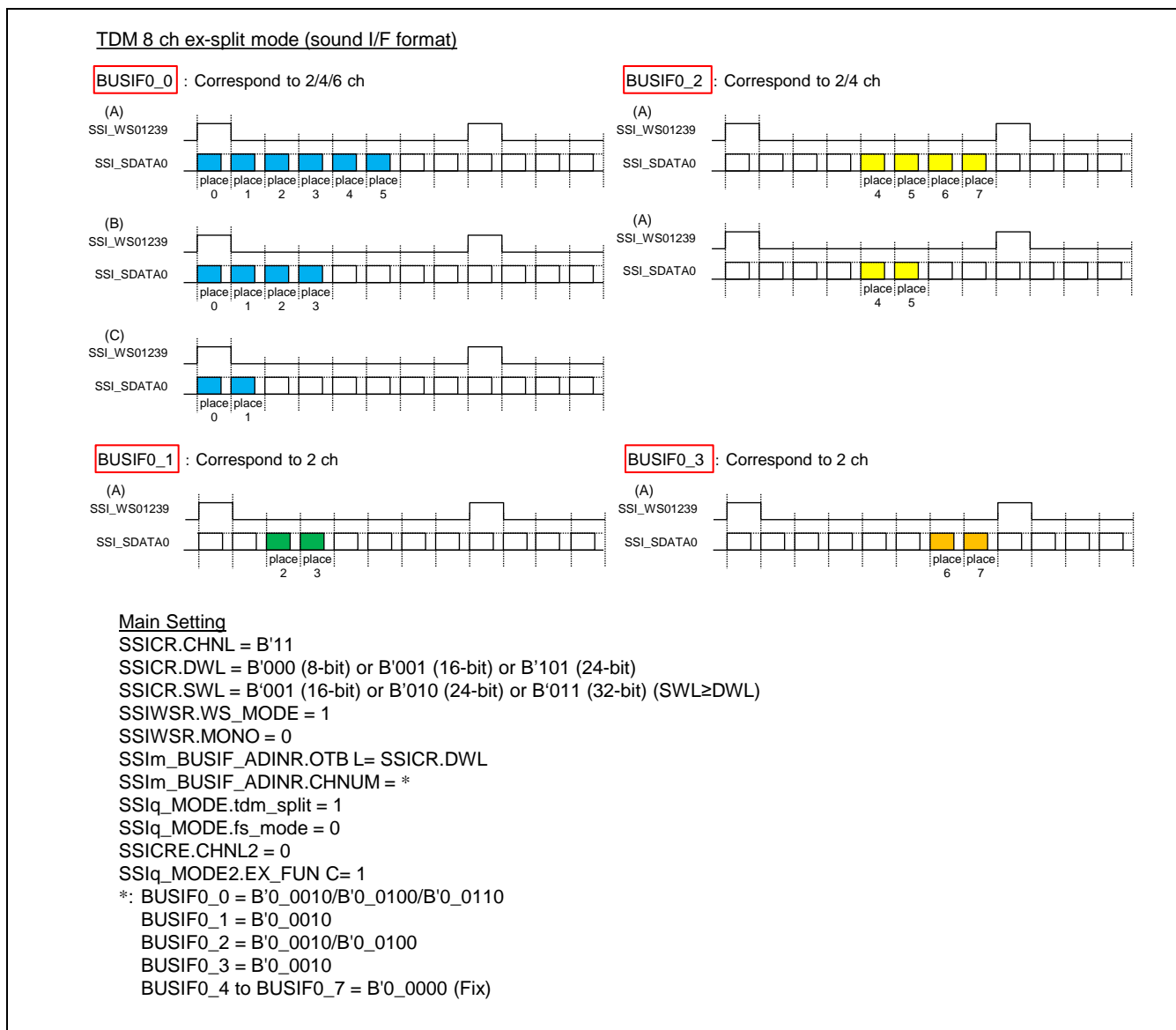


Figure 40.23 TDM 16ch ex-split mode (TDM 8ch split)

- Notes:
1. Transmission and reception cannot coexist.
 2. Data input to BUSIFn 0 to 7 (n = 0, 1, 2, 3, 4, 9) should be operated synchronously with the SSI sampling frequency (WS signal cycle).
 3. Another setting of SSIm_BUSIF_ADINR.OTBL cannot coexist within the same SSI.
(ex. Setting of SSIm_BUSIF_ADINR.OTBL = B'1000 (16bit) and SSIm_BUSIF_ADINR.OTBL = B'0000 (24bit) cannot coexist within the same SSI.)

40.3.10 TDM Format Extend Split Function (TDM Ex-Split Mode) [RZ/G2E]

The TDM format extend split function (TDM ex-split mode) allows connecting eight independent stereo or multi data inside the LSI and outputting the data in the TDM format, and splitting TDM format input data into four independent stereo or multi data and distributing the data inside the LSI. This function allows TDM-8ch (TDM-6ch, 4ch are not supported). This function is supported by SSI0, SSI1, SSI2, SSI3, SSI4 and SSI9. Setting to share place in the stream data with multiple BUSIF channels is prohibited. For example, when BUSIF0_0 used as 6channel, BUSIF0-1 and BUSIF0-2 can not be used.

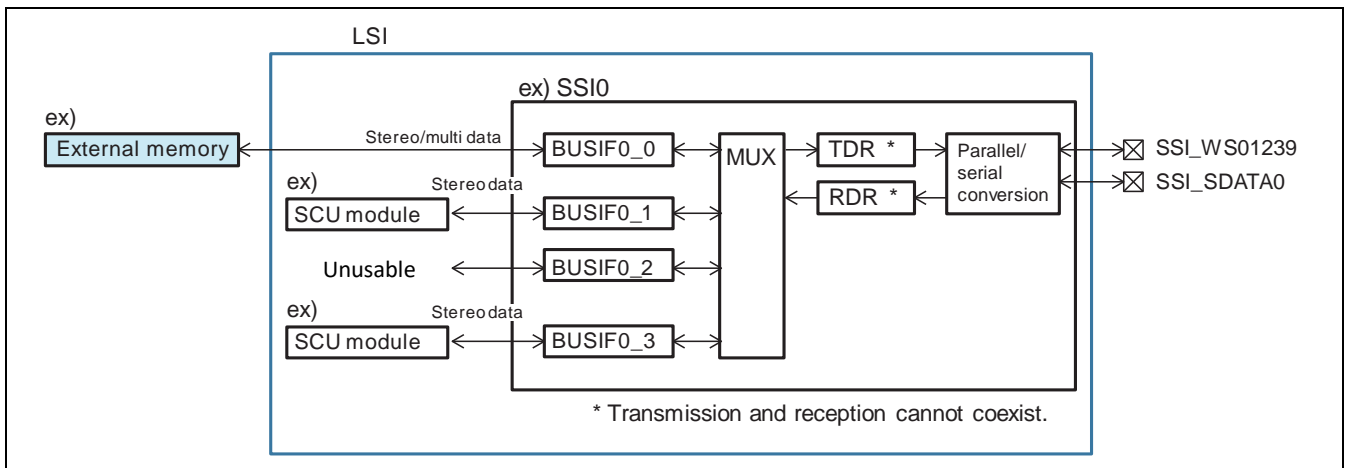


Figure 40.24 TDM Ex-Split Mode

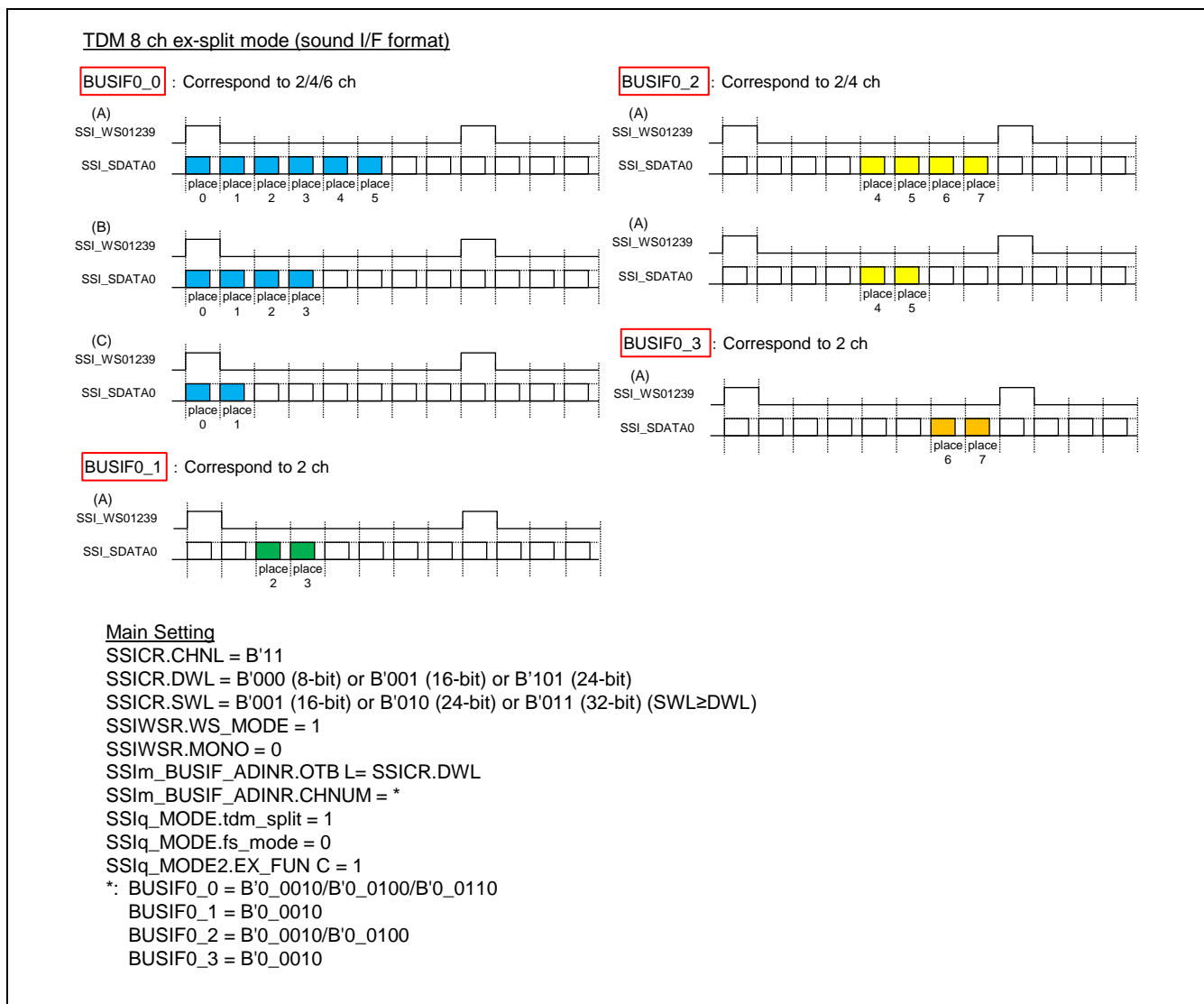


Figure 40.25 TDM 8ch ex-split mode (TDM 8ch split)

- Notes:
1. Transmission and reception cannot coexist.
 2. Data input to BUSIFn 0 to 3 (n = 0, 1, 2, 3, 4, 9) should be operated synchronously with the SSI sampling frequency (WS signal cycle).
 3. Another setting of SSIIm_BUSIF_ADINR.OTBL cannot coexist within the same SSI.
(ex. Setting of SSIIm_BUSIF_ADINR.OTBL = B'1000 (16bit) and SSIIm_BUSIF_ADINR.OTBL = B'0000 (24bit) cannot coexist within the same SSI.)

40.3.11 SSI Pin Connections

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Multiple SSI modules (in combinations of SSI0, SSI1, SSI2, SSI3, and SSI9, combination of SSI3, SSI4, and SSI9, and combination of SSI7 and SSI8) can operate in synchronization with sharing of the same SSI_SCK and SSI_WS signals. Figure 40.26 to Figure 40.30 show how to connect the pins for each combination of SSIs.

In the case of the combination of SSI0, SSI1, SSI2, SSI3, and SSI9, SSI0 as the master should start transfer first thus allowing output on SSI_WS01239 and SSI_SCK01239, after which SSI1, SSI2, SSI3, and SSI9 as the slaves should be used. If the master-side transfer is stopped by setting the CONT bit in SSIWSR to 0, the slave-side SSI1, SSI2, SSI3, and SSI9 should be stopped because output through SSI_WS01239 is not possible. This also applies to the master and slaves in the combination of SSI3, SSI4, and SSI9, and to the master and slave in the combination of SSI7 and SSI8.

[RZ/G2E]

Multiple SSI modules (in combinations of SSI0, SSI1, SSI2, SSI3, and SSI9, combination of SSI3, SSI4, and SSI9, and combination of SSI7 and SSI8) can operate in synchronization with sharing of the same SSI_SCK and SSI_WS signals. Figure 40.31 to Figure 40.35 show how to connect the pins for each combination of SSIs.

In the case of the combination of SSI0, SSI1, SSI2, SSI3, and SSI9, SSI0 as the master should start transfer first thus allowing output on SSI_WS01239 and SSI_SCK01239, after which SSI1, SSI2, SSI3, and SSI9 as the slaves should be used. If the master-side transfer is stopped by setting the CONT bit in SSIWSR to 0, the slave-side SSI1, SSI2, SSI3, and SSI9 should be stopped because output through SSI_WS01239 is not possible. This also applies to the master and slaves in the combination of SSI3, SSI4, and SSI9, and to the master and slave in the combination of SSI7 and SSI8.

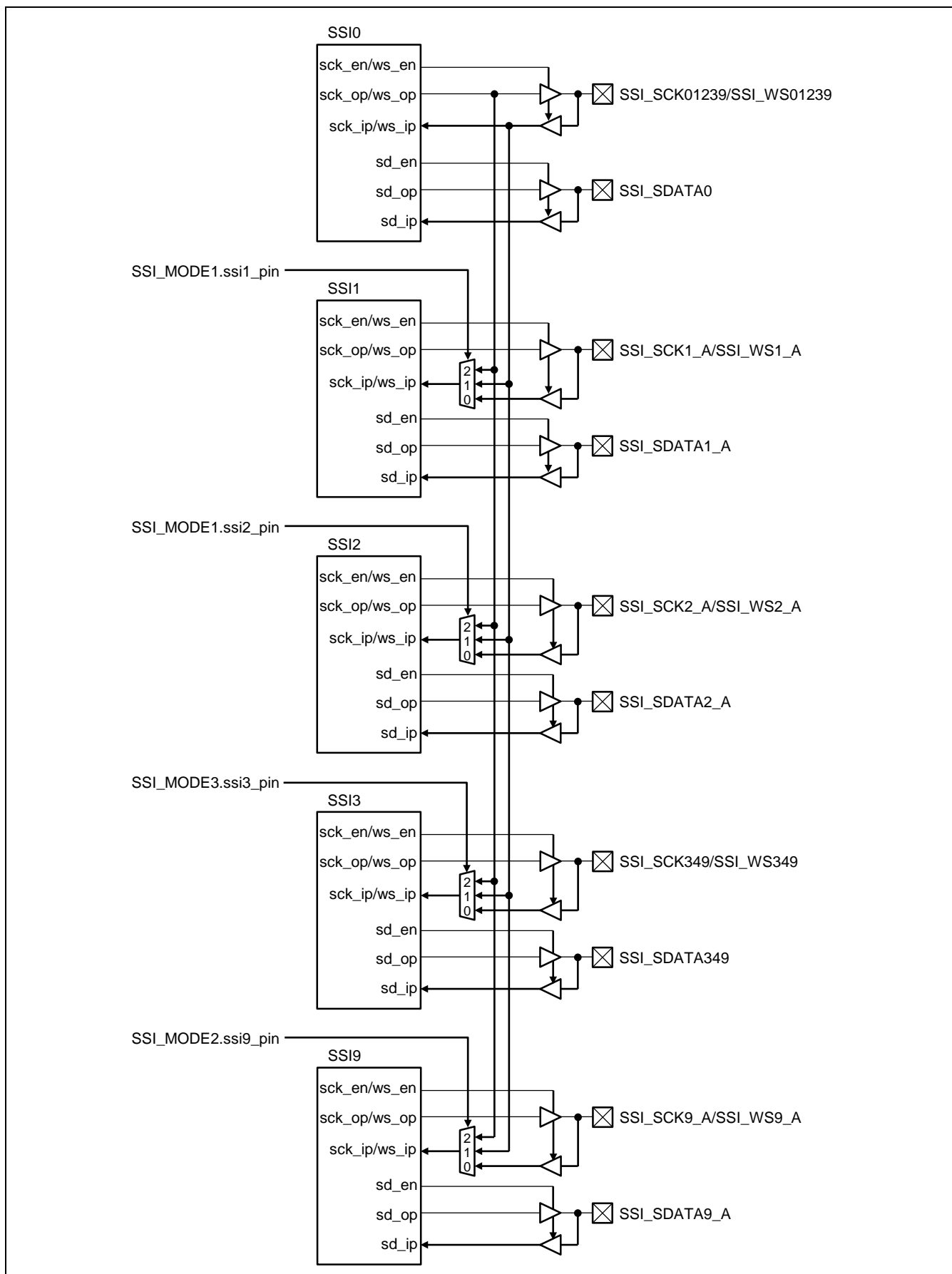


Figure 40.26 Pin Connections for Combination of SSI0, SSI1, SSI2, SSI3, and SSI9 (Group A)
 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

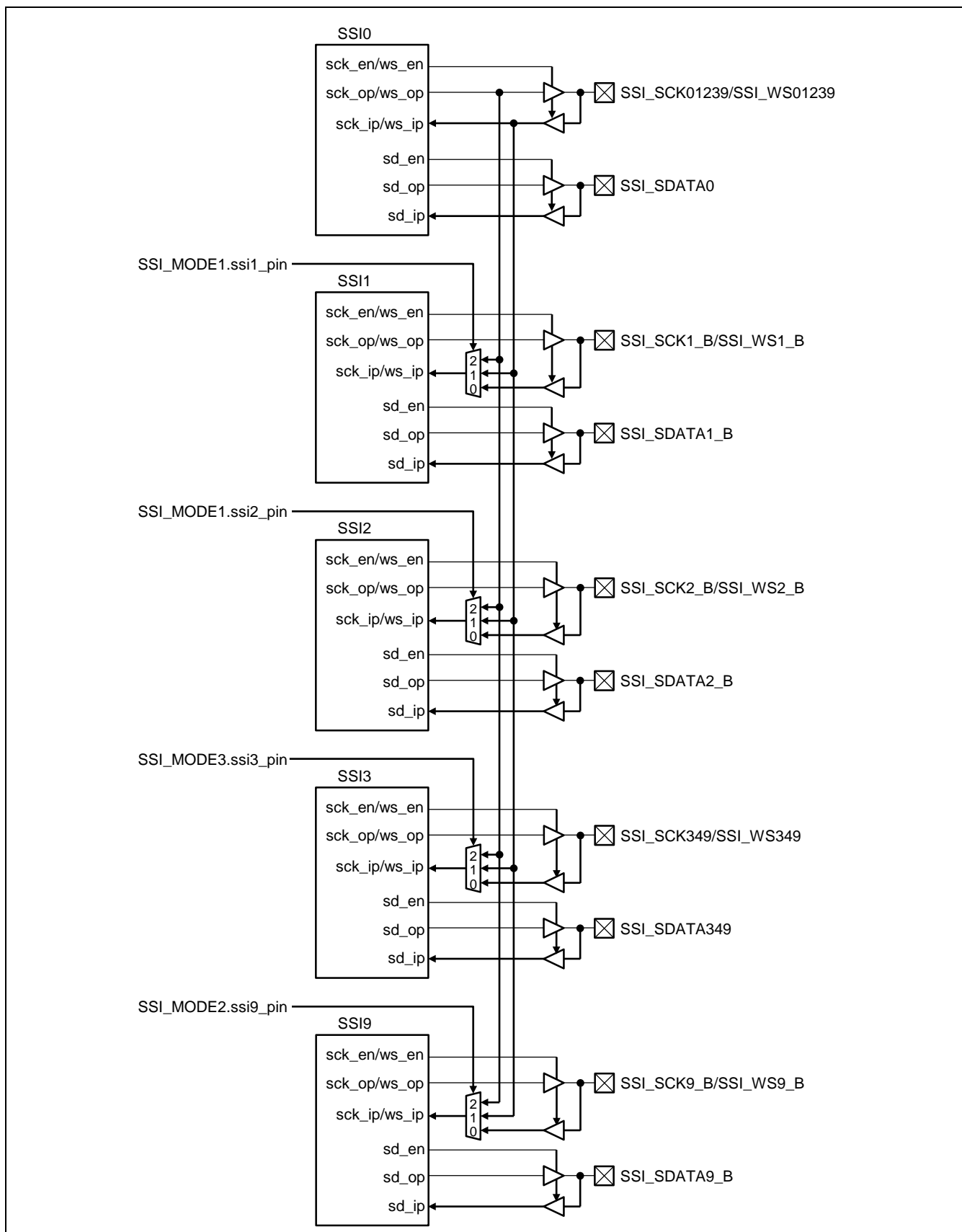


Figure 40.27 Pin Connections for Combination of SSI0, SSI1, SSI2, SSI3, and SSI9 (Group B)
[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

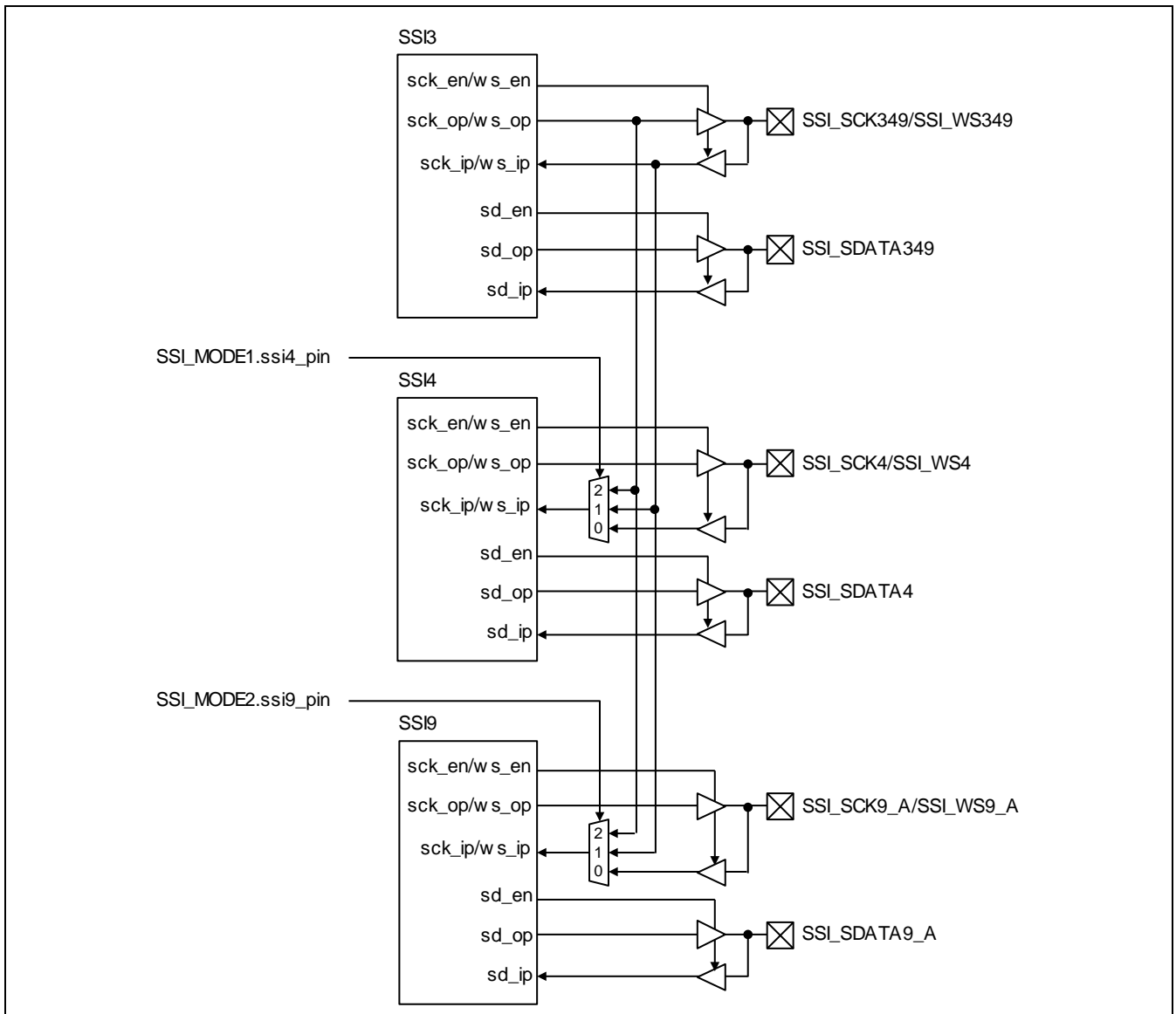


Figure 40.28 Pin Connections for Combination of SSI3, SSI4, and SSI9 (Group A)
[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

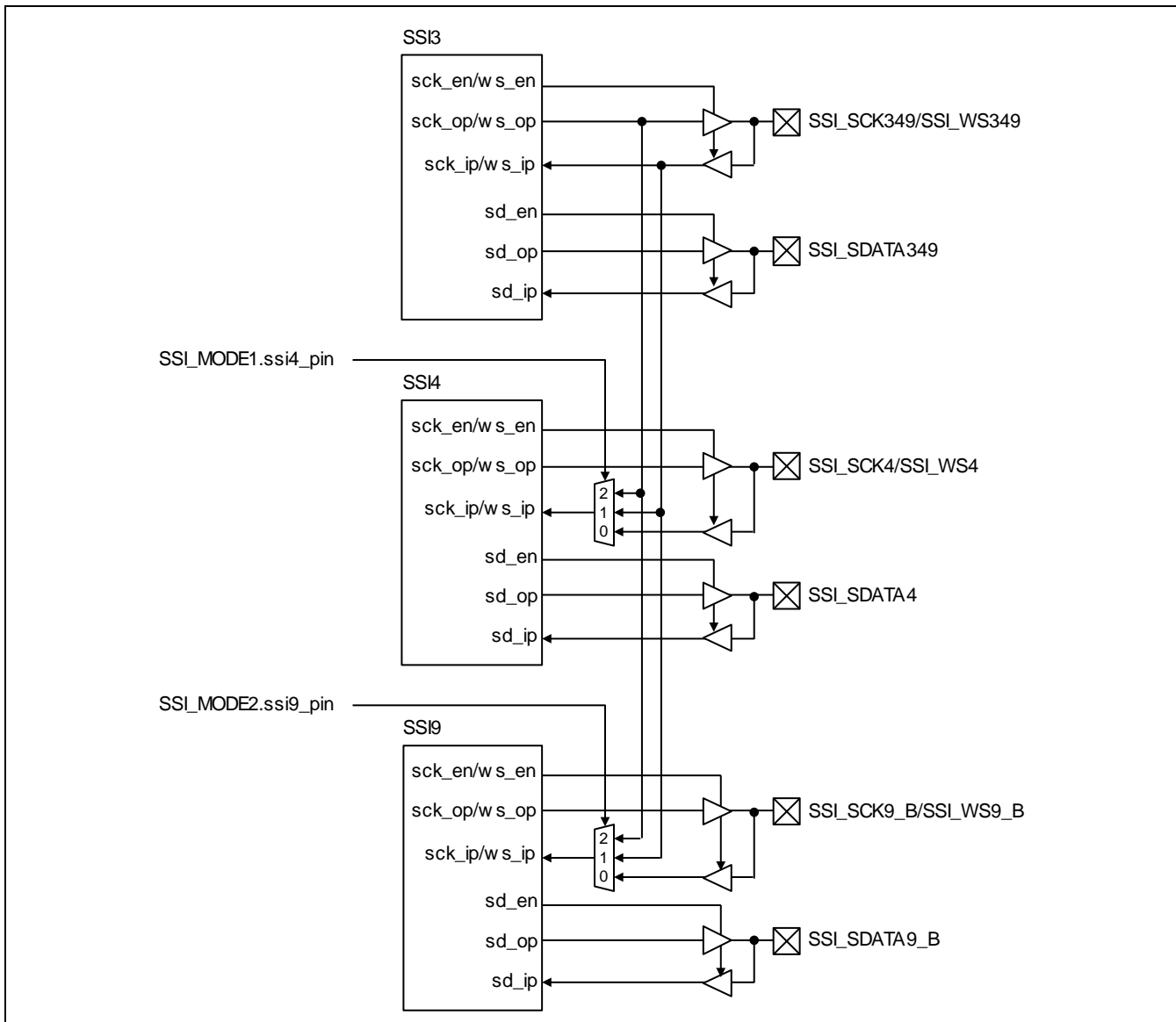


Figure 40.29 Pin Connections for Combination of SSI3, SSI4, and SSI9 (Group B)
[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Note: The above configurations in Figure 40.28 and Figure 40.29 are prohibited if SSI0 SCK/WS is selected for SSI3 as shown in Figure 40.26 and Figure 40.27.

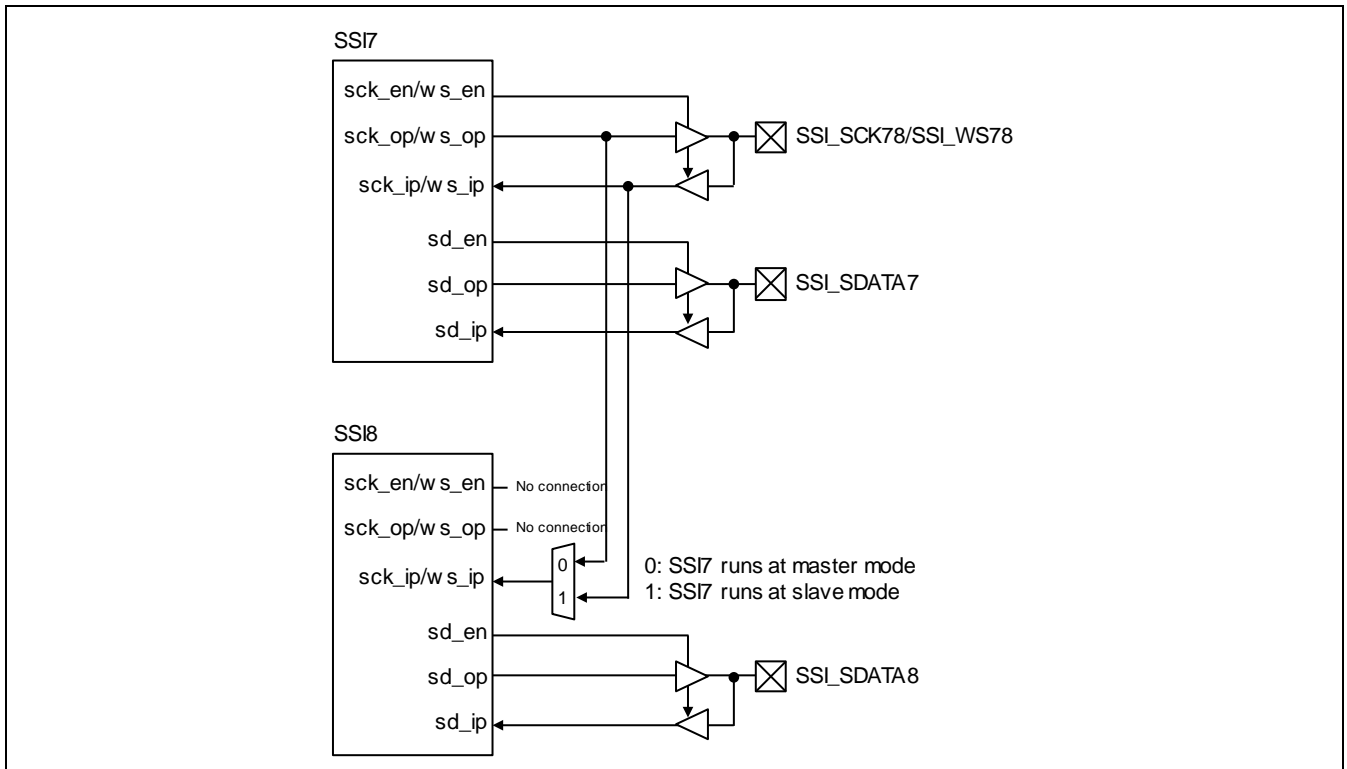


Figure 40.30 Pin Connections for Combination of SSI7 and SSI8
 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

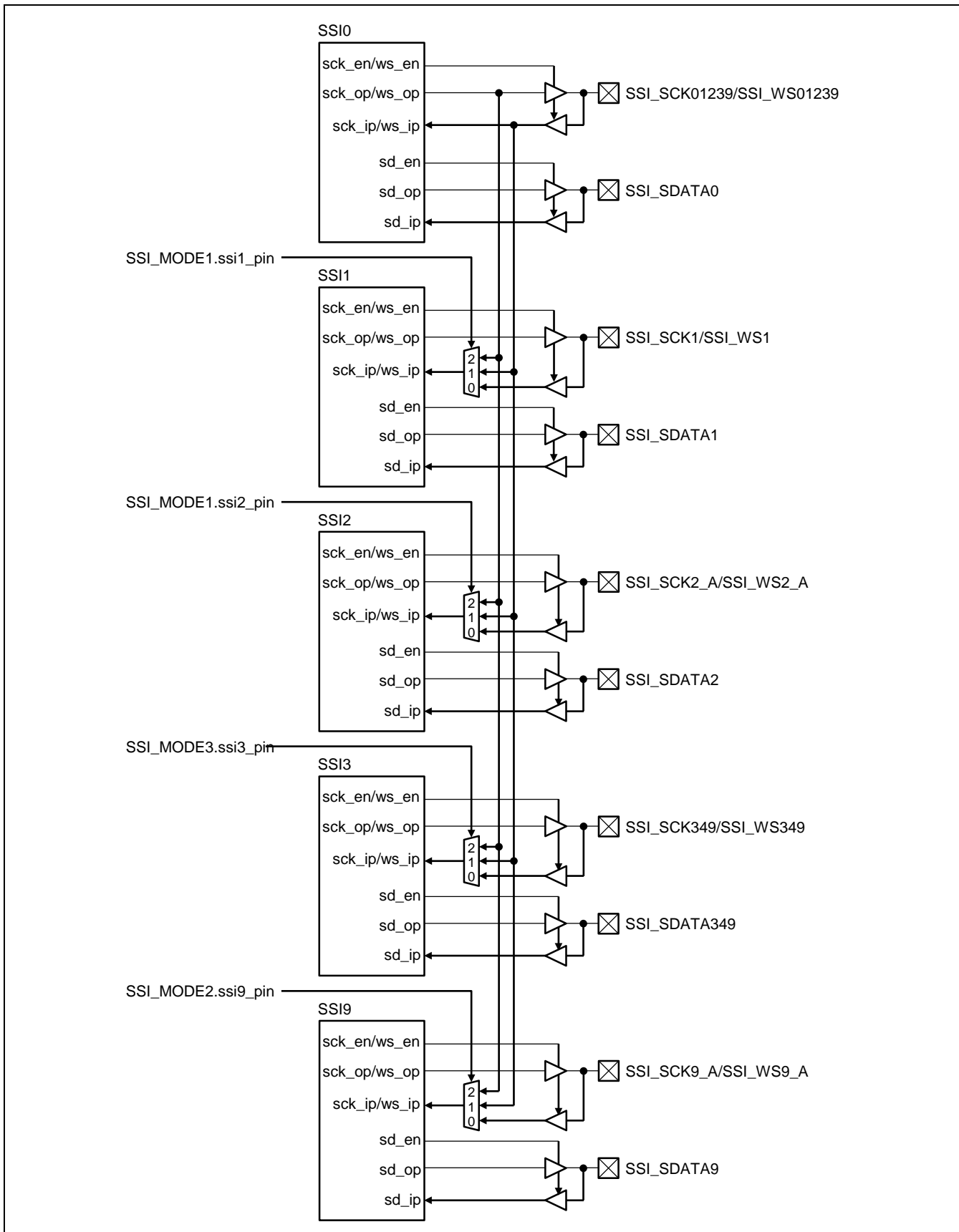


Figure 40.31 Pin Connections for Combination of SSI0, SSI1, SSI2, SSI3, and SSI9 (Group A)
[RZ/G2E]

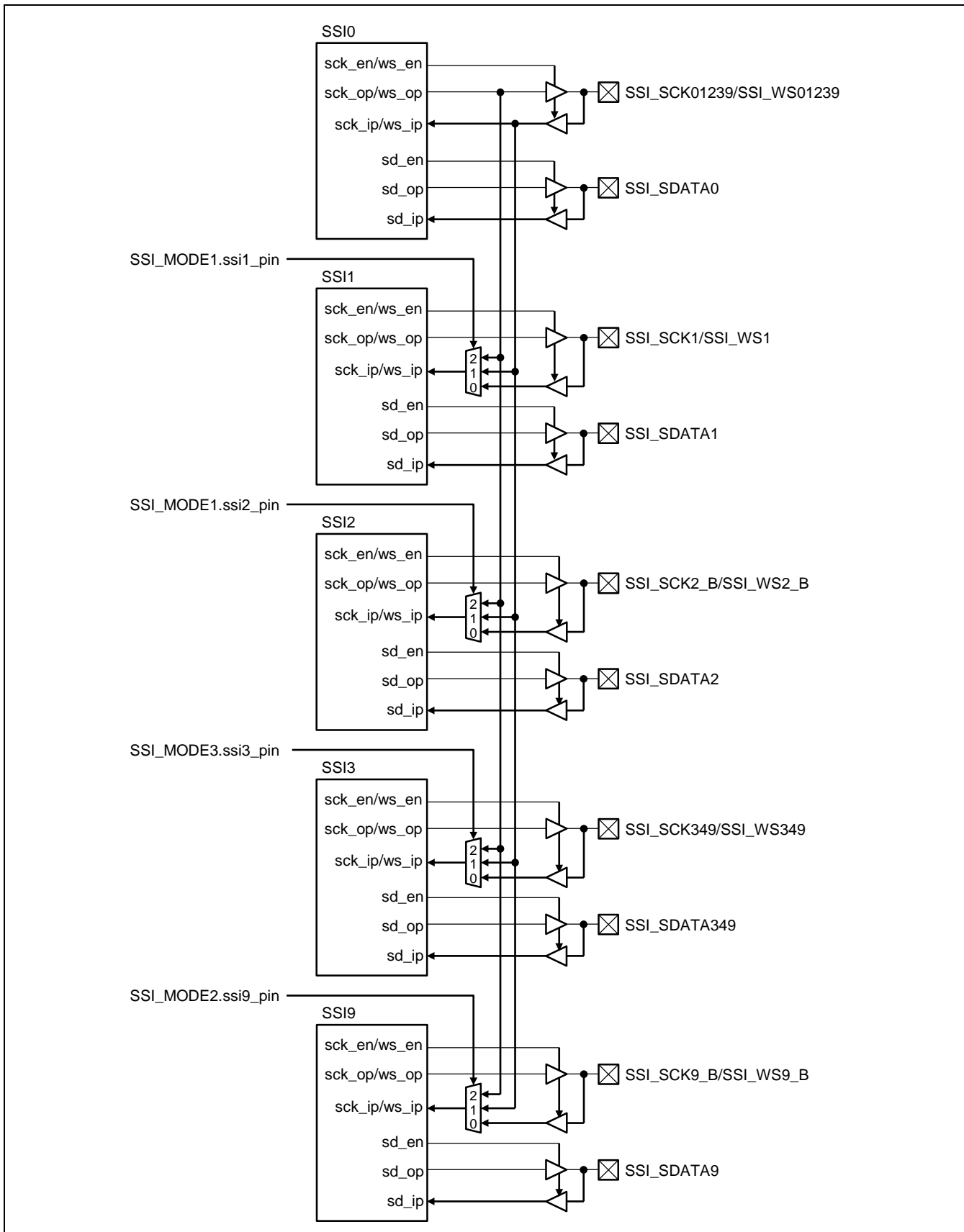


Figure 40.32 Pin Connections for Combination of SSI0, SSI1, SSI2, SSI3, and SSI9 (Group B)
 [RZ/G2E]

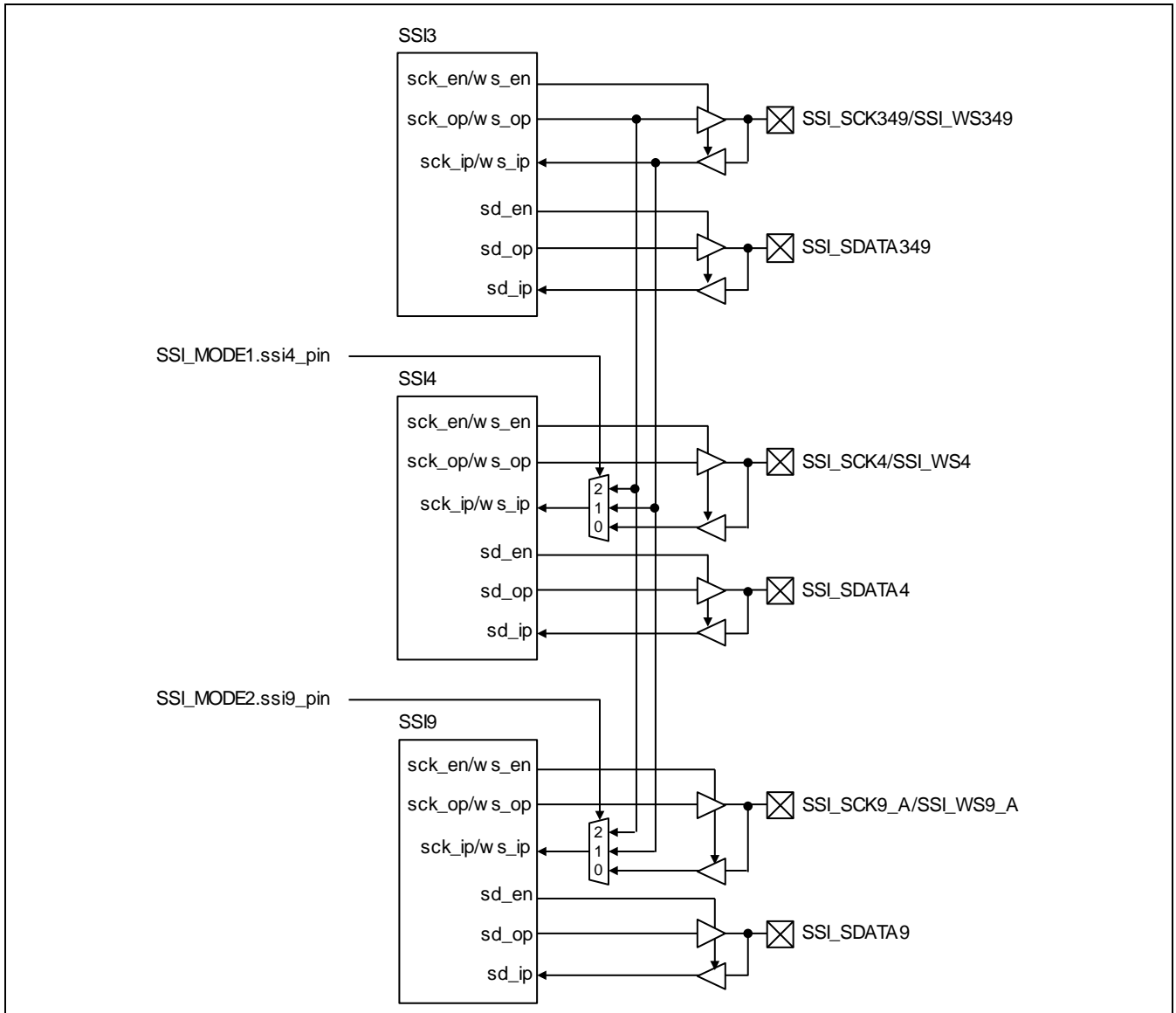


Figure 40.33 Pin Connections for Combination of SSI3, SSI4, and SSI9 (Group A) [RZ/G2E]

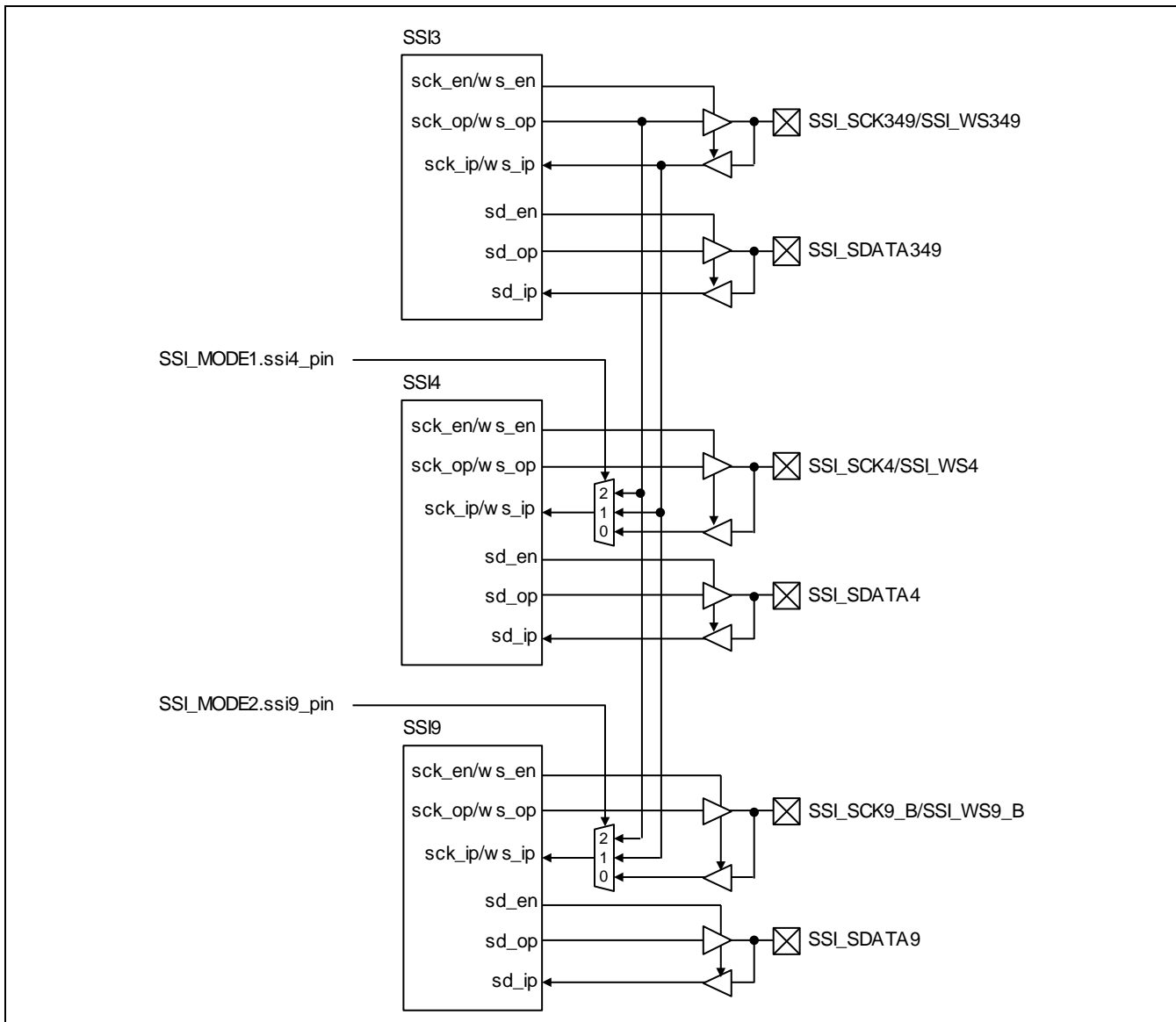


Figure 40.34 Pin Connections for Combination of SSI3, SSI4, and SSI9 (Group B) [RZ/G2E]

Note: The above configurations in Figure 40.33 and Figure 40.34 are prohibited if SSI0 SCK/WS is selected for SSI3 as shown in Figure 40.31 and Figure 40.32.

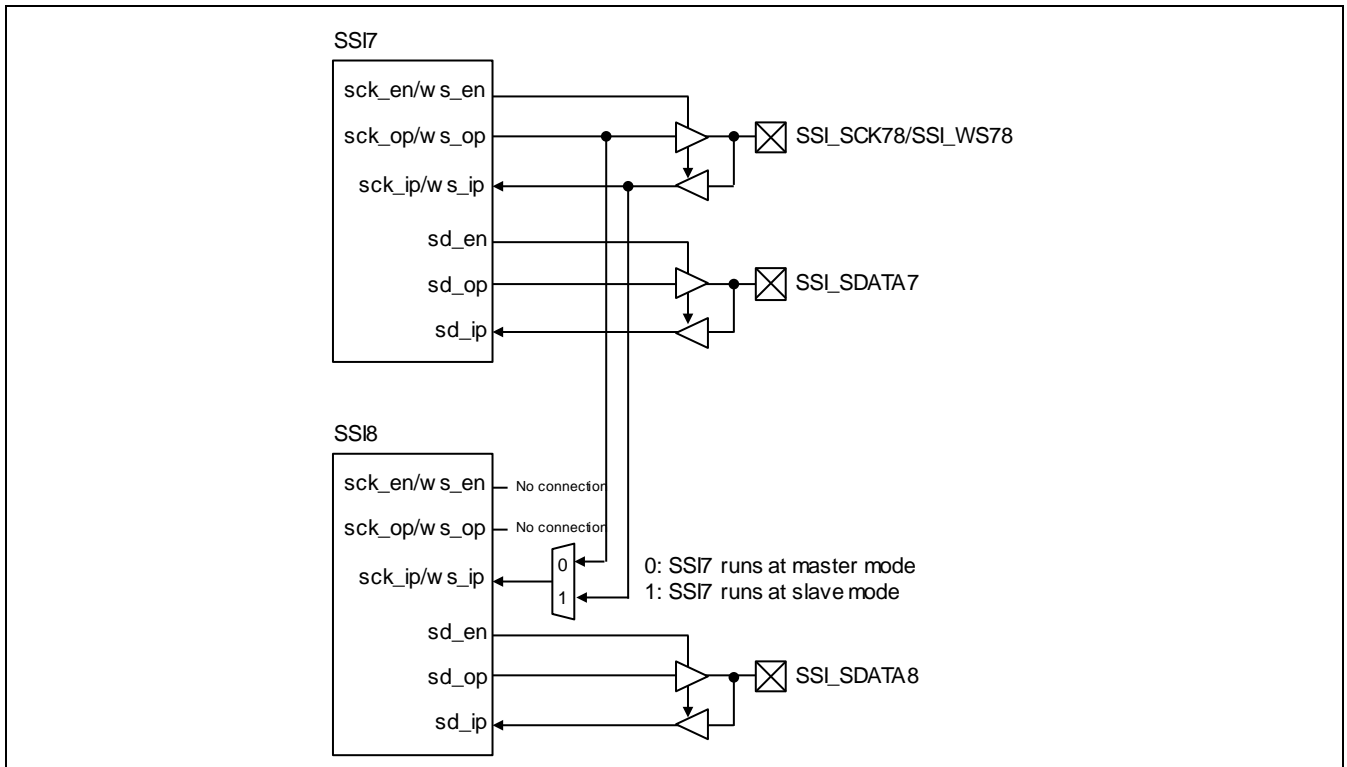


Figure 40.35 Pin Connections for Combination of SSI7 and SSI8 [RZ/G2E]

40.3.12 HDMI Signal Selection [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

SSIU selects SSI signals (SSI_SCK, SSI_WS, and SSI_SDATA (up to from four SSIs)) that is transmitted to HDMI-IF.

SSI_SCK and SSI_WS is generated from one SSI. Selectable SSIs for SSI_SDATA depend on which SSI is selected for SSI_SCK and SSI_WS. For example, in case that SSI_SCK and SSI_WS from SSI0 are selected, SSI_SDATA is data from SSI0, SSI1, SSI2, SSI3, or SSI9.

Section 40.2.42 and section 40.2.43 and Table 40.7 explain selectable combination of SSIs.

Only stereo data format is supported. Selected SSI has to perform at master-mode.

Table 40.7 Selectable combination of SSIs

| SSI_SCK/SSI_WS | SSI_SDATA |
|----------------|-----------|
| SSI0 | SSI0 |
| | SSI1 |
| | SSI2 |
| | SSI3 |
| | SSI9 |
| SSI1 | SSI1 |
| SSI2 | SSI2 |
| SSI3 | SSI3 |
| SSI4 | SSI4 |
| SSI5 | SSI5 |
| SSI6 | SSI6 |
| | SSI7 |
| SSI7 | SSI7 |
| | SSI8 |
| SSI9 | SSI9 |

40.3.13 Procedure of SSIU Transfer

Figure 40.36 to Figure 40.33 show the operation flow.

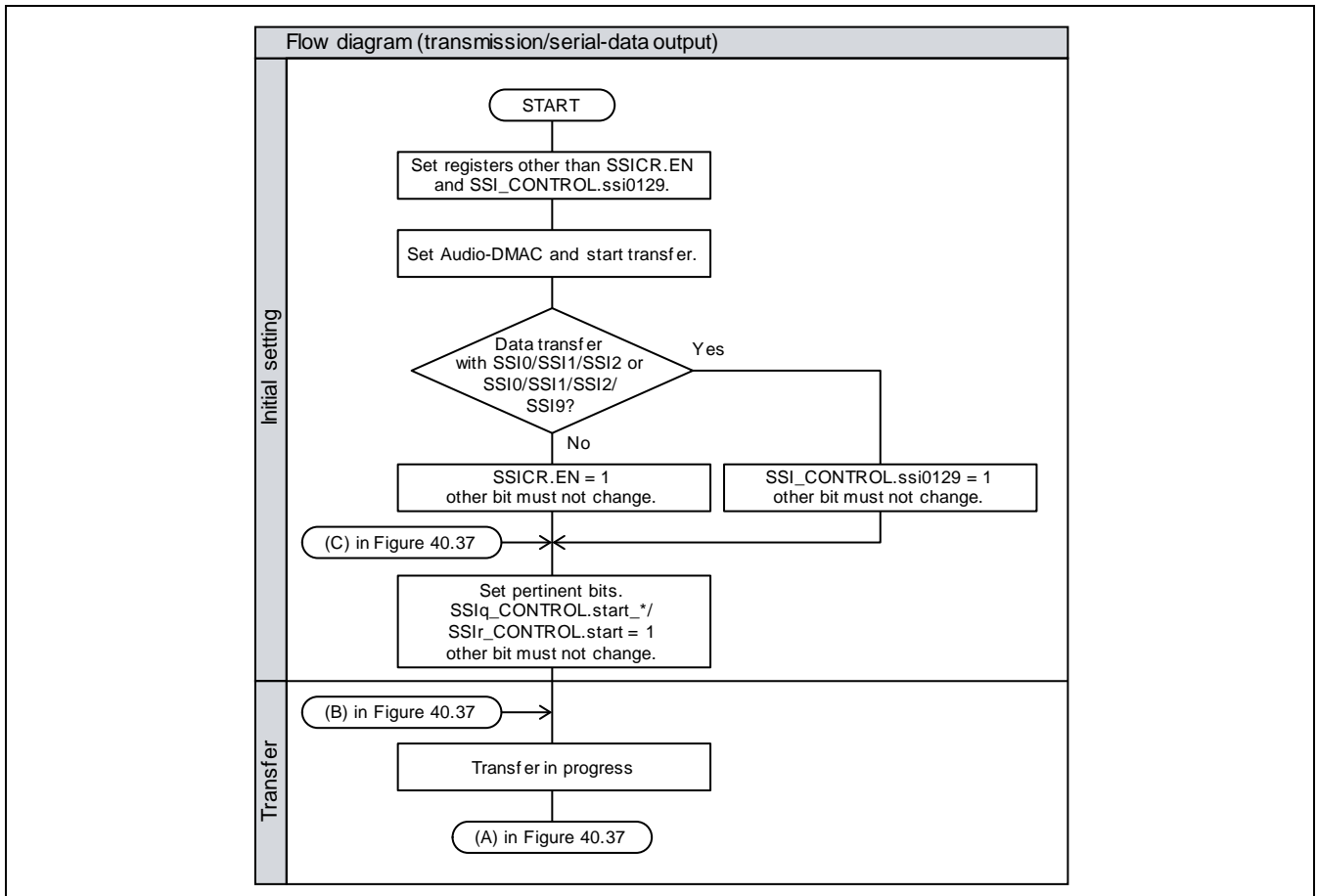


Figure 40.36 Transmission (1)

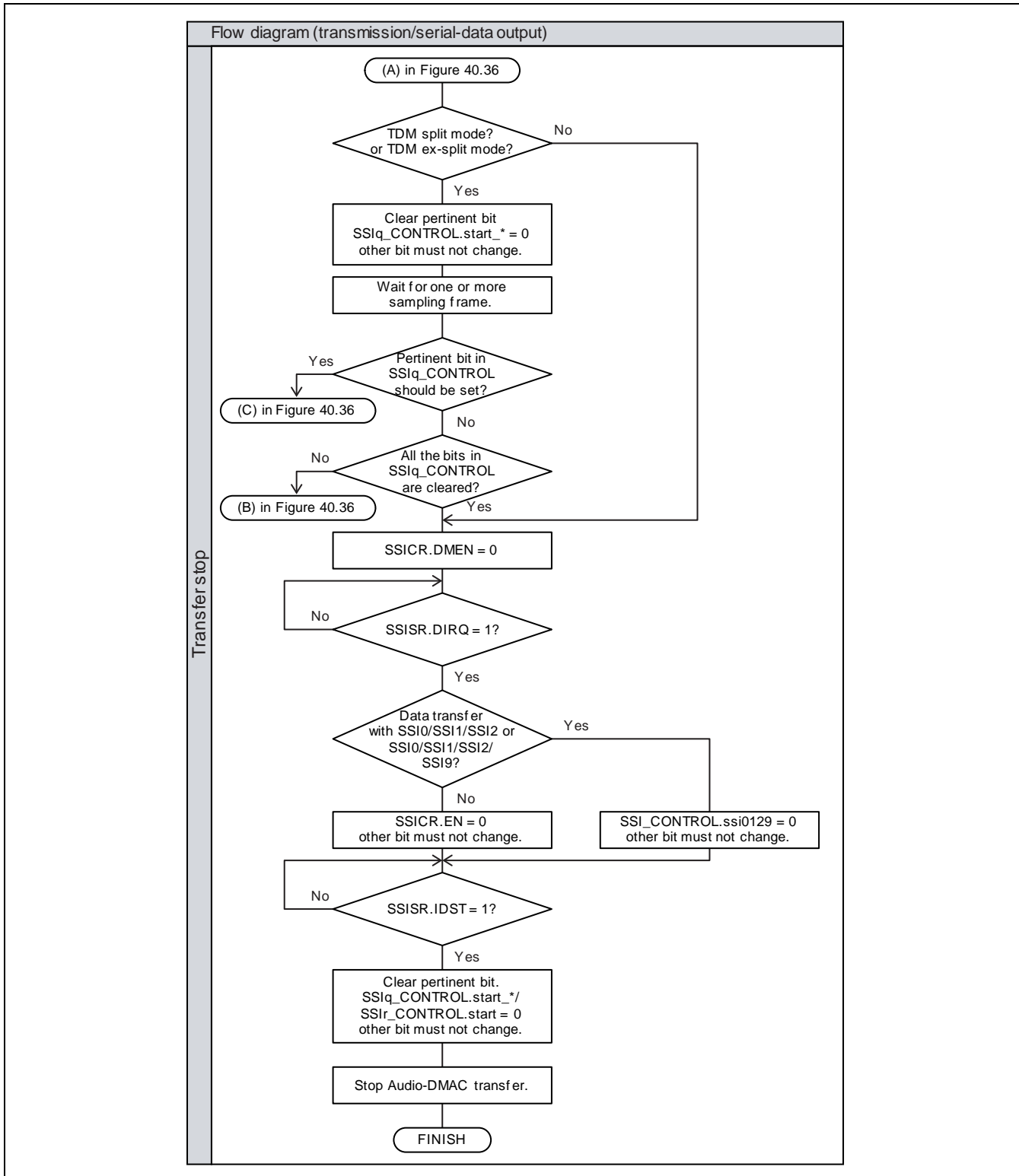


Figure 40.37 Transmission (2)

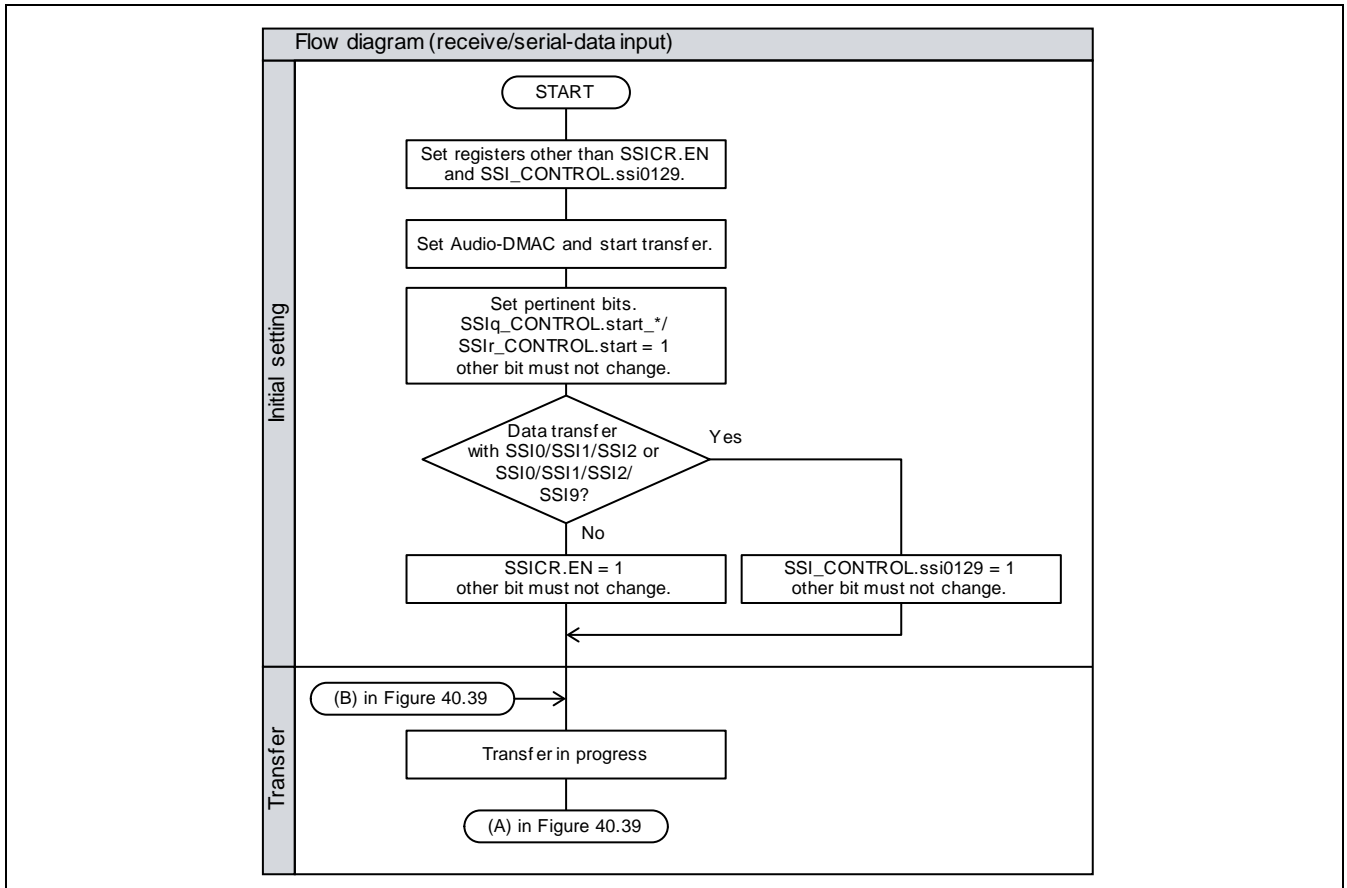


Figure 40.38 Reception (1)

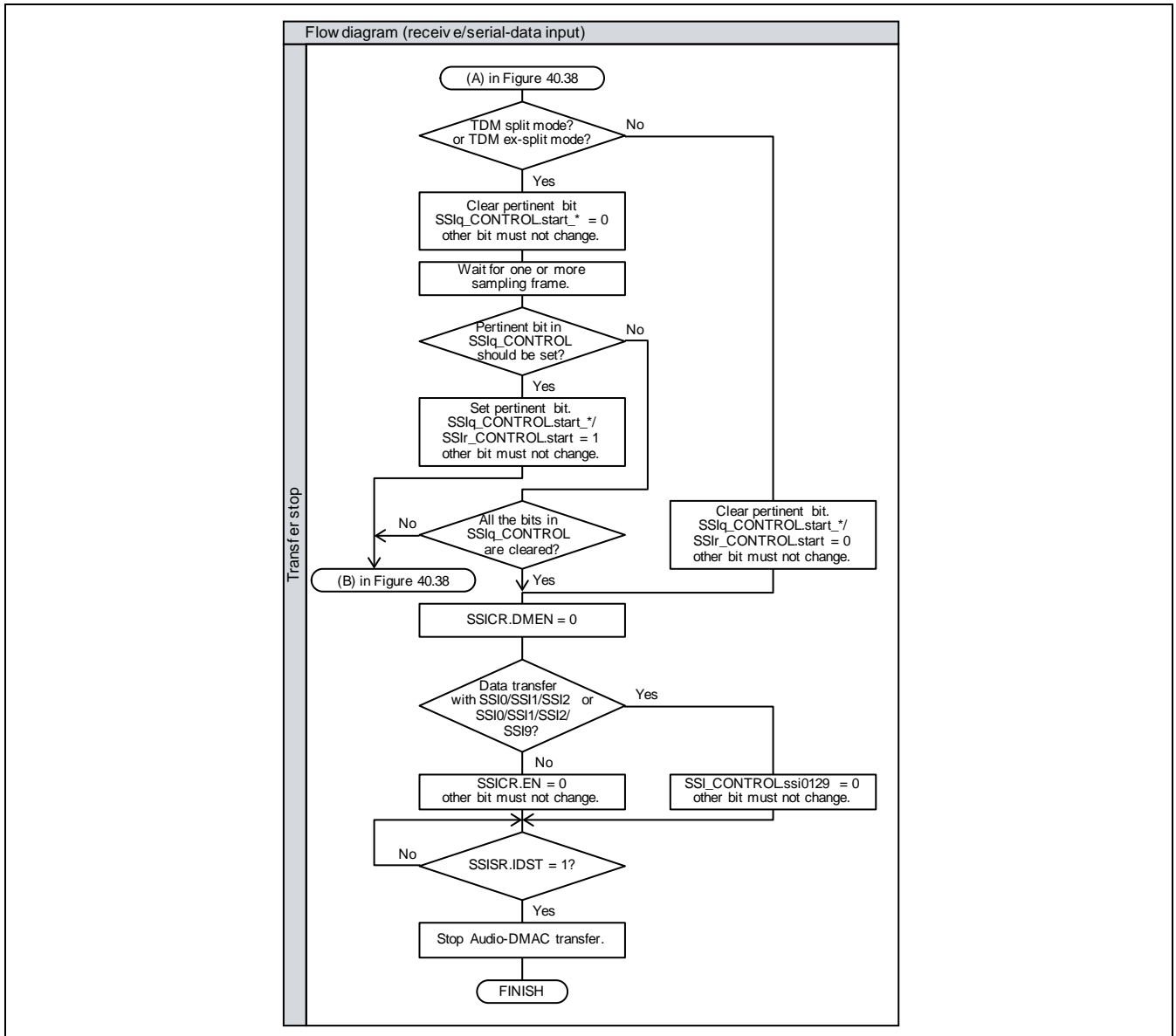


Figure 40.39 Reception (2)

40.3.14 Interrupts of SSI Setting

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

SSI can output interrupt signals when one of several events happen. Figure 40.40 illustrates how SSI generates interrupt signals and Table 40.8 and Table 40.9 indicate combinations of relevant registers for each SSI to generate interrupt signals (refer to explanations of relevant registers (section 40.2) and SSI (section 41))

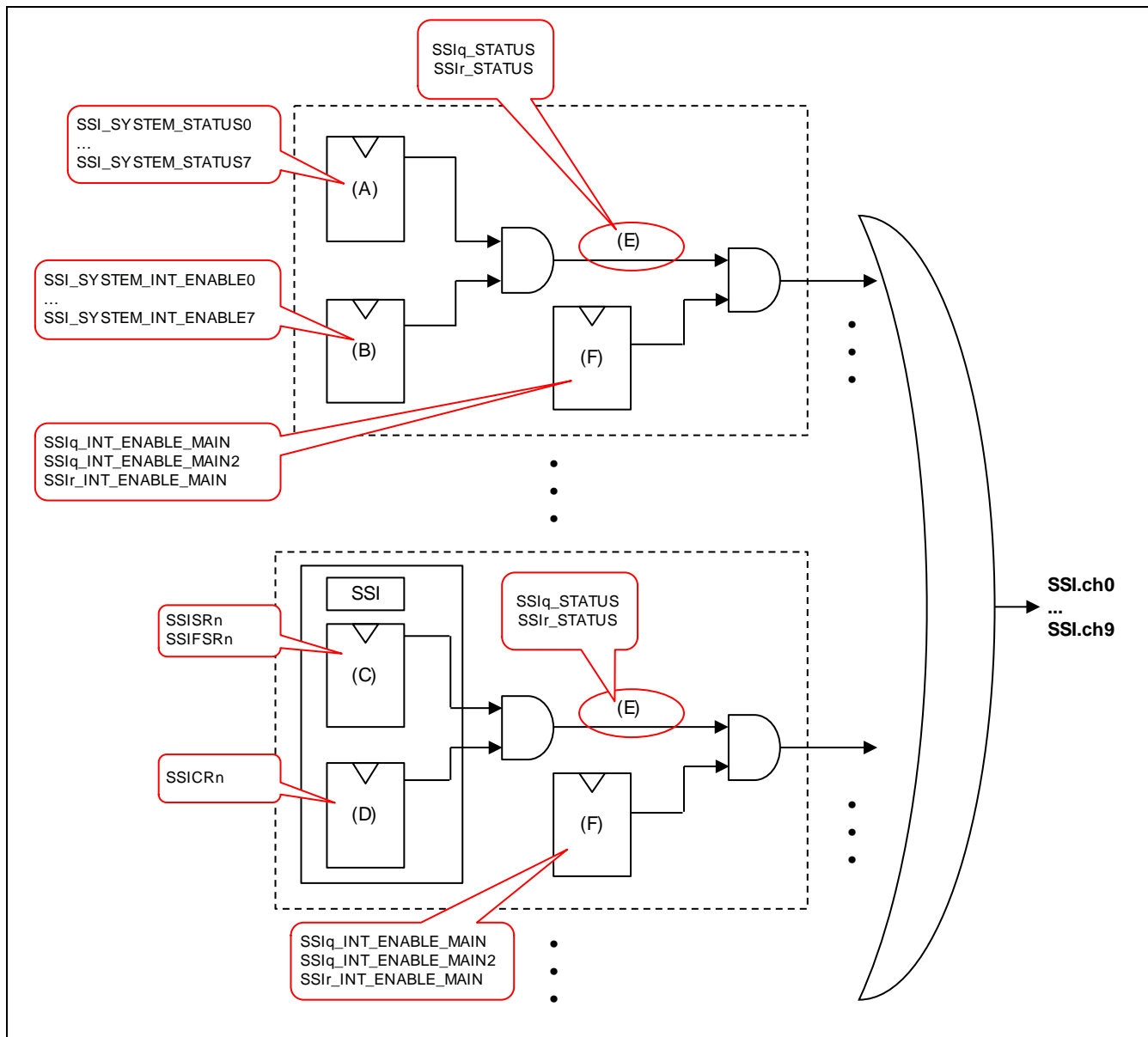


Figure 40.40 Mechanism of generating interrupt signals

Table 40.8 Combinations of relevant registers for each SSI (1)

| SSI | BUSIF | (A) | (B) | (E) | (F) |
|------|--------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| | | Register Name (Interrupt Factor) | Register Name (Interrupt Enable) | Register Name (Interrupt Factor) | Register Name (Interrupt Enable) |
| SSI0 | BUSIF0-0 ... BUSIF0-3 | SSI_SYSTEM_ STATUS0 | SSI_SYSTEM_INT_ ENABLE0 | SSI0-0_STATUS | SSI0-0_INT_ENABLE_ MAIN |
| | | SSI_SYSTEM_ STATUS2 | SSI_SYSTEM_INT_ ENABLE2 | SSI0-0_STATUS | SSI0-0_INT_ENABLE_ MAIN |
| | BUSIF0-4 ... BUSIF0-7 | SSI_SYSTEM_ STATUS4 | SSI_SYSTEM_INT_ ENABLE4 | SSI0-0_STATUS2 | SSI0-0_INT_ENABLE_ MAIN2 |
| | | SSI_SYSTEM_ STATUS6 | SSI_SYSTEM_INT_ ENABLE6 | SSI0-0_STATUS2 | SSI0-0_INT_ENABLE_ MAIN2 |
| SSI1 | BUSIF1-0 ... BUSIF1-3 | SSI_SYSTEM_ STATUS0 | SSI_SYSTEM_INT_ ENABLE0 | SSI1-0_STATUS | SSI1-0_INT_ENABLE_ MAIN |
| | | SSI_SYSTEM_ STATUS2 | SSI_SYSTEM_INT_ ENABLE2 | SSI1-0_STATUS | SSI1-0_INT_ENABLE_ MAIN |
| | BUSIF1-4 ... BUSIF1-7 | SSI_SYSTEM_ STATUS4 | SSI_SYSTEM_INT_ ENABLE4 | SSI1-0_STATUS2 | SSI1-0_INT_ENABLE_ MAIN2 |
| | | SSI_SYSTEM_ STATUS6 | SSI_SYSTEM_INT_ ENABLE6 | SSI1-0_STATUS2 | SSI1-0_INT_ENABLE_ MAIN2 |
| SSI2 | BUSIF2-0 ... BUSIF2-3 | SSI_SYSTEM_ STATUS0 | SSI_SYSTEM_INT_ ENABLE0 | SSI2-0_STATUS | SSI2-0_INT_ENABLE_ MAIN |
| | | SSI_SYSTEM_ STATUS2 | SSI_SYSTEM_INT_ ENABLE2 | SSI2-0_STATUS | SSI2-0_INT_ENABLE_ MAIN |
| | BUSIF2-4 ... BUSIF2-7 | SSI_SYSTEM_ STATUS4 | SSI_SYSTEM_INT_ ENABLE4 | SSI2-0_STATUS2 | SSI2-0_INT_ENABLE_ MAIN2 |
| | | SSI_SYSTEM_ STATUS6 | SSI_SYSTEM_INT_ ENABLE6 | SSI2-0_STATUS2 | SSI2-0_INT_ENABLE_ MAIN2 |
| SSI3 | BUSIF3-0 ... BUSIF3-3 | SSI_SYSTEM_ STATUS0 | SSI_SYSTEM_INT_ ENABLE0 | SSI3-0_STATUS | SSI3-0_INT_ENABLE_ MAIN |
| | | SSI_SYSTEM_ STATUS2 | SSI_SYSTEM_INT_ ENABLE2 | SSI3-0_STATUS | SSI3-0_INT_ENABLE_ MAIN |
| | BUSIF3-4 ... BUSIF3-7 | SSI_SYSTEM_ STATUS4 | SSI_SYSTEM_INT_ ENABLE4 | SSI3-0_STATUS2 | SSI3-0_INT_ENABLE_ MAIN2 |
| | | SSI_SYSTEM_ STATUS6 | SSI_SYSTEM_INT_ ENABLE6 | SSI3-0_STATUS2 | SSI3-0_INT_ENABLE_ MAIN2 |
| SSI4 | BUSIF4-0 ... BUSIF4-3 | SSI_SYSTEM_ STATUS0 | SSI_SYSTEM_INT_ ENABLE0 | SSI4-0_STATUS | SSI4-0_INT_ENABLE_ MAIN |
| | | SSI_SYSTEM_ STATUS2 | SSI_SYSTEM_INT_ ENABLE2 | SSI4-0_STATUS | SSI4-0_INT_ENABLE_ MAIN |
| | BUSIF4-4 ... BUSIF4-7 | SSI_SYSTEM_ STATUS4 | SSI_SYSTEM_INT_ ENABLE4 | SSI4-0_STATUS2 | SSI4-0_INT_ENABLE_ MAIN2 |
| | | SSI_SYSTEM_ STATUS6 | SSI_SYSTEM_INT_ ENABLE6 | SSI4-0_STATUS2 | SSI4-0_INT_ENABLE_ MAIN2 |
| SSI9 | BUSIF9-0 ... BUSIF9-3 | SSI_SYSTEM_ STATUS1 | SSI_SYSTEM_INT_ ENABLE1 | SSI9-0_STATUS | SSI9-0_INT_ENABLE_ MAIN |
| | | SSI_SYSTEM_ STATUS3 | SSI_SYSTEM_INT_ ENABLE3 | SSI9-0_STATUS | SSI9-0_INT_ENABLE_ MAIN |
| | BUSIF9-4 ... BUSIF9-7 | SSI_SYSTEM_ STATUS5 | SSI_SYSTEM_INT_ ENABLE5 | SSI9-0_STATUS2 | SSI9-0_INT_ENABLE_ MAIN2 |
| | | SSI_SYSTEM_ STATUS7 | SSI_SYSTEM_INT_ ENABLE7 | SSI9-0_STATUS2 | SSI9-0_INT_ENABLE_ MAIN2 |

Table 40.9 Combinations of relevant registers for each SSI (2)

| SSI | (C) | (D) | (E) | (F) |
|------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| | Register Name (Interrupt Factor) | Register Name (Interrupt Enable) | Register Name (Interrupt Factor) | Register Name (Interrupt Enable) |
| SSI0 | SSISR0 | SSICR0 | SSI0-0_STATUS | SSI0-0_INT_ENABLE_MAIN |
| | SSIFSR0 | SSICR0 | SSI0-0_STATUS | SSI0-0_INT_ENABLE_MAIN |
| SSI1 | SSISR1 | SSICR1 | SSI1-0_STATUS | SSI1-0_INT_ENABLE_MAIN |
| | SSIFSR1 | SSICR1 | SSI1-0_STATUS | SSI1-0_INT_ENABLE_MAIN |
| SSI2 | SSISR2 | SSICR2 | SSI2-0_STATUS | SSI2-0_INT_ENABLE_MAIN |
| | SSIFSR2 | SSICR2 | SSI2-0_STATUS | SSI2-0_INT_ENABLE_MAIN |
| SSI3 | SSISR3 | SSICR3 | SSI3-0_STATUS | SSI3-0_INT_ENABLE_MAIN |
| | SSIFSR3 | SSICR3 | SSI3-0_STATUS | SSI3-0_INT_ENABLE_MAIN |
| SSI4 | SSISR4 | SSICR4 | SSI4-0_STATUS | SSI4-0_INT_ENABLE_MAIN |
| | SSIFSR4 | SSICR4 | SSI4-0_STATUS | SSI4-0_INT_ENABLE_MAIN |
| SSI5 | SSISR5 | SSICR5 | SSI5_STATUS | SSI5_INT_ENABLE_MAIN |
| | SSIFSR5 | SSICR5 | SSI5_STATUS | SSI5_INT_ENABLE_MAIN |
| SSI6 | SSISR6 | SSICR6 | SSI6_STATUS | SSI6_INT_ENABLE_MAIN |
| | SSIFSR6 | SSICR6 | SSI6_STATUS | SSI6_INT_ENABLE_MAIN |
| SSI7 | SSISR7 | SSICR7 | SSI7_STATUS | SSI7_INT_ENABLE_MAIN |
| | SSIFSR7 | SSICR7 | SSI7_STATUS | SSI7_INT_ENABLE_MAIN |
| SSI8 | SSISR8 | SSICR8 | SSI8_STATUS | SSI8_INT_ENABLE_MAIN |
| | SSIFSR8 | SSICR8 | SSI8_STATUS | SSI8_INT_ENABLE_MAIN |
| SSI9 | SSISR9 | SSICR9 | SSI9-0_STATUS | SSI9-0_INT_ENABLE_MAIN |
| | SSIFSR9 | SSICR9 | SSI9-0_STATUS | SSI9-0_INT_ENABLE_MAIN |

[RZ/G2E]

SSI can output interrupt signals when one of several events happen. Figure 40.41 illustrates how SSI generates interrupt signals and Table 40.10 and Table 40.11 indicate combinations of relevant registers for each SSI to generate interrupt signals (refer to explanations of relevant registers (section 40.2) and SSI (section 41))

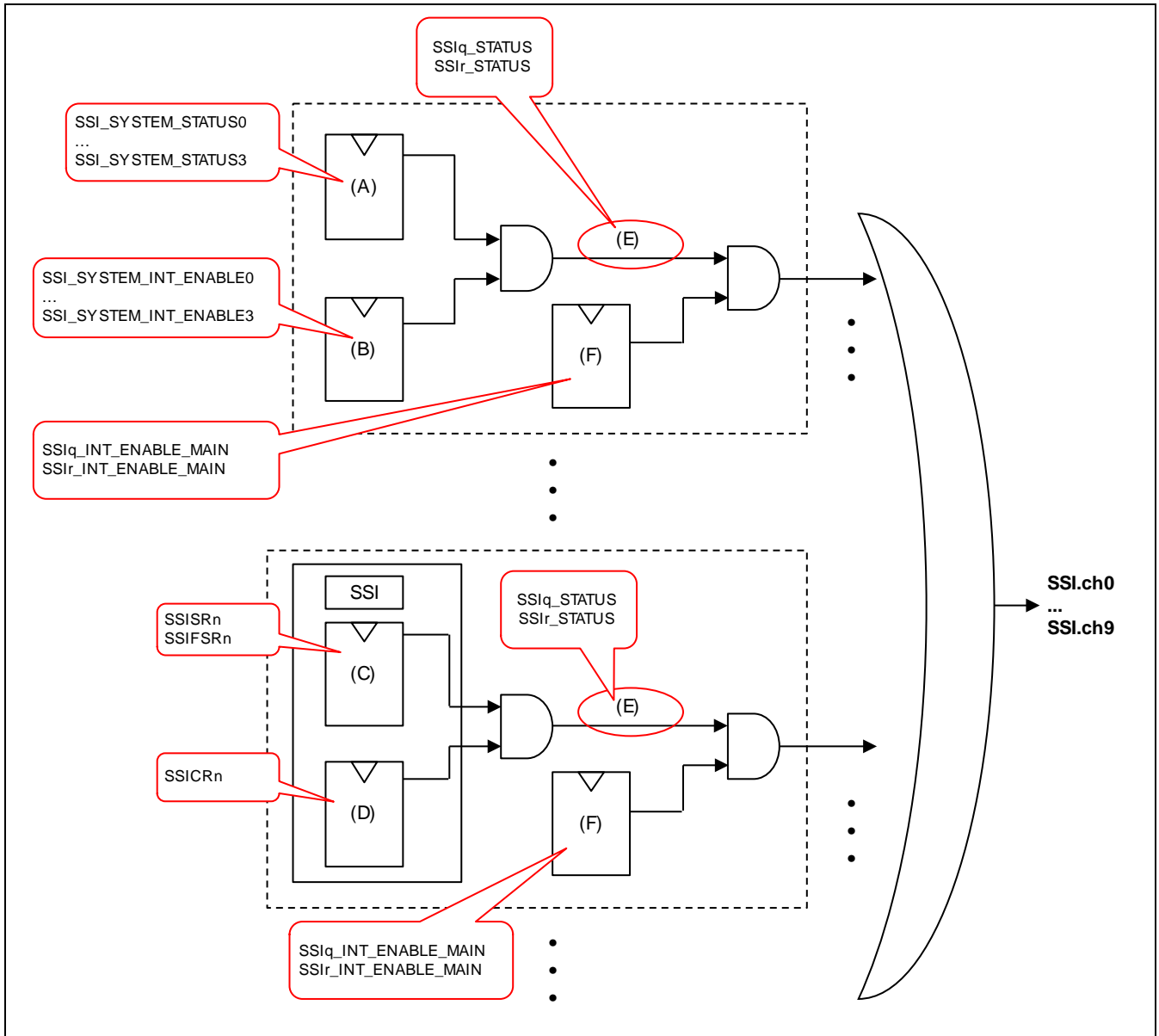


Figure 40.41 Mechanism of generating interrupt signals

Table 40.10 Combinations of relevant registers for each SSI (1)

| SSI | BUSIF | (A) | (B) | (E) | (F) |
|------|--------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| | | Register Name (Interrupt Factor) | Register Name (Interrupt Enable) | Register Name (Interrupt Factor) | Register Name (Interrupt Enable) |
| SSI0 | BUSIF0-0 ... BUSIF0-3 | SSI_SYSTEM_ STATUS0 | SSI_SYSTEM_INT_ ENABLE0 | SSI0-0_STATUS | SSI0-0_INT_ENABLE_ MAIN |
| | | SSI_SYSTEM_ STATUS2 | SSI_SYSTEM_INT_ ENABLE2 | SSI0-0_STATUS | SSI0-0_INT_ENABLE_ MAIN |
| SSI1 | BUSIF1-0 ... BUSIF1-3 | SSI_SYSTEM_ STATUS0 | SSI_SYSTEM_INT_ ENABLE0 | SSI1-0_STATUS | SSI1-0_INT_ENABLE_ MAIN |
| | | SSI_SYSTEM_ STATUS2 | SSI_SYSTEM_INT_ ENABLE2 | SSI1-0_STATUS | SSI1-0_INT_ENABLE_ MAIN |
| SSI2 | BUSIF2-0 ... BUSIF2-3 | SSI_SYSTEM_ STATUS0 | SSI_SYSTEM_INT_ ENABLE0 | SSI2-0_STATUS | SSI2-0_INT_ENABLE_ MAIN |
| | | SSI_SYSTEM_ STATUS2 | SSI_SYSTEM_INT_ ENABLE2 | SSI2-0_STATUS | SSI2-0_INT_ENABLE_ MAIN |
| SSI3 | BUSIF3-0 ... BUSIF3-3 | SSI_SYSTEM_ STATUS0 | SSI_SYSTEM_INT_ ENABLE0 | SSI3-0_STATUS | SSI3-0_INT_ENABLE_ MAIN |
| | | SSI_SYSTEM_ STATUS2 | SSI_SYSTEM_INT_ ENABLE2 | SSI3-0_STATUS | SSI3-0_INT_ENABLE_ MAIN |
| SSI4 | BUSIF4-0 ... BUSIF4-3 | SSI_SYSTEM_ STATUS0 | SSI_SYSTEM_INT_ ENABLE0 | SSI4-0_STATUS | SSI4-0_INT_ENABLE_ MAIN |
| | | SSI_SYSTEM_ STATUS2 | SSI_SYSTEM_INT_ ENABLE2 | SSI4-0_STATUS | SSI4-0_INT_ENABLE_ MAIN |
| SSI9 | BUSIF9-0 ... BUSIF9-3 | SSI_SYSTEM_ STATUS1 | SSI_SYSTEM_INT_ ENABLE1 | SSI9-0_STATUS | SSI9-0_INT_ENABLE_ MAIN |
| | | SSI_SYSTEM_ STATUS3 | SSI_SYSTEM_INT_ ENABLE3 | SSI9-0_STATUS | SSI9-0_INT_ENABLE_ MAIN |

Table 40.11 Combinations of relevant registers for each SSI (2)

| SSI | (C) | (D) | (E) | (F) |
|------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| | Register Name (Interrupt Factor) | Register Name (Interrupt Enable) | Register Name (Interrupt Factor) | Register Name (Interrupt Enable) |
| SSI0 | SSISR0 | SSICR0 | SSI0-0_STATUS | SSI0-0_INT_ENABLE_MAIN |
| | SSIFSR0 | SSICR0 | SSI0-0_STATUS | SSI0-0_INT_ENABLE_MAIN |
| SSI1 | SSISR1 | SSICR1 | SSI1-0_STATUS | SSI1-0_INT_ENABLE_MAIN |
| | SSIFSR1 | SSICR1 | SSI1-0_STATUS | SSI1-0_INT_ENABLE_MAIN |
| SSI2 | SSISR2 | SSICR2 | SSI2-0_STATUS | SSI2-0_INT_ENABLE_MAIN |
| | SSIFSR2 | SSICR2 | SSI2-0_STATUS | SSI2-0_INT_ENABLE_MAIN |
| SSI3 | SSISR3 | SSICR3 | SSI3-0_STATUS | SSI3-0_INT_ENABLE_MAIN |
| | SSIFSR3 | SSICR3 | SSI3-0_STATUS | SSI3-0_INT_ENABLE_MAIN |
| SSI4 | SSISR4 | SSICR4 | SSI4-0_STATUS | SSI4-0_INT_ENABLE_MAIN |
| | SSIFSR4 | SSICR4 | SSI4-0_STATUS | SSI4-0_INT_ENABLE_MAIN |
| SSI5 | SSISR5 | SSICR5 | SSI5_STATUS | SSI5_INT_ENABLE_MAIN |
| | SSIFSR5 | SSICR5 | SSI5_STATUS | SSI5_INT_ENABLE_MAIN |
| SSI6 | SSISR6 | SSICR6 | SSI6_STATUS | SSI6_INT_ENABLE_MAIN |
| | SSIFSR6 | SSICR6 | SSI6_STATUS | SSI6_INT_ENABLE_MAIN |
| SSI7 | SSISR7 | SSICR7 | SSI7_STATUS | SSI7_INT_ENABLE_MAIN |
| | SSIFSR7 | SSICR7 | SSI7_STATUS | SSI7_INT_ENABLE_MAIN |
| SSI8 | SSISR8 | SSICR8 | SSI8_STATUS | SSI8_INT_ENABLE_MAIN |
| | SSIFSR8 | SSICR8 | SSI8_STATUS | SSI8_INT_ENABLE_MAIN |
| SSI9 | SSISR9 | SSICR9 | SSI9-0_STATUS | SSI9-0_INT_ENABLE_MAIN |
| | SSIFSR9 | SSICR9 | SSI9-0_STATUS | SSI9-0_INT_ENABLE_MAIN |

40.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

40.4.1 Note on Transfer

If an underflow or overflow occurs, stop the transfer and restart it.

Do not stop serial clock (SCK) and word select (WS) signals during data transfer. Confirm that SSISR.IDST = 1 before stopping SCK and WS signals.

In case of transmission at slave mode, it is necessary that WS signal is stable. Data transmission at slave mode should be started after more than two frames period since WS signal is entered and is stable.

In case of multiple SSI modules and case that one SSI works at master mode and the others do at slave mode, data transfer should be started after more than two frames period since registers of SSI at master mode are set.

41. Serial Sound Interface (SSI)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

41.1 Overview

The serial sound interface (hereinafter referred to as the "SSI") is a transceiver module designed to send or receive audio data interfacing with a variety of devices offering I2S format. It also supports multi-channel mode in addition to other common formats.

41.1.1 Features

The SSI has the following features:

- Number of channels: Maximum of four (when a multichannel format is specified).
- The SSI module can serve as both a transmitter and a receiver.
- Asynchronous transfer takes place between the data buffer and the shift register.
- Only the MSB first data alignment is supported.
- A value as the dividing ratio for the clock used by the serial bus interface is selectable.
- Controlling of data transmission or reception with DMAC or interrupt requests is possible.
- TDM format is supported.
- The frequency range of SCK signal is from 297.3 kHz to 12.5MHz at master mode, and from 297.3 kHz to 15.1 MHz at slave mode.
- The WS continue function by which operation can be performed without stopping WS signal is supported.
- Monaural mode (8 bits or 16 bits) is supported.
- In monaural mode, WS signal pulse width can be changed (short or long frame).
- If the sampling clock frequency is switched during transfer, it will be notified of the CPU with interrupts (a function to detect switching frequency).
- Operating mode: non-compressed mode (Not support compressed mode).
- Supports versatile serial audio formats (I2S/left justified/right justified).
- Supports master/slave functions.
- Programmable word clock, bit clock generation functions.

41.1.2 Block Diagram

Figure 41.1 shows a block diagram of a single SSI module.

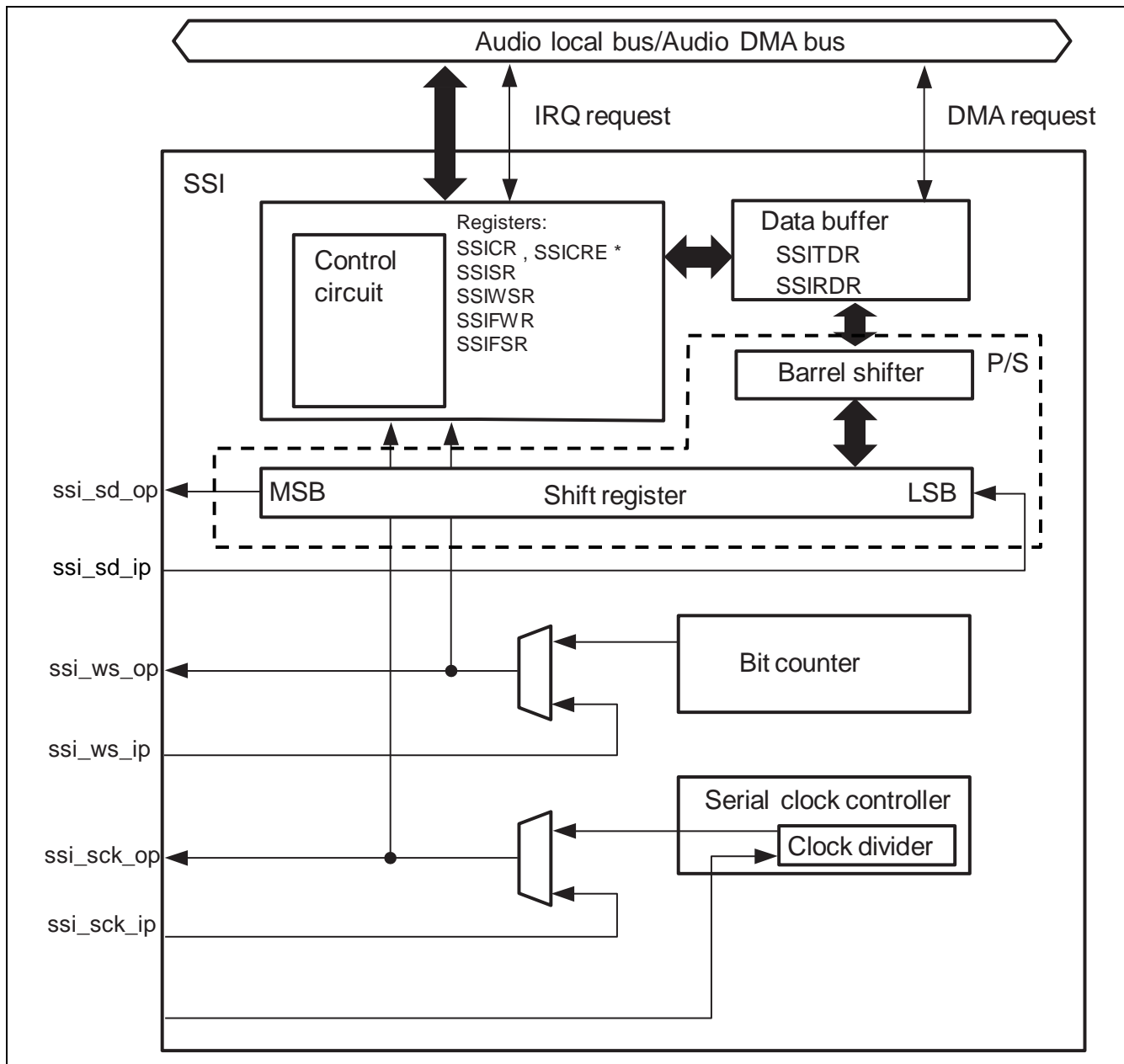


Figure 41.1 Block Diagram of SSI

Note: * SSICRE is available RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N

41.1.3 External Pins

No external pins

41.1.4 Register Configuration

The SSI has the following registers. Note that the module numbers are basically omitted from the register names in the text.

Table 41.1 Register Configurations

| Module | Register Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--------|---------------------------|--------------|-------|-------------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 0 | Control Register 0 | SSICR0 | R/W | H'EC54_1000 | 32 | √ | √ | √ | √ |
| | Status Register 0 | SSISR0 | R/W*1 | H'EC54_1004 | 32 | √ | √ | √ | √ |
| | Transmit Data Register 0 | SSITDR0 | R/W | H'EC54_1008/ H'EC24_1008*3 | 32 | √ | √ | √ | √ |
| | Receive Data Register 0 | SSIRDR0 | R | H'EC54_100C/ H'EC24_100C*3 | 32 | √ | √ | √ | √ |
| | WS Mode Register 0 | SSIWSR0 | R/W | H'EC54_1020 | 32 | √ | √ | √ | √ |
| | FS Mode Register 0 | SSIFMR0 | R/W | H'EC54_1024 | 32 | √ | √ | √ | √ |
| | FS Status Register 0 | SSIFSR0 | R/W*2 | H'EC54_1028 | 32 | √ | √ | √ | √ |
| | Control register extend 0 | SSICRE0 | R/W | H'EC54_1030 | 32 | √ | √ | √ | — |
| 1 | Control Register 1 | SSICR1 | R/W | H'EC54_1040 | 32 | √ | √ | √ | √ |
| | Status Register 1 | SSISR1 | R/W*1 | H'EC54_1044 | 32 | √ | √ | √ | √ |
| | Transmit Data Register 1 | SSITDR1 | R/W | H'EC54_1048/ H'EC24_1048*3 | 32 | √ | √ | √ | √ |
| | Receive Data Register 1 | SSIRDR1 | R | H'EC54_104C/ H'EC24_104C*3 | 32 | √ | √ | √ | √ |
| | WS Mode Register 1 | SSIWSR1 | R/W | H'EC54_1060 | 32 | √ | √ | √ | √ |
| | FS Mode Register 1 | SSIFMR1 | R/W | H'EC54_1064 | 32 | √ | √ | √ | √ |
| | FS Status Register 1 | SSIFSR1 | R/W*2 | H'EC54_1068 | 32 | √ | √ | √ | √ |
| | Control Register Extend 1 | SSICRE1 | R/W | H'EC54_1070 | 32 | √ | √ | √ | — |
| 2 | Control Register 2 | SSICR2 | R/W | H'EC54_1080 | 32 | √ | √ | √ | √ |
| | Status Register 2 | SSISR2 | R/W*1 | H'EC54_1084 | 32 | √ | √ | √ | √ |
| | Transmit Data Register 2 | SSITDR2 | R/W | H'EC54_1088/ H'EC24_1088*3 | 32 | √ | √ | √ | √ |
| | Receive Data Register 2 | SSIRDR2 | R | H'EC54_108C/ H'EC24_108C*3 | 32 | √ | √ | √ | √ |
| | WS Mode Register 2 | SSIWSR2 | R/W | H'EC54_10A0 | 32 | √ | √ | √ | √ |
| | FS Mode Register 2 | SSIFMR2 | R/W | H'EC54_10A4 | 32 | √ | √ | √ | √ |
| | FS Status Register 2 | SSIFSR2 | R/W*2 | H'EC54_10A8 | 32 | √ | √ | √ | √ |
| | Control Register Extend 2 | SSICRE2 | R/W | H'EC54_10B0 | 32 | √ | √ | √ | — |

| Module | Register Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--------|---------------------------|--------------|-------|-------------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 3 | Control Register 3 | SSICR3 | R/W | H'EC54_10C0 | 32 | √ | √ | √ | √ |
| | Status Register 3 | SSISR3 | R/W*1 | H'EC54_10C4 | 32 | √ | √ | √ | √ |
| | Transmit Data Register 3 | SSITDR3 | R/W | H'EC54_10C8/ H'EC24_10C8*3 | 32 | √ | √ | √ | √ |
| | Receive Data Register 3 | SSIRDR3 | R | H'EC54_10CC/ H'EC24_10CC*3 | 32 | √ | √ | √ | √ |
| | WS Mode Register 3 | SSIWSR3 | R/W | H'EC54_10E0 | 32 | √ | √ | √ | √ |
| | FS Mode Register 3 | SSIFMR3 | R/W | H'EC54_10E4 | 32 | √ | √ | √ | √ |
| | FS Status Register 3 | SSIFSR3 | R/W*2 | H'EC54_10E8 | 32 | √ | √ | √ | √ |
| | Control Register Extend 3 | SSICRE3 | R/W | H'EC54_10F0 | 32 | √ | √ | √ | — |
| 4 | Control Register 4 | SSICR4 | R/W | H'EC54_1100 | 32 | √ | √ | √ | √ |
| | Status Register 4 | SSISR4 | R/W*1 | H'EC54_1104 | 32 | √ | √ | √ | √ |
| | Transmit Data Register 4 | SSITDR4 | R/W | H'EC54_1108/ H'EC24_1108*3 | 32 | √ | √ | √ | √ |
| | Receive Data Register 4 | SSIRDR4 | R | H'EC54_110C/ H'EC24_110C*3 | 32 | √ | √ | √ | √ |
| | WS Mode Register 4 | SSIWSR4 | R/W | H'EC54_1120 | 32 | √ | √ | √ | √ |
| | FS Mode Register 4 | SSIFMR4 | R/W | H'EC54_1124 | 32 | √ | √ | √ | √ |
| | FS Status Register 4 | SSIFSR4 | R/W*2 | H'EC54_1128 | 32 | √ | √ | √ | √ |
| | Control Register Extend 4 | SSICRE4 | R/W | H'EC54_1130 | 32 | √ | √ | √ | — |
| 5 | Control Register 5 | SSICR5 | R/W | H'EC54_1140 | 32 | √ | √ | √ | √ |
| | Status Register 5 | SSISR5 | R/W*1 | H'EC54_1144 | 32 | √ | √ | √ | √ |
| | Transmit Data Register 5 | SSITDR5 | R/W | H'EC54_1148/ H'EC24_1148*3 | 32 | √ | √ | √ | √ |
| | Receive Data Register 5 | SSIRDR5 | R | H'EC54_114C/ H'EC24_114C*3 | 32 | √ | √ | √ | √ |
| | WS Mode Register 5 | SSIWSR5 | R/W | H'EC54_1160 | 32 | √ | √ | √ | √ |
| | FS Mode Register 5 | SSIFMR5 | R/W | H'EC54_1164 | 32 | √ | √ | √ | √ |
| | FS Status Register 5 | SSIFSR5 | R/W*2 | H'EC54_1168 | 32 | √ | √ | √ | √ |
| | Control Register Extend 5 | SSICRE5 | R/W | H'EC54_1170 | 32 | √ | √ | √ | — |
| 6 | Control Register 6 | SSICR6 | R/W | H'EC54_1180 | 32 | √ | √ | √ | √ |
| | Status Register 6 | SSISR6 | R/W*1 | H'EC54_1184 | 32 | √ | √ | √ | √ |
| | Transmit Data Register 6 | SSITDR6 | R/W | H'EC54_1188/ H'EC24_1188*3 | 32 | √ | √ | √ | √ |
| | Receive Data Register 6 | SSIRDR6 | R | H'EC54_118C/ H'EC24_118C*3 | 32 | √ | √ | √ | √ |
| | WS Mode Register 6 | SSIWSR6 | R/W | H'EC54_11A0 | 32 | √ | √ | √ | √ |
| | FS Mode Register 6 | SSIFMR6 | R/W | H'EC54_11A4 | 32 | √ | √ | √ | √ |
| | FS Status Register 6 | SSIFSR6 | R/W*2 | H'EC54_11A8 | 32 | √ | √ | √ | √ |
| | Control Register Extend 6 | SSICRE6 | R/W | H'EC54_11B0 | 32 | √ | √ | √ | — |

| | | | | | | Second Generation RZ/G Series Products | | | |
|--------|---------------------------|--------------|-------|-------------------------------|-------------|---|----------------------------|--------|--------|
| Module | Register Name | Abbreviation | R/W | Address | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 7 | Control Register 7 | SSICR7 | R/W | H'EC54_11C0 | 32 | √ | √ | √ | √ |
| | Status Register 7 | SSISR7 | R/W*1 | H'EC54_11C4 | 32 | √ | √ | √ | √ |
| | Transmit Data Register 7 | SSITDR7 | R/W | H'EC54_11C8/ H'EC24_11C8*3 | 32 | √ | √ | √ | √ |
| | Receive Data Register 7 | SSIRDR7 | R | H'EC54_11CC/ H'EC24_11CC*3 | 32 | √ | √ | √ | √ |
| | WS Mode Register 7 | SSIWSR7 | R/W | H'EC54_11E0 | 32 | √ | √ | √ | √ |
| | FS Mode Register 7 | SSIFMR7 | R/W | H'EC54_11E4 | 32 | √ | √ | √ | √ |
| | FS Status Register 7 | SSIFSR7 | R/W*2 | H'EC54_11E8 | 32 | √ | √ | √ | √ |
| 8 | Control Register 8 | SSICR8 | R/W | H'EC54_1200 | 32 | √ | √ | √ | √ |
| | Status Register 8 | SSISR8 | R/W*1 | H'EC54_1204 | 32 | √ | √ | √ | √ |
| | Transmit Data Register 8 | SSITDR8 | R/W | H'EC54_1208/ H'EC24_1208*3 | 32 | √ | √ | √ | √ |
| | Receive Data Register 8 | SSIRDR8 | R | H'EC54_120C/ H'EC24_120C*3 | 32 | √ | √ | √ | √ |
| | WS Mode Register 8 | SSIWSR8 | R/W | H'EC54_1220 | 32 | √ | √ | √ | √ |
| | FS Mode Register 8 | SSIFMR8 | R/W | H'EC54_1224 | 32 | √ | √ | √ | √ |
| | FS Status Register 8 | SSIFSR8 | R/W*2 | H'EC54_1228 | 32 | √ | √ | √ | √ |
| 9 | Control Register 9 | SSICR9 | R/W | H'EC54_1240 | 32 | √ | √ | √ | √ |
| | Status Register 9 | SSISR9 | R/W*1 | H'EC54_1244 | 32 | √ | √ | √ | √ |
| | Transmit Data Register 9 | SSITDR9 | R/W | H'EC54_1248/ H'EC24_1248*3 | 32 | √ | √ | √ | √ |
| | Receive Data Register 9 | SSIRDR9 | R | H'EC54_124C/ H'EC24_124C*3 | 32 | √ | √ | √ | √ |
| | WS Mode Register 9 | SSIWSR9 | R/W | H'EC54_1260 | 32 | √ | √ | √ | √ |
| | FS Mode Register 9 | SSIFMR9 | R/W | H'EC54_1264 | 32 | √ | √ | √ | √ |
| | FS Status Register 9 | SSIFSR9 | R/W*2 | H'EC54_1268 | 32 | √ | √ | √ | √ |
| | Control Register Extend 9 | SSICRE9 | R/W | H'EC54_1270 | 32 | √ | √ | √ | — |

Notes: Do not write to any addresses other than listed above. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed above are undefined.

1. For this register, bits 26 and 27 are readable/writable bits, although the others are read-only bits. For details, refer to section 41.2.2 Status Register (SSISRn).
2. For this register, bits 14 and 15 are readable/writable bits, although the others are read-only bits. For details, refer to section 41.2.7 FS Status Register (SSIFSRn).
3. H'EC54_1xxx and H'EC24_1xxx are used with PIO access and audio-DMAC, respectively.

41.1.5 Connected Module**Table 41.2 Connected Module**

| Module name | Connected module name | Function of connected module |
|--------------------|------------------------------|---|
| SSI | APMU | Access the Register |
| | PFC | Select External pins |
| | CPG | Output Clocks |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |
| | INTC-AP | Control to interrupt |
| | SSIU | Serial Sound Interface Unit |
| | ADG | Output Clocks for Audio module |
| | SCU | Sampling Rate Converter Unit |
| | Audio-DMAC | Control Direct Memory Access for Audio module |
| | Audio-DMACpp | Control Direct Memory Access for Audio modules connected to the audio local bus |

41.2 Register Description

[Legend for Register Descriptions]

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R: Read-only. The write value should always be 0.

41.2.1 Control Register (SSICRn)

Note: n = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SSICR is a readable/writable 32-bit register that controls the IRQ, selects the polarity status, and sets operating mode.

| | | | | | | | | | | | | | | | | |
|----------------|-------|------|------|------|------|------|------|------|-----------|-----------|-----|-----|----------|-----|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | FORCE | — | FIEN | DMEN | UIEN | OIEN | IIEN | DIEN | CHNL[1:0] | DWL[2:0] | | | SWL[2:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SCKD | SWSD | SCKP | SWSP | SPDP | SDTA | PDTA | DEL | — | CKDV[2:0] | | | MUEN | — | TRMD | EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 31 | FORCE | B'0 | R/W | Fixed The bit should always be set to 1. |
| 30 | — | B'0 | R | Reserved The read value is 0. The write value should always be 0. |
| 29 | FIEN | B'0 | R/W | Frequency Switching Detection Interrupt Enable 0: Frequency switching detection interrupt is disabled. 1: Frequency switching detection interrupt is enabled. |
| 28 | DMEN | B'0 | R/W | DMA Enable Enables or disables the DMA request. 0: DMA request is disabled. 1: DMA request is enabled. |
| 27 | UIEN | B'0 | R/W | Underflow Interrupt Enable 0: Underflow interrupt is disabled. 1: Underflow Interrupt is enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|---|
| 26 | OIEN | B'0 | R/W | Overflow Interrupt Enable 0: Overflow interrupt is disabled. 1: Overflow interrupt is enabled. |
| 25 | IEN | B'0 | R/W | Idle Mode Interrupt Enable 0: Idle mode interrupt is disabled. 1: Idle mode interrupt is enabled. |
| 24 | DIEN | B'0 | R/W | Data Interrupt Enable 0: Data interrupt is disabled. 1: Data interrupt is enabled. |
| 23, 22 | CHNL[1:0] | B'00 | R/W | Channels These bits set the number of channels in each system word. When the stereo format or multi-channel format is used (WS_MODE = 0, MONO = 0, and WIDTH = B'0_0000 in the WS Mode Register): B'00: A system word has one channel. B'01: A system word has two channels. B'10: A system word has three channels. B'11: A system word has four channels. When the TDM format is used (WS_MODE = 1, MONO = 0, and WIDTH = B'0_0000 in the WS Mode Register): B'00: Setting prohibited * B'01: A TDM frame consists of four system words. B'10: A TDM frame consists of six system words. B'11: A TDM frame consists of eight system words. * In case of CHNL = B'00 and SSICREm.CHNL2 = 1, a TDM frame consists of sixteen system words. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] When the monaural format is used (WS_MODE = 1, MONO = 1, and WIDTH = B'0_0001 to B'1_1111 in the WS Mode Register): B'00: A monaural frame consists of one system word. B'01: Setting prohibited B'10: Setting prohibited B'11: Setting prohibited |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 21 to 19 | DWL[2:0] | B'000 | R/W | <p>Data Word Length</p> <p>These bits set the number of bits in a data word.</p> <p>When the stereo format or multi-channel format (WS_MODE = 0, MONO = 0, and WIDTH = B'0_0000 in the WS Mode Register) or TDM format (WS_MODE = 1, MONO = 0, and WIDTH = B'0_0000 in the WS Mode Register) is used:</p> <p>B'000: 8 bits B'001: 16 bits B'010: 18 bits B'011: 20 bits B'100: 22 bits B'101: 24 bits B'110: 32 bits B'111: Setting prohibited</p> <p>When the monaural format is used (WS_MODE = 1, MONO = 1, and WIDTH = B'0_0001 to B'1_1111 in the WS Mode Register):</p> <p>B'000: 8 bits B'001: 16 bits Other than above: Setting prohibited</p> |
| 18 to 16 | SWL[2:0] | B'000 | R/W | <p>System Word Length</p> <p>These bits set the number of bits in a system word.</p> <p>When the stereo format or multi-channel format (WS_MODE = 0, MONO = 0, and WIDTH = B'0_0000 in the WS Mode Register) or TDM format (WS_MODE = 1, MONO = 0, and WIDTH = B'0_0000 in the WS Mode Register) is used:</p> <p>B'000: 8 bits B'001: 16 bits B'010: 24 bits B'011: 32 bits B'100: 48 bits B'101: 64 bits B'110: 128 bits B'111: 256 bits</p> <p>When the monaural format is used (WS_MODE = 1, MONO = 1, and WIDTH = B'0_0001 to B'1_1111 in the WS Mode Register):</p> <p>B'000: 16 bits B'001: 32 bits B'010: 48 bits B'011: 64 bits B'100: 96 bits B'101: 128 bits B'110: 256 bits B'111: 512 bits</p> |
| 15 | SCKD | B'0 | R/W | <p>Serial Bit Clock Direction</p> <p>0: Serial bit clock is input, slave mode. 1: Serial bit clock is output, master mode.</p> <p>Note: (SCKD, SWSD) = (0, 0) or (1, 1) can be set to SSI0 to SSI7 and SSI9. To SSI8, only (SCKD, SWSD) = (0, 0) can be set. Other settings are prohibited.</p> |

| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | |
|---|----------------------|----------------------|-----|---|--|----------|----------|---|---------------------|----------------------|---|----------------------|---------------------|---|---------------------|----------------------|---|----------------------|---------------------|
| 14 | SWSD | B'0 | R/W | <p>Serial WS Direction</p> <p>0: Serial word select is input, slave mode. 1: Serial word select is output, master mode.</p> <p>Note: (SCKD, SWSD) = (0, 0) or (1, 1) can be set to SSI0 to SSI7 and SSI9. To SSI8, only (SCKD, SWSD) = (0, 0) can be set. Other settings are prohibited.</p> | | | | | | | | | | | | | | | |
| 13 | SCKP | B'0 | R/W | <p>Serial Bit Clock Polarity</p> <p>0: SSI_WS and SSI_SDATA change at the SSI_SCK falling edge (sampled at the SCK rising edge). 1: SSI_WS and SSI_SDATA change at the SSI_SCK rising edge (sampled at the SCK falling edge).</p> <table border="1"> <thead> <tr> <th></th> <th>SCKP = 0</th> <th>SCKP = 1</th> </tr> </thead> <tbody> <tr> <td>SSI_SDATA input sampling timing at the time of reception (TRMD = 0)</td> <td>SSI_SCK rising edge</td> <td>SSI_SCK falling edge</td> </tr> <tr> <td>SSI_SDATA output change timing at the time of transmission (TRMD = 1)</td> <td>SSI_SCK falling edge</td> <td>SSI_SCK rising edge</td> </tr> <tr> <td>SSI_WS input sampling timing at the time of slave mode (SWSD = 0)</td> <td>SSI_SCK rising edge</td> <td>SSI_SCK falling edge</td> </tr> <tr> <td>SSI_WS output change timing at the time of master mode (SWSD = 1)</td> <td>SSI_SCK falling edge</td> <td>SSI_SCK rising edge</td> </tr> </tbody> </table> | | SCKP = 0 | SCKP = 1 | SSI_SDATA input sampling timing at the time of reception (TRMD = 0) | SSI_SCK rising edge | SSI_SCK falling edge | SSI_SDATA output change timing at the time of transmission (TRMD = 1) | SSI_SCK falling edge | SSI_SCK rising edge | SSI_WS input sampling timing at the time of slave mode (SWSD = 0) | SSI_SCK rising edge | SSI_SCK falling edge | SSI_WS output change timing at the time of master mode (SWSD = 1) | SSI_SCK falling edge | SSI_SCK rising edge |
| | SCKP = 0 | SCKP = 1 | | | | | | | | | | | | | | | | | |
| SSI_SDATA input sampling timing at the time of reception (TRMD = 0) | SSI_SCK rising edge | SSI_SCK falling edge | | | | | | | | | | | | | | | | | |
| SSI_SDATA output change timing at the time of transmission (TRMD = 1) | SSI_SCK falling edge | SSI_SCK rising edge | | | | | | | | | | | | | | | | | |
| SSI_WS input sampling timing at the time of slave mode (SWSD = 0) | SSI_SCK rising edge | SSI_SCK falling edge | | | | | | | | | | | | | | | | | |
| SSI_WS output change timing at the time of master mode (SWSD = 1) | SSI_SCK falling edge | SSI_SCK rising edge | | | | | | | | | | | | | | | | | |
| 12 | SWSP | B'0 | R/W | <p>Serial WS Polarity</p> <p>The value of this bit must not be changed when the SSI module enable (EN) bit in this register is set to 1.</p> <ul style="list-style-type: none"> When the stereo format or multi-channel format is used (WS_MODE = 0, MONO = 0, and WIDTH = B'0_0000 in the WS Mode Register): 0: SSI_WS is low for 1st channel, high for 2nd channel. 1: SSI_WS is high for 1st channel, low for 2nd channel. When the TDM format is used (WS_MODE = 1, MONO = 0, and WIDTH = B'0_0000 in the WS Mode Register): 0: The SYNC pulse is high over the period of system word 1, and low otherwise. 1: The SYNC pulse is low for over the period of system word 1, and high otherwise. When the monaural format is used (WS_MODE = 1, MONO = 1, and WIDTH = B'0_0001 to B'1_1111 in the WS Mode Register): 0: WS pulse is high over the period specified with the WIDTH bits in the WS Mode Register, and low otherwise. 1: WS pulse is low over the period specified with the WIDTH bits in the WS Mode Register, and high otherwise. | | | | | | | | | | | | | | | |
| 11 | SPDP | B'0 | R/W | <p>Serial Padding Polarity</p> <p>0: Padding bits are low. 1: Padding bits are high.</p> <p>Padding bits are low when MUEN = 1. (The mute function takes priority.)</p> | | | | | | | | | | | | | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 10 | SDTA | B'0 | R/W | <p>Serial Data Alignment</p> <p>This bit should be set to 0 when the monaural format is used (WS_MODE = 1, MONO = 1, and WIDTH = B'0_0001 to B'1_1111 in the WS Mode Register).</p> <p>0: Transmitting and receiving in the order of serial data and padding bits.</p> <p>1: Transmitting and receiving in the order of padding bits and serial data.</p> |
| 9 | PDTA | B'0 | R/W | <p>Parallel Data Alignment</p> <p>When the length of data word is 32, 16 or 8 bits, this configuration field has no meaning. This bit should be set to 0 when the monaural format is used (WS_MODE = 1, MONO = 1, and WIDTH = B'0_0001 to B'1_1111 in the WS Mode Register).</p> <p>This bit applies to SSIRDR in receive mode and SSITDR in transmit mode.</p> <p>0: Parallel data (SSITDR, SSIRDR) is left-aligned.</p> <p>1: Parallel data (SSITDR, SSIRDR) is right-aligned.</p> <p>DWL = 000 (with a data word length of 8 bits), the PDTA setting is ignored.</p> <p>All data bits in SSIRDR or SSITDR are used on the audio serial bus. Four data words are transmitted or received at each 32-bit access. The first data word is derived from bits 7 to 0, the second from bits 15 to 8, the third from bits 23 to 16 and the last data word is derived from bits 31 to 24.</p> <p>DWL = B'001 (with a data word length of 16 bits), the PDTA setting is ignored.</p> <p>All data bits in SSIRDR or SSITDR are used on the audio serial bus. Two data words are transmitted or received at each 32-bit access. The first data word is derived from bits 15 to 0 and the second data word is derived from bits 31 to 16.</p> <p>DWL = B'010, B'011, B'100, B'101 (with a data word length of 18, 20, 22 or 24 bits), PDTA = 0 (left-aligned)</p> <p>The data bits used in SSIRDR or SSITDR are the following: Bits 31 down to (32 minus the number of bits in the data word length specified by DWL).</p> <p>That is, if DWL = B'011, the data word length is 20 bits; therefore, bits 31 to 12 in either SSIRDR or SSITDR are used. All other bits are ignored or reserved.</p> <p>DWL = B'010, B'011, B'100, B'101 (with a data word length of 18, 20, 22 or 24 bits), PDTA = 1 (right-aligned)</p> <p>The data bits used in SSIRDR or SSITDR are the following: Bits (the number of bits in the data word length specified by DWL minus 1) to 0 i.e. if DWL = B'011, then DWL = 20 and bits 19 to 0 are used in either SSIRDR or SSITDR. All other bits are ignored or reserved.</p> <p>DWL = B'110 (with a data word length of 32 bits), the PDTA setting is ignored.</p> <p>All data bits in SSIRDR or SSITDR are used on the audio serial bus.</p> |
| 8 | DEL | B'0 | R/W | <p>Serial Data Delay</p> <p>0: One clock cycle delay between SSI_WS and SSI_SDATA</p> <p>1: No delay between SSI_WS and SSI_SDATA</p> |
| 7 | — | B'0 | R | <p>Reserved</p> <p>The read value is 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|--|
| 6 to 4 | CKDV[2:0] | B'000 | R/W | <p>Serial Oversampling Clock Division Ratio</p> <p>Set the ratio between oversampling clock, CLK_FS, and the serial bit clock.</p> <p>These bits are ignored if SCKD = 0.</p> <p>The serial bit clock is used in the shift register and is provided on the SSI_SCK module pin.</p> <p>B'000: Serial bit clock frequency = oversampling clock frequency/1 B'001: Serial bit clock frequency = oversampling clock frequency/2 B'010: Serial bit clock frequency = oversampling clock frequency/4 B'011: Serial bit clock frequency = oversampling clock frequency/8 B'100: Serial bit clock frequency = oversampling clock frequency/16 B'101: Serial bit clock frequency = oversampling clock frequency/6 B'110: Serial bit clock frequency = oversampling clock frequency/12 B'111: Setting prohibited</p> <p>CKDV = 000 is invalid when WS_MODE = 1 or CONT = 1 in the WS Mode Register.</p> |
| 3 | MUEN | B'0 | R/W | <p>Serial Data Output Disable</p> <p>0: Module is not muted. 1: Module is muted.</p> <p>Note: This bit can be used to stop output (low output) or to enable output. However, the operation is not synchronized with the change of SSI_WS.</p> |
| 2 | — | B'0 | R | <p>Reserved</p> <p>The read value is 0. The write value should always be 0.</p> |
| 1 | TRMD | B'0 | R/W | <p>Transmit/Receive Mode Select</p> <p>0: Module is in receive mode. 1: Module is in transmit mode.</p> |
| 0 | EN | B'0 | R/W | <p>SSI Module Enable</p> <p>0: Module is disabled. 1: Module is enabled.</p> <p>When changing EN bit, do not change other bits.</p> |

41.2.2 Status Register (SSISRn)

Note: n = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SSISR consists of status flags indicating the operational status of the SSI module and bits indicating the current channel numbers and word numbers.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|------|-------|-------|------|------|----|----|----|----|-----------|------|------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | DMRQ | UIRQ | OIRQ | IIRQ | DIRQ | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | 0 | 0 | 0 | — | 0 | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R/WC0 | R/WC0 | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CHNO[1:0] | SWNO | IDST | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | 0 | 0 | 1 | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 29 | — | Undefined | R | Reserved The read value is not guaranteed. The write value should always be 0. |
| 28 | DMRQ | B'0 | R | DMA Request Status Flag This status flag allows the CPU to recognize the value of the DMA request pin on the SSI module. <ul style="list-style-type: none"> • TRMD = 0 (receive mode) If DMRQ = 1, the SSIRD has unread data. If SSIRD is read, DMRQ = 0 until there is new unread data. • TRMD = 1 (transmit mode) If DMRQ = 1, SSITDR requires data to be written to continue the transmission to the audio serial bus. Once data is written to SSITDR, DMRQ = 0 until it requires further transmit data. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 27 | UIRQ | B'0 | R/WC0 | <p>Underflow Error Interrupt Status Flag</p> <p>This status flag indicates that data was supplied at a lower rate than was required.</p> <p>In either case, this bit is set to 1 regardless of the value of the UIEN bit and can be cleared by writing 0 to this bit.</p> <p>If UIRQ = 1 and UIEN = 1, an interrupt occurs.</p> <ul style="list-style-type: none"> • TRMD = 0 (receive mode) If UIRQ = 1, SSIRDR was read before there was new unread data indicated by the DMRQ or DIRQ bit. This can lead to the same received data being stored twice by the host leading to potential corruption of multi-channel data. • TRMD = 1 (transmit mode) If UIRQ = 1, SSITDR did not have data written to it before it was required for transmission. This will lead to the same data being transmitted once more and a potential corruption of multi-channel data. This is more serious error than a receive mode underflow as the output SSI data results in error. <p>Note: When underflow error occurs, the current data in the data buffer of this module is transmitted until the next data is filled.</p> |
| 26 | OIRQ | B'0 | R/WC0 | <p>Overflow Error Interrupt Status Flag</p> <p>This status flag indicates that data was supplied at a higher rate than was required.</p> <p>In either case this bit is set to 1 regardless of the value of the OIEN bit and can be cleared by writing 0 to this bit.</p> <p>If OIRQ = 1 and OIEN = 1, an interrupt occurs.</p> <ul style="list-style-type: none"> • TRMD = 0 (receive mode) If OIRQ = 1, SSIRDR was not read before there was new unread data written to it. This will lead to the loss of a data and a potential corruption of multi-channel data. • TRMD = 1 (transmit mode) If OIRQ = 1, SSITDR had data written to it before it was transferred to the shift register. This will lead to the loss of a data and a potential corruption of multi-channel data. <p>Note: When overflow error occurs, the current data in the data buffer of this module is overwritten by the next incoming data from the SSI interface.</p> |
| 25 | IIRQ | Undefined | R | <p>Idle Mode Interrupt Status Flag</p> <p>This interrupt status flag indicates whether the SSI module is in idle state.</p> <p>This bit is set regardless of the value of the I IEN bit to allow polling. "Idle state" refers to the state where the serial bus has been stopped after activation of the SSI.</p> <p>The interrupt can be masked by clearing I IEN, but cannot be cleared by writing to this bit.</p> <p>If IIRQ = 1 and I IEN = 1, an interrupt occurs.</p> <p>0: The SSI module is not in idle state. 1: The SSI module is in idle state.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 24 | DIRQ | B'0 | R | <p>Data Interrupt Status Flag</p> <p>This status flag indicates that the module has data to be read or requires data to be written.</p> <p>In either case this bit is set to 1 regardless of the value of the DIEN bit to allow polling.</p> <p>The interrupt can be masked by clearing DIEN, but cannot be cleared by writing to this bit.</p> <p>If DIRQ = 1 and DIEN = 1, an interrupt occurs.</p> <ul style="list-style-type: none"> • TRMD = 0 (receive mode) <ul style="list-style-type: none"> 0: No unread data in SSIRDR 1: Unread data in SSIRDR • TRMD = 1 (transmit mode) <ul style="list-style-type: none"> 0: Transmit buffer is full. 1: Transmit buffer is empty and requires data to be written to SSITDR. |
| 23 to 4 | — | Undefined | R | <p>Reserved</p> <p>The read value is not guaranteed. The write value should always be 0.</p> |
| 3, 2 | CHNO[1:0] | B'00 | R | <p>Channel Number</p> <p>These bits indicate the current channel number. However, if the length of data words is 8 or 16 bits, the value of these bits is meaningless.</p> <p>B'00: 1st channel B'01: 2nd channel B'10: 2nd channel B'11: 4th channel</p> <ul style="list-style-type: none"> • TRMD = 0 (receive mode) <p>CHNO indicates which channel the data in SSIRDR currently represents. This value will change as the data in SSIRDR is updated from the shift register.</p> • TRMD = 1 (transmit mode) <p>CHNO indicates which channel is required to be written to SSITDR. This value will change as the data is copied to the shift register, regardless of whether the data is written to SSITDR.</p> <p>These bits cannot be used when WS_MODE = 1 or CONT = 1 in the WS Mode Register.</p> |
| 1 | SWNO | B'1 | R | <p>System Word Number</p> <p>This status bit indicates the current word number. However, if the length of data words is 8 or 16 bits, the value of this bit is meaningless.</p> <ul style="list-style-type: none"> • TRMD = 0 (receive mode) <p>SWNO indicates which system word the data in SSIRDR currently represents. This value will change as the data in SSIRDR is updated from the shift register, regardless of whether SSIRDR has been read.</p> • TRMD = 1 (transmit mode) <p>SWNO indicates which system word is required to be written to SSITDR. This value will change as the data is copied to the shift register, regardless of whether the data is written to SSITDR.</p> <p>This bit cannot be used when WS_MODE = 1 or CONT = 1 in the WS Mode Register.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | IDST | Undefined | R | <p>Idle Mode Status Flag</p> <p>This status flag indicates that the serial bus activity has stopped. This bit is cleared if EN = 1 and the serial bus are currently active. This bit is automatically set to 1 under the following conditions.</p> <ul style="list-style-type: none"> • SSI = Master transmitter (SWSD = 1 and TRMD = 1) This bit is set to 1 when the EN bit is cleared and data written in SSITDR has been output from the serial data input/output pin (SSI_SDATA). • SSI = Master receiver (SWSD = 1 and TRMD = 0) This bit is set to 1 when the EN bit is cleared and transfer of the current system word is completed. • SSI = Slave transmitter/receiver (SWSD = 0) This bit is set to 1 when the EN bit is cleared and transfer of the current system word is completed. <p>Note: If an external device stops the serial bus clock before transfer of the current system word is completed, this bit is not set.</p> |

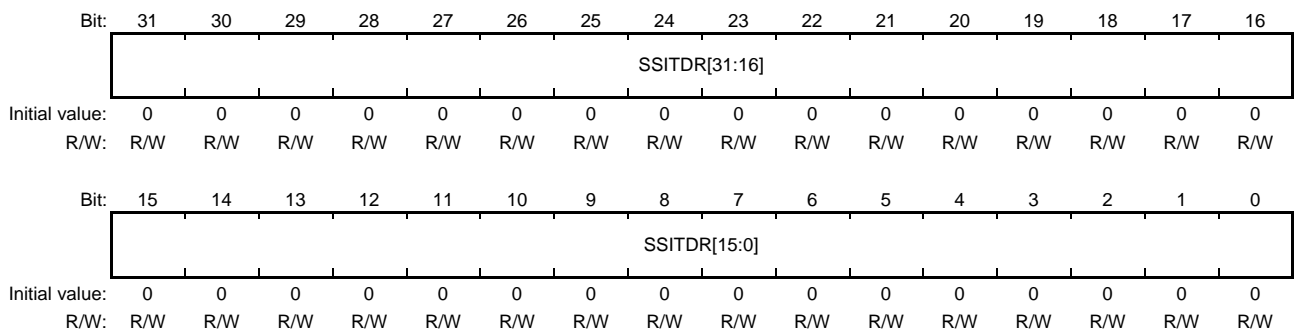
41.2.3 Transmit Data Register (SSITDRn)

Note: n = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SSITDR is a 32-bit register that holds data to be transmitted.

Data written to this register is transferred to the shift register upon transmission request. If the data word length is less than 32 bits, the alignment is determined by the setting of the PDTA control bit in SSICR. The data in the buffer can be accessed by reading this register.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|---|
| 31 to 0 | SSITDR[31:0] | All 0 | R/W | Transmit Data Data transmitted from SSI. |

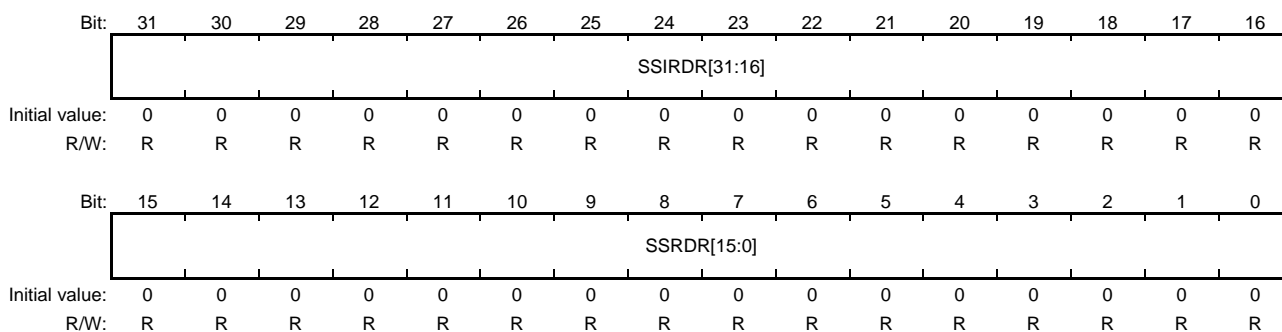
41.2.4 Receive Data Register (SSIRDRn)

Note: n = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SSIRDR is a 32-bit register that stores receive messages.

Data in this register is transferred from the shift register each time data word is received. If the data word length is less than 32 bits, the alignment is determined by the setting of the PDTA control bit in SSICR.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|---|
| 31 to 0 | SSIRDR[31:0] | All 0 | R | Receive Data Data received from external pins. |

41.2.5 WS Mode Register (SSIWSRn)

Note: n = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SSIWSR is a 32-bit readable/writable register that sets the TDM format, monaural format, and WS continue function.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|------|----|----|----|------------|-----|-----|------|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | WIDTH[4:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | CONT | — | — | — | — | — | — | MONO | WS_MODE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 21 | — | All 0 | R | Reserved The read value is 0. The write value should always be 0. |
| 20 to 16 | WIDTH[4:0] | All 0 | R/W | SYNC Pulse Width Change <ul style="list-style-type: none"> When the TDM format is used (WS_MODE = 1 and MONO = 0 in this register): B'0_0000: TDM format Other than B'0_0000: Setting prohibited When the monaural format is used (WS_MODE = 1 and MONO = 1 in this register): B'0_0000: Setting prohibited B'0_0001: Pulse width is equivalent to one cycle of SCK. B'0_0010: Pulse width is equivalent to two cycles of SCK. ... B'1_1111: Pulse width is equivalent to 31 cycles of SCK. Note that the set value of the system word length bits (SWL) in SSICR should be greater than that of the SYNC pulse width bits (WIDTH) when the monaural format is used. |
| 15 to 9 | — | All 0 | R | Reserved The read value is 0. The write value should always be 0. |
| 8 | CONT | B'0 | R/W | WS Continue Function 0: WS continue function is disabled. 1: WS continue function is enabled. Note: This bit can only be set in master mode (SSICR.SCKD = 1, SSICR.SWSD = 1). |
| 7 to 2 | — | All 0 | R | Reserved The read value is 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 1 | MONO | B'0 | R/W | TDM Format/Monaural Format Selects the TDM format or monaural format. When the normal stereo format, multi-channel format, this bit should be 0. 0: TDM format 1: Monaural format |
| 0 | WS_MODE | B'0 | R/W | WS Mode 0: Stereo format, multi-channel format 1: TDM format, monaural format |

41.2.6 FS Mode Register (SSIFMRn)

Note: n = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SSIFMR is used to set the frequency switching detection function.

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-----------|-----|-----|-----|-----|------|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | DTCT[5:0] | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | — | — | — | — | — | CTDV[1:0] | | — | — | — | FSEN | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R | R | R | R/W | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 22 | — | All 0 | R | Reserved The read value is 0. The write value should always be 0. |
| 21 to 16 | DTCT[5:0] | All 0 | R/W | Frequency Switching Detection Range Set B'00_0100: Set the value in the range as 5. Other than above: Setting prohibited. |
| 15 to 6 | — | All 0 | R | Reserved The read value is 0. The write value should always be 0. |
| 5, 4 | CTDV[1:0] | B'00 | R/W | Bus Clock Division Ratio Sets the bus clock division ratio used for the frequency switching detection function. B'00: Setting prohibited. B'01: Setting prohibited. B'10: Setting prohibited. B'11: Set the switching detection clock frequency = bus clock frequency/8. |
| 3 to 1 | — | All 0 | R | Reserved The read value is 0. The write value should always be 0. |
| 0 | FSEN | B'0 | R/W | Frequency Switching Detection Function Enable 0: The frequency switching detection function is disabled. 1: The frequency switching detection function is enabled. Note: This bit must be enabled after the desired settings have been made for SWSP in SSICR and WS_MODE in SSIWSR. |

41.2.7 FS Status Register (SSIFSRn)

Note: n = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SSIFSR is used to read the frequency switching detection status and information on the frequency counter.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|----|----|------------|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | FCST | DTST | — | — | FCNT[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/WC0 | R/WC0 | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-------|---|
| 31 to 16 | — | All 0 | R | Reserved The read value is 0. The write value should always be 0. |
| 15 | FCST | B'0 | R/WC0 | WS Stopped Status Flag In the frequency switching detection function, this status flag reflects the state that the WS signal is stopped. When FCST = 1, this bit indicates that the WS signal has stopped (the state is detected only when FSEN in SSIFMR = 1). Write 0 if this bit is cleared to 0. |
| 14 | DTST | B'0 | R/WC0 | Frequency Switching Detection Status Flag In the frequency switching detection function, this status flag reflects the state that frequency switching has been detected. When DTST = 1, this bit indicates that the switching has been detected (the state is detected only when FSEN in SSIFMR = 1). Write 0 if this bit is cleared to 0. |
| 13, 12 | — | B'00 | R | Reserved The read value is 0. The write value should always be 0. |
| 11 to 0 | FCNT[11:0] | All 0 | R | Frequency Count Monitor When the frequency switching detection function is enabled, the count value between WS signal edges is reflected according to the setting of the operating mode and the serial WS polarity (SWSP in SSICR). This value is updated when the frequency is switched and the WS signal has been detected. |

41.2.8 Control Register Extend (SSICREm) (m = 0 to 4, 9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

CHNL2 is an extension bit of SSICRn.CHNL[1:0]. This bit should be set 0 except SSIq_MODE2.ex_func in SSIU is 1 and TDM format is not selected (SSIWSRn.WS_MODE = 1 and SSIWSRn.MONO = 0).

This register should be set up before SSICRn is set up.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CHNL2 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved The read value is 0. The write value should always be 0. |
| 0 | CHNL2 | B'0 | R/W | Extension bit of SSICRn.CHNL[1:0] 0: Default value 1: A TDM frame consists of sixteen system words. In this case, SSICRn.CHNL should be set B'00. |

41.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

41.3.1 Bus Format

The SSI module can operate as a transmitter or a receiver and can be configured into many serial bus formats in either mode.

The bus format can be selected from one of the four major modes shown in Table 41.3.

Table 41.3 Bus Format for SSI Module

| | SSICR | | | | | | | | | | | | | | SSIWSR | | | SSICRE * | | | | | |
|--------------------|-------|------|------|--------------|------|------|-----|--------------------|------|-----|------|------|------|------|--------|----------|----------|--------------|---------|--------------|------|-------|--|
| | TRMD | SCKD | SWSD | EN | MUEN | DIEN | IEN | OIEN | UIEN | DEL | PDTA | SDTA | SPDP | SWSP | SCKP | SWL[2:0] | DWL[2:0] | CHNL[1:0] | WS_MODE | MONO | CONT | CHNL2 | |
| Slave receiver | 0 | 0 | 0 | Control bits | | | | Configuration bits | | | | | | | | | | WS mode bits | | Control bits | | | |
| Slave transmitter | 1 | 0 | 0 | | | | | | | | | | | | | | | | | | | | |
| Master receiver | 0 | 1 | 1 | | | | | | | | | | | | | | | | | | | | |
| Master transmitter | 1 | 1 | 1 | | | | | | | | | | | | | | | | | | | | |

Note: * [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] only available.

(1) Slave receiver

This mode allows the module to receive serial data from another device. The clock and word select signal used for the serial data stream is also supplied from an external device. If these signals do not conform to the format specified in the configuration fields of the SSI module, operation is not guaranteed.

(2) Slave transmitter

This mode allows the module to transmit serial data to another device. The clock and word select signal used for the serial data stream is also supplied from an external device. If these signals do not conform to the format specified in the configuration fields of the SSI module, operation is not guaranteed.

(3) Master receiver

This mode allows the module to receive serial data from another device. The clock and word select signals are internally derived from the CLK_FS input clock. The format of these signals is defined in the configuration fields of the SSI module. If the incoming data does not follow the configured format, operation is not guaranteed.

(4) Master transmitter

This mode allows the module to transmit serial data to another device. The clock and word select signals are internally derived from the CLK_FS input clock. The format of these signals is defined in the configuration fields of the SSI module.

(5) Setting for each format

Table 41.4 shows the setting for each format:

Table 41.4 Setting for Each Format

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| Format | SSICR | SSIWSR | | | SSICRE |
|----------------------|-----------------------|---------|------|----------------------|---------|
| | CHNL[1:0] | WS_MODE | MONO | WIDTH[4:0] | CHNL2 |
| Stereo format | B'00 | 0 | 0 | B'0_0000 | B'0 |
| Multi-channel format | B'01/B'10/B'11 | 0 | 0 | B'0_0000 | B'0 |
| Monaural format | B'00 | 1 | 1 | B'0_0001 to B'1_1111 | B'0 |
| TDM format | B'00/B'01/B'10/B'11 * | 1 | 0 | B'0_0000 | B'0/1 * |

Note: * SSICR.CHNL[1:0] should be set B'00 to when SSICRE.CHNL2 = 1.

[RZ/G2E]

| Format | SSICR | SSIWSR | | |
|----------------------|-----------------------|---------|------|----------------------|
| | CHNL[1:0] | WS_MODE | MONO | WIDTH[4:0] |
| Stereo format | B'00 | 0 | 0 | B'0_0000 |
| Multi-channel format | B'01/B'10/B'11 | 0 | 0 | B'0_0000 |
| Monaural format | B'00 | 1 | 1 | B'0_0001 to B'1_1111 |
| TDM format | B'00/B'01/B'10/B'11 * | 1 | 0 | B'0_0000 |

(6) Configuration bits (related to word length)

There are many configurations the SSI module supports, but only some of the combinations are shown below for the I2S, left-aligned, and right-aligned formats.

Figure 41.2 and Figure 41.3 demonstrate the supported I2S format both with and without padding. Padding occurs when the data word length is smaller than the system word length.

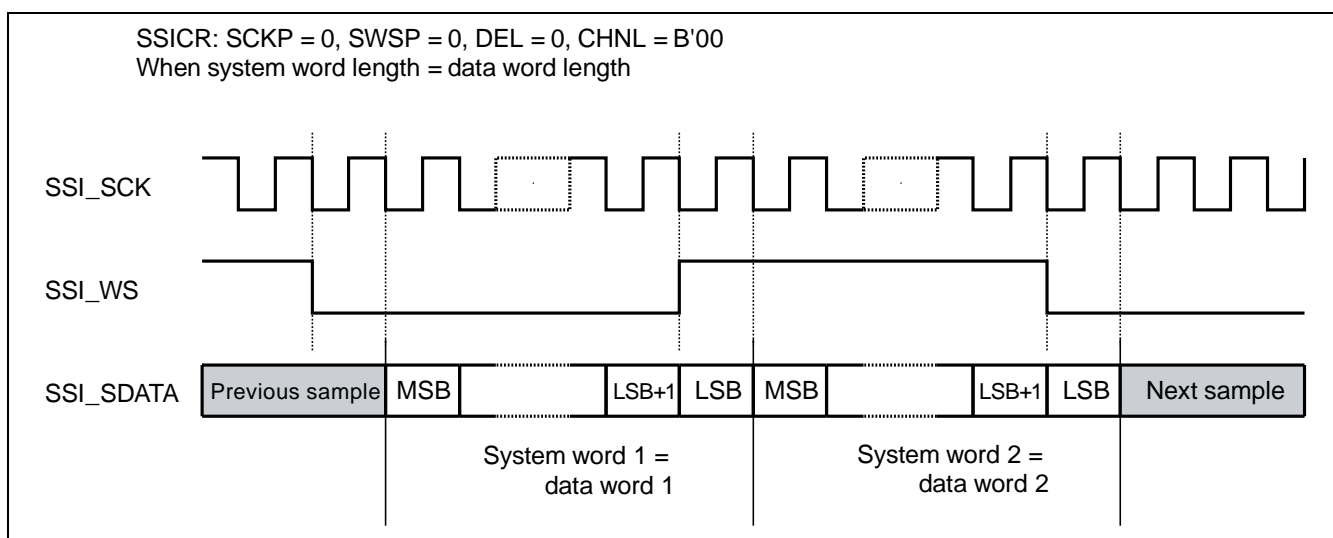


Figure 41.2 I2S Format (without Padding)

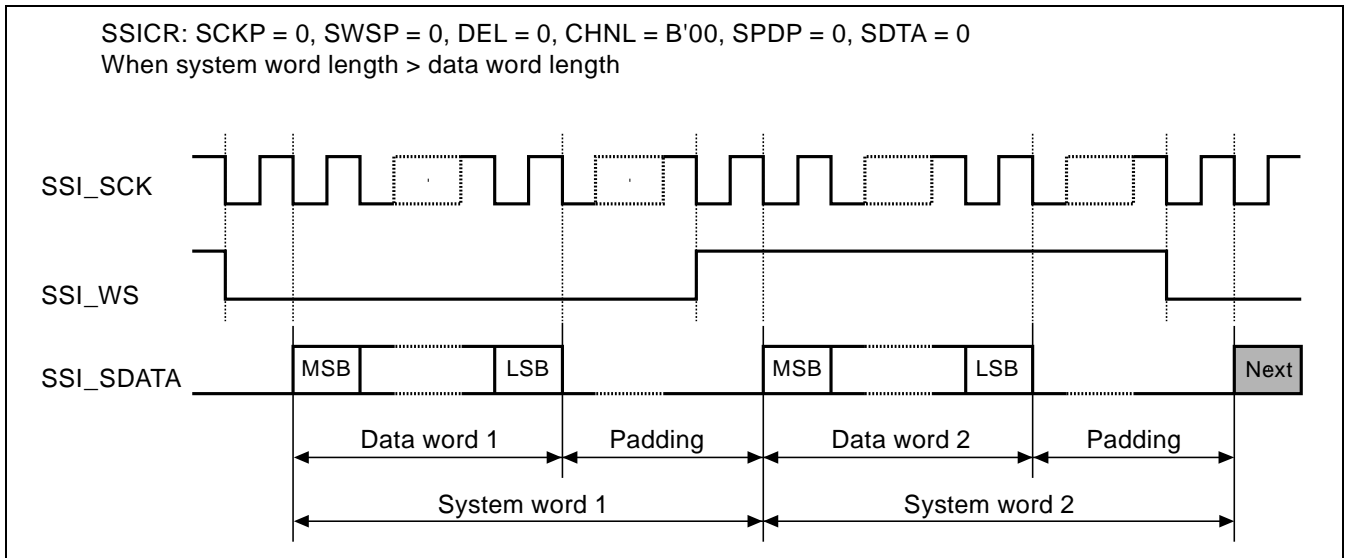


Figure 41.3 I2S Format (with Padding)

Figure 41.4 shows the left-aligned format and Figure 41.5 shows the right-aligned format. Padding is assumed in both cases, but may not be present in a final implementation if the system word length equals the data word length.

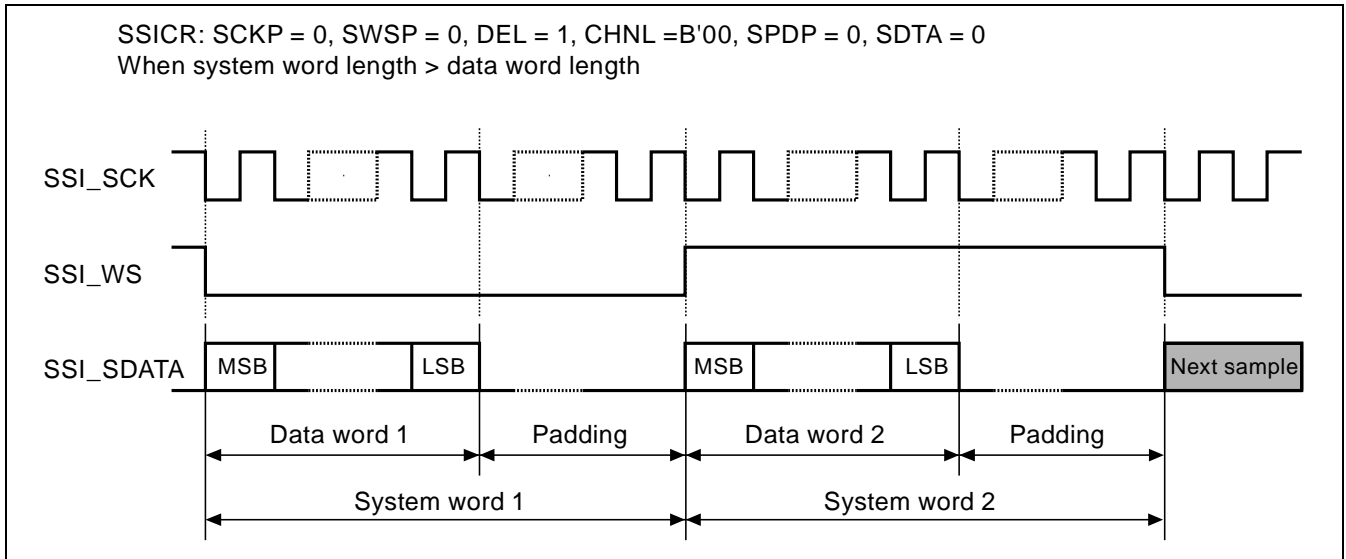


Figure 41.4 Left-Aligned Format
 (Transmitting and Receiving in the Order of Serial Data and Padding Bits)

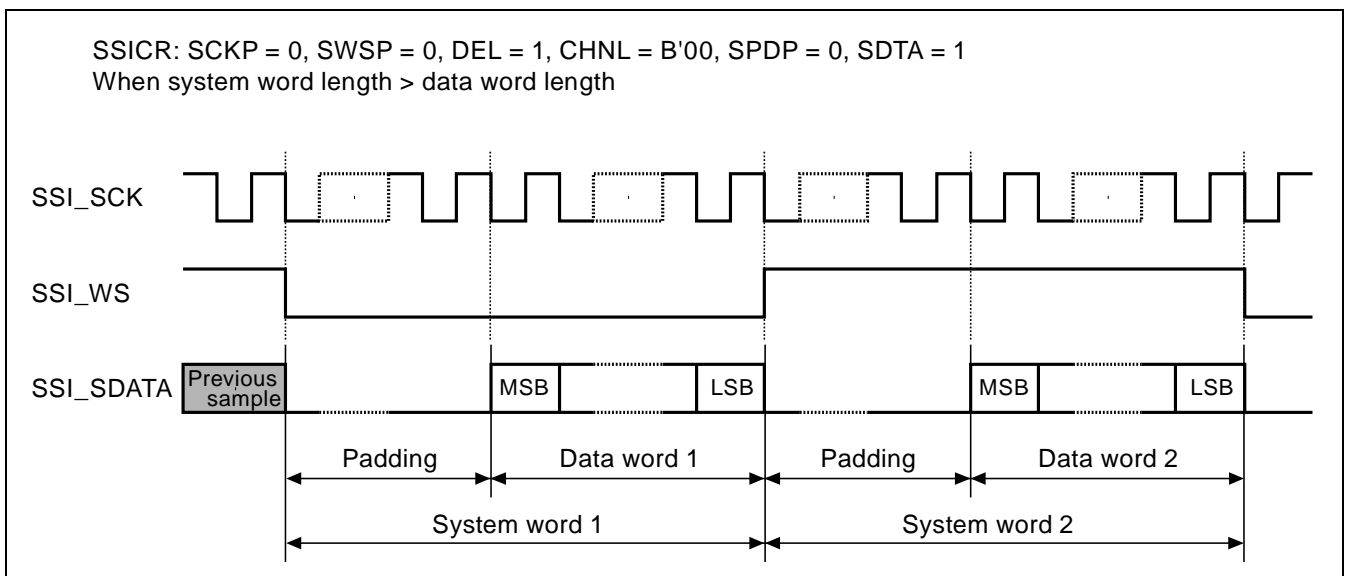


Figure 41.5 Right-Aligned Format
 (Transmitting and Receiving in the Order of Padding Bits and Serial Data)

(7) Multi-channel formats

Some devices extend the definition of the I2S format and allow more than two channels to be transferred within two system words.

The SSI module supports the transfer of two, three, and four channels by using the CHNL, SWL and DWL bits only when the system word length (SWL) is greater than or equal to the data word length (DWL) multiplied by channels (CHNL).

Table 41.5 shows the number of padding bits for each of the valid setting. If setting is not valid, "—" is indicated instead of a number.

Table 41.5 The Number of Padding Bits for Each Valid Setting

| Padding Bits Per System Word | | DWL[2:0] | B'000 | B'001 | B'010 | B'011 | B'100 | B'101 | B'110 | |
|------------------------------|----------------------------------|-----------|---------------------|-------|-------|-------|-------|-------|-------|-----|
| CHNL [1:0] | Decoded Channels per System Word | SWL [2:0] | Decoded Word Length | 8 | 16 | 18 | 20 | 22 | 24 | 32 |
| B'00 | 1 | B'000 | 8 | 0 | — | — | — | — | — | — |
| | | B'001 | 16 | 8 | 0 | — | — | — | — | — |
| | | B'010 | 24 | 16 | 8 | 6 | 4 | 2 | 0 | — |
| | | B'011 | 32 | 24 | 16 | 14 | 12 | 10 | 8 | 0 |
| | | B'100 | 48 | 40 | 32 | 30 | 28 | 26 | 24 | 16 |
| | | B'101 | 64 | 56 | 48 | 46 | 44 | 42 | 40 | 32 |
| | | B'110 | 128 | 120 | 112 | 110 | 108 | 106 | 104 | 96 |
| | | B'111 | 256 | 248 | 240 | 238 | 236 | 234 | 232 | 224 |
| B'01 | 2 | B'000 | 8 | — | — | — | — | — | — | — |
| | | B'001 | 16 | 0 | — | — | — | — | — | — |
| | | B'010 | 24 | 8 | — | — | — | — | — | — |
| | | B'011 | 32 | 16 | 0 | — | — | — | — | — |
| | | B'100 | 48 | 32 | 16 | 12 | 8 | 4 | 0 | — |
| | | B'101 | 64 | 48 | 32 | 28 | 24 | 20 | 16 | 0 |
| | | B'110 | 128 | 112 | 96 | 92 | 88 | 84 | 80 | 64 |
| | | B'111 | 256 | 240 | 224 | 220 | 216 | 212 | 208 | 192 |
| B'10 | 3 | B'000 | 8 | — | — | — | — | — | — | — |
| | | B'001 | 16 | — | — | — | — | — | — | — |
| | | B'010 | 24 | 0 | — | — | — | — | — | — |
| | | B'011 | 32 | 8 | — | — | — | — | — | — |
| | | B'100 | 48 | 24 | 0 | — | — | — | — | — |
| | | B'101 | 64 | 40 | 16 | 10 | 4 | — | — | — |
| | | B'110 | 128 | 104 | 80 | 74 | 68 | 62 | 56 | 32 |
| | | B'111 | 256 | 232 | 208 | 202 | 196 | 190 | 184 | 160 |

| Padding Bits Per System Word | | DWL[2:0] | B'000 | B'001 | B'010 | B'011 | B'100 | B'101 | B'110 | B'111 |
|------------------------------|----------------------------------|-----------|---------------------|-------|-------|-------|-------|-------|-------|-------|
| CHNL [1:0] | Decoded Channels per System Word | SWL [2:0] | Decoded Word Length | 8 | 16 | 18 | 20 | 22 | 24 | 32 |
| B'11 | 4 | B'000 | 8 | — | — | — | — | — | — | — |
| | | B'001 | 16 | — | — | — | — | — | — | — |
| | | B'010 | 24 | — | — | — | — | — | — | — |
| | | B'011 | 32 | 0 | — | — | — | — | — | — |
| | | B'100 | 48 | 16 | — | — | — | — | — | — |
| | | B'101 | 64 | 32 | 0 | — | — | — | — | — |
| | | B'110 | 128 | 96 | 64 | 56 | 48 | 40 | 32 | 0 |
| | | B'111 | 256 | 224 | 192 | 184 | 176 | 168 | 160 | 128 |

When the SSI module acts as a transmitter, each word written to SSITDR is transmitted to the serial audio bus in the order they are written. When the SSI module acts as a receiver, each word received by the serial audio bus is read in the order received from SSIRDR.

Figure 41.6 to Figure 41.8 show how two, three and four channels are transferred to the serial audio bus. Note that there are no padding bits in the first example, the second example is left-aligned and the third is right-aligned. This selection is arbitrary and is just for demonstration purposes only.

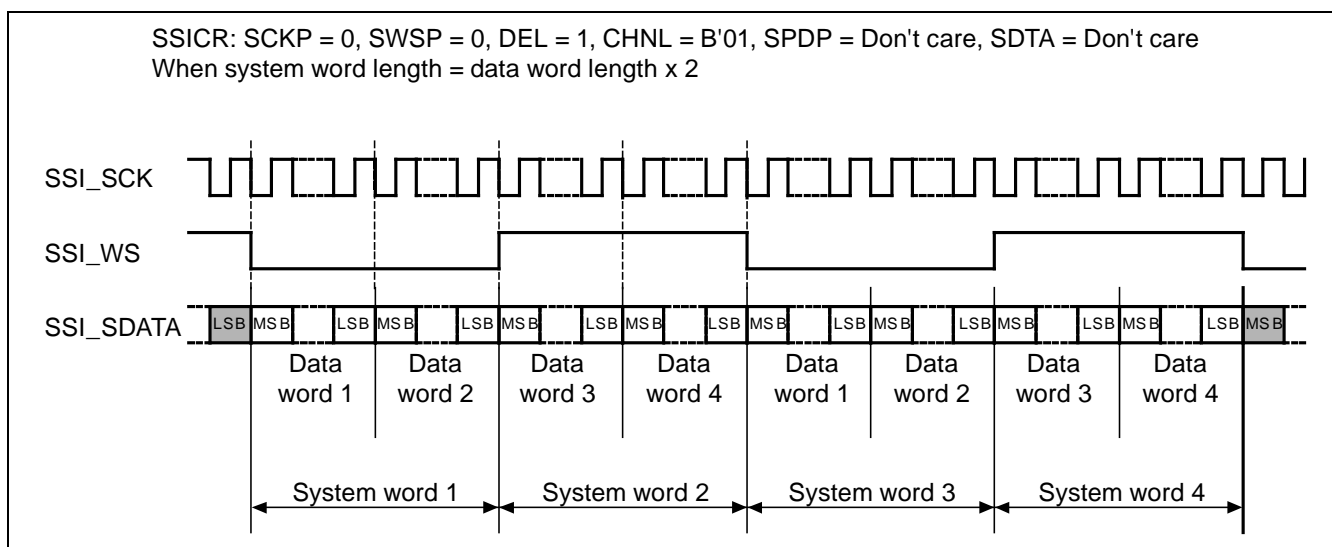


Figure 41.6 Multi-Channel Format (Two Channels without Padding)

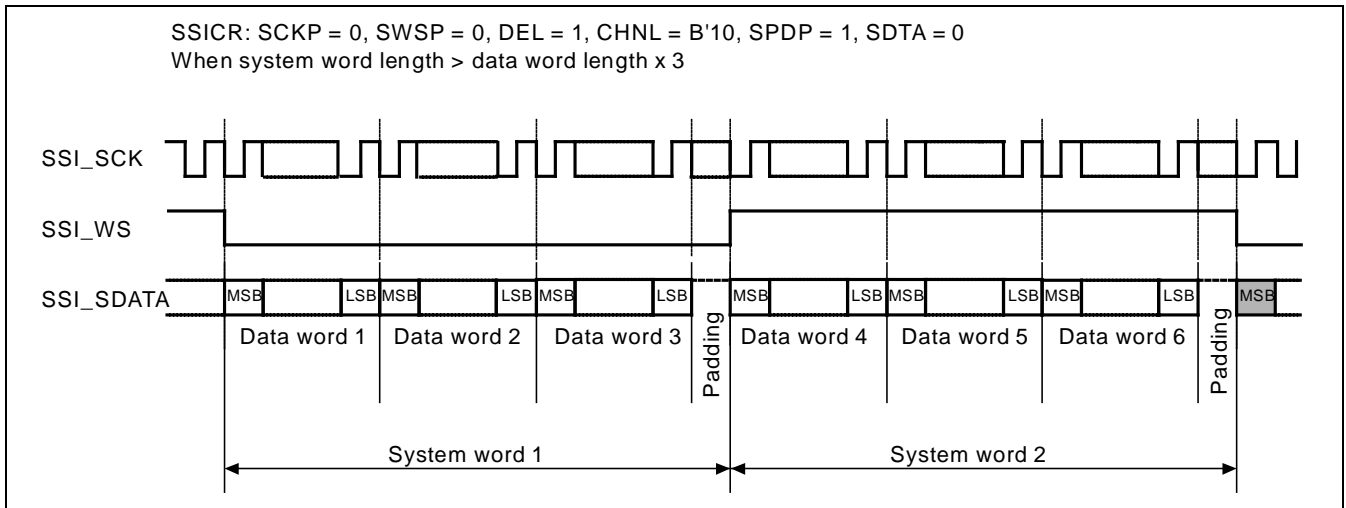


Figure 41.7 Multi-Channel Format (Three Channels with High Padding)

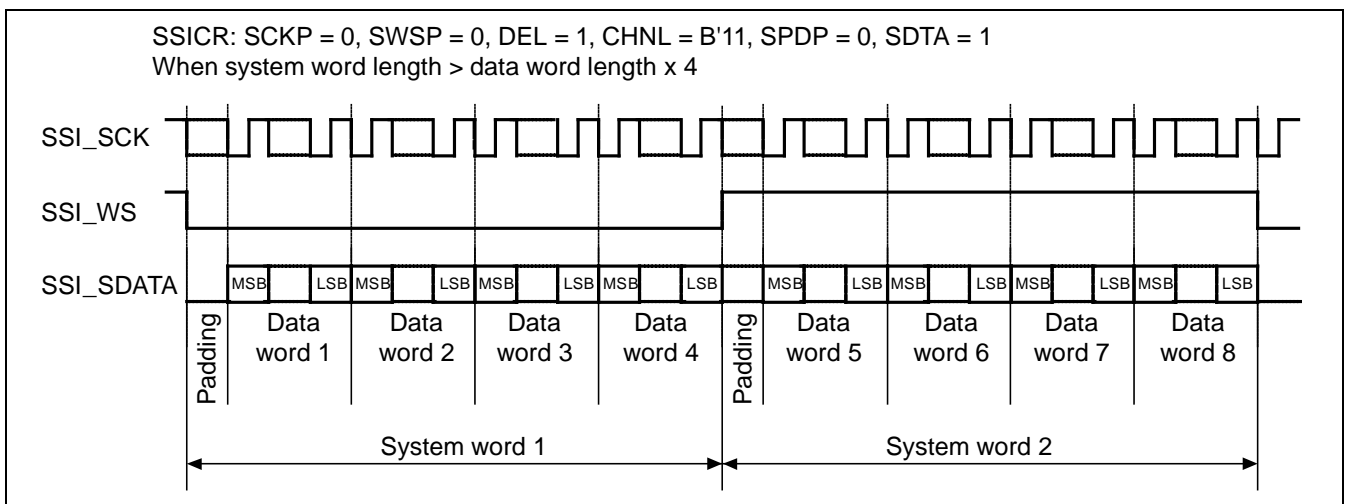


Figure 41.8 Multi-Channel Format (Four Channels; Transmitting and Receiving in the Order of Padding Bits and Serial Data; with Padding)

(8) TDM formats

The TDM format is used for connecting the SSI with a multi-channel device which supports TDM. The TDM format is set by using the WS_MODE and MONO bits in SSIWSR. When the SWSP bit in SSICR is 0 with the TDM format, SSI_WS is driven high over the period of system word 1, and pulled low otherwise. When the SWSP bit is 1, SSI_WS is pulled low over the period of system word 1, and driven high otherwise. The pulses generated on the SSI_WS signal are referred to as SYNC pulses.

Figure 41.9 and Figure 41.10 show the TDM formats without padding and with padding.

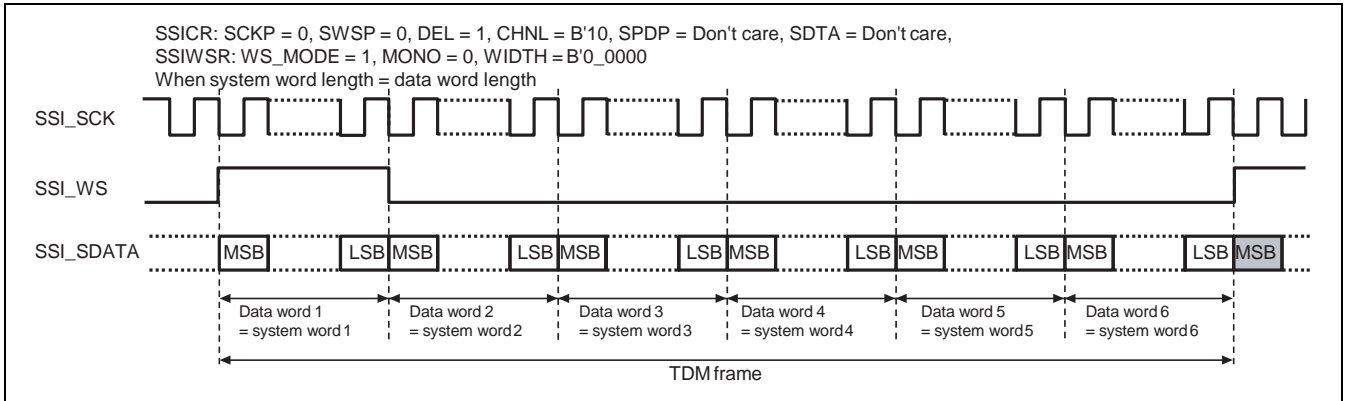


Figure 41.9 TDM Format (Six System Words, without Padding)

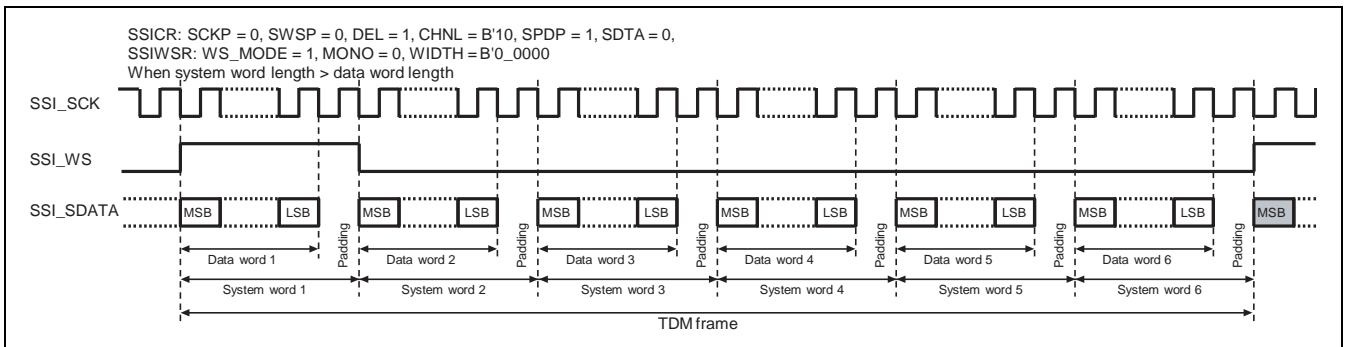


Figure 41.10 TDM Format (Six System Words, with Padding)

The following describes operation in each mode:

(a) Master transmitter

By a transfer start trigger (setting the EN bit in the control register to 1), a transfer of system word 1 begins synchronously with the SYNC pulses.

By a transfer stop trigger (setting the EN bit in the control register to 0), a transfer is stopped at the end of the currently-transferred system word and the SDATA signal is output according to the SPDP setting in SSICR (when SPDP = 0, a low-level signal is output).

If transmit data is not ready in the SSI module during transmission, an underflow will occur. When an underflow occurs, data to be output for the SYNC pulses cannot be determined. Therefore, stop and reconfigure the transfer.

(b) Master receiver

By a transfer start trigger (setting the EN bit in the control register to 1), a reception of system word 1 data begins at the point where the SSI module recognizes a SYNC pulse.

By a transfer stop trigger (setting the EN bit in the control register to 0), a reception is stopped at the end of the currently-transferred system word.

The receive data register should not be read when it does not have data. If read, an underflow will occur. When an underflow occurs, stop and reconfigure the transfer.

The receive data should not be written to before read. If written to, an overflow will occur. When an overflow occurs, stop and reconfigure the transfer.

(c) Slave transmitter

By a transfer start trigger (setting the EN bit in the control register to 1), a transfer of system word 1 begins synchronously with the SYNC pulses.

By a transfer stop trigger (setting the EN bit in the control register to 0), a transfer is stopped at the end of the currently-transferred system word and the SDATA signal is output according to the SPDP setting in SSICR (when SPDP = 0, a low-level signal is output).

If transmit data is not ready in the SSI module during transmission, an underflow will occur. When an underflow occurs, data to be output for SYNC pulses cannot be determined. Therefore, stop and re-set the transfer.

Transfers cannot be performed if the SCK signal and SYNC pulses are not provided to the SSI module during transfer.

(d) Slave receiver

By a transfer start trigger (setting the EN bit in the control register to 1), a reception of system word 1 data begins at the point where the SSI module recognizes a SYNC pulse.

By a transfer stop trigger (setting the EN bit in the control register to 0), a reception is stopped at the end of the currently-transferred system word.

The receive data should not be read when it does not have data. If read, an underflow will occur. When an underflow occurs, stop and reconfigure the transfer.

The receive data register should not be written to before read. If written to, an overflow will occur. When an overflow occurs, stop and reconfigure the transfer.

Transfer cannot proceed if the SCK signal and SYNC pulses are not being provided to the SSI module during transfer.

(9) Monaural format

The monaural format is set by using the WS_MODE and MONO bits in SSIWSR. The available data lengths are 8 bits and 16 bits. When the SWSP bit in SSICR is 0 and the monaural format is selected, SSI_WS is driven high over the period from bit 1 to bit 31, and pulled low otherwise. When the SWSP bit is 1, SSI_WS is pulled low over the period from bit 1 to bit 31, and driven high otherwise. The pulses generated as the SSI_WS signal are referred to as SYNC pulses. The width of the SYNC pulses can be set to a value in bits between 1 and 31 by using the WIDTH bits in SSIWSR. The setting must be smaller than the system word length (SSICR.SWL). In the monaural format, the system word length is equal to one monaural frame period.

Figure 41.11 shows the left-aligned monaural format and Figure 41.12 shows the monaural format with a delay of one bit-period.

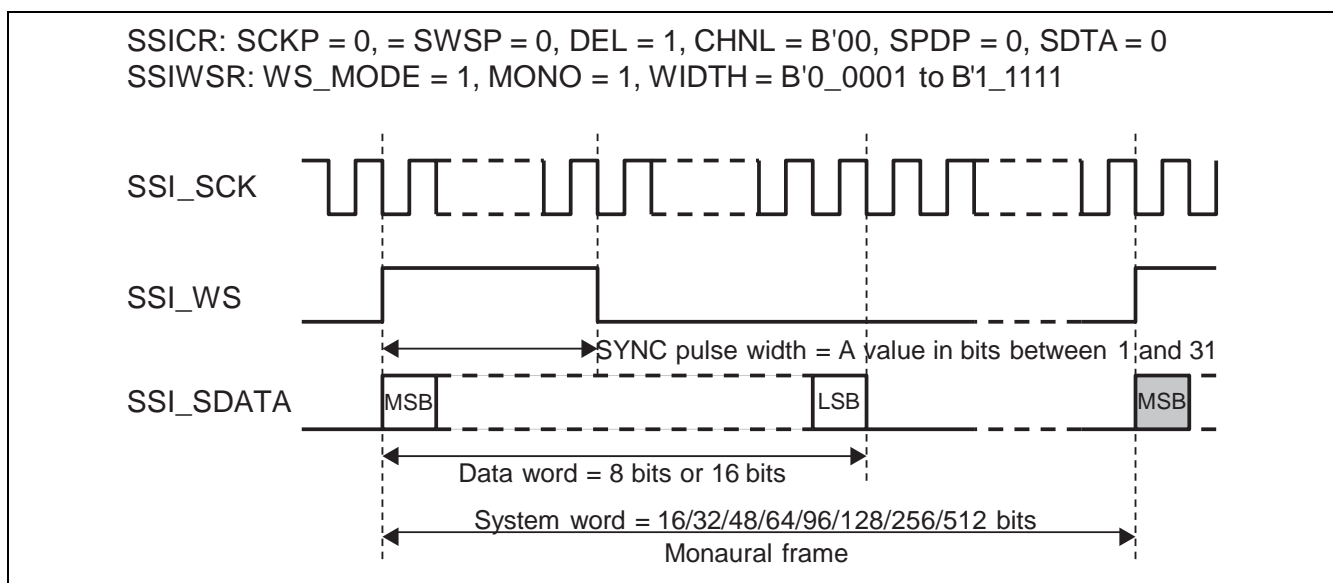


Figure 41.11 Monaural Format (Left-Aligned)

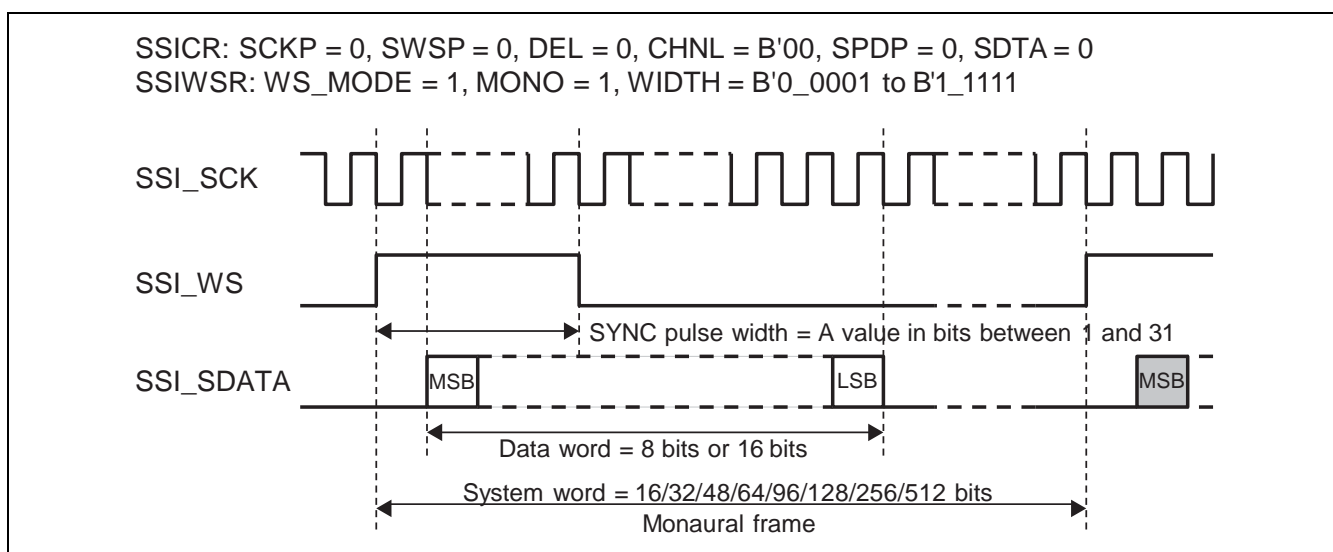
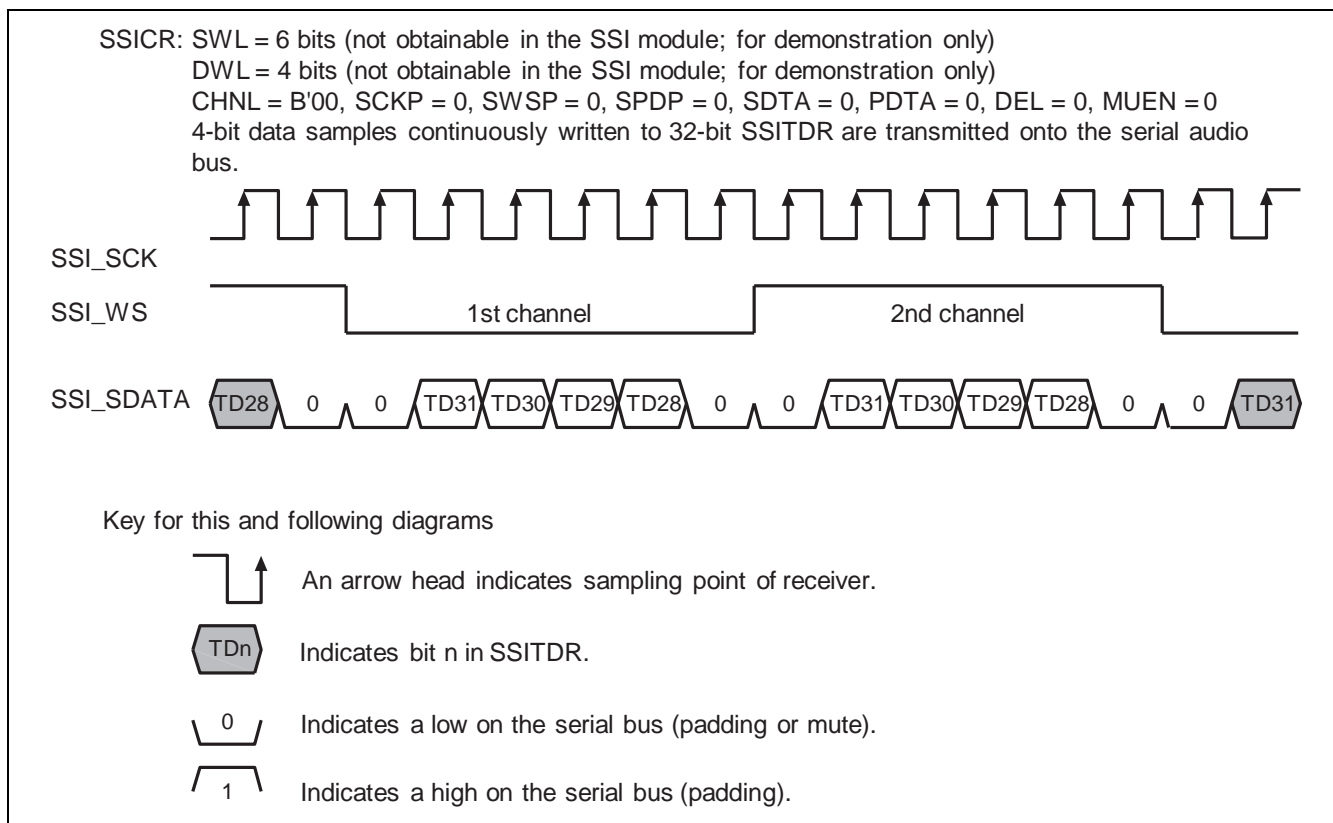


Figure 41.12 Monaural Format (with 1-bit Delay)

(10) Configuration bits

Several more configuration bits are shown below. These bits are not mutually exclusive, but some combinations may not be useful for any other device.

These configuration bits are described below with reference to Figure 41.13, Basic Sample Format.



**Figure 41.13 Basic Sample Format
 (Transmit Mode with Example System/Data Word Length)**

Figure 41.13 uses a system word length of 6 bits and a data word length of 4 bits. These settings are not possible with the SSI module but are used only for clarification of the other configuration bits.

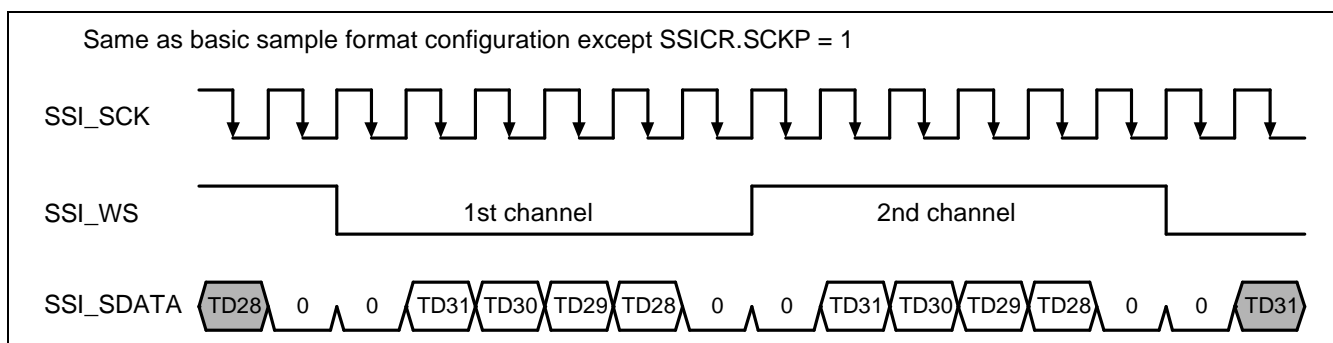


Figure 41.14 Inverted Clock

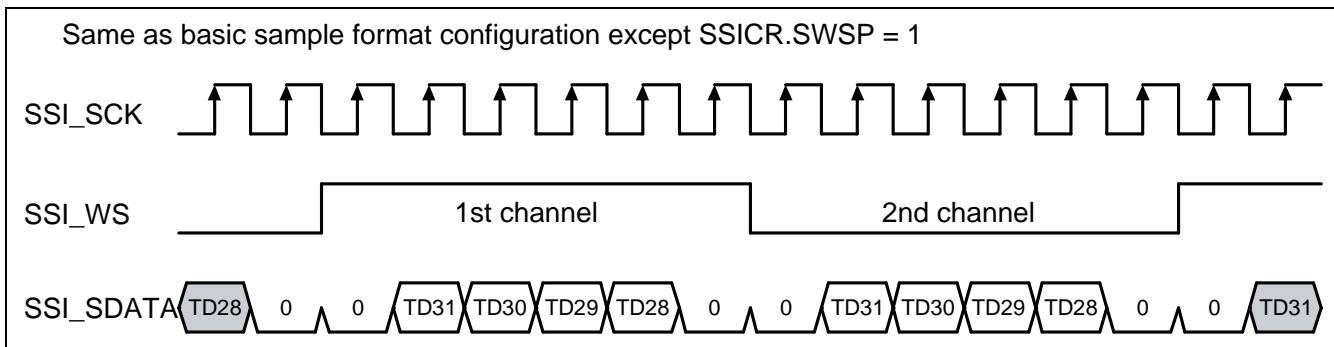


Figure 41.15 Inverted Word Select

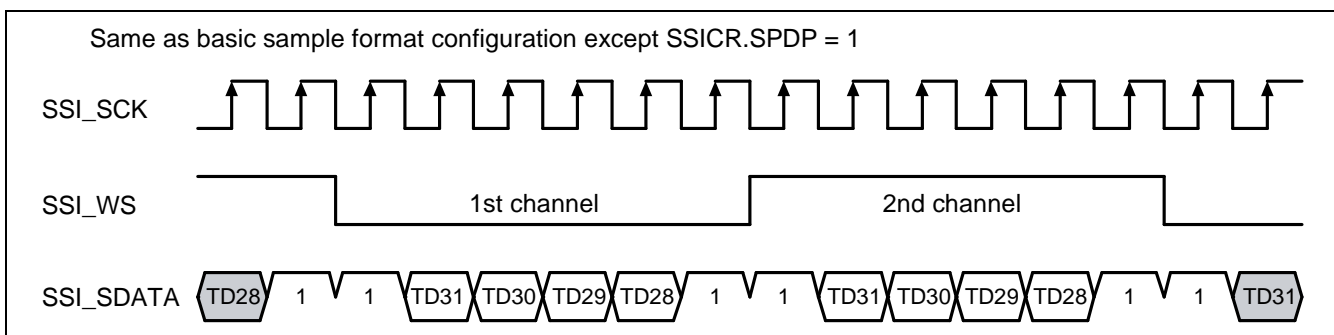


Figure 41.16 Inverted Padding Polarity

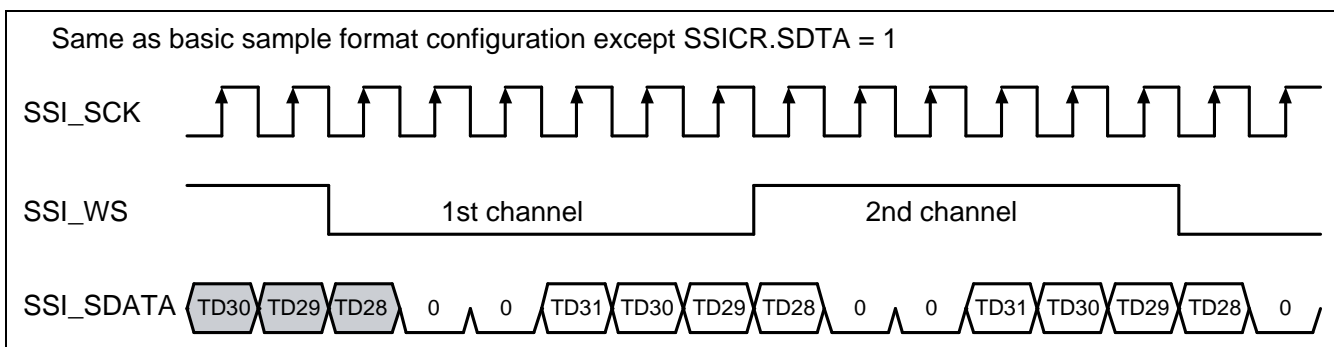


Figure 41.17 Transmitting and Receiving in the Order of Padding Bits and Serial Data; with Delay

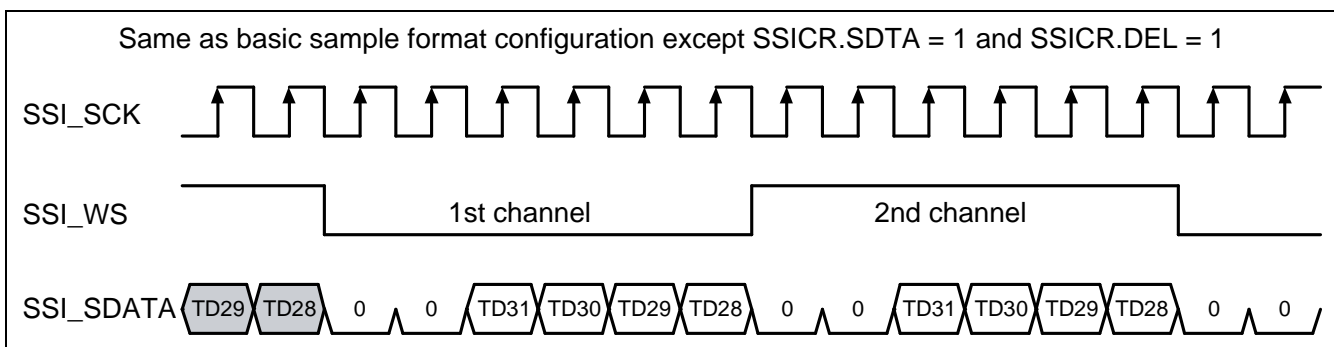


Figure 41.18 Transmitting and Receiving in the Order of Padding Bits and Serial Data; without Delay

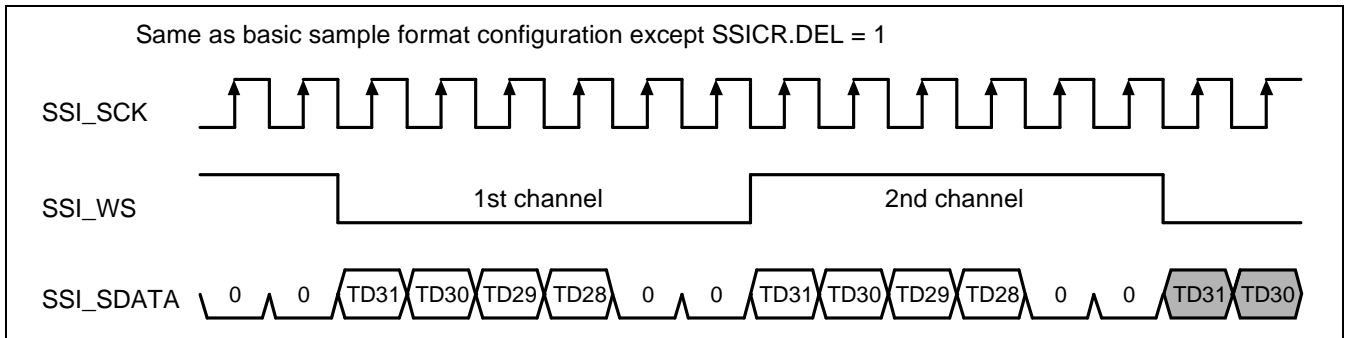


Figure 41.19 Transmitting and Receiving in the Order of Serial Data and Padding Bits; without Delay

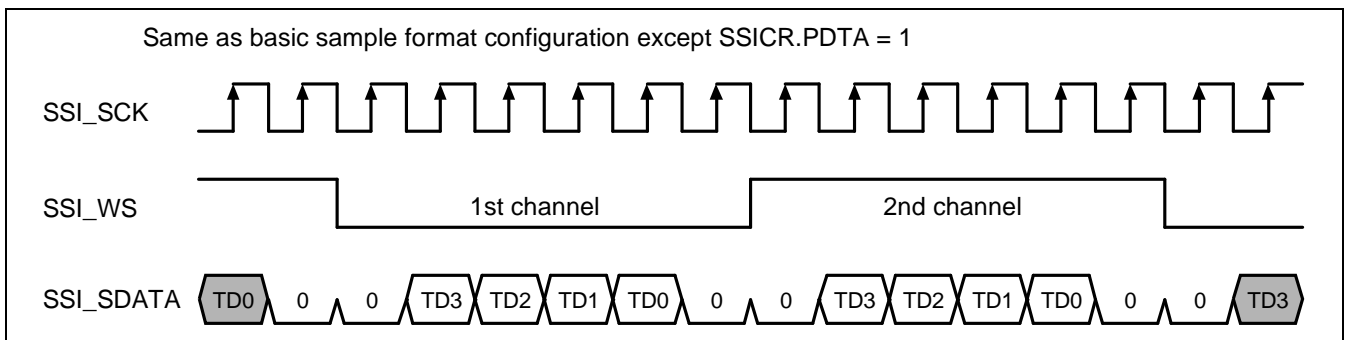


Figure 41.20 Parallel Right-Aligned with Delay

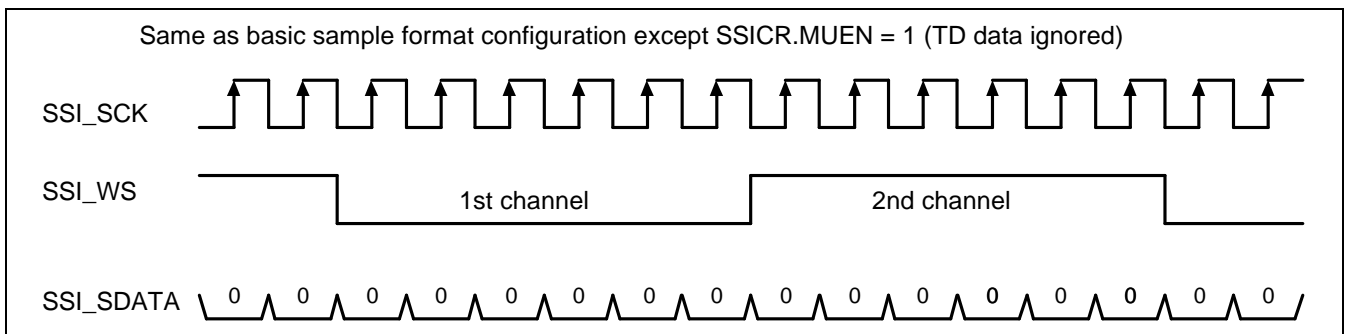


Figure 41.21 Mute Enabled

(11) WS continue function

The WS continue function is used to output SSI_WS signal continuously regardless of whether data transfer is enabled or disabled. The WS continue function can be set by using the CONT bit in SSIWSR. This function can be combined with master mode only. When this function is enabled, the SSI_WS signal continues to operate even if the EN bit in SSICR is cleared to 0 (stopping a transfer). When this function is disabled, the SSI_WS signal operates synchronously with the EN bit.

Figure 41.22 and Figure 41.23 show operations when the WS continue function is enabled and disabled, respectively.

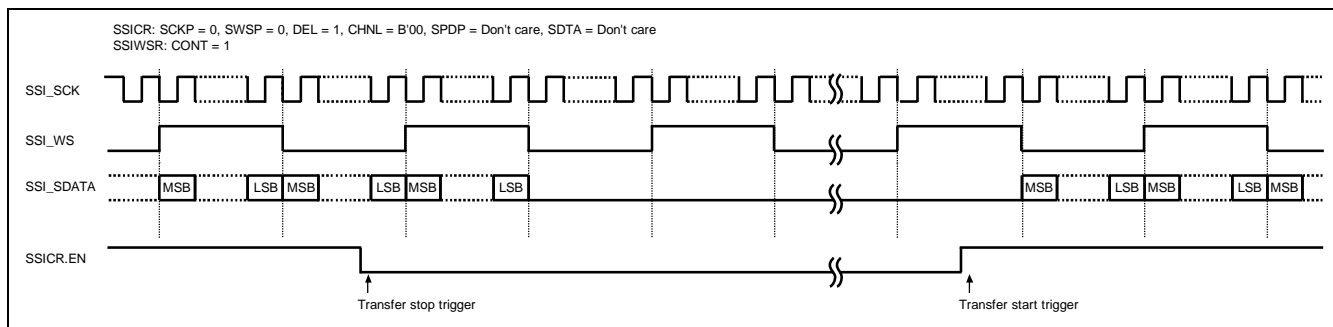


Figure 41.22 WS Continue Function Operation (When the Function is Enabled)

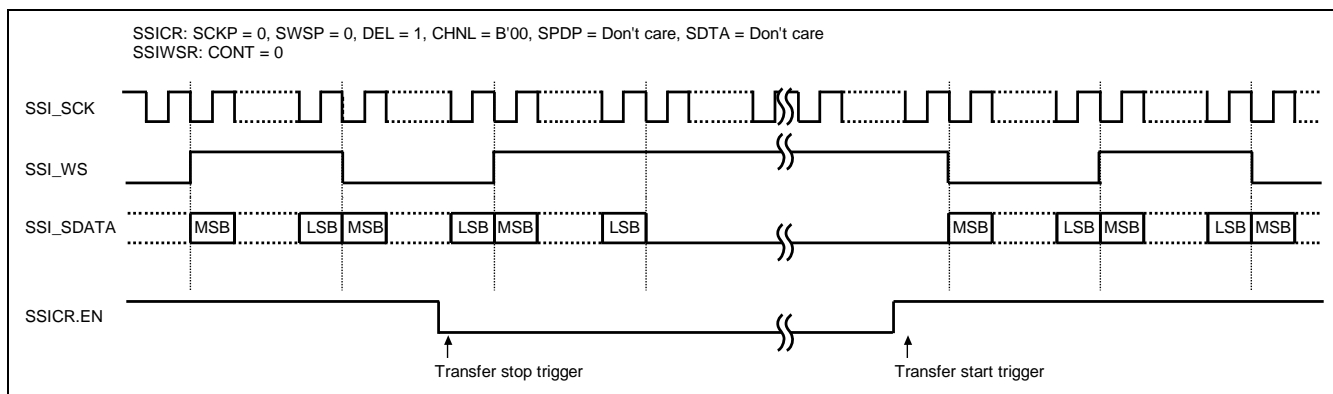


Figure 41.23 WS Continue Function Operation (When the Function is Disabled)

(12) Bit alignment in transferred data

Figure 41.24 to Figure 41.36 show the bit alignment in transmit or receive data. Only the MSB first alignment is available. The waveforms in Figure 41.24 to Figure 41.36 are setting examples without delay (SSICR.DEL is set to 1, the number of bits specified with SSICR.DWL is the same as that specified with SSICR.SWL).

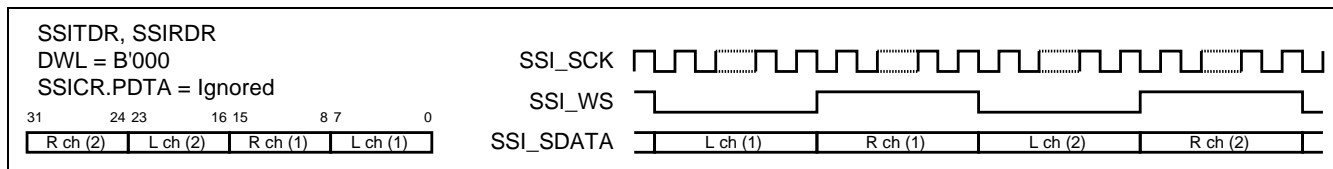


Figure 41.24 Bit Alignment in 8-Bit Stereo Format

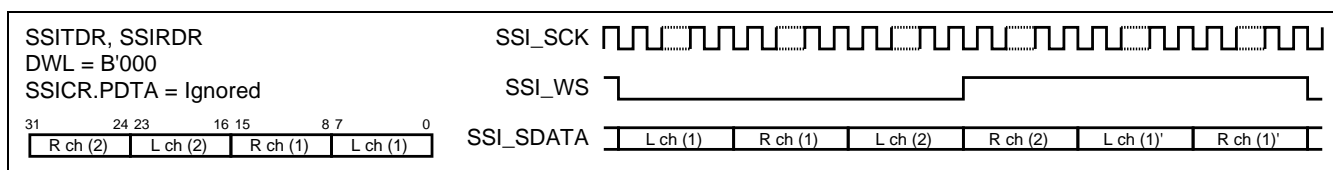


Figure 41.25 Bit Alignment in 8-Bit Multi-Channel Format (Three Channels)

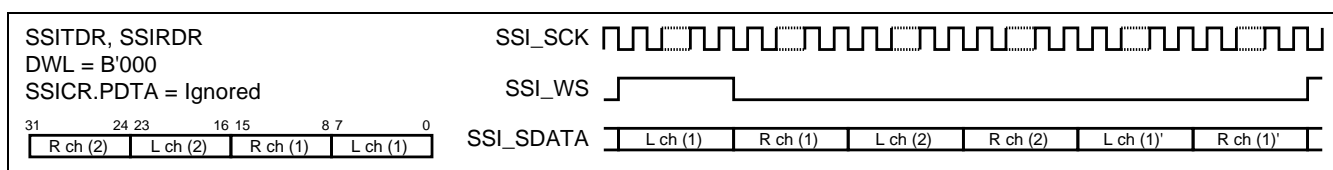


Figure 41.26 Bit Alignment in 8-Bit TDM Format (Six System Words)

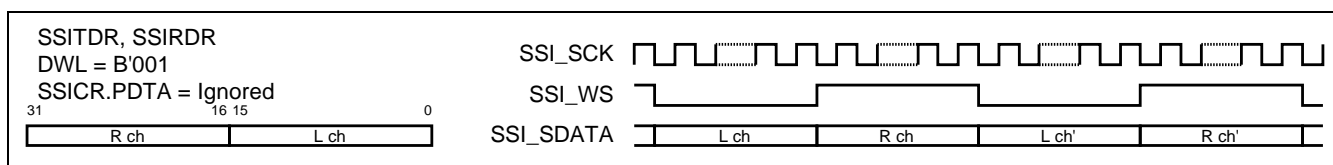


Figure 41.27 Bit Alignment in 16-Bit Stereo Format

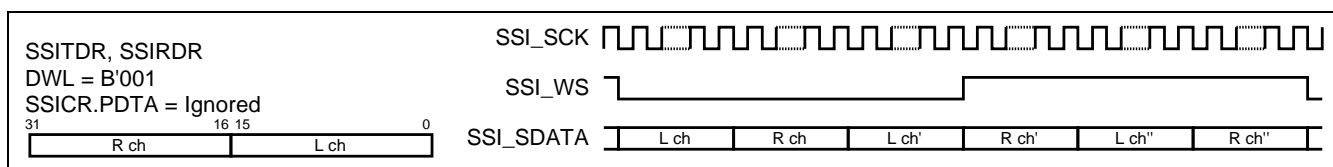


Figure 41.28 Bit Alignment in 16-Bit Multi-Channel Format (Three Channels)

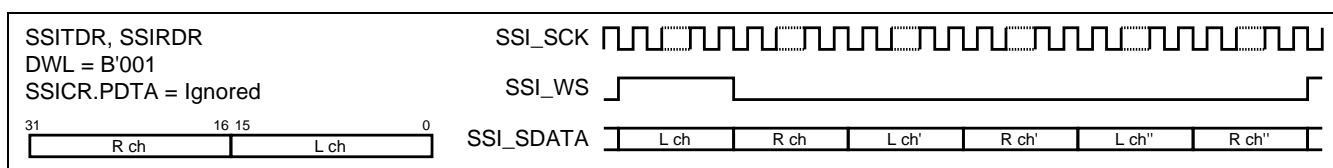


Figure 41.29 Bit Alignment in 16-Bit TDM Format (Six System Words)

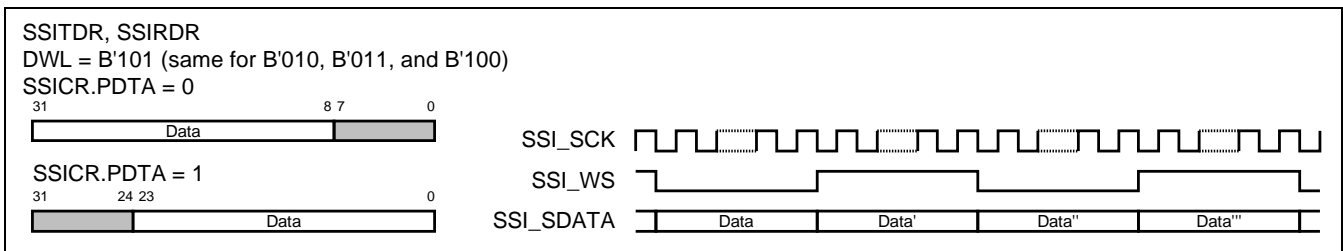


Figure 41.30 Bit Alignment in 18-/20-/22-/24-Bit Stereo Format

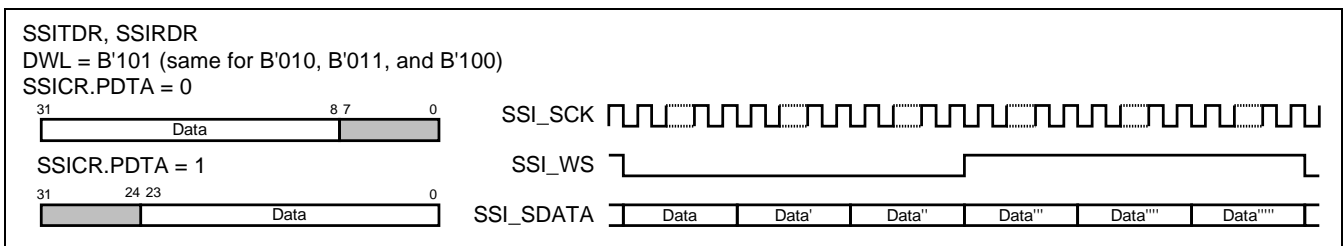


Figure 41.31 Bit Alignment in 18-/20-/22-/24-Bit Multi-Channel Format (Three Channels)

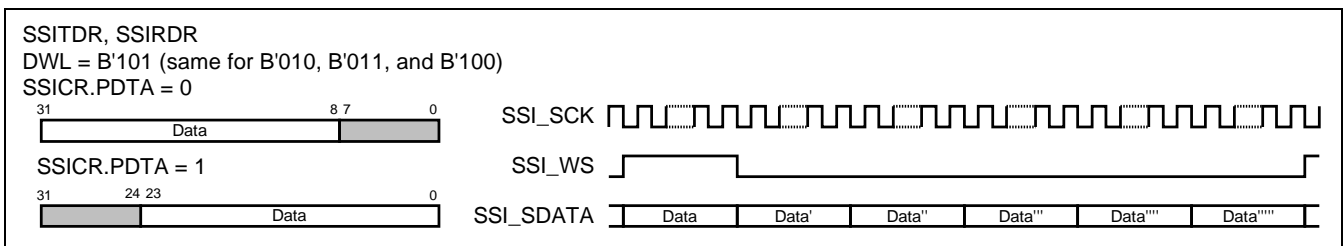


Figure 41.32 Bit Alignment in 18-/20-/22-/24-Bit TDM Format (Six System Words)

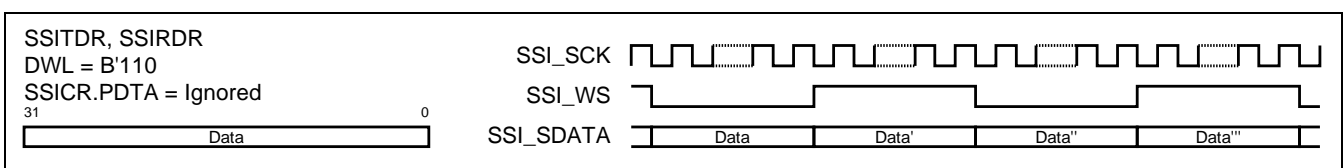


Figure 41.33 Bit Alignment in 32-Bit Stereo Format

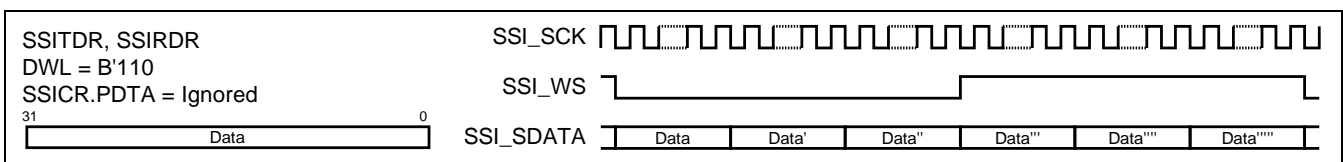


Figure 41.34 Bit Alignment in 32-Bit Multi-Channel Format (Three Channels)

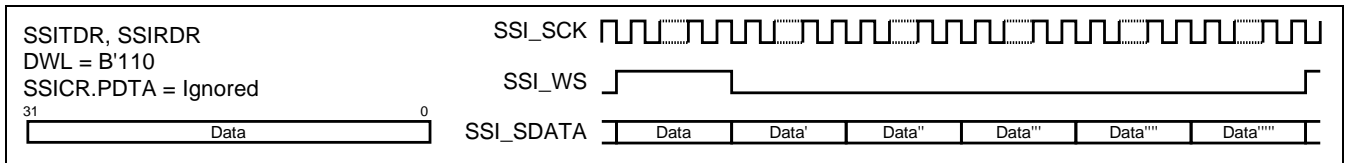


Figure 41.35 Bit Alignment in 32-Bit TDM Format (Six System Words)

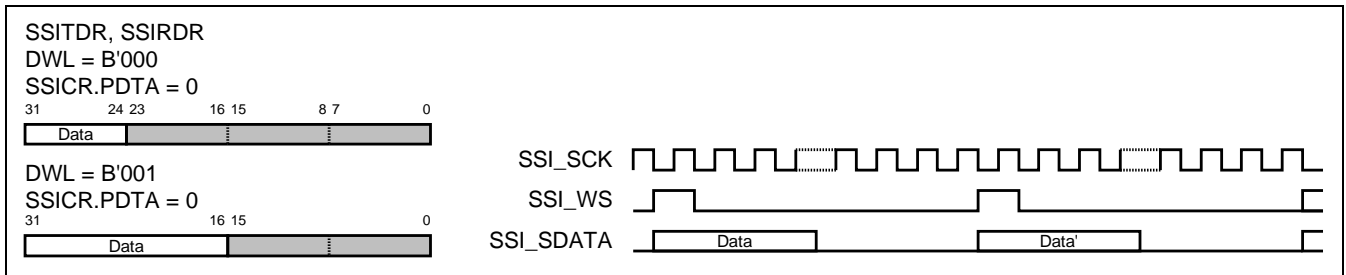


Figure 41.36 Bit Alignment in 8-/16-Bit Monaural Format

41.3.2 Operating Modes

There are three modes of operation: configuration, module enabled and module disabled. Figure 41.37 shows how the module enters each of these modes.

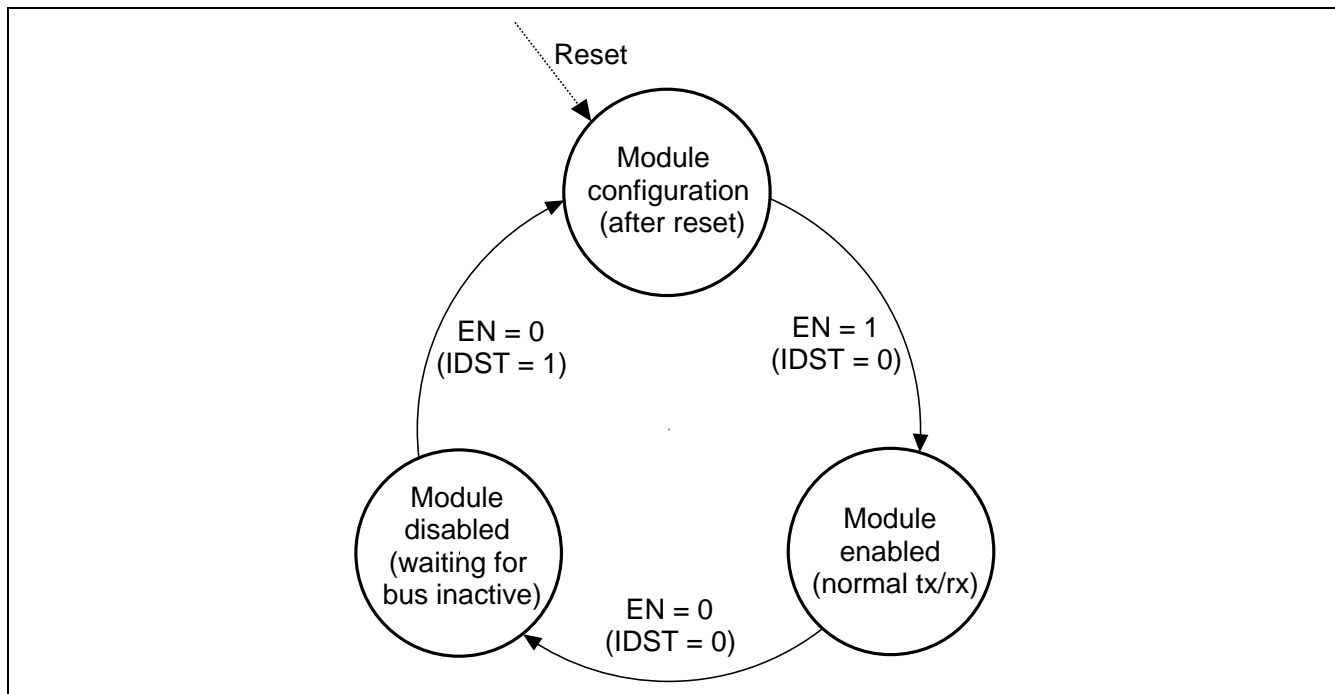


Figure 41.37 Operating Modes

(1) Configuration mode

This mode is entered after the module is released from reset. All required configuration fields in the control register should be defined in this mode, before the SSI module is enabled by setting the EN bit.

Setting the EN bit causes the module to enter the module enabled mode.

(2) Module enabled mode

Operation of the module in this mode is dependent on the operating mode selected. For details, refer to section 41.3.3 Transmit Operation and section 41.3.4 Receive Operation.

41.3.3 Transmit Operation

Transmission can be controlled either by DMA or interrupt.

DMA control is preferred to reduce the processor load. In DMA control mode, the processor will only receive interrupts if there is an underflow or overflow of data or the DMAC has finished its transfer.

The alternative method is using the interrupts that the SSI module generates to supply data as required. This mode has a higher interrupt load as the module is only double buffered and will require data to be written at least every system word period.

When disabling the module, the SSI clock* must remain present until the SSI module is in idle state, indicated by the SSISR.IIRQ bit.

Figure 41.38 shows the transmit operation in DMA control mode, and Figure 41.39 shows the transmit operation in interrupt control mode.

Note: * Input clock from the SSI_SCK pin when SCKD = 0.
Input clock from the CLK_FS pin when SCKD = 1.

Use the flowcharts in Figure 41.38 and Figure 41.39 when bits ind9 to ind0, which are covered in the description of SSI Mode Register 0 (SSI_MODE0) in Section 40, Serial Sound Interface Unit (SSIU), are set to 1 (independent SSI transfer).

(1) Transmission using DMA controller

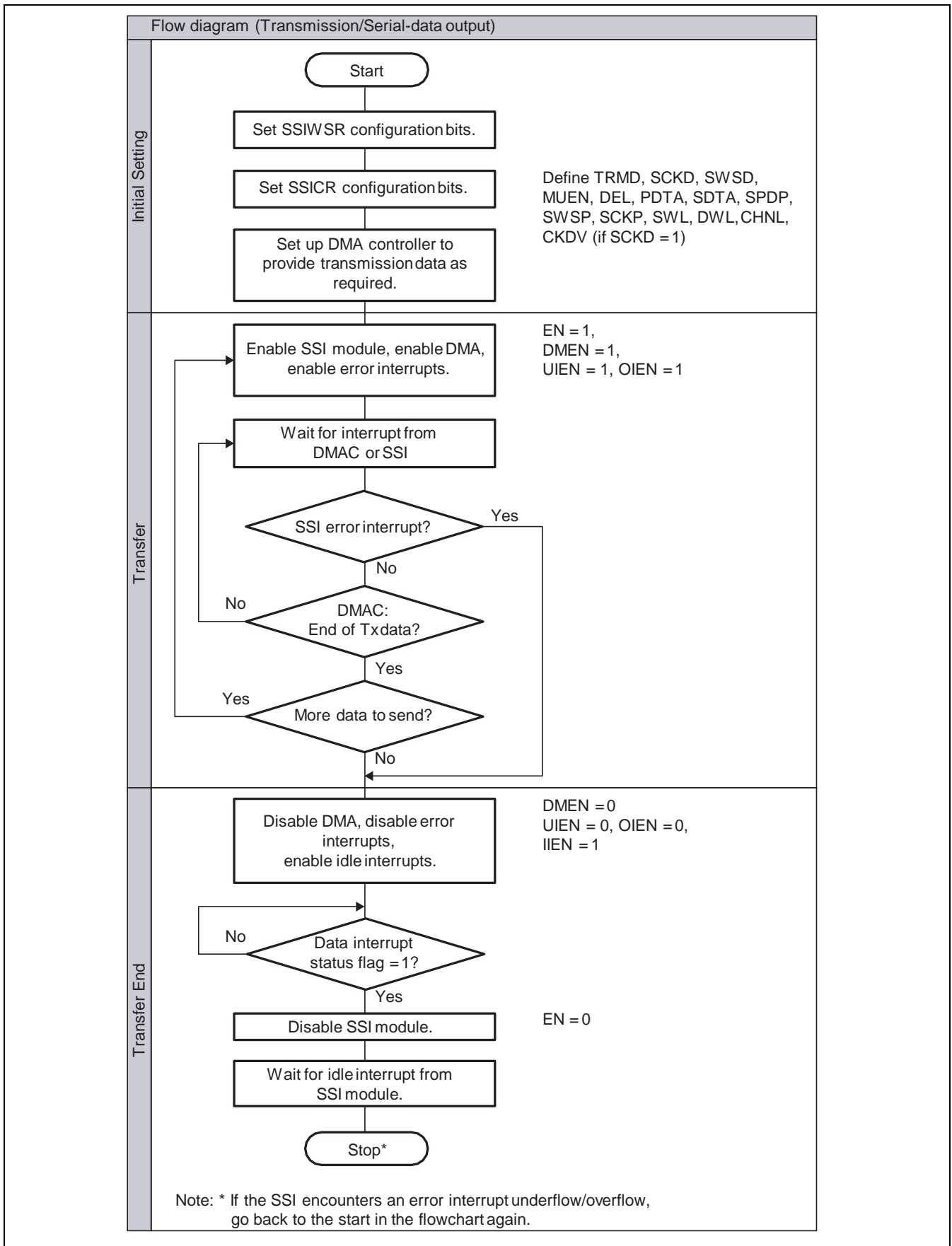


Figure 41.38 Transmission Using DMA Controller

(2) Transmission using interrupt data flow control

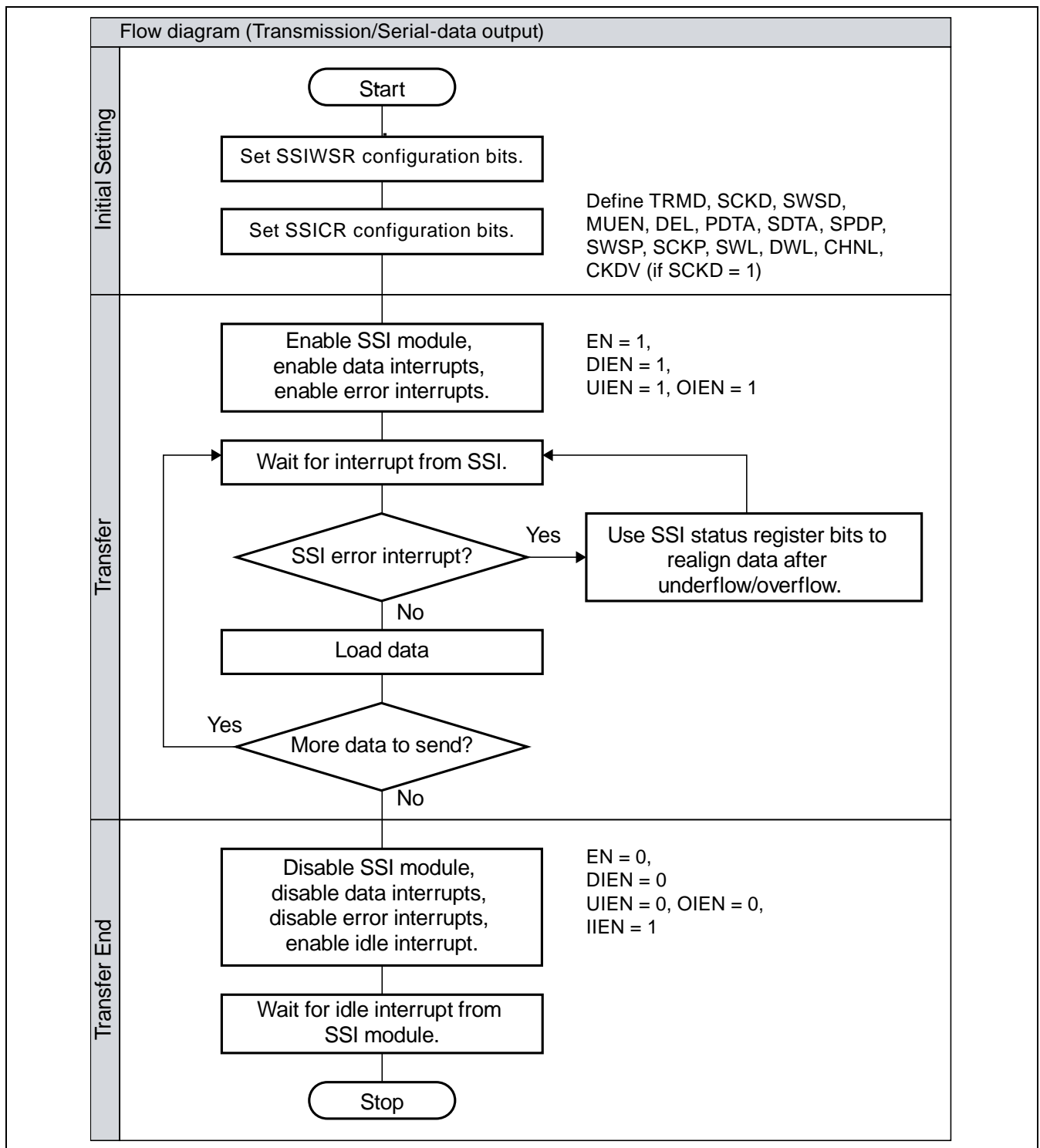


Figure 41.39 Transmission Using Interrupt Data Flow Control

41.3.4 Receive Operation

Like transmission, reception can be controlled either by DMA or interrupt.

Figure 41.40 and Figure 41.41 show the flow of operation.

When disabling the SSI module, the SSI clock* must be kept supplied until the SSISR.IIRQ bit is in idle state.

Note: * Input clock from the SSI_SCK pin when SCKD = 0.
Input clock from the CLK_FS pin when SCKD = 1.

Use the flowcharts in Figure 41.40 and Figure 41.41 when bits ind9 to ind0, which are covered in the description of SSI Mode Register 0 (SSI_MODE0) in Section 40, Serial Sound Interface Unit (SSIU), are set to 1 (independent SSI transfer).

(1) Reception using DMA controller

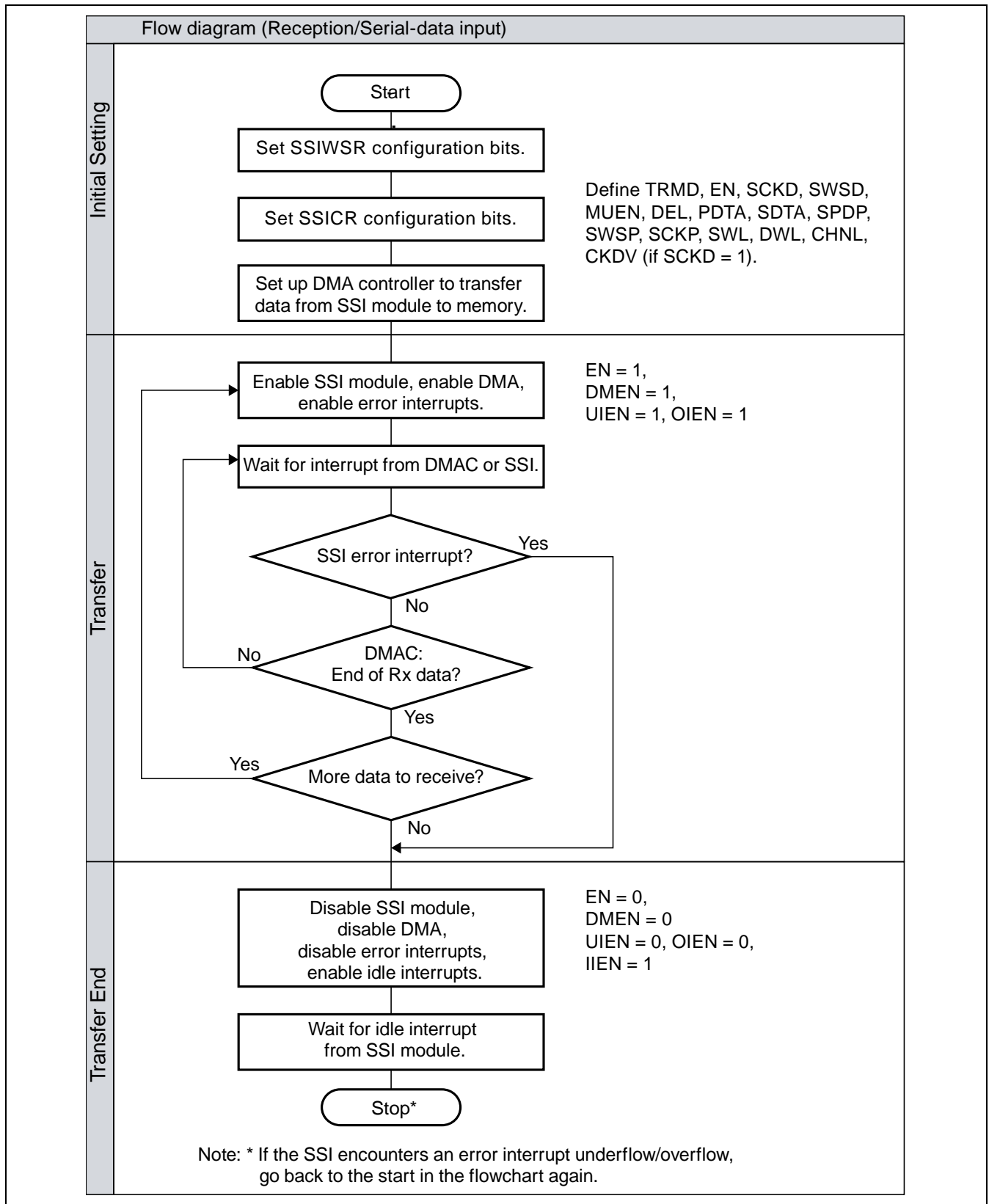


Figure 41.40 Reception Using DMA Controller

(2) Reception using interrupt data flow control

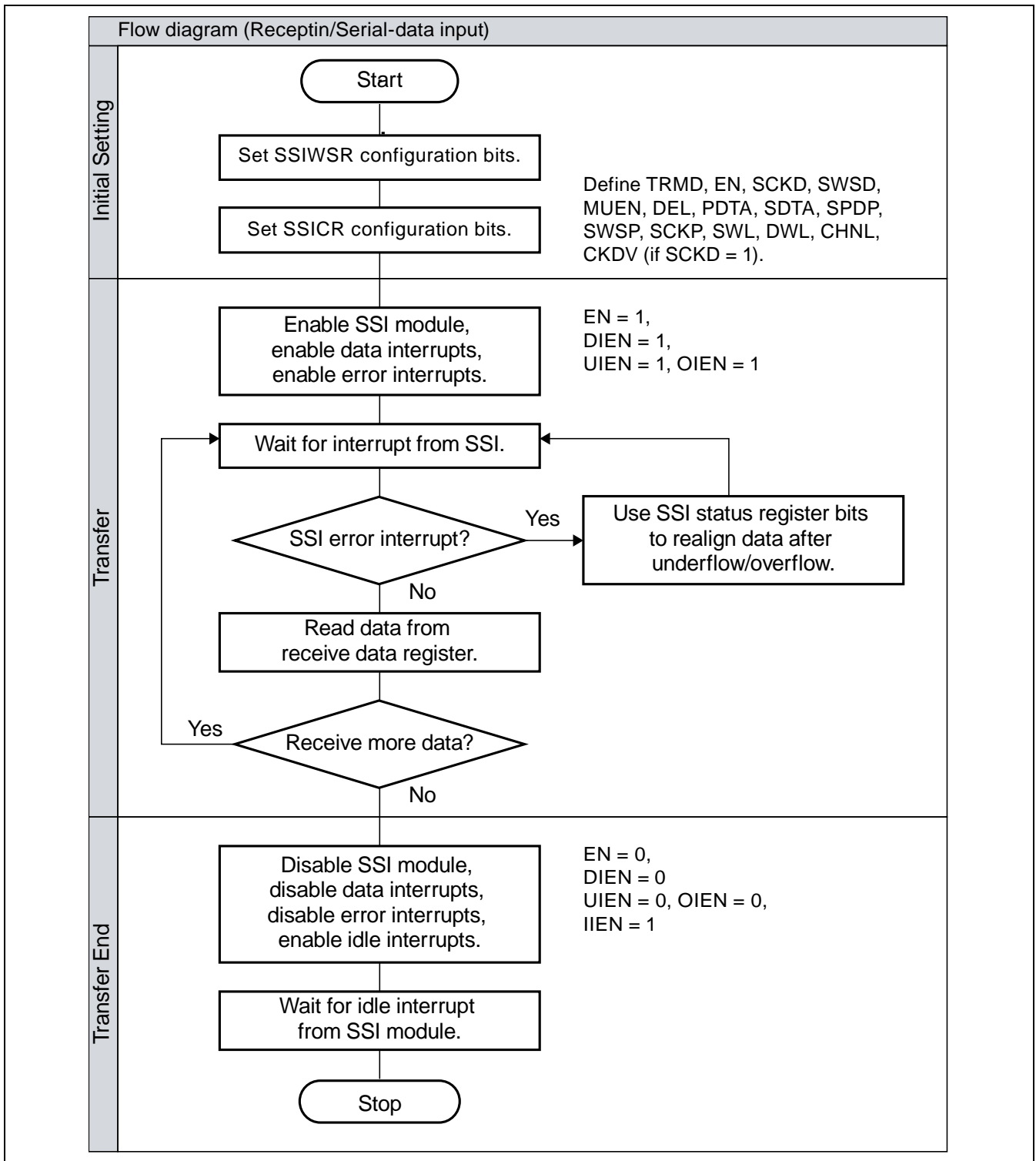


Figure 41.41 Reception Using Interrupt Data Flow Control

41.3.5 Serial Bit Clock Control

This function is used to control and select which clock is used for the serial bus interface.

If the serial clock direction is set to input ($SSICR.SCKD = 0$), the SSI module is in clock slave mode and the shift register uses the bit clock that was input to the SSI_SCK pin.

If the serial clock direction is set to output ($SSICR.SCKD = 1$), the SSI module is in clock master mode, and the shift register uses the bit clock that is input from the CLK_FS pin or the bit clock that is obtained by dividing the input from the CLK_FS pin. In the latter case, the clock input from the CLK_FS pin is divided by the ratio in the serial oversampling clock divide ratio (CKDV) in SSICR and used as the bit clock in the shift register.

In either case, the SSI_SCK pin output is the same as the bit clock.

41.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

41.4.1 Limitations from Overflow during Receive DMA Operation

If an overflow occurs while the receive DMA is in operation, the module should be restarted. The receive buffer in the SSI consists of 32-bit registers that share the L and R channels. Therefore, data to be received at the L channel may sometimes be received at the R channel if an overflow occurs, for example, under the following condition: the control register (SSICR) has a 32-bit setting for both data word length (DWL2 to DWL0) and system word length (SWL2 to SWL).

If an overflow is confirmed with the overflow error interrupt or overflow error status flag (SSISR. OIRQ), write 0 to the EN and DMEN bits in SSICR to disable DMA in the SSI module, thus stopping the operation (In this case, the controller setting should also be stopped). After this, write 0 to the OIRQ bit in SSISR to clear the overflow status, set DMA again and restart the transfer.

41.4.2 Limitations on Combinations of Modes Related to Common Pins to SSI7 and SSI8

Table 41.6 shows the available combinations of operating modes for SSI7 and SSI8. Operating modes can be set with the control registers for the SSI7 and SSI8 (SSICR).

Table 41.6 Usable Operating Mode Combinations for SSI7 and SSI8

| No. | Operating Mode | | Operating Mode of External Device | | Function |
|-----|----------------|--------|-----------------------------------|-----------|--|
| | SSI7 | SSI8 | SSI7 Side | SSI8 Side | |
| 1 | Slave | Slave | Master | Slave | SSI7 and SSI8 operate synchronously with SSI_WS78 and SSI_SCK78 input. |
| | | | Slave | Master | |
| | | | Slave | Slave | |
| 2 | Master | Slave | Slave | Slave | SSI8 and an external device operate synchronously with WS and SCK of SSI7. |
| 3 | Slave | Master | — | — | Setting prohibited |
| 4 | Master | Master | — | — | Setting prohibited |

There are the following limitations on the operating mode combinations shown in Table 41.6.

For No. 1, SSI_WS78 and SSI_SCK78 should be input before either of SSI7 or SSI8 starts data transfer.

For No. 2, the SSI7 as the master should start transfer first thus allowing SSI_WS78 and SSI_SCK78 to be output, and then the SSI8 as the slave should be used. If the master-side transfer is stopped with the CONT bit in SSIWSR being 0, the slave-side SSI8 should be stopped because SSI_WS78 is not output.

41.4.3 Limitations on Slave Mode Operation

When this LSI is used in slave mode, in ending a data transfer process, data transfer on this LSI should be stopped (SSICR.EN = 0) before the input word selection signal (SSI_WS) is stopped.

In slave mode, data transfer is stopped by clearing the EN bit in SSICR (setting to stop transfer) and detecting the falling edge of the word selection signal (SSI_WS). If the input word selection signal is stopped first, the falling edge of the word selection signal cannot be detected and data transfer cannot be ended successfully.

In case of transmission at slave mode, it is necessary that WS signal is stable. Data transmission at slave mode should be started after more than two frames period since WS signal is entered and is stable.

In case of multiple SSI modules and case that one SSI works at master mode and the others do at slave mode, data transfer should be started after more than two frames period since registers of SSI at master mode are set.

41.4.4 Limitations on Changes to Settings

The SSI_SCK and SSI_WS signals are not guaranteed immediately after changes of the WS mode bit in the WS Mode Register (SSIWSR) and configuration bits in the control register (SSICR). Settings must not be changed dynamically if this would affect any connected device.

41.4.5 Stopping or Resuming Transmission

Follow the procedure below to stop or resume transmission.

Stopping transmission:

1. Set the DMEN bit in SSICR to 0 for stopping transmission.
2. Wait for SSISR.DIRQ = 1 by using polling or interrupts.
3. Set the EN bit in SSICR to 0 to stop transmission.
4. Check that SSISR.IDST is 1.

Resuming transmission:

Set DMEN and EN bits in SSICR to 1 for resuming transmission (The DMEN bit should be set to 1 at the same time of or before setting the EN bit).

41.4.6 Stable period of SCK and WS

The period during which SCK signal and WS signal stabilizes is shown below. It is assumed that the clock supplied from ADG is stable.

SCK signal (when SSI is used in the master mode)

1. Set the SSICR (exclude EN bit) register.
2. Wait count end to setting value of SSICR.CKDV (counted by CLK_FS).
3. SCK signal is stable.

WS signal (when SSI is used in the master mode)

1. Set the SSIWSR and SSICR (exclude EN bit) register.
2. Wait 2 frames.
3. WS signal is stable.

41.4.7 About CLK_FS input clock

For details on generation and selection of master clock signal, see section 42. Audio Clock Generator (ADG).

42. Audio Clock Generator (ADG)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

42.1 Overview

The audio clock generator (ADG) selects and supplies the necessary clock for the SSIU, EAVB-IF, SCU module. It also divides the frequency of the selected clock and sends it outside the chip.

42.1.1 Features

- Selects the clock signal from the AUDIO_CLKA, AUDIO_CLKB, AUDIO_CLKC pin or the internal clock and supplies it to the SSIU, EAVB-IF, SCU module.
- The frequency of the clock signals from the AUDIO_CLKA, AUDIO_CLKB, and AUDIO_CLKC pins and the internal clock can be divided before use.
- The divided clock can be output through the AUDIO_CLKOUT, AUDIO_CLKOUT1, AUDIO_CLKOUT2 and AUDIO_CLKOUT3 pins.
- Generate the Media Clock Recovery for EAVB-IF module.

Note: The presence or absence of pins depends on the product. For details, see Table 42.1.

42.1.2 Block Diagram

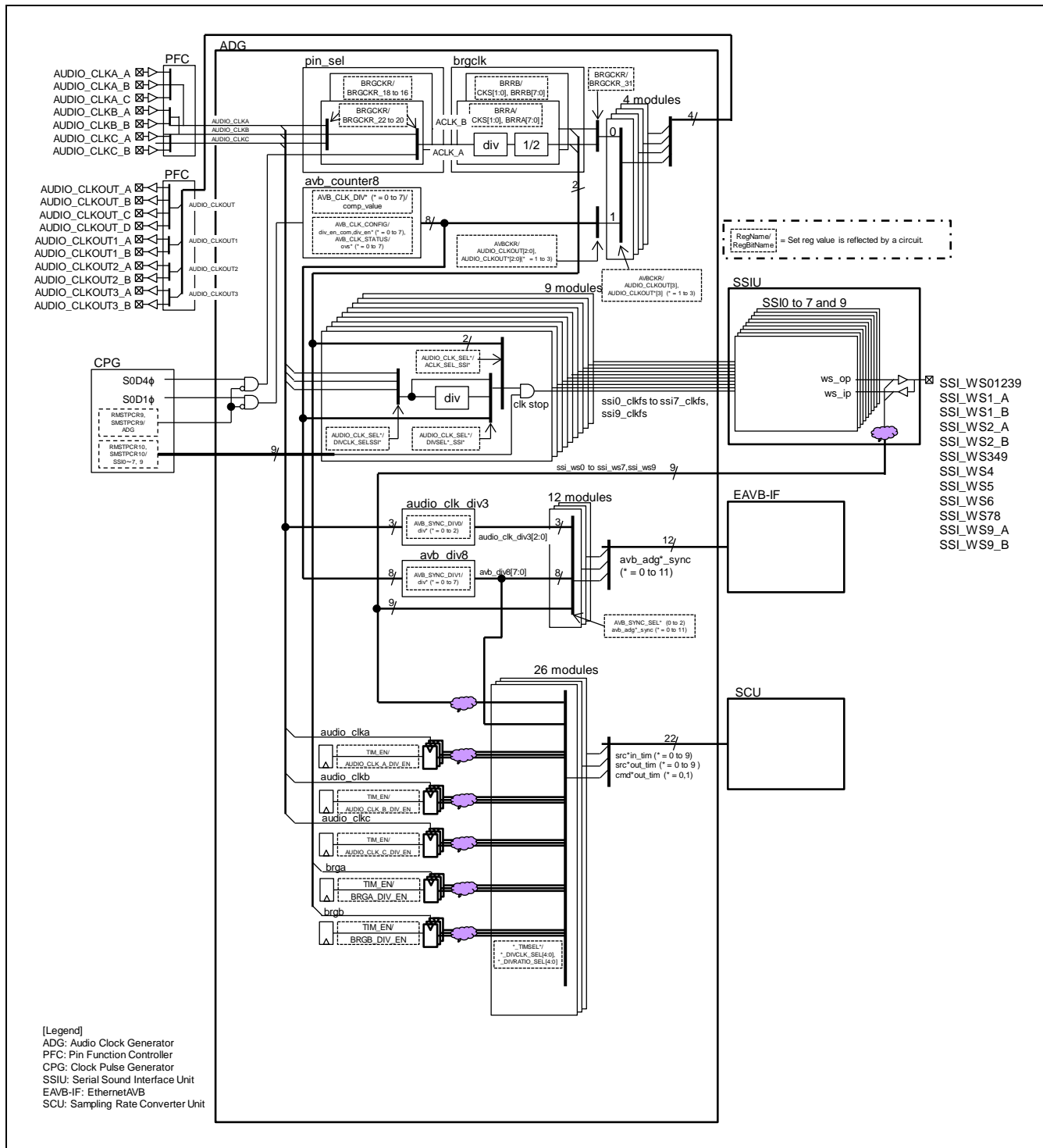


Figure 42.1 Block Diagram [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

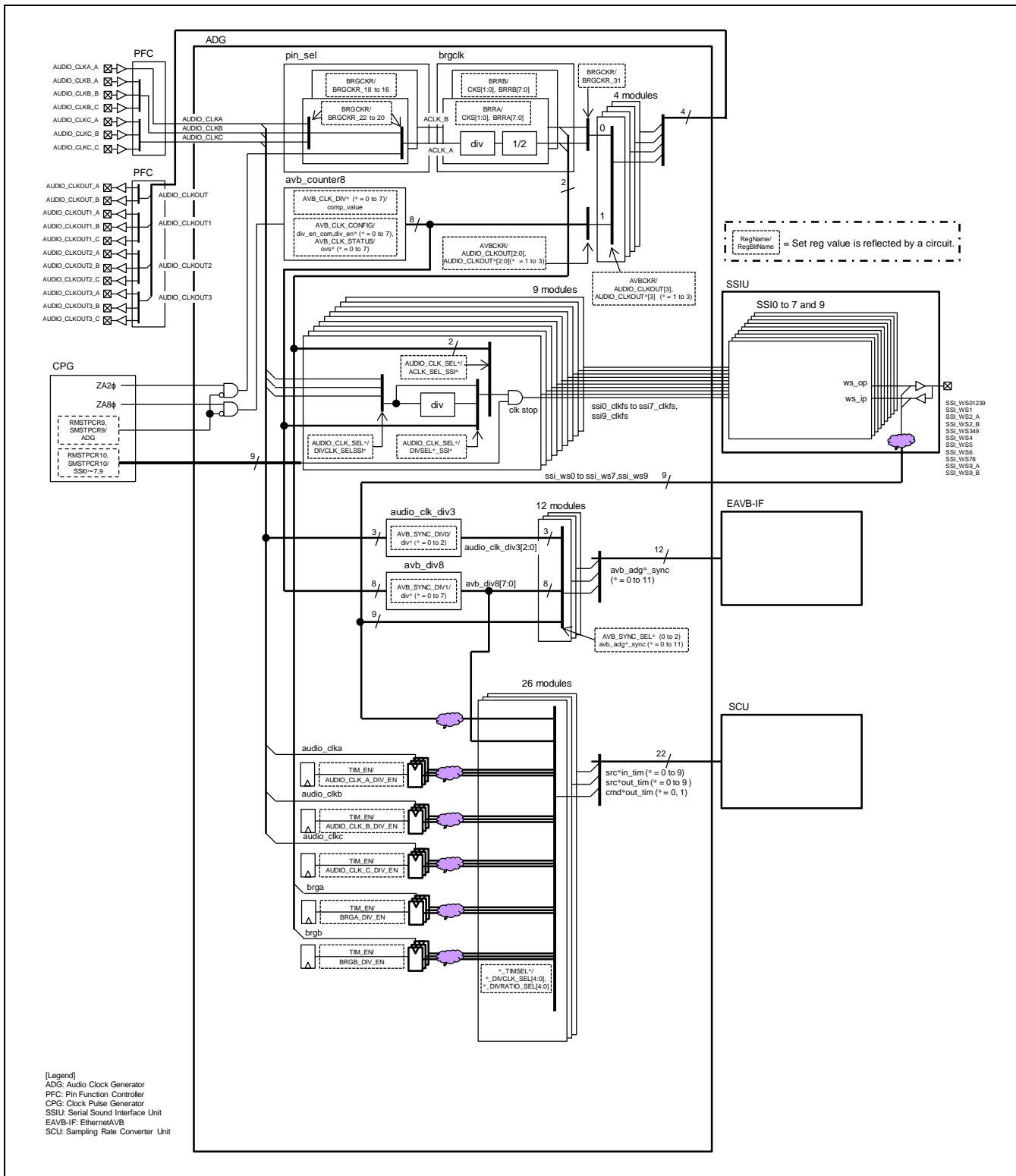


Figure 42.2 Block Diagram [RZ/G2E]

42.1.3 External Pins

Table 42.1 Pin Configuration

| Pin Name | I/O | Function | Second Generation RZ/G Series Products | | | |
|-----------------|--------|------------------|---|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| AUDIO_CLKA_A | Input | AUDIO CLOCK A | √ | √ | √ | — |
| AUDIO_CLKA_B | Input | AUDIO CLOCK A | √ | √ | √ | — |
| AUDIO_CLKA_C | Input | AUDIO CLOCK A | √ | √ | √ | — |
| AUDIO_CLKA | Input | AUDIO CLOCK A | — | — | — | √ |
| AUDIO_CLKB_A | Input | AUDIO CLOCK B | √ | √ | √ | √ |
| AUDIO_CLKB_B | Input | AUDIO CLOCK B | √ | √ | √ | √ |
| AUDIO_CLKB_C | Input | AUDIO CLOCK B | — | — | — | √ |
| AUDIO_CLKC_A | Input | AUDIO CLOCK C | √ | √ | √ | √ |
| AUDIO_CLKC_B | Input | AUDIO CLOCK C | √ | √ | √ | √ |
| AUDIO_CLKC_C | Input | AUDIO CLOCK C | — | — | — | √ |
| AUDIO_CLKOUT_A | Output | AUDIO CLOCK OUT | √ | √ | √ | √ |
| AUDIO_CLKOUT_B | Output | AUDIO CLOCK OUT | √ | √ | √ | √ |
| AUDIO_CLKOUT_C | Output | AUDIO CLOCK OUT | √ | √ | √ | — |
| AUDIO_CLKOUT_D | Output | AUDIO CLOCK OUT | √ | √ | √ | — |
| AUDIO_CLKOUT1_A | Output | AUDIO CLOCK OUT1 | √ | √ | √ | √ |
| AUDIO_CLKOUT1_B | Output | AUDIO CLOCK OUT1 | √ | √ | √ | √ |
| AUDIO_CLKOUT1_C | Output | AUDIO CLOCK OUT1 | — | — | — | √ |
| AUDIO_CLKOUT2_A | Output | AUDIO CLOCK OUT2 | √ | √ | √ | √ |
| AUDIO_CLKOUT2_B | Output | AUDIO CLOCK OUT2 | √ | √ | √ | √ |
| AUDIO_CLKOUT2_C | Output | AUDIO CLOCK OUT2 | — | — | — | √ |
| AUDIO_CLKOUT3_A | Output | AUDIO CLOCK OUT3 | √ | √ | √ | √ |
| AUDIO_CLKOUT3_B | Output | AUDIO CLOCK OUT3 | √ | √ | √ | √ |
| AUDIO_CLKOUT3_C | Output | AUDIO CLOCK OUT3 | — | — | — | √ |

42.1.4 Register Configuration

Table 42.2 shows the register configuration. The base address is H'EC5A_0000. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined.

Table 42.2 Register Configuration

| Name | Abbreviation | R/W | Address | Initial Value | Access size | Second Generation RZ/G Series Products | | | |
|-------------------------------------|----------------|-----|---------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| BRGA Baud Rate Setting Register | BARRA | R/W | H'000 | H'0000_00FF | 32 | √ | √ | √ | √ |
| BRGB Baud Rate Setting Register | BRRB | R/W | H'004 | H'0000_00FF | 32 | √ | √ | √ | √ |
| BRG Clock Select Register | BRGCKR | R/W | H'008 | H'2300_0000 | 32 | √ | √ | √ | √ |
| Audio Clock Select Register 0 | AUDIO_CLK_SEL0 | R/W | H'00C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Audio Clock Select Register 1 | AUDIO_CLK_SEL1 | R/W | H'010 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Audio Clock Select Register 2 | AUDIO_CLK_SEL2 | R/W | H'014 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Timing Signal Enable Register | TIM_EN | R/W | H'030 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC Input Timing Select Register 0 | SRCIN_TIMSEL0 | R/W | H'034 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC Input Timing Select Register 1 | SRCIN_TIMSEL1 | R/W | H'038 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC Input Timing Select Register 2 | SRCIN_TIMSEL2 | R/W | H'03C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC Input Timing Select Register 3 | SRCIN_TIMSEL3 | R/W | H'040 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC Input Timing Select Register 4 | SRCIN_TIMSEL4 | R/W | H'044 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC Output Timing Select Register 0 | SRCOUT_TIMSEL0 | R/W | H'048 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC Output Timing Select Register 1 | SRCOUT_TIMSEL1 | R/W | H'04C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC Output Timing Select Register 2 | SRCOUT_TIMSEL2 | R/W | H'050 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC Output Timing Select Register 3 | SRCOUT_TIMSEL3 | R/W | H'054 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC Output Timing Select Register 4 | SRCOUT_TIMSEL4 | R/W | H'058 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CMD Output Timing Select Register | CMDOUT_TIMSEL | R/W | H'05C | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB Clock Select Register | AVBCKR | R/W | H'100 | H'0000_0000 | 32 | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Name | Abbreviation | R/W | Address | Initial Value | Access size | Second Generation RZ/G Series Products | | | |
|---|----------------|-----|---------|---------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| AVB sync select Register 0 | AVB_SYNC_SEL0 | R/W | H'104 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB sync select Register 1 | AVB_SYNC_SEL1 | R/W | H'10C | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB sync select Register 2 | AVB_SYNC_SEL2 | R/W | H'110 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB sync divide Register 0 | AVB_SYNC_DIV0 | R/W | H'114 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB sync divide Register 1 | AVB_SYNC_DIV1 | R/W | H'118 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB clock divider comparison value register 0 | AVB_CLK_DIV0 | R/W | H'11C | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB clock divider comparison value register 1 | AVB_CLK_DIV1 | R/W | H'120 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB clock divider comparison value register 2 | AVB_CLK_DIV2 | R/W | H'124 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB clock divider comparison value register 3 | AVB_CLK_DIV3 | R/W | H'128 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB clock divider comparison value register 4 | AVB_CLK_DIV4 | R/W | H'12C | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB clock divider comparison value register 5 | AVB_CLK_DIV5 | R/W | H'130 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB clock divider comparison value register 6 | AVB_CLK_DIV6 | R/W | H'134 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB clock divider comparison value register 7 | AVB_CLK_DIV7 | R/W | H'138 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB clock divider configuration register | AVB_CLK_CONFIG | R/W | H'13C | H'0000_0000 | 32 | √ | √ | √ | √ |

42.1.5 Connected Module

Table 42.3 Connected modules

| Module name | Connected module name | Function of connected module |
|-----------------------|-----------------------|------------------------------|
| Audio Clock Generator | APMU | Access the Register |
| | PFC | Select External pins |
| | CPG | Output Clocks |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |
| | SSIU | Serial Sound Interface Unit |
| | SSI | Serial Sound Interface |
| | SCU | Sampling Rate Converter Unit |
| | EAVB-IF | EthernetAVB Interface |

42.2 Register Description

[Legend for Register Description]

Initial value: Register value after a reset.

—: Undefined value

R/W: Bit or field is readable and writable. The written value can be read.

R/WC1: The bit or field is readable and writable. Writing 1 to the bit initializes the bit.

All registers are accessed in longword units.

42.2.1 BRGA Baud Rate Setting Register (BRR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: BRR is a 32-bit readable/writable register that specifies the baud rate for ACLK_A (see Table 42.4).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----------|----------|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | CKS[1:0] | BRR[7:0] | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 10 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 9, 8 | CKS[1:0] | B'00 | R/W | These bits specify the clock source for the on-chip baud rate generator. B'00: ACLK_A B'01: ACLK_A/4 B'10: ACLK_A/16 B'11: ACLK_A/64 |
| 7 to 0 | BRR[7:0] | H'FF | R/W | These bits specify the division ratio. For details, see Table 42.4. |

Table 42.4 Division Ratio for BRGA

| BRGA Operating Clock (CKS[1], CKS[0]) | Division Ratio BRR (N = 0 to 255) | Calculating Formula |
|---------------------------------------|-----------------------------------|---------------------|
| ACLK_A | 1/2, 1/4, 1/6, ..., 1/512 | 1 / (2(N + 1)) |
| ACLK_A/4 | 1/8, 1/16, 1/24, ..., 1/2048 | 1 / (8(N + 1)) |
| ACLK_A/16 | 1/32, 1/64, 1/96, ..., 1/8192 | 1 / (32(N + 1)) |
| ACLK_A/64 | 1/128, 1/256, 1/384, ..., 1/32768 | 1 / (128(N + 1)) |

Note: Use the output of BRGA in less than 25 MHz.

42.2.2 BRGB Baud Rate Setting Register (BRRB)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: BRRB is a 32-bit readable/writable register that specifies the baud rate for ACLK_B (see Table 42.5).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----------|-----|-----------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | CKS[1:0] | | BRRB[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 10 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 9, 8 | CKS[1:0] | B'00 | R/W | These bits specify the clock source for the on-chip baud rate generator. B'00: ACLK_B B'01: ACLK_B/4 B'10: ACLK_B/16 B'11: ACLK_B/64 |
| 7 to 0 | BRRB[7:0] | H'FF | R/W | These bits specify the division ratio. For details, see Table 42.5. |

Table 42.5 Division Ratio for BRGB

| BRGB Operating Clock (CKS[1], CKS[0]) | Division Ratio BRRB (N = 0 to 255) | Calculating Formula |
|---------------------------------------|------------------------------------|---------------------|
| ACLK_B | 1/2, 1/4, 1/6, ..., 1/512 | 1 / (2(N + 1)) |
| ACLK_B/4 | 1/8, 1/16, 1/24, ..., 1/2048 | 1 / (8(N + 1)) |
| ACLK_B/16 | 1/32, 1/64, 1/96, ..., 1/8192 | 1 / (32(N + 1)) |
| ACLK_B/64 | 1/128, 1/256, 1/384, ..., 1/32768 | 1 / (128(N + 1)) |

Note: Use the output of BRGB in less than 25 MHz.

42.2.3 BRG Clock Select Register (BRGCKR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: BRGCKR selects the clocks input to and output from the ADG.

| | | | | | | | | | | | | | | | | |
|----------------|---------------|----|----|----|----|----|----|----|----|---------------|---------------|---------------|----|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | BRGCK R_31 | — | — | — | — | — | — | — | — | BRGCK R_22 | BRGCK R_21 | BRGCK R_20 | — | BRGCK R_18 | BRGCK R_17 | BRGCK R_16 |
| Initial value: | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 | BRGCKR_31 | B'0 | R/W | This bit selects the clock signal output to the AUDIO_CLKOUT, AUDIO_CLKOUT1, AUDIO_CLKOUT2, AUDIO_CLKOUT3 or external pin. 0: BRGA output clock 1: BRGB output clock |
| 30 to 23 | — | H'46 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 22 | BRGCKR_22 | B'0 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| 21 | BRGCKR_21 | B'0 | | These bits select the clock signal input to the BRGA. |
| 20 | BRGCKR_20 | B'0 | | B'000: AUDIO_CLKA B'001: AUDIO_CLKB B'01x: S0D4 ϕ (200 MHz) B'100: AUDIO_CLKC B'101: Fixed at 0 B'11x: Fixed at 0 x = Don't care |
| | | | | [RZ/G2E] These bits select the clock signal input to the BRGA. B'000: AUDIO_CLKA B'001: AUDIO_CLKB B'01x: ZA2 ϕ (200 MHz) B'100: AUDIO_CLKC B'101: Fixed at 0 B'11x: Fixed at 0 x = Don't care |
| 19 | — | B'0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 18 | BRGCKR_18 | B'0 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| 17 | BRGCKR_17 | B'0 | | These bits select the clock signal input to the BRGB. |
| 16 | BRGCKR_16 | B'0 | | B'000: AUDIO_CLKA B'001: AUDIO_CLKB B'01x: S0D4 ϕ (200 MHz) B'100: AUDIO_CLKC B'101: Fixed at 0 B'11x: Fixed at 0 x = Don't care [RZ/G2E] These bits select the clock signal input to the BRGB. B'000: AUDIO_CLKA B'001: AUDIO_CLKB B'01x: ZA2 ϕ (200 MHz) B'100: AUDIO_CLKC B'101: Fixed at 0 B'11x: Fixed at 0 x = Don't care |
| 15 to 0 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

42.2.4 Audio Clock Select Register 0 (AUDIO_CLK_SEL0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AUDIO_CLK_SEL0 selects the clocks for the SSIU (for SSI0 to SSI3).

| | | | | | | | | | | | | | | | | |
|----------------|----------------------|-----|------------------------|-----|--------------|-----|--------------------------|-----|----------------------|-----|------------------------|-----|--------------|-----|--------------------------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | DIVSEL_SSI3 [1:0] | | ACLK_SEL_SSI3 [1:0] | | DIVSEL2_SSI3 | | DIVCLK_SEL_SSI3 [2:0] | | DIVSEL_SSI2 [1:0] | | ACLK_SEL_SSI2 [1:0] | | DIVSEL2_SSI2 | | DIVCLK_SEL_SSI2 [2:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DIVSEL_SSI1 [1:0] | | ACLK_SEL_SSI1 [1:0] | | DIVSEL2_SSI1 | | DIVCLK_SEL_SSI1 [2:0] | | DIVSEL_SSI0 [1:0] | | ACLK_SEL_SSI0 [1:0] | | DIVSEL2_SSI0 | | DIVCLK_SEL_SSI0 [2:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|--------------------|---------------|-----|---|
| 31, 30 | DIVSEL_SSI3[1:0] | B'00 | R/W | SSI3 Frequency Divider Select |
| 27 | DIVSEL2_SSI3 | B'0 | R/W | [27], [31:30]: ACLK_SEL_SSI3[1:0] = B'00 B'000: Not divided B'001: Divided by 2 B'010: Divided by 4 B'011: Divided by 8 B'100: Divided by 16 B'101: Divided by 32 B'110, B'111: Fixed at 0 ACLK_SEL_SSI3[1:0] = B'11 B'000: avb_counter8[0] B'001: avb_counter8[1] B'010: avb_counter8[2] B'011: avb_counter8[3] B'100: avb_counter8[4] B'101: avb_counter8[5] B'110: avb_counter8[6] B'111: avb_counter8[7] |
| 29, 28 | ACLK_SEL_SSI3[1:0] | B'00 | R/W | SSI3 Clock Select B'00: DIVCLK B'01: BRGA output clock B'10: BRGB output clock B'11: avb_counter8 output clock |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------------|---------------|-----|---|
| 26 to 24 | DIVCLK_SEL_SSI3 [2:0] | B'000 | R/W | SSI3 Clock Select B'000: Fixed at 0 B'001: AUDIO_CLKA B'010: AUDIO_CLKB B'011: AUDIO_CLKC B'100: Reserved Others: Setting prohibited |
| 23, 22 | DIVSEL_SSI2[1:0] | B'00 | R/W | SSI2 Frequency Divider Select |
| 19 | DIVSEL2_SSI2 | B'0 | R/W | [19], [23:22]: ACLK_SEL_SSI2[1:0] = B'00 B'000: Not divided B'001: Divided by 2 B'010: Divided by 4 B'011: Divided by 8 B'100: Divided by 16 B'101: Divided by 32 B'110, B'111: Fixed at 0 ACLK_SEL_SSI2[1:0] = B'11 B'000: avb_counter8[0] B'001: avb_counter8[1] B'010: avb_counter8[2] B'011: avb_counter8[3] B'100: avb_counter8[4] B'101: avb_counter8[5] B'110: avb_counter8[6] B'111: avb_counter8[7] |
| 21, 20 | ACLK_SEL_SSI2[1:0] | B'00 | R/W | SSI2 Clock Select B'00: DIVCLK B'01: BRGA output clock B'10: BRGB output clock B'11: avb_counter8 output clock |
| 18 to 16 | DIVCLK_SEL_SSI2 [2:0] | B'000 | R/W | SSI2 Clock Select B'000: Fixed at 0 B'001: AUDIO_CLKA B'010: AUDIO_CLKB B'011: AUDIO_CLKC B'100: Reserved Others: Setting Prohibited |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------------|---------------|-----|---|
| 15, 14 | DIVSEL_SSI1[1:0] | B'00 | R/W | SSI1 Frequency Divider Select |
| 11 | DIVSEL2_SSI1 | B'0 | R/W | [11], [15:14]: ACLK_SEL_SSI1[1:0] = B'00 B'000: Not divided B'001: Divided by 2 B'010: Divided by 4 B'011: Divided by 8 B'100: Divided by 16 B'101: Divided by 32 B'110, B'111: Fixed at 0 ACLK_SEL_SSI1[1:0] = B'11 B'000: avb_counter8[0] B'001: avb_counter8[1] B'010: avb_counter8[2] B'011: avb_counter8[3] B'100: avb_counter8[4] B'101: avb_counter8[5] B'110: avb_counter8[6] B'111: avb_counter8[7] |
| 13, 12 | ACLK_SEL_SSI1[1:0] | B'00 | R/W | SSI1 Clock Select B'00: DIVCLK B'01: BRGA output clock B'10: BRGB output clock B'11: avb_counter8 output clock |
| 10 to 8 | DIVCLK_SEL_SSI1 [2:0] | B'000 | R/W | SSI1 Clock Select B'000: Fixed at 0 B'001: AUDIO_CLKA B'010: AUDIO_CLKB B'011: AUDIO_CLKC B'100: Reserved Others: Setting Prohibited |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------------------|---------------|-----|--|
| 7, 6 | DIVSEL_SSI0[1:0] | B'00 | R/W | SSI0 Frequency Divider Select |
| 3 | DIVSEL2_SSI0 | B'0 | R/W | [3], [7:6]: ACLK_SEL_SSI0[1:0] = B'00 B'000: Not divided B'001: Divided by 2 B'010: Divided by 4 B'011: Divided by 8 B'100: Divided by 16 B'101: Divided by 32 B'110, B'111: Fixed at 0 ACLK_SEL_SSI0[1:0] = B'11 B'000: avb_counter8[0] B'001: avb_counter8[1] B'010: avb_counter8[2] B'011: avb_counter8[3] B'100: avb_counter8[4] B'101: avb_counter8[5] B'110: avb_counter8[6] B'111: avb_counter8[7] |
| 5, 4 | ACLK_SEL_SSI0[1:0] | B'00 | R/W | SSI0 Clock Select B'00: DIVCLK B'01: BRGA output clock B'10: BRGB output clock B'11: avb_counter8 output clock |
| 2 to 0 | DIVCLK_SEL_SSI0 [2:0] | B'000 | R/W | SSI0 Clock Select B'000: Fixed at 0 B'001: AUDIO_CLKA B'010: AUDIO_CLKB B'011: AUDIO_CLKC B'100: Reserved Others: Setting Prohibited |

42.2.5 Audio Clock Select Register 1 (AUDIO_CLK_SEL1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AUDIO_CLK_SEL1 selects the clocks for the SSIU (for SSI4 to SSI7).

| | | | | | | | | | | | | | | | | |
|----------------|-------------------|-----|---------------------|-----|--------------|-----------------------|-----|-------------------|-----|---------------------|-----|--------------|-----------------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | DIVSEL_SSI7 [1:0] | | ACLK_SEL_SSI7 [1:0] | | DIVSEL2_SSI7 | DIVCLK_SEL_SSI7 [2:0] | | DIVSEL_SSI6 [1:0] | | ACLK_SEL_SSI6 [1:0] | | DIVSEL2_SSI6 | DIVCLK_SEL_SSI6 [1:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DIVSEL_SSI5 [1:0] | | ACLK_SEL_SSI5 [1:0] | | DIVSEL2_SSI5 | DIVCLK_SEL_SSI5 [2:0] | | DIVSEL_SSI4 [1:0] | | ACLK_SEL_SSI4 [1:0] | | DIVSEL2_SSI4 | DIVCLK_SEL_SSI4 [1:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------------|---------------|-----|---|
| 31, 30 | DIVSEL_SSI7 [1:0] | B'00 | R/W | SSI7 Frequency Divider Select |
| 27 | DIVSEL2_SSI7 | B'0 | R/W | [27], [31:30]: ACLK_SEL_SSI7 [1:0] = B'00 B'000: Not divided B'001: Divided by 2 B'010: Divided by 4 B'011: Divided by 8 B'100: Divided by 16 B'101: Divided by 32 B'110, B'111: Fixed at 0 ACLK_SEL_SSI7 [1:0] = B'11 B'000: avb_counter8 [0] B'001: avb_counter8 [1] B'010: avb_counter8 [2] B'011: avb_counter8 [3] B'100: avb_counter8 [4] B'101: avb_counter8 [5] B'110: avb_counter8 [6] B'111: avb_counter8 [7] |
| 29, 28 | ACLK_SEL_SSI7 [1:0] | B'00 | R/W | SSI7 Clock Select B'00: DIVCLK B'01: BRGA output clock B'10: BRGB output clock B'11: avb_counter8 output clock |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------------|---------------|-----|--|
| 26 to 24 | DIVCLK_SEL_SSI7 [2:0] | B'000 | R/W | SSI7 Clock Select B'000: Fixed at 0 B'001: AUDIO_CLKA B'010: AUDIO_CLKB B'011: AUDIO_CLKC B'100: Reserved Others: Setting Prohibited |
| 23, 22 | DIVSEL_SSI6[1:0] | B'00 | R/W | SSI6 Frequency Divider Select [19], [23:22]: ACLK_SEL_SSI6[1:0] = B'00 B'000: Not divided B'001: Divided by 2 B'010: Divided by 4 B'011: Divided by 8 B'100: Divided by 16 B'101: Divided by 32 B'110, B'111: Fixed at 0 ACLK_SEL_SSI6[1:0] = B'11 B'000: avb_counter8[0] B'001: avb_counter8[1] B'010: avb_counter8[2] B'011: avb_counter8[3] B'100: avb_counter8[4] B'101: avb_counter8[5] B'110: avb_counter8[6] B'111: avb_counter8[7] |
| 19 | DIVSEL2_SSI6 | B'0 | R/W | |
| 21, 20 | ACLK_SEL_SSI6[1:0] | B'00 | R/W | SSI6 Clock Select B'00: DIVCLK B'01: BRGA output clock B'10: BRGB output clock B'11: avb_counter8 output clock |
| 18 to 16 | DIVCLK_SEL_SSI6 [2:0] | B'000 | R/W | SSI6 Clock Select B'000: Fixed at 0 B'001: AUDIO_CLKA B'010: AUDIO_CLKB B'011: AUDIO_CLKC B'100: Reserved Others: Setting Prohibited |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------------|---------------|-----|---|
| 15, 14 | DIVSEL_SSI5[1:0] | B'00 | R/W | SSI5 Frequency Divider Select |
| 11 | DIVSEL2_SSI5 | B'0 | R/W | [11], [15:14]: ACLK_SEL_SSI5[1:0] = B'00 B'000: Not divided B'001: Divided by 2 B'010: Divided by 4 B'011: Divided by 8 B'100: Divided by 16 B'101: Divided by 32 B'110, B'111: Fixed at 0 ACLK_SEL_SSI5[1:0] = B'11 B'000: avb_counter8[0] B'001: avb_counter8[1] B'010: avb_counter8[2] B'011: avb_counter8[3] B'100: avb_counter8[4] B'101: avb_counter8[5] B'110: avb_counter8[6] B'111: avb_counter8[7] |
| 13, 12 | ACLK_SEL_SSI5[1:0] | B'00 | R/W | SSI5 Clock Select B'00: DIVCLK B'01: BRGA output clock B'10: BRGB output clock B'11: avb_counter8 output clock |
| 10 to 8 | DIVCLK_SEL_SSI5 [2:0] | B'000 | R/W | SSI5 Clock Select B'000: Fixed at 0 B'001: AUDIO_CLKA B'010: AUDIO_CLKB B'011: AUDIO_CLKC B'100: Reserved Others: Setting Prohibited |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------------------|---------------|-----|--|
| 7, 6 | DIVSEL_SSI4[1:0] | B'00 | R/W | SSI4 Frequency Divider Select |
| 3 | DIVSEL2_SSI4 | B'0 | R/W | [3], [7:6]: ACLK_SEL_SSI4[1:0] = B'00 B'000: Not divided B'001: Divided by 2 B'010: Divided by 4 B'011: Divided by 8 B'100: Divided by 16 B'101: Divided by 32 B'110, B'111: Fixed at 0 ACLK_SEL_SSI4[1:0] = B'11 B'000: avb_counter8[0] B'001: avb_counter8[1] B'010: avb_counter8[2] B'011: avb_counter8[3] B'100: avb_counter8[4] B'101: avb_counter8[5] B'110: avb_counter8[6] B'111: avb_counter8[7] |
| 5, 4 | ACLK_SEL_SSI4[1:0] | B'00 | R/W | SSI4 Clock Select B'00: DIVCLK B'01: BRGA output clock B'10: BRGB output clock B'11: avb_counter8 output clock |
| 2 to 0 | DIVCLK_SEL_SSI4 [2:0] | B'000 | R/W | SSI4 Clock Select B'000: Fixed at 0 B'001: AUDIO_CLKA B'010: AUDIO_CLKB B'011: AUDIO_CLKC B'100: Reserved Others: Setting Prohibited |

42.2.6 Audio Clock Select Register 2 (AUDIO_CLK_SEL2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AUDIO_CLK_SEL2 selects the clocks for the SSIU (for SSI9).

| | | | | | | | | | | | | | | | | |
|----------------|-------------------|-----|---------------------|-----|--------------|-----|-----------------------|-----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DIVSEL_SSI9 [1:0] | | ACLK_SEL_SSI9 [1:0] | | DIVSEL2_SSI9 | | DIVCLK_SEL_SSI9 [2:0] | | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 15, 14 | DIVSEL_SSI9 [1:0] | B'00 | R/W | SSI9 Frequency Divider Select |
| 11 | DIVSEL2_SSI9 | B'0 | R/W | [11], [15:14] ACLK_SEL_SSI9 [1:0] = B'00 B'000: Not divided B'001: Divided by 2 B'010: Divided by 4 B'011: Divided by 8 B'100: Divided by 16 B'101: Divided by 32 B'110, B'111: Fixed at 0 ACLK_SEL_SSI9 [1:0] = B'11 B'000: avb_counter8[0] B'001: avb_counter8[1] B'010: avb_counter8[2] B'011: avb_counter8[3] B'100: avb_counter8[4] B'101: avb_counter8[5] B'110: avb_counter8[6] B'111: avb_counter8[7] |
| 13, 12 | ACLK_SEL_SSI9 [1:0] | B'00 | R/W | SSI9 Clock Select B'00: DIVCLK B'01: BRGA output clock B'10: BRGB output clock B'11: avb_counter8 output clock |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------------|---------------|-----|--|
| 10 to 8 | DIVCLK_SEL_SSI9[2:0] | B'000 | R/W | SSI9 Clock Select B'000: Fixed at 0 B'001: AUDIO_CLKA B'010: AUDIO_CLKB B'011: AUDIO_CLKC B'100: Reserved Others: Setting Prohibited |
| 7 to 0 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

42.2.7 Timing Signal Enable Register (TIM_EN)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: TIM_EN is enabling of the Timing Signal supplied to SCU.

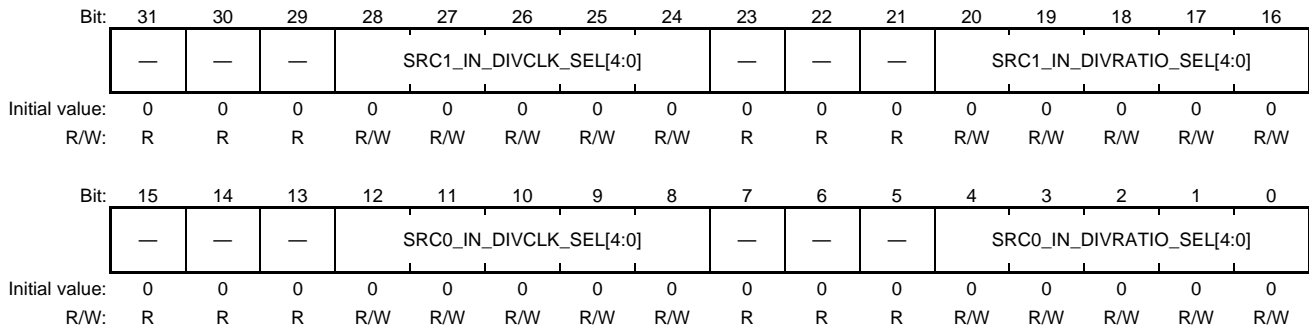
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-------------|-------------|-------------------|-------------------|-------------------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | BRGB_TIM_EN | BRGA_TIM_EN | AUDIO_CLKC_TIM_EN | AUDIO_CLKB_TIM_EN | AUDIO_CLKA_TIM_EN | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W* | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------------|---------------|-----|---|
| 31 to 6 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |
| 5 | BRGB_TIM_EN | B'0 | R/W | This bit is enabling of the Timing Signal generated from the BRGB frequency divider. 0: Disables timing signal. 1: Enables timing signal. |
| 4 | BRGA_TIM_EN | B'0 | R/W | This bit is enabling of the Timing Signal generated from the BRGA frequency divider. 0: Disables timing signal. 1: Enables timing signal. |
| 3 | AUDIO_CLKC_TIM_EN | B'0 | R/W | This bit is enabling of the Timing Signal generated from the AUDIO_CLKC frequency divider. 0: Disables timing signal. 1: Enables timing signal. |
| 2 | AUDIO_CLKB_TIM_EN | B'0 | R/W | This bit is enabling of the Timing Signal generated from the AUDIO_CLKB frequency divider. 0: Disables timing signal. 1: Enables timing signal. |
| 1 | AUDIO_CLKA_TIM_EN | B'0 | R/W | This bit is enabling of the Timing Signal generated from the AUDIO_CLKA frequency divider. 0: Disables timing signal. 1: Enables timing signal. |
| 0 | — | B'0 | R/W | Reserved |

42.2.8 SRC Input Timing Select Register 0 (SRCIN_TIMSEL0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCIN_TIMSEL0 selects the input timing signals for SRC0 and SRC1.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------------|---------------|-----|--|
| 31 to 29 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 28 to 24 | SRC1_IN_DIVCLK_SEL [4:0] | All 0 | R/W | SRC1 Input Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------------|---------------|-----|---|
| 23 to 21 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 20 to 16 | SRC1_IN_DIVRATIO_SEL [4:0] | All 0 | R/W | SRC1 Input Timing Signal Frequency Division Ratio Select These bits select the frequency division ratio for the clock signal selected using SRC1_IN_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided. B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited |
| 15 to 13 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------------------|---------------|-----|--|
| 12 to 8 | SRC0_IN_DIVCLK_SEL [4:0] | All 0 | R/W | SRC0 Input Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |
| 7 to 5 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------------------|---------------|-----|--|
| 4 to 0 | SRC0_IN_DIVRATIO_SEL [4:0] | All 0 | R/W | <p>SRC0 Input Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC0_IN_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |

42.2.9 SRC Input Timing Select Register 1 (SRCIN_TIMSEL1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCIN_TIMSEL1 selects the input timing signals for SRC2 and SRC3.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-------------------------|-----|-----|-----|-----|----|----|---------------------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | SRC3_IN_DIVCLK_SEL[4:0] | | | | — | — | — | SRC3_IN_DIVRATIO_SEL[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | SRC2_IN_DIVCLK_SEL[4:0] | | | | — | — | — | SRC2_IN_DIVRATIO_SEL[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------------|---------------|-----|--|
| 31 to 29 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 28 to 24 | SRC3_IN_DIVCLK_SEL [4:0] | All 0 | R/W | SRC3 Input Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------------|---------------|-----|---|
| 23 to 21 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 20 to 16 | SRC3_IN_DIVRATIO_SEL[4:0] | All 0 | R/W | SRC3 Input Timing Signal Frequency Division Ratio Select These bits select the frequency division ratio for the clock signal selected using SRC3_IN_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided. B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited |
| 15 to 13 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------------------|---------------|-----|--|
| 12 to 8 | SRC2_IN_DIVCLK_SEL [4:0] | All 0 | R/W | SRC2 Input Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |
| 7 to 5 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------------------|---------------|-----|--|
| 4 to 0 | SRC2_IN_DIVRATIO_SEL[4:0] | All - | R/W | <p>SRC2 Input Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC2_IN_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |

42.2.10 SRC Input Timing Select Register 2 (SRCIN_TIMSEL2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCIN_TIMSEL2 selects the input timing signals for SRC4 and SRC5.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-------------------------|-----|-----|-----|-----|----|----|---------------------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | SRC5_IN_DIVCLK_SEL[4:0] | | | | — | — | — | SRC5_IN_DIVRATIO_SEL[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | SRC4_IN_DIVCLK_SEL[4:0] | | | | — | — | — | SRC4_IN_DIVRATIO_SEL[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------------|---------------|-----|--|
| 31 to 29 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 28 to 24 | SRC5_IN_DIVCLK_SEL [4:0] | All 0 | R/W | SRC5 Input Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------------|---------------|-----|--|
| 23 to 21 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 20 to 16 | SRC5_IN_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>SRC5 Input Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC5_IN_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |
| 15 to 13 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

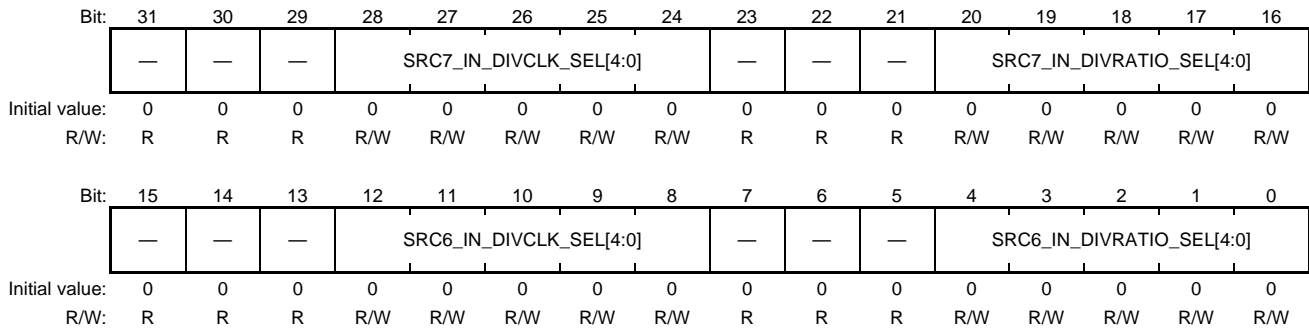
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------------|---------------|-----|--|
| 12 to 8 | SRC4_IN_DIVCLK_SEL [4:0] | All 0 | R/W | SRC4 Input Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |
| 7 to 5 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------------------|---------------|-----|--|
| 4 to 0 | SRC4_IN_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>SRC4 Input Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC4_IN_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |

42.2.11 SRC Input Timing Select Register 3 (SRCIN_TIMSEL3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCIN_TIMSEL3 selects the input timing signals for SRC6 and SRC7.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------------|---------------|-----|--|
| 31 to 29 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 28 to 24 | SRC7_IN_DIVCLK_SEL [4:0] | All 0 | R/W | SRC7 Input Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------------|---------------|-----|--|
| 23 to 21 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 20 to 16 | SRC7_IN_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>SRC7 Input Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC7_IN_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |
| 15 to 13 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

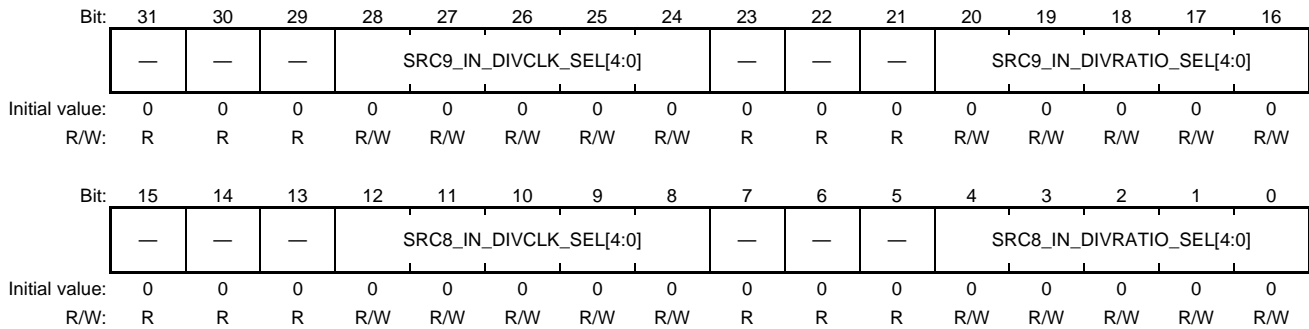
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------------|---------------|-----|--|
| 12 to 8 | SRC6_IN_DIVCLK_SEL [4:0] | All 0 | R/W | SRC6 Input Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |
| 7 to 5 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------------------|---------------|-----|--|
| 4 to 0 | SRC6_IN_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>SRC6 Input Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC6_IN_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |

42.2.12 SRC Input Timing Select Register 4 (SRCIN_TIMSEL4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCIN_TIMSEL4 selects the input timing signals for SRC8 and SRC9.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------------|---------------|-----|--|
| 31 to 29 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 28 to 24 | SRC9_IN_DIVCLK_SEL [4:0] | All 0 | R/W | SRC9 Input Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------------|---------------|-----|--|
| 23 to 21 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 20 to 16 | SRC9_IN_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>SRC9 Input Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC9_IN_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |
| 15 to 13 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

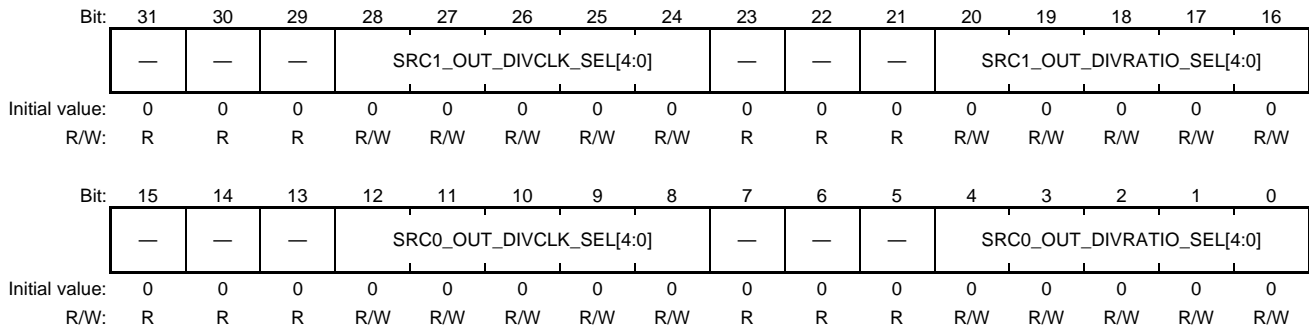
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------------|---------------|-----|--|
| 12 to 8 | SRC8_IN_DIVCLK_SEL [4:0] | All 0 | R/W | SRC8 Input Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |
| 7 to 5 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------------------|---------------|-----|--|
| 4 to 0 | SRC8_IN_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>SRC8 Input Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC8_IN_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |

42.2.13 SRC Output Timing Select Register 0 (SRCOUT_TIMSEL0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCOUT_TIMSEL0 selects the output timing signals for SRC0 and SRC1.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------------|---------------|-----|---|
| 31 to 29 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 28 to 24 | SRC1_OUT_DIVCLK_SEL [4:0] | All 0 | R/W | SRC1 Output Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------------|---------------|-----|---|
| 23 to 21 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 20 to 16 | SRC1_OUT_DIVRATIO_SEL[4:0] | All 0 | R/W | SRC1 Output Timing Signal Frequency Division Ratio Select These bits select the frequency division ratio for the clock signal selected using SRC1_OUT_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided. B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited |
| 15 to 13 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

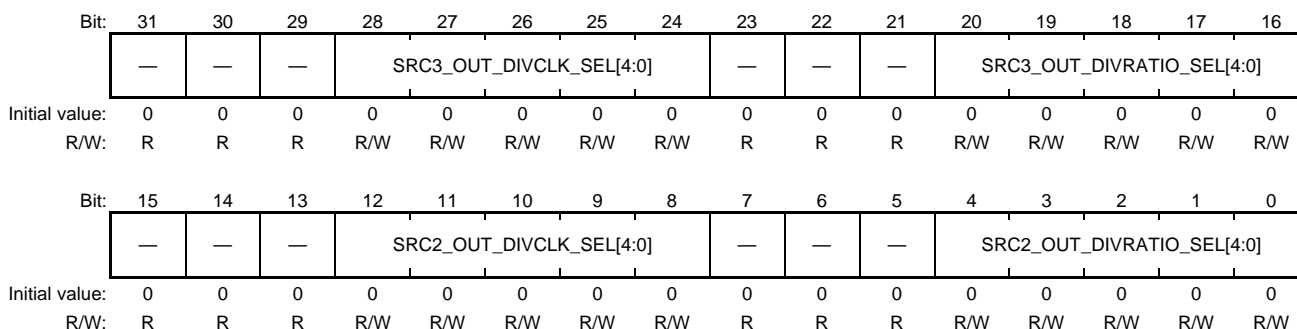
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------------------|---------------|-----|---|
| 12 to 8 | SRC0_OUT_DIVCLK_SEL [4:0] | All 0 | R/W | SRC0 Output Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |
| 7 to 5 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------------------|---------------|-----|--|
| 4 to 0 | SRC0_OUT_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>SRC0 Output Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC0_OUT_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |

42.2.14 SRC Output Timing Select Register 1 (SRCOUT_TIMSEL1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCOUT_TIMSEL1 selects the output timing signals for SRC2 and SRC3.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------------|---------------|-----|---|
| 31 to 29 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 28 to 24 | SRC3_OUT_DIVCLK_SEL [4:0] | All 0 | R/W | SRC3 Output Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------------|---------------|-----|---|
| 23 to 21 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 20 to 16 | SRC3_OUT_DIVRATIO_SEL[4:0] | All 0 | R/W | SRC3 Output Timing Signal Frequency Division Ratio Select These bits select the frequency division ratio for the clock signal selected using SRC3_OUT_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided. B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited |
| 15 to 13 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

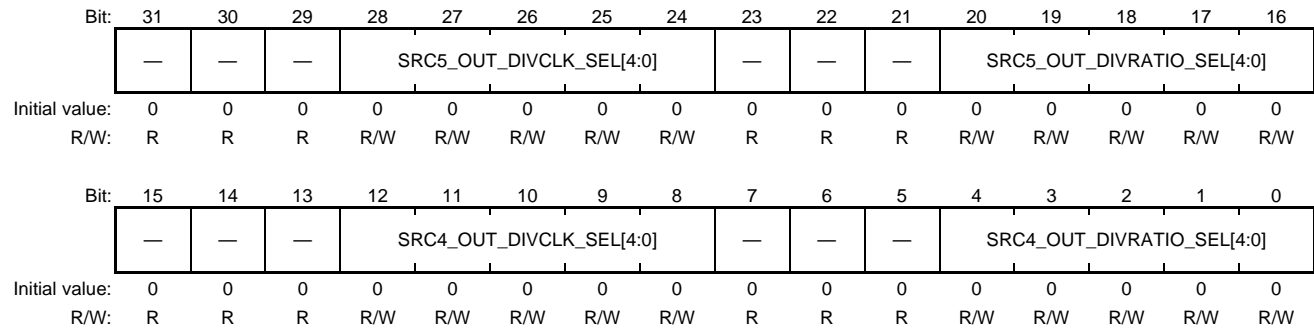
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------------------|---------------|-----|--|
| 12 to 8 | SRC2_OUT_DIVCLK_SEL [4:0] | All 0 | R/W | <p>SRC2 Output Timing Signal Select</p> <p>B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved</p> |
| 7 to 5 | — | All 0 | R | <p>Reserved</p> <p>The initial value is always read from these bits. The write value should always be the initial value.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------------------|---------------|-----|--|
| 4 to 0 | SRC2_OUT_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>SRC2 Output Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC2_OUT_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |

42.2.15 SRC Output Timing Select Register 2 (SRCOUT_TIMSEL2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCOUT_TIMSEL2 selects the output timing signals for SRC4 and SRC5.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------------|---------------|-----|---|
| 31 to 29 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 28 to 24 | SRC5_OUT_DIVCLK_SEL [4:0] | All 0 | R/W | SRC5 Output Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------------|---------------|-----|--|
| 23 to 21 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 20 to 16 | SRC5_OUT_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>SRC5 Output Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC5_OUT_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |
| 15 to 13 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------------------|---------------|-----|---|
| 12 to 8 | SRC4_OUT_DIVCLK_SEL [4:0] | All 0 | R/W | SRC4 Output Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |
| 7 to 5 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------------------|---------------|-----|--|
| 4 to 0 | SRC4_OUT_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>SRC4 Output Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC4_OUT_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |

42.2.16 SRC Output Timing Select Register 3 (SRCOUT_TIMSEL3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCOUT_TIMSEL3 selects the output timing signals for SRC6 and SRC7.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|--------------------------|-----|-----|-----|-----|----|----|----------------------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | SRC7_OUT_DIVCLK_SEL[4:0] | | | | — | — | — | SRC7_OUT_DIVRATIO_SEL[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | SRC6_OUT_DIVCLK_SEL[4:0] | | | | — | — | — | SRC6_OUT_DIVRATIO_SEL[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------------|---------------|-----|---|
| 31 to 29 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 28 to 24 | SRC7_OUT_DIVCLK_SEL [4:0] | All 0 | R/W | SRC7 Output Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------------|---------------|-----|--|
| 23 to 21 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 20 to 16 | SRC7_OUT_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>SRC7 Output Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC7_OUT_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |
| 15 to 13 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------------------|---------------|-----|---|
| 12 to 8 | SRC6_OUT_DIVCLK_SEL [4:0] | All 0 | R/W | SRC6 Output Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |
| 7 to 5 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------------------|---------------|-----|--|
| 4 to 0 | SRC6_OUT_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>SRC6 Output Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC6_OUT_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |

42.2.17 SRC Output Timing Select Register 4 (SRCOUT_TIMSEL4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCOUT_TIMSEL4 selects the output timing signals for SRC8 and SRC9.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|--------------------------|-----|-----|-----|-----|----|----|----------------------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | SRC9_OUT_DIVCLK_SEL[4:0] | | | | — | — | — | SRC9_OUT_DIVRATIO_SEL[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | SRC8_OUT_DIVCLK_SEL[4:0] | | | | — | — | — | SRC8_OUT_DIVRATIO_SEL[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------------|---------------|-----|---|
| 31 to 29 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 28 to 24 | SRC9_OUT_DIVCLK_SEL [4:0] | All 0 | R/W | SRC9 Output Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------------|---------------|-----|---|
| 23 to 21 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |
| 20 to 16 | SRC9_OUT_DIVRATIO_SEL[4:0] | All 0 | R/W | SRC9 Output Timing Signal Frequency Division Ratio Select These bits select the frequency division ratio for the clock signal selected using SRC9_OUT_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided. B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited |
| 15 to 13 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------------------|---------------|-----|--|
| 12 to 8 | SRC8_OUT_DIVCLK_SEL [4:0] | All 0 | R/W | <p>SRC8 Output Timing Signal Select</p> <p>B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved</p> |
| 7 to 5 | — | All 0 | R | <p>Reserved</p> <p>The initial value is always read from these bits. The write value should always be the initial value.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------------------|---------------|-----|--|
| 4 to 0 | SRC8_OUT_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>SRC8 Output Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using SRC8_OUT_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |

42.2.18 CMD Output Timing Select Register (CMDOUT_TIMSEL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CMDOUT_TIMSEL selects the output timing signals for CMD0 and CMD1.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|--------------------------|-----|-----|-----|-----|----|----|----------------------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | CMD1_OUT_DIVCLK_SEL[4:0] | | | | — | — | — | CMD1_OUT_DIVRATIO_SEL[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | CMD0_OUT_DIVCLK_SEL[4:0] | | | | — | — | — | CMD0_OUT_DIVRATIO_SEL[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 29 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------------------|---------------|-----|---|
| 28 to 24 | CMD1_OUT_DIVCLK_SEL [4:0] | All 0 | R/W | CMD1 Output Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |
| 23 to 21 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------------|---------------|-----|--|
| 20 to 16 | CMD1_OUT_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>CMD1 Output Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using CMD1_OUT_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |
| 15 to 13 | — | All 0 | R | <p>Reserved</p> <p>The initial value is always read from these bits. The write value should always be the initial value.</p> |

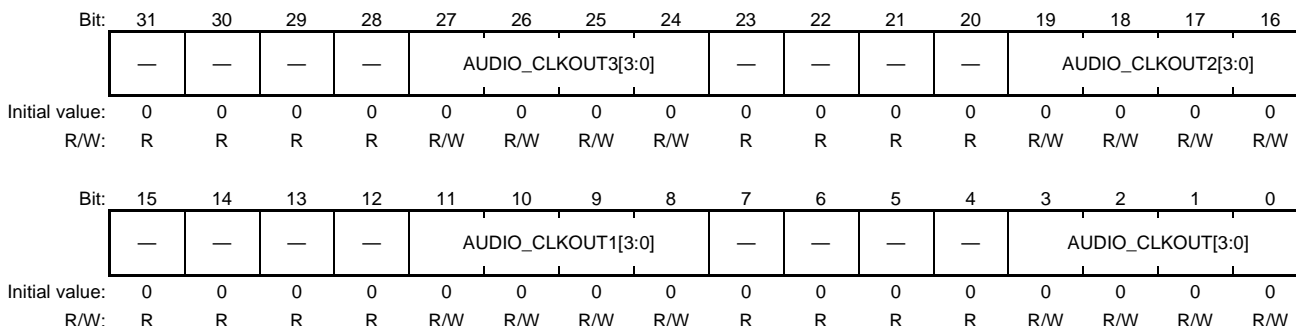
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------------------|---------------|-----|---|
| 12 to 8 | CMD0_OUT_DIVCLK_SEL [4:0] | All 0 | R/W | CMD0 Output Timing Signal Select B'0_0000: Divided AUDIO_CLKA B'0_0001: Divided AUDIO_CLKB B'0_0010: Divided AUDIO_CLKC B'0_0011: Divided BRGA B'0_0100: Divided BRGB B'0_0101: Reserved B'0_0110: ssi_ws0 B'0_0111: ssi_ws1 B'0_1000: ssi_ws2 B'0_1001: ssi_ws3 B'0_1010: ssi_ws4 B'0_1011: ssi_ws5 B'0_1100: ssi_ws6 B'0_1101: ssi_ws7 B'0_1110: ssi_ws9 B'0_1111: Setting prohibited B'1_0000: avb_div8[0] B'1_0001: avb_div8[1] B'1_0010: avb_div8[2] B'1_0011: avb_div8[3] B'1_0100: avb_div8[4] B'1_0101: avb_div8[5] B'1_0110: avb_div8[6] B'1_0111: avb_div8[7] Others: reserved |
| 7 to 5 | — | All 0 | R | Reserved The initial value is always read from these bits. The write value should always be the initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------------------|---------------|-----|--|
| 4 to 0 | CMD0_OUT_DIVRATIO_SEL[4:0] | All 0 | R/W | <p>CMD0 Output Timing Signal Frequency Division Ratio Select</p> <p>These bits select the frequency division ratio for the clock signal selected using CMD0_OUT_DIVCLK_SEL[3:0]. Note that ssi_ws signals cannot be divided.</p> <p>B'0_0000: Divided by 2 B'0_0001: Divided by 4 B'0_0010: Divided by 6 B'0_0011: Divided by 8 B'0_0100: Divided by 12 B'0_0101: Divided by 16 B'0_0110: Divided by 24 B'0_0111: Divided by 32 B'0_1000: Divided by 48 B'0_1001: Divided by 64 B'0_1010: Divided by 96 B'0_1011: Divided by 128 B'0_1100: Divided by 192 B'0_1101: Divided by 256 B'0_1110: Divided by 384 B'0_1111: Divided by 512 B'1_0000: Divided by 768 B'1_0001: Divided by 1024 B'1_0010: Divided by 1536 B'1_0011: Divided by 2048 B'1_0100: Divided by 3072 B'1_0101: Divided by 4096 B'1_0110: Divided by 6144 B'1_0111: Divided by 8192 B'1_1000: Divided by 12288 B'1_1001: Divided by 16384 B'1_1010: Divided by 24576 B'1_1011: Divided by 32768 B'1_1100: Divided by 49152 B'1_1101: Divided by 98304 Others: Setting prohibited</p> |

42.2.19 AVB Clock Select Register (AVBCKR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVBCKR selects the clocks input to and output from the ADG.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|---|
| 31 to 28 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |
| 27 to 24 | AUDIO_CLKOUT3 [3:0] | H'0 | R/W | AUDIO_CLKOUT3 Output clock selection. B'0XXX: BRGA/BRGB select B'1000: avb_counter8[0] B'1001: avb_counter8[1] B'1010: avb_counter8[2] B'1011: avb_counter8[3] B'1100: avb_counter8[4] B'1101: avb_counter8[5] B'1110: avb_counter8[6] B'1111: avb_counter8[7] |
| 23 to 20 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |
| 19 to 16 | AUDIO_CLKOUT2 [3:0] | H'0 | R/W | AUDIO_CLKOUT2 Output clock selection. B'0XXX: BRGA/BRGB select B'1000: avb_counter8[0] B'1001: avb_counter8[1] B'1010: avb_counter8[2] B'1011: avb_counter8[3] B'1100: avb_counter8[4] B'1101: avb_counter8[5] B'1110: avb_counter8[6] B'1111: avb_counter8[7] |
| 15 to 12 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |

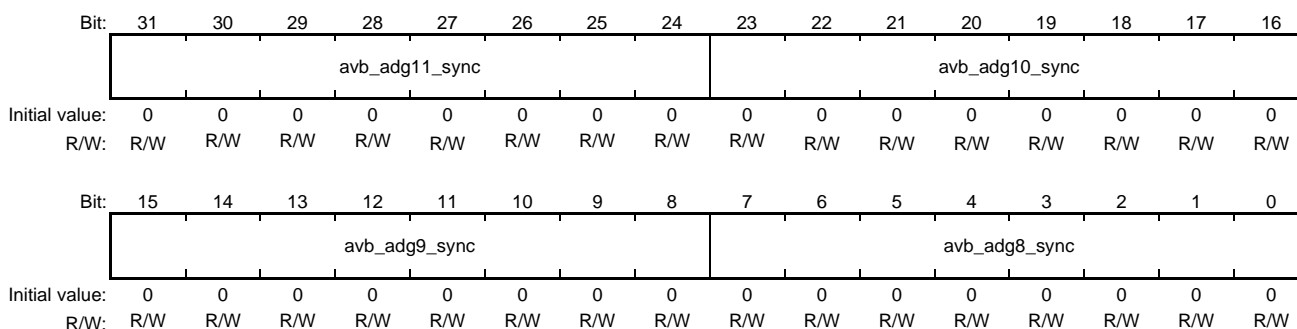
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------------|---------------|-----|---|
| 11 to 8 | AUDIO_CLKOUT1 [3:0] | H'0 | R/W | AUDIO_CLKOUT1 Output clock selection. B'0XXX: BRGA/BRGB select B'1000: avb_counter8[0] B'1001: avb_counter8[1] B'1010: avb_counter8[2] B'1011: avb_counter8[3] B'1100: avb_counter8[4] B'1101: avb_counter8[5] B'1110: avb_counter8[6] B'1111: avb_counter8[7] |
| 7 to 4 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |
| 3 to 0 | AUDIO_CLKOUT [3:0] | H'0 | R/W | AUDIO_CLKOUT Output clock selection. B'0XXX: BRGA/BRGB select B'1000: avb_counter8[0] B'1001: avb_counter8[1] B'1010: avb_counter8[2] B'1011: avb_counter8[3] B'1100: avb_counter8[4] B'1101: avb_counter8[5] B'1110: avb_counter8[6] B'1111: avb_counter8[7] |

Note: Use AUDIO_CLKOUT, AUDIO_CLKOUT1, AUDIO_CLKOUT2, and AUDIO_CLKOUT3 in less than 25 MHz.

42.2.20 AVB sync select Register 0 (AVB_SYNC_SEL0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVB_SYNC_SEL0 selects the output clocks for the EAVB-I/F.

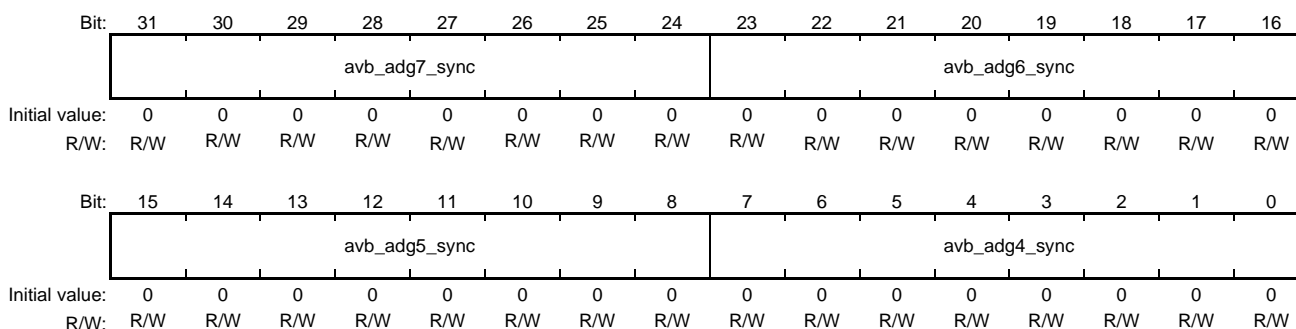


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 24 | avb_adg11_sync | H'00 | R/W | avb_adg*_sync(* = 11 to 8) Output Timing Signal Select |
| 23 to 16 | avb_adg10_sync | H'00 | R/W | B'0000_0000: fixed '0' output |
| 15 to 8 | avb_adg9_sync | H'00 | R/W | B'0000_0001 to B'0000_0111: reserved (fixed '0' output) |
| 7 to 0 | avb_adg8_sync | H'00 | R/W | B'0000_1000: audio_clk_div3[0] B'0000_1001: audio_clk_div3[1] B'0000_1010: audio_clk_div3[2] B'0000_1011 to B'0000_1111: reserved (fixed '0' output) B'0001_0000: ssi_ws0 B'0001_0001: ssi_ws1 B'0001_0010: ssi_ws2 B'0001_0011: ssi_ws3 B'0001_0100: ssi_ws4 B'0001_0101: ssi_ws5 B'0001_0110: ssi_ws6 B'0001_0111: ssi_ws7 B'0001_1000: reserved B'0001_1001: ssi_ws9 B'0001_1010 to B'0001_1111: reserved (fixed '0' output) B'0010_0000: avb_div8[0] B'0010_0001: avb_div8[1] B'0010_0010: avb_div8[2] B'0010_0011: avb_div8[3] B'0010_0100: avb_div8[4] B'0010_0101: avb_div8[5] B'0010_0110: avb_div8[6] B'0010_0111: avb_div8[7] B'0010_1000 to B'1111_1111: reserved (fixed '0' output) |

42.2.21 AVB sync select Register 1 (AVB_SYNC_SEL1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVB_SYNC_SEL1 selects the output clocks for the EAVB-I/F.

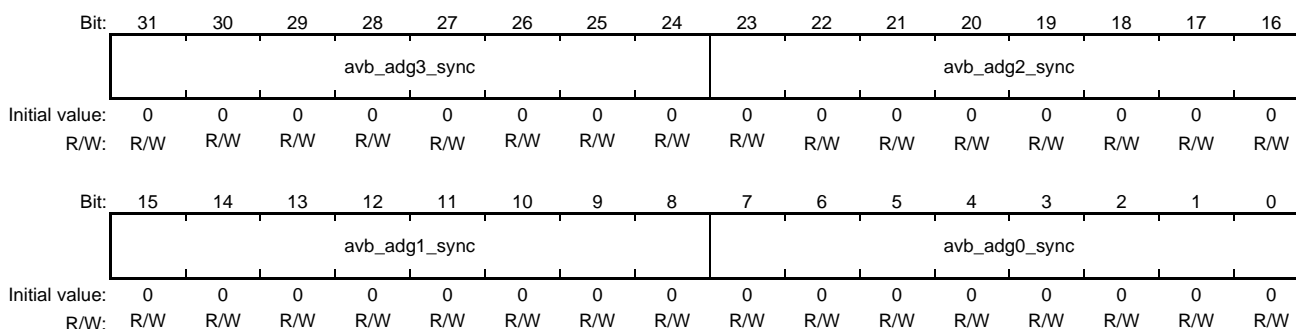


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 24 | avb_adg7_sync | H'00 | R/W | avb_adg*_sync(* = 7to 4) Output Timing Signal Select |
| 23 to 16 | avb_adg6_sync | H'00 | R/W | B'0000_0000: fixed '0' output |
| 15 to 8 | avb_adg5_sync | H'00 | R/W | B'0000_0001 to B'0000_0111: reserved (fixed '0' output) |
| 7 to 0 | avb_adg4_sync | H'00 | R/W | B'0000_1000: audio_clk_div3[0] B'0000_1001: audio_clk_div3[1] B'0000_1010: audio_clk_div3[2] B'0000_1011 to B'0000_1111: reserved (fixed '0' output) B'0001_0000: ssi_ws0 B'0001_0001: ssi_ws1 B'0001_0010: ssi_ws2 B'0001_0011: ssi_ws3 B'0001_0100: ssi_ws4 B'0001_0101: ssi_ws5 B'0001_0110: ssi_ws6 B'0001_0111: ssi_ws7 B'0001_1000: reserved B'0001_1001: ssi_ws9 B'0001_1010 to B'0001_1111: reserved (fixed '0' output) B'0010_0000: avb_div8[0] B'0010_0001: avb_div8[1] B'0010_0010: avb_div8[2] B'0010_0011: avb_div8[3] B'0010_0100: avb_div8[4] B'0010_0101: avb_div8[5] B'0010_0110: avb_div8[6] B'0010_0111: avb_div8[7] B'0010_1000 to B'1111_1111: reserved (fixed '0' output) |

42.2.22 AVB sync select Register 2 (AVB_SYNC_SEL2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVB_SYNC_SEL2 selects the output clocks for the EAVB-I/F.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 24 | avb_adg3_sync | H'00 | R/W | avb_adg*_sync(* = 3 to 0) Output Timing Signal Select |
| 23 to 16 | avb_adg2_sync | H'00 | R/W | B'0000_0000: fixed '0' output |
| 15 to 8 | avb_adg1_sync | H'00 | R/W | B'0000_0001 to B'0000_0111: reserved (fixed '0' output) |
| 7 to 0 | avb_adg0_sync | H'00 | R/W | B'0000_1000: audio_clk_div3[0] B'0000_1001: audio_clk_div3[1] B'0000_1010: audio_clk_div3[2] B'0000_1011 to B'0000_1111: reserved (fixed '0' output) B'0001_0000: ssi_ws0 B'0001_0001: ssi_ws1 B'0001_0010: ssi_ws2 B'0001_0011: ssi_ws3 B'0001_0100: ssi_ws4 B'0001_0101: ssi_ws5 B'0001_0110: ssi_ws6 B'0001_0111: ssi_ws7 B'0001_1000: reserved B'0001_1001: ssi_ws9 B'0001_1010 to B'0001_1111: reserved (fixed '0' output) B'0010_0000: avb_div8[0] B'0010_0001: avb_div8[1] B'0010_0010: avb_div8[2] B'0010_0011: avb_div8[3] B'0010_0100: avb_div8[4] B'0010_0101: avb_div8[5] B'0010_0110: avb_div8[6] B'0010_0111: avb_div8[7] B'0010_1000 to B'1111_1111: reserved (fixed '0' output) |

42.2.23 AVB sync divide Register 0 (AVB_SYNC_DIV0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVB_SYNC_DIV0 selects the clocks to audio_clk_div3

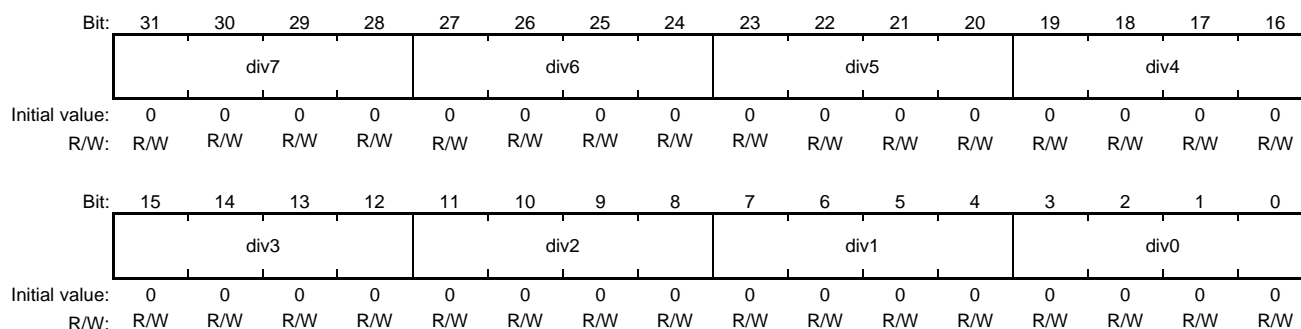
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|------|-----|-----|------|-----|-----|------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | div2 | | | div1 | | | div0 | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |
| 11 to 8 | div2 | H'0 | R/W | audio_clk_div3[2] Frequency Divider Setting: AUDIO_CLKC divided by 2 ⁿ |
| 7 to 4 | div1 | H'0 | R/W | audio_clk_div3[1] Frequency Divider Setting: AUDIO_CLKB divided by 2 ⁿ |
| 3 to 0 | div0 | H'0 | R/W | audio_clk_div3[0] Frequency Divider Setting: AUDIO_CLKA divided by 2 ⁿ |

42.2.24 AVB sync divide Register 1 (AVB_SYNC_DIV1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVB_SYNC_DIV1 selects the clocks to avb_div8

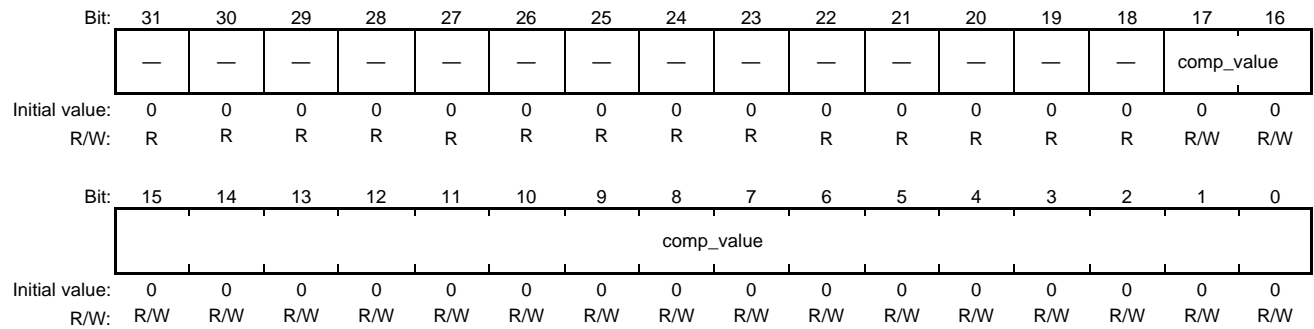


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 28 | div7 | H'0 | R/W | avb_div8[7] Frequency Divider Setting: avb_counter8[7] divided by 2 ⁿ |
| 27 to 24 | div6 | H'0 | R/W | avb_div8[6] Frequency Divider Setting: avb_counter8[6] divided by 2 ⁿ |
| 23 to 20 | div5 | H'0 | R/W | avb_div8[5] Frequency Divider Setting: avb_counter8[5] divided by 2 ⁿ |
| 19 to 16 | div4 | H'0 | R/W | avb_div8[4] Frequency Divider Setting: avb_counter8[4] divided by 2 ⁿ |
| 15 to 12 | div3 | H'0 | R/W | avb_div8[3] Frequency Divider Setting: avb_counter8[3] divided by 2 ⁿ |
| 11 to 8 | div2 | H'0 | R/W | avb_div8[2] Frequency Divider Setting: avb_counter8[2] divided by 2 ⁿ |
| 7 to 4 | div1 | H'0 | R/W | avb_div8[1] Frequency Divider Setting: avb_counter8[1] divided by 2 ⁿ |
| 3 to 0 | div0 | H'0 | R/W | avb_div8[0] Frequency Divider Setting: avb_counter8[0] divided by 2 ⁿ |

42.2.25 AVB clock divider comparison value register 0 (AVB_CLK_DIV0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVB_CLK_DIV0 set the comparison value to avb_counter8[0]



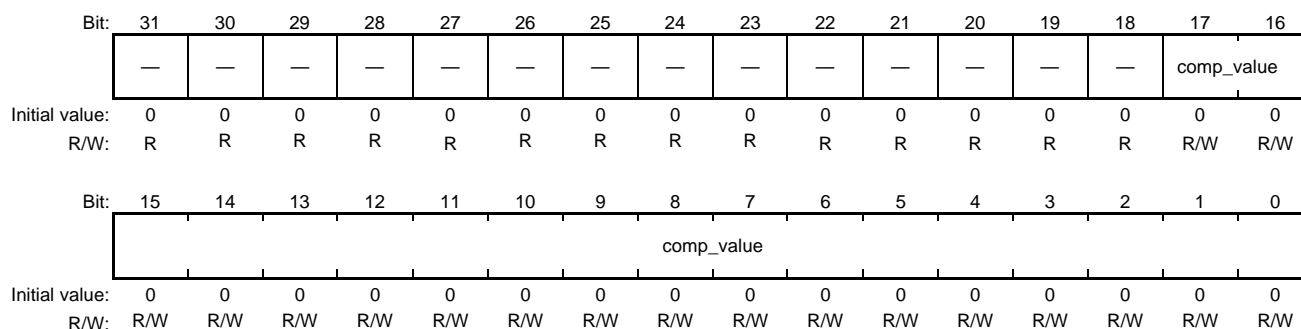
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 18 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |
| 17 to 0 | comp_value | H'0_0000 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] AVB_CLK_DIV0 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and S0D1φ(800 MHz). [RZ/G2E] AVB_CLK_DIV0 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and ZA8φ(600 MHz). |

- Notes: 1. Use avb_counter8[0] in less than 25 MHz.
2. "comp_value" is not updated unless "AVB_CLK_CONFIG.div_en_com" is set.

42.2.26 AVB clock divider comparison value register 1 (AVB_CLK_DIV1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVB_CLK_DIV1 set the comparison value to avb_counter8[1]



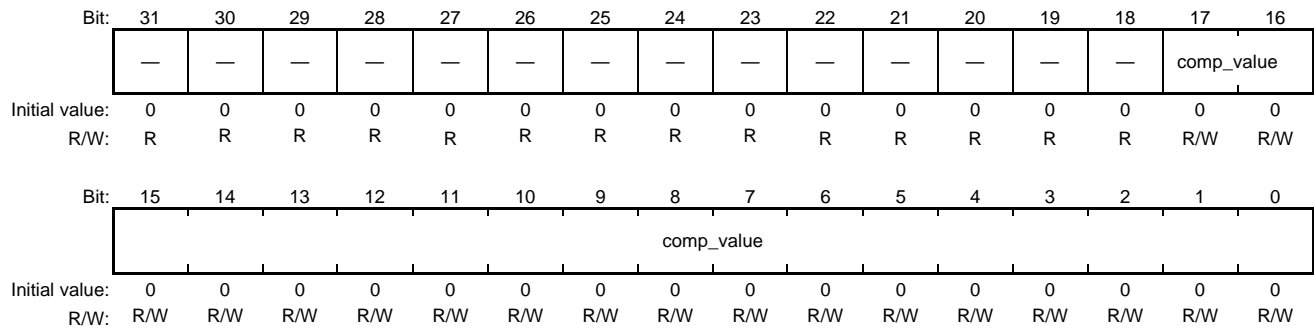
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 18 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |
| 17 to 0 | comp_value | H'0_0000 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] AVB_CLK_DIV1 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and S0D1φ(800 MHz). [RZ/G2E][RZ/G2E] AVB_CLK_DIV1 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and ZA8φ (600 MHz). |

- Notes: 1. Use avb_counter8[1] in less than 25 MHz.
2. "comp_value" is not updated unless "AVB_CLK_CONFIG.div_en_com" is set.

42.2.27 AVB clock divider comparison value register 2 (AVB_CLK_DIV2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVB_CLK_DIV2 set the comparison value to avb_counter8[2]



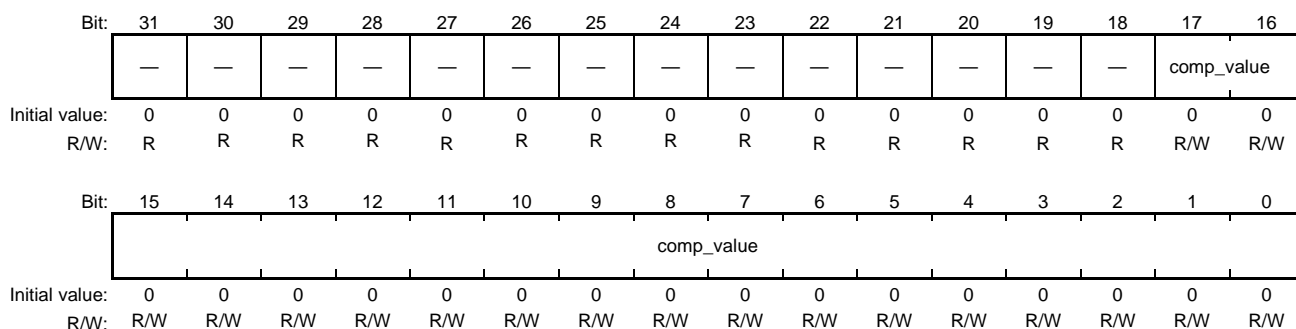
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 18 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |
| 17 to 0 | comp_value | H'0_0000 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] AVB_CLK_DIV2 comparison value (Setting area = H'0_0000 ~ H'3_ffc0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and S0D1φ(800 MHz). [RZ/G2E][RZ/G2E] AVB_CLK_DIV2 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and ZA8φ (600 MHz). |

- Notes: 1. Use avb_counter8[2] in less than 25 MHz.
2. "comp_value" is not updated unless "AVB_CLK_CONFIG.div_en_com" is set.

42.2.28 AVB clock divider comparison value register 3 (AVB_CLK_DIV3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVB_CLK_DIV3 set the comparison value to avb_counter8[3]



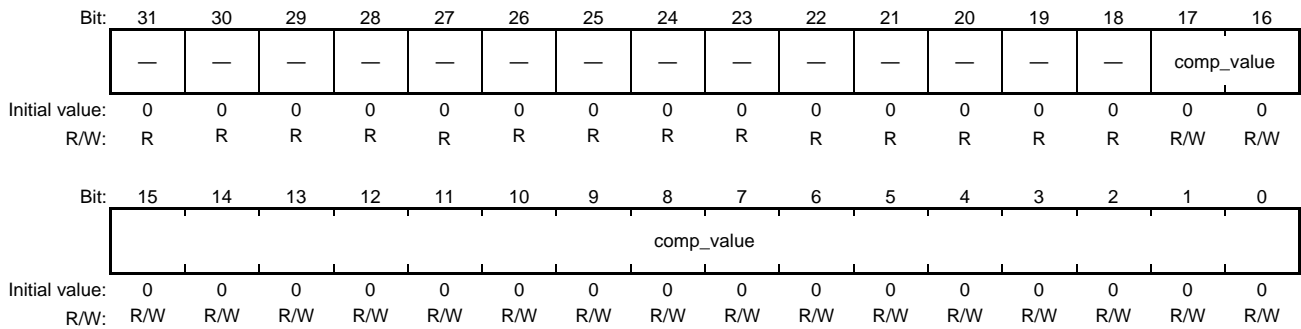
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 18 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |
| 17 to 0 | comp_value | H'0_0000 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] AVB_CLK_DIV3 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and S0D1φ(800 MHz). [RZ/G2E][RZ/G2E] AVB_CLK_DIV3 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and ZA8φ (600 MHz). |

- Notes: 1. Use avb_counter8[3] in less than 25 MHz.
2. "comp_value" is not updated unless "AVB_CLK_CONFIG.div_en_com" is set.

42.2.29 AVB clock divider comparison value register 4 (AVB_CLK_DIV4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVB_CLK_DIV4 set the comparison value to avb_counter8[4]



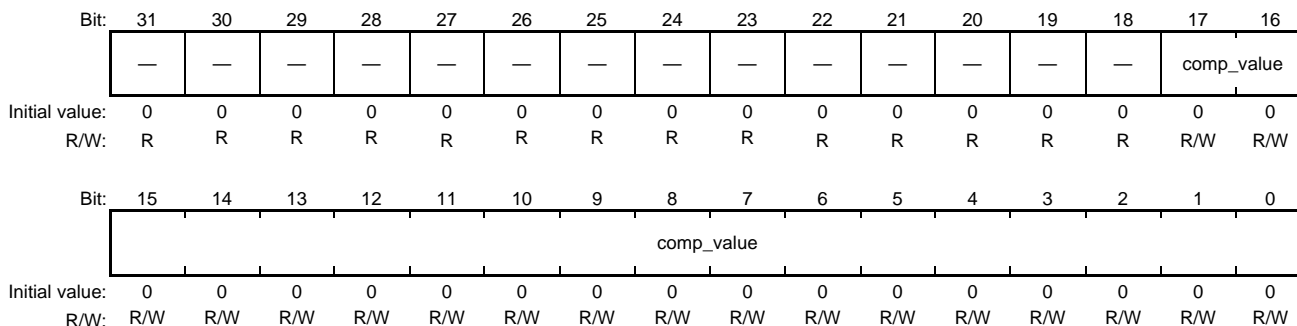
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 18 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |
| 17 to 0 | comp_value | H'0_0000 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] AVB_CLK_DIV4 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and S0D1φ(800 MHz). [RZ/G2E] [RZ/G2E] AVB_CLK_DIV4 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and ZA8φ (600 MHz). |

- Notes: 1. Use avb_counter8[4] in less than 25 MHz.
2. "comp_value" is not updated unless "AVB_CLK_CONFIG.div_en_com" is set.

42.2.30 AVB clock divider comparison value register 5 (AVB_CLK_DIV5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVB_CLK_DIV5 set the comparison value to avb_counter8[5]



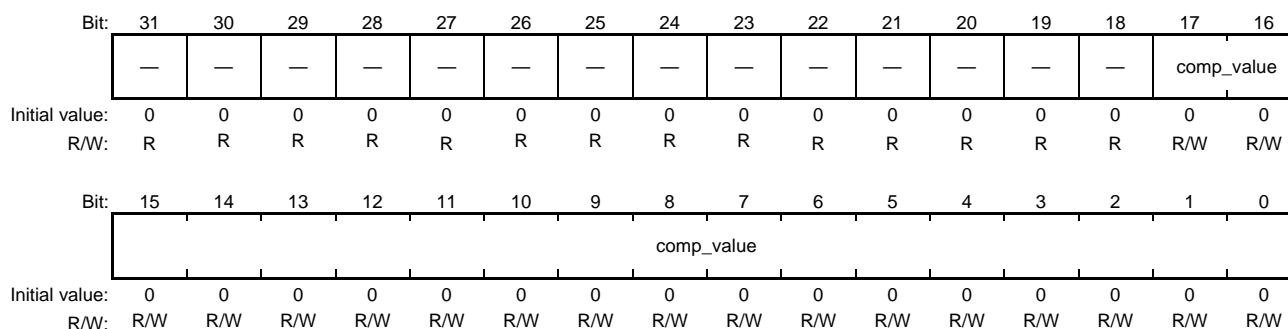
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 18 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |
| 17 to 0 | comp_value | H'0_0000 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] AVB_CLK_DIV5 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and S0D1φ(800 MHz). [RZ/G2E][RZ/G2E] AVB_CLK_DIV5 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and ZA8φ (600 MHz). |

- Notes: 1. Use avb_counter8[5] in less than 25 MHz.
2. "comp_value" is not updated unless "AVB_CLK_CONFIG.div_en_com" is set.

42.2.31 AVB clock divider comparison value register 6 (AVB_CLK_DIV6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVB_CLK_DIV6 set the comparison value to avb_counter8[6]



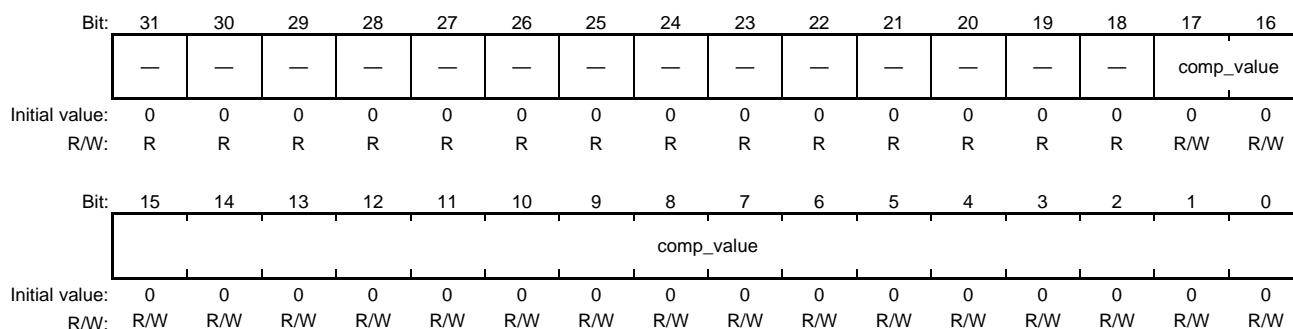
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 18 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |
| 17 to 0 | comp_value | H'0_0000 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] AVB_CLK_DIV6 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and S0D1φ(800 MHz). [RZ/G2E][RZ/G2E] AVB_CLK_DIV6 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and ZA8φ (600 MHz). |

- Notes: 1. Use avb_counter8[6] in less than 25 MHz.
2. "comp_value" is not updated unless "AVB_CLK_CONFIG.div_en_com" is set.

42.2.32 AVB clock divider comparison value register 7 (AVB_CLK_DIV7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVB_CLK_DIV7 set the comparison value to avb_counter8[7]



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 18 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |
| 17 to 0 | comp_value | H'0_0000 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] AVB_CLK_DIV7 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and S0D1φ(800 MHz). [RZ/G2E][RZ/G2E] AVB_CLK_DIV7 comparison value (Setting area = H'0_0000 ~ H'3_FFC0) A value is determined in consideration of the gap of gPTP timer(EAVB-I/F) and ZA8φ (600 MHz). |

- Notes: 1. Use avb_counter8[7] in less than 25 MHz.
2. "comp_value" is not updated unless "AVB_CLK_CONFIG.div_en_com" is set.

42.2.33 AVB clock divider configuration register (AVB_CLK_CONFIG)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: AVB_CLK_CONFIG set the divider enable to avb_counter8

| | | | | | | | | | | | | | | | | | | |
|----------------|------------|----|----|----|----|----|----|----|-------------|-----|-----|-----|-----|-----|-----|-----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | div_en_com | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| | | | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | div_en[7:0] | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|--|
| 31 | div_en_com | B'0 | R/W | divider enable (common) 1: enable divider 0: disable all divider |
| 30 to 8 | — | All 0 | R | Reserved The initial value is always read from this bit. The write value should always be the initial value. |
| 7 to 0 | div_en[7:0] | All 0 | R/W | divider enable (for AVB_CLK_DIV7 to AVB_CLK_DIV0) 1: enable divider and load comparison value 0: disable divider |

42.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The following sections give examples of setting the ADG registers in two cases.

(1) Supplying SSI with Audio Clock

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

The following procedure can be used to divide the internal clock (S0D4 ϕ : 200 MHz) by d'18 and supply the resulting BRGB clock (11.111 MHz) to SSI.

If the procedure is applied before starting SSI3, steps 1 to 5 can be executed in any order.

1. Set BRRB so that the frequency division ratio should be d'18.
2. Select internal clock (S0D4 ϕ) as the clock to be input to BRGB using BRGCKR.
3. Select the BRGB output clock as SSI3 clock using AUDIO_CLK_SEL0.
4. Release the ADG from the module standby state using RMSTPCR9 or SMSTPCR9 of the CPG module.
5. Release SSI3 from the module standby state using RMSTPCR10 or SMSTPCR10 of the CPG module.

[RZ/G2E]

The following procedure can be used to divide the internal clock (ZA2 ϕ in the case of 200 MHz) by d'18 and supply the resulting BRGB clock (11.111 MHz) to SSI.

If the procedure is applied before starting SSI3, steps 1 to 5 can be executed in any order.

1. Set BRRB so that the frequency division ratio should be d'18.
2. Select internal clock (ZA2 ϕ in the case of 200 MHz) as the clock to be input to BRGB using BRGCKR.
3. Select the BRGB output clock as SSI3 clock using AUDIO_CLK_SEL0.
4. Release the ADG from the module standby state using RMSTPCR9 or SMSTPCR9 of the CPG module.
5. Release SSI3 from the module standby state using RMSTPCR10 or SMSTPCR10 of the CPG module.

Table 42.6 Setting Example 1

| Register Name | Address | Setting Value | Description |
|--------------------------|----------------------------|-------------------|--|
| BRRB | H'EC5A_0004 | H'0000_0008 | Divided by d'18 |
| BRGCKR | H'EC5A_0008 | H'2302_0000 | [RZ/G2H, RZ/G2M V1.3, RZ/G RZ/G2M V3.0, RZ/G2N] Selects S0D4 ϕ as the clock to be input to BRGB. [RZ/G2E] Selects ZA2 ϕ as the clock to be input to BRGB. |
| AUDIO_CLK_SEL0 | H'EC5A_000C | H'2000_0000 | Selects the BRGB output clock as the clock to be output to SSI3. |
| RMSTPCR9* or SMSTPCR9* | H'E615_0984 H'E615_0994 | Set bit 22. | Releases the ADG from the module standby state. |
| RMSTPCR10* or SMSTPCR10* | H'E615_0988 H'E615_0998 | Set bits 12 to 0. | Releases the SSI3 from the module standby state. |

Note: * CPG register is enabled by writing the inverse of the value to CPGWPR register.

(2) The function of avb_counter8

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

The avb_counter8 is a clock generator. The avb_counter8 has dividers which operates with S0D1φ clock. The avb_counter8 has eight output clocks. The avb_counter8 has the internal counter and the setting registers (“AVB clock divider comparison value register 0 (AVB_CLK_DIV0)” to “AVB clock divider comparison value register 7 (AVB_CLK_DIV7)”). The internal counter and setting registers consist of Integral field of 12 bits, and Fractional field of 6 bits. The increment value of the internal counter is H'40. This increment value is fixation. The output clock is toggled when the value of the internal counter becomes beyond the value of AVB_CLK_DIV*.comp_value (* = 0 to 7). After that, the internal counter subtracts the value of AVB_CLK_DIV*. comp_value (* = 0 to 7) and continues operation. In addition, note setting up AVB_CLK_DIV*.comp_value (* = 0 to 7) after setting AVB_CLK_CONFIG.div_en_com.

[RZ/G2E]

The avb_counter8 is a clock generator. The avb_counter8 has dividers which operates with ZA8φ clock. The avb_counter8 has eight output clocks. The avb_counter8 has the internal counter and the setting registers (“AVB clock divider comparison value register 0 (AVB_CLK_DIV0)” to “AVB clock divider comparison value register 7 (AVB_CLK_DIV7)”). The internal counter and setting registers consist of Integral field of 12 bits, and Fractional field of 6 bits. The increment value of the internal counter is H'40. This increment value is fixation. The output clock is toggled when the value of the internal counter becomes beyond the value of AVB_CLK_DIV*.comp_value (* = 0 to 7). After that, the internal counter subtracts the value of AVB_CLK_DIV*. comp_value(* = 0 to 7) and continues operation. In addition, note setting up AVB_CLK_DIV*.comp_value(* = 0 to 7) after setting AVB_CLK_CONFIG.div_en_com.

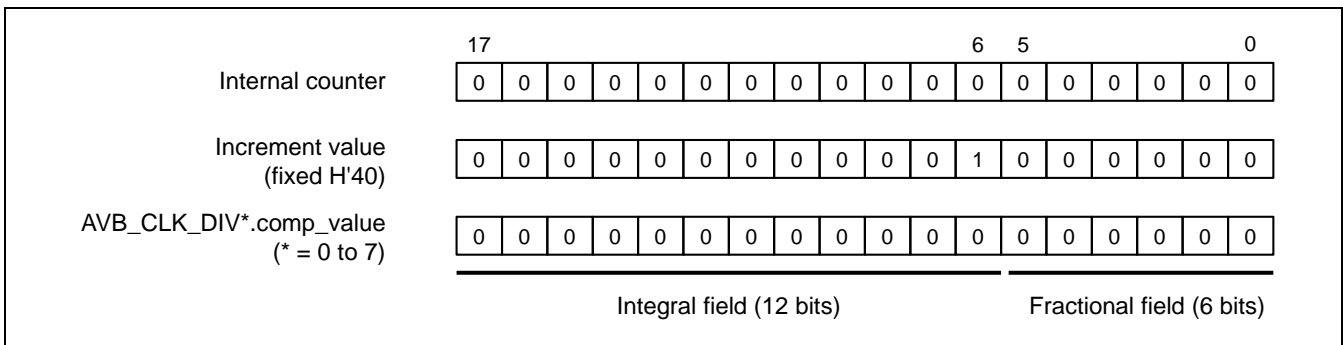


Figure 42.3 Structure of the internal counter and setting registers

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Table 42.7 The example of the setting of AVB_CLK_DIV*.comp_value (* = 0 to 7)

| AVB_CLK_DIV*.comp_value (* = 0 to 7)(Hex) | Integral field (dec) | Fractional field (dec) | Frequency of output clock (MHz) |
|--|----------------------|---------------------------|------------------------------------|
| 00413 | 16 | 19 | 24.545 |
| 00412 | 16 | 18 | 24.568 |
| 00411 | 16 | 17 | 24.592 |

Frequency of the output clock [MHz]

$$= 1 / (((Integral\ field\ value * 1.25ns) * 64 + (Fractional\ field\ value * 1.25ns)) / 32)$$

(SOD1φ clock = 800 MHz(1.25ns))

[RZ/G2E]

Table 42.8 The example of the setting of AVB_CLK_DIV*.comp_value (* = 0 to 7)

| AVB_CLK_DIV*.comp_value (* = 0 to 7)(Hex) | Integral field (dec) | Fractional field (dec) | Frequency of output clock (MHz) |
|--|----------------------|---------------------------|------------------------------------|
| 0030F | 12 | 15 | 24.521 |
| 0030E | 12 | 14 | 24.552 |
| 0030D | 12 | 13 | 24.584 |

Frequency of the output clock [MHz]

$$= 1 / (((Integral\ field\ value * 1.67ns) * 64 + (Fractional\ field\ value * 1.67ns)) / 32)$$

(ZA8φ clock = 600 MHz(1.67ns))

42.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

42.4.1 Note on using AUDIO_CLK and Input/Output timing signal in stable condition

AUDIO_CLK and Input/Output timing signal should be in a stable state before starting transfer of related audio modules. When Input/Output timing signal counts the number of divider register setting, it becomes stable.

42.4.2 Note on using ZA2 ϕ

When ADG use ZA2 ϕ , it can not apply SSCG. For details, refer to 8.2.20 [RZ/G2E]

43. Sampling Rate Converter Unit (SCU)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

43.1 Overview

The SCU has ten SRC modules (six for high-sound-quality type; four for general-sound-quality type) that are useful for synchronization of asynchronous data, which is necessary for data transfer with external memory or external devices. It also provides the functions to change the number of channels, perform mixing, and control the volume.

Note: When the SRCs are not in use, configure the system with the same clock sources for AUDIO CLOCK that is input to ADG, and INIC™ or devices for I2S that are connected to external modules, and all thus operate at the same sampling frequency.

43.1.1 Features

(1) Sampling Rate Converter (SRC)

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

- Asynchronous sampling rate conversion is available
- Supports resolutions up to 24 bits
- High-sound-quality type (THD + N* is -132 dB) and general-sound-quality type (THD + N* is -96 dB)
- Automatically generates antialiasing filter coefficients
- Four modules support one, two, four, six, or eight channels, and six modules support one or two channels.

Note: * Total harmonic distortion plus noise

[RZ/G2E]

- Asynchronous sampling rate conversion is available
- Supports resolutions up to 24 bits
- High-sound-quality type (THD + N* is -132 dB) and general-sound-quality type (THD + N* is -96 dB)
- Automatically generates antialiasing filter coefficients
- Two modules support one, two, four, six, or eight channels, and eight modules support one or two channels.

Note: * Total harmonic distortion plus noise

(2) Channel Transfer Unit (CTU)

- Downmixing and splitter functions
 - Conversion of eight input channels into two output channels
 - Conversion of six input channels into two output channels
 - Conversion of two input channels into four sets of two output channels
 - Conversion of one input channel into eight sets of one output channel
 - No conversion

(3) Mixer (MIX)

- Mixing (adds) two to four sources into one
- Ratio for adding sources is selectable
- Ratio is dynamically changeable

- Mixing with volume ramp is available (ramp period is selectable)

(4) Digital Volume and Mute Function (DVC)

- Volume control function including digital volume, volume ramp, and zero-crossing mute
- The digital volume function is specified by a 24-bit fixed-point value within the range from 0 to 8 times (mute or -120 to 18 dB)
- The volume ramp function can be used for soft mute, fade-in, fade-out, or desired volume adjustment
- The volume ramp period can be changed within the sampling range from the 0th to 23rd power of 2
- The zero-crossing mute function silences the sound at the zero-crossing point of the audio data

The CTU, MIX, and DVC functional blocks are collectively called "CMD".

43.1.2 Block Diagram

Figure 43.1 shows the SCU block diagram.

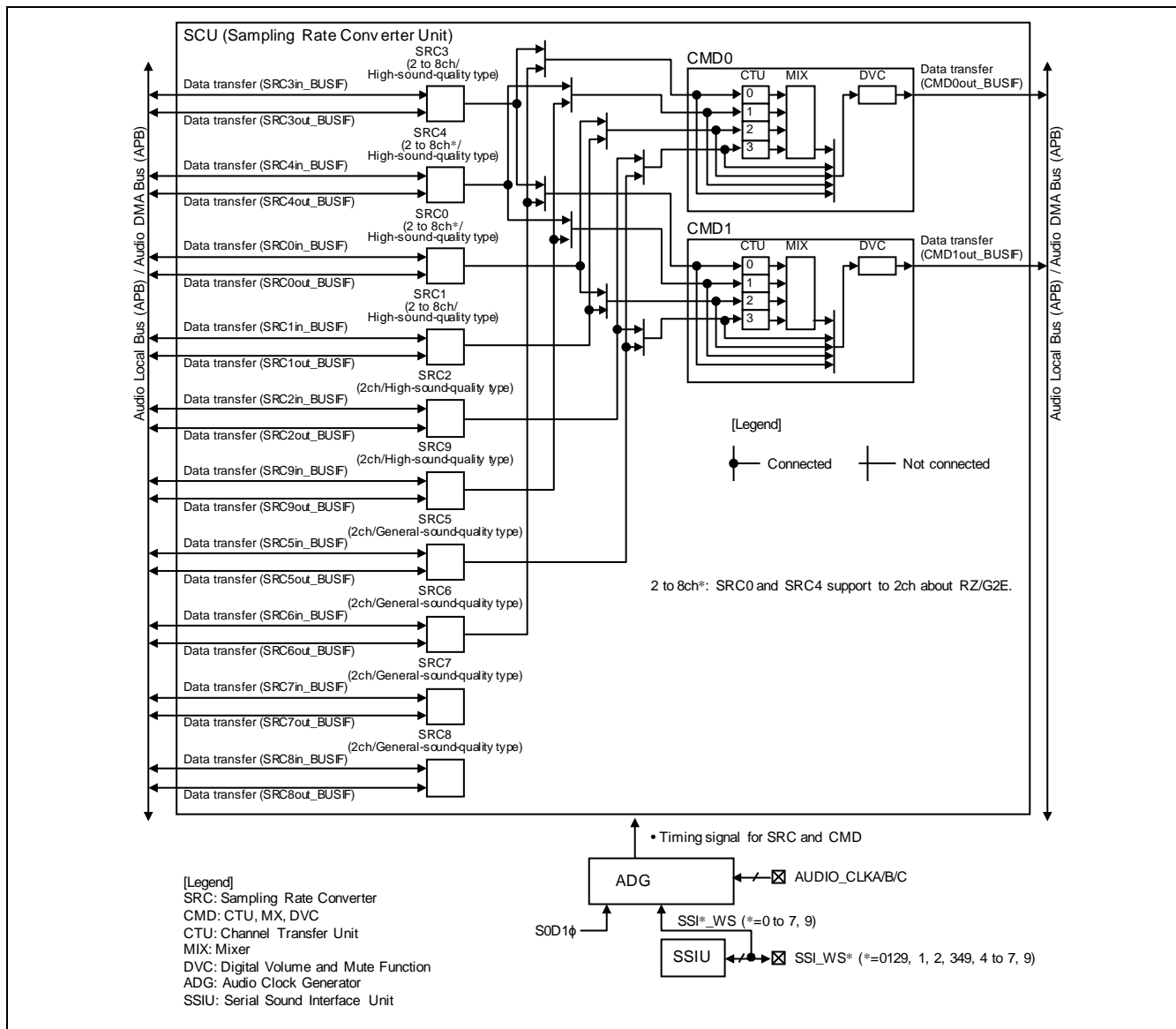


Figure 43.1 Block Diagram of SCU

43.1.3 External Pins

There's no external pin of this module.

43.1.4 Register Configuration

Table 43.1 shows the register configuration. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined. Access the register as a longword (32 bits). Operation cannot be guaranteed if the register is not accessed as a longword.

Table 43.1 Register Configuration

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|----------------------------------|--------------------|-----|-------------|---|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SRC0in_BUSIF_MODE Register | SRC0in_BUSIF_MODE | R/W | H'EC50_0000 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC0out_BUSIF_MODE Register | SRC0out_BUSIF_MODE | R/W | H'EC50_0004 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC0_BUSIF_DALIGN Register | SRC0_BUSIF_DALIGN | R/W | H'EC50_0008 | H'7654_3210 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] H'0000_0010 [RZ/G2E] | 32 | √ | √ | √ | √ |
| SRC0_MODE Register | SRC0_MODE | R/W | H'EC50_000C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC0 Control Register | SRC0_CONTROL | R/W | H'EC50_0010 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC0 Status Register | SRC0_STATUS | R | H'EC50_0014 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC0 Interrupt Enable Register 0 | SRC0_INT_ENABLE0 | R/W | H'EC50_0018 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC1in_BUSIF_MODE Register | SRC1in_BUSIF_MODE | R/W | H'EC50_0020 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC1out_BUSIF_MODE Register | SRC1out_BUSIF_MODE | R/W | H'EC50_0024 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC1_BUSIF_DALIGN Register | SRC1_BUSIF_DALIGN | R/W | H'EC50_0028 | H'7654_3210 | 32 | √ | √ | √ | √ |
| SRC1_MODE Register | SRC1_MODE | R/W | H'EC50_002C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC1 Control Register | SRC1_CONTROL | R/W | H'EC50_0030 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC1 Status Register | SRC1_STATUS | R | H'EC50_0034 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC1 Interrupt Enable Register 0 | SRC1_INT_ENABLE0 | R/W | H'EC50_0038 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC2in_BUSIF_MODE Register | SRC2in_BUSIF_MODE | R/W | H'EC50_0040 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC2out_BUSIF_MODE Register | SRC2out_BUSIF_MODE | R/W | H'EC50_0044 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC2_BUSIF_DALIGN Register | SRC2_BUSIF_DALIGN | R/W | H'EC50_0048 | H'0000_0010 | 32 | √ | √ | √ | √ |
| SRC2_MODE Register | SRC2_MODE | R/W | H'EC50_004C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC2 Control Register | SRC2_CONTROL | R/W | H'EC50_0050 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC2 Status Register | SRC2_STATUS | R | H'EC50_0054 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC2 Interrupt Enable Register 0 | SRC2_INT_ENABLE0 | R/W | H'EC50_0058 | H'0000_0000 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|----------------------------------|--------------------|-----|-------------|---|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SRC3in_BUSIF_MODE Register | SRC3in_BUSIF_MODE | R/W | H'EC50_0060 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC3out_BUSIF_MODE Register | SRC3out_BUSIF_MODE | R/W | H'EC50_0064 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC3_BUSIF_DALIGN Register | SRC3_BUSIF_DALIGN | R/W | H'EC50_0068 | H'7654_3210 | 32 | √ | √ | √ | √ |
| SRC3_MODE Register | SRC3_MODE | R/W | H'EC50_006C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC3 Control Register | SRC3_CONTROL | R/W | H'EC50_0070 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC3 Status Register | SRC3_STATUS | R | H'EC50_0074 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC3 Interrupt Enable Register 0 | SRC3_INT_ENABLE0 | R/W | H'EC50_0078 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC4in_BUSIF_MODE Register | SRC4in_BUSIF_MODE | R/W | H'EC50_0080 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC4out_BUSIF_MODE Register | SRC4out_BUSIF_MODE | R/W | H'EC50_0084 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC4_BUSIF_DALIGN Register | SRC4_BUSIF_DALIGN | R/W | H'EC50_0088 | H'7654_3210 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] H'0000_0010 [RZ/G2E] | 32 | √ | √ | √ | √ |
| SRC4_MODE Register | SRC4_MODE | R/W | H'EC50_008C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC4 Control Register | SRC4_CONTROL | R/W | H'EC50_0090 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC4 Status Register | SRC4_STATUS | R | H'EC50_0094 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC4 Interrupt Enable Register 0 | SRC4_INT_ENABLE0 | R/W | H'EC50_0098 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC5in_BUSIF_MODE Register | SRC5in_BUSIF_MODE | R/W | H'EC50_00A0 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC5out_BUSIF_MODE Register | SRC5out_BUSIF_MODE | R/W | H'EC50_00A4 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC5_BUSIF_DALIGN Register | SRC5_BUSIF_DALIGN | R/W | H'EC50_00A8 | H'0000_0010 | 32 | √ | √ | √ | √ |
| SRC5_MODE Register | SRC5_MODE | R/W | H'EC50_00AC | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC5 Control Register | SRC5_CONTROL | R/W | H'EC50_00B0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC5 Status Register | SRC5_STATUS | R | H'EC50_00B4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC5 Interrupt Enable Register 0 | SRC5_INT_ENABLE0 | R/W | H'EC50_00B8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC6in_BUSIF_MODE Register | SRC6in_BUSIF_MODE | R/W | H'EC50_00C0 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC6out_BUSIF_MODE Register | SRC6out_BUSIF_MODE | R/W | H'EC50_00C4 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC6_BUSIF_DALIGN Register | SRC6_BUSIF_DALIGN | R/W | H'EC50_00C8 | H'0000_0010 | 32 | √ | √ | √ | √ |
| SRC6_MODE Register | SRC6_MODE | R/W | H'EC50_00CC | H'0000_0000 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|----------------------------------|--------------------|-----|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SRC6 Control Register | SRC6_CONTROL | R/W | H'EC50_00D0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC6 Status Register | SRC6_STATUS | R | H'EC50_00D4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC6 Interrupt Enable Register 0 | SRC6_INT_ENABLE0 | R/W | H'EC50_00D8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC7in_BUSIF_MODE Register | SRC7in_BUSIF_MODE | R/W | H'EC50_00E0 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC7out_BUSIF_MODE Register | SRC7out_BUSIF_MODE | R/W | H'EC50_00E4 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC7_BUSIF_DALIGN Register | SRC7_BUSIF_DALIGN | R/W | H'EC50_00E8 | H'0000_0010 | 32 | √ | √ | √ | √ |
| SRC7_MODE Register | SRC7_MODE | R/W | H'EC50_00EC | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC7 Control Register | SRC7_CONTROL | R/W | H'EC50_00F0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC7 Status Register | SRC7_STATUS | R | H'EC50_00F4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC7 Interrupt Enable Register 0 | SRC7_INT_ENABLE0 | R/W | H'EC50_00F8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC8in_BUSIF_MODE Register | SRC8in_BUSIF_MODE | R/W | H'EC50_0100 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC8out_BUSIF_MODE Register | SRC8out_BUSIF_MODE | R/W | H'EC50_0104 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC8_BUSIF_DALIGN Register | SRC8_BUSIF_DALIGN | R/W | H'EC50_0108 | H'0000_0010 | 32 | √ | √ | √ | √ |
| SRC8_MODE Register | SRC8_MODE | R/W | H'EC50_010C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC8 Control Register | SRC8_CONTROL | R/W | H'EC50_0110 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC8 Status Register | SRC8_STATUS | R | H'EC50_0114 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC8 Interrupt Enable Register 0 | SRC8_INT_ENABLE0 | R/W | H'EC50_0118 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC9in_BUSIF_MODE Register | SRC9in_BUSIF_MODE | R/W | H'EC50_0120 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC9out_BUSIF_MODE Register | SRC9out_BUSIF_MODE | R/W | H'EC50_0124 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRC9_BUSIF_DALIGN Register | SRC9_BUSIF_DALIGN | R/W | H'EC50_0128 | H'0000_0010 | 32 | √ | √ | √ | √ |
| SRC9_MODE Register | SRC9_MODE | R/W | H'EC50_012C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC9 Control Register | SRC9_CONTROL | R/W | H'EC50_0130 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC9 Status Register | SRC9_STATUS | R | H'EC50_0134 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC9 Interrupt Enable Register 0 | SRC9_INT_ENABLE0 | R/W | H'EC50_0138 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CMD0 out_BUSIF_MODE Register | CMD0out_BUSIF_MODE | R/W | H'EC50_0184 | H'0000_0001 | 32 | √ | √ | √ | √ |
| CMD0_BUSIF_DALIGN Register | CMD0_BUSIF_DALIGN | R/W | H'EC50_0188 | H'7654_3210 | 32 | √ | √ | √ | √ |
| CMD0_ROUTE_SELECT Register | CMD0_ROUTE_SELECT | R/W | H'EC50_018C | H'0000_0000 | 32 | √ | √ | √ | √ |
| CMD0 Control Register | CMD0_CONTROL | R/W | H'EC50_0190 | H'0000_0000 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|--|----------------------------|-----------|---------------------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| CMD1 out_BUSIF_MODE Register | CMD1out_BUSIF_MODE | R/W | H'EC50_01A4 | H'0000_0001 | 32 | √ | √ | √ | √ |
| CMD1_BUSIF_DALIGN Register | CMD1_BUSIF_DALIGN | R/W | H'EC50_01A8 | H'7654_3210 | 32 | √ | √ | √ | √ |
| CMD1_ROUTE_SELECT Register | CMD1_ROUTE_SELECT | R/W | H'EC50_01AC | H'0000_0000 | 32 | √ | √ | √ | √ |
| CMD1 Control Register | CMD1_CONTROL | R/W | H'EC50_01B0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SCU_SYSTEM Status Register 0 | SCU_SYSTEM_STATUS 0 | R/W C1 | H'EC50_01C8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SCU_SYSTEM Interrupt Enable Register 0 | SCU_SYSTEM_INT_ENA BLE0 | R/W | H'EC50_01CC | H'0000_0000 | 32 | √ | √ | √ | √ |
| SCU_SYSTEM Status Register 1 | SCU_SYSTEM_STATUS 1 | R/W C1 | H'EC50_01D0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SCU_SYSTEM Interrupt Enable Register 1 | SCU_SYSTEM_INT_ENA BLE1 | R/W | H'EC50_01D4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC registers | — | — | SRC0_BASE: H'EC50_0200 | — | — | √ | √ | √ | √ |
| | | | SRC1_BASE: H'EC50_0240 | — | — | √ | √ | √ | √ |
| | | | SRC2_BASE: H'EC50_0280 | — | — | √ | √ | √ | √ |
| | | | SRC3_BASE: H'EC50_02C0 | — | — | √ | √ | √ | √ |
| | | | SRC4_BASE: H'EC50_0300 | — | — | √ | √ | √ | √ |
| | | | SRC5_BASE: H'EC50_0340 | — | — | √ | √ | √ | √ |
| | | | SRC6_BASE: H'EC50_0380 | — | — | √ | √ | √ | √ |
| | | | SRC7_BASE: H'EC50_03C0 | — | — | √ | √ | √ | √ |
| | | | SRC8_BASE: H'EC50_0400 | — | — | √ | √ | √ | √ |
| | | | SRC9_BASE: H'EC50_0440 | — | — | √ | √ | √ | √ |
| SRCm Software Reset Register | SRCm_SWRSR | R/W | SRCm_BASE + H'00 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRCm SRC Initialization Register | SRCm_SRCIR | R/W | SRCm_BASE + H'04 | H'0000_0001 | 32 | √ | √ | √ | √ |
| SRCm Audio Information Register | SRCm_ADINR | R/W | SRCm_BASE + H'14 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRCm IFS Control Register | SRCm_IFSCR | R/W | SRCm_BASE + H'1C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRCm IFS Value Setting Register | SRCm_IFSVR | R/W | SRCm_BASE + H'20 | H'0000_0000 | 32 | √ | √ | √ | √ |

| | | | | | | Second Generation RZ/G Series Products | | | |
|--|--------------|-----|--|--------------------|-------------|---|----------------------------|--------|--------|
| Name | Abbreviation | R/W | Address | Initial Value | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SRCm SRC Control Register | SRCm_SRCCR | R/W | SRCm_BASE + H'24 | H'0000_0000_* * | 32 | √ | √ | √ | √ |
| SRCm Buffer Size DATA RAM Setting Register | SRCm_BSDSR | R/W | SRCm_BASE + H'2C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRCm Buffer Size IJEC RAM Setting Register | SRCm_BSISR | R/W | SRCm_BASE + H'38 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTU registers | — | — | CTU00_BASE: H'EC50_0500 CTU01_BASE: H'EC50_0600 CTU02_BASE: H'EC50_0700 CTU03_BASE: H'EC50_0800 CTU10_BASE: H'EC50_0900 CTU11_BASE: H'EC50_0A00 CTU12_BASE: H'EC50_0B00 CTU13_BASE: H'EC50_0C00 | — | — | √ | √ | √ | √ |
| CTUn Software Reset Register | CTUn_SWRSR | R/W | CTUn_BASE + H'00 | H'0000_0001 | 32 | √ | √ | √ | √ |
| CTUn CTU Initialization Register | CTUn_CTUIR | R/W | CTUn_BASE + H'04 | H'0000_0001 | 32 | √ | √ | √ | √ |
| CTUn Audio Information Register | CTUn_ADINR | R/W | CTUn_BASE + H'08 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn CTU Pass Mode Register | CTUn_CPMR | R/W | CTUn_BASE + H'10 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Mode Register | CTUn_SCMDR | R/W | CTUn_BASE + H'14 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e00 Register | CTUn_SV00R | R/W | CTUn_BASE + H'18 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e01 Register | CTUn_SV01R | R/W | CTUn_BASE + H'1C | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e02 Register | CTUn_SV02R | R/W | CTUn_BASE + H'20 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e03 Register | CTUn_SV03R | R/W | CTUn_BASE + H'24 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e04 Register | CTUn_SV04R | R/W | CTUn_BASE + H'28 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e05 Register | CTUn_SV05R | R/W | CTUn_BASE + H'2C | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e06 Register | CTUn_SV06R | R/W | CTUn_BASE + H'30 | H'0000_0000 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|-------------------------------|--------------|-----|------------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| CTUn Scale Value e07 Register | CTUn_SV07R | R/W | CTUn_BASE + H'34 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e10 Register | CTUn_SV10R | R/W | CTUn_BASE + H'38 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e11 Register | CTUn_SV11R | R/W | CTUn_BASE + H'3C | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e12 Register | CTUn_SV12R | R/W | CTUn_BASE + H'40 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e13 Register | CTUn_SV13R | R/W | CTUn_BASE + H'44 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e14 Register | CTUn_SV14R | R/W | CTUn_BASE + H'48 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e15 Register | CTUn_SV15R | R/W | CTUn_BASE + H'4C | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e16 Register | CTUn_SV16R | R/W | CTUn_BASE + H'50 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e17 Register | CTUn_SV17R | R/W | CTUn_BASE + H'54 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e20 Register | CTUn_SV20R | R/W | CTUn_BASE + H'58 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e21 Register | CTUn_SV21R | R/W | CTUn_BASE + H'5C | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e22 Register | CTUn_SV22R | R/W | CTUn_BASE + H'60 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e23 Register | CTUn_SV23R | R/W | CTUn_BASE + H'64 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e24 Register | CTUn_SV24R | R/W | CTUn_BASE + H'68 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e25 Register | CTUn_SV25R | R/W | CTUn_BASE + H'6C | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e26 Register | CTUn_SV26R | R/W | CTUn_BASE + H'70 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e27 Register | CTUn_SV27R | R/W | CTUn_BASE + H'74 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e30 Register | CTUn_SV30R | R/W | CTUn_BASE + H'78 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e31 Register | CTUn_SV31R | R/W | CTUn_BASE + H'7C | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e32 Register | CTUn_SV32R | R/W | CTUn_BASE + H'80 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e33 Register | CTUn_SV33R | R/W | CTUn_BASE + H'84 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e34 Register | CTUn_SV34R | R/W | CTUn_BASE + H'88 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e35 Register | CTUn_SV35R | R/W | CTUn_BASE + H'8C | H'0000_0000 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---------------------------------------|--------------|-----|--|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| CTUn Scale Value e36 Register | CTUn_SV36R | R/W | CTUn_BASE + H'90 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CTUn Scale Value e37 Register | CTUn_SV37R | R/W | CTUn_BASE + H'94 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MIX registers | — | — | MIX0_BASE: H'EC50_0D00 MIX1_BASE: H'EC50_0D40 | — | — | √ | √ | √ | √ |
| MIXp Software Reset Register | MIXp_SWRSR | R/W | MIXp_BASE + H'00 | H'0000_0001 | 32 | √ | √ | √ | √ |
| MIXp MIX Initialization Register | MIXp_MIXIR | R/W | MIXp_BASE + H'04 | H'0000_0001 | 32 | √ | √ | √ | √ |
| MIXp Audio Information Register | MIXp_ADINR | R/W | MIXp_BASE + H'08 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MIXp MIX Mode Register | MIXp_MIXMR | R/W | MIXp_BASE + H'10 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MIXp MIX Volume Period Register | MIXp_MVPDR | R/W | MIXp_BASE + H'14 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MIXp MIX Decibel A Register | MIXp_MDBAR | R/W | MIXp_BASE + H'18 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MIXp MIX Decibel B Register | MIXp_MDBBR | R/W | MIXp_BASE + H'1C | H'0000_0000 | 32 | √ | √ | √ | √ |
| MIXp MIX Decibel C Register | MIXp_MDBCR | R/W | MIXp_BASE + H'20 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MIXp MIX Decibel D Register | MIXp_MDBDR | R/W | MIXp_BASE + H'24 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MIXp MIX Decibel Enable Register | MIXp_MDBER | R/W | MIXp_BASE + H'28 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MIXp MIX Status Register | MIXp_MIXSR | R | MIXp_BASE + H'2C | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVC registers | — | — | DVC0_BASE: H'EC50_0E00 DVC1_BASE: H'EC50_0F00 | — | — | √ | √ | √ | √ |
| DVCp Software Reset Register | DVCp_SWRSR | R/W | DVCp_BASE + H'00 | H'0000_0001 | 32 | √ | √ | √ | √ |
| DVCp DVU Initialization Register | DVCp_DVUIR | R/W | DVCp_BASE + H'04 | H'0000_0001 | 32 | √ | √ | √ | √ |
| DVCp Audio Information Register | DVCp_ADINR | R/W | DVCp_BASE + H'08 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp DVU Control Register | DVCp_DVUCR | R/W | DVCp_BASE + H'10 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp Zero Cross Mute Control Register | DVCp_ZCMCR | R/W | DVCp_BASE + H'14 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp Volume Ramp Control Register | DVCp_VRCTR | R/W | DVCp_BASE + H'18 | H'0000_0000 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|--------------------------------------|--------------|-----|------------------------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DVCp Volume Ramp Period Register | DVCp_VRPDR | R/W | DVCp_BASE + H'1C | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp Volume Ramp decibel Register | DVCp_VRDBR | R/W | DVCp_BASE + H'20 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp Volume Ramp wait time Register | DVCp_VRWTR | R/W | DVCp_BASE + H'24 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp Volume Value Setting 0 Register | DVCp_VOL0R | R/W | DVCp_BASE + H'28 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp Volume Value Setting 1 Register | DVCp_VOL1R | R/W | DVCp_BASE + H'2C | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp Volume Value Setting 2 Register | DVCp_VOL2R | R/W | DVCp_BASE + H'30 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp Volume Value Setting 3 Register | DVCp_VOL3R | R/W | DVCp_BASE + H'34 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp Volume Value Setting 4 Register | DVCp_VOL4R | R/W | DVCp_BASE + H'38 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp Volume Value Setting 5 Register | DVCp_VOL5R | R/W | DVCp_BASE + H'3C | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp Volume Value Setting 6 Register | DVCp_VOL6R | R/W | DVCp_BASE + H'40 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp Volume Value Setting 7 Register | DVCp_VOL7R | R/W | DVCp_BASE + H'44 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp DVU Enable Register | DVCp_DVUER | R/W | DVCp_BASE + H'48 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DVCp DVU Status Register | DVCp_DVUSR | R | DVCp_BASE + H'4C | H'0000_0008 | 32 | √ | √ | √ | √ |
| DVCp Interrupt Enable Register | DVCp_DVIER | R/W | DVCp_BASE + H'50 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC0 in Write Data Register | SRC0in_BUSIF | —/W | H'EC00_0000/ H'EC30_0000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC1 in Write Data Register | SRC1in_BUSIF | —/W | H'EC00_0400/ H'EC30_0400* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC2 in Write Data Register | SRC2in_BUSIF | —/W | H'EC00_0800/ H'EC30_0800* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC3 in Write Data Register | SRC3in_BUSIF | —/W | H'EC00_0C00/ H'EC30_0C00* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC4 in Write Data Register | SRC4in_BUSIF | —/W | H'EC00_1000/ H'EC30_1000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC5 in Write Data Register | SRC5in_BUSIF | —/W | H'EC00_1400/ H'EC30_1400* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC6 in Write Data Register | SRC6in_BUSIF | —/W | H'EC00_1800/ H'EC30_1800* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC7 in Write Data Register | SRC7in_BUSIF | —/W | H'EC00_1C00/ H'EC30_1C00* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC8 in Write Data Register | SRC8in_BUSIF | —/W | H'EC00_2000/ H'EC30_2000* | H'0000_0000 | 32 | √ | √ | √ | √ |

| | | | | | | Second Generation RZ/G Series Products | | | |
|-----------------------------|---------------|-----|------------------------------|---------------|-------------|---|----------------------------|--------|--------|
| Name | Abbreviation | R/W | Address | Initial Value | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SRC9 in Write Data Register | SRC9in_BUSIF | —/W | H'EC00_2400/ H'EC30_2400* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC0 out Read Data Register | SRC0out_BUSIF | R | H'EC00_4000/ H'EC30_4000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC1 out Read Data Register | SRC1out_BUSIF | R | H'EC00_4400/ H'EC30_4400* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC2 out Read Data Register | SRC2out_BUSIF | R | H'EC00_4800/ H'EC30_4800* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC3 out Read Data Register | SRC3out_BUSIF | R | H'EC00_4C00/ H'EC30_4C00* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC4 out Read Data Register | SRC4out_BUSIF | R | H'EC00_5000/ H'EC30_5000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC5 out Read Data Register | SRC5out_BUSIF | R | H'EC00_5400/ H'EC30_5400* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC6 out Read Data Register | SRC6out_BUSIF | R | H'EC00_5800/ H'EC30_5800* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC7 out Read Data Register | SRC7out_BUSIF | R | H'EC00_5C00/ H'EC30_5C00* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC8 out Read Data Register | SRC8out_BUSIF | R | H'EC00_6000/ H'EC30_6000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| SRC9 out Read Data Register | SRC9out_BUSIF | R | H'EC00_6400/ H'EC30_6400* | H'0000_0000 | 32 | √ | √ | √ | √ |
| CMD0 out Read Data Register | CMD0out_BUSIF | R | H'EC00_8000/ H'EC30_8000* | H'0000_0000 | 32 | √ | √ | √ | √ |
| CMD1 out Read Data Register | CMD1out_BUSIF | R | H'EC00_8400/ H'EC30_8400* | H'0000_0000 | 32 | √ | √ | √ | √ |

Notes: m = 0 to 9; n = 00, 01, 02, 03, 10, 11, 12, or 13; p = 0 or 1

* H'EC00_xxxx is the address for transferring by Audio-DMAC. H'EC30_xxxx is the address for transferring by Audio-DMACpp.

** This register is just what to show the initial value here, but, set a value according to 43.2.23 by all means when you use it.

43.1.5 Connected Module**Table 43.2 Connected modules**

| Module name | Connected module name | Function of connected module |
|------------------------------|------------------------------|---|
| Sampling Rate Converter Unit | APMU | Access the Register |
| | PFC | Select External pins |
| | CPG | Output Clocks |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |
| | INTC-AP | Control to interrupt |
| | SSIU | Serial Sound Interface Unit |
| | SSI | Serial Sound Interface |
| | ADG | Output Clocks for Audio module |
| | Audio-DMACpp | Control Direct Memory Access for Audio modules connected to the audio local bus |
| | Audio-DMAC | Control Direct Memory Access for Audio module |

43.2 Register Description

Legend for Register Description

| | |
|----------------|---|
| Initial value: | Register value after a reset. H'xxxx represents a hexadecimal number. Others are represented in binary numbers. |
| —: | Undefined value |
| R/W: | Readable/writable. The written value can be read. |
| R: | Read-only. The write value should always be 0. |
| R/WC0: | Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored. |
| R/WC1: | Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored. |
| W: | Write-only. Reading this bit is prohibited. When the bit is reserved, the write value should always be 0. |
| —/W: | Write-only. The read value is undefined. |

All access to registers is made in longword units.

43.2.1 SRCm(in/out)_BUSIF_MODE Register (SRCm(in/out)_BUSIF_MODE)

Note: m = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm (in/out)_BUSIF_MODE sets the initial setting for the bus interface.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|---------|---------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | sft_dir | sft_num | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | dma |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 | sft_dir | B'0 | R/W | srcm(in/out)_busif_shift_dir Selects the bit-shift direction for position adjustment of the valid SRCm(in/out)_BUSIF input and output data. 0: Shift to left. 1: Shift to right. |
| 19 to 16 | sft_num | H'0 | R/W | srcm(in/out)_busif_shift_num Selects the bit-shift count for position adjustment of the valid SRCm(in/out)_BUSIF input and output data. B'0000: 0 bit B'0001: 1 bit B'0010: 2 bits B'0011: 3 bits B'0100: 4 bits B'0101: 5 bits B'0110: 6 bits B'0111: 7 bits B'1000: 8 bits B'1001: 9 bits B'1010: 10 bits B'1011: 11 bits B'1100: 12 bits B'1101: 13 bits B'1110: 14 bits B'1111: 15 bits |
| 15 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | dma | B'1 | R/W | srcm(in/out)_busif_dma Selects the access type for SRCm(in/out)_BUSIF. 0: PIO access (setting prohibited) 1: DMA access Be sure to specify the DMA access. |

43.2.2 SRCm_BUSIF_DALIGN Register (SRCm_BUSIF_DALIGN)

Note: m = 0, 1, 3, or 4 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 m = 1 and 3 [RZ/G2E]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm_BUSIF_DALIGN determines the initial settings of the SRCm route.

| | | | | | | | | | | | | | | | | |
|----------------|----|--------|-----|-----|----|--------|-----|-----|----|--------|-----|-----|----|--------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | place7 | | | — | place6 | | | — | place5 | | | — | place4 | | |
| Initial value: | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | place3 | | | — | place2 | | | — | place1 | | | — | place0 | | |
| Initial value: | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 30 to 28 | place7 | H'7 | R/W | Changes the stream data order. These bits are used for the 8-channel setting. For the 6- or less-channel setting, the initial value should not be changed. The data order is changed before input to the SRC. Selects the input-side data to be output to place 7 on the output side. B'000: Data at input-side place 0 is sent to output-side place 7. B'001: Data at input-side place 1 is sent to output-side place 7. B'010: Data at input-side place 2 is sent to output-side place 7. B'011: Data at input-side place 3 is sent to output-side place 7. B'100: Data at input-side place 4 is sent to output-side place 7. B'101: Data at input-side place 5 is sent to output-side place 7. B'110: Data at input-side place 6 is sent to output-side place 7. B'111: Data at input-side place 7 is sent to output-side place 7. |
| 27 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 26 to 24 | place6 | H'6 | R/W | <p>Changes the stream data order. These bits are used for the 8-channel setting. For the 6- or less-channel setting, the initial value should not be changed.</p> <p>The data order is changed before input to the SRC.</p> <p>Selects the input-side data to be output to place 6 on the output side.</p> <p>B'000: Data at input-side place 0 is sent to output-side place 6. B'001: Data at input-side place 1 is sent to output-side place 6. B'010: Data at input-side place 2 is sent to output-side place 6. B'011: Data at input-side place 3 is sent to output-side place 6. B'100: Data at input-side place 4 is sent to output-side place 6. B'101: Data at input-side place 5 is sent to output-side place 6. B'110: Data at input-side place 6 is sent to output-side place 6. B'111: Data at input-side place 7 is sent to output-side place 6.</p> |
| 23 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 22 to 20 | place5 | H'5 | R/W | <p>Changes the stream data order. These bits are used for the 6- or more-channel setting. For the 4- or less-channel setting, the initial value should not be changed.</p> <p>The data order is changed before input to the SRC.</p> <p>Selects the input-side data to be output to place 5 on the output side.</p> <p>B'000: Data at input-side place 0 is sent to output-side place 5. B'001: Data at input-side place 1 is sent to output-side place 5. B'010: Data at input-side place 2 is sent to output-side place 5. B'011: Data at input-side place 3 is sent to output-side place 5. B'100: Data at input-side place 4 is sent to output-side place 5. B'101: Data at input-side place 5 is sent to output-side place 5. B'110: Data at input-side place 6 is sent to output-side place 5. B'111: Data at input-side place 7 is sent to output-side place 5.</p> |
| 19 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 18 to 16 | place4 | H'4 | R/W | <p>Changes the stream data order. These bits are used for the 6- or more-channel setting. For the 4- or less-channel setting, the initial value should not be changed.</p> <p>The data order is changed before input to the SRC.</p> <p>Selects the input-side data to be output to place 4 on the output side.</p> <p>B'000: Data at input-side place 0 is sent to output-side place 4. B'001: Data at input-side place 1 is sent to output-side place 4. B'010: Data at input-side place 2 is sent to output-side place 4. B'011: Data at input-side place 3 is sent to output-side place 4. B'100: Data at input-side place 4 is sent to output-side place 4. B'101: Data at input-side place 5 is sent to output-side place 4. B'110: Data at input-side place 6 is sent to output-side place 4. B'111: Data at input-side place 7 is sent to output-side place 4.</p> |
| 15 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 14 to 12 | place3 | H'3 | R/W | <p>Changes the stream data order. These bits are used for the 4- or more-channel setting. For the 2- or less-channel setting, the initial value should not be changed.</p> <p>The data order is changed before input to the SRC.</p> <p>Selects the input-side data to be output to place 3 on the output side.</p> <p>B'000: Data at input-side place 0 is sent to output-side place 3. B'001: Data at input-side place 1 is sent to output-side place 3. B'010: Data at input-side place 2 is sent to output-side place 3. B'011: Data at input-side place 3 is sent to output-side place 3. B'100: Data at input-side place 4 is sent to output-side place 3. B'101: Data at input-side place 5 is sent to output-side place 3. B'110: Data at input-side place 6 is sent to output-side place 3. B'111: Data at input-side place 7 is sent to output-side place 3.</p> |
| 11 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 10 to 8 | place2 | H'2 | R/W | <p>Changes the stream data order. These bits are used for the 4- or more-channel setting. For the 2- or less-channel setting, the initial value should not be changed.</p> <p>The data order is changed before input to the SRC.</p> <p>Selects the input-side data to be output to place 2 on the output side.</p> <p>B'000: Data at input-side place 0 is sent to output-side place 2. B'001: Data at input-side place 1 is sent to output-side place 2. B'010: Data at input-side place 2 is sent to output-side place 2. B'011: Data at input-side place 3 is sent to output-side place 2. B'100: Data at input-side place 4 is sent to output-side place 2. B'101: Data at input-side place 5 is sent to output-side place 2. B'110: Data at input-side place 6 is sent to output-side place 2. B'111: Data at input-side place 7 is sent to output-side place 2.</p> |
| 7 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 6 to 4 | place1 | H'1 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed before input to the SRC.</p> <p>Selects the input-side data to be output to place 1 on the output side.</p> <p>B'000: Data at input-side place 0 is sent to output-side place 1. B'001: Data at input-side place 1 is sent to output-side place 1. B'010: Data at input-side place 2 is sent to output-side place 1. B'011: Data at input-side place 3 is sent to output-side place 1. B'100: Data at input-side place 4 is sent to output-side place 1. B'101: Data at input-side place 5 is sent to output-side place 1. B'110: Data at input-side place 6 is sent to output-side place 1. B'111: Data at input-side place 7 is sent to output-side place 1.</p> |
| 3 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 2 to 0 | place0 | H'0 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed before input to the SRC.</p> <p>Selects the input-side data to be output to place 0 on the output side.</p> <p>B'000: Data at input-side place 0 is sent to output-side place 0.</p> <p>B'001: Data at input-side place 1 is sent to output-side place 0.</p> <p>B'010: Data at input-side place 2 is sent to output-side place 0.</p> <p>B'011: Data at input-side place 3 is sent to output-side place 0.</p> <p>B'100: Data at input-side place 4 is sent to output-side place 0.</p> <p>B'101: Data at input-side place 5 is sent to output-side place 0.</p> <p>B'110: Data at input-side place 6 is sent to output-side place 0.</p> <p>B'111: Data at input-side place 7 is sent to output-side place 0.</p> |

43.2.3 SRCn_BUSIF_DALIGN Register (SRCn_BUSIF_DALIGN)

Note: n = 2, 5, 6, 7, 8, or 9 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 n = 0, 2, 4, 5, 6, 7, 8, or 9 [RZ/G2E]

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCn_BUSIF_DALIGN determines the initial settings of the SRCn route.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|--------|----|----|----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | place1 | — | — | — | place0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | place1 | B'1 | R/W | Changes the stream data order. The data order is changed before input to the SRC. Selects the input-side data to be output to place 1 on the output side. 0: Data at input-side place 0 is sent to output-side place 1. 1: Data at input-side place 1 is sent to output-side place 1. |
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | place0 | B'0 | R/W | Changes the stream data order. The data order is changed before input to the SRC. Selects the input-side data to be output to place 0 on the output side. 0: Data at input-side place 0 is sent to output-side place 0. 1: Data at input-side place 1 is sent to output-side place 0. |

43.2.4 SRCm_MODE Register (SRCm_MODE)

Note: m = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm_MODE determines the initial settings of the SRCm route.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----------|---------|----|----|----|----|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | sync_out | sync_in | — | — | — | — | — | — | — | uf_data |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | src |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 26 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25 | sync_out | B'0 | R/W | srcmout_sync Selects how to treat data in the SRC output buffer. 0: Asynchronous SRC 1: Synchronous SRC |
| 24 | sync_in | B'0 | R/W | srcmin_sync Selects how to treat data in the SRC input buffer. 0: Asynchronous SRC 1: Synchronous SRC |
| 23 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | uf_data | B'0 | R/W | srcm_uf_data_sel Selects how to treat data when an underflow occurs in the SRC input buffer. 0: Data before the underflow occurs is output. 1: All 0s are output. |
| 15 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | src | B'0 | R/W | srcm_src Selects whether to use SRCm. 0: SRCm is not used. 1: SRCm is used. |

- Notes:
1. Set one of sync_in and sync_out bits, when SRCm is used in synchronous mode by setting SRCMD bit in SRCm_SRCCR register.
 2. If sync_out is set in synchronous mode, data should directly transmit to memory. (Data should not transmit throughout CMD.)

43.2.5 SRCm Control Register (SRCm_CONTROL)

Note: m = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm_CONTROL starts or stops transfer.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|-----------|----|----|----|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | start_out | — | — | — | start_in |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | start_out | B'0 | R/W | SRCmout_start_flag Starts or stops transfer via SRCm. 0: Stops transfer. 1: Starts transfer. |
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | start_in | B'0 | R/W | SRCmin_start_flag Starts or stops transfer via SRCm. 0: Stops transfer. 1: Starts transfer. |

Note: When CMD is used with route SRC0, SRC1, SRC2, SRC3, SRC4, SRC5, SRC6 or SRC9 SRCmout_start_flag should be set to 0 to stop transfer. When CMD is not used with route SRC0, SRC1, SRC2, SRC3, SRC4, SRC5, SRC6 or SRC9, or when route SRC7 or SRC8 is used, SRCmin_start_flag and SRCmout_start_flag should be set simultaneously.

43.2.6 SRCm Status Register (SRCm_STATUS)

Note: m = 0 to 6, 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm_STATUS indicates the state of each SRC. When a register bit is set, the corresponding interrupt signal is output. However, if interrupt outputs are masked by the SRCm interrupt enable register, interrupt signals are not output.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|---------|---------|----|----|---------|---------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | dvc1 | — | — | — | dvc0 | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | uf_src0 | of_src0 | — | — | of_src1 | uf_src1 | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | dvc1 | B'0 | R | dvc1 Indicates the state of the DVC1_DVUSR register collectively. The state information is used only for the interrupt signal that is enabled by DVC1_DVIER. |
| 27 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | dvc0 | B'0 | R | dvc0 Indicates the state of the DVC0_DVUSR register collectively. The state information is used only for the interrupt signal that is enabled by DVC0_DVIER. |
| 23 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 | uf_src0 | B'0 | R | under_flow_srcout Indicates the state of the output buffer of SRCout in the SCU_SYSTEM_STATUS1 register. The state information is used when the interrupt signal is enabled by SCU_SYSTEM_INT_ENABLE1. |
| 12 | of_src0 | B'0 | R | over_flow_srcout Indicates the state of the output buffer of SRCout in the SCU_SYSTEM_STATUS0 register. The state information is used when the interrupt signal is enabled by SCU_SYSTEM_INT_ENABLE0. |
| 11, 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 9 | of_srcin | B'0 | R | over_flow_srcin Indicates the state of the output buffer of SRCin in the SCU_SYSTEM_STATUS1 register. The state information is used when the interrupt signal is enabled by SCU_SYSTEM_INT_ENABLE1. |
| 8 | uf_srcin | B'0 | R | under_flow_srcin Indicates the state of the output buffer of SRCin in the SCU_SYSTEM_STATUS0 register. The state information is used when the interrupt signal is enabled by SCU_SYSTEM_INT_ENABLE0. |
| 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

43.2.7 SRCn Status Register (SRCn_STATUS)

Note: n = 7 and 8

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCn_STATUS indicates the state of each SRC. When a register bit is set, the corresponding interrupt signal is output. However, if interrupt outputs are masked by the SRCn interrupt enable register, interrupt signals are not output.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|---------|---------|----|----|---------|---------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | uf_srco | of_srco | — | — | of_src1 | uf_src1 | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 | uf_srco | B'0 | R | under_flow_srcout Indicates the state of the output buffer of SRCout in the SCU_SYSTEM_STATUS1 register. The state information is used when the interrupt signal is enabled by SCU_SYSTEM_INT_ENABLE1. |
| 12 | of_srco | B'0 | R | over_flow_srcout Indicates the state of the output buffer of SRCout in the SCU_SYSTEM_STATUS0 register. The state information is used when the interrupt signal is enabled by SCU_SYSTEM_INT_ENABLE0. |
| 11, 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | of_src1 | B'0 | R | over_flow_srcin Indicates the state of the output buffer of SRCin in the SCU_SYSTEM_STATUS1 register. The state information is used when the interrupt signal is enabled by SCU_SYSTEM_INT_ENABLE1. |
| 8 | uf_src1 | B'0 | R | under_flow_srcin Indicates the state of the output buffer of SRCin in the SCU_SYSTEM_STATUS0 register. The state information is used when the interrupt signal is enabled by SCU_SYSTEM_INT_ENABLE0. |
| 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

43.2.8 SRCm Interrupt Enable Register 0 (SRCm_INT_ENABLE0)

Note: m = 0 to 6, 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm_INT_ENABLE0 enables or disables output of interrupts corresponding to the states indicated in the SRCm status register.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|------------|------------|----|----|-------------|-------------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | dvc1_ie | — | — | — | dvc0_ie | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | uf_srco_ie | of_srco_ie | — | — | of_src_i_ie | uf_src_i_ie | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R | R | R/W | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | dvc1_ie | B'0 | R/W | dvc1_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 27 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | dvc0_ie | B'0 | R/W | dvc0_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 23 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 | uf_srco_ie | B'0 | R/W | under_flow_srcout_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 12 | of_srco_ie | B'0 | R/W | over_flow_srcout_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 11, 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | of_src_i_ie | B'0 | R/W | over_flow_srcin_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 8 | uf_srcie | B'0 | R/W | under_flow_srcin_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

43.2.9 SRCn Interrupt Enable Register 0 (SRCn_INT_ENABLE0)

Note: n = 7 and 8

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCn_INT_ENABLE0 enables or disables output of interrupts corresponding to the states indicated in the SRCn status register.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|------------|------------|----|----|----------|----------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | uf_srco_ie | of_srco_ie | — | — | of_srcie | uf_srcie | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R | R | R/W | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 | uf_srco_ie | B'0 | R/W | under_flow_srcout_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 12 | of_srco_ie | B'0 | R/W | over_flow_srcout_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 11, 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | of_srcie | B'0 | R/W | over_flow_srcin_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 8 | uf_srcie | B'0 | R/W | under_flow_srcin_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

43.2.10 CMDn out_BUSIF_MODE Register (CMDn out_BUSIF_MODE)

Note: n = 0, 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CMDn out_BUSIF_MODE sets the initial setting for the bus interface

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|---------|---------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | sft_dir | sft_num | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | dma |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 | sft_dir | B'0 | R/W | cmdnout_busif_shift_dir Selects the bit-shift direction for position adjustment of the valid CMDnout_BUSIF input and output data. 0: Shift to left. 1: Shift to right. |
| 19 to 16 | sft_num | H'0 | R/W | cmdnout_busif_shift_num Selects the bit-shift count for position adjustment of the valid CMDnout_BUSIF input and output data. B'0000: 0 bit B'0001: 1 bit B'0010: 2 bits B'0011: 3 bits B'0100: 4 bits B'0101: 5 bits B'0110: 6 bits B'0111: 7 bits B'1000: 8 bits B'1001: 9 bits B'1010: 10 bits B'1011: 11 bits B'1100: 12 bits B'1101: 13 bits B'1110: 14 bits B'1111: 15 bits |
| 15 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------------|-----------------|----------------------|------------|--|
| 0 | dma | B'1 | R/W | CMDnout_busif_dma Selects the access type for CMDnout_BUSIF. 0: PIO access (setting prohibited) 1: DMA access Be sure to specify the DMA access. |

43.2.11 CMDn_BUSIF_DALIGN Register (CMDn_BUSIF_DALIGN)

Note: n = 0, 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CMDn_BUSIF_DALIGN determines the initial settings of the CMDn route.

| | | | | | | | | | | | | | | | | |
|----------------|----|--------|-----|-----|----|--------|-----|-----|----|--------|-----|-----|----|--------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | place7 | | | — | place6 | | | — | place5 | | | — | place4 | | |
| Initial value: | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | place3 | | | — | place2 | | | — | place1 | | | — | place0 | | |
| Initial value: | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 30 to 28 | place7 | H'7 | R/W | Changes the stream data order. These bits are used for the 8-channel setting. For the 6- or less-channel setting, the initial value should not be changed. The data order is changed after output from the CMD. Selects the input-side data to be output to place 7 on the output side. B'000: Data at input-side place0 is sent to output-side place7. B'001: Data at input-side place1 is sent to output-side place7. B'010: Data at input-side place2 is sent to output-side place7. B'011: Data at input-side place3 is sent to output-side place7. B'100: Data at input-side place4 is sent to output-side place7. B'101: Data at input-side place5 is sent to output-side place7. B'110: Data at input-side place6 is sent to output-side place7. B'111: Data at input-side place7 is sent to output-side place7. |
| 27 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 26 to 24 | place6 | H'6 | R/W | Changes the stream data order. These bits are used for the 8-channel setting. For the 6- or less-channel setting, the initial value should not be changed. The data order is changed after output from the CMD. Selects the input-side data to be output to place 6 on the output side. B'000: Data at input-side place0 is sent to output-side place6. B'001: Data at input-side place1 is sent to output-side place6. B'010: Data at input-side place2 is sent to output-side place6. B'011: Data at input-side place3 is sent to output-side place6. B'100: Data at input-side place4 is sent to output-side place6. B'101: Data at input-side place5 is sent to output-side place6. B'110: Data at input-side place6 is sent to output-side place6. B'111: Data at input-side place7 is sent to output-side place6. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 23 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 22 to 20 | place5 | H'5 | R/W | Changes the stream data order. These bits are used for the 6- or more-channel setting. For the 4- or less-channel setting, the initial value should not be changed. The data order is changed after output from the CMD. Selects the input-side data to be output to place 5 on the output side. B'000: Data at input-side place0 is sent to output-side place5. B'001: Data at input-side place1 is sent to output-side place5. B'010: Data at input-side place2 is sent to output-side place5. B'011: Data at input-side place3 is sent to output-side place5. B'100: Data at input-side place4 is sent to output-side place5. B'101: Data at input-side place5 is sent to output-side place5. B'110: Data at input-side place6 is sent to output-side place5. B'111: Data at input-side place7 is sent to output-side place5. |
| 19 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 18 to 16 | place4 | H'4 | R/W | Changes the stream data order. These bits are used for the 6- or more-channel setting. For the 4- or less-channel setting, the initial value should not be changed. The data order is changed after output from the CMD. Selects the input-side data to be output to place 4 on the output side. B'000: Data at input-side place0 is sent to output-side place4. B'001: Data at input-side place1 is sent to output-side place4. B'010: Data at input-side place2 is sent to output-side place4. B'011: Data at input-side place3 is sent to output-side place4. B'100: Data at input-side place4 is sent to output-side place4. B'101: Data at input-side place5 is sent to output-side place4. B'110: Data at input-side place6 is sent to output-side place4. B'111: Data at input-side place7 is sent to output-side place4. |
| 15 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 14 to 12 | place3 | H'3 | R/W | Changes the stream data order. These bits are used for the 4- or more-channel setting. For the 2- or less-channel setting, the initial value should not be changed. The data order is changed after output from the CMD. Selects the input-side data to be output to place 3 on the output side. B'000: Data at input-side place0 is sent to output-side place3. B'001: Data at input-side place1 is sent to output-side place3. B'010: Data at input-side place2 is sent to output-side place3. B'011: Data at input-side place3 is sent to output-side place3. B'100: Data at input-side place4 is sent to output-side place3. B'101: Data at input-side place5 is sent to output-side place3. B'110: Data at input-side place6 is sent to output-side place3. B'111: Data at input-side place7 is sent to output-side place3. |
| 11 | — | 0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 10 to 8 | place2 | H'2 | R/W | <p>Changes the stream data order. These bits are used for the 4- or more-channel setting. For the 2- or less-channel setting, the initial value should not be changed.</p> <p>The data order is changed after output from the CMD.</p> <p>Selects the input-side data to be output to place 2 on the output side.</p> <p>B'000: Data at input-side place0 is sent to output-side place2. B'001: Data at input-side place1 is sent to output-side place2. B'010: Data at input-side place2 is sent to output-side place2. B'011: Data at input-side place3 is sent to output-side place2. B'100: Data at input-side place4 is sent to output-side place2. B'101: Data at input-side place5 is sent to output-side place2. B'110: Data at input-side place6 is sent to output-side place2. B'111: Data at input-side place7 is sent to output-side place2.</p> |
| 7 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 6 to 4 | place1 | H'1 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed after output from the CMD.</p> <p>Selects the input-side data to be output to place 1 on the output side.</p> <p>B'000: Data at input-side place0 is sent to output-side place1. B'001: Data at input-side place1 is sent to output-side place1. B'010: Data at input-side place2 is sent to output-side place1. B'011: Data at input-side place3 is sent to output-side place1. B'100: Data at input-side place4 is sent to output-side place1. B'101: Data at input-side place5 is sent to output-side place1. B'110: Data at input-side place6 is sent to output-side place1. B'111: Data at input-side place7 is sent to output-side place1.</p> |
| 3 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 2 to 0 | place0 | H'0 | R/W | <p>Changes the stream data order.</p> <p>The data order is changed after output from the CMD.</p> <p>Selects the input-side data to be output to place 0 on the output side.</p> <p>B'000: Data at input-side place0 is sent to output-side place0. B'001: Data at input-side place1 is sent to output-side place0. B'010: Data at input-side place2 is sent to output-side place0. B'011: Data at input-side place3 is sent to output-side place0. B'100: Data at input-side place4 is sent to output-side place0. B'101: Data at input-side place5 is sent to output-side place0. B'110: Data at input-side place6 is sent to output-side place0. B'111: Data at input-side place7 is sent to output-side place0.</p> |

43.2.12 CMDn_ROUTE_SELECT Register (CMDn_ROUTE_SELECT)

Note: n = 0, 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[, RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

Function: CMDn_ROUTE_SELECT selects the sound route for each CMD (CTU, MIX, and DVC). Refer to figures Figure 43.1 and Figure 43.11 for the sound routes.

| | | | | | | | | | | | | | | | | |
|----------------|------------|----|----|------------|----------------|-----|-----|------------|----|----|----|----|----|----------|-----|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | synco_cmd[3:0] | | | | — | — | — | — | — | cmd_case | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | cmdin_ctu1 | — | — | cmdin_ctu0 | — | — | — | cmdin_ctu3 | — | — | — | — | — | — | — | cmdin_ctu2 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R/W | R | R | R | R/W | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 28 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 | synco_cmd[3] | B'0 | R/W | synco_cmd[3] Selects how to treat the CMD output data timing. Refer to Figure 43.11 CMD Block Diagram. 0: CMD output data timing 1: Synchronous SRC output data timing (Only when sync_out = 1 in SRCm_MODE register) |
| 26 to 24 | synco_cmd[2:0] | H'0 | R/W | synco_cmd[2:0] Selects SRC for CMD output data timing. If synco_cmd[3] = 0, these bits are ignored. B'000: SRC0 B'001: SRC1 B'010: SRC2 B'011: SRC3 B'100: SRC4 B'101: SRC5 B'110: SRC6 B'111: SRC9 |
| 23 to 19 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 18 to 16 | cmd_case | H'0 | R/W | <p>cmdn_case_sel</p> <p>Selects the route in CMDn.</p> <p>B'000: Input audio data0 to 3 → CTU → MIX → DVC route is used.</p> <p>B'001: Input audio data0 (from SRC3 or SRC6) → DVC route is used.</p> <p>B'010: Input audio data1 (from SRC4 or SRC9) → DVC route is used.</p> <p>B'011: Input audio data2(from SRC0 or SRC1) → DVC route is used.</p> <p>B'100: Input audio data3 (from SRC2 or SRC5) → DVC route is used.</p> <p>Others: Setting prohibited.</p> |
| 15 | cmdin_ctu1 | B'0 | R/W | <p>cmdnin_ctu1_sel</p> <p>Selects SRC for Input audio data1 of CMDn.</p> <p>0: SRC4 route is used.</p> <p>1: SRC9 route is used.</p> |
| 14 to 13 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 12 | cmdin_ctu0 | B'0 | R/W | <p>cmdnin_ctu0_sel</p> <p>Selects SRC for Input audio data0 of CMDn.</p> <p>0: SRC3 route is used.</p> <p>1: SRC6 route is used.</p> |
| 11 to 9 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 8 | cmdin_ctu3 | B'0 | R/W | <p>cmdnin_ctu3_sel</p> <p>Selects SRC for Input audio data3 of CMDn.</p> <p>0: SRC2 route is used.</p> <p>1: SRC5 route is used.</p> |
| 7 to 1 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 0 | cmdin_ctu2 | B'0 | R/W | <p>cmdnin_ctu2_sel</p> <p>Selects SRC for Input audio data2 of CMDn.</p> <p>0: SRC0 route is used.</p> <p>1: SRC1 route is used.</p> |

43.2.13 CMDn Control Register (CMDn_CONTROL)

Note: n = 0, 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CMDn_CONTROL starts or stops transfer.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|-----------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | start_out | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | start_out | B'0 | R/W | CMDnout_start_flag Starts or stops transfer via CMDn. 0: Stops transfer. 1: Starts transfer. |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

43.2.14 SCU_SYSTEM Status Register 0 (SCU_SYSTEM_STATUS0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SCU_SYSTEM_STATUS0 indicates the internal buffer state. When a register bit is set, the corresponding interrupt signal is output. However, if interrupt outputs are masked by the SYSTEM interrupt enable register 0, interrupt signals are not output.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----------|----------|----|----|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | of_cmd1o | of_cmd0o | — | — | of_src9o | of_src8o | of_src7o | of_src6o | of_src5o | of_src4o | of_src3o | of_src2o | of_src1o | of_src0o |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/WC1 | R/WC1 | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | uf_src9i | uf_src8i | uf_src7i | uf_src6i | uf_src5i | uf_src4i | uf_src3i | uf_src2i | uf_src1i | uf_src0i |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-------|---|
| 31, 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 | of_cmd1o | B'0 | R/WC1 | over_flow_cmd1out Indicates the state of the CMD1out output buffer. 0: Normal operation 1: An overflow has occurred. |
| 28 | of_cmd0o | B'0 | R/WC1 | over_flow_cmd0out Indicates the state of the CMD0out output buffer. 0: Normal operation 1: An overflow has occurred. |
| 27, 26 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25 | of_src9o | B'0 | R/WC1 | over_flow_src9out Indicates the state of the SRC9out output buffer. 0: Normal operation 1: An overflow has occurred. |
| 24 | of_src8o | B'0 | R/WC1 | over_flow_src8out Indicates the state of the SRC8out output buffer. 0: Normal operation 1: An overflow has occurred. |
| 23 | of_src7o | B'0 | R/WC1 | over_flow_src7out Indicates the state of the SRC7out output buffer. 0: Normal operation 1: An overflow has occurred. |
| 22 | of_src6o | B'0 | R/WC1 | over_flow_src6out Indicates the state of the SRC6out output buffer. 0: Normal operation 1: An overflow has occurred. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|---|
| 21 | of_src5o | B'0 | R/WC1 | over_flow_src5out Indicates the state of the SRC5out output buffer. 0: Normal operation 1: An overflow has occurred. |
| 20 | of_src4o | B'0 | R/WC1 | over_flow_src4out Indicates the state of the SRC4out output buffer. 0: Normal operation 1: An overflow has occurred. |
| 19 | of_src3o | B'0 | R/WC1 | over_flow_src3out Indicates the state of the SRC3out output buffer. 0: Normal operation 1: An overflow has occurred. |
| 18 | of_src2o | B'0 | R/WC1 | over_flow_src2out Indicates the state of the SRC2out output buffer. 0: Normal operation 1: An overflow has occurred. |
| 17 | of_src1o | B'0 | R/WC1 | over_flow_src1out Indicates the state of the SRC1out output buffer. 0: Normal operation 1: An overflow has occurred. |
| 16 | of_src0o | B'0 | R/WC1 | over_flow_src0out Indicates the state of the SRC0out output buffer. 0: Normal operation 1: An overflow has occurred. |
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | uf_src9i | B'0 | R/WC1 | under_flow_src9in Indicates the state of the SRC9in input buffer. 0: Normal operation 1: An underflow has occurred. |
| 8 | uf_src8i | B'0 | R/WC1 | under_flow_src8in Indicates the state of the SRC8in input buffer. 0: Normal operation 1: An underflow has occurred. |
| 7 | uf_src7i | B'0 | R/WC1 | under_flow_src7in Indicates the state of the SRC7in input buffer. 0: Normal operation 1: An underflow has occurred. |
| 6 | uf_src6i | B'0 | R/WC1 | under_flow_src6in Indicates the state of the SRC6in input buffer. 0: Normal operation 1: An underflow has occurred. |
| 5 | uf_src5i | B'0 | R/WC1 | under_flow_src5in Indicates the state of the SRC5in input buffer. 0: Normal operation 1: An underflow has occurred. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 4 | uf_src4i | B'0 | R/WC1 | under_flow_src4in Indicates the state of the SRC4in input buffer. 0: Normal operation 1: An underflow has occurred. |
| 3 | uf_src3i | B'0 | R/WC1 | under_flow_src3in Indicates the state of the SRC3in input buffer. 0: Normal operation 1: An underflow has occurred. |
| 2 | uf_src2i | B'0 | R/WC1 | under_flow_src2in Indicates the state of the SRC2in input buffer. 0: Normal operation 1: An underflow has occurred. |
| 1 | uf_src1i | B'0 | R/WC1 | under_flow_src1in Indicates the state of the SRC1in input buffer. 0: Normal operation 1: An underflow has occurred. |
| 0 | uf_src0i | B'0 | R/WC1 | under_flow_src0in Indicates the state of the SRC0in input buffer. 0: Normal operation 1: An underflow has occurred. |

43.2.15 SCU_SYSTEM Interrupt Enable Register 0 (SCU_SYSTEM_INT_ENABLE0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SCU_SYSTEM_INT_ENABLE0 enables or disables output of interrupts corresponding to the states indicated in the SCU_SYSTEM status register 0.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | of_src9 o_ie | of_src8 o_ie | of_src7 o_ie | of_src6 o_ie | of_src5 o_ie | of_src4 o_ie | of_src3 o_ie | of_src2 o_ie | of_src1 o_ie | of_src0 o_ie |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | uf_src9i _ie | uf_src8i _ie | uf_src7i _ie | uf_src6i _ie | uf_src5i _ie | uf_src4i _ie | uf_src3i _ie | uf_src2i _ie | uf_src1i _ie | uf_src0i _ie |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-------------|---------------|-----|---|
| 31, 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 28 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 27, 26 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25 | of_src9o_ie | B'0 | R/W | over_flow_src9out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 24 | of_src8o_ie | B'0 | R/W | over_flow_src8out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 23 | of_src7o_ie | B'0 | R/W | over_flow_src7out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 22 | of_src6o_ie | B'0 | R/W | over_flow_src6out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 21 | of_src5o_ie | B'0 | R/W | over_flow_src5out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 20 | of_src4o_ie | B'0 | R/W | over_flow_src4out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 19 | of_src3o_ie | B'0 | R/W | over_flow_src3out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 18 | of_src2o_ie | B'0 | R/W | over_flow_src2out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 17 | of_src1o_ie | B'0 | R/W | over_flow_src1out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 16 | of_src0o_ie | B'0 | R/W | over_flow_src0out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | uf_src9i_ie | B'0 | R/W | under_flow_src9in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 8 | uf_src8i_ie | B'0 | R/W | under_flow_src8in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 7 | uf_src7i_ie | B'0 | R/W | under_flow_src7in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 6 | uf_src6i_ie | B'0 | R/W | under_flow_src6in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 5 | uf_src5i_ie | B'0 | R/W | under_flow_src5in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 4 | uf_src4i_ie | B'0 | R/W | under_flow_src4in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 3 | uf_src3i_ie | B'0 | R/W | under_flow_src3in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 2 | uf_src2i_ie | B'0 | R/W | under_flow_src2in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 1 | uf_src1i_ie | B'0 | R/W | under_flow_src1in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 0 | uf_src0i_ie | B'0 | R/W | under_flow_src0in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

43.2.16 SCU_SYSTEM Status Register 1 (SCU_SYSTEM_STATUS1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SCU_SYSTEM_STATUS1 indicates the internal buffer state in synchronous mode. When a register bit is set, the corresponding interrupt signal is output. However, if interrupt outputs are masked by the SYSTEM interrupt enable register 1, interrupt signals are not output.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | uf_src9o | uf_src8o | uf_src7o | uf_src6o | uf_src5o | uf_src4o | uf_src3o | uf_src2o | uf_src1o | uf_src0o |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | of_src9i | of_src8i | of_src7i | of_src6i | of_src5i | of_src4i | of_src3i | of_src2i | of_src1i | of_src0i |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|---|
| 31 to 26 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25 | uf_src9o | B'0 | R/WC1 | under_flow_src9out Indicates the state of the SRC9out output buffer. 0: Normal operation 1: An underflow has occurred. |
| 24 | uf_src8o | B'0 | R/WC1 | under_flow_src8out Indicates the state of the SRC8out output buffer. 0: Normal operation 1: An underflow has occurred. |
| 23 | uf_src7o | B'0 | R/WC1 | under_flow_src7out Indicates the state of the SRC7out output buffer. 0: Normal operation 1: An underflow has occurred. |
| 22 | uf_src6o | B'0 | R/WC1 | under_flow_src6out Indicates the state of the SRC6out output buffer. 0: Normal operation 1: An underflow has occurred. |
| 21 | uf_src5o | B'0 | R/WC1 | under_flow_src5out Indicates the state of the SRC5out output buffer. 0: Normal operation 1: An underflow has occurred. |
| 20 | uf_src4o | B'0 | R/WC1 | under_flow_src4out Indicates the state of the SRC4out output buffer. 0: Normal operation 1: An underflow has occurred. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|---|
| 19 | uf_src3o | B'0 | R/WC1 | under_flow_src3out Indicates the state of the SRC3out output buffer. 0: Normal operation 1: An underflow has occurred. |
| 18 | uf_src2o | B'0 | R/WC1 | under_flow_src2out Indicates the state of the SRC2out output buffer. 0: Normal operation 1: An underflow has occurred. |
| 17 | uf_src1o | B'0 | R/WC1 | under_flow_src1out Indicates the state of the SRC1out output buffer. 0: Normal operation 1: An underflow has occurred. |
| 16 | uf_src0o | B'0 | R/WC1 | under_flow_src0out Indicates the state of the SRC0out output buffer. 0: Normal operation 1: An underflow has occurred. |
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | of_src9i | B'0 | R/WC1 | over_flow_src9in Indicates the state of the SRC9in input buffer. 0: Normal operation 1: An overflow has occurred. |
| 8 | of_src8i | B'0 | R/WC1 | over_flow_src8in Indicates the state of the SRC8in input buffer. 0: Normal operation 1: An overflow has occurred. |
| 7 | of_src7i | B'0 | R/WC1 | over_flow_src7in Indicates the state of the SRC7in input buffer. 0: Normal operation 1: An overflow has occurred. |
| 6 | of_src6i | B'0 | R/WC1 | over_flow_src6in Indicates the state of the SRC6in input buffer. 0: Normal operation 1: An overflow has occurred. |
| 5 | of_src5i | B'0 | R/WC1 | over_flow_src5in Indicates the state of the SRC5in input buffer. 0: Normal operation 1: An overflow has occurred. |
| 4 | of_src4i | B'0 | R/WC1 | over_flow_src4in Indicates the state of the SRC4in input buffer. 0: Normal operation 1: An overflow has occurred. |
| 3 | of_src3i | B'0 | R/WC1 | over_flow_src3in Indicates the state of the SRC3in input buffer. 0: Normal operation 1: An overflow has occurred. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 2 | of_src2i | B'0 | R/WC1 | over_flow_src2in Indicates the state of the SRC2in input buffer. 0: Normal operation 1: An overflow has occurred. |
| 1 | of_src1i | B'0 | R/WC1 | over_flow_src1in Indicates the state of the SRC1in input buffer. 0: Normal operation 1: An overflow has occurred. |
| 0 | of_src0i | B'0 | R/WC1 | over_flow_src0in Indicates the state of the SRC0in input buffer. 0: Normal operation 1: An overflow has occurred. |

43.2.17 SCU_SYSTEM Interrupt Enable Register 1 (SCU_SYSTEM_INT_ENABLE1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SCU_SYSTEM_INT_ENABLE1 enables or disables output of interrupts corresponding to the states indicated in the SCU_SYSTEM status register 1.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | uf_src9 o_ie | uf_src8 o_ie | uf_src7 o_ie | uf_src6 o_ie | uf_src5 o_ie | uf_src4 o_ie | uf_src3 o_ie | uf_src2 o_ie | uf_src1 o_ie | uf_src0 o_ie |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | of_src9i _ie | of_src8i _ie | of_src7i _ie | of_src6i _ie | of_src5i _ie | of_src4i _ie | of_src3i _ie | of_src2i _ie | of_src1i _ie | of_src0i _ie |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 26 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25 | uf_src9o_ie | B'0 | R/W | under_flow_src9out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 24 | uf_src8o_ie | B'0 | R/W | under_flow_src8out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 23 | uf_src7o_ie | B'0 | R/W | under_flow_src7out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 22 | uf_src6o_ie | B'0 | R/W | under_flow_src6out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 21 | uf_src5o_ie | B'0 | R/W | under_flow_src5out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 20 | uf_src4o_ie | B'0 | R/W | under_flow_src4out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 19 | uf_src3o_ie | B'0 | R/W | under_flow_src3out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 18 | uf_src2o_ie | B'0 | R/W | under_flow_src2out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 17 | uf_src1o_ie | B'0 | R/W | under_flow_src1out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 16 | uf_src0o_ie | B'0 | R/W | under_flow_src0out_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | of_src9i_ie | B'0 | R/W | over_flow_src9in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 8 | of_src8i_ie | B'0 | R/W | over_flow_src8in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 7 | of_src7i_ie | B'0 | R/W | over_flow_src7in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 6 | of_src6i_ie | B'0 | R/W | over_flow_src6in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 5 | of_src5i_ie | B'0 | R/W | over_flow_src5in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 4 | of_src4i_ie | B'0 | R/W | over_flow_src4in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 3 | of_src3i_ie | B'0 | R/W | over_flow_src3in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 2 | of_src2i_ie | B'0 | R/W | over_flow_src2in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 1 | of_src1i_ie | B'0 | R/W | over_flow_src1in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 0 | of_src0i_ie | B'0 | R/W | over_flow_src0in_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

43.2.18 SRCm Software Reset Register (SRCm_SWRSR)

Note: m = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm_SWRSR is a 32-bit readable/writable register that controls operation/reset of the SRC internal circuits.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SWRST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | SWRST | B'1 | R/W | Software Reset While this bit is 0, the SRC internal circuits are put in the reset state. SRCm_* registers except this register are reset. Therefore, they should be set again after the reset is canceled. 0: Resets the SRC 1: SRC enters the operating state |

43.2.19 SRCm SRC Initialization Register (SRCm_SRCIR)

Note: m = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm_SRCIR is a 32-bit readable/writable register that initializes the operation of the SRC internal circuits.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | INIT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | INIT | B'1 | R/W | Initialization of Processing When this bit is set to 1, the SRC processing is initialized. This bit should be cleared to 0 after it was set to 1. 0: Processing State 1: Initialization (sets the initial setting of other registers) |

43.2.20 SRCm Audio Information Register (SRCm_ADINR)

Note: m = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm_ADINR is a 32-bit readable/writable register that selects channel number and bit length of output audio data.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|-----------|------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | OTBL[4:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CHNUM[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 to 16 | OTBL[4:0] | H'00 | R/W | Bit Length of Output Audio Data. These bits set the bit length of output audio data. B'0_0000: 24 bits B'0_0001: Reserved B'0_0010: 22 bits B'0_0011: Reserved B'0_0100: 20 bits B'0_0101: Reserved B'0_0110: 18 bits B'0_0111: Reserved B'0_1000: 16 bits B'0_1001 to B'0_1111: Reserved B'1_0000: 8 bits B'1_0001 to B'1_1111: Reversed |
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|--|
| 3 to 0 | CHNUM[3:0] | H'0 | R/W | <p>Channel Number</p> <p>These bits set the input data channel number.</p> <p>m = 0, 1, 3, or 4 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] m = 1 and 3 [RZ/G2E]</p> <p>B'0000: 0 (None) B'0001: 1 channel B'0010: 2 channels B'0011: Reserved B'0100: 4 channels B'0101: Reserved B'0110: 6 channels B'0111: Reserved B'1000: 8 channels B'1001 to B'1111: Reserved</p> <p>m = 2, 5 to 9 [, RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] m = 0, 2, 4, 5 or 9 [RZ/G2E]</p> <p>B'0000: 0 (None) B'0001: 1 channel B'0010: 2 channels B'0011 to B'1111: Reserved</p> |

43.2.21 SRCm IFS Control Register (SRCm_IFSCR)

Note: m = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm_IFSCR is a 32-bit readable/writable register that controls INTIFS value.

| | | | | | | | | | | | | | | | | | | |
|----------------|-------------|----|----|----|----|----|----|----|-----------------------|-----|-----|-----|-----|-----|-----|-----------|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | fast_ijk_en | — | — | — | — | — | — | — | fast_ijk_setting[7:0] | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R/W | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | INTIFS EN | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------------|---------------|-----|--|
| 31 | fast_ijk_en | B'0 | R/W | fast_ijk Enable It can diminish convergence time to input frequency deflection. 0: Reserved 1: It can be gradual decrease of convergence time. This function is valid if SRCm_SRCCR Register SRCMD = 0 (Asynchronous SRC). |
| 30 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 20 | fast_ijk_setting[7:4] | H'0 | R/W | The period that to recognize “stable” by ijc_fsi calculator. This bit should be set to H'2. (Only when fast_ijk_en = 1, as bit 31) |
| 19 to 16 | fast_ijk_setting[3:0] | H'0 | R/W | The period that to rerun calculation by ijc_fsi calculator. This bit should be set to H'4. (Only when fast_ijk_en = 1, as bit 31) |
| 15 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | INTIFSEN | B'0 | R/W | INTIFS Value Setting Enable This bit controls the INTIFS bit of SRCm_IFSVR register. This bit is effective only in asynchronous SRC Mode (bit SRCMD in SRCm_SRCCR register is 0) 0: Disables INTIFS bit of SRCm_IFSVR register 1: Enables INTIFS bit of SRCm_IFSVR register Basically, this bit should be set to 1. |

43.2.22 SRCm IFS Value Setting Register (SRCm_IFSVR)

Note: m = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

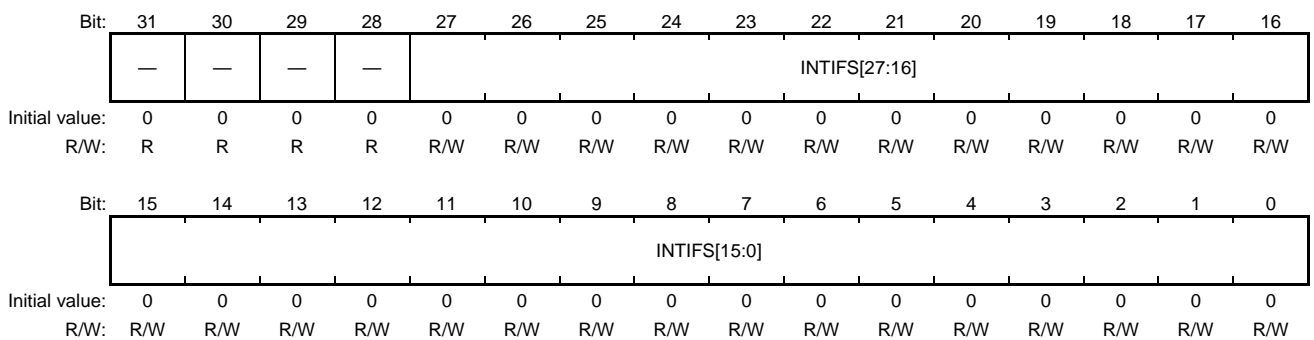
Function: SRCm_IFSVR is a 32-bit readable/writable register that sets the value of INTIFS. The INTIFS is the initial value of FSI of sampling rate conversion function. SRC detects input sampling rate and output sampling rate automatically. By doing this, it calculates the FSI value so that the below formula is satisfied.

$$F_{in}/F_{out} \approx FSI/FSO$$

The INTIFS value is used for the initial value of FSI. If this setting is disabled (INTIFSEN bit of SRCm_IFSCR register is 0), SRC will use FSO for initial value of FSI.

$$INTIFS = F_{in} \times FSO/F_{out}$$

- Note:
- Fin: Input sampling frequency
 - Fout: Output sampling frequency
 - FSI: Input sampling rate
 - FSO: Output sampling rate (Fixed value: $2^{22} = H'040_0000$)



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|---|
| 31 to 28 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 to 0 | INTIFS[27:0] | H'000_0000 | R/W | Initial Value of FSI These bits set initial value of FSI. These bits should be set the ratio of input and output sampling-rate within the range of the restriction decided from channel number. Example: $F_{in} = 32 \text{ kHz}$ $F_{out} = 44.1 \text{ kHz}$ $FSI = 2^{22} \times 32000 / 44100 = 3043485 = H'02E_709D$ For the setting value examples, see Table 43.10. |

Note: When SRCm is used in synchronous mode (SRCMD bit in SRCm_SRCR register is 1), INTIFS can change without SRC initialization. On the operating state, INTIFS can change within 1%. However, the change of the SRC conversion rate means that it will set jitter by manual operation. Evaluate sound quality, and use this function as far as you judged sound quality is no problem. If on the occasion of the change more than 1%, it should do SRC initialization (cf. section 43.3.5(1)(d)) after changing INTIFS.

43.2.23 SRCm SRC Control Register (SRCm_SRCCR)

Note: m = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm_SRCCR is a 32-bit readable/writable register that controls the operation of sampling rate converter.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----|----|----|----|-----|----|----|----|-----|----|----|----|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SRCM D |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R/W | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | — | B'0 | R/W | Reserved This bit should be set to 1. |
| 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | — | B'0 | R/W | Reserved This bit should be set to 1. |
| 11 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | — | B'0 | R/W | Reserved This bit should be set to 1. |
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | — | B'0 | R/W | Reserved This bit should be set to 1. |
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | SRCMD | B'0 | R/W | Select SRC Mode 0: Asynchronous SRC 1: Synchronous SRC |

Note: Set sync_in or sync_out bit in SRCm_MODE register, when SRCm is used in synchronous mode by setting SRCMD = 1.

43.2.24 SRCm Buffer Size DATA RAM Setting Register (SRCm_BSDSR)

Note: m = 0 to 4, 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm_BSDSR is a 32-bit readable/writable register that controls the buffer size of DATA RAM. This register should set with SRCm_BSISR register. See Table 43.8 and 43.9 for the combination of settings.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|---------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | BUFDATA[10:0] | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

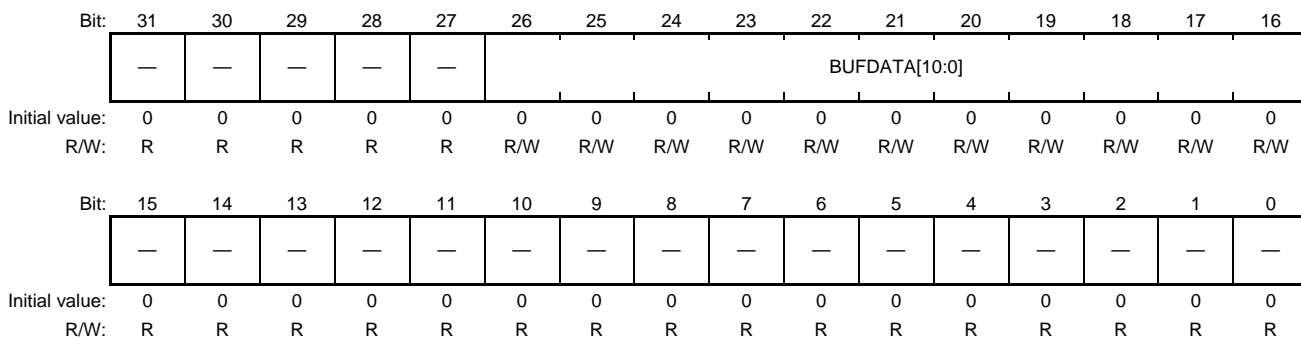
| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | |
|---------------|----------------|---------------|-----|--|---------------|-------|---------|-------|---------|-------|---------|-------|---------|-------|---------|-------|-------|-------|
| 31 to 27 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | |
| 26 to 16 | BUFDATA [10:0] | H'000 | R/W | These bits should be set appropriately according to the following table. Refer to Table 43.8 and 43.9 for the combination of FSO/FSI ratio, channel number, and latency. | | | | | | | | | | | | | | |
| | | | | <table border="1"> <thead> <tr> <th>FSO/FSI Ratio</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>6 - 1/6</td> <td>H'180</td> </tr> <tr> <td>6 - 1/4</td> <td>H'100</td> </tr> <tr> <td>6 - 1/3</td> <td>H'0C0</td> </tr> <tr> <td>6 - 1/2</td> <td>H'080</td> </tr> <tr> <td>6 - 2/3</td> <td>H'060</td> </tr> <tr> <td>6 - 1</td> <td>H'040</td> </tr> </tbody> </table> | FSO/FSI Ratio | Value | 6 - 1/6 | H'180 | 6 - 1/4 | H'100 | 6 - 1/3 | H'0C0 | 6 - 1/2 | H'080 | 6 - 2/3 | H'060 | 6 - 1 | H'040 |
| FSO/FSI Ratio | Value | | | | | | | | | | | | | | | | | |
| 6 - 1/6 | H'180 | | | | | | | | | | | | | | | | | |
| 6 - 1/4 | H'100 | | | | | | | | | | | | | | | | | |
| 6 - 1/3 | H'0C0 | | | | | | | | | | | | | | | | | |
| 6 - 1/2 | H'080 | | | | | | | | | | | | | | | | | |
| 6 - 2/3 | H'060 | | | | | | | | | | | | | | | | | |
| 6 - 1 | H'040 | | | | | | | | | | | | | | | | | |
| 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | |

43.2.25 SRCm Buffer Size DATA RAM Setting Register (SRCm_BSDSR)

Note: m = 5 to 8

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCn_BSDSR is a 32-bit readable/writable register that controls the buffer size of DATA RAM. This register should set with SRCm_BSISR register. See Table 43.8 and 43.9 for the combination of settings.



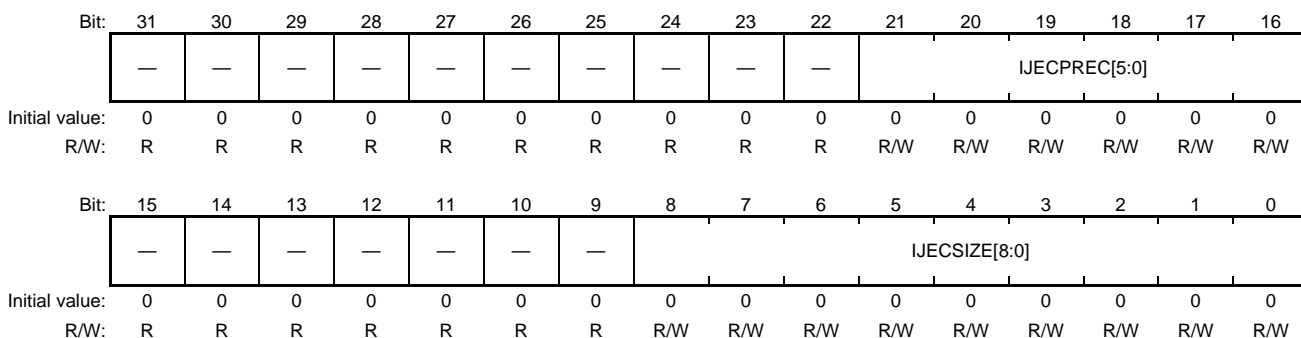
| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | |
|---------------|----------------|---------------|-----|--|---------------|-------|---------|-------|---------|-------|---------|-------|---------|-------|---------|-------|-------|-------|
| 31 to 27 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | |
| 26 to 16 | BUFDATA [10:0] | H'000 | R/W | These bits should be set appropriately according to the following table. Refer to Table 43.8 and 43.9 for the combination of FSO/FSI ratio, channel number, and latency. | | | | | | | | | | | | | | |
| | | | | <table border="1"> <thead> <tr> <th>FSO/FSI Ratio</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>6 - 1/6</td> <td>H'240</td> </tr> <tr> <td>6 - 1/4</td> <td>H'180</td> </tr> <tr> <td>6 - 1/3</td> <td>H'120</td> </tr> <tr> <td>6 - 1/2</td> <td>H'0C0</td> </tr> <tr> <td>6 - 2/3</td> <td>H'090</td> </tr> <tr> <td>6 - 1</td> <td>H'060</td> </tr> </tbody> </table> | FSO/FSI Ratio | Value | 6 - 1/6 | H'240 | 6 - 1/4 | H'180 | 6 - 1/3 | H'120 | 6 - 1/2 | H'0C0 | 6 - 2/3 | H'090 | 6 - 1 | H'060 |
| FSO/FSI Ratio | Value | | | | | | | | | | | | | | | | | |
| 6 - 1/6 | H'240 | | | | | | | | | | | | | | | | | |
| 6 - 1/4 | H'180 | | | | | | | | | | | | | | | | | |
| 6 - 1/3 | H'120 | | | | | | | | | | | | | | | | | |
| 6 - 1/2 | H'0C0 | | | | | | | | | | | | | | | | | |
| 6 - 2/3 | H'090 | | | | | | | | | | | | | | | | | |
| 6 - 1 | H'060 | | | | | | | | | | | | | | | | | |
| 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | |

43.2.26 SRCm Buffer Size IJEC RAM Setting Register (SRCm_BSISR)

Note: m = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm_BSISR is a 32-bit readable/writable register that controls the precision value and buffer size of IJEC RAM. This register should set with SRCm_BSISR/SRCn_BSISR register. See Table 43.8 and 43.9 for the combination of settings.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 to 16 | IJECPREC[5:0] | H'00 | R/W | These bits should be set to H'10. |
| 15 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 to 0 | IJECSIZE[8:0] | H'000 | R/W | These bits should be set appropriately according to the following table. |

Refer to Table 43.8 and 43.9 for the combination of FSO/FSI ratio, channel number, and latency.

| FSO/FSI Ratio | Value |
|---------------|-------|
| 6 - 1/6 | H'60 |
| 6 - 1/4 | H'40 |
| 6 - 1/3 | H'30 |
| 6 - 1/2 | H'20 |
| 6 - 2/3 | H'20 |
| 6 - 1 | H'20 |

43.2.27 CTUn Software Reset Register (CTUn_SWRSR)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SWRSR is a 32-bit readable/writable register that controls operation/reset of the CTU internal circuits.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SWRST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | SWRST | B'1 | R/W | Software Reset While this bit is 0, the CTU internal circuits are put in the reset state. CTUn_* registers except this register are reset. Therefore, they should be set again after the reset is canceled. 0: Resets the CTU 1: Operating state |

43.2.28 CTUn CTU Initialization Register (CTUn_CTUIR)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_CTUIR is a 32-bit readable/writable register that initializes the operation of the CTU internal circuits.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | INIT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | INIT | B'1 | R/W | Initialization of Processing When this bit is set to 1, the CTU processing is initialized. This bit should be cleared to 0 after it was set to 1. 0: Processing State 1: Initialization (sets the initial setting of other registers) |

43.2.29 CTUn Audio Information Register (CTUn_ADINR)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_ADINR is a 32-bit readable/writable register that selects channel number.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CHNUM[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

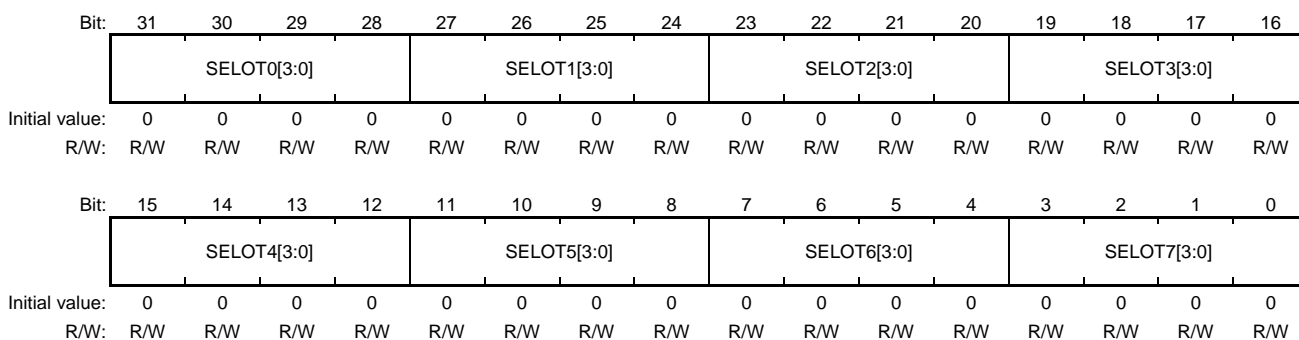
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | CHNUM[3:0] | H'0 | R/W | Channel Number These bits set the input data channel number. B'0000: 0 (None) B'0001: 1 channel B'0010: 2 channels B'0011: Reserved B'0100: 4 channels B'0101: Reserved B'0110: 6 channels B'0111: Reserved B'1000: 8 channels B'1001 to B'1111: Reserved |

43.2.30 CTUn CTU Pass Mode Register (CTUn_CPMDR)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_CPMDR is a 32-bit readable/writable register that controls the pass of channel data for each output.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 28 | SELOT0[3:0] | H'0 | R/W | Select output data for channel 0 These bits select the output data for channel 0. B'0000: Connect input data of channel 0 B'0001: Connect input data of channel 0 B'0010: Connect input data of channel 1 B'0011: Connect input data of channel 2 B'0100: Connect input data of channel 3 B'0101: Connect input data of channel 4 B'0110: Connect input data of channel 5 B'0111: Connect input data of channel 6 B'1000: Connect input data of channel 7 B'1001: Connect calculated data by scale values of matrix row 0 B'1010: Connect calculated data by scale values of matrix row 1 B'1011: Connect calculated data by scale values of matrix row 2 B'1100: Connect calculated data by scale values of matrix row 3 Others: Setting prohibited. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 27 to 24 | SELOT1[3:0] | H'0 | R/W | <p>Select output data for channel 1</p> <p>These bits select the output data for channel 1.</p> <p>B'0000: Connect input data of channel 1 B'0001: Connect input data of channel 0 B'0010: Connect input data of channel 1 B'0011: Connect input data of channel 2 B'0100: Connect input data of channel 3 B'0101: Connect input data of channel 4 B'0110: Connect input data of channel 5 B'0111: Connect input data of channel 6 B'1000: Connect input data of channel 7 B'1001: Connect calculated data by scale values of matrix row 0 B'1010: Connect calculated data by scale values of matrix row 1 B'1011: Connect calculated data by scale values of matrix row 2 B'1100: Connect calculated data by scale values of matrix row 3 Others: Setting prohibited.</p> |
| 23 to 20 | SELOT2[3:0] | H'0 | R/W | <p>Select output data for channel 2</p> <p>These bits select the output data for channel 2.</p> <p>B'0000: Connect input data of channel 2 B'0001: Connect input data of channel 0 B'0010: Connect input data of channel 1 B'0011: Connect input data of channel 2 B'0100: Connect input data of channel 3 B'0101: Connect input data of channel 4 B'0110: Connect input data of channel 5 B'0111: Connect input data of channel 6 B'1000: Connect input data of channel 7 B'1001: Connect calculated data by scale values of matrix row 0 B'1010: Connect calculated data by scale values of matrix row 1 B'1011: Connect calculated data by scale values of matrix row 2 B'1100: Connect calculated data by scale values of matrix row 3 Others: Setting prohibited.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 19 to 16 | SELOT3[3:0] | H'0 | R/W | <p>Select output data for channel 3</p> <p>These bits select the output data for channel 3.</p> <p>B'0000: Connect input data of channel 3 B'0001: Connect input data of channel 0 B'0010: Connect input data of channel 1 B'0011: Connect input data of channel 2 B'0100: Connect input data of channel 3 B'0101: Connect input data of channel 4 B'0110: Connect input data of channel 5 B'0111: Connect input data of channel 6 B'1000: Connect input data of channel 7 B'1001: Connect calculated data by scale values of matrix row 0 B'1010: Connect calculated data by scale values of matrix row 1 B'1011: Connect calculated data by scale values of matrix row 2 B'1100: Connect calculated data by scale values of matrix row 3 Others: Setting prohibited.</p> |
| 15 to 12 | SELOT4[3:0] | H'0 | R/W | <p>Select output data for channel 4</p> <p>These bits select the output data for channel 4.</p> <p>B'0000: Connect input data of channel 4 B'0001: Connect input data of channel 0 B'0010: Connect input data of channel 1 B'0011: Connect input data of channel 2 B'0100: Connect input data of channel 3 B'0101: Connect input data of channel 4 B'0110: Connect input data of channel 5 B'0111: Connect input data of channel 6 B'1000: Connect input data of channel 7 B'1001: Connect calculated data by scale values of matrix row 0 B'1010: Connect calculated data by scale values of matrix row 1 B'1011: Connect calculated data by scale values of matrix row 2 B'1100: Connect calculated data by scale values of matrix row 3 Others: Setting prohibited.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|---|
| 11 to 8 | SELOT5[3:0] | H'0 | R/W | <p>Select output data for channel 5</p> <p>These bits select the output data for channel 5.</p> <p>B'0000: Connect input data of channel 5 B'0001: Connect input data of channel 0 B'0010: Connect input data of channel 1 B'0011: Connect input data of channel 2 B'0100: Connect input data of channel 3 B'0101: Connect input data of channel 4 B'0110: Connect input data of channel 5 B'0111: Connect input data of channel 6 B'1000: Connect input data of channel 7 B'1001: Connect calculated data by scale values of matrix row 0 B'1010: Connect calculated data by scale values of matrix row 1 B'1011: Connect calculated data by scale values of matrix row 2 B'1100: Connect calculated data by scale values of matrix row 3 Others: Setting prohibited.</p> |
| 7 to 4 | SELOT6[3:0] | H'0 | R/W | <p>Select output data for channel 6</p> <p>These bits select the output data for channel 6.</p> <p>B'0000: Connect input data of channel 6 B'0001: Connect input data of channel 0 B'0010: Connect input data of channel 1 B'0011: Connect input data of channel 2 B'0100: Connect input data of channel 3 B'0101: Connect input data of channel 4 B'0110: Connect input data of channel 5 B'0111: Connect input data of channel 6 B'1000: Connect input data of channel 7 B'1001: Connect calculated data by scale values of matrix row 0 B'1010: Connect calculated data by scale values of matrix row 1 B'1011: Connect calculated data by scale values of matrix row 2 B'1100: Connect calculated data by scale values of matrix row 3 Others: Setting prohibited.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-------------|---------------|-----|--|
| 3 to 0 | SELOT7[3:0] | H'0 | R/W | <p>Select output data for channel 7.</p> <p>These bits select the output data for channel 7.</p> <p>B'0000: Connect input data of channel 7</p> <p>B'0001: Connect input data of channel 0</p> <p>B'0010: Connect input data of channel 1</p> <p>B'0011: Connect input data of channel 2</p> <p>B'0100: Connect input data of channel 3</p> <p>B'0101: Connect input data of channel 4</p> <p>B'0110: Connect input data of channel 5</p> <p>B'0111: Connect input data of channel 6</p> <p>B'1000: Connect input data of channel 7</p> <p>B'1001: Connect calculated data by scale values of matrix row 0</p> <p>B'1010: Connect calculated data by scale values of matrix row 1</p> <p>B'1011: Connect calculated data by scale values of matrix row 2</p> <p>B'1100: Connect calculated data by scale values of matrix row 3</p> <p>Others: Setting prohibited.</p> |

43.2.31 CTUn Scale Mode Register (CTUn_SCMDR)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SCMDR is a 32-bit readable/writable register that selects the number of rows calculated by matrix.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-----------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | SCMD[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W |

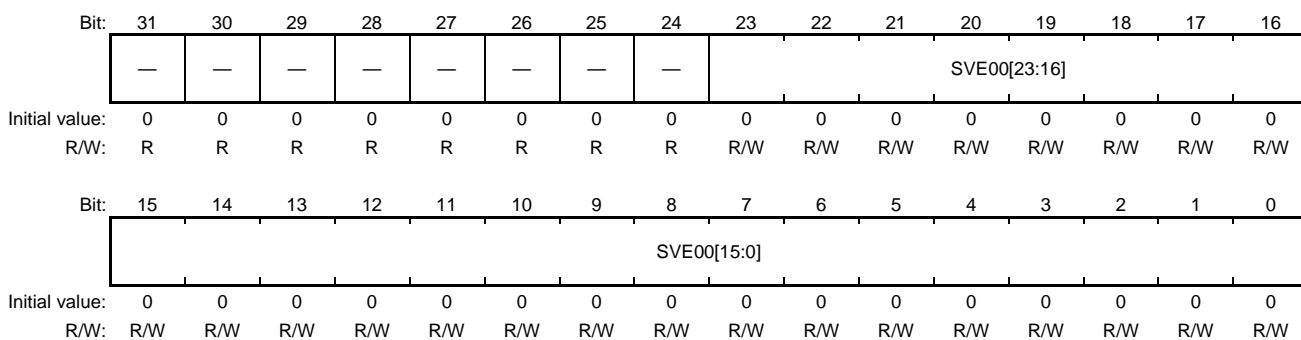
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 to 0 | SCMD[2:0] | H'0 | R/W | The number of rows calculated by matrix These bits select the number of rows calculated by matrix by the scale values. B'000: No operation B'001: Calculate matrix row 0 by scale values. B'010: Calculate matrix row 0 and 1 by scale values. B'011: Calculate matrix row 0 and 1 and 2 by scale values. B'100: Calculate matrix row 0 and 1 and 2 and 3 by scale values. Others: Setting prohibited. |

43.2.32 CTUn Scale Value e00 Register (CTUn_SV00R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV00R is a 32-bit readable/writable register that sets the scale value for channel 0 of matrix row 0.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE00[23:0] | H'00_0000 | R/W | Scale Value e00 for Input Channel 0 of Matrix Row 0 These bits set the scale value for input data of channel 0 of matrix row 0. SVE00[23]: Sign bit SVE00[22]: Integer bit SVE00[21:0]: Decimal bits |

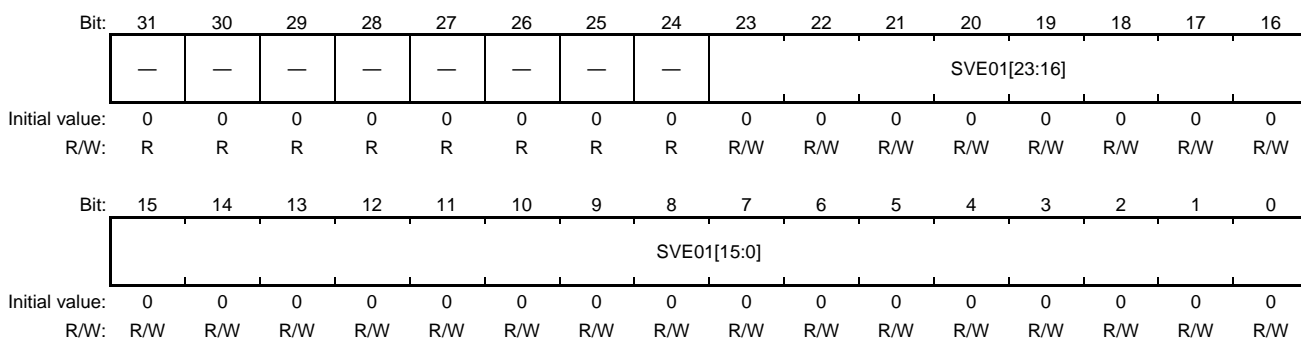
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.33 CTUn Scale Value e01 Register (CTUn_SV01R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV01R is a 32-bit readable/writable register that sets the scale value for channel 1 of matrix row 0.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE01[23:0] | H'00_0000 | R/W | Scale Value e01 for Input Channel 1 of Matrix Row 0 These bits set the scale value for input data of channel 1 of matrix row 0. SVE01[23]: Sign bit SVE01[22]: Integer bit SVE01[21:0]: Decimal bits |

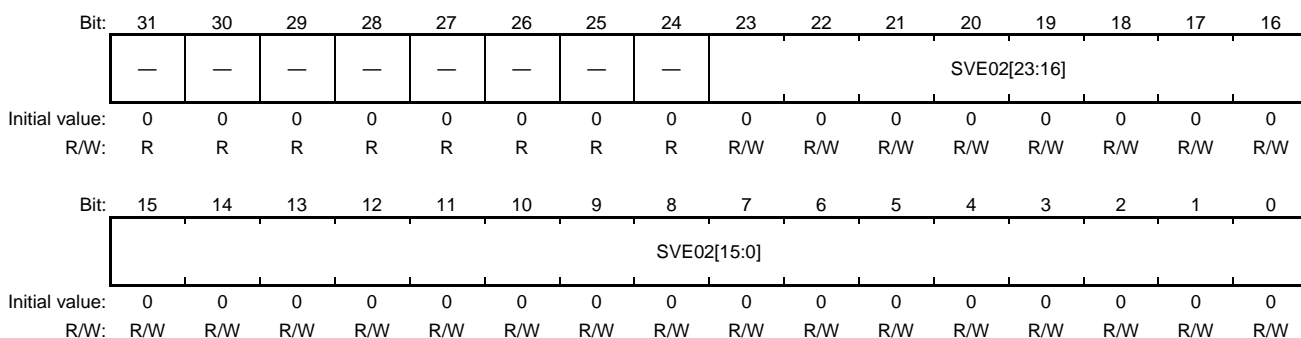
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.34 CTUn Scale Value e02 Register (CTUn_SV02R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV02R is a 32-bit readable/writable register that sets the scale value for channel 2 of matrix row 0.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE02[23:0] | H'00_0000 | R/W | Scale Value e02 for Input Channel 2 of Matrix Row 0 These bits set the scale value for input data of channel 2 of matrix row 0. SVE02[23]: Sign bit SVE02[22]: Integer bit SVE02[21:0]: Decimal bits |

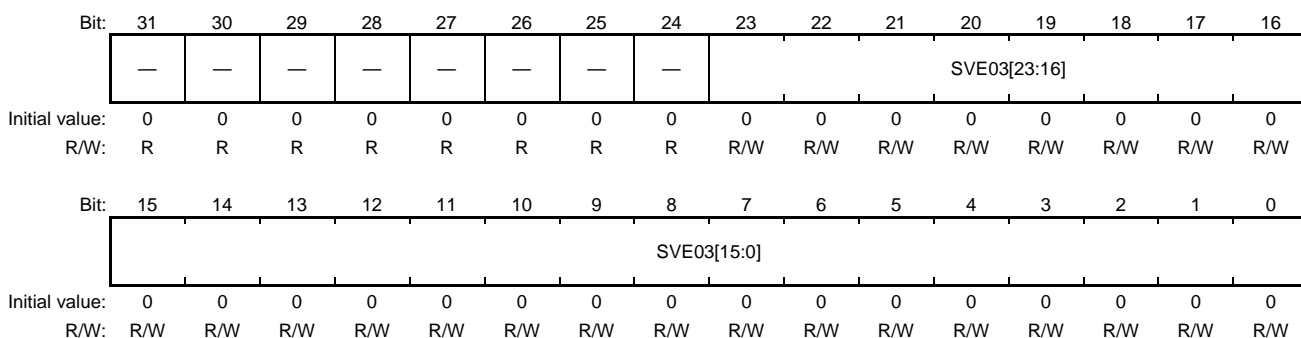
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.35 CTUn Scale Value e03 Register (CTUn_SV03R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV03R is a 32-bit readable/writable register that sets the scale value for channel 3 of matrix row 0.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE03[23:0] | H'00_0000 | R/W | Scale Value e03 for Input Channel 3 of Matrix Row 0 These bits set the scale value for input data of channel 3 of matrix row 0. SVE03[23]: Sign bit SVE03[22]: Integer bit SVE03[21:0]: Decimal bits |

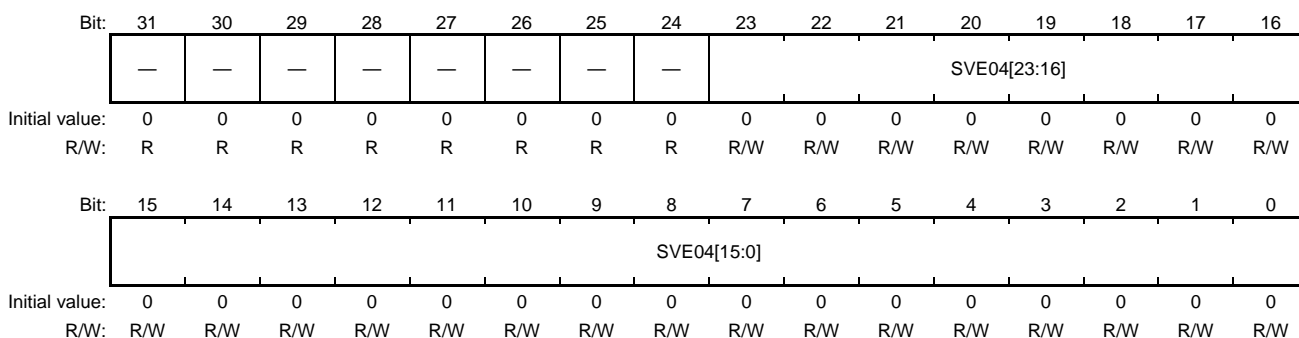
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.36 CTUn Scale Value e04 Register (CTUn_SV04R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV04R is a 32-bit readable/writable register that sets the scale value for channel 4 of matrix row 0.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE04[23:0] | H'00_0000 | R/W | Scale Value e04 for Input Channel 4 of Matrix Row 0 These bits set the scale value for input data of channel 4 of matrix row 0. SVE04[23]: Sign bit SVE04[22]: Integer bit SVE04[21:0]: Decimal bits |

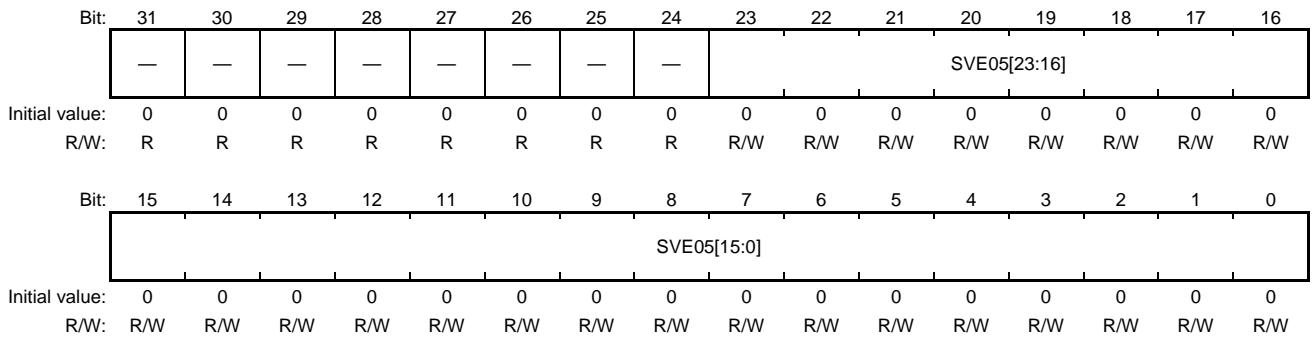
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.37 CTUn Scale Value e05 Register (CTUn_SV05R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV05R is a 32-bit readable/writable register that sets the scale value for channel 5 of matrix row 0.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE05[23:0] | H'00_0000 | R/W | Scale Value e05 for Input Channel 5 of Matrix Row 0 These bits set the scale value for input data of channel 5 of matrix row 0. SVE05[23]: Sign bit SVE05[22]: Integer bit SVE05[21:0]: Decimal bits |

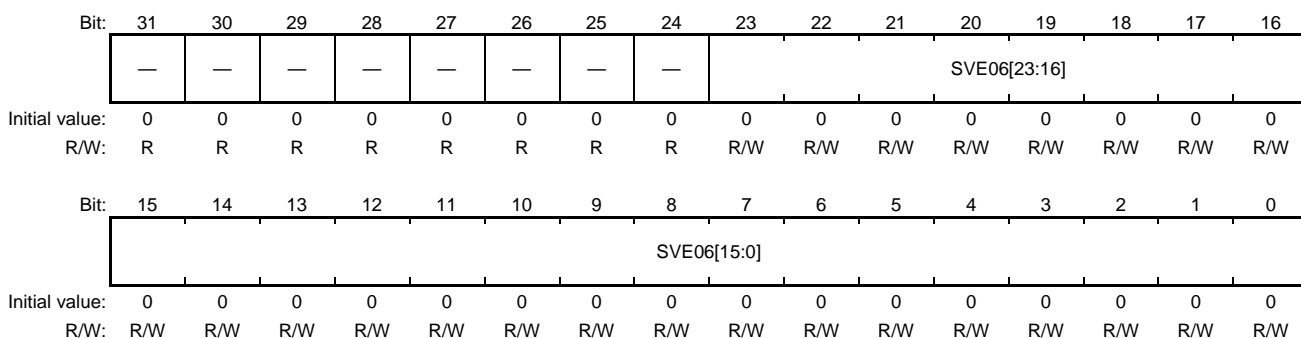
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.38 CTUn Scale Value e06 Register (CTUn_SV06R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV06R is a 32-bit readable/writable register that sets the scale value for channel 6 of matrix row 0.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE06[23:0] | H'00_0000 | R/W | Scale Value e06 for Input Channel 6 of Matrix Row 0 These bits set the scale value for input data of channel 6 of matrix row 0. SVE06[23]: Sign bit SVE06[22]: Integer bit SVE06[21:0]: Decimal bits |

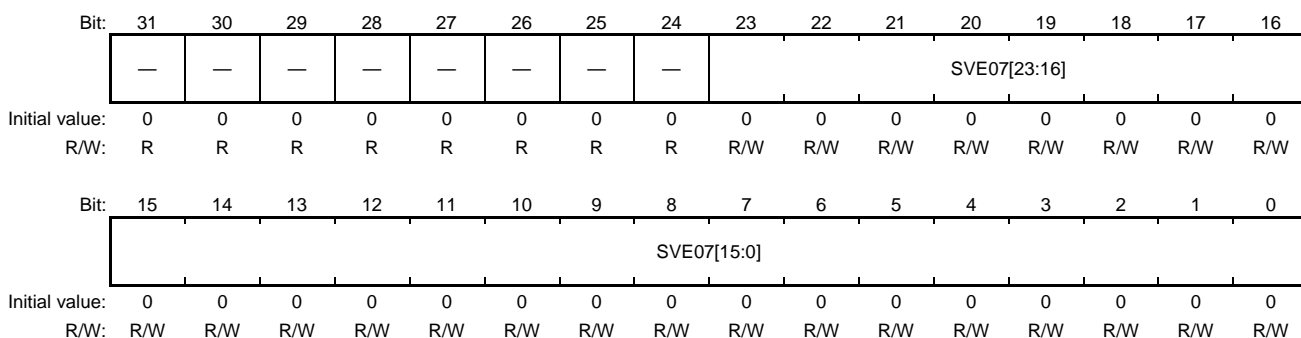
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.39 CTUn Scale Value e07 Register (CTUn_SV07R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV07R is a 32-bit readable/writable register that sets the scale value for channel 7 of matrix row 0.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE07[23:0] | H'00_0000 | R/W | Scale Value e07 for Input Channel 7 of Matrix Row 0 These bits set the scale value for input data of channel 7 of matrix row 0. SVE07[23]: Sign bit SVE07[22]: Integer bit SVE07[21:0]: Decimal bits |

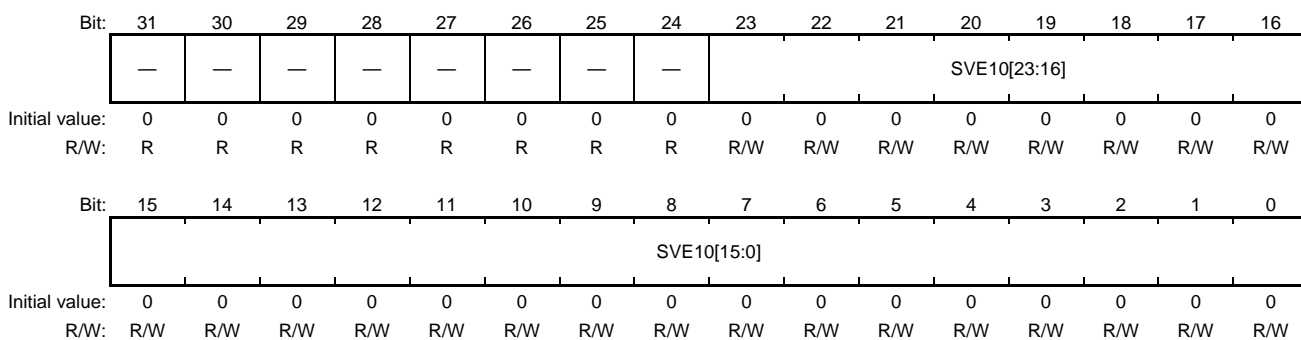
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.40 CTUn Scale Value e10 Register (CTUn_SV10R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV10R is a 32-bit readable/writable register that sets the scale value for channel 0 of matrix row 1.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE10[23:0] | H'00_0000 | R/W | Scale Value e10 for Input Channel 0 of Matrix Row 1 These bits set the scale value for input data of channel 0 of matrix row 1. SVE10[23]: Sign bit SVE10[22]: Integer bit SVE10[21:0]: Decimal bits |

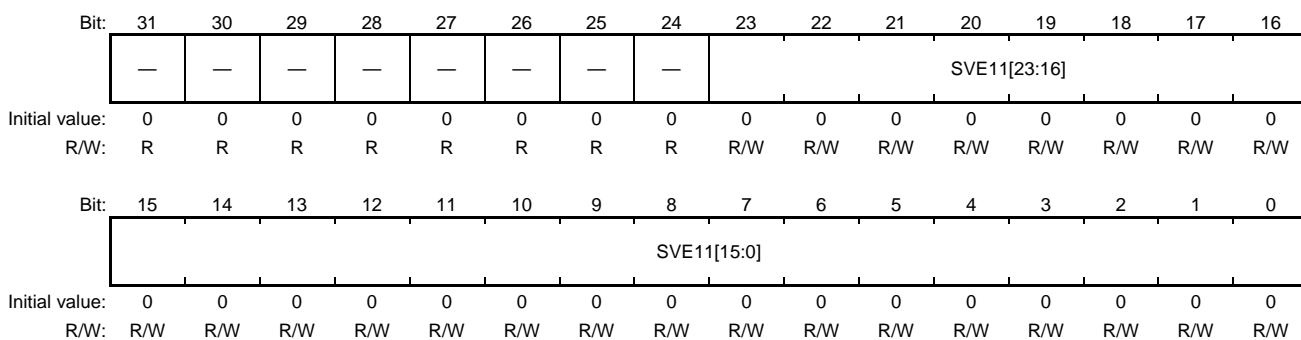
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.41 CTUn Scale Value e11 Register (CTUn_SV11R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV11R is a 32-bit readable/writable register that sets the scale value for channel 1 of matrix row 1.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE11[23:0] | H'00_0000 | R/W | Scale Value e11 for Input Channel 1 of Matrix Row 1 These bits set the scale value for input data of channel 1 of matrix row 1. SVE11[23]: Sign bit SVE11[22]: Integer bit SVE11[21:0]: Decimal bits |

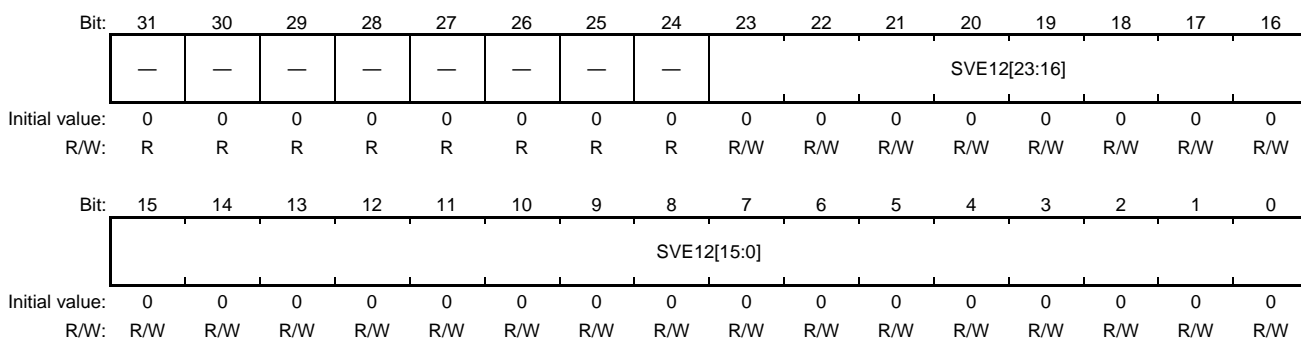
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.42 CTUn Scale Value e12 Register (CTUn_SV12R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV12R is a 32-bit readable/writable register that sets the scale value for channel 2 of matrix row 1.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE12[23:0] | H'00_0000 | R/W | Scale Value e12 for Input Channel 2 of Matrix Row 1 These bits set the scale value for input data of channel 2 of matrix row 1. SVE12[23]: Sign bit SVE12[22]: Integer bit SVE12[21:0]: Decimal bits |

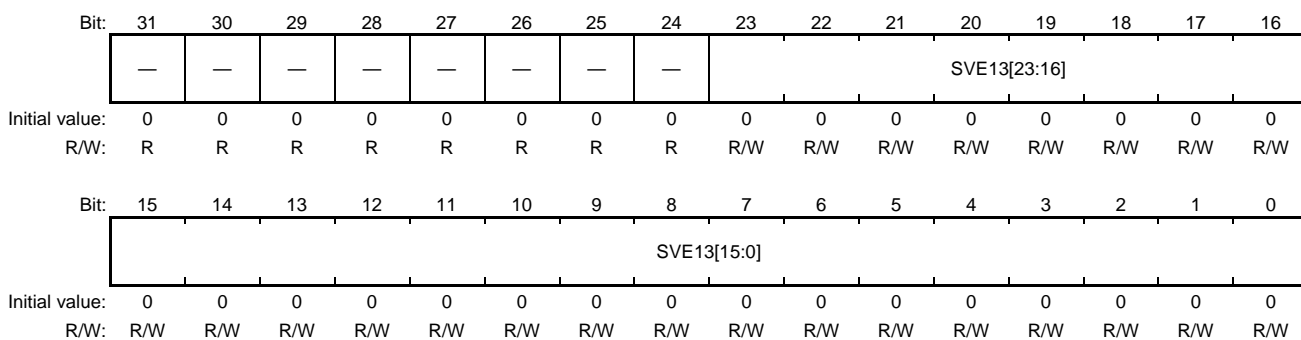
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.43 CTUn Scale Value e13 Register (CTUn_SV13R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV13R is a 32-bit readable/writable register that sets the scale value for channel 3 of matrix row 1.



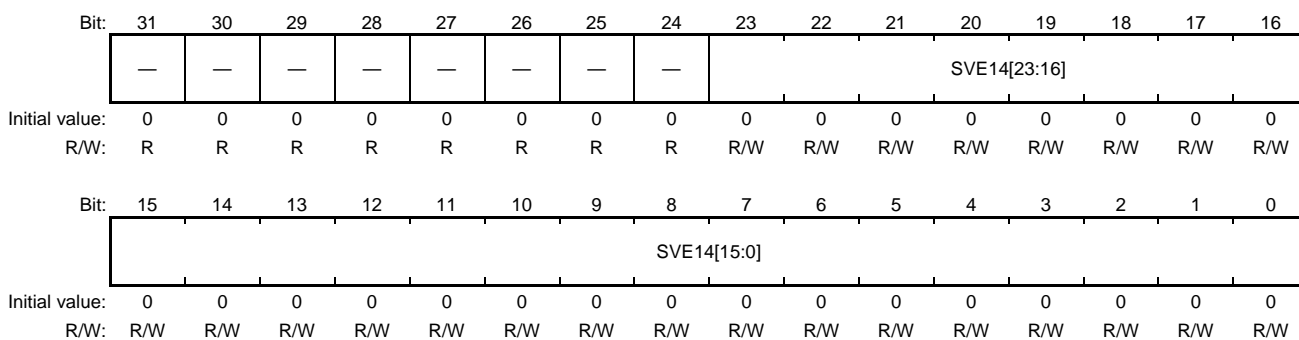
| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|-----------------------|--|-----------|--|-------|--|--|-------|--|--|-------|--------|------|-------|--------|------|-----------|---|---|-----------|---|---|-------|-------|-------|-------|-------|-------|-----------|---|---|-----------|---|---|-------|-------|-------|-------|-------|-------|-----------|-----------------------|------|-------|-------|-------|-----------|----------|-----------|-----------|-----------------------|------|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 23 to 0 | SVE13[23:0] | H'00_0000 | R/W | Scale Value e13 for Input Channel 3 of Matrix Row 1 These bits set the scale value for input data of channel 3 of matrix row 1. SVE13[23]: Sign bit SVE13[22]: Integer bit SVE13[21:0]: Decimal bits | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | <table border="1"> <thead> <tr> <th colspan="3">plus</th> <th colspan="3">minus</th> </tr> <tr> <th>Value</th> <th>[time]</th> <th>[dB]</th> <th>Value</th> <th>[time]</th> <th>[dB]</th> </tr> </thead> <tbody> <tr> <td>H'7F_FFFF</td> <td>2</td> <td>6</td> <td>H'80_0000</td> <td>2</td> <td>6</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'40_0000</td> <td>1</td> <td>0</td> <td>H'C0_0000</td> <td>1</td> <td>0</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'00_0001</td> <td>2.38×10^{-7}</td> <td>-132</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'00_0000</td> <td>0 (Mute)</td> <td>$-\infty$</td> <td>H'FF_FFFF</td> <td>2.38×10^{-7}</td> <td>-132</td> </tr> </tbody> </table> | | | plus | | | minus | | | Value | [time] | [dB] | Value | [time] | [dB] | H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 | | | | | | | H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 | | | | | | | H'00_0001 | 2.38×10^{-7} | -132 | | | | H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |
| plus | | | minus | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Value | [time] | [dB] | Value | [time] | [dB] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

43.2.44 CTUn Scale Value e14 Register (CTUn_SV14R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV14R is a 32-bit readable/writable register that sets the scale value for channel 4 of matrix row 1.



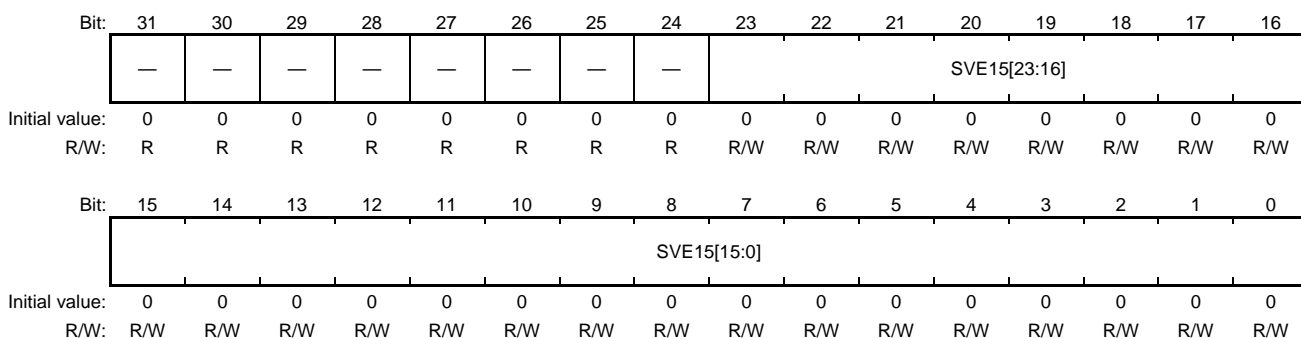
| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|-----------------------|--|-----------|--|-------|--|--|-------|--|--|-------|--------|------|-------|--------|------|-----------|---|---|-----------|---|---|-------|-------|-------|-------|-------|-------|-----------|---|---|-----------|---|---|-------|-------|-------|-------|-------|-------|-----------|-----------------------|------|-------|-------|-------|-----------|----------|-----------|-----------|-----------------------|------|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 23 to 0 | SVE14[23:0] | H'00_0000 | R/W | Scale Value e14 for Input Channel 4 of Matrix Row 1 These bits set the scale value for input data of channel 4 of matrix row 1. SVE14[23]: Sign bit SVE14[22]: Integer bit SVE14[21:0]: Decimal bits | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | <table border="1"> <thead> <tr> <th colspan="3">plus</th> <th colspan="3">minus</th> </tr> <tr> <th>Value</th> <th>[time]</th> <th>[dB]</th> <th>Value</th> <th>[time]</th> <th>[dB]</th> </tr> </thead> <tbody> <tr> <td>H'7F_FFFF</td> <td>2</td> <td>6</td> <td>H'80_0000</td> <td>2</td> <td>6</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'40_0000</td> <td>1</td> <td>0</td> <td>H'C0_0000</td> <td>1</td> <td>0</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'00_0001</td> <td>2.38×10^{-7}</td> <td>-132</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'00_0000</td> <td>0 (Mute)</td> <td>$-\infty$</td> <td>H'FF_FFFF</td> <td>2.38×10^{-7}</td> <td>-132</td> </tr> </tbody> </table> | | | plus | | | minus | | | Value | [time] | [dB] | Value | [time] | [dB] | H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 | | | | | | | H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 | | | | | | | H'00_0001 | 2.38×10^{-7} | -132 | | | | H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |
| plus | | | minus | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Value | [time] | [dB] | Value | [time] | [dB] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

43.2.45 CTUn Scale Value e15 Register (CTUn_SV15R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV15R is a 32-bit readable/writable register that sets the scale value for channel 5 of matrix row 1.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE15[23:0] | H'00_0000 | R/W | Scale Value e15 for Input Channel 5 of Matrix Row 1 These bits set the scale value for input data of channel 5 of matrix row 1. SVE15[23]: Sign bit SVE15[22]: Integer bit SVE15[21:0]: Decimal bits |

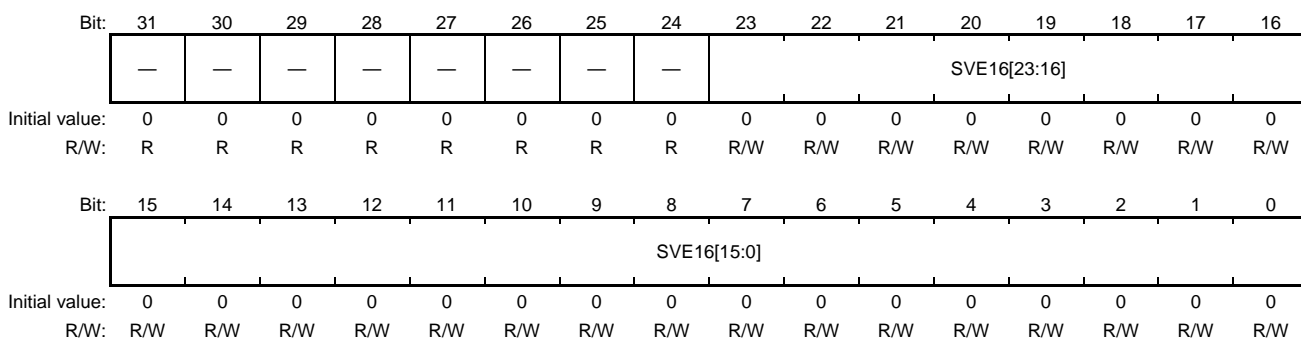
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.46 CTUn Scale Value e16 Register (CTUn_SV16R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV16R is a 32-bit readable/writable register that sets the scale value for channel 6 of matrix row 1.



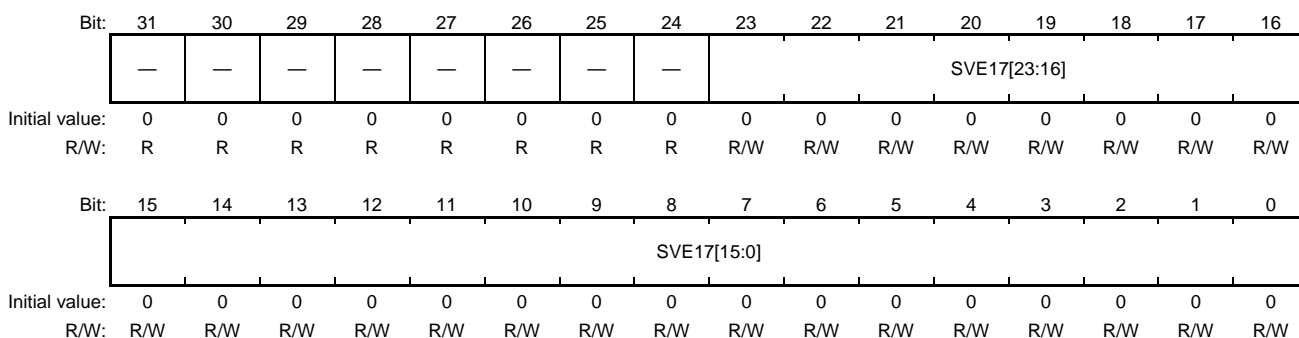
| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|-----------------------|--|-----------|--|-------|--|--|-------|--|--|-------|--------|------|-------|--------|------|-----------|---|---|-----------|---|---|-------|-------|-------|-------|-------|-------|-----------|---|---|-----------|---|---|-------|-------|-------|-------|-------|-------|-----------|-----------------------|------|-------|-------|-------|-----------|----------|-----------|-----------|-----------------------|------|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 23 to 0 | SVE16[23:0] | H'00_0000 | R/W | Scale Value e16 for Input Channel 6 of Matrix Row 1 These bits set the scale value for input data of channel 6 of matrix row 1. SVE16[23]: Sign bit SVE16[22]: Integer bit SVE16[21:0]: Decimal bits | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | <table border="1"> <thead> <tr> <th colspan="3">plus</th> <th colspan="3">minus</th> </tr> <tr> <th>Value</th> <th>[time]</th> <th>[dB]</th> <th>Value</th> <th>[time]</th> <th>[dB]</th> </tr> </thead> <tbody> <tr> <td>H'7F_FFFF</td> <td>2</td> <td>6</td> <td>H'80_0000</td> <td>2</td> <td>6</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'40_0000</td> <td>1</td> <td>0</td> <td>H'C0_0000</td> <td>1</td> <td>0</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'00_0001</td> <td>2.38×10^{-7}</td> <td>-132</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'00_0000</td> <td>0 (Mute)</td> <td>$-\infty$</td> <td>H'FF_FFFF</td> <td>2.38×10^{-7}</td> <td>-132</td> </tr> </tbody> </table> | | | plus | | | minus | | | Value | [time] | [dB] | Value | [time] | [dB] | H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 | | | | | | | H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 | | | | | | | H'00_0001 | 2.38×10^{-7} | -132 | | | | H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |
| plus | | | minus | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Value | [time] | [dB] | Value | [time] | [dB] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

43.2.47 CTUn Scale Value e17 Register (CTUn_SV17R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV17R is a 32-bit readable/writable register that sets the scale value for channel 7 of matrix row 1.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE17[23:0] | H'00_0000 | R/W | Scale Value e17 for Input Channel 7 of Matrix Row 1 These bits set the scale value for input data of channel 7 of matrix row 1. SVE17[23]: Sign bit SVE17[22]: Integer bit SVE17[21:0]: Decimal bits |

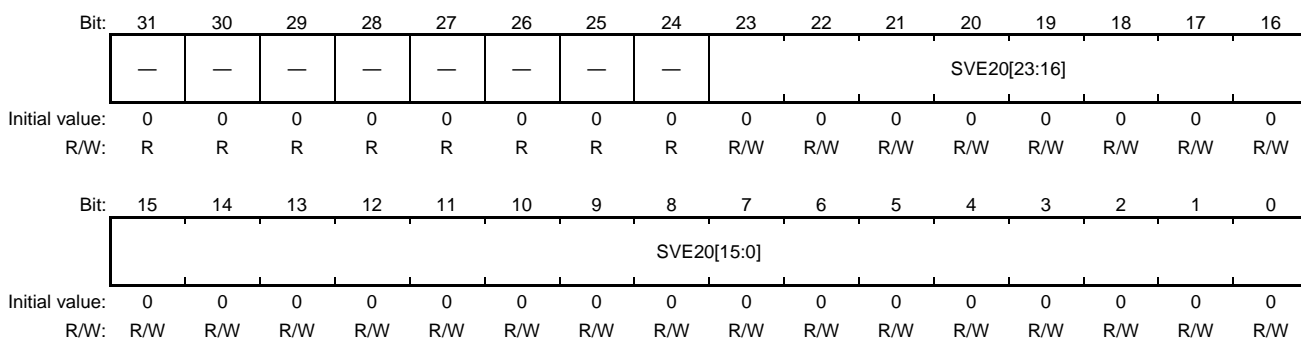
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.48 CTUn Scale Value e20 Register (CTUn_SV20R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV20R is a 32-bit readable/writable register that sets the scale value for channel 0 of matrix row 2.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE20[23:0] | H'00_0000 | R/W | Scale Value e20 for Input Channel 0 of Matrix Row 2 These bits set the scale value for input data of channel 0 of matrix row 2. SVE20[23]: Sign bit SVE20[22]: Integer bit SVE20[21:0]: Decimal bits |

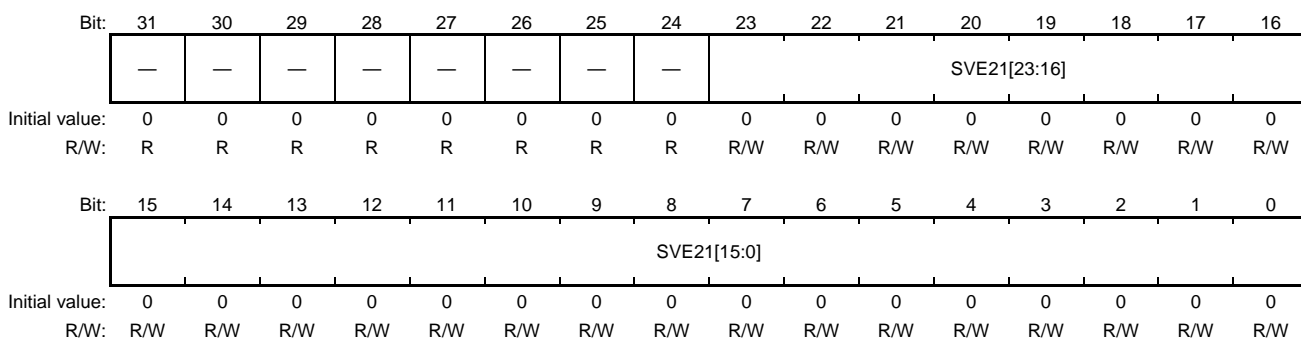
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.49 CTUn Scale Value e21 Register (CTUn_SV21R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV21R is a 32-bit readable/writable register that sets the scale value for channel 1 of matrix row 2.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE21[23:0] | H'00_0000 | R/W | Scale Value e21 for Input Channel 1 of Matrix Row 2 These bits set the scale value for input data of channel 1 of matrix row 2. SVE21[23]: Sign bit SVE21[22]: Integer bit SVE21[21:0]: Decimal bits |

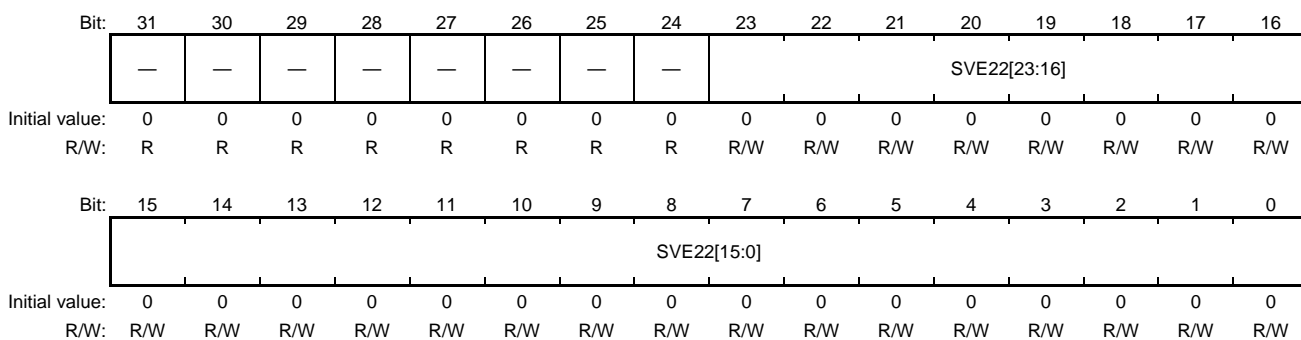
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.50 CTUn Scale Value e22 Register (CTUn_SV22R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV22R is a 32-bit readable/writable register that sets the scale value for channel 2 of matrix row 2.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE22[23:0] | H'00_0000 | R/W | Scale Value e22 for Input Channel 2 of Matrix Row 2 These bits set the scale value for input data of channel 2 of matrix row 2. SVE22[23]: Sign bit SVE22[22]: Integer bit SVE22[21:0]: Decimal bits |

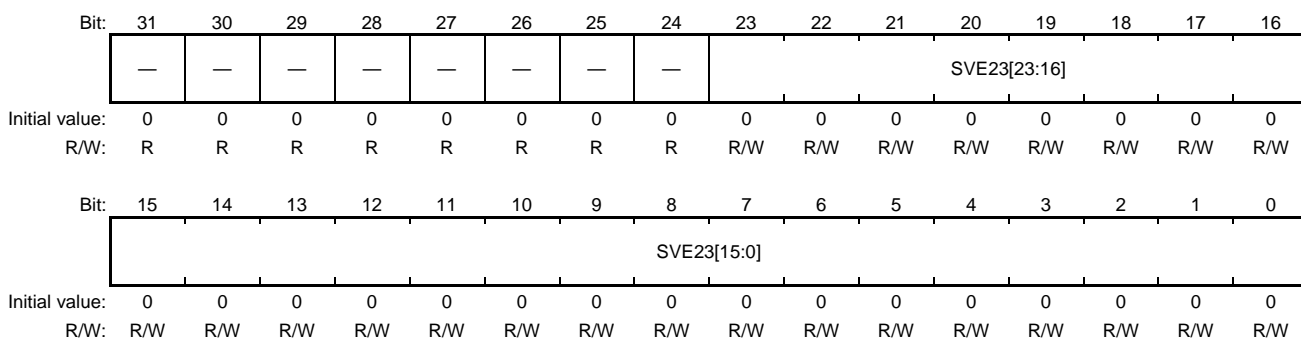
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.51 CTUn Scale Value e23 Register (CTUn_SV23R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV23R is a 32-bit readable/writable register that sets the scale value for channel 3 of matrix row 2.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE23[23:0] | H'00_0000 | R/W | Scale Value e23 for Input Channel 3 of Matrix Row 2 These bits set the scale value for input data of channel 3 of matrix row 2. SVE23[23]: Sign bit SVE23[22]: Integer bit SVE23[21:0]: Decimal bits |

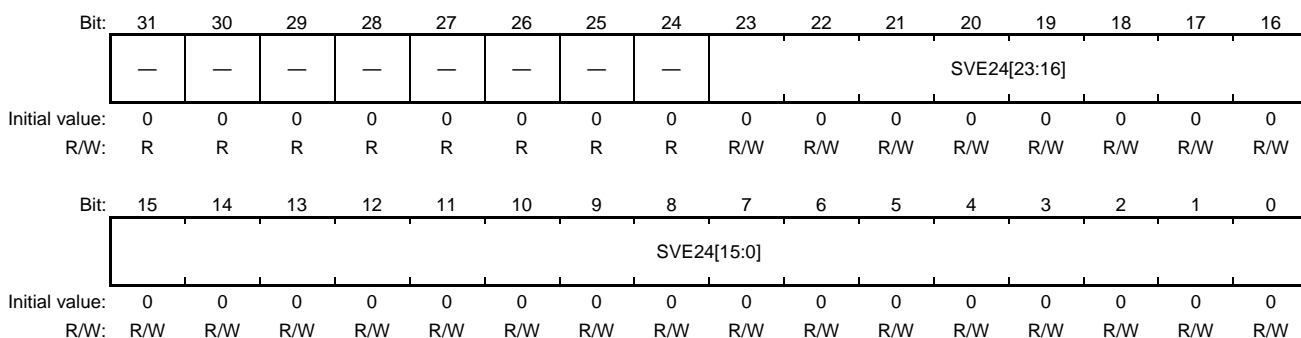
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.52 CTUn Scale Value e24 Register (CTUn_SV24R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV24R is a 32-bit readable/writable register that sets the scale value for channel 4 of matrix row 2.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE24[23:0] | H'00_0000 | R/W | Scale Value e24 for Input Channel 4 of Matrix Row 2 These bits set the scale value for input data of channel 4 of matrix row 2. SVE24[23]: Sign bit SVE24[22]: Integer bit SVE24[21:0]: Decimal bits |

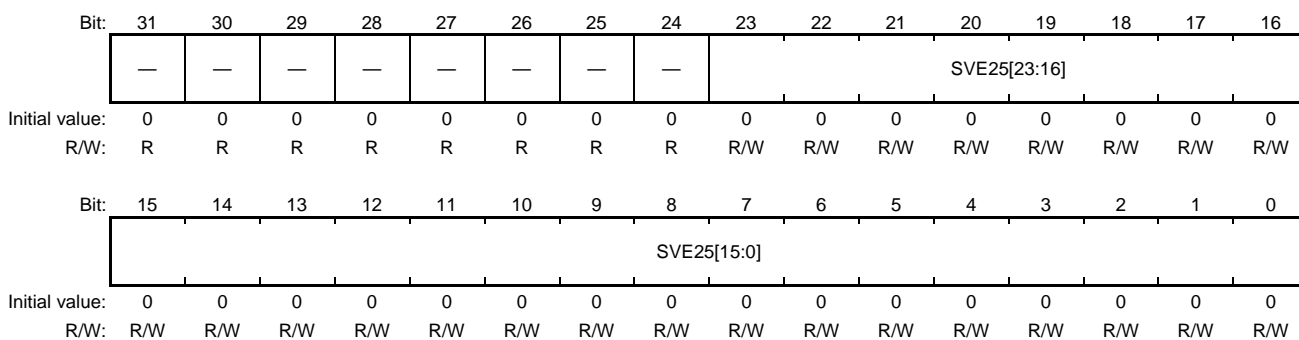
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.53 CTUn Scale Value e25 Register (CTUn_SV25R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV25R is a 32-bit readable/writable register that sets the scale value for channel 5 of matrix row 2.



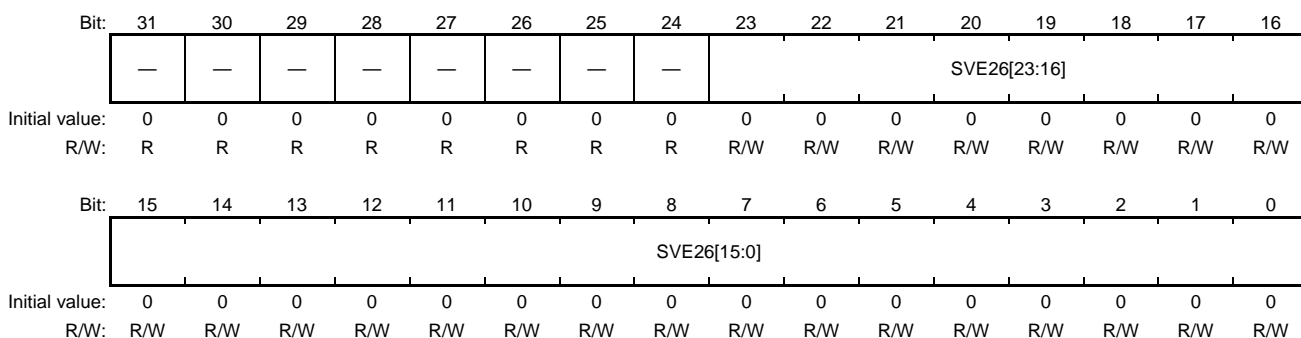
| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--|-----------------------|---------------|-----------|--|-------|--|--|-------|--|--|-------|--------|------|-------|--------|------|-----------|---|---|-----------|---|---|-------|-------|-------|-------|-------|-------|-----------|---|---|-----------|---|---|-------|-------|-------|-------|-------|-------|-----------|-----------------------|------|-------|-------|-------|-----------|----------|-----------|-----------|-----------------------|------|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 23 to 0 | SVE25[23:0] | H'00_0000 | R/W | Scale Value e25 for Input Channel 5 of Matrix Row 2 These bits set the scale value for input data of channel 5 of matrix row 2. SVE25[23]: Sign bit SVE25[22]: Integer bit SVE25[21:0]: Decimal bits | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <table border="1"> <thead> <tr> <th colspan="3">plus</th> <th colspan="3">minus</th> </tr> <tr> <th>Value</th> <th>[time]</th> <th>[dB]</th> <th>Value</th> <th>[time]</th> <th>[dB]</th> </tr> </thead> <tbody> <tr> <td>H'7F_FFFF</td> <td>2</td> <td>6</td> <td>H'80_0000</td> <td>2</td> <td>6</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'40_0000</td> <td>1</td> <td>0</td> <td>H'C0_0000</td> <td>1</td> <td>0</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'00_0001</td> <td>2.38×10^{-7}</td> <td>-132</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'00_0000</td> <td>0 (Mute)</td> <td>$-\infty$</td> <td>H'FF_FFFF</td> <td>2.38×10^{-7}</td> <td>-132</td> </tr> </tbody> </table> | | | | | plus | | | minus | | | Value | [time] | [dB] | Value | [time] | [dB] | H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 | | | | | | | H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 | | | | | | | H'00_0001 | 2.38×10^{-7} | -132 | | | | H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |
| plus | | | minus | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Value | [time] | [dB] | Value | [time] | [dB] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

43.2.54 CTUn Scale Value e26 Register (CTUn_SV26R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV26R is a 32-bit readable/writable register that sets the scale value for channel 6 of matrix row 2.



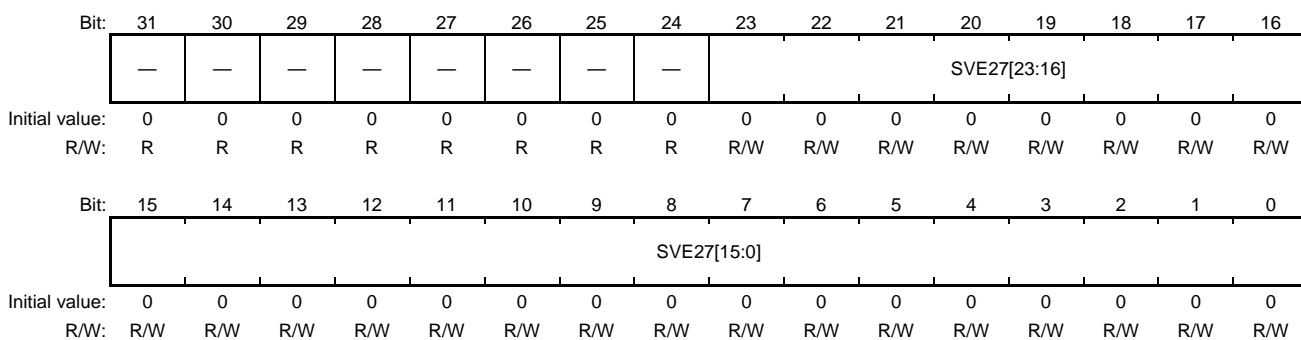
| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|-----------------------|--|-----------|--|-------|--|--|-------|--|--|-------|--------|------|-------|--------|------|-----------|---|---|-----------|---|---|-------|-------|-------|-------|-------|-------|-----------|---|---|-----------|---|---|-------|-------|-------|-------|-------|-------|-----------|-----------------------|------|-------|-------|-------|-----------|----------|-----------|-----------|-----------------------|------|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 23 to 0 | SVE26[23:0] | H'00_0000 | R/W | Scale Value e26 for Input Channel 6 of Matrix Row 2 These bits set the scale value for input data of channel 6 of matrix row 2. SVE26[23]: Sign bit SVE26[22]: Integer bit SVE26[21:0]: Decimal bits | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | <table border="1"> <thead> <tr> <th colspan="3">plus</th> <th colspan="3">minus</th> </tr> <tr> <th>Value</th> <th>[time]</th> <th>[dB]</th> <th>Value</th> <th>[time]</th> <th>[dB]</th> </tr> </thead> <tbody> <tr> <td>H'7F_FFFF</td> <td>2</td> <td>6</td> <td>H'80_0000</td> <td>2</td> <td>6</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'40_0000</td> <td>1</td> <td>0</td> <td>H'C0_0000</td> <td>1</td> <td>0</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'00_0001</td> <td>2.38×10^{-7}</td> <td>-132</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'00_0000</td> <td>0 (Mute)</td> <td>$-\infty$</td> <td>H'FF_FFFF</td> <td>2.38×10^{-7}</td> <td>-132</td> </tr> </tbody> </table> | | | plus | | | minus | | | Value | [time] | [dB] | Value | [time] | [dB] | H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 | | | | | | | H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 | | | | | | | H'00_0001 | 2.38×10^{-7} | -132 | | | | H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |
| plus | | | minus | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Value | [time] | [dB] | Value | [time] | [dB] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

43.2.55 CTUn Scale Value e27 Register (CTUn_SV27R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV27R is a 32-bit readable/writable register that sets the scale value for channel 7 of matrix row 2.



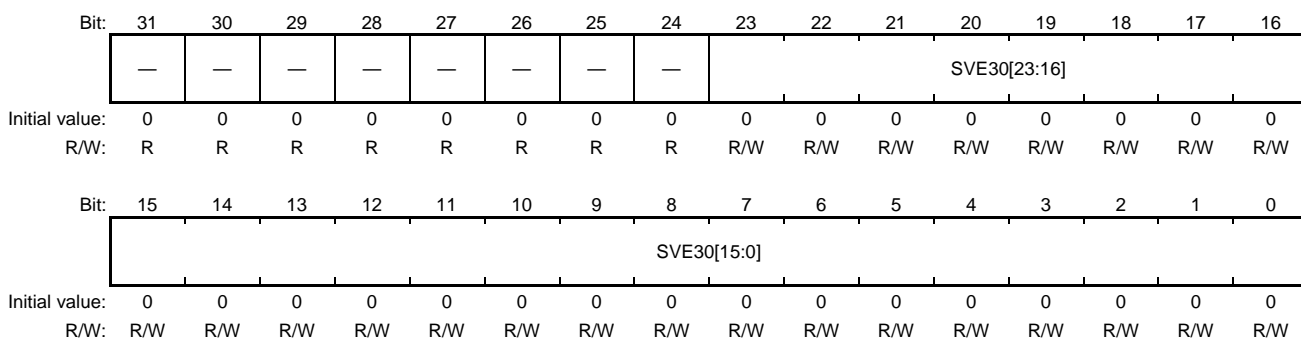
| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|-----------------------|--|-----------|--|-------|--|--|-------|--|--|-------|--------|------|-------|--------|------|-----------|---|---|-----------|---|---|-------|-------|-------|-------|-------|-------|-----------|---|---|-----------|---|---|-------|-------|-------|-------|-------|-------|-----------|-----------------------|------|-------|-------|-------|-----------|----------|-----------|-----------|-----------------------|------|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 23 to 0 | SVE27[23:0] | H'00_0000 | R/W | Scale Value e27 for Input Channel 7 of Matrix Row 2 These bits set the scale value for input data of channel 7 of matrix row 2. SVE27[23]: Sign bit SVE27[22]: Integer bit SVE27[21:0]: Decimal bits | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | <table border="1"> <thead> <tr> <th colspan="3">plus</th> <th colspan="3">minus</th> </tr> <tr> <th>Value</th> <th>[time]</th> <th>[dB]</th> <th>Value</th> <th>[time]</th> <th>[dB]</th> </tr> </thead> <tbody> <tr> <td>H'7F_FFFF</td> <td>2</td> <td>6</td> <td>H'80_0000</td> <td>2</td> <td>6</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'40_0000</td> <td>1</td> <td>0</td> <td>H'C0_0000</td> <td>1</td> <td>0</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'00_0001</td> <td>2.38×10^{-7}</td> <td>-132</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'00_0000</td> <td>0 (Mute)</td> <td>$-\infty$</td> <td>H'FF_FFFF</td> <td>2.38×10^{-7}</td> <td>-132</td> </tr> </tbody> </table> | | | plus | | | minus | | | Value | [time] | [dB] | Value | [time] | [dB] | H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 | | | | | | | H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 | | | | | | | H'00_0001 | 2.38×10^{-7} | -132 | | | | H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |
| plus | | | minus | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Value | [time] | [dB] | Value | [time] | [dB] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

43.2.56 CTUn Scale Value e30 Register (CTUn_SV30R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV30R is a 32-bit readable/writable register that sets the scale value for channel 0 of matrix row 3.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE30[23:0] | H'00_0000 | R/W | Scale Value e30 for Input Channel 0 of Matrix Row 3 These bits set the scale value for input data of channel 0 of matrix row 3. SVE30[23]: Sign bit SVE30[22]: Integer bit SVE30[21:0]: Decimal bits |

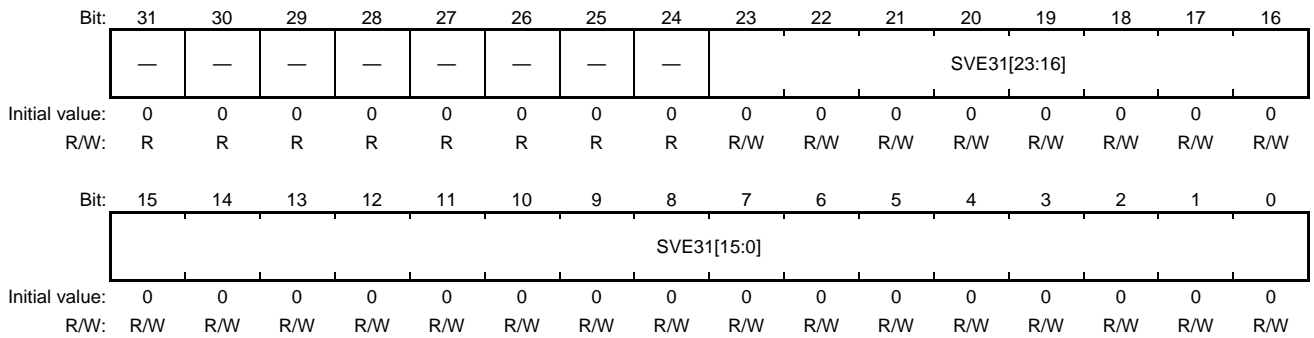
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.57 CTUn Scale Value e31 Register (CTUn_SV31R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV31R is a 32-bit readable/writable register that sets the scale value for channel 1 of matrix row 3.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE31[23:0] | H'00_0000 | R/W | Scale Value e31 for Input Channel 1 of Matrix Row 3 These bits set the scale value for input data of channel 1 of matrix row 3. SVE31[23]: Sign bit SVE31[22]: Integer bit SVE31[21:0]: Decimal bits |

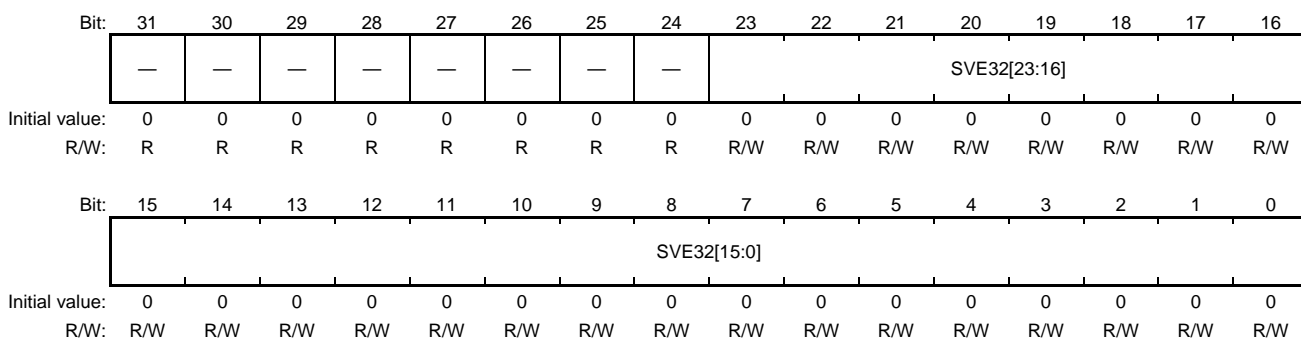
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.58 CTUn Scale Value e32 Register (CTUn_SV32R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV32R is a 32-bit readable/writable register that sets the scale value for channel 2 of matrix row 3.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE32[23:0] | H'00_0000 | R/W | Scale Value e32 for Input Channel 2 of Matrix Row 3 These bits set the scale value for input data of channel 2 of matrix row 3. SVE32[23]: Sign bit SVE32[22]: Integer bit SVE32[21:0]: Decimal bits |

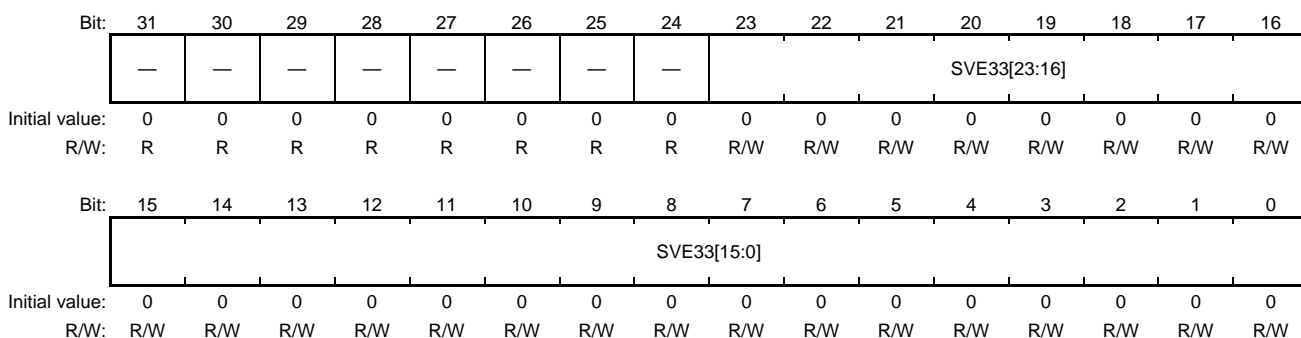
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.59 CTUn Scale Value e33 Register (CTUn_SV33R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV33R is a 32-bit readable/writable register that sets the scale value for channel 3 of matrix row 3.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE33[23:0] | H'00_0000 | R/W | Scale Value e33 for Input Channel 3 of Matrix Row 3 These bits set the scale value for input data of channel 3 of matrix row 3. SVE33[23]: Sign bit SVE33[22]: Integer bit SVE33[21:0]: Decimal bits |

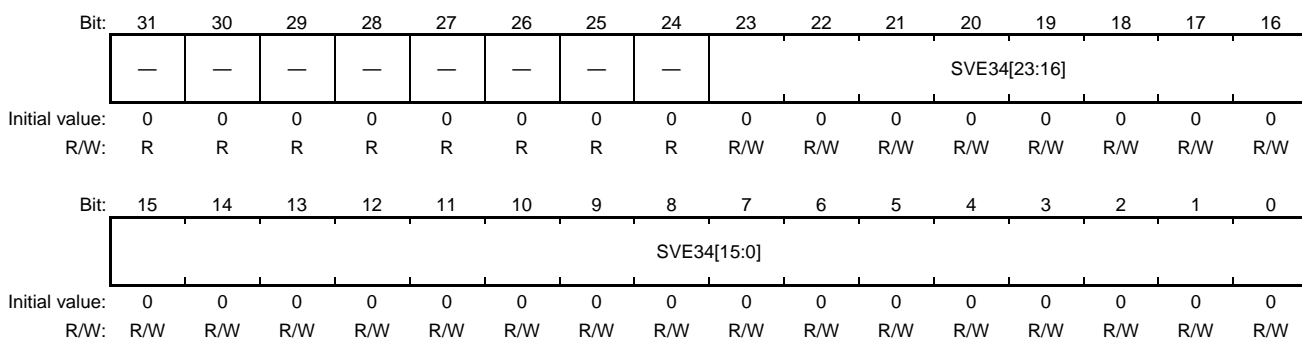
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.60 CTUn Scale Value e34 Register (CTUn_SV34R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV34R is a 32-bit readable/writable register that sets the scale value for channel 4 of matrix row 3.



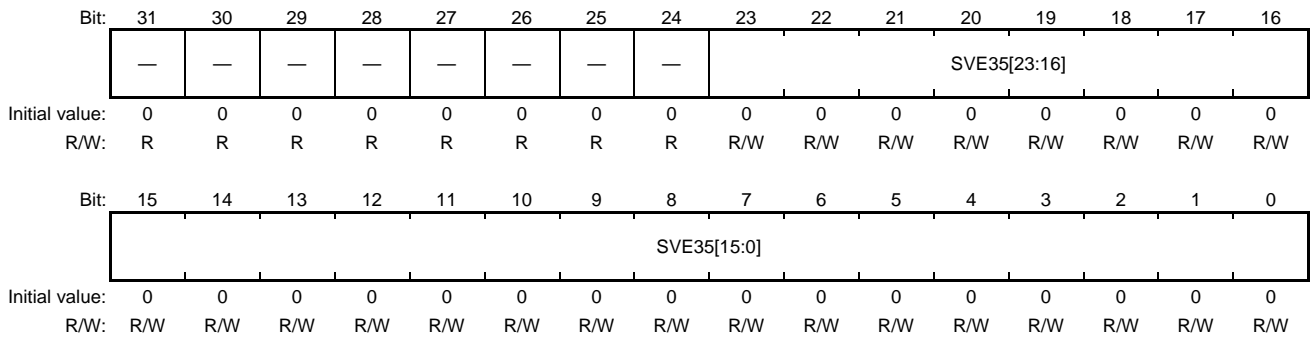
| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|-----------------------|--|-----------|--|-------|--|--|-------|--|--|-------|--------|------|-------|--------|------|-----------|---|---|-----------|---|---|-------|-------|-------|-------|-------|-------|-----------|---|---|-----------|---|---|-------|-------|-------|-------|-------|-------|-----------|-----------------------|------|-------|-------|-------|-----------|----------|-----------|-----------|-----------------------|------|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 23 to 0 | SVE34[23:0] | H'00_0000 | R/W | Scale Value e34 for Input Channel 4 of Matrix Row 3 These bits set the scale value for input data of channel 4 of matrix row 3. SVE34[23]: Sign bit SVE34[22]: Integer bit SVE34[21:0]: Decimal bits | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | <table border="1"> <thead> <tr> <th colspan="3">plus</th> <th colspan="3">minus</th> </tr> <tr> <th>Value</th> <th>[time]</th> <th>[dB]</th> <th>Value</th> <th>[time]</th> <th>[dB]</th> </tr> </thead> <tbody> <tr> <td>H'7F_FFFF</td> <td>2</td> <td>6</td> <td>H'80_0000</td> <td>2</td> <td>6</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'40_0000</td> <td>1</td> <td>0</td> <td>H'C0_0000</td> <td>1</td> <td>0</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'00_0001</td> <td>2.38×10^{-7}</td> <td>-132</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>H'00_0000</td> <td>0 (Mute)</td> <td>$-\infty$</td> <td>H'FF_FFFF</td> <td>2.38×10^{-7}</td> <td>-132</td> </tr> </tbody> </table> | | | plus | | | minus | | | Value | [time] | [dB] | Value | [time] | [dB] | H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 | | | | | | | H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 | | | | | | | H'00_0001 | 2.38×10^{-7} | -132 | | | | H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |
| plus | | | minus | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Value | [time] | [dB] | Value | [time] | [dB] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

43.2.61 CTUn Scale Value e35 Register (CTUn_SV35R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV35R is a 32-bit readable/writable register that sets the scale value for channel 5 of matrix row 3.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE35[23:0] | H'00_0000 | R/W | Scale Value e35 for Input Channel 5 of Matrix Row 3 These bits set the scale value for input data of channel 5 of matrix row 3. SVE35[23]: Sign bit SVE35[22]: Integer bit SVE35[21:0]: Decimal bits |

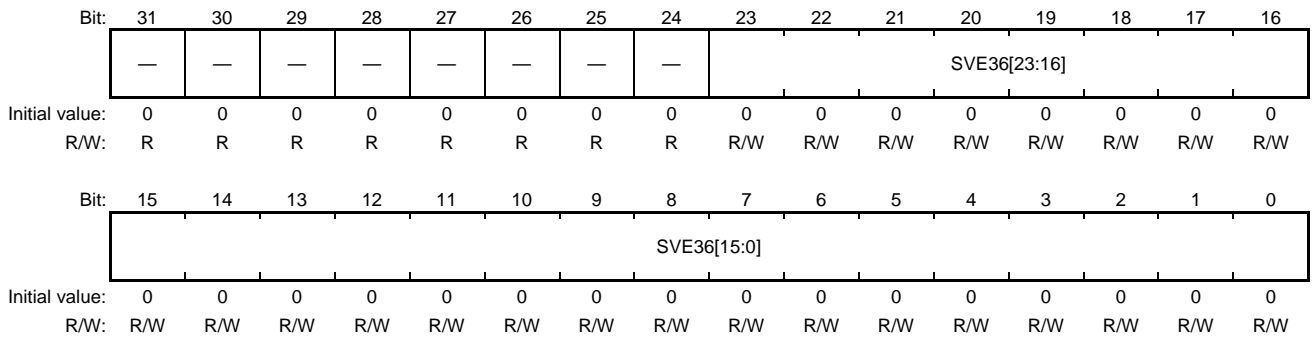
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.62 CTUn Scale Value e36 Register (CTUn_SV36R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV36R is a 32-bit readable/writable register that sets the scale value for channel 6 of matrix row 3.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE36[23:0] | H'00_0000 | R/W | Scale Value e36 for Input Channel 6 of Matrix Row 3 These bits set the scale value for input data of channel 6 of matrix row 3. SVE36[23]: Sign bit SVE36[22]: Integer bit SVE36[21:0]: Decimal bits |

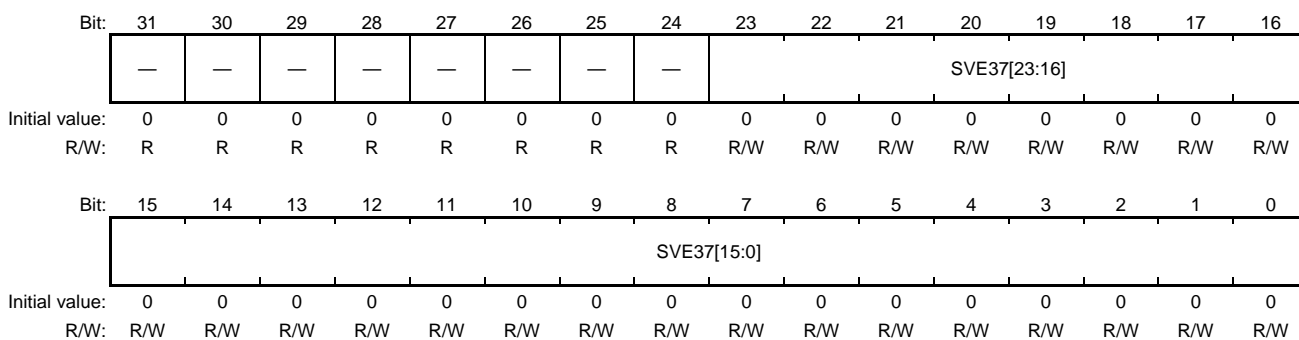
| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.63 CTUn Scale Value e37 Register (CTUn_SV37R)

Note: n = 00, 01, 02, 03, 10, 11, 12, or 13

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CTUn_SV37R is a 32-bit readable/writable register that sets the scale value for channel 7 of matrix row 3.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | SVE37[23:0] | H'00_0000 | R/W | Scale Value e37 for Input Channel 7 of Matrix Row 3 These bits set the scale value for input data of channel 7 of matrix row 3. SVE37[23]: Sign bit SVE37[22]: Integer bit SVE37[21:0]: Decimal bits |

| plus | | | minus | | |
|-----------|-----------------------|-----------|-----------|-----------------------|-------|
| Value | [time] | [dB] | Value | [time] | [dB] |
| H'7F_FFFF | 2 | 6 | H'80_0000 | 2 | 6 |
| | | | | | |
| H'40_0000 | 1 | 0 | H'C0_0000 | 1 | 0 |
| | | | | | |
| H'00_0001 | 2.38×10^{-7} | -132 | | | |
| H'00_0000 | 0 (Mute) | $-\infty$ | H'FF_FFFF | 2.38×10^{-7} | -132 |

43.2.64 MIXp Software Reset Register (MIXp_SWRSR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: MIXp_SWRSR is a 32-bit readable/writable register that controls operation/reset of the MIX internal circuits.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SWRST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | SWRST | B'1 | R/W | Software Reset While this bit is 0, the MIX internal circuits are put in the reset state. MIXp_* registers except this register are reset. Therefore, they should be set again after the reset is canceled. 0: Resets the MIX 1: Operating state |

43.2.65 MIXp MIX Initialization Register (MIXp_MIXIR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: MIXp_MIXIR is a 32-bit readable/writable register that initializes the operation of the MIX internal circuits.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | INIT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | INIT | B'1 | R/W | Initialization of Processing When this bit is set to 1, the MIX processing is initialized. This bit should be cleared to 0 after it was set to 1. 0: Processing State 1: Initialization (sets the initial setting of other registers) |

43.2.66 MIXp Audio Information Register (MIXp_ADINR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: MIXp_ADINR is a 32-bit readable/writable register that selects channel number.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CHNUM[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | CHNUM[3:0] | H'0 | R/W | Channel Number These bits set the output data channel number B'0000: 0 (None) B'0001: 1 channel B'0010: 2 channels B'0011: Reserved B'0100: 4 channels B'0101: Reserved B'0110: 6 channels B'0111: Reserved B'1000: 8 channels B'1001 to B'1111: Reserved |

43.2.67 MIXp MIX Mode Register (MIXp_MIXMR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: MIXp_MIXMR is a 32-bit readable/writable register that controls the mix mode.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | MIX MODE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | MIXMODE | B'0 | R/W | MIX Mode This bit controls the mix mode. 0: Selects volume step mixer 1: Selects volume ramp mixer |

43.2.68 MIXp MIX Volume Period Register (MIXp_MVPDR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: MIXp_MVPDR is a 32-bit readable/writable register that sets the value of the change of the volume a sample.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|--------------|-----|-----|-----|----|----|----|-------------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | MXPDPUP[3:0] | | | — | — | — | — | MXPDDW[3:0] | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 8 | MXPDPUP[3:0] | H'0 | R/W | MIX Period for Volume Up These bits set the decibel value that changed by one sample for volume up. And when MIXMODE bit of MIXp_MIXMR register is 1, this setting can be used. B'0000: 128 dB / 1 sample B'0001: 64 dB / 1 sample B'0010: 32 dB / 1 sample B'0011: 16 dB / 1 sample B'0100: 8 dB / 1 sample B'0101: 4 dB / 1 sample B'0110: 2 dB / 1 sample B'0111: 1 dB / 1 sample B'1000: 0.5 dB / 1 sample B'1001: 0.25 dB / 1 sample B'1010: 0.125 dB / 1 sample B'1011 to B'1111: Reserved |
| 7 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-------------|---------------|-----|---|
| 3 to 0 | MXPDDW[3:0] | H'0 | R/W | <p>MIX Period for Volume Down</p> <p>These bits set the decibel value that changed by one sample for volume down. And when MIXMODE bit of MIXp_MIXMR register is 1, this setting can be used.</p> <p>B'0000: -128 dB / 1 sample B'0001: -64 dB / 1 sample B'0010: -32 dB / 1 sample B'0011: -16 dB / 1 sample B'0100: -8 dB / 1 sample B'0101: -4 dB / 1 sample B'0110: -2 dB / 1 sample B'0111: -1 dB / 1 sample B'1000: -0.5 dB / 1 sample B'1001: -0.25 dB / 1 sample B'1010: -0.125 dB / 1 sample B'1011 to B'1111: Reserved</p> |

43.2.69 MIXp MIX Decibel A Register (MIXp_MDBAR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: MIXp_MDBAR is a 32-bit readable/writable register that sets the decibel (gain level) of system A.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | MIXDBA[9:0] | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 to 0 | MIXDBA[9:0] | H'000 | R/W | dB of System A These bits control the decibel (gain level) of system A. The value can select 1024 points from 0 dB to -∞ dB at intervals of 0.125 dB. |

| Value | [time] | [dB] | Value | [time] | [dB] |
|-------|--------|---------|-------|----------------------|---------|
| H'000 | 1 | 0 | | | |
| | | | H'091 | 0.125 | -18.125 |
| H'031 | 0.5 | -6.125 | | | |
| | | | H'3FE | 4.1×10^{-7} | -127.75 |
| H'061 | 0.25 | -12.125 | H'3FF | 0 (Mute) | -∞ |

The relation between the setting value and the decibel is shown in Table 43.11, Table 43.12, Table 43.13 and Table 43.14.

43.2.70 MIXp MIX Decibel B Register (MIXp_MDBBR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: MIXp_MDBBR is a 32-bit readable/writable register that sets the decibel (gain level) of system B.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | MIXDBB[9:0] | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|-------------|---------------|-------|---|-----------|--------|------|-------|--------|------|-------|---|---|-------|-------|-------|-------|-------|-------|-------|-------|---------|-------|-----|--------|-------|-------|-------|-------|-------|-------|-------|----------------------|---------|-------|------|---------|-------|----------|-----------|
| 31 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9 to 0 | MIXDBB[9:0] | H'000 | R/W | dB of System B These bits control the decibel (gain level) of system B. The value can select 1024 points from 0 dB to $-\infty$ dB at intervals of 0.125 dB. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <table border="1"> <thead> <tr> <th>Value</th> <th>[time]</th> <th>[dB]</th> <th>Value</th> <th>[time]</th> <th>[dB]</th> </tr> </thead> <tbody> <tr> <td>H'000</td> <td>1</td> <td>0</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>H'091</td> <td>0.125</td> <td>-18.125</td> </tr> <tr> <td>H'031</td> <td>0.5</td> <td>-6.125</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>H'3FE</td> <td>4.1×10^{-7}</td> <td>-127.75</td> </tr> <tr> <td>H'061</td> <td>0.25</td> <td>-12.125</td> <td>H'3FF</td> <td>0 (Mute)</td> <td>$-\infty$</td> </tr> </tbody> </table> | | | | | Value | [time] | [dB] | Value | [time] | [dB] | H'000 | 1 | 0 | | | | | | | H'091 | 0.125 | -18.125 | H'031 | 0.5 | -6.125 | | | | | | | H'3FE | 4.1×10^{-7} | -127.75 | H'061 | 0.25 | -12.125 | H'3FF | 0 (Mute) | $-\infty$ |
| Value | [time] | [dB] | Value | [time] | [dB] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'000 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | H'091 | 0.125 | -18.125 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'031 | 0.5 | -6.125 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | H'3FE | 4.1×10^{-7} | -127.75 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'061 | 0.25 | -12.125 | H'3FF | 0 (Mute) | $-\infty$ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| The relation between the setting value and the decibel is shown in Table 43.11, Table 43.12, Table 43.13 and Table 43.14. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

43.2.71 MIXp MIX Decibel C Register (MIXp_MDBCR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: MIXp_MDBCR is a 32-bit readable/writable register that sets the decibel (gain level) of system C.

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | MIXDBC[9:0] | | | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 to 0 | MIXDBC[9:0] | H'000 | R/W | dB of System C These bits control the decibel (gain level) of system C. The value can select 1024 points from 0 dB to -∞ dB at intervals of 0.125 dB. |

| Value | [time] | [dB] | Value | [time] | [dB] |
|-------|--------|---------|-------|----------------------|---------|
| H'000 | 1 | 0 | | | |
| | | | H'091 | 0.125 | -18.125 |
| H'031 | 0.5 | -6.125 | | | |
| | | | H'3FE | 4.1×10^{-7} | -127.75 |
| H'061 | 0.25 | -12.125 | H'3FF | 0 (Mute) | -∞ |

The relation between the setting value and the decibel is shown in Table 43.11, Table 43.12, Table 43.13 and Table 43.14.

43.2.72 MIXp MIX Decibel D Register (MIXp_MDBDR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: MIXp_MDBDR is a 32-bit readable/writable register that sets the decibel (gain level) of system D.

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | MIXDBD[9:0] | | | | | | | | | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |

| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|-------------|---------------|-------|--|---------|--------|------|-------|--------|------|-------|---|---|-------|-------|-------|-------|-------|-------|-------|-------|---------|-------|-----|--------|-------|-------|-------|-------|-------|-------|-------|----------------------|---------|-------|------|---------|-------|----------|----|
| 31 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9 to 0 | MIXDBD[9:0] | H'000 | R/W | dB of System D These bits control the decibel (gain level) of system D. The value can select 1024 points from 0 dB to -∞ dB at intervals of 0.125 dB. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <table border="1"> <thead> <tr> <th>Value</th> <th>[time]</th> <th>[dB]</th> <th>Value</th> <th>[time]</th> <th>[dB]</th> </tr> </thead> <tbody> <tr> <td>H'000</td> <td>1</td> <td>0</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>H'091</td> <td>0.125</td> <td>-18.125</td> </tr> <tr> <td>H'031</td> <td>0.5</td> <td>-6.125</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>H'3FE</td> <td>4.1×10^{-7}</td> <td>-127.75</td> </tr> <tr> <td>H'061</td> <td>0.25</td> <td>-12.125</td> <td>H'3FF</td> <td>0 (Mute)</td> <td>-∞</td> </tr> </tbody> </table> | | | | | Value | [time] | [dB] | Value | [time] | [dB] | H'000 | 1 | 0 | | | | | | | H'091 | 0.125 | -18.125 | H'031 | 0.5 | -6.125 | | | | | | | H'3FE | 4.1×10^{-7} | -127.75 | H'061 | 0.25 | -12.125 | H'3FF | 0 (Mute) | -∞ |
| Value | [time] | [dB] | Value | [time] | [dB] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'000 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | H'091 | 0.125 | -18.125 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'031 | 0.5 | -6.125 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | H'3FE | 4.1×10^{-7} | -127.75 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'061 | 0.25 | -12.125 | H'3FF | 0 (Mute) | -∞ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| The relation between the setting value and the decibel is shown in Table 43.11, Table 43.12, Table 43.13 and Table 43.14. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

43.2.73 MIXp MIX Decibel Enable Register (MIXp_MDBER)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: MIXp_MDBER is a 32-bit readable/writable register that controls the dB value.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | MIX DBEN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | MIXDBEN | B'0 | R/W | MIX dB Enable This bit controls the dB value that sets in MIXp_MDBAR, MIXp_MDBBR, MIXp_MDBCR and MIXp_MDBDR registers. 0: Disables the setting of dB 1: Enables the setting of dB |

43.2.74 MIXp MIX Status Register (MIXp_MIXSR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: MIXp_MIXSR is a 32-bit readable register that indicates the status of the volume ramp.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | MRPSTS | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1, 0 | MRPSTS | H'0 | R | MIX Volume Ramp Status These bits indicate the volume ramp status of mix operation. B'00: Level of volume ramp is stable B'01: Volume ramp down B'10: Volume ramp up Others: Setting prohibited. |

43.2.75 DVCp Software Reset Register (DVCp_SWRSR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_SWRSR is a 32-bit readable/writable register that controls operation/reset of the DVC internal circuits.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SWRST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | SWRST | B'1 | R/W | Software Reset While this bit is 0, the DVC internal circuits are put in the reset state. DVCp_* registers except this register are reset. Therefore, they should be set again after the reset is canceled. 0: Resets the DVC 1: Operating state |

43.2.76 DVCp DVU Initialization Register (DVCp_DVUIR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_DVUIR is a 32-bit readable/writable register that initializes the operation of the DVC internal circuits.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | INIT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | INIT | B'1 | R/W | Initialization of Processing When this bit is set to 1, the DVC processing is initialized. This bit should be cleared to 0 after it was set to 1. 0: Processing State 1: Initialization (sets the initial setting of other registers) |

43.2.77 DVCp Audio Information Register (DVCp_ADINR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_ADINR is a 32-bit readable/writable register that selects channel number and bit length of output audio data.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|------------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | OTBL[4:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | CHNUM[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 to 16 | OTBL[4:0] | H'00 | R/W | Bit Length of Output Audio Data. These bits set the bit length of output audio data. B'0_0000: 24 bits B'0_0001: Reserved B'0_0010: 22 bits B'0_0011: Reserved B'0_0100: 20 bits B'0_0101: Reserved B'0_0110: 18 bits B'0_0111: Reserved B'0_1000: 16 bits B'0_1001 to 0_1111: Reserved B'1_0000: 8 bits B'1_0001 to 1_1111: Reversed |
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------------|-----------------|----------------------|------------|--|
| 3 to 0 | CHNUM[3:0] | H'0 | R/W | Channel Number These bits set the output data channel number B'0000: 0 (None) B'0001: 1 channel B'0010: 2 channels B'0011: Reserved B'0100: 4 channels B'0101: Reserved B'0110: 6 channels B'0111: Reserved B'1000: 8 channels B'1001 to B'1111: Reserved |

43.2.78 DVCp DVU Control Register (DVCp_DVUCR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_DVUCR is a 32-bit readable/writable register that selects the mode of function to operate.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|------|----|----|----|------|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | VVMD | — | — | — | VRMD | — | — | — | ZCMD |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | VVMD | B'0 | R/W | Select Digital Volume Value Mode This bit selects the digital volume value function. 0: Sleep the digital volume value function 1: Use the digital volume value function |
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | VRMD | B'0 | R/W | Select Volume Ramp Mode This bit selects the volume ramp function. 0: Sleep the volume ramp function 1: Use the volume ramp function |
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | ZCMD | B'0 | R/W | Select Zero Cross Mute Mode This bit selects the zero cross mute function. 0: Sleep the zero cross mute function 1: Use the zero cross mute function |

43.2.79 DVCp Zero Cross Mute Control Register (DVCp_ZCMCR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_ZCMCR is a 32-bit readable/writable register that controls the operation of the zero cross mute function for each channel.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | ZCEN7 | ZCEN6 | ZCEN5 | ZCEN4 | ZCEN3 | ZCEN2 | ZCEN1 | ZCEN0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | ZCEN7 | B'0 | R/W | Zero Cross Mute Enable for Channel 7 This bit controls the operation of the zero cross mute function for channel 7. 0: Disables the operation of the zero cross mute function 1: Enables the operation of the zero cross mute function |
| 6 | ZCEN6 | B'0 | R/W | Zero Cross Mute Enable for Channel 6 This bit controls the operation of the zero cross mute function for channel 6. 0: Disables the operation of the zero cross mute function 1: Enables the operation of the zero cross mute function |
| 5 | ZCEN5 | B'0 | R/W | Zero Cross Mute Enable for Channel 5 This bit controls the operation of the zero cross mute function for channel 5. 0: Disables the operation of the zero cross mute function 1: Enables the operation of the zero cross mute function |
| 4 | ZCEN4 | B'0 | R/W | Zero Cross Mute Enable for Channel 4 This bit controls the operation of the zero cross mute function for channel 4. 0: Disables the operation of the zero cross mute function 1: Enables the operation of the zero cross mute function |
| 3 | ZCEN3 | B'0 | R/W | Zero Cross Mute Enable for Channel 3 This bit controls the operation of the zero cross mute function for channel 3. 0: Disables the operation of the zero cross mute function 1: Enables the operation of the zero cross mute function |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | ZCEN2 | B'0 | R/W | Zero Cross Mute Enable for Channel 2 This bit controls the operation of the zero cross mute function for channel 2. 0: Disables the operation of the zero cross mute function 1: Enables the operation of the zero cross mute function |
| 1 | ZCEN1 | B'0 | R/W | Zero Cross Mute Enable for Channel 1 This bit controls the operation of the zero cross mute function for channel 1. 0: Disables the operation of the zero cross mute function 1: Enables the operation of the zero cross mute function |
| 0 | ZCEN0 | B'0 | R/W | Zero Cross Mute Enable for Channel 0 This bit controls the operation of the zero cross mute function for channel 0. 0: Disables the operation of the zero cross mute function 1: Enables the operation of the zero cross mute function |

43.2.80 DVCp Volume Ramp Control Register (DVCp_VRCTR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_VRCTR is a 32-bit readable/writable register that controls the operation of the volume ramp function for each channel.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | VREN7 | VREN6 | VREN5 | VREN4 | VREN3 | VREN2 | VREN1 | VREN0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | VREN7 | B'0 | R/W | Volume Ramp Enable for Channel 7 This bit controls the operation of the volume ramp function for channel 7. 0: Disables the operation of the volume ramp function 1: Enables the operation of the volume ramp function |
| 6 | VREN6 | B'0 | R/W | Volume Ramp Enable for Channel 6 This bit controls the operation of the volume ramp function for channel 6. 0: Disables the operation of the volume ramp function 1: Enables the operation of the volume ramp function |
| 5 | VREN5 | B'0 | R/W | Volume Ramp Enable for Channel 5 This bit controls the operation of the volume ramp function for channel 5. 0: Disables the operation of the volume ramp function 1: Enables the operation of the volume ramp function |
| 4 | VREN4 | B'0 | R/W | Volume Ramp Enable for Channel 4 This bit controls the operation of the volume ramp function for channel 4. 0: Disables the operation of the volume ramp function 1: Enables the operation of the volume ramp function |
| 3 | VREN3 | B'0 | R/W | Volume Ramp Enable for Channel 3 This bit controls the operation of the volume ramp function for channel 3. 0: Disables the operation of the volume ramp function 1: Enables the operation of the volume ramp function |
| 2 | VREN2 | B'0 | R/W | Volume Ramp Enable for Channel 2 This bit controls the operation of the volume ramp function for channel 2. 0: Disables the operation of the volume ramp function 1: Enables the operation of the volume ramp function |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | VREN1 | B'0 | R/W | Volume Ramp Enable for Channel 1 This bit controls the operation of the volume ramp function for channel 1. 0: Disables the operation of the volume ramp function 1: Enables the operation of the volume ramp function |
| 0 | VREN0 | B'0 | R/W | Volume Ramp Enable for Channel 0 This bit controls the operation of the volume ramp function for channel 0. 0: Disables the operation of the volume ramp function 1: Enables the operation of the volume ramp function |

Note: All of these bits should be set to 1 when VRMD bit of DVCP_DVUCR register is 1.

43.2.81 DVCp Volume Ramp Period Register (DVCp_VRPDR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_VRPDR is a 32-bit readable/writable register that controls.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-------------|-----|-----|-----|-----|----|----|-------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | VRPDUP[4:0] | | | | — | — | — | VRPDDW[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 8 | VRPDUP[4:0] | H'00 | R/W | Volume Ramp Period for Volume Up B'0_0000: 1 [sample] (128 dB/1 step) B'0_0001: 2 [sample] (64 dB/1 step) B'0_0010: 4 [sample] (32 dB/1 step) B'0_0011: 8 [sample] (16 dB/1 step) B'0_0100: 16 [sample] (8 dB/1 step) B'0_0101: 32 [sample] (4 dB/1 step) B'0_0110: 64 [sample] (2 dB/1 step) B'0_0111: 128 [sample] (1 dB/1 step) B'0_1000: 256 [sample] (0.5 dB/1 step) B'0_1001: 512 [sample] (0.25 dB/1 step) B'0_1010: 1024 [sample] (0.125 dB/1 step) B'0_1011: 2048 [sample] (0.125 dB/2 steps) B'0_1100: 4096 [sample] (0.125 dB/4 steps) B'0_1101: 8192 [sample] (0.125 dB/8 steps) B'0_1110: 16384 [sample] (0.125 dB/16 steps) B'0_1111: 32768 [sample] (0.125 dB/32 steps) B'1_0000: 65536 [sample] (0.125 dB/64 steps) B'1_0001: 131072 [sample] (0.125 dB/128 steps) B'1_0010: 262144 [sample] (0.125 dB/256 steps) B'1_0011: 524288 [sample] (0.125 dB/512 steps) B'1_0100: 1048576 [sample] (0.125 dB/1024 steps) B'1_0101: 2097152 [sample] (0.125 dB/2048 steps) B'1_0110: 4194304 [sample] (0.125 dB/4096 steps) B'1_0111: 8388608 [sample] (0.125 dB/8192 steps) B'1_1000 to B'1_1111: Reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-------------|---------------|-----|--|
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | VRPDDW[4:0] | H'00 | R/W | Volume Ramp Period for Volume Down B'0_0000: 1 [sample] (-128 dB/1 step) B'0_0001: 2 [sample] (-64 dB/1 step) B'0_0010: 4 [sample] (-32 dB/1 step) B'0_0011: 8 [sample] (-16 dB/1 step) B'0_0100: 16 [sample] (-8 dB/1 step) B'0_0101: 32 [sample] (-4 dB/1 step) B'0_0110: 64 [sample] (-2 dB/1 step) B'0_0111: 128 [sample] (-1 dB/1 step) B'0_1000: 256 [sample] (-0.5 dB/1 step) B'0_1001: 512 [sample] (-0.25 dB/1 step) B'0_1010: 1024 [sample] (-0.125 dB/1 step) B'0_1011: 2048 [sample] (-0.125 dB/2 steps) B'0_1100: 4096 [sample] (-0.125 dB/4 steps) B'0_1101: 8192 [sample] (-0.125 dB/8 steps) B'0_1110: 16384 [sample] (-0.125 dB/16 steps) B'0_1111: 32768 [sample] (-0.125 dB/32 steps) B'1_0000: 65536 [sample] (-0.125 dB/64 steps) B'1_0001: 131072 [sample] (-0.125 dB/128 steps) B'1_0010: 262144 [sample] (-0.125 dB/256 steps) B'1_0011: 524288 [sample] (-0.125 dB/512 steps) B'1_0100: 1048576 [sample] (-0.125 dB/1024 steps) B'1_0101: 2097152 [sample] (-0.125 dB/2048 steps) B'1_0110: 4194304 [sample] (-0.125 dB/4096 steps) B'1_0111: 8388608 [sample] (-0.125 dB/8192 steps) B'1_1000 to B'1_1111: Reserved |

43.2.82 DVCp Volume Ramp Decibel Register (DVCp_VRDBR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_VRDBR is a 32-bit readable/writable register that sets the decibel (gain level) of volume ramp.

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | — | VRDB[9:0] | | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 to 0 | VRDB[9:0] | H'000 | R/W | dB of Volume Ramp These bits control the decibel (gain level) of volume ramp. The value can select 1024 points from 0 dB to -∞ dB at intervals of 0.125 dB. |

| Value | [time] | [dB] | Value | [time] | [dB] |
|-------|--------|---------|-------|----------------------|---------|
| H'000 | 1 | 0 | | | |
| | | | H'091 | 0.125 | -18.125 |
| H'031 | 0.5 | -6.125 | | | |
| | | | H'3FE | 4.1×10^{-7} | -127.75 |
| H'061 | 0.25 | -12.125 | H'3FF | 0 (Mute) | -∞ |

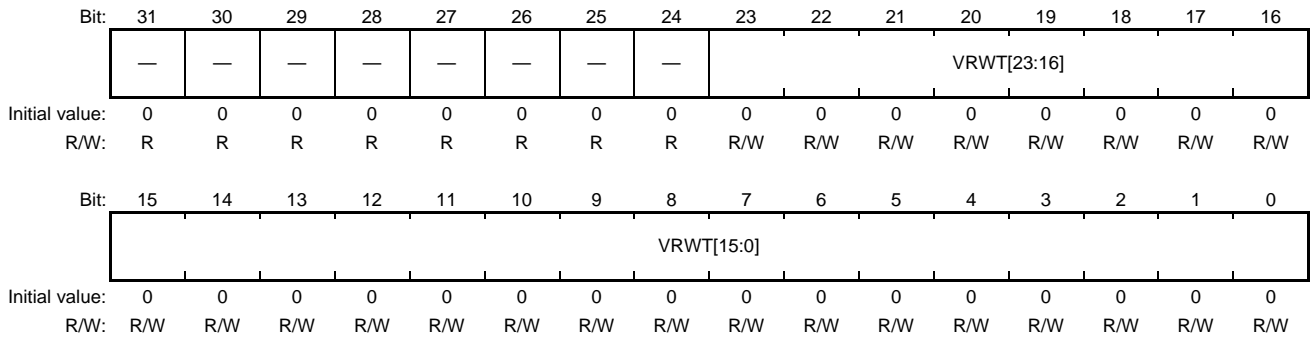
The relation between the setting value and the decibel is shown in Table 43.11, Table 43.12, Table 43.13 and Table 43.14.

43.2.83 DVCp Volume Ramp Wait Time Register (DVCp_VRWTR)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_VRWTR is a 32-bit readable/writable register that sets the standby time to start the operation of the volume ramp function.



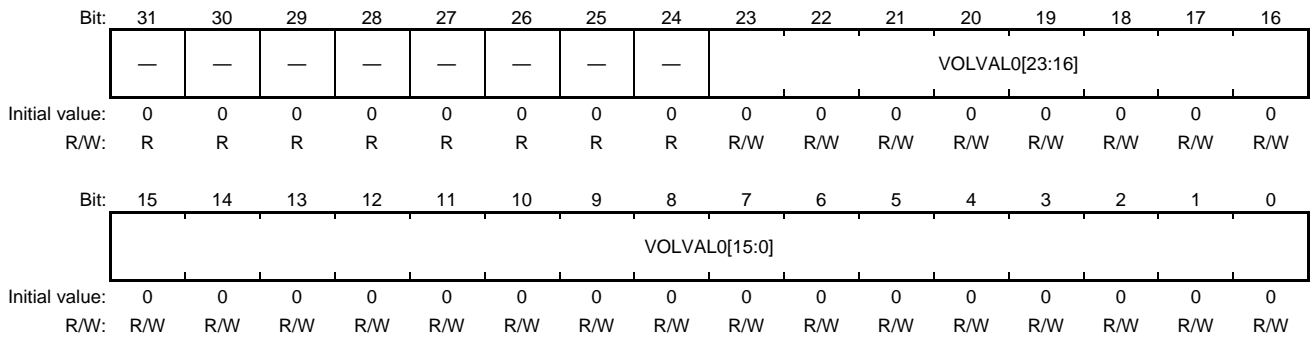
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | VRWT[23:0] | H'00_0000 | R/W | Volume Ramp Wait Time These bits set the standby time to adjust the start timing of operation of the volume ramp function when the setting of DVCp_VRDBR register is changed. If the internal counter of the DVC logic reached to the value of these bits, the volume ramp function starts to operate to change the volume to target volume of DVCp_VRDBR register. |

43.2.84 DVCp Volume Value Setting 0 Register (DVCp_VOL0R)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_VOL0R is a 32-bit readable/writable register that sets digital volume value for channel 0.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | VOLVAL0 [23:0] | H'00_0000 | R/W | Digital Volume Value for Channel 0 These bits set the digital volume of channel 0. The maximum value is 8-time (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL0[23]: Sign bit (The write value should be 0.) VOLVAL0[22:20]: Integer bits VOLVAL0[19:0]: Decimal bits |

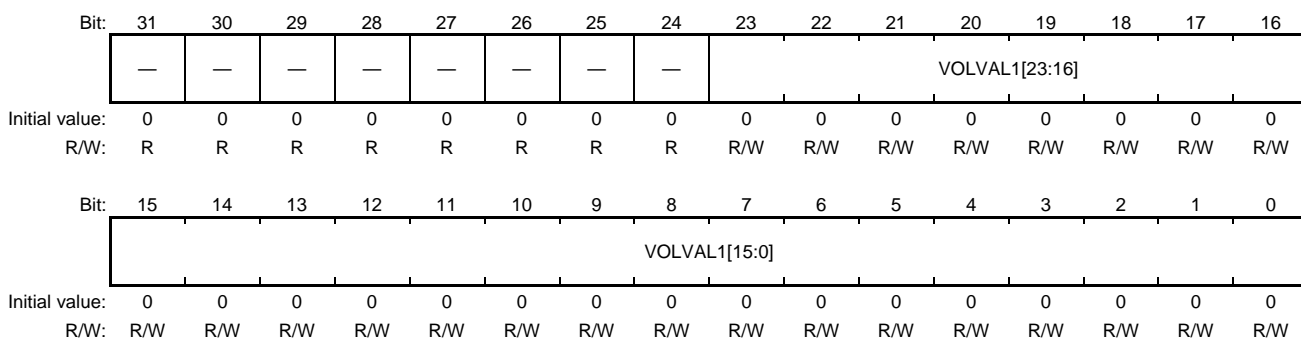
| Value | [time] | [dB] | Value | [time] | [dB] |
|-----------|--------|------|-----------|----------------------|------|
| H'7F_FFFF | 8 | 18 | H'08_0000 | 0.5 | -6 |
| ... | ... | ... | ... | ... | ... |
| H'10_0000 | 1 | 0 | H'00_0001 | 9.5×10^{-7} | -120 |
| ... | ... | ... | H'00_0000 | 0 | -∞ |

43.2.85 DVCp Volume Value Setting 1 Register (DVCp_VOL1R)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_VOL1R is a 32-bit readable/writable register that sets digital volume value for channel 1.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | VOLVAL1 [23:0] | H'00_0000 | R/W | Digital Volume Value for Channel 1 These bits set the digital volume of channel 1. The maximum value is 8-time (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL1[23]: Sign bit (The write value should be 0.) VOLVAL1[22:20]: Integer bits VOLVAL1[19:0]: Decimal bits |

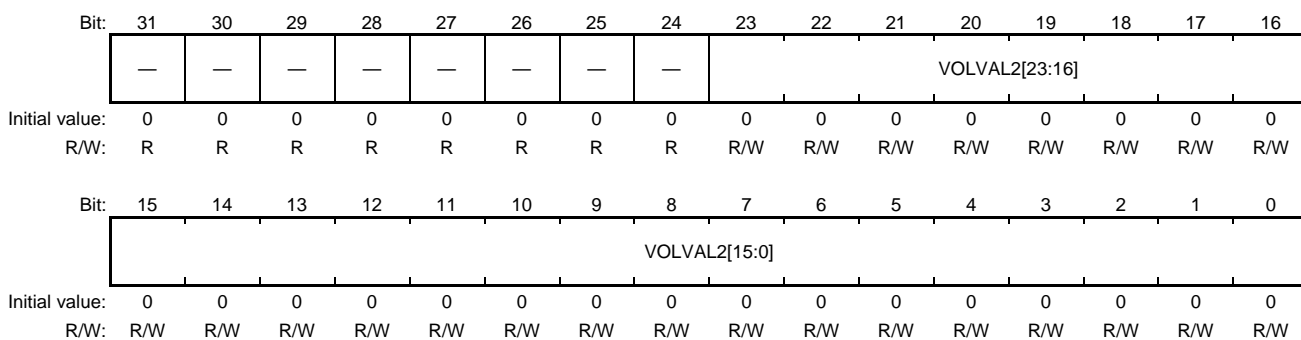
| Value | [time] | [dB] | Value | [time] | [dB] |
|-----------|--------|------|-----------|----------------------|------|
| H'7F_FFFF | 8 | 18 | H'08_0000 | 0.5 | -6 |
| ... | ... | ... | ... | ... | ... |
| H'10_0000 | 1 | 0 | H'00_0001 | 9.5×10^{-7} | -120 |
| ... | ... | ... | H'00_0000 | 0 | -∞ |

43.2.86 DVCp Volume Value Setting 2 Register (DVCp_VOL2R)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_VOL2R is a 32-bit readable/writable register that sets digital volume value for channel 2.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | VOLVAL2 [23:0] | H'00_0000 | R/W | Digital Volume Value for Channel 2 These bits set the digital volume of channel 2. The maximum value is 8-time (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL2[23]: Sign bit (The write value should be 0.) VOLVAL2[22:20]: Integer bits VOLVAL2[19:0]: Decimal bits |

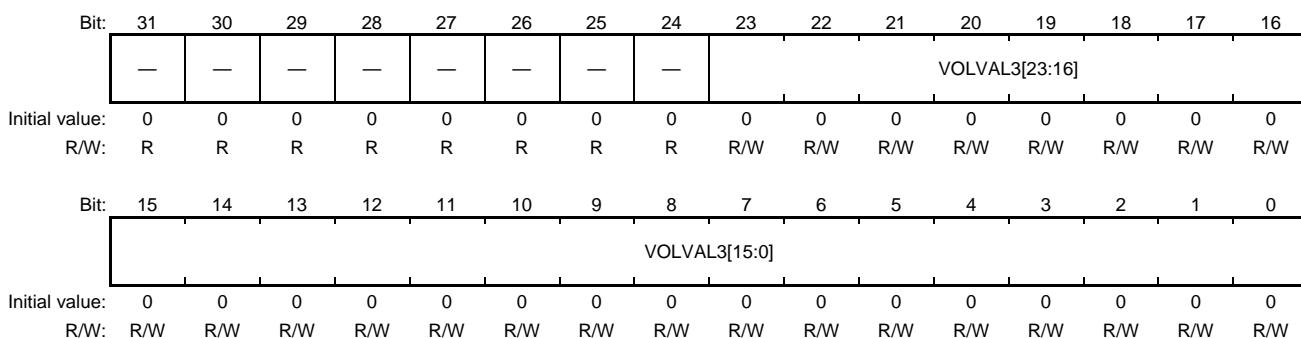
| Value | [time] | [dB] | Value | [time] | [dB] |
|-----------|--------|------|-----------|----------------------|------|
| H'7F_FFFF | 8 | 18 | H'08_0000 | 0.5 | -6 |
| ... | ... | ... | ... | ... | ... |
| H'10_0000 | 1 | 0 | H'00_0001 | 9.5×10^{-7} | -120 |
| ... | ... | ... | H'00_0000 | 0 | -∞ |

43.2.87 DVCp Volume Value Setting 3 Register (DVCp_VOL3R)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_VOL3R is a 32-bit readable/writable register that sets digital volume value for channel 3.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | VOLVAL3 [23:0] | H'00_0000 | R/W | Digital Volume Value for Channel 3 These bits set the digital volume of channel 3. The maximum value is 8-time (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL3[23]: Sign bit (The write value should be 0.) VOLVAL3[22:20]: Integer bits VOLVAL3[19:0]: Decimal bits |

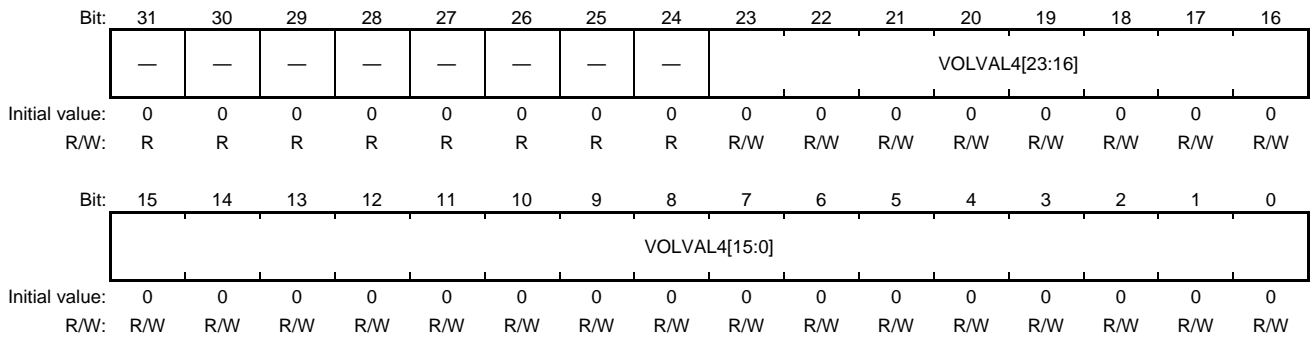
| Value | [time] | [dB] | Value | [time] | [dB] |
|-----------|--------|------|-----------|----------------------|------|
| H'7F_FFFF | 8 | 18 | H'08_0000 | 0.5 | -6 |
| ... | ... | ... | ... | ... | ... |
| H'10_0000 | 1 | 0 | H'00_0001 | 9.5×10^{-7} | -120 |
| ... | ... | ... | H'00_0000 | 0 | -∞ |

43.2.88 DVCp Volume Value Setting 4 Register (DVCp_VOL4R)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_VOL4R is a 32-bit readable/writable register that sets digital volume value for channel 4.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | VOLVAL4 [23:0] | H'00_0000 | R/W | Digital Volume Value for Channel 4 These bits set the digital volume of channel 4. The maximum value is 8-time (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL4[23]: Sign bit (The write value should be 0.) VOLVAL4[22:20]: Integer bits VOLVAL4[19:0]: Decimal bits |

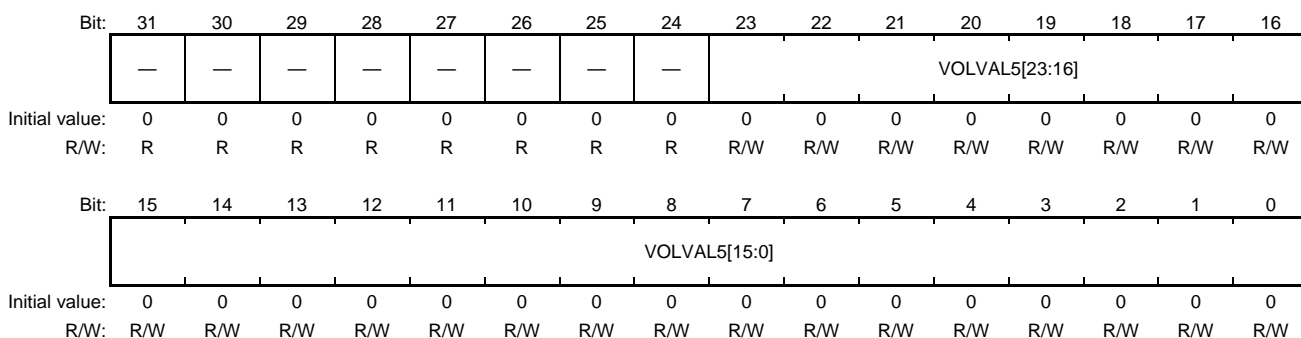
| Value | [time] | [dB] | Value | [time] | [dB] |
|-----------|--------|------|-----------|----------------------|------|
| H'7F_FFFF | 8 | 18 | H'08_0000 | 0.5 | -6 |
| ... | ... | ... | ... | ... | ... |
| H'10_0000 | 1 | 0 | H'00_0001 | 9.5×10^{-7} | -120 |
| ... | ... | ... | H'00_0000 | 0 | -∞ |

43.2.89 DVCp Volume Value Setting 5 Register (DVCp_VOL5R)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_VOL5R is a 32-bit readable/writable register that sets digital volume value for channel 5.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | VOLVAL5 [23:0] | H'00_0000 | R/W | Digital Volume Value for Channel 5 These bits set the digital volume of channel 5. The maximum value is 8-time (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL5[23]: Sign bit (The write value should be 0.) VOLVAL5[22:20]: Integer bits VOLVAL5[19:0]: Decimal bits |

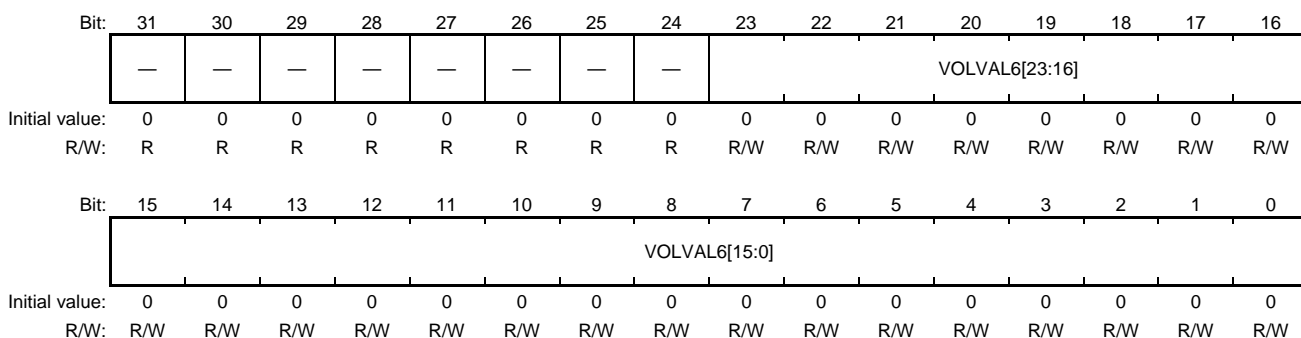
| Value | [time] | [dB] | Value | [time] | [dB] |
|-----------|--------|------|-----------|----------------------|------|
| H'7F_FFFF | 8 | 18 | H'08_0000 | 0.5 | -6 |
| ... | ... | ... | ... | ... | ... |
| H'10_0000 | 1 | 0 | H'00_0001 | 9.5×10^{-7} | -120 |
| ... | ... | ... | H'00_0000 | 0 | -∞ |

43.2.90 DVCp Volume Value Setting 6 Register (DVCp_VOL6R)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_VOL6R is a 32-bit readable/writable register that sets digital volume value for channel 6.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | VOLVAL6 [23:0] | H'00_0000 | R/W | Digital Volume Value for Channel 6 These bits set the digital volume of channel 6. The maximum value is 8-time (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL6[23]: Sign bit (The write value should be 0.) VOLVAL6[22:20]: Integer bits VOLVAL6[19:0]: Decimal bits |

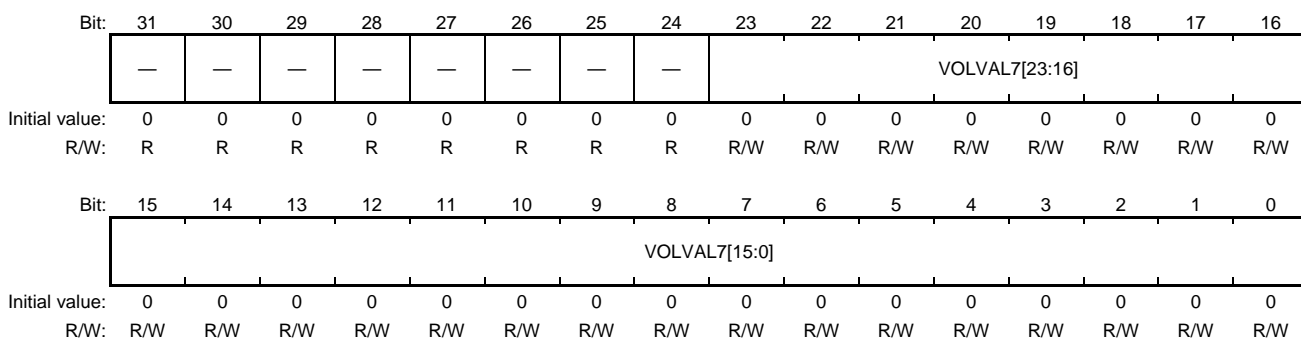
| Value | [time] | [dB] | Value | [time] | [dB] |
|-----------|--------|------|-----------|----------------------|------|
| H'7F_FFFF | 8 | 18 | H'08_0000 | 0.5 | -6 |
| ... | ... | ... | ... | ... | ... |
| H'10_0000 | 1 | 0 | H'00_0001 | 9.5×10^{-7} | -120 |
| ... | ... | ... | H'00_0000 | 0 | -∞ |

43.2.91 DVCp Volume Value Setting 7 Register (DVCp_VOL7R)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_VOL7R is a 32-bit readable/writable register that sets digital volume value for channel 7.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | VOLVAL7 [23:0] | H'00_0000 | R/W | Digital Volume Value for Channel 7 These bits set the digital volume of channel 7. The maximum value is 8-time (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL7[23]: Sign bit (The write value should be 0.) VOLVAL7[22:20]: Integer bits VOLVAL7[19:0]: Decimal bits |

| Value | [time] | [dB] | Value | [time] | [dB] |
|-----------|--------|------|-----------|----------------------|------|
| H'7F_FFFF | 8 | 18 | H'08_0000 | 0.5 | -6 |
| ... | ... | ... | ... | ... | ... |
| H'10_0000 | 1 | 0 | H'00_0001 | 9.5×10^{-7} | -120 |
| ... | ... | ... | H'00_0000 | 0 | -∞ |

43.2.92 DVCp DVU Enable Register (DVCp_DVUER)

Note: p = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_DVUER is a 32-bit readable/writable register that controls the setting of DVC registers.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | DVCEN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | DVCEN | B'0 | R/W | DVC Register Setting Enable This bit controls the setting of DVC registers (DVCp_ZCMCR, DVCp_VRCTR, DVCp_VRPDR, DVCp_VRDBR, DVCp_VOL0R, DVCp_VOL1R, DVCp_VOL2R, DVCp_VOL3R, DVCp_VOL4R, DVCp_VOL5R, DVCp_VOL6R, DVCp_VOL7R). 0: Disables the setting of dvc registers to DVC logic 1: Enables the setting of dvc registers to DVC logic |

43.2.93 DVCp DVU Status Register (DVCp_DVUSR)

Note: p = 0, 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_DVUSR is a 32-bit readable register that indicates the status of the zero cross mute and the volume ramp. This register is used for debug.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-------------|-------|-------|-------|-----------------|----------------|------------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | ALLZS TS | ZSTS7 | ZSTS6 | ZSTS5 | ZSTS4 | ZSTS3 | ZSTS2 | ZSTS1 | ZSTS0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | VRSTS_ LEVEL | VRSTS_ MUTE | VRSTS[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | ALLZSTS | B'0 | R | All-Channel Zero Cross Mute Status 0: Not all-channel zero cross mute state 1: All-channel zero cross mute state Note: The state is determined only based on the channels for which the zero cross mute function is set. |
| 23 | ZSTS7 | B'0 | R | Zero Cross Mute Status of Channel 7 This bit indicates the zero cross mute status of channel 7. 0: Not mute status 1: Mute status |
| 22 | ZSTS6 | B'0 | R | Zero Cross Mute Status of Channel 6 This bit indicates the zero cross mute status of channel 6. 0: Not mute status 1: Mute status |
| 21 | ZSTS5 | B'0 | R | Zero Cross Mute Status of Channel 5 This bit indicates the zero cross mute status of channel 5. 0: Not mute status 1: Mute status |
| 20 | ZSTS4 | B'0 | R | Zero Cross Mute Status of Channel 4 This bit indicates the zero cross mute status of channel 4. 0: Not mute status 1: Mute status |
| 19 | ZSTS3 | B'0 | R | Zero Cross Mute Status of Channel 3 This bit indicates the zero cross mute status of channel 3. 0: Not mute status 1: Mute status |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|--|
| 18 | ZSTS2 | B'0 | R | Zero Cross Mute Status of Channel 2 This bit indicates the zero cross mute status of channel 2. 0: Not mute status 1: Mute status |
| 17 | ZSTS1 | B'0 | R | Zero Cross Mute Status of Channel 1 This bit indicates the zero cross mute status of channel 1. 0: Not mute status 1: Mute status |
| 16 | ZSTS0 | B'0 | R | Zero Cross Mute Status of Channel 0 This bit indicates the zero cross mute status of channel 0. 0: Not mute status 1: Mute status |
| 15 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | VRSTS_LEVE L | B'0 | R | Volume Ramp Level Status 0: The volume ramp level is not the level specified by the DVCp_VRDBR register. (VRSTS is not 011.) 1: The volume ramp level is the level specified by the DVCp_VRDBR register. (VRSTS is 011.) |
| 3 | VRSTS_MUTE | B'1 | R | Volume Ramp Mute Status 0: Not mute status 1: Mute status |
| 2 to 0 | VRSTS[2:0] | H'0 | R | Volume Ramp Status These bits indicate the volume ramp status. B'000: Mute status B'001: Volume ramp down B'010: Volume ramp up B'011: The volume ramp level is the level specified by the DVCp_VRDBR register. B'100: Volume of input data is maintained (Volume is 1-time) B'101~111: Reserved bits |

43.2.94 DVCp Interrupt Enable Register (DVCp_DVIER)

Note: p = 0, 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: DVCp_DVIER enables or disables output of interrupts corresponding to the states indicated in the DVCp_DVUSR register.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----------------|--------------|--------------|--------------|--------------------|-------------------|--------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | ALLZS TS_ie | ZSTS7 _ie | ZSTS6 _ie | ZSTS5 _ie | ZSTS4 _ie | ZSTS3 _ie | ZSTS2 _ie | ZSTS1 _ie | ZSTS0 _ie |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | VRSTS_L EVEL_ie | VRSTS_M UTE_ie | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | ALLZSTS_ie | B'0 | R/W | allzsts_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 23 | ZSTS7_ie | B'0 | R/W | zsts7_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 22 | ZSTS6_ie | B'0 | R/W | zsts6_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 21 | ZSTS5_ie | B'0 | R/W | zsts5_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 20 | ZSTS4_ie | B'0 | R/W | zsts4_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 19 | ZSTS3_ie | B'0 | R/W | zsts3_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 18 | ZSTS2_ie | B'0 | R/W | zsts2_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 17 | ZSTS1_ie | B'0 | R/W | zsts1_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

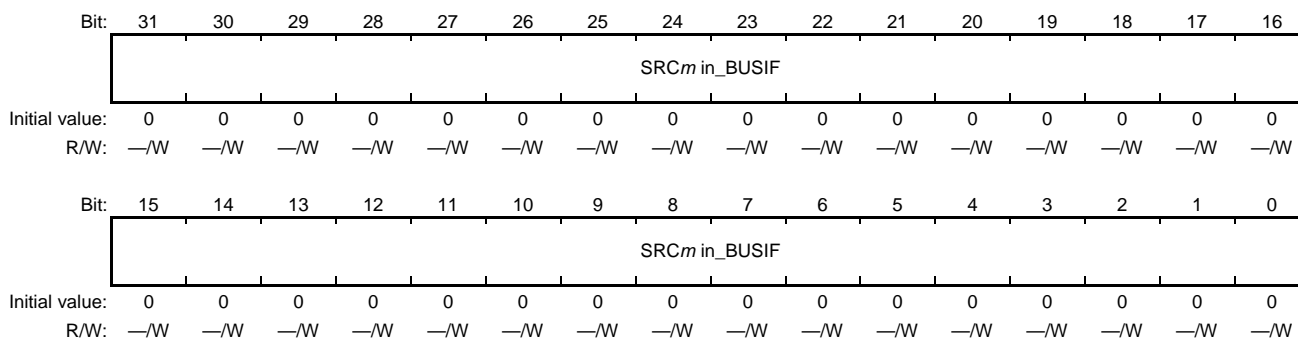
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|---|
| 16 | ZSTS0_ie | B'0 | R/W | zsts0_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 15 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | VRSTS_LEVEL_ie | B'0 | R/W | vrsts_level_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 3 | VRSTS_MUTE_ie | B'0 | R/W | vrsts_mute_int_enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 2 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

43.2.95 SRCm in Write Data Register (SRCm in_BUSIF)

Note: m = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm in_BUSIF are window registers in which data is stored during data transfer via SRCm in_BUSIF. These registers are used for transmission.



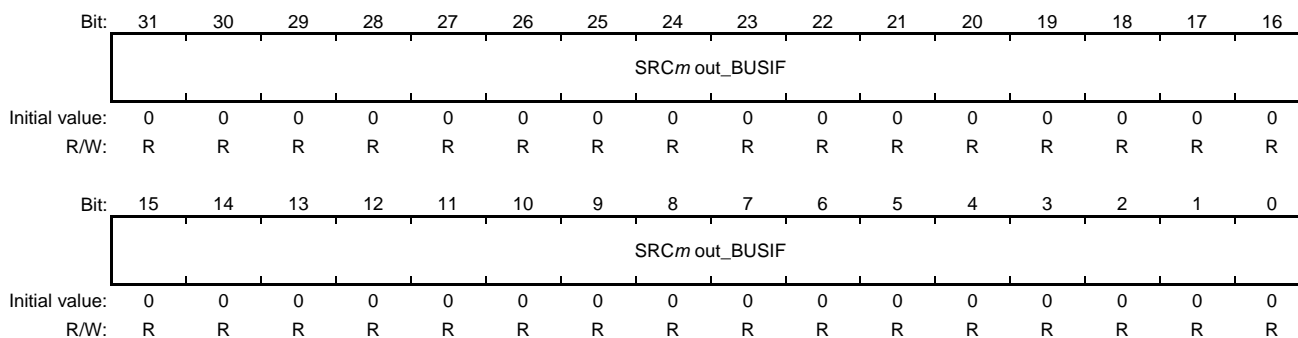
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 0 | SRCm in_BUSIF | All 0 | -/W | These bits are a window register in which data is stored during data transfer via SRCmin_BUSIF. This register is used for transmission. |

43.2.96 SRCm out Read Data Register (SRCm out_BUSIF)

Note: m = 0 to 9

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: SRCm out_BUSIF are window registers in which data is stored during data transfer via SRCm out_BUSIF. These registers are used for reception.



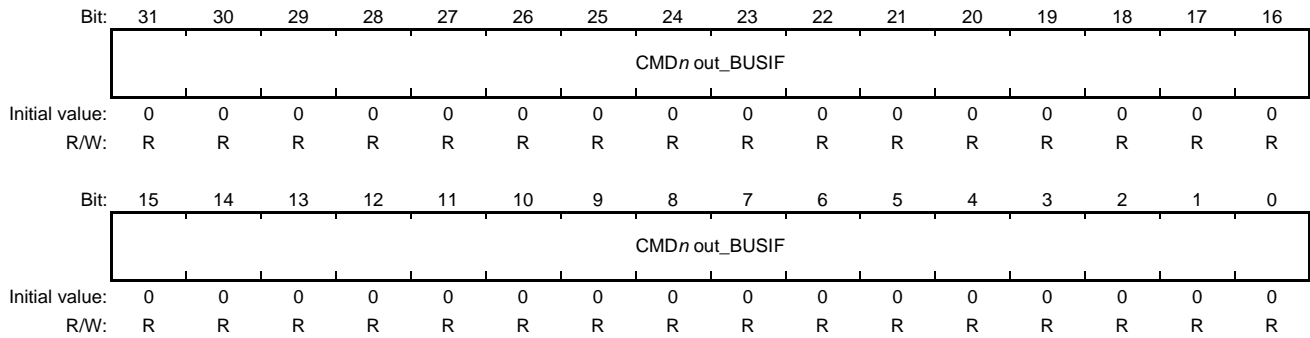
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|---|
| 31 to 0 | SRCm out_BUSIF | All 0 | R | These bits are a window register in which data is stored during data transfer via SRCmout_BUSIF. This register is used for reception. |

43.2.97 CMDn out Read Data Register (CMDn out_BUSIF)

Note: n = 0 or 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: CMDn out_BUSIF are window registers in which data is stored during data transfer via CMDn out_BUSIF. These registers are used for reception.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|---|
| 31 to 0 | CMDn out_BUSIF | All 0 | R | These bits are a window register in which data is stored during data transfer via CMDnout_BUSIF. This register is used for reception. |

43.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

43.3.1 Procedure for Initializing the SCU

Figures Figure 43.2 and Figure 43.3 show the procedure for initializing the SCU. For details on the register settings, refer to section 43.2, Register Description.

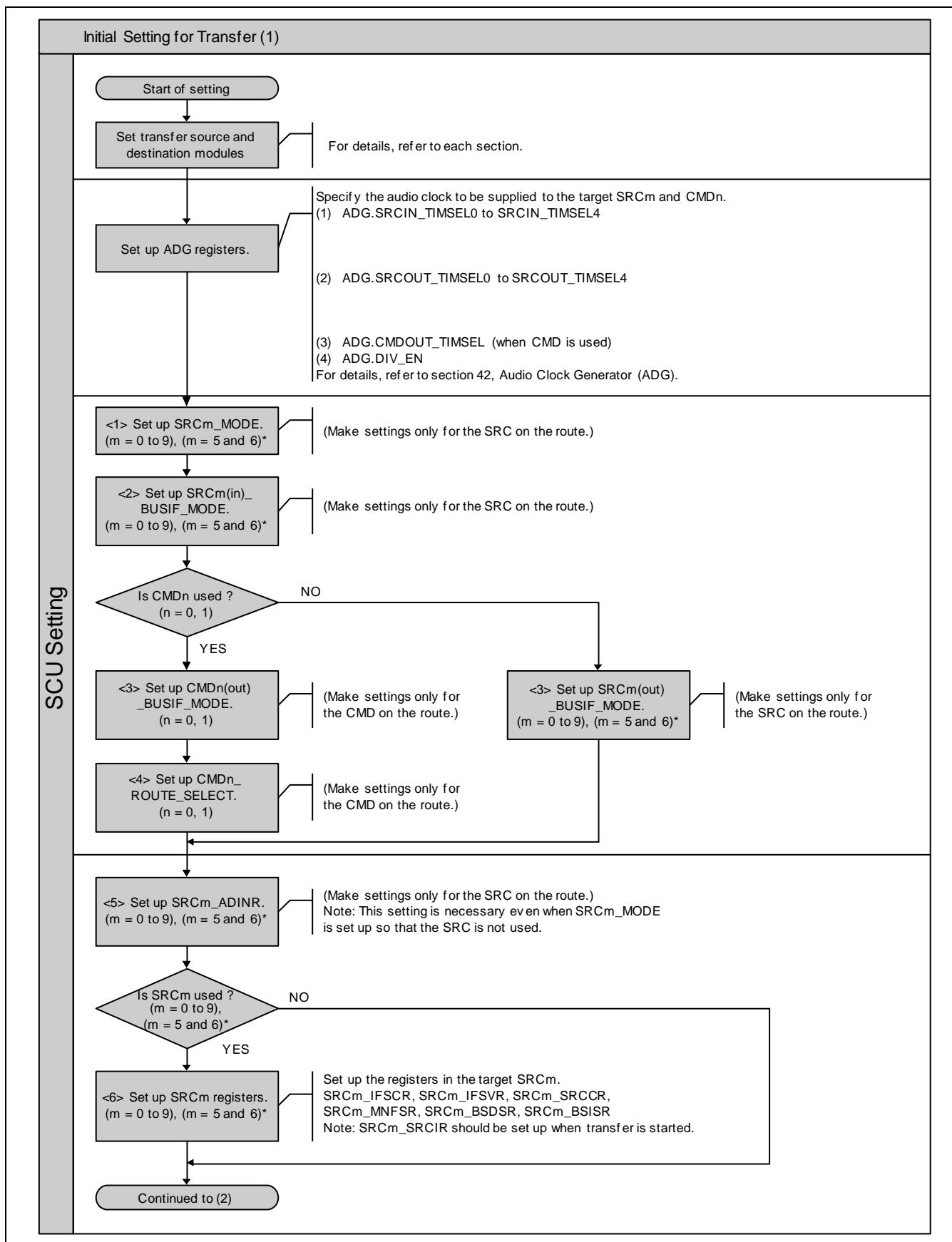


Figure 43.2 Procedure for Initializing the SCU (1)

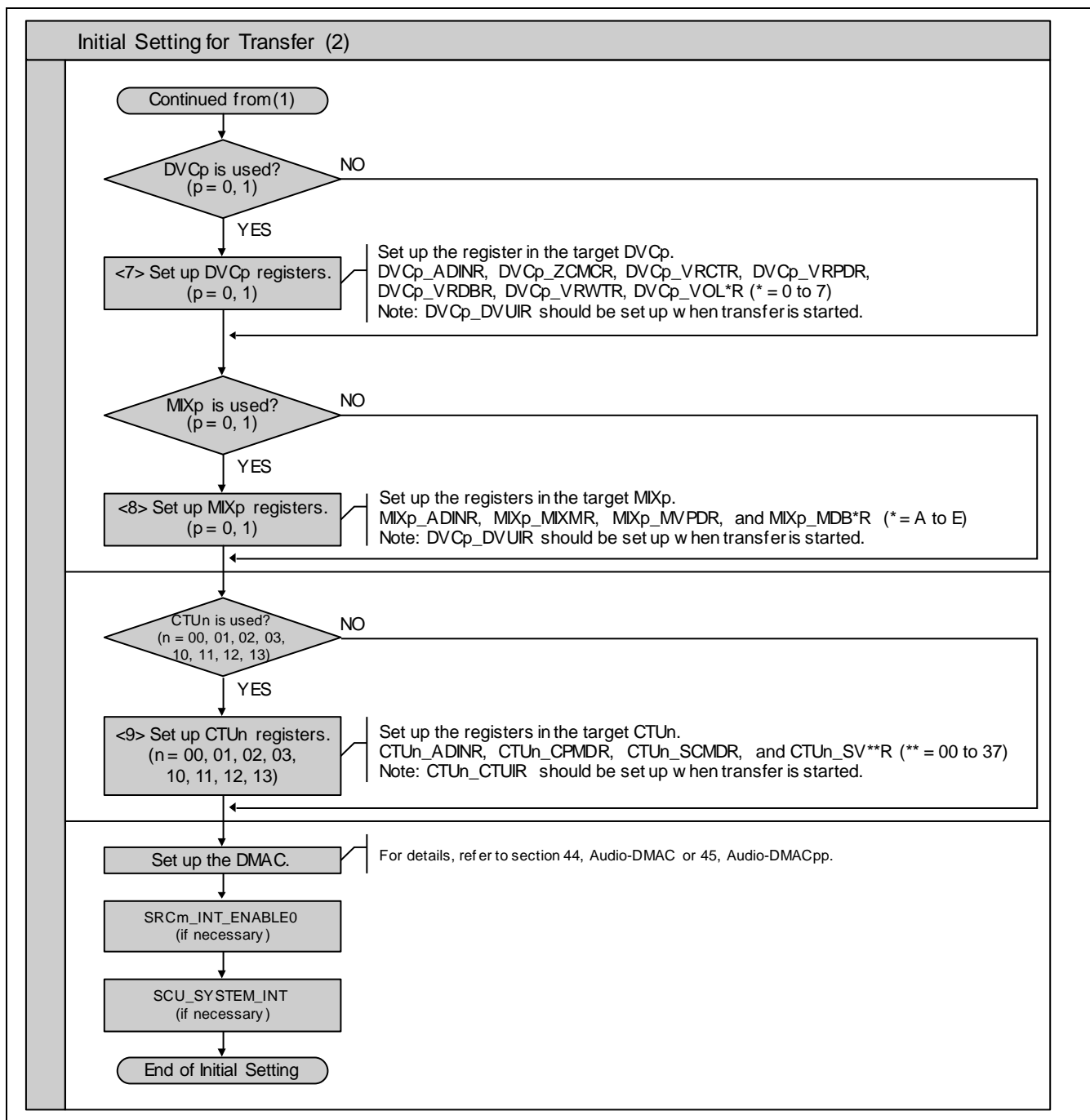


Figure 43.3 Procedure for Initializing the SCU (2)

43.3.2 Procedure for Starting and Stopping Transfer by the SCU

Figure 43.4 shows the procedure for starting and stopping the transfer by the SCU. For details on the register settings, refer to section 43.2, Register Description.

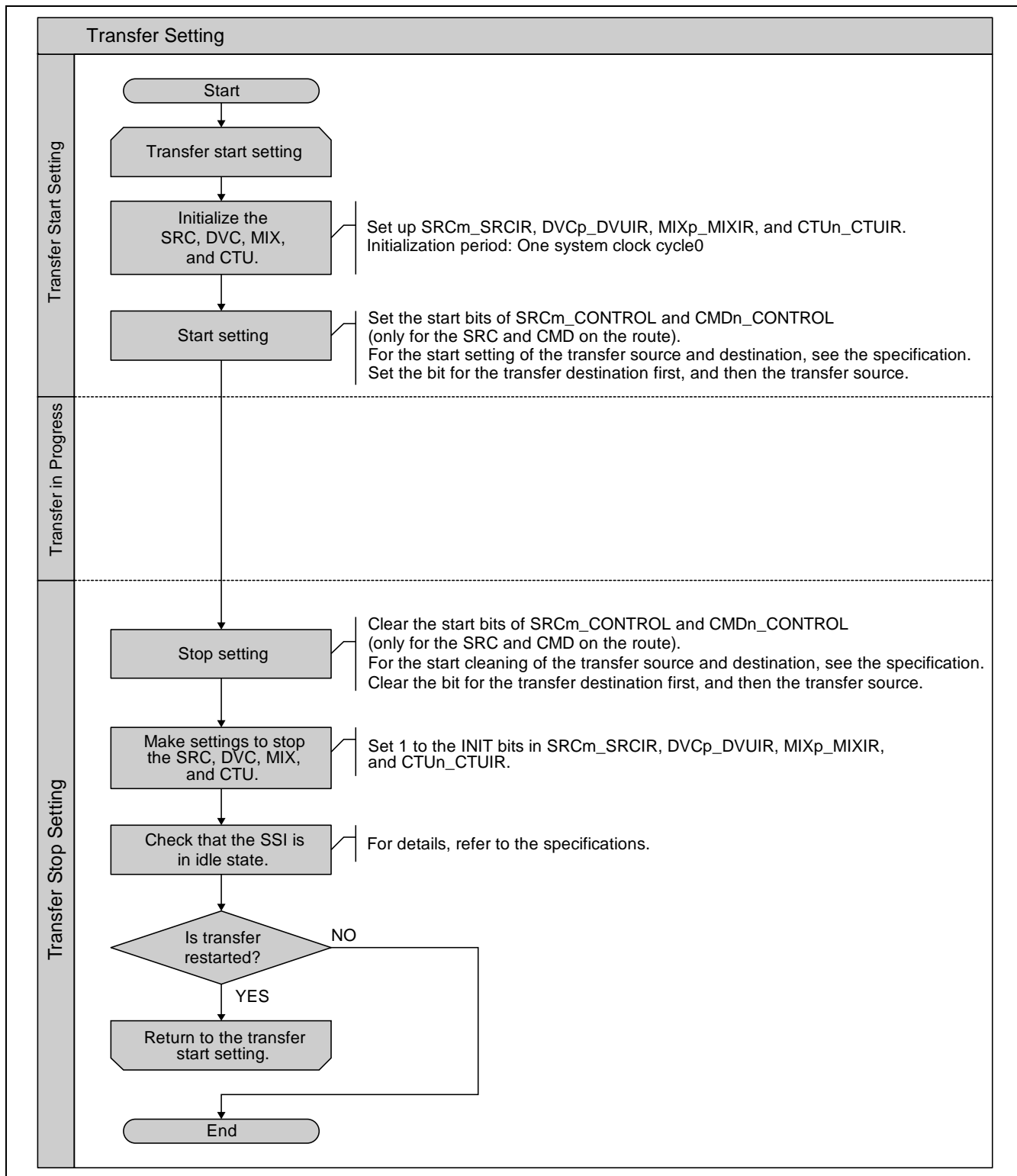


Figure 43.4 Procedure for Starting and Stopping Transfer by the SCU

43.3.3 Data Format for Data Transfer BUSIF

Table 43.3 shows the data formats handled in the SCU. When writing or reading data through the data transfer BUSIF (SRCm in_BUSIF, SRCm out_BUSIF, or CMDn out_BUSIF), align data to match the appropriate data format from those in table Table 43.3. Data alignment can be changed through the SRCm(in/out)_BUSIF_MODE, SRCm_MODE register or CMDnout_BUSIF_MODE register (see section 43.2.4, SRCm_MODE Register (SRCm_MODE) or section 43.2.10, CMDn out_BUSIF_MODE Register (CMDn out_BUSIF_MODE)).

Table 43.3 Data Formats Handled in the SCU

*1. LSB is changeable, depends on stereo data width.

| | |
|---|---|
| 24-bit stereo data, multichannel data | <p>31 8 7 0</p> <p>MSB LSB x x x x</p> |
| 17 to 23-bit stereo data, multichannel data | <p>31 15 to 9 8 7 0</p> <p>MSB LSB x x x x x x</p> <p>*1</p> |
| 16-bit stereo data, multichannel data | <p>31 16 15 0</p> <p>MSB LSB MSB LSB</p> <p>(Lch) (Rch)</p> |
| 9- to 15-bit stereo data, multichannel data | <p>31 23 to 17 16 15 7 to 1 0</p> <p>MSB LSB x x MSB LSB x x</p> <p>(Lch) (Rch)</p> <p>*1 *1</p> |
| 8-bit stereo data | <p>31 24 23 16 15 8 7 0</p> <p>MSB MSB MSB MSB MSB MSB MSB MSB</p> <p>(Rch/2) (Lch/2) (Rch/1) (Lch/1)</p> |
| 16-bit monaural data | <p>31 16 15 0</p> <p>MSB MSB x x x x x x x x</p> |
| 8-bit monaural data | <p>31 24 23 16 15 0</p> <p>MSB MSB x x x x x x x x x x x x x x</p> |

- Notes:
1. Write 0 to the "x" bits in the table when writing data through the data transfer SRCm in_BUSIF. When reading data through the SRCm out_BUSIF or CMDn out_BUSIF, ignore the values read from these bits.
 2. In the 8-bit stereo data format, (Lch/1) and (Rch/1) indicate the data pair to be processed first and (Lch/2) and (Rch/2) indicate the next data pair to be processed.
 3. Only the MSB-first data formats can be used in the SRU.

43.3.4 Changing Data Order in Channel Units

The data order can be changed in channel units in the SCU immediately before input to the SRC and immediately after output from the CMD. Table 43.4 and Table 43.5 show the data places in each data format. Use SRCm_BUSIF_DALIGN or SRCn_BUSIF_DALIGN to change the order before input to the SRC (see section 43.2.2, SRCm_BUSIF_DALIGN Register (SRCm_BUSIF_DALIGN) or section 43.2.3, SRCn_BUSIF_DALIGN Register (SRCn_BUSIF_DALIGN)). Use CMDn_BUSIF_DALIGN to change the order immediately after output from the CMD (see section 43.2.11, CMDn_BUSIF_DALIGN Register (CMDn_BUSIF_DALIGN)). Table 43.4 and Table 43.5 show the data places when these registers are set to the initial values.

Table 43.4 Data Places in Each Data Format (1)

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---------|---------|---------|---|------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|-----|------|---------|---------|-----|------|---------|-----|---|------|---------|--|---|------|---------|--|---|------|---------|--|---|-----|-----|--|---|
| <p>BUSIF</p> <ul style="list-style-type: none"> · Stereo (2 channels) · 24 bits | <p>External memory image</p> <table style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td></td> <td style="text-align: center;">0</td> </tr> <tr> <td>H'00</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'04</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'08</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'0C</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> </table> | | 31 | | 0 | H'00 | Place 0 | | * | H'04 | Place 1 | | * | H'08 | Place 0 | | * | H'0C | Place 1 | | * | ... | ... | | * | | | | | | | | | | | | | | | | |
| | 31 | | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 1 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 0 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 1 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF</p> <ul style="list-style-type: none"> · Stereo (2 channels) · 16 bits | <p>External memory image</p> <table style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td></td> <td style="text-align: center;">0</td> </tr> <tr> <td>H'00</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td></td> </tr> <tr> <td>H'04</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td></td> </tr> <tr> <td>H'08</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td></td> </tr> <tr> <td>H'0C</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td></td> </tr> <tr> <td>...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td></td> </tr> </table> | | 31 | | 0 | H'00 | Place 0 | Place 1 | | H'04 | Place 0 | Place 1 | | H'08 | Place 0 | Place 1 | | H'0C | Place 0 | Place 1 | | ... | ... | ... | | | | | | | | | | | | | | | | | |
| | 31 | | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF</p> <ul style="list-style-type: none"> · Stereo (2 channels) · 8 bits | <p>External memory image</p> <table style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td></td> <td></td> <td style="text-align: center;">0</td> </tr> <tr> <td>H'00</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> </tr> <tr> <td>H'04</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> </tr> <tr> <td>...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;">...</td> </tr> </table> | | 31 | | | 0 | H'00 | Place 1 | Place 0 | Place 1 | Place 0 | H'04 | Place 1 | Place 0 | Place 1 | Place 0 | ... | ... | ... | ... | ... | | | | | | | | | | | | | | | | | | | | |
| | 31 | | | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 1 | Place 0 | Place 1 | Place 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 1 | Place 0 | Place 1 | Place 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | ... | ... | ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF</p> <ul style="list-style-type: none"> · Monaural (1 channel) · 8/16 bits | <p>External memory image</p> <table style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td></td> <td style="text-align: center;">0</td> </tr> <tr> <td>H'00</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'04</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'08</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'0C</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> </table> | | 31 | | 0 | H'00 | Place 0 | | * | H'04 | Place 0 | | * | H'08 | Place 0 | | * | H'0C | Place 0 | | * | ... | ... | | * | | | | | | | | | | | | | | | | |
| | 31 | | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 0 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 0 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 0 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF</p> <ul style="list-style-type: none"> · Multichannel (4 channels) · 24 bits | <p>External memory image</p> <table style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td></td> <td style="text-align: center;">0</td> </tr> <tr> <td>H'00</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'04</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'08</td> <td style="border: 1px solid black; padding: 2px;">Place 2</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'0C</td> <td style="border: 1px solid black; padding: 2px;">Place 3</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'10</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'14</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'18</td> <td style="border: 1px solid black; padding: 2px;">Place 2</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>H'1C</td> <td style="border: 1px solid black; padding: 2px;">Place 3</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> <tr> <td>...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;"></td> <td style="border: 1px solid black; padding: 2px;">*</td> </tr> </table> | | 31 | | 0 | H'00 | Place 0 | | * | H'04 | Place 1 | | * | H'08 | Place 2 | | * | H'0C | Place 3 | | * | H'10 | Place 0 | | * | H'14 | Place 1 | | * | H'18 | Place 2 | | * | H'1C | Place 3 | | * | ... | ... | | * |
| | 31 | | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 1 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 2 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 3 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'10 | Place 0 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'14 | Place 1 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'18 | Place 2 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'1C | Place 3 | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF</p> <ul style="list-style-type: none"> · Multichannel (4 channels) · 16 bits | <p>External memory image</p> <table style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td></td> <td style="text-align: center;">0</td> </tr> <tr> <td>H'00</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td></td> </tr> <tr> <td>H'04</td> <td style="border: 1px solid black; padding: 2px;">Place 2</td> <td style="border: 1px solid black; padding: 2px;">Place 3</td> <td></td> </tr> <tr> <td>H'08</td> <td style="border: 1px solid black; padding: 2px;">Place 0</td> <td style="border: 1px solid black; padding: 2px;">Place 1</td> <td></td> </tr> <tr> <td>H'0C</td> <td style="border: 1px solid black; padding: 2px;">Place 2</td> <td style="border: 1px solid black; padding: 2px;">Place 3</td> <td></td> </tr> <tr> <td>...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td style="border: 1px solid black; padding: 2px;">...</td> <td></td> </tr> </table> | | 31 | | 0 | H'00 | Place 0 | Place 1 | | H'04 | Place 2 | Place 3 | | H'08 | Place 0 | Place 1 | | H'0C | Place 2 | Place 3 | | ... | ... | ... | | | | | | | | | | | | | | | | | |
| | 31 | | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 2 | Place 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 2 | Place 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Table 43.5 Data Places in Each Data Format (2)

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|--|---------|----|---|------|---------|---------|------|---------|---------|------|---------|---------|------|---------|---------|------|---------|---------|------|---------|-----|------|---------|---|------|---------|---|------|---------|-----|-----|-----|---|
| <p>BUSIF</p> <ul style="list-style-type: none"> · Multichannel (6 channels) · 24 bits | <p>External memory image</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td style="text-align: center;">0</td> </tr> <tr> <td>H'00</td> <td style="text-align: center;">Place 0</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'04</td> <td style="text-align: center;">Place 1</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'08</td> <td style="text-align: center;">Place 2</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'0C</td> <td style="text-align: center;">Place 3</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'10</td> <td style="text-align: center;">Place 4</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'14</td> <td style="text-align: center;">Place 5</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'18</td> <td style="text-align: center;">Place 0</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'1C</td> <td style="text-align: center;">Place 1</td> <td style="text-align: center;">*</td> </tr> <tr> <td>...</td> <td style="text-align: center;">...</td> <td style="text-align: center;">...</td> </tr> </table> | | 31 | 0 | H'00 | Place 0 | * | H'04 | Place 1 | * | H'08 | Place 2 | * | H'0C | Place 3 | * | H'10 | Place 4 | * | H'14 | Place 5 | * | H'18 | Place 0 | * | H'1C | Place 1 | * | ... | ... | ... | | | |
| | 31 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 1 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 2 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 3 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'10 | Place 4 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'14 | Place 5 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'18 | Place 0 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'1C | Place 1 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF</p> <ul style="list-style-type: none"> · Multichannel (6 channels) · 16 bits | <p>External memory image</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td style="text-align: center;">0</td> </tr> <tr> <td>H'00</td> <td style="text-align: center;">Place 0</td> <td style="text-align: center;">Place 1</td> </tr> <tr> <td>H'04</td> <td style="text-align: center;">Place 2</td> <td style="text-align: center;">Place 3</td> </tr> <tr> <td>H'08</td> <td style="text-align: center;">Place 4</td> <td style="text-align: center;">Place 5</td> </tr> <tr> <td>H'0C</td> <td style="text-align: center;">Place 0</td> <td style="text-align: center;">Place 1</td> </tr> <tr> <td>...</td> <td style="text-align: center;">...</td> <td style="text-align: center;">...</td> </tr> </table> | | 31 | 0 | H'00 | Place 0 | Place 1 | H'04 | Place 2 | Place 3 | H'08 | Place 4 | Place 5 | H'0C | Place 0 | Place 1 | ... | ... | ... | | | | | | | | | | | | | | | |
| | 31 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 2 | Place 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 4 | Place 5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF</p> <ul style="list-style-type: none"> · Multichannel (8 channels) · 24 bits | <p>External memory image</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td style="text-align: center;">0</td> </tr> <tr> <td>H'00</td> <td style="text-align: center;">Place 0</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'04</td> <td style="text-align: center;">Place 1</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'08</td> <td style="text-align: center;">Place 2</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'0C</td> <td style="text-align: center;">Place 3</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'10</td> <td style="text-align: center;">Place 4</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'14</td> <td style="text-align: center;">Place 5</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'18</td> <td style="text-align: center;">Place 6</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'1C</td> <td style="text-align: center;">Place 7</td> <td style="text-align: center;">*</td> </tr> <tr> <td>H'20</td> <td style="text-align: center;">Place 0</td> <td style="text-align: center;">*</td> </tr> <tr> <td>...</td> <td style="text-align: center;">...</td> <td style="text-align: center;">*</td> </tr> </table> | | 31 | 0 | H'00 | Place 0 | * | H'04 | Place 1 | * | H'08 | Place 2 | * | H'0C | Place 3 | * | H'10 | Place 4 | * | H'14 | Place 5 | * | H'18 | Place 6 | * | H'1C | Place 7 | * | H'20 | Place 0 | * | ... | ... | * |
| | 31 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 1 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 2 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 3 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'10 | Place 4 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'14 | Place 5 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'18 | Place 6 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'1C | Place 7 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'20 | Place 0 | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | * | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <p>BUSIF</p> <ul style="list-style-type: none"> · Multichannel (8 channels) · 16 bits | <p>External memory image</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td style="text-align: center;">0</td> </tr> <tr> <td>H'00</td> <td style="text-align: center;">Place 0</td> <td style="text-align: center;">Place 1</td> </tr> <tr> <td>H'04</td> <td style="text-align: center;">Place 2</td> <td style="text-align: center;">Place 3</td> </tr> <tr> <td>H'08</td> <td style="text-align: center;">Place 4</td> <td style="text-align: center;">Place 5</td> </tr> <tr> <td>H'0C</td> <td style="text-align: center;">Place 6</td> <td style="text-align: center;">Place 7</td> </tr> <tr> <td>H'10</td> <td style="text-align: center;">Place 0</td> <td style="text-align: center;">Place 1</td> </tr> <tr> <td>...</td> <td style="text-align: center;">...</td> <td style="text-align: center;">...</td> </tr> </table> | | 31 | 0 | H'00 | Place 0 | Place 1 | H'04 | Place 2 | Place 3 | H'08 | Place 4 | Place 5 | H'0C | Place 6 | Place 7 | H'10 | Place 0 | Place 1 | ... | ... | ... | | | | | | | | | | | | |
| | 31 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'00 | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'04 | Place 2 | Place 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'08 | Place 4 | Place 5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0C | Place 6 | Place 7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'10 | Place 0 | Place 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | ... | ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

43.3.5 SRC Block

The SRC is a block for implementing the sampling rate conversion function, which can convert asynchronous sampling rates.

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

- Asynchronous sampling rate conversion is available
- Supports resolutions up to 24 bits
- High-sound-quality type (THD + N* is -132 dB) and general-sound-quality type (THD + N* is -96 dB)
- Automatically generates antialiasing filter coefficients
- Four modules support one, two, four, six, or eight channels, and six modules support one or two channels.

Note: * Total harmonic distortion plus noise

[RZ/G2E]

- Asynchronous sampling rate conversion is available
- Supports resolutions up to 24 bits
- High-sound-quality type (THD + N* is -132 dB) and general-sound-quality type (THD + N* is -96 dB)
- Automatically generates antialiasing filter coefficients
- Two modules support one, two, four, six, or eight channels, and eight modules support one or two channels.

Note: * Total harmonic distortion plus noise

Figure 43.5 shows the SRC block diagram.

The "Input audio data" in the figure indicates the data before SRC processing is applied. The "Output audio data" indicates the data after SRC processing is applied.

The "Input data timing" is a signal to show the sampling period for the input audio data. The "Output data timing" is a signal to show the sampling period for the output audio data. When CMD is used in the route, the output data timing signal is the same signal as the output data timing signal used in the CMD.

In synchronous mode specified by setting the SRCMD bit in the SRCm_SRCCR register, when the sync_in bit in the SRCm_MODE register is set to 1, the SRCm doesn't need the input data timing signal, and when the sync_out bit in the SRCm_MODE register is set to 1, the SRCm doesn't need the output data timing signal.

Before using the SRC function, be sure to specify the sampling rate through the SRCm IFS value setting register (see section 43.2.22, SRCm IFS Value Setting Register (SRCm_IFSVR)).

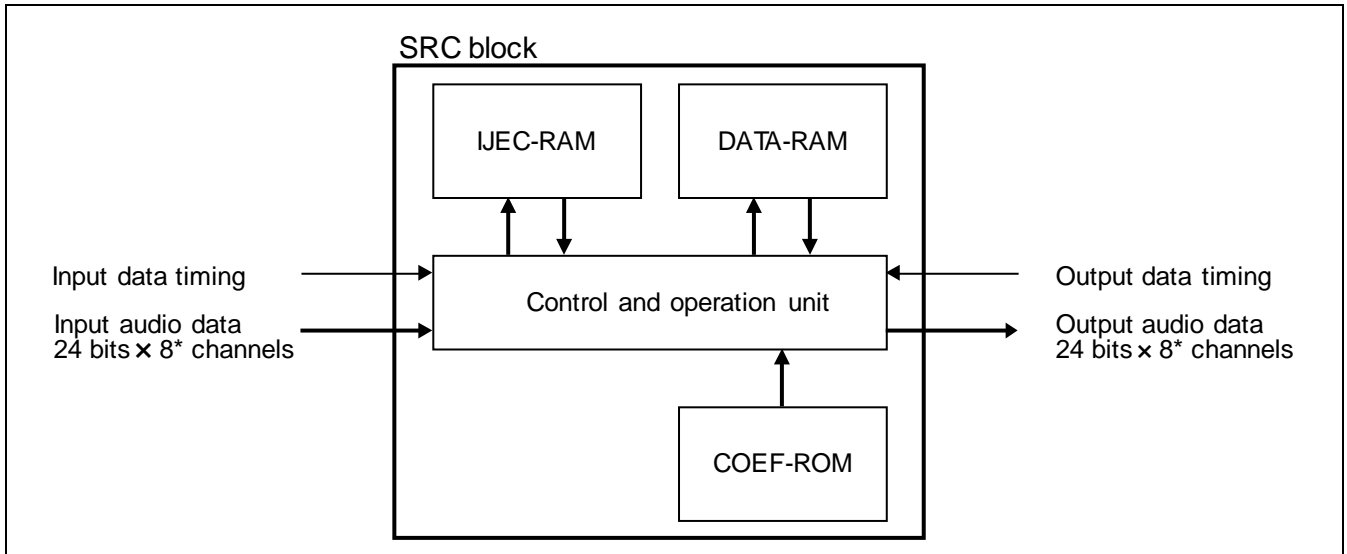


Figure 43.5 SRC Block Diagram

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Table 43.6 shows the SRC functions.

Table 43.6 SRC Functions

| Item | | Performance | | | |
|--|--------------------|-----------------------------------|------------------|------------------|------------------|
| Type of SRC | | Asynchronous/ Synchronous SRC | | | |
| Channel number | | 1 or 2 | 4 | 6 | 8 |
| Sampling rate of input source | SRC0 | 8k to 192k [Hz] | 8k to 192k [Hz] | 8k to 192k [Hz] | 8k to 192k [Hz] |
| | SRC1, SRC3, SRC4 | 8k to 192k [Hz] | 8k to 192k [Hz] | 8k to 128k [Hz] | 8k to 96k [Hz] |
| | SRC2, SRC9 | 8k to 192k [Hz] | — | — | — |
| | SRC5 to SRC8 | 8k to 192k [Hz] | — | — | — |
| Sampling rate of output source | SRC0 | 8k to 192k [Hz] | 8k to 192k [Hz] | 8k to 192k [Hz] | 8k to 192k [Hz] |
| | SRC1, SRC3, SRC4 | 8k to 192k [Hz] | 8k to 192k [Hz] | 8k to 128k [Hz] | 8k to 96k [Hz] |
| | SRC2, SRC9 | 8k to 192k [Hz] | — | — | — |
| | SRC5 to SRC8 | 8k to 192k [Hz] | — | — | — |
| Ratio of input and output sampling rates (FSO/FSI ratio) | SRC0 | 6 to 1/6 [times] | 6 to 1/4 [times] | 6 to 1/4 [times] | 6 to 1/4 [times] |
| | SRC1, SRC3, SRC4 | 6 to 1/6 [times] | 6 to 1/4 [times] | 6 to 1/2 [times] | 6 to 1/2 [times] |
| | SRC2, SRC9 | 6 to 1/6 [times] | — | — | — |
| | SRC5 to SRC8 | 6 to 1/6 [times] | — | — | — |
| Sound quality (THD+N) | SRC0 to SRC4, SRC9 | -132 dB (for high-quality type) | | | |
| | SRC5 to SRC8 | -96 dB (for general-quality type) | | | |

[RZ/G2E] Table 43.7 shows the SRC functions.

Table 43.7 SRC Functions

| Item | | Performance | | | |
|--|------------------------|-----------------------------------|------------------|------------------|------------------|
| Type of SRC | | Asynchronous/ Synchronous SRC | | | |
| Channel number | | 1 or 2 | 4 | 6 | 8 |
| Sampling rate of input source | SRC1, SRC3 | 8k to 192k [Hz] | 8k to 192k [Hz] | 8k to 128k [Hz] | 8k to 96k [Hz] |
| | SRC0, SRC2, SRC4, SRC9 | 8k to 192k [Hz] | — | — | — |
| | SRC5 to SRC8 | 8k to 192k [Hz] | — | — | — |
| Sampling rate of output source | SRC1, SRC3 | 8k to 192k [Hz] | 8k to 192k [Hz] | 8k to 128k [Hz] | 8k to 96k [Hz] |
| | SRC0, SRC2, SRC4, SRC9 | 8k to 192k [Hz] | — | — | — |
| | SRC5 to SRC8 | 8k to 192k [Hz] | — | — | — |
| Ratio of input and output sampling rates (FSO/FSI ratio) | SRC1, SRC3 | 6 to 1/6 [times] | 6 to 1/4 [times] | 6 to 1/2 [times] | 6 to 1/2 [times] |
| | SRC0, SRC2, SRC4, SRC9 | 6 to 1/6 [times] | — | — | — |
| | SRC5 to SRC8 | 6 to 1/6 [times] | — | — | — |
| Sound quality (THD+N) | SRC0 to SRC4, SRC9 | -132 dB (for high-quality type) | | | |
| | SRC5 to SRC8 | -96 dB (for general-quality type) | | | |

Table 43.8 and 43.9 show the IJEC RAM and DATA RAM buffer size settings and the latencies.

Latency is different by setting of channel number.

Select the combination of settings BUFDATA[10:0] bits in SRCm_BSDSR/ SRCn_BSDSR register and IJECSIZE[8:0] bits in SRCm_BSISR register from Table 43.8 and 43.9. Operation cannot be guaranteed if the register is not set as specified combination.

Output delay depends on FSO/FSI ratio and it is calculated from following formula.

Output delay [Output sample] = (Processing delay) × (FSO/FSI ratio) + (Logic delay*)

*Logic delay = 3[Output sample]

Table 43.8 Combination of Register Setting Related to FSO/FSI Ratio and Channel, Latency
[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| Item | Setting Range | | Register Setting | | | Latency (Example case calculated by formula) | | | | | | | | | | Note |
|--------------------|---------------|----------------|------------------|-----------------|---------------------------|---|------|------|-----|-----|-----|-----|-----|-----|-----|------|
| | FSO/FSI Ratio | Channel Number | IJEC SIZE [8:0] | BUF DATA [10:0] | Processing Delay [sample] | | | | | | | | | | | |
| | | | | | | 6 | 4 | 3 | 2 | 1 | 2/3 | 1/2 | 1/3 | 1/4 | 1/6 | |
| SRC0 | 6 to 1/6 | 1 to 2 | H'60 | H'180 | 241 | 1449 | 967 | 726 | 485 | 244 | 164 | 124 | 83 | 63 | 43 | * |
| | 6 to 1/4 | 1 to 8 | H'40 | H'100 | 161 | 969 | 647 | 486 | 325 | 164 | 110 | 84 | 57 | 43 | | |
| | 6 to 1/3 | 1 to 8 | H'30 | H'0C0 | 121 | 729 | 487 | 366 | 245 | 124 | 84 | 64 | 43 | | | |
| | 6 to 1/2 | 1 to 8 | H'20 | H'080 | 81 | 489 | 327 | 246 | 165 | 84 | 57 | 44 | | | | |
| | 6 to 2/3 | 1 to 8 | H'20 | H'060 | 65 | 393 | 263 | 198 | 133 | 68 | 46 | | | | | |
| | 6 to 1 | 1 to 8 | H'20 | H'040 | 49 | 297 | 199 | 150 | 101 | 52 | | | | | | |
| SRC1 | 6 to 1/6 | 1 to 2 | H'60 | H'180 | 241 | 1449 | 967 | 726 | 485 | 244 | 164 | 124 | 83 | 63 | 43 | * |
| SRC3 | 6 to 1/4 | 1 to 4 | H'40 | H'100 | 161 | 969 | 647 | 486 | 325 | 164 | 110 | 84 | 57 | 43 | | |
| SRC4 | 6 to 1/3 | 1 to 4 | H'30 | H'0C0 | 121 | 729 | 487 | 366 | 245 | 124 | 84 | 64 | 43 | | | |
| | 6 to 1/2 | 1 to 8 | H'20 | H'080 | 81 | 489 | 327 | 246 | 165 | 84 | 57 | 44 | | | | |
| | 6 to 2/3 | 1 to 8 | H'20 | H'060 | 65 | 393 | 263 | 198 | 133 | 68 | 46 | | | | | |
| | 6 to 1 | 1 to 8 | H'20 | H'040 | 49 | 297 | 199 | 150 | 101 | 52 | | | | | | |
| SRC2 | 6 to 1/6 | 1 to 2 | H'60 | H'180 | 241 | 1449 | 967 | 726 | 485 | 244 | 164 | 124 | 83 | 63 | 43 | * |
| SRC9 | 6 to 1/4 | 1 to 2 | H'40 | H'100 | 161 | 969 | 647 | 486 | 325 | 164 | 110 | 84 | 57 | 43 | | |
| | 6 to 1/3 | 1 to 2 | H'30 | H'0C0 | 121 | 729 | 487 | 366 | 245 | 124 | 84 | 64 | 43 | | | |
| | 6 to 1/2 | 1 to 2 | H'20 | H'080 | 81 | 489 | 327 | 246 | 165 | 84 | 57 | 44 | | | | |
| | 6 to 2/3 | 1 to 2 | H'20 | H'060 | 65 | 393 | 263 | 198 | 133 | 68 | 46 | | | | | |
| | 6 to 1 | 1 to 2 | H'20 | H'040 | 49 | 297 | 199 | 150 | 101 | 52 | | | | | | |
| SRC5 to SRC8 | 6 to 1/6 | 1 to 2 | H'60 | H'240 | 337 | 2025 | 1351 | 1014 | 677 | 340 | 228 | 172 | 115 | 87 | 59 | * |
| | 6 to 1/4 | 1 to 2 | H'40 | H'180 | 225 | 1353 | 903 | 678 | 453 | 228 | 153 | 116 | 78 | 59 | | |
| | 6 to 1/3 | 1 to 2 | H'30 | H'120 | 169 | 1017 | 679 | 510 | 341 | 172 | 116 | 88 | 59 | | | |
| | 6 to 1/2 | 1 to 2 | H'20 | H'0C0 | 113 | 681 | 455 | 342 | 229 | 116 | 78 | 60 | | | | |
| | 6 to 2/3 | 1 to 2 | H'20 | H'090 | 89 | 537 | 359 | 270 | 181 | 92 | 62 | | | | | |
| | 6 to 1 | 1 to 2 | H'20 | H'060 | 65 | 393 | 263 | 198 | 133 | 68 | | | | | | |

Note: Set up "*" case by common setting, if there are no problem for the latency.

Table 43.9 Combination of Register Setting Related to FSO/FSI Ratio and Channel, Latency [RZ/G2E]

| Item | Setting Range | | Register Setting | | | Latency (Example case calculated by formula) | | | | | | | | | | |
|--------------------|---------------|----------------|------------------|-------------|---------------------------|---|------|------|-----|-----|-----|-----|-----|-----|-----|------|
| | FSO/FSI Ratio | Channel Number | IJEC | BUF | Processing Delay [sample] | | | | | | | | | | | |
| | | | SIZE [8:0] | DATA [10:0] | | 6 | 4 | 3 | 2 | 1 | 2/3 | 1/2 | 1/3 | 1/4 | 1/6 | Note |
| SRC1 | 6 to 1/6 | 1 to 2 | H'60 | H'180 | 241 | 1449 | 967 | 726 | 485 | 244 | 164 | 124 | 83 | 63 | 43 | * |
| SRC3 | 6 to 1/4 | 1 to 4 | H'40 | H'100 | 161 | 969 | 647 | 486 | 325 | 164 | 110 | 84 | 57 | 43 | | |
| | 6 to 1/3 | 1 to 4 | H'30 | H'0C0 | 121 | 729 | 487 | 366 | 245 | 124 | 84 | 64 | 43 | | | |
| | 6 to 1/2 | 1 to 8 | H'20 | H'080 | 81 | 489 | 327 | 246 | 165 | 84 | 57 | 44 | | | | |
| | 6 to 2/3 | 1 to 8 | H'20 | H'060 | 65 | 393 | 263 | 198 | 133 | 68 | 46 | | | | | |
| | 6 to 1 | 1 to 8 | H'20 | H'040 | 49 | 297 | 199 | 150 | 101 | 52 | | | | | | |
| SRC0 | 6 to 1/6 | 1 to 2 | H'60 | H'180 | 241 | 1449 | 967 | 726 | 485 | 244 | 164 | 124 | 83 | 63 | 43 | * |
| SRC2 | 6 to 1/4 | 1 to 2 | H'40 | H'100 | 161 | 969 | 647 | 486 | 325 | 164 | 110 | 84 | 57 | 43 | | |
| SRC4 | 6 to 1/3 | 1 to 2 | H'30 | H'0C0 | 121 | 729 | 487 | 366 | 245 | 124 | 84 | 64 | 43 | | | |
| SRC9 | 6 to 1/2 | 1 to 2 | H'20 | H'080 | 81 | 489 | 327 | 246 | 165 | 84 | 57 | 44 | | | | |
| | 6 to 2/3 | 1 to 2 | H'20 | H'060 | 65 | 393 | 263 | 198 | 133 | 68 | 46 | | | | | |
| | 6 to 1 | 1 to 2 | H'20 | H'040 | 49 | 297 | 199 | 150 | 101 | 52 | | | | | | |
| SRC5 to SRC8 | 6 to 1/6 | 1 to 2 | H'60 | H'240 | 337 | 2025 | 1351 | 1014 | 677 | 340 | 228 | 172 | 115 | 87 | 59 | * |
| | 6 to 1/4 | 1 to 2 | H'40 | H'180 | 225 | 1353 | 903 | 678 | 453 | 228 | 153 | 116 | 78 | 59 | | |
| | 6 to 1/3 | 1 to 2 | H'30 | H'120 | 169 | 1017 | 679 | 510 | 341 | 172 | 116 | 88 | 59 | | | |
| | 6 to 1/2 | 1 to 2 | H'20 | H'0C0 | 113 | 681 | 455 | 342 | 229 | 116 | 78 | 60 | | | | |
| | 6 to 2/3 | 1 to 2 | H'20 | H'090 | 89 | 537 | 359 | 270 | 181 | 92 | 62 | | | | | |
| | 6 to 1 | 1 to 2 | H'20 | H'060 | 65 | 393 | 263 | 198 | 133 | 68 | | | | | | |

Note: Set up "*" case by common setting, if there are no problem for the latency.

Table 43.10 shows the INTIFS setting examples in the SRC block.

Table 43.10 INTIFS Values in SRCm_IFSVR Register for Some Specific Cases

| INTIFS[27:0] | | | | | | | |
|---------------------------|----------------------------|------------|------------|------------|------------|------------|------------|
| Input sampling rate [kHz] | Output sampling rate [kHz] | | | | | | |
| | 8 | 16 | 32 | 44.1 | 48 | 96 | 192 |
| 8 | H'040_0000 | H'020_0000 | H'010_0000 | H'00B_9C27 | H'00A_AAAA | — | — |
| 11.025 | H'058_3333 | H'02C_1999 | H'016_0CCC | H'010_0000 | H'00E_B333 | — | — |
| 12 | H'060_0000 | H'030_0000 | H'018_0000 | H'011_6A3B | H'010_0000 | — | — |
| 16 | H'080_0000 | H'040_0000 | H'020_0000 | H'017_384E | H'015_5555 | H'00A_AAAA | — |
| 22.05 | H'0B0_6666 | H'058_3333 | H'02C_1999 | H'020_0000 | H'01D_6666 | H'00E_B333 | — |
| 24 | H'0C0_0000 | H'060_0000 | H'030_0000 | H'022_D476 | H'020_0000 | H'010_0000 | — |
| 32 | H'100_0000 | H'080_0000 | H'040_0000 | H'02E_709D | H'02A_AAAA | H'015_5555 | H'00A_AAAA |
| 44.1 | H'160_CCCC | H'0B0_6666 | H'058_3333 | H'040_0000 | H'03A_CCCC | H'01D_6666 | H'00E_B333 |
| 48 | H'180_0000 | H'0C0_0000 | H'060_0000 | H'045_A8EC | H'040_0000 | H'020_0000 | H'010_0000 |
| 64 | — | H'100_0000 | H'080_0000 | H'05C_E13B | H'055_5555 | H'02A_AAAA | H'015_5555 |
| 88.2 | — | H'160_CCCC | H'0B0_6666 | H'080_0000 | H'075_9999 | H'03A_CCCC | H'01D_6666 |
| 96 | — | H'180_0000 | H'0C0_0000 | H'08B_51D9 | H'080_0000 | H'040_0000 | H'020_0000 |
| 176.4 | — | H'2C1_9999 | H'160_CCCC | H'100_0000 | H'0EB_3333 | H'075_9999 | H'03A_CCCC |
| 192 | — | — | H'180_0000 | H'116_A3B3 | H'100_0000 | H'080_0000 | H'040_0000 |

(1) Register Setting Procedure

The following describes the SRC register setting procedures. The register should be used according to the following procedures.

(a) SRC Processing Procedure

Figure 43.6 shows the processing procedure of SRC.

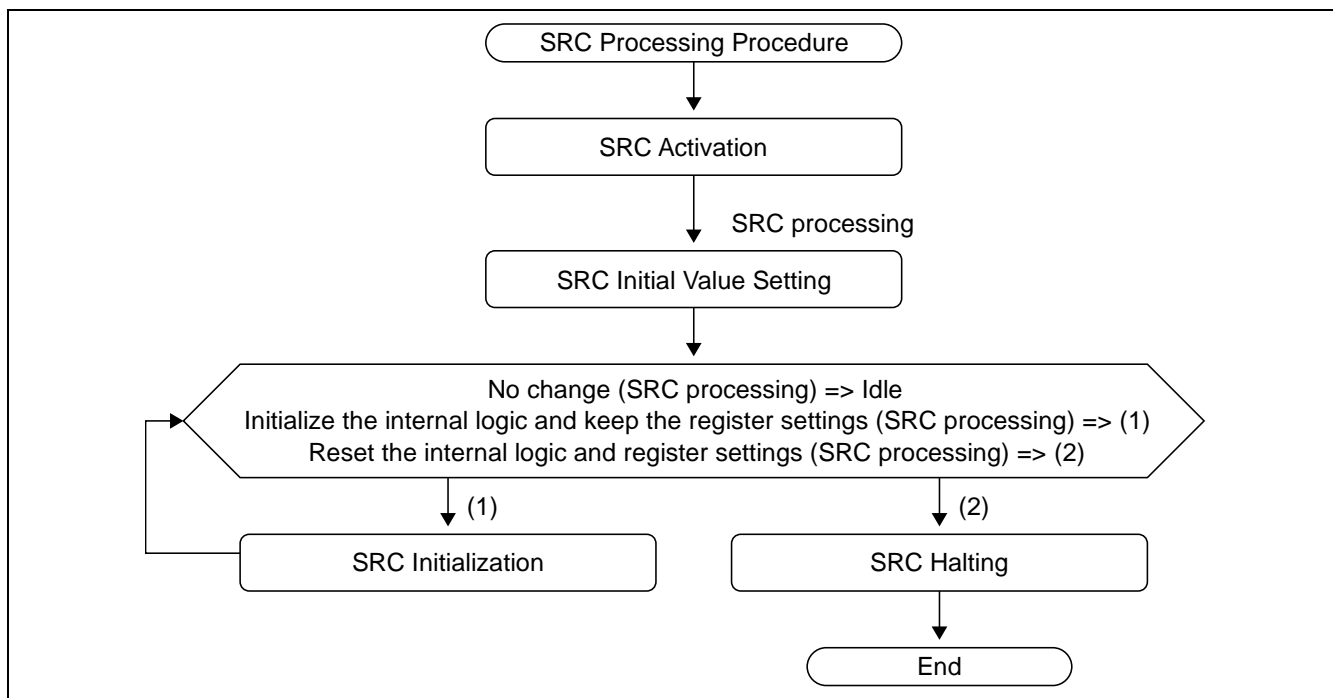


Figure 43.6 SRC Processing Procedure

(b) SRC Activation

Figure 43.7 shows the SRC activation flowchart. When the SRC is activated, a software reset should be set to initialize logic and register setting. Then, the software reset should be cleared and ready to operate SRC. When reset the SRC function, all registers are initialized. If the SRC function was initialized by a hardware reset, it doesn't need to initialize the SRC function by software reset.

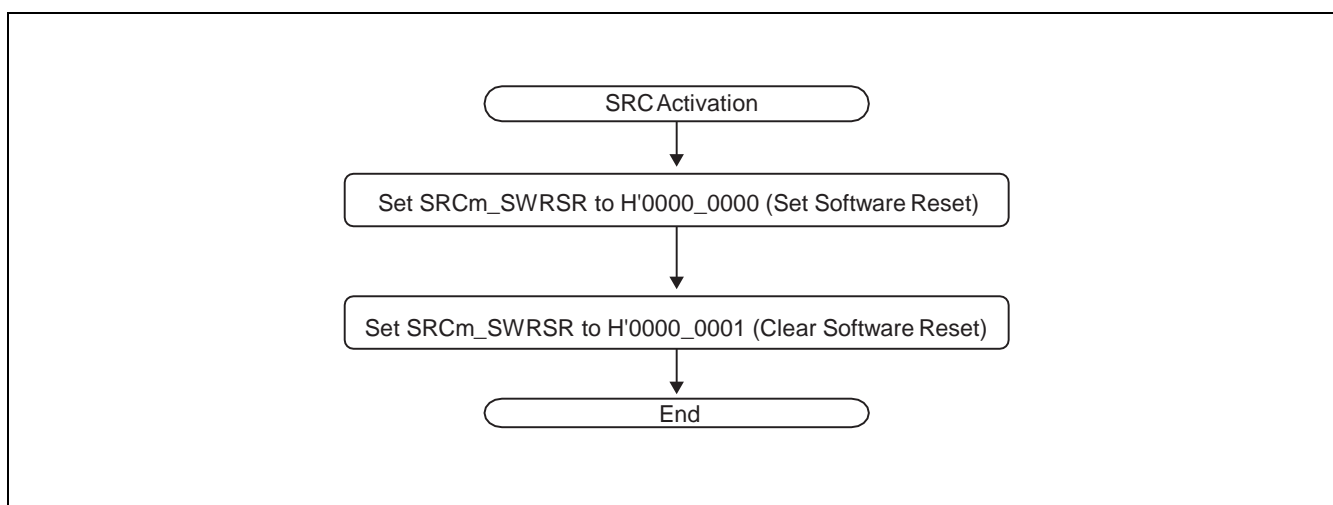


Figure 43.7 SRC Activation Flowchart

(c) SRC Initial Value Setting

Figure 43.8 shows the flowchart of the SRC initial value setting. Before operate SRC function, the initial values should be set.

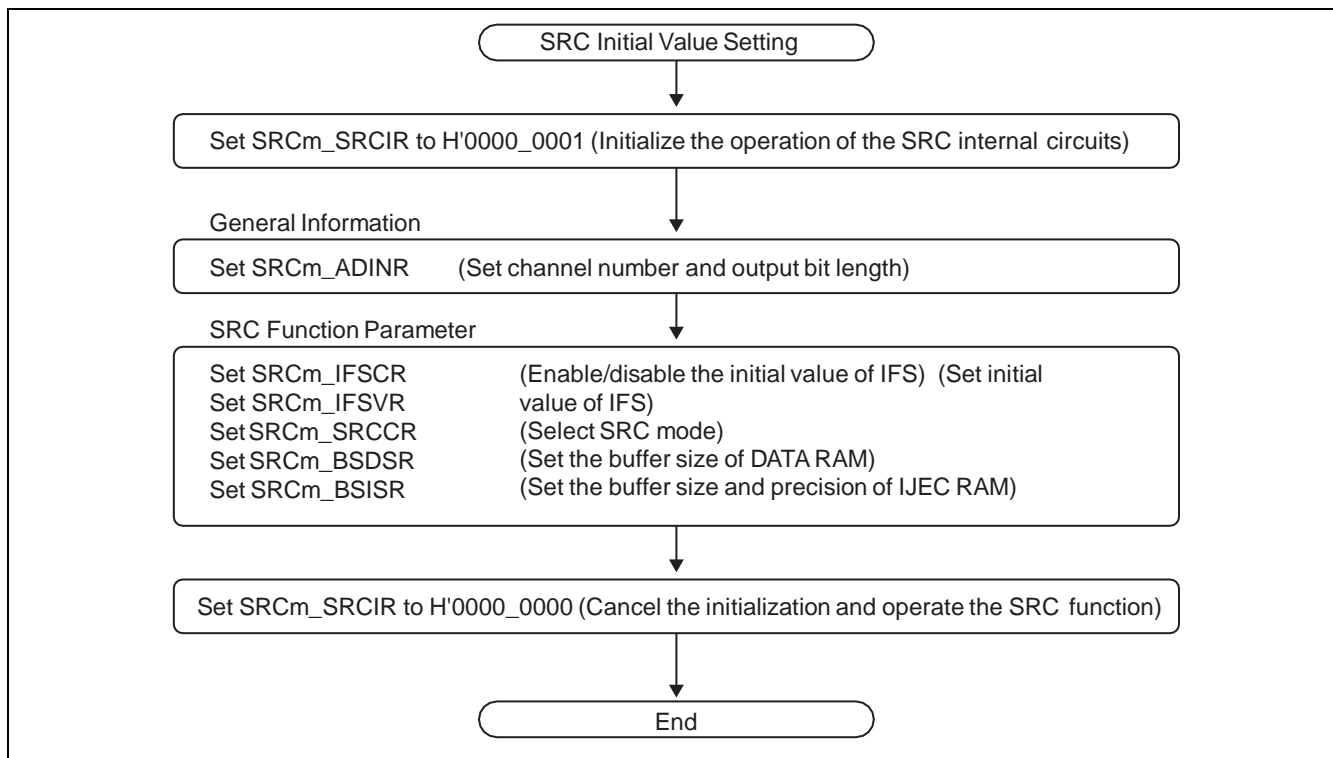


Figure 43.8 SRC Initial Value Setting Flowchart

(d) SRC Initialization

Figure 43.9 shows the SRC initialization flowchart. SRCm_SRCIR register doesn't initialize the register settings and these values are maintained. Before cancel the initialization, it is necessary to set or check the register settings of peripheral IP. And also it is necessary to provide the input and output timing signal.

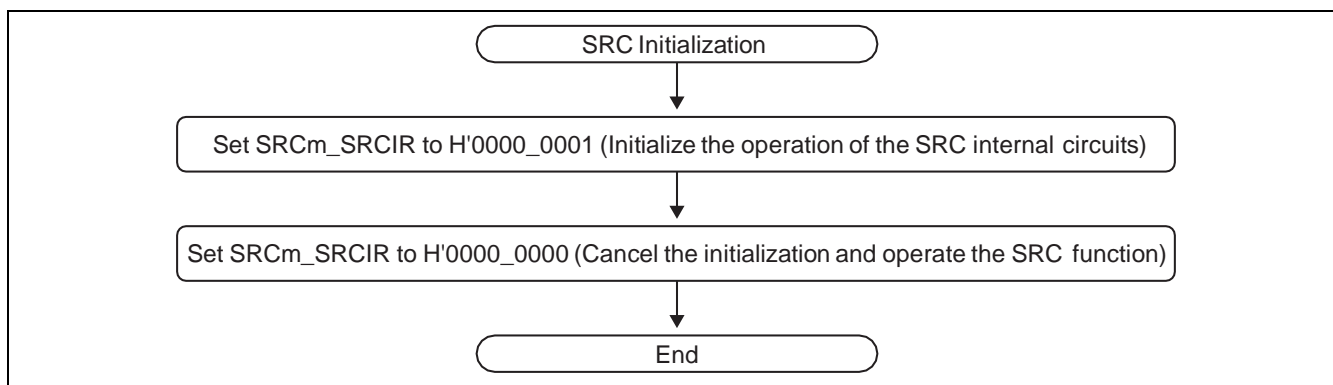


Figure 43.9 SRC Initialization Flowchart

(e) SRC Halting

Figure 43.10 shows the flowchart of SRC halting. When SRC is halting, it should be initialize or reset by software reset or hardware reset. Before initialize or reset the SRC function, it is necessary to confirm the register settings and operation of peripheral IP.

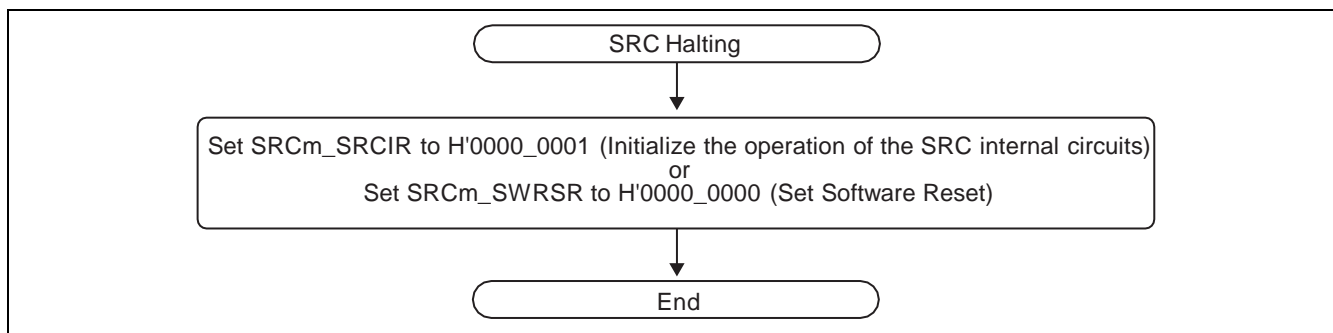


Figure 43.10 SRC Halting Flowchart

43.3.6 CMD Block

The CMD is a block for implementing the channel transfer unit (CTU block), mixing (MIX block), and digital volume and mute (DVC block) functions. Figure 43.11 shows the CMD block diagram.

The "Input audio data 0" to "Input audio data 3" in the figure indicate the data before the CMD function is applied. The "Output audio data" indicates the data after the CMD function is applied.

The "Output data timing" is a signal to show the sampling period for the output audio data. In addition, give the same signals to CMD output timing signal and output data timing signal used in the SRC.

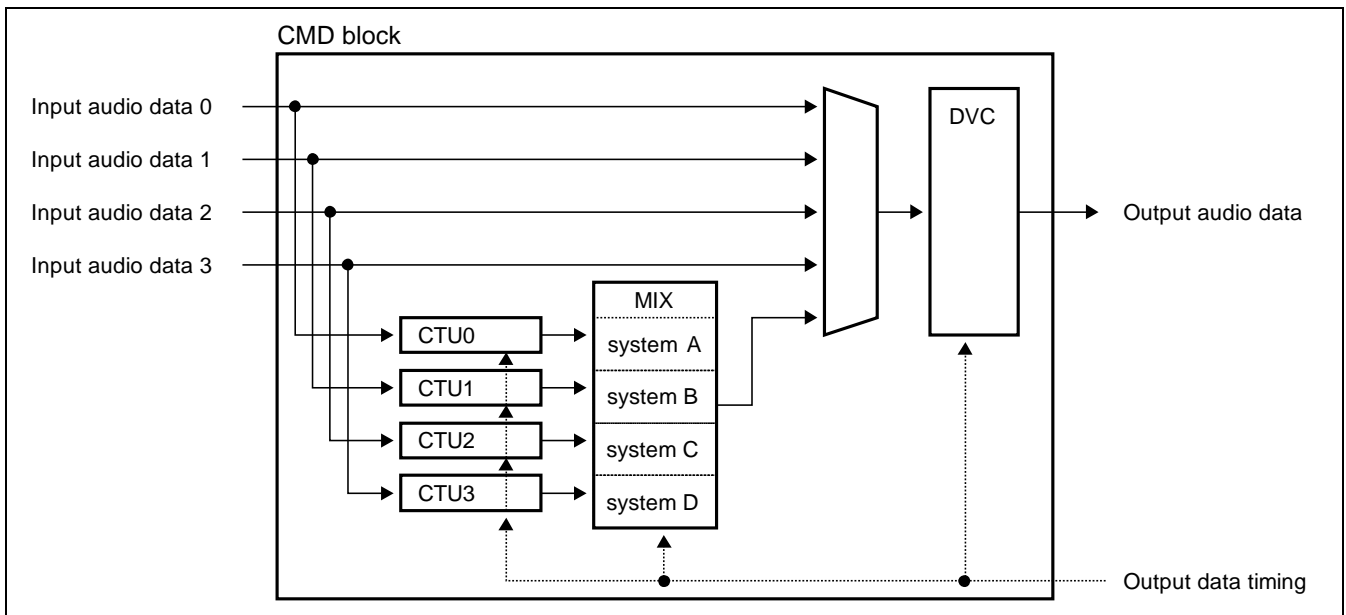


Figure 43.11 CMD Block Diagram

43.3.7 Functional Blocks in CMD

(1) CTU Block

The CTU is a block for implementing channel transfer unit functions such as downmixing and splitter functions.

- Downmixing and splitter functions
 - Conversion of eight input channels into two output channels
 - Conversion of six input channels into two output channels
 - Conversion of two input channels into four sets of two output channels
 - Conversion of one input channel into eight sets of one output channel
 - No conversion

The settings of the Scale Value e00 to Scale Value e37 registers are calculated by using the following formula (signed bits are excluded):

$$\text{Setting (decimal)} = 10^X \times 4194304$$

$$X = (\text{value in dB}) / 20$$

(a) Register Setting Procedure

The following describes the CTU register setting procedures. The register should be used according to the following procedures.

- CTU Processing Procedure

Figure 43.12 shows the processing procedure of CTU.

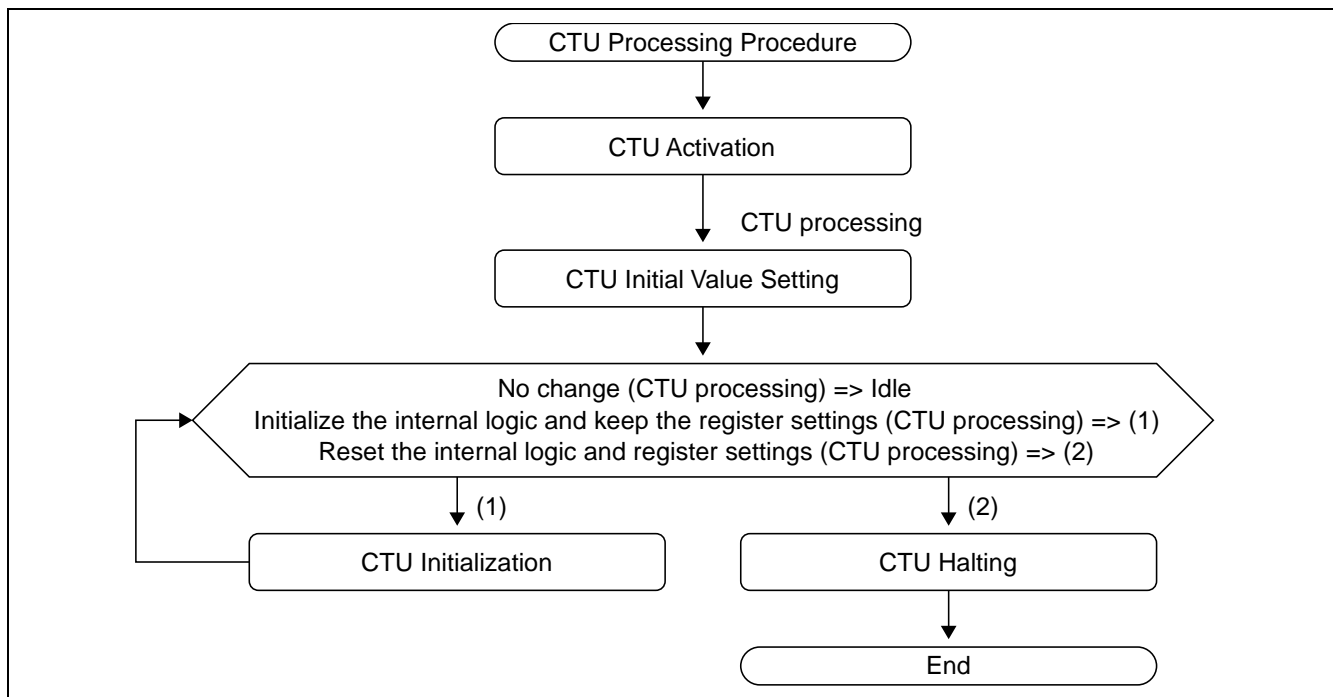


Figure 43.12 CTU Processing Procedure

- CTU Activation

Figure 43.13 shows the CTU activation flowchart. When the CTU is activated, a software reset should be set to initialize logic and register setting. Then, the software reset should be cleared and ready to operate CTU. When reset the CTU function, all registers are initialized. If the CTU function was initialized by a hardware reset, it doesn't need to initialize the CTU function by software reset.

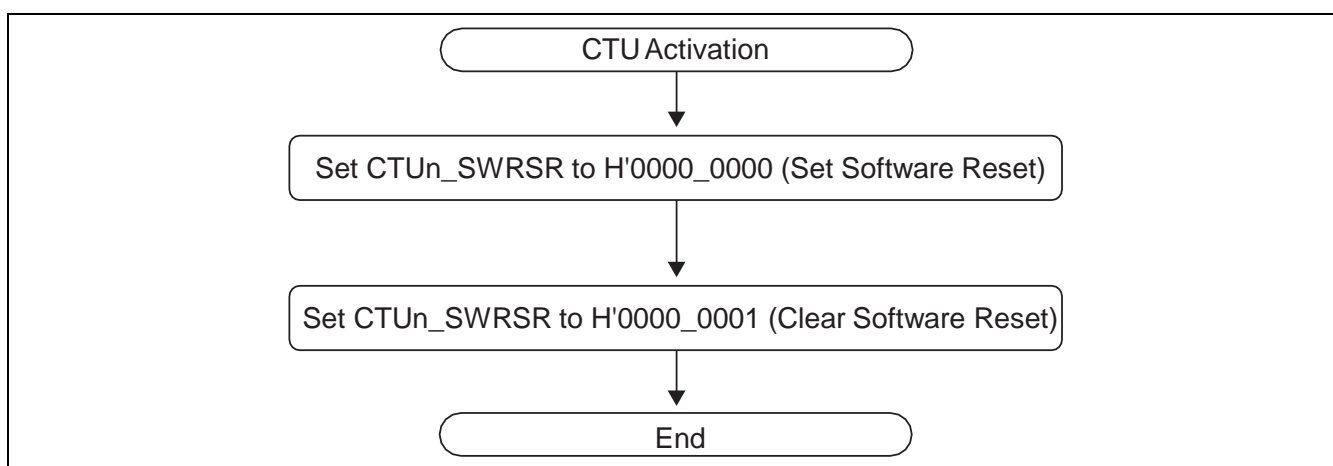


Figure 43.13 CTU Activation Flowchart

- CTU Initial Value Setting

Figure 43.14 shows the flowchart of the CTU initial value setting. Before operate CTU function, the initial values should be set. After set the register setting and cancel the initialization, CTU module starts the operation.

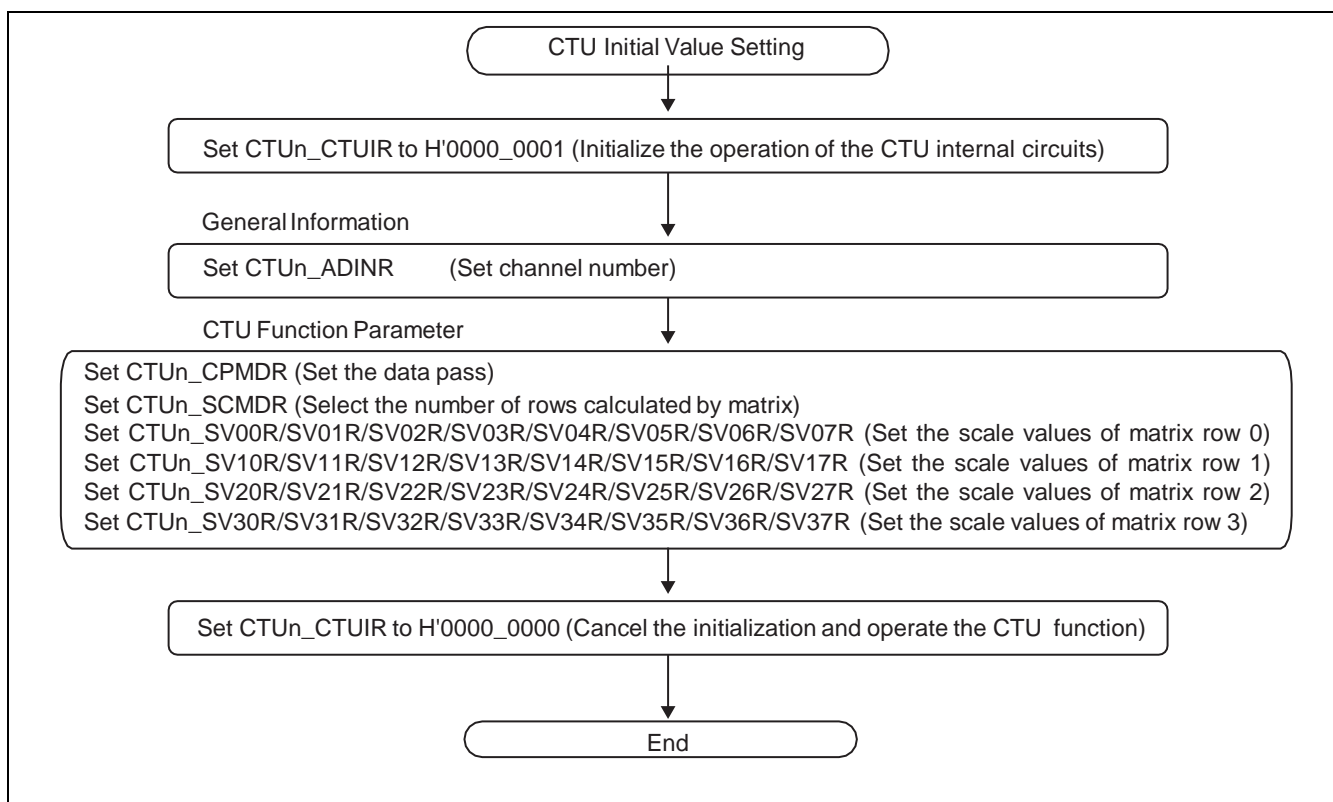


Figure 43.14 CTU Initial Value Setting Flowchart

- CTU Initialization

Figure 43.15 shows the CTU initialization flowchart. CTUn_CTUIR register doesn't initialize the register settings and these values are maintained. Before cancel the initialization, it is necessary to set or check the register settings of peripheral IP.

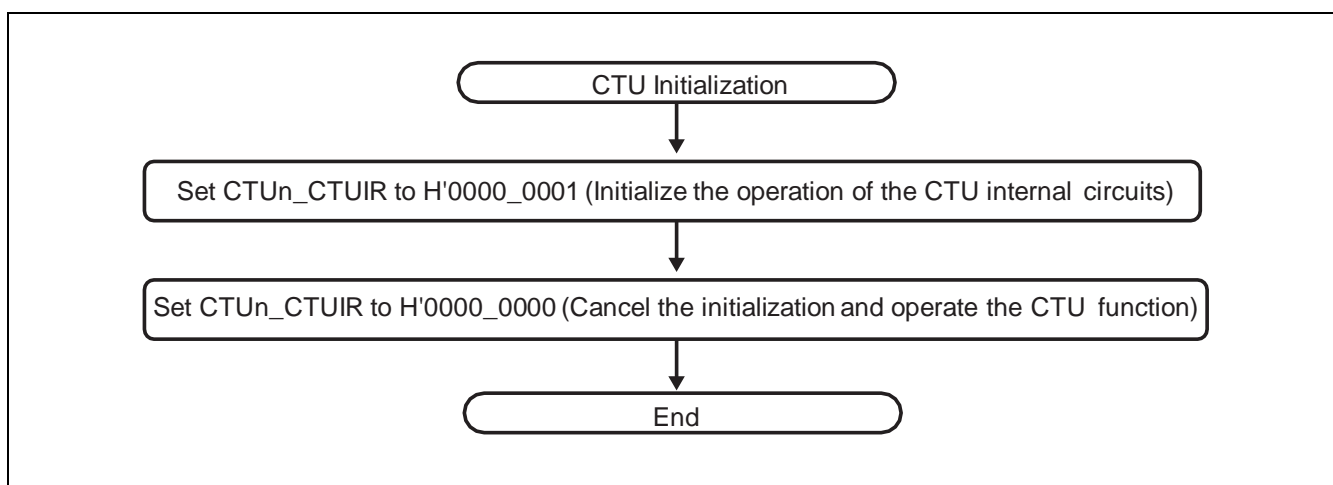


Figure 43.15 CTU Initialization Flowchart

- CTU Halting

Figure 43.16 shows the flowchart of CTU halting. When CTU is halting, it should be initialize or reset by software reset or hardware reset. Before initialize or reset the CTU function, it is necessary to confirm the register settings and operation of peripheral IP.

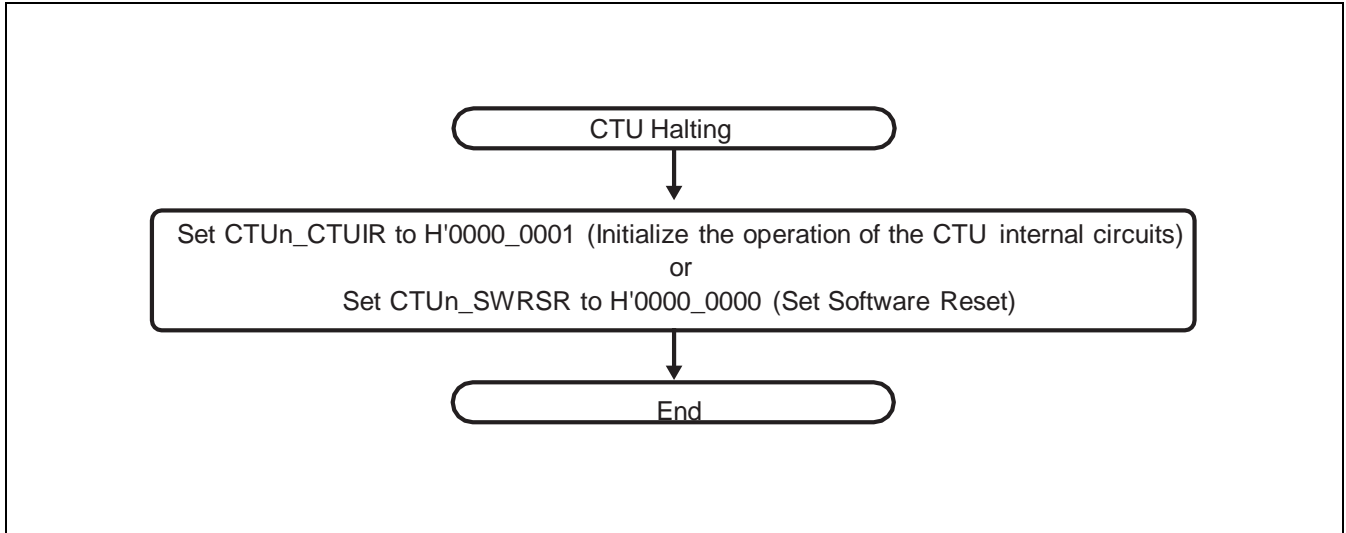


Figure 43.16 CTU Halting Flowchart

- CTU – Example

The channel conversion executes the linear transformation (matrix operation) as follows.

$$\begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{pmatrix} = \begin{pmatrix} e_{00} & e_{01} & e_{02} & e_{03} & e_{04} & e_{05} & e_{06} & e_{07} \\ e_{10} & e_{11} & e_{12} & e_{13} & e_{14} & e_{15} & e_{16} & e_{17} \\ e_{20} & e_{21} & e_{22} & e_{23} & e_{24} & e_{25} & e_{26} & e_{27} \\ e_{30} & e_{31} & e_{32} & e_{33} & e_{34} & e_{35} & e_{36} & e_{37} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix}$$

x_i : Input data of channel i ($i = 0$ to 7)

y_i : Output data of channel i ($i = 0$ to 7)

e_j : CTUn_SVjR.SVEj ($j = 00, 01, 02, 03, 04, 05, 06, 07, 10, 11, 12, 13, 14, 15, 16, 17, 20, 21, 22, 23, 24, 25, 26, 27, 30, 31, 32, 33, 34, 35, 36, 37$)

Figure 43.17 shows 2 channels fold-down.

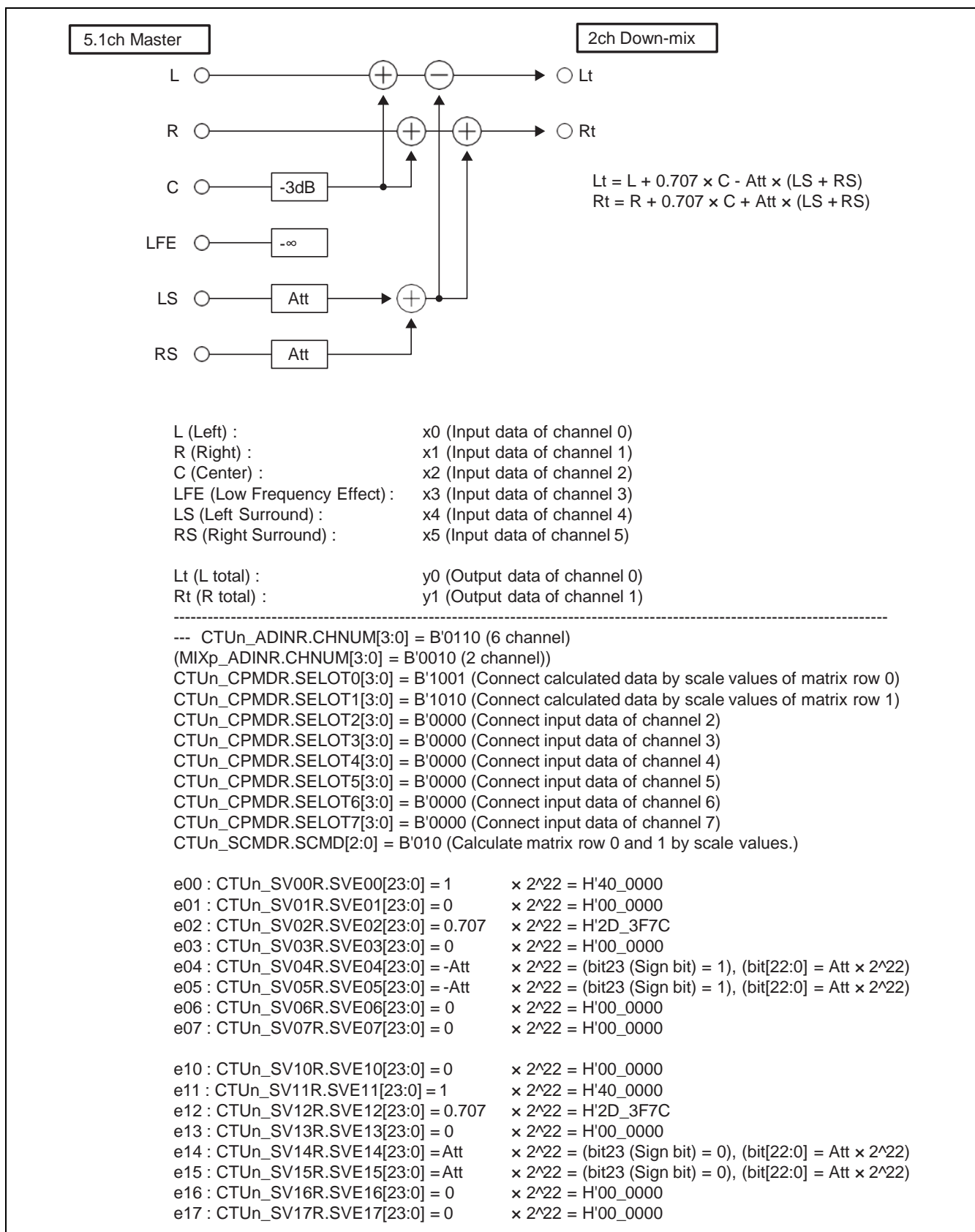


Figure 43.17 2 channels fold-down (Lt/Rt Downmix)

(2) MIX Block

The MIX block is for mixing (adding) streams from two to four audio data sources into a single stream.

- Ratio for adding sources is selectable
- Ratio is dynamically changeable
- Mixing with volume ramp is available (ramp period is selectable)

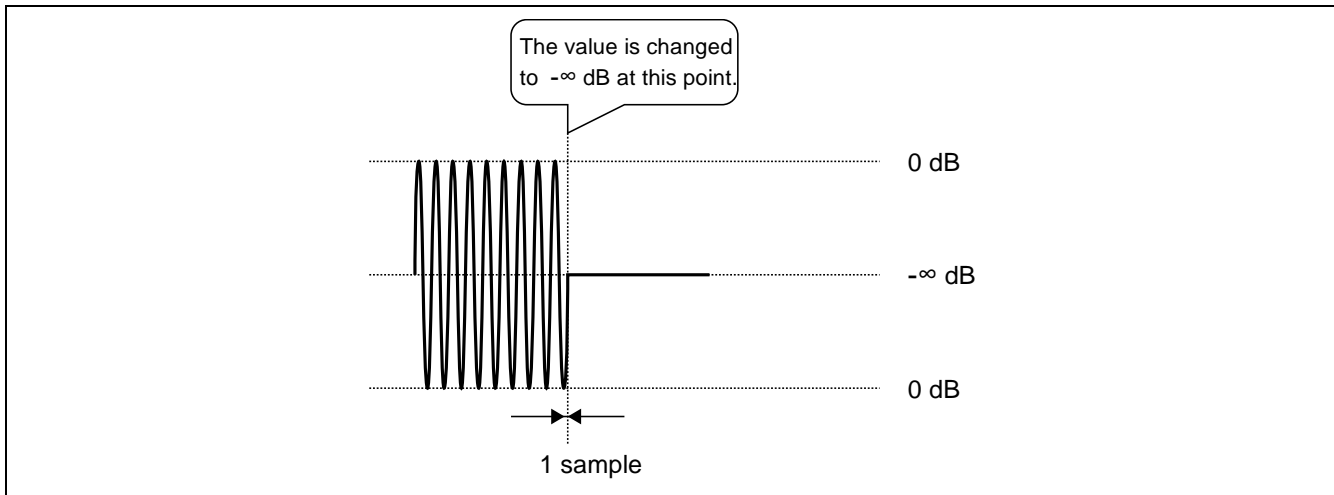


Figure 43.18 Step Processing

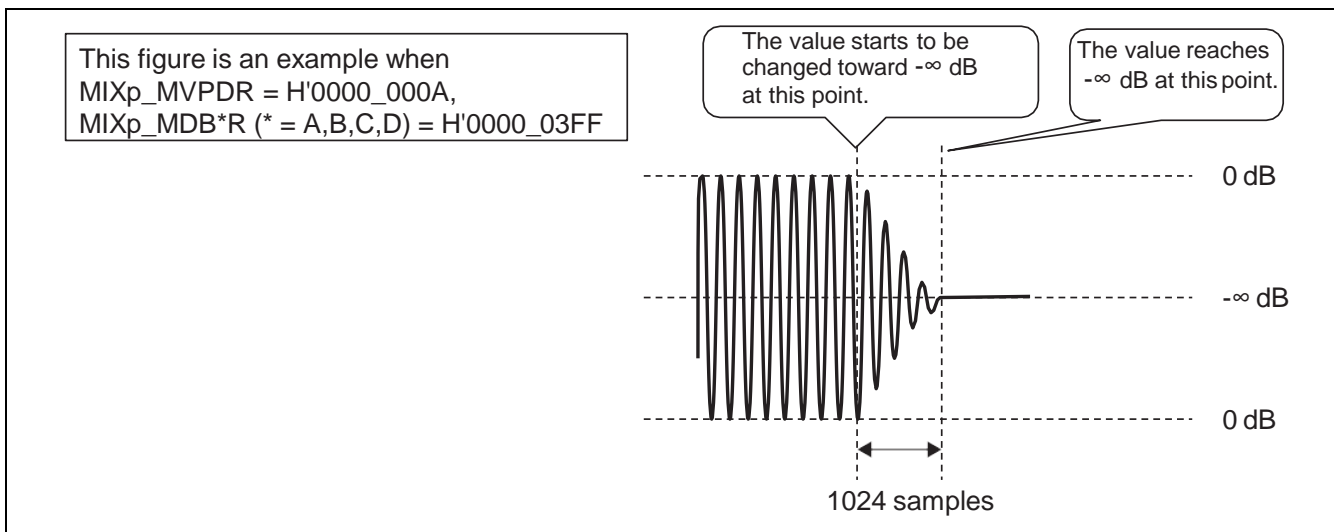


Figure 43.19 Ramp Processing

Table 43.11 to Table 43.14 show the settings of the MIX decibel register and the corresponding levels in decibels.

Table 43.11 Settings of MIXp_MDBAR, MIXp_MDBBR, MIXp_MDBCR, and MIXp_MDBDR and Corresponding Values in Decibels (No. 0 to 255)

| No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB |
|----|-------------|--------|----|-------------|--------|-----|-------------|---------|-----|-------------|---------|
| 0 | 000 | 0 | 39 | 027 | -4.875 | 78 | 04E | -9.75 | 117 | 075 | -14.625 |
| 1 | 001 | -0.125 | 40 | 028 | -5 | 79 | 04F | -9.875 | 118 | 076 | -14.75 |
| 2 | 002 | -0.25 | 41 | 029 | -5.125 | 80 | 050 | -10 | 119 | 077 | -14.875 |
| 3 | 003 | -0.375 | 42 | 02A | -5.25 | 81 | 051 | -10.125 | 120 | 078 | -15 |
| 4 | 004 | -0.5 | 43 | 02B | -5.375 | 82 | 052 | -10.25 | 121 | 079 | -15.125 |
| 5 | 005 | -0.625 | 44 | 02C | -5.5 | 83 | 053 | -10.375 | 122 | 07A | -15.25 |
| 6 | 006 | -0.75 | 45 | 02D | -5.625 | 84 | 054 | -10.5 | 123 | 07B | -15.375 |
| 7 | 007 | -0.875 | 46 | 02E | -5.75 | 85 | 055 | -10.625 | 124 | 07C | -15.5 |
| 8 | 008 | -1 | 47 | 02F | -5.875 | 86 | 056 | -10.75 | 125 | 07D | -15.625 |
| 9 | 009 | -1.125 | 48 | 030 | -6 | 87 | 057 | -10.875 | 126 | 07E | -15.75 |
| 10 | 00A | -1.25 | 49 | 031 | -6.125 | 88 | 058 | -11 | 127 | 07F | -15.875 |
| 11 | 00B | -1.375 | 50 | 032 | -6.25 | 89 | 059 | -11.125 | 128 | 080 | -16 |
| 12 | 00C | -1.5 | 51 | 033 | -6.375 | 90 | 05A | -11.25 | 129 | 081 | -16.125 |
| 13 | 00D | -1.625 | 52 | 034 | -6.5 | 91 | 05B | -11.375 | 130 | 082 | -16.25 |
| 14 | 00E | -1.75 | 53 | 035 | -6.625 | 92 | 05C | -11.5 | 131 | 083 | -16.375 |
| 15 | 00F | -1.875 | 54 | 036 | -6.75 | 93 | 05D | -11.625 | 132 | 084 | -16.5 |
| 16 | 010 | -2 | 55 | 037 | -6.875 | 94 | 05E | -11.75 | 133 | 085 | -16.625 |
| 17 | 011 | -2.125 | 56 | 038 | -7 | 95 | 05F | -11.875 | 134 | 086 | -16.75 |
| 18 | 012 | -2.25 | 57 | 039 | -7.125 | 96 | 060 | -12 | 135 | 087 | -16.875 |
| 19 | 013 | -2.375 | 58 | 03A | -7.25 | 97 | 061 | -12.125 | 136 | 088 | -17 |
| 20 | 014 | -2.5 | 59 | 03B | -7.375 | 98 | 062 | -12.25 | 137 | 089 | -17.125 |
| 21 | 015 | -2.625 | 60 | 03C | -7.5 | 99 | 063 | -12.375 | 138 | 08A | -17.25 |
| 22 | 016 | -2.75 | 61 | 03D | -7.625 | 100 | 064 | -12.5 | 139 | 08B | -17.375 |
| 23 | 017 | -2.875 | 62 | 03E | -7.75 | 101 | 065 | -12.625 | 140 | 08C | -17.5 |
| 24 | 018 | -3 | 63 | 03F | -7.875 | 102 | 066 | -12.75 | 141 | 08D | -17.625 |
| 25 | 019 | -3.125 | 64 | 040 | -8 | 103 | 067 | -12.875 | 142 | 08E | -17.75 |
| 26 | 01A | -3.25 | 65 | 041 | -8.125 | 104 | 068 | -13 | 143 | 08F | -17.875 |
| 27 | 01B | -3.375 | 66 | 042 | -8.25 | 105 | 069 | -13.125 | 144 | 090 | -18 |
| 28 | 01C | -3.5 | 67 | 043 | -8.375 | 106 | 06A | -13.25 | 145 | 091 | -18.125 |
| 29 | 01D | -3.625 | 68 | 044 | -8.5 | 107 | 06B | -13.375 | 146 | 092 | -18.25 |
| 30 | 01E | -3.75 | 69 | 045 | -8.625 | 108 | 06C | -13.5 | 147 | 093 | -18.375 |
| 31 | 01F | -3.875 | 70 | 046 | -8.75 | 109 | 06D | -13.625 | 148 | 094 | -18.5 |
| 32 | 020 | -4 | 71 | 047 | -8.875 | 110 | 06E | -13.75 | 149 | 095 | -18.625 |
| 33 | 021 | -4.125 | 72 | 048 | -9 | 111 | 06F | -13.875 | 150 | 096 | -18.75 |
| 34 | 022 | -4.25 | 73 | 049 | -9.125 | 112 | 070 | -14 | 151 | 097 | -18.875 |
| 35 | 023 | -4.375 | 74 | 04A | -9.25 | 113 | 071 | -14.125 | 152 | 098 | -19 |
| 36 | 024 | -4.5 | 75 | 04B | -9.375 | 114 | 072 | -14.25 | 153 | 099 | -19.125 |
| 37 | 025 | -4.625 | 76 | 04C | -9.5 | 115 | 073 | -14.375 | 154 | 09A | -19.25 |
| 38 | 026 | -4.75 | 77 | 04D | -9.625 | 116 | 074 | -14.5 | 155 | 09B | -19.375 |

| No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB |
|-----|-------------|---------|-----|-------------|---------|-----|-------------|---------|-----|-------------|---------|
| 156 | 09C | -19.5 | 181 | 0B5 | -22.625 | 206 | 0CE | -25.75 | 231 | 0E7 | -28.875 |
| 157 | 09D | -19.625 | 182 | 0B6 | -22.75 | 207 | 0CF | -25.875 | 232 | 0E8 | -29 |
| 158 | 09E | -19.75 | 183 | 0B7 | -22.875 | 208 | 0D0 | -26 | 233 | 0E9 | -29.125 |
| 159 | 09F | -19.875 | 184 | 0B8 | -23 | 209 | 0D1 | -26.125 | 234 | 0EA | -29.25 |
| 160 | 0A0 | -20 | 185 | 0B9 | -23.125 | 210 | 0D2 | -26.25 | 235 | 0EB | -29.375 |
| 161 | 0A1 | -20.125 | 186 | 0BA | -23.25 | 211 | 0D3 | -26.375 | 236 | 0EC | -29.5 |
| 162 | 0A2 | -20.25 | 187 | 0BB | -23.375 | 212 | 0D4 | -26.5 | 237 | 0ED | -29.625 |
| 163 | 0A3 | -20.375 | 188 | 0BC | -23.5 | 213 | 0D5 | -26.625 | 238 | 0EE | -29.75 |
| 164 | 0A4 | -20.5 | 189 | 0BD | -23.625 | 214 | 0D6 | -26.75 | 239 | 0EF | -29.875 |
| 165 | 0A5 | -20.625 | 190 | 0BE | -23.75 | 215 | 0D7 | -26.875 | 240 | 0F0 | -30 |
| 166 | 0A6 | -20.75 | 191 | 0BF | -23.875 | 216 | 0D8 | -27 | 241 | 0F1 | -30.125 |
| 167 | 0A7 | -20.875 | 192 | 0C0 | -24 | 217 | 0D9 | -27.125 | 242 | 0F2 | -30.25 |
| 168 | 0A8 | -21 | 193 | 0C1 | -24.125 | 218 | 0DA | -27.25 | 243 | 0F3 | -30.375 |
| 169 | 0A9 | -21.125 | 194 | 0C2 | -24.25 | 219 | 0DB | -27.375 | 244 | 0F4 | -30.5 |
| 170 | 0AA | -21.25 | 195 | 0C3 | -24.375 | 220 | 0DC | -27.5 | 245 | 0F5 | -30.625 |
| 171 | 0AB | -21.375 | 196 | 0C4 | -24.5 | 221 | 0DD | -27.625 | 246 | 0F6 | -30.75 |
| 172 | 0AC | -21.5 | 197 | 0C5 | -24.625 | 222 | 0DE | -27.75 | 247 | 0F7 | -30.875 |
| 173 | 0AD | -21.625 | 198 | 0C6 | -24.75 | 223 | 0DF | -27.875 | 248 | 0F8 | -31 |
| 174 | 0AE | -21.75 | 199 | 0C7 | -24.875 | 224 | 0E0 | -28 | 249 | 0F9 | -31.125 |
| 175 | 0AF | -21.875 | 200 | 0C8 | -25 | 225 | 0E1 | -28.125 | 250 | 0FA | -31.25 |
| 176 | 0B0 | -22 | 201 | 0C9 | -25.125 | 226 | 0E2 | -28.25 | 251 | 0FB | -31.375 |
| 177 | 0B1 | -22.125 | 202 | 0CA | -25.25 | 227 | 0E3 | -28.375 | 252 | 0FC | -31.5 |
| 178 | 0B2 | -22.25 | 203 | 0CB | -25.375 | 228 | 0E4 | -28.5 | 253 | 0FD | -31.625 |
| 179 | 0B3 | -22.375 | 204 | 0CC | -25.5 | 229 | 0E5 | -28.625 | 254 | 0FE | -31.75 |
| 180 | 0B4 | -22.5 | 205 | 0CD | -25.625 | 230 | 0E6 | -28.75 | 255 | 0FF | -31.875 |

Table 43.12 Settings of MIXp_MDBAR, MIXp_MDBBR, MIXp_MDBCR, and MIXp_MDBDR and Corresponding Values in Decibels (No. 256 to 511)

| No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB |
|-----|-------------|---------|-----|-------------|---------|-----|-------------|---------|-----|-------------|---------|
| 256 | 100 | -32 | 297 | 129 | -37.125 | 338 | 152 | -42.25 | 379 | 17B | -47.375 |
| 257 | 101 | -32.125 | 298 | 12A | -37.25 | 339 | 153 | -42.375 | 380 | 17C | -47.5 |
| 258 | 102 | -32.25 | 299 | 12B | -37.375 | 340 | 154 | -42.5 | 381 | 17D | -47.625 |
| 259 | 103 | -32.375 | 300 | 12C | -37.5 | 341 | 155 | -42.625 | 382 | 17E | -47.75 |
| 260 | 104 | -32.5 | 301 | 12D | -37.625 | 342 | 156 | -42.75 | 383 | 17F | -47.875 |
| 261 | 105 | -32.625 | 302 | 12E | -37.75 | 343 | 157 | -42.875 | 384 | 180 | -48 |
| 262 | 106 | -32.75 | 303 | 12F | -37.875 | 344 | 158 | -43 | 385 | 181 | -48.125 |
| 263 | 107 | -32.875 | 304 | 130 | -38 | 345 | 159 | -43.125 | 386 | 182 | -48.25 |
| 264 | 108 | -33 | 305 | 131 | -38.125 | 346 | 15A | -43.25 | 387 | 183 | -48.375 |
| 265 | 109 | -33.125 | 306 | 132 | -38.25 | 347 | 15B | -43.375 | 388 | 184 | -48.5 |
| 266 | 10A | -33.25 | 307 | 133 | -38.375 | 348 | 15C | -43.5 | 389 | 185 | -48.625 |
| 267 | 10B | -33.375 | 308 | 134 | -38.5 | 349 | 15D | -43.625 | 390 | 186 | -48.75 |
| 268 | 10C | -33.5 | 309 | 135 | -38.625 | 350 | 15E | -43.75 | 391 | 187 | -48.875 |
| 269 | 10D | -33.625 | 310 | 136 | -38.75 | 351 | 15F | -43.875 | 392 | 188 | -49 |
| 270 | 10E | -33.75 | 311 | 137 | -38.875 | 352 | 160 | -44 | 393 | 189 | -49.125 |
| 271 | 10F | -33.875 | 312 | 138 | -39 | 353 | 161 | -44.125 | 394 | 18A | -49.25 |
| 272 | 110 | -34 | 313 | 139 | -39.125 | 354 | 162 | -44.25 | 395 | 18B | -49.375 |
| 273 | 111 | -34.125 | 314 | 13A | -39.25 | 355 | 163 | -44.375 | 396 | 18C | -49.5 |
| 274 | 112 | -34.25 | 315 | 13B | -39.375 | 356 | 164 | -44.5 | 397 | 18D | -49.625 |
| 275 | 113 | -34.375 | 316 | 13C | -39.5 | 357 | 165 | -44.625 | 398 | 18E | -49.75 |
| 276 | 114 | -34.5 | 317 | 13D | -39.625 | 358 | 166 | -44.75 | 399 | 18F | -49.875 |
| 277 | 115 | -34.625 | 318 | 13E | -39.75 | 359 | 167 | -44.875 | 400 | 190 | -50 |
| 278 | 116 | -34.75 | 319 | 13F | -39.875 | 360 | 168 | -45 | 401 | 191 | -50.125 |
| 279 | 117 | -34.875 | 320 | 140 | -40 | 361 | 169 | -43.125 | 402 | 192 | -50.25 |
| 280 | 118 | -35 | 321 | 141 | -40.125 | 362 | 16A | -43.25 | 403 | 193 | -50.375 |
| 281 | 119 | -35.125 | 322 | 142 | -40.25 | 363 | 16B | -43.375 | 404 | 194 | -50.5 |
| 282 | 11A | -35.25 | 323 | 143 | -40.375 | 364 | 16C | -45.5 | 405 | 195 | -50.625 |
| 283 | 11B | -35.375 | 324 | 144 | -40.5 | 365 | 16D | -45.625 | 406 | 196 | -50.75 |
| 284 | 11C | -35.5 | 325 | 145 | -40.625 | 366 | 16E | -45.75 | 407 | 197 | -50.875 |
| 285 | 11D | -35.625 | 326 | 146 | -40.75 | 367 | 16F | -45.875 | 408 | 198 | -51 |
| 286 | 11E | -35.75 | 327 | 147 | -40.875 | 368 | 170 | -46 | 409 | 199 | -51.125 |
| 287 | 11F | -35.875 | 328 | 148 | -41 | 369 | 171 | -46.125 | 410 | 19A | -51.25 |
| 288 | 120 | -36 | 329 | 149 | -41.125 | 370 | 172 | -46.25 | 411 | 19B | -51.375 |
| 289 | 121 | -36.125 | 330 | 14A | -41.25 | 371 | 173 | -46.375 | 412 | 19C | -51.5 |
| 290 | 122 | -36.25 | 331 | 14B | -41.375 | 372 | 174 | -46.5 | 413 | 19D | -51.625 |
| 291 | 123 | -36.375 | 332 | 14C | -41.5 | 373 | 175 | -46.625 | 414 | 19E | -51.75 |
| 292 | 124 | -36.5 | 333 | 14D | -41.625 | 374 | 176 | -46.75 | 415 | 19F | -51.875 |
| 293 | 125 | -36.625 | 334 | 14E | -41.75 | 375 | 177 | -46.875 | 416 | 1A0 | -52 |
| 294 | 126 | -36.75 | 335 | 14F | -41.875 | 376 | 178 | -47 | 417 | 1A1 | -52.125 |
| 295 | 127 | -36.875 | 336 | 150 | -42 | 377 | 179 | -47.125 | 418 | 1A2 | -52.25 |
| 296 | 128 | -37 | 337 | 151 | -42.125 | 378 | 17A | -47.25 | 419 | 1A3 | -52.375 |

| No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB |
|-----|-------------|---------|-----|-------------|---------|-----|-------------|---------|-----|-------------|---------|
| 420 | 1A4 | -52.5 | 443 | 1BB | -55.375 | 466 | 1D2 | -58.25 | 489 | 1E9 | -61.125 |
| 421 | 1A5 | -52.625 | 444 | 1BC | -55.5 | 467 | 1D3 | -58.375 | 490 | 1EA | -61.25 |
| 422 | 1A6 | -52.75 | 445 | 1BD | -55.625 | 468 | 1D4 | -58.5 | 491 | 1EB | -61.375 |
| 423 | 1A7 | -52.875 | 446 | 1BE | -55.75 | 469 | 1D5 | -58.625 | 492 | 1EC | -61.5 |
| 424 | 1A8 | -53 | 447 | 1BF | -55.875 | 470 | 1D6 | -58.75 | 493 | 1ED | -61.625 |
| 425 | 1A9 | -53.125 | 448 | 1C0 | -56 | 471 | 1D7 | -58.875 | 494 | 1EE | -61.75 |
| 426 | 1AA | -53.25 | 449 | 1C1 | -56.125 | 472 | 1D8 | -59 | 495 | 1EF | -61.875 |
| 427 | 1AB | -53.375 | 450 | 1C2 | -56.25 | 473 | 1D9 | -59.125 | 496 | 1F0 | -62 |
| 428 | 1AC | -53.5 | 451 | 1C3 | -56.375 | 474 | 1DA | -59.25 | 497 | 1F1 | -62.125 |
| 429 | 1AD | -53.625 | 452 | 1C4 | -56.5 | 475 | 1DB | -59.375 | 498 | 1F2 | -62.25 |
| 430 | 1AE | -53.75 | 453 | 1C5 | -56.625 | 476 | 1DC | -59.5 | 499 | 1F3 | -62.375 |
| 431 | 1AF | -53.875 | 454 | 1C6 | -56.75 | 477 | 1DD | -59.625 | 500 | 1F4 | -62.5 |
| 432 | 1B0 | -54 | 455 | 1C7 | -56.875 | 478 | 1DE | -59.75 | 501 | 1F5 | -62.625 |
| 433 | 1B1 | -54.125 | 456 | 1C8 | -57 | 479 | 1DF | -59.875 | 502 | 1F6 | -62.75 |
| 434 | 1B2 | -54.25 | 457 | 1C9 | -57.125 | 480 | 1E0 | -60 | 503 | 1F7 | -62.875 |
| 435 | 1B3 | -54.375 | 458 | 1CA | -57.25 | 481 | 1E1 | -60.125 | 504 | 1F8 | -63 |
| 436 | 1B4 | -54.5 | 459 | 1CB | -57.375 | 482 | 1E2 | -60.25 | 505 | 1F9 | -63.125 |
| 437 | 1B5 | -54.625 | 460 | 1CC | -57.5 | 483 | 1E3 | -60.375 | 506 | 1FA | -63.25 |
| 438 | 1B6 | -54.75 | 461 | 1CD | -57.625 | 484 | 1E4 | -60.5 | 507 | 1FB | -63.375 |
| 439 | 1B7 | -54.875 | 462 | 1CE | -57.75 | 485 | 1E5 | -60.625 | 508 | 1FC | -63.5 |
| 440 | 1B8 | -55 | 463 | 1CF | -57.875 | 486 | 1E6 | -60.75 | 509 | 1FD | -63.625 |
| 441 | 1B9 | -55.125 | 464 | 1D0 | -58 | 487 | 1E7 | -60.875 | 510 | 1FE | -63.75 |
| 442 | 1BA | -55.25 | 465 | 1D1 | -58.125 | 488 | 1E8 | -61 | 511 | 1FF | -63.875 |

Table 43.13 Settings of MIXp_MDBAR, MIXp_MDBBR, MIXp_MDBCR, and MIXp_MDBDR and Corresponding Values in Decibels (No. 512 to 767)

| No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB |
|-----|-------------|---------|-----|-------------|---------|-----|-------------|---------|-----|-------------|---------|
| 512 | 200 | -64 | 553 | 229 | -69.125 | 594 | 252 | -74.25 | 635 | 27B | -79.375 |
| 513 | 201 | -64.125 | 554 | 22A | -69.25 | 595 | 253 | -74.375 | 636 | 27C | -79.5 |
| 514 | 202 | -64.25 | 555 | 22B | -69.375 | 596 | 254 | -74.5 | 637 | 27D | -79.625 |
| 515 | 203 | -64.375 | 556 | 22C | -69.5 | 597 | 255 | -74.625 | 638 | 27E | -79.75 |
| 516 | 204 | -64.5 | 557 | 22D | -69.625 | 598 | 256 | -74.75 | 639 | 27F | -79.875 |
| 517 | 205 | -64.625 | 558 | 22E | -69.75 | 599 | 257 | -74.875 | 640 | 280 | -80 |
| 518 | 206 | -64.75 | 559 | 22F | -69.875 | 600 | 258 | -75 | 641 | 281 | -80.125 |
| 519 | 207 | -64.875 | 560 | 230 | -70 | 601 | 259 | -75.125 | 642 | 282 | -80.25 |
| 520 | 208 | -65 | 561 | 231 | -70.125 | 602 | 25A | -75.25 | 643 | 283 | -80.375 |
| 521 | 209 | -65.125 | 562 | 232 | -70.25 | 603 | 25B | -75.375 | 644 | 284 | -80.5 |
| 522 | 20A | -65.25 | 563 | 233 | -70.375 | 604 | 25C | -75.5 | 645 | 285 | -80.625 |
| 523 | 20B | -65.375 | 564 | 234 | -70.5 | 605 | 25D | -75.625 | 646 | 286 | -80.75 |
| 524 | 20C | -65.5 | 565 | 235 | -70.625 | 606 | 25E | -75.75 | 647 | 287 | -80.875 |
| 525 | 20D | -65.625 | 566 | 236 | -70.75 | 607 | 25F | -75.875 | 648 | 288 | -81 |
| 526 | 20E | -65.75 | 567 | 237 | -70.875 | 608 | 260 | -76 | 649 | 289 | -81.125 |
| 527 | 20F | -65.875 | 568 | 238 | -71 | 609 | 261 | -76.125 | 650 | 28A | -81.25 |
| 528 | 210 | -66 | 569 | 239 | -71.125 | 610 | 262 | -76.25 | 651 | 28B | -81.375 |
| 529 | 211 | -66.125 | 570 | 23A | -71.25 | 611 | 263 | -76.375 | 652 | 28C | -81.5 |
| 530 | 212 | -66.25 | 571 | 23B | -71.375 | 612 | 264 | -76.5 | 653 | 28D | -81.625 |
| 531 | 213 | -66.375 | 572 | 23C | -71.5 | 613 | 265 | -76.625 | 654 | 28E | -81.75 |
| 532 | 214 | -66.5 | 573 | 23D | -71.625 | 614 | 266 | -76.75 | 655 | 28F | -81.875 |
| 533 | 215 | -66.625 | 574 | 23E | -71.75 | 615 | 267 | -76.875 | 656 | 290 | -82 |
| 534 | 216 | -66.75 | 575 | 23F | -71.875 | 616 | 268 | -77 | 657 | 291 | -82.125 |
| 535 | 217 | -66.875 | 576 | 240 | -72 | 617 | 269 | -77.125 | 658 | 292 | -82.25 |
| 536 | 218 | -67 | 577 | 241 | -72.125 | 618 | 26A | -77.25 | 659 | 293 | -82.375 |
| 537 | 219 | -67.125 | 578 | 242 | -72.25 | 619 | 26B | -77.375 | 660 | 294 | -82.5 |
| 538 | 21A | -67.25 | 579 | 243 | -72.375 | 620 | 26C | -77.5 | 661 | 295 | -82.625 |
| 539 | 21B | -67.375 | 580 | 244 | -72.5 | 621 | 26D | -77.625 | 662 | 296 | -82.75 |
| 540 | 21C | -67.5 | 581 | 245 | -72.625 | 622 | 26E | -77.75 | 663 | 297 | -82.875 |
| 541 | 21D | -67.625 | 582 | 246 | -72.75 | 623 | 26F | -77.875 | 664 | 298 | -83 |
| 542 | 21E | -67.75 | 583 | 247 | -72.875 | 624 | 270 | -78 | 665 | 299 | -83.125 |
| 543 | 21F | -67.875 | 584 | 248 | -73 | 625 | 271 | -78.125 | 666 | 29A | -83.25 |
| 544 | 220 | -68 | 585 | 249 | -73.125 | 626 | 272 | -78.25 | 667 | 29B | -83.375 |
| 545 | 221 | -68.125 | 586 | 24A | -73.25 | 627 | 273 | -78.375 | 668 | 29C | -83.5 |
| 546 | 222 | -68.25 | 587 | 24B | -73.375 | 628 | 274 | -78.5 | 669 | 29D | -83.625 |
| 547 | 223 | -68.375 | 588 | 24C | -73.5 | 629 | 275 | -78.625 | 670 | 29E | -83.75 |
| 548 | 224 | -68.5 | 589 | 24D | -73.625 | 630 | 276 | -78.75 | 671 | 29F | -83.875 |
| 549 | 225 | -68.625 | 590 | 24E | -73.75 | 631 | 277 | -78.875 | 672 | 2A0 | -84 |
| 550 | 226 | -68.75 | 591 | 24F | -73.875 | 632 | 278 | -79 | 673 | 2A1 | -84.125 |
| 551 | 227 | -68.875 | 592 | 250 | -74 | 633 | 279 | -79.125 | 674 | 2A2 | -84.25 |
| 552 | 228 | -69 | 593 | 251 | -74.125 | 634 | 27A | -79.25 | 675 | 2A3 | -84.375 |

| No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB |
|-----|-------------|---------|-----|-------------|---------|-----|-------------|---------|-----|-------------|---------|
| 676 | 2A4 | -84.5 | 699 | 2BB | -87.375 | 722 | 2D2 | -90.25 | 745 | 2E9 | -93.125 |
| 677 | 2A5 | -84.625 | 700 | 2BC | -87.5 | 723 | 2D3 | -90.375 | 746 | 2EA | -93.25 |
| 678 | 2A6 | -84.75 | 701 | 2BD | -87.625 | 724 | 2D4 | -90.5 | 747 | 2EB | -93.375 |
| 679 | 2A7 | -84.875 | 702 | 2BE | -87.75 | 725 | 2D5 | -90.625 | 748 | 2EC | -93.5 |
| 680 | 2A8 | -85 | 703 | 2BF | -87.875 | 726 | 2D6 | -90.75 | 749 | 2ED | -93.625 |
| 681 | 2A9 | -85.125 | 704 | 2C0 | -88 | 727 | 2D7 | -90.875 | 750 | 2EE | -93.75 |
| 682 | 2AA | -85.25 | 705 | 2C1 | -88.125 | 728 | 2D8 | -91 | 751 | 2EF | -93.875 |
| 683 | 2AB | -85.375 | 706 | 2C2 | -88.25 | 729 | 2D9 | -91.125 | 752 | 2F0 | -94 |
| 684 | 2AC | -85.5 | 707 | 2C3 | -88.375 | 730 | 2DA | -91.25 | 753 | 2F1 | -94.125 |
| 685 | 2AD | -85.625 | 708 | 2C4 | -88.5 | 731 | 2DB | -91.375 | 754 | 2F2 | -94.25 |
| 686 | 2AE | -85.75 | 709 | 2C5 | -88.625 | 732 | 2DC | -91.5 | 755 | 2F3 | -94.375 |
| 687 | 2AF | -85.875 | 710 | 2C6 | -88.75 | 733 | 2DD | -91.625 | 756 | 2F4 | -94.5 |
| 688 | 2B0 | -86 | 711 | 2C7 | -88.875 | 734 | 2DE | -91.75 | 757 | 2F5 | -94.625 |
| 689 | 2B1 | -86.125 | 712 | 2C8 | -89 | 735 | 2DF | -91.875 | 758 | 2F6 | -94.75 |
| 690 | 2B2 | -86.25 | 713 | 2C9 | -89.125 | 736 | 2E0 | -92 | 759 | 2F7 | -94.875 |
| 691 | 2B3 | -86.375 | 714 | 2CA | -89.25 | 737 | 2E1 | -92.125 | 760 | 2F8 | -95 |
| 692 | 2B4 | -86.5 | 715 | 2CB | -89.375 | 738 | 2E2 | -92.25 | 761 | 2F9 | -95.125 |
| 693 | 2B5 | -86.625 | 716 | 2CC | -89.5 | 739 | 2E3 | -92.375 | 762 | 2FA | -95.25 |
| 694 | 2B6 | -86.75 | 717 | 2CD | -89.625 | 740 | 2E4 | -92.5 | 763 | 2FB | -95.375 |
| 695 | 2B7 | -86.875 | 718 | 2CE | -89.75 | 741 | 2E5 | -92.625 | 764 | 2FC | -95.5 |
| 696 | 2B8 | -87 | 719 | 2CF | -89.875 | 742 | 2E6 | -92.75 | 765 | 2FD | -95.625 |
| 697 | 2B9 | -87.125 | 720 | 2D0 | -90 | 743 | 2E7 | -92.875 | 766 | 2FE | -95.75 |
| 698 | 2BA | -87.25 | 721 | 2D1 | -90.125 | 744 | 2E8 | -93 | 767 | 2FF | -95.875 |

Table 43.14 Settings of MIXp_MDBAR, MIXp_MDBBR, MIXp_MDBCR, and MIXp_MDBDR and Corresponding Values in Decibels (No. 768 to 1023)

| No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB |
|-----|-------------|----------|-----|-------------|----------|-----|-------------|----------|-----|-------------|----------|
| 768 | 300 | -96 | 809 | 329 | -101.125 | 850 | 352 | -106.25 | 891 | 37B | -111.375 |
| 769 | 301 | -96.125 | 810 | 32A | -101.25 | 851 | 353 | -106.375 | 892 | 37C | -111.5 |
| 770 | 302 | -96.25 | 811 | 32B | -101.375 | 852 | 354 | -106.5 | 893 | 37D | -111.625 |
| 771 | 303 | -96.375 | 812 | 32C | -101.5 | 853 | 355 | -106.625 | 894 | 37E | -111.75 |
| 772 | 304 | -96.5 | 813 | 32D | -101.625 | 854 | 356 | -106.75 | 895 | 37F | -111.875 |
| 773 | 305 | -96.625 | 814 | 32E | -101.75 | 855 | 357 | -106.875 | 896 | 380 | -112 |
| 774 | 306 | -96.75 | 815 | 32F | -101.875 | 856 | 358 | -107 | 897 | 381 | -112.125 |
| 775 | 307 | -96.875 | 816 | 330 | -102 | 857 | 359 | -107.125 | 898 | 382 | -112.25 |
| 776 | 308 | -97 | 817 | 331 | -102.125 | 858 | 35A | -107.25 | 899 | 383 | -112.375 |
| 777 | 309 | -97.125 | 818 | 332 | -102.25 | 859 | 35B | -107.375 | 900 | 384 | -112.5 |
| 778 | 30A | -97.25 | 819 | 333 | -102.375 | 860 | 35C | -107.5 | 901 | 385 | -112.625 |
| 779 | 30B | -97.375 | 820 | 334 | -102.5 | 861 | 35D | -107.625 | 902 | 386 | -112.75 |
| 780 | 30C | -97.5 | 821 | 335 | -102.625 | 862 | 35E | -107.75 | 903 | 387 | -112.875 |
| 781 | 30D | -97.625 | 822 | 336 | -102.75 | 863 | 35F | -107.875 | 904 | 388 | -113 |
| 782 | 30E | -97.75 | 823 | 337 | -102.875 | 864 | 360 | -108 | 905 | 389 | -113.125 |
| 783 | 30F | -97.875 | 824 | 338 | -103 | 865 | 361 | -108.125 | 906 | 38A | -113.25 |
| 784 | 310 | -98 | 825 | 339 | -103.125 | 866 | 362 | -108.25 | 907 | 38B | -113.375 |
| 785 | 311 | -98.125 | 826 | 33A | -103.25 | 867 | 363 | -108.375 | 908 | 38C | -113.5 |
| 786 | 312 | -98.25 | 827 | 33B | -103.375 | 868 | 364 | -108.5 | 909 | 38D | -113.625 |
| 787 | 313 | -98.375 | 828 | 33C | -103.5 | 869 | 365 | -108.625 | 910 | 38E | -113.75 |
| 788 | 314 | -98.5 | 829 | 33D | -103.625 | 870 | 366 | -108.75 | 911 | 38F | -113.875 |
| 789 | 315 | -98.625 | 830 | 33E | -103.75 | 871 | 367 | -108.875 | 912 | 390 | -114 |
| 790 | 316 | -98.75 | 831 | 33F | -103.875 | 872 | 368 | -109 | 913 | 391 | -114.125 |
| 791 | 317 | -98.875 | 832 | 340 | -104 | 873 | 369 | -109.125 | 914 | 392 | -114.25 |
| 792 | 318 | -99 | 833 | 341 | -104.125 | 874 | 36A | -109.25 | 915 | 393 | -114.375 |
| 793 | 319 | -99.125 | 834 | 342 | -104.25 | 875 | 36B | -109.375 | 916 | 394 | -114.5 |
| 794 | 31A | -99.25 | 835 | 343 | -104.375 | 876 | 36C | -109.5 | 917 | 395 | -114.625 |
| 795 | 31B | -99.375 | 836 | 344 | -104.5 | 877 | 36D | -109.625 | 918 | 396 | -114.75 |
| 796 | 31C | -99.5 | 837 | 345 | -104.625 | 878 | 36E | -109.75 | 919 | 397 | -114.875 |
| 797 | 31D | -99.625 | 838 | 346 | -104.75 | 879 | 36F | -109.875 | 920 | 398 | -115 |
| 798 | 31E | -99.75 | 839 | 347 | -104.875 | 880 | 370 | -110 | 921 | 399 | -115.125 |
| 799 | 31F | -99.875 | 840 | 348 | -105 | 881 | 371 | -110.125 | 922 | 39A | -115.25 |
| 800 | 320 | -100 | 841 | 349 | -105.125 | 882 | 372 | -110.25 | 923 | 39B | -115.375 |
| 801 | 321 | -100.125 | 842 | 34A | -105.25 | 883 | 373 | -110.375 | 924 | 39C | -115.5 |
| 802 | 322 | -100.25 | 843 | 34B | -105.375 | 884 | 374 | -110.5 | 925 | 39D | -115.625 |
| 803 | 323 | -100.375 | 844 | 34C | -105.5 | 885 | 375 | -110.625 | 926 | 39E | -115.75 |
| 804 | 324 | -100.5 | 845 | 34D | -105.625 | 886 | 376 | -110.75 | 927 | 39F | -115.875 |
| 805 | 325 | -100.625 | 846 | 34E | -105.75 | 887 | 377 | -110.875 | 928 | 3A0 | -116 |
| 806 | 326 | -100.75 | 847 | 34F | -105.875 | 888 | 378 | -111 | 929 | 3A1 | -116.125 |
| 807 | 327 | -100.875 | 848 | 350 | -106 | 889 | 379 | -111.125 | 930 | 3A2 | -116.25 |
| 808 | 328 | -101 | 849 | 351 | -106.125 | 890 | 37A | -111.25 | 931 | 3A3 | -116.375 |

| No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB | No | Value (HEX) | dB |
|-----|-------------|----------|-----|-------------|----------|------|-------------|----------|------|-------------|----------|
| 932 | 3A4 | -116.5 | 955 | 3BB | -119.375 | 978 | 3D2 | -122.25 | 1001 | 3E9 | -125.125 |
| 933 | 3A5 | -116.625 | 956 | 3BC | -119.5 | 979 | 3D3 | -122.375 | 1002 | 3EA | -125.25 |
| 934 | 3A6 | -116.75 | 957 | 3BD | -119.625 | 980 | 3D4 | -122.5 | 1003 | 3EB | -125.375 |
| 935 | 3A7 | -116.875 | 958 | 3BE | -119.75 | 981 | 3D5 | -122.625 | 1004 | 3EC | -125.5 |
| 936 | 3A8 | -117 | 959 | 3BF | -119.875 | 982 | 3D6 | -122.75 | 1005 | 3ED | -125.625 |
| 937 | 3A9 | -117.125 | 960 | 3C0 | -120 | 983 | 3D7 | -122.875 | 1006 | 3EE | -125.75 |
| 938 | 3AA | -117.25 | 961 | 3C1 | -120.125 | 984 | 3D8 | -123 | 1007 | 3EF | -125.875 |
| 939 | 3AB | -117.375 | 962 | 3C2 | -120.25 | 985 | 3D9 | -123.125 | 1008 | 3F0 | -126 |
| 940 | 3AC | -117.5 | 963 | 3C3 | -120.375 | 986 | 3DA | -123.25 | 1009 | 3F1 | -126.125 |
| 941 | 3AD | -117.625 | 964 | 3C4 | -120.5 | 987 | 3DB | -123.375 | 1010 | 3F2 | -126.25 |
| 942 | 3AE | -117.75 | 965 | 3C5 | -120.625 | 988 | 3DC | -123.5 | 1011 | 3F3 | -126.375 |
| 943 | 3AF | -117.875 | 966 | 3C6 | -120.75 | 989 | 3DD | -123.625 | 1012 | 3F4 | -126.5 |
| 944 | 3B0 | -118 | 967 | 3C7 | -120.875 | 990 | 3DE | -123.75 | 1013 | 3F5 | -126.625 |
| 945 | 3B1 | -118.125 | 968 | 3C8 | -121 | 991 | 3DF | -123.875 | 1014 | 3F6 | -126.75 |
| 946 | 3B2 | -118.25 | 969 | 3C9 | -121.125 | 992 | 3E0 | -124 | 1015 | 3F7 | -126.875 |
| 947 | 3B3 | -118.375 | 970 | 3CA | -121.25 | 993 | 3E1 | -124.125 | 1016 | 3F8 | -127 |
| 948 | 3B4 | -118.5 | 971 | 3CB | -121.375 | 994 | 3E2 | -124.25 | 1017 | 3F9 | -127.125 |
| 949 | 3B5 | -118.625 | 972 | 3CC | -121.5 | 995 | 3E3 | -124.375 | 1018 | 3FA | -127.25 |
| 950 | 3B6 | -118.75 | 973 | 3CD | -121.625 | 996 | 3E4 | -124.5 | 1019 | 3FB | -127.375 |
| 951 | 3B7 | -118.875 | 974 | 3CE | -121.75 | 997 | 3E5 | -124.625 | 1020 | 3FC | -127.5 |
| 952 | 3B8 | -119 | 975 | 3CF | -121.875 | 998 | 3E6 | -124.75 | 1021 | 3FD | -127.625 |
| 953 | 3B9 | -119.125 | 976 | 3D0 | -122 | 999 | 3E7 | -124.875 | 1022 | 3FE | -127.75 |
| 954 | 3BA | -119.25 | 977 | 3D1 | -122.125 | 1000 | 3E8 | -125 | 1023 | 3FF | -∞ |

(a) Register Setting Procedure

The following describes the MIX register setting procedures. The register should be used according to the following procedures.

- MIX Processing Procedure

Figure 43.20 shows the processing procedure of MIX.

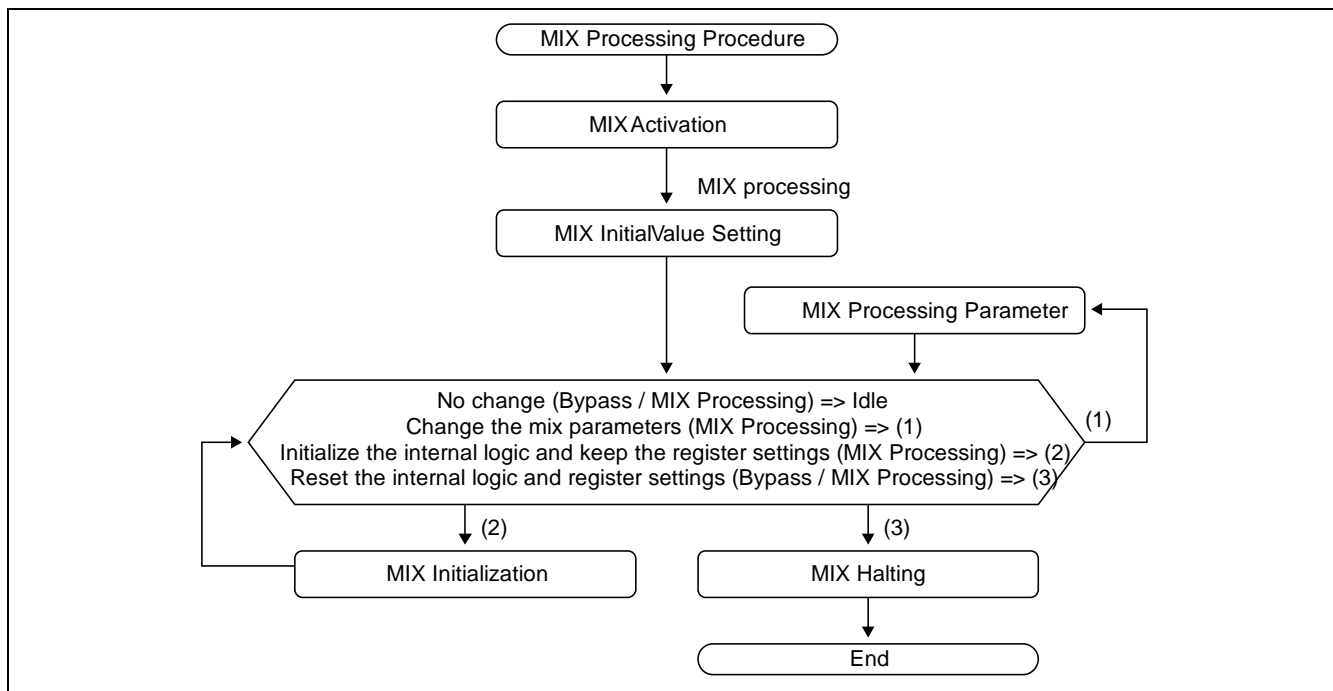


Figure 43.20 MIX Processing Procedure

- MIX Activation

Figure 43.21 shows the MIX activation flowchart. When the MIX is activated, a software reset should be set to initialize logic and register setting. Then, the software reset should be cleared and ready to operate MIX. When reset the MIX function, all registers are initialized. If the MIX function was initialized by a hardware reset, it doesn't need to initialize the MIX function by software reset.

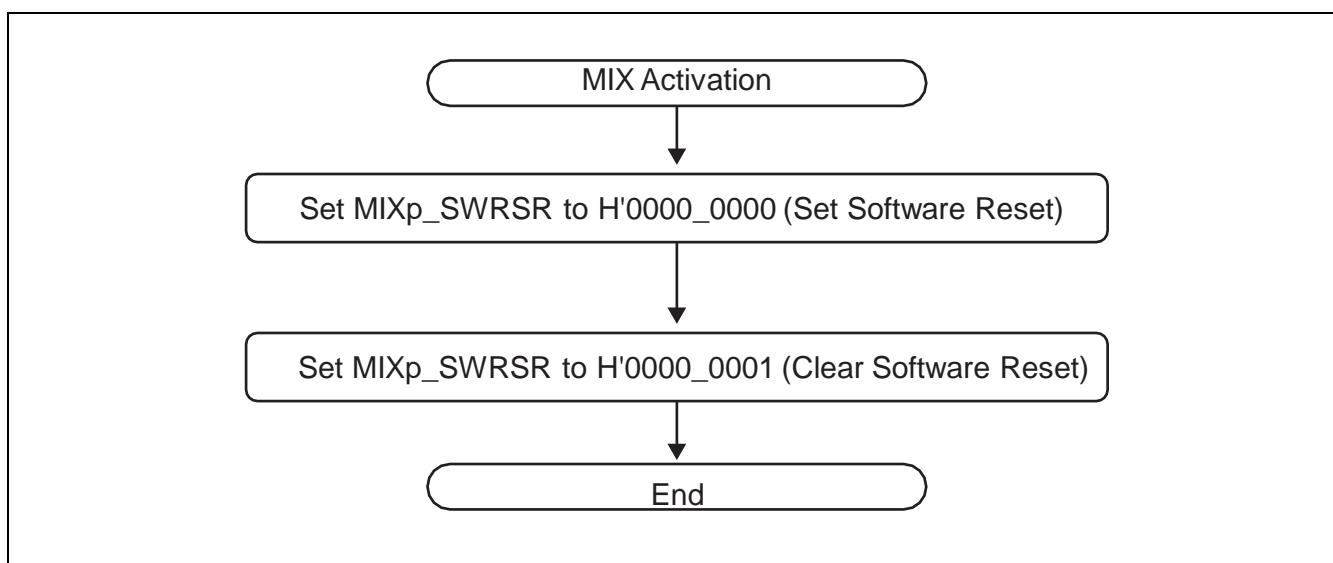


Figure 43.21 MIX Activation Flowchart

• MIX Initial Value Setting

Figure 43.22 shows the flowchart of the MIX initial value setting. Before operate MIX function, the initial values should be set. After set the register setting and cancel the initialization, MIX module changes the volume of each system by volume step to the dB value of MIXp_MDBAR and MIXp_MDBBR and MIXp_MDBCR and MIXp_MDBDR regardless of MIXp_MIXMR setting.

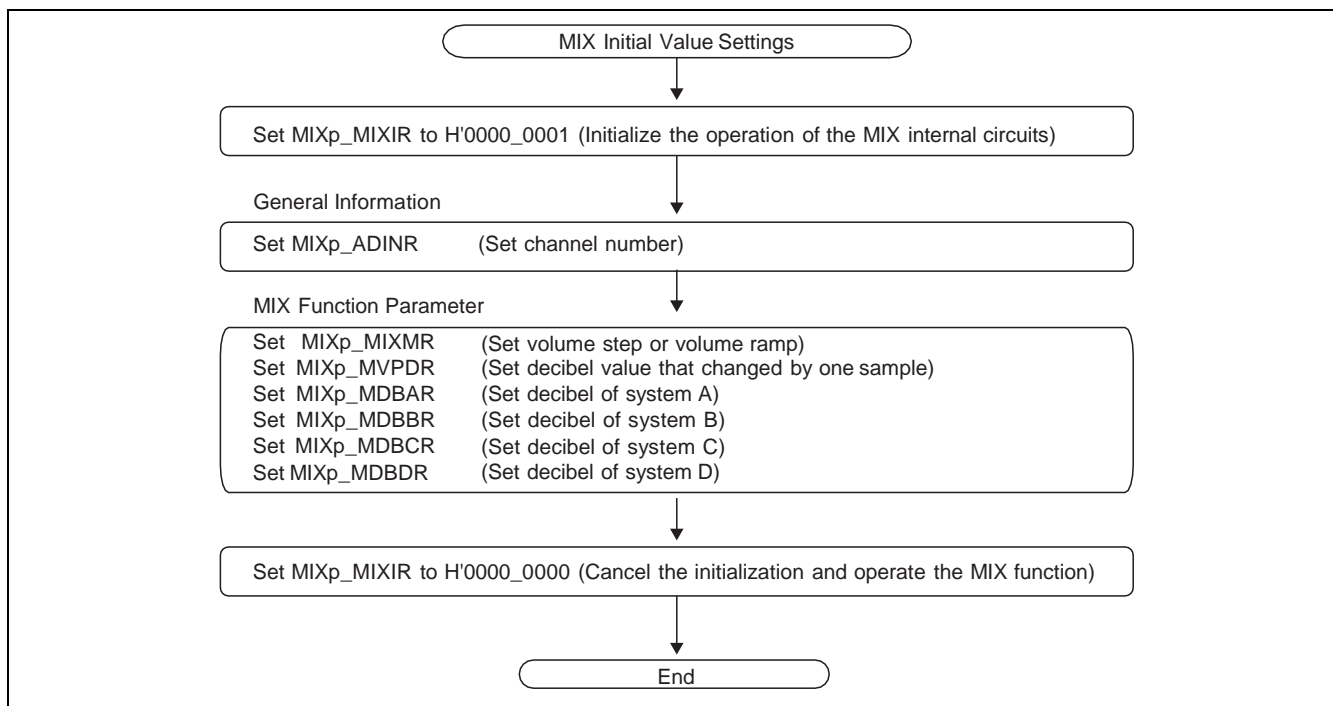


Figure 43.22 MIX Initial Value Setting Flowchart

• MIX Processing Parameter

Figure 43.23 shows the MIX processing parameter flowchart. Before change the parameter of decibel for each system, the setting of the MIXp_MDBER should be disables. After that set the decibels of MIXp_MDBAR and MIXp_MDBBR and MIXp_MDBCR and MIXp_MDBDR and enables the MIXp_MDBER for reflect the decibel setting to the MIX processing.

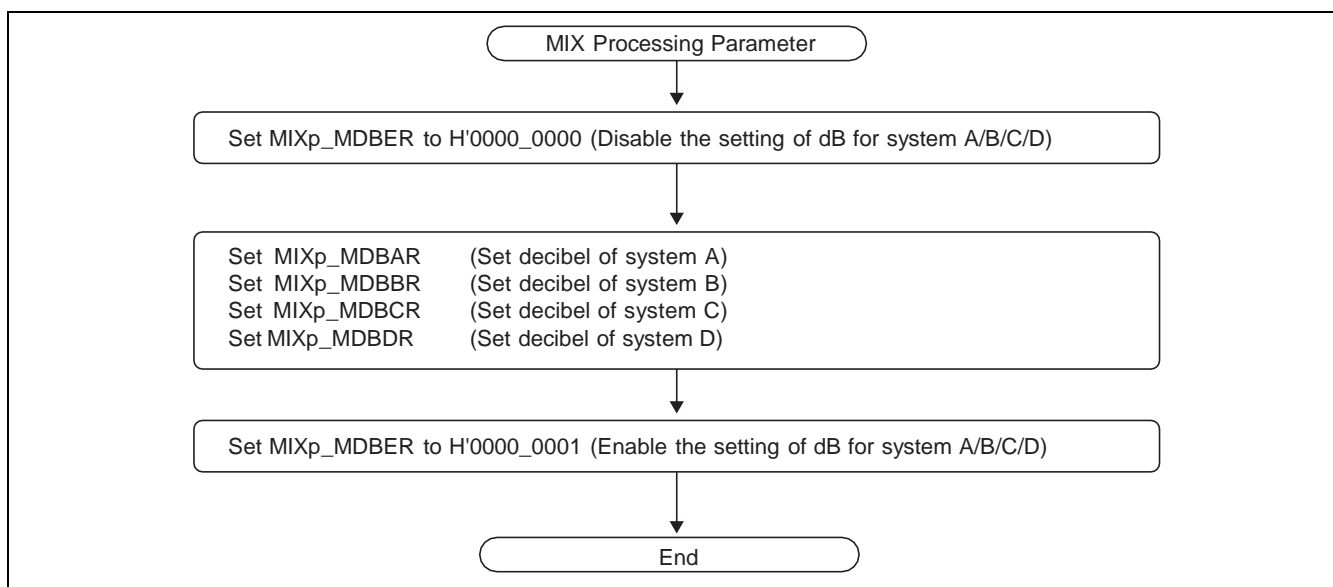


Figure 43.23 MIX Processing Parameter Flowchart

- MIX Initialization

Figure 43.24 shows the MIX initialization flowchart. MIXp_MIXIR register doesn't initialize the register settings and these values are maintained. Before cancel the initialization, it is necessary to set or check the register settings of peripheral IP.

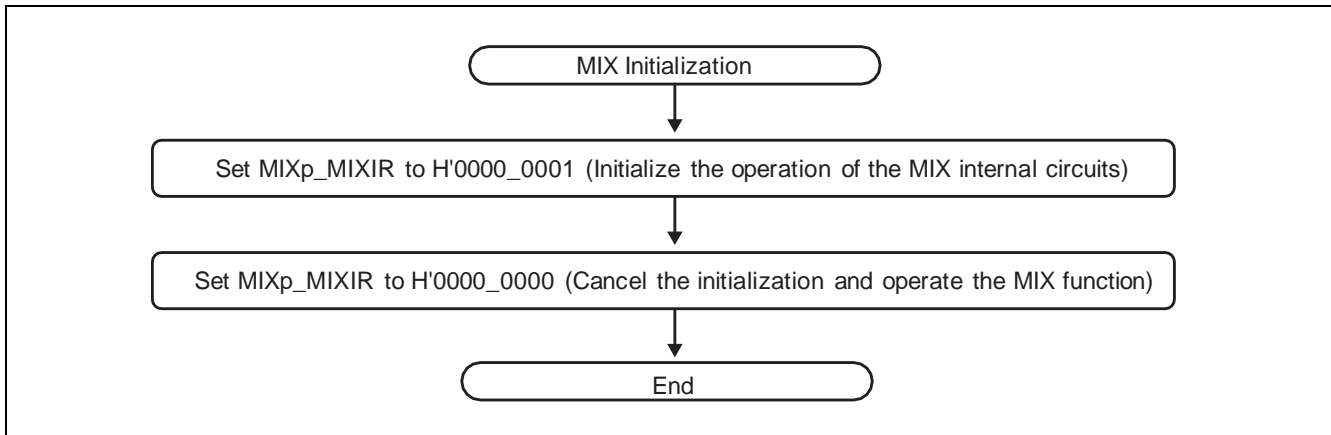


Figure 43.24 MIX Initialization Flowchart

- MIX Halting

Figure 43.25 shows the flowchart of MIX halting. When MIX is halting, it should be initialize or reset by software reset or hardware reset. Before initialize or reset the MIX function, it is necessary to confirm the register settings and operation of peripheral IP.

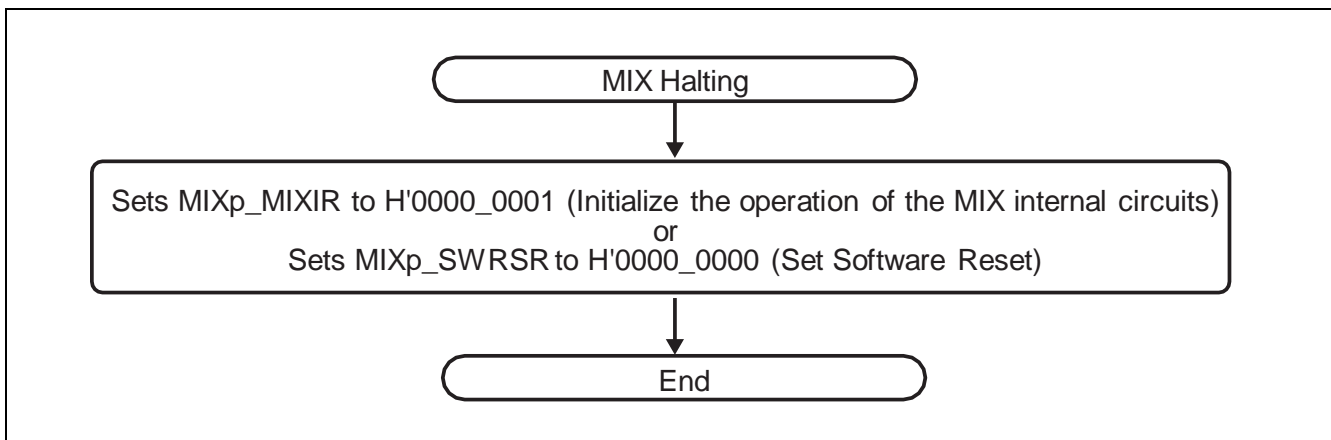


Figure 43.25 MIX Halting Flowchart

(3) DVC Block

The DVC is a block for implementing the volume and mute functions.

- Volume control function including digital volume, volume ramp, and zero-crossing mute
- The digital volume function is specified by a 24-bit fixed-point value within the range from 0 to 8 times (mute or -120 to 18 dB)
- The volume ramp function can be used for soft mute, fade-in, fade-out, or desired volume adjustment
- The volume ramp period can be changed within the sampling range from the 0th to 23rd power of 2
- The zero-crossing mute function silences the sound at the zero-crossing point of the audio data

Table 43.15 shows the DVC functions.

Table 43.15 DVC Functions

| Item | Performance |
|--------------------|--|
| Digital volume | Range: -120 dB to 18 dB (9.5×10^{-7} times to 8 times) |
| Volume ramp | Volume ramp is used for many kinds of operation (soft mute, fade-in, fade-out, and volume adjustment by ramp) Ramp period: $2^0/f_{so}$ to $2^{23}/f_{so}$ Examples: ($1/f_{so}$: -128 dB/1 step) ($2/f_{so}$: -64 dB/1 step) ($4/f_{so}$: -32 dB/1 step) : ($128/f_{so}$: -1 dB/1 step) ($256/f_{so}$: -0.5 dB/1 step) ($512/f_{so}$: -0.25 dB/1 step) ($1024/f_{so}$: -0.125 dB/1 step) ($2048/f_{so}$: -0.125 dB/2 steps) ($4096/f_{so}$: -0.125 dB/4 steps) : ($8388608/f_{so}$: -0.125 dB/8192 steps) |
| Zero-crossing mute | Mute the signal at zero-crossing point |

The settings and corresponding values for the volume ramp are the same as those listed in Table 43.11 to Table 43.14.

The settings and corresponding values for the digital volume are calculated by using the following formula:

$$\text{Setting (decimal)} = 10^X \times 1048576$$

$$X = (\text{value in dB}) / 20$$

(a) Register Setting Procedure

The following describes the DVC register setting procedures. The register should be used according to the following procedures.

- DVC Processing Procedure

Figure 43.26 shows the processing procedure of DVC.

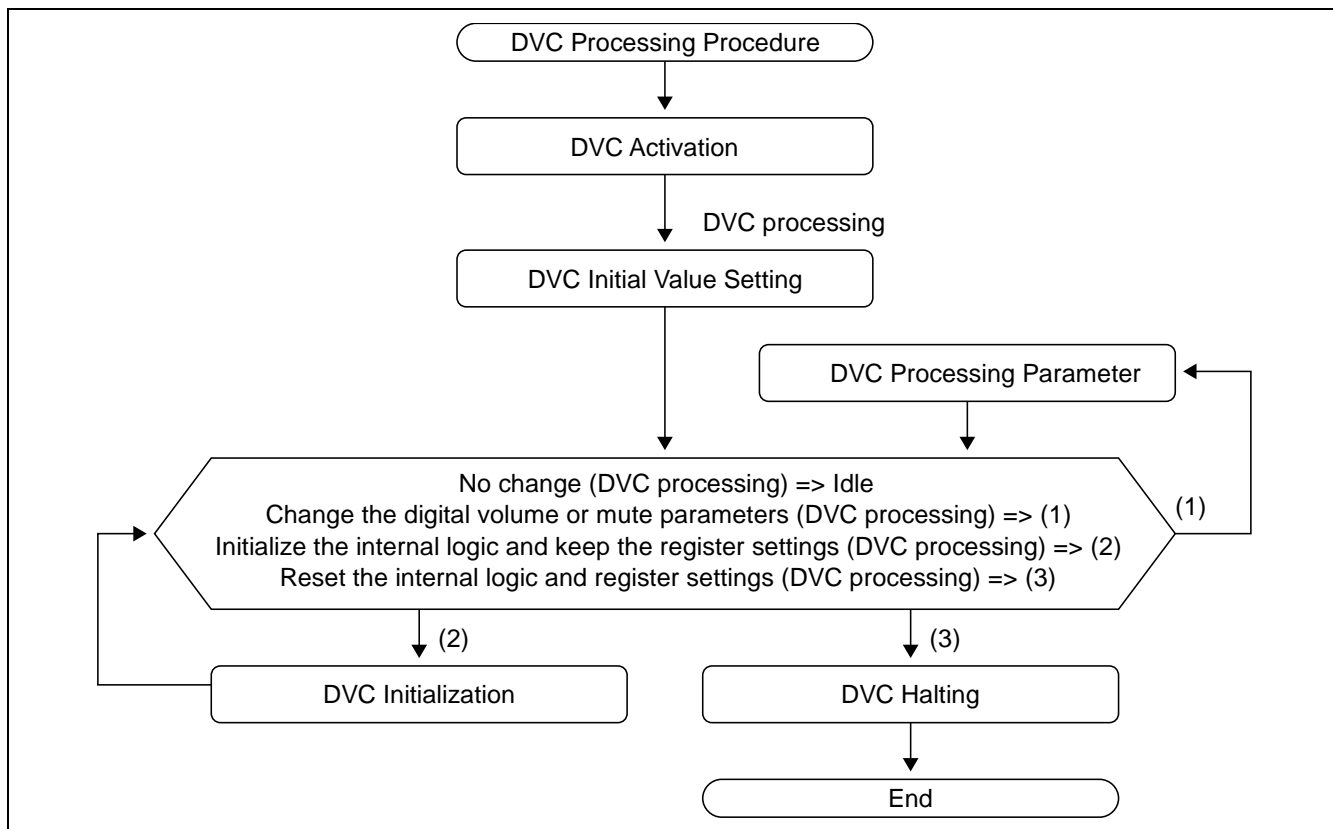


Figure 43.26 DVC Processing Procedure

- DVC Activation

Figure 43.27 shows the DVC activation flowchart. When the DVC is activated, a software reset should be set to initialize logic and register setting. Then, the software reset should be cleared and ready to operate DVC. When reset the DVC function, all registers are initialized. If the DVC function was initialized by a hardware reset, it doesn't need to initialize the DVC function by software reset.

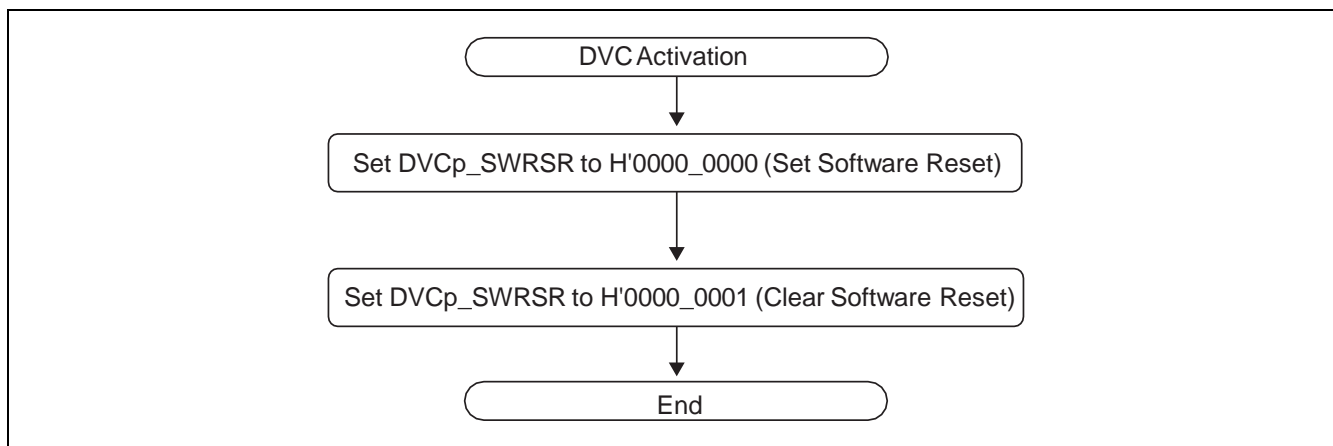


Figure 43.27 DVC Activation Flowchart

- DVC Initial Value Setting

Figure 43.28 shows the flowchart of the DVC initial value setting. Before operate DVC function, the initial values should be set.

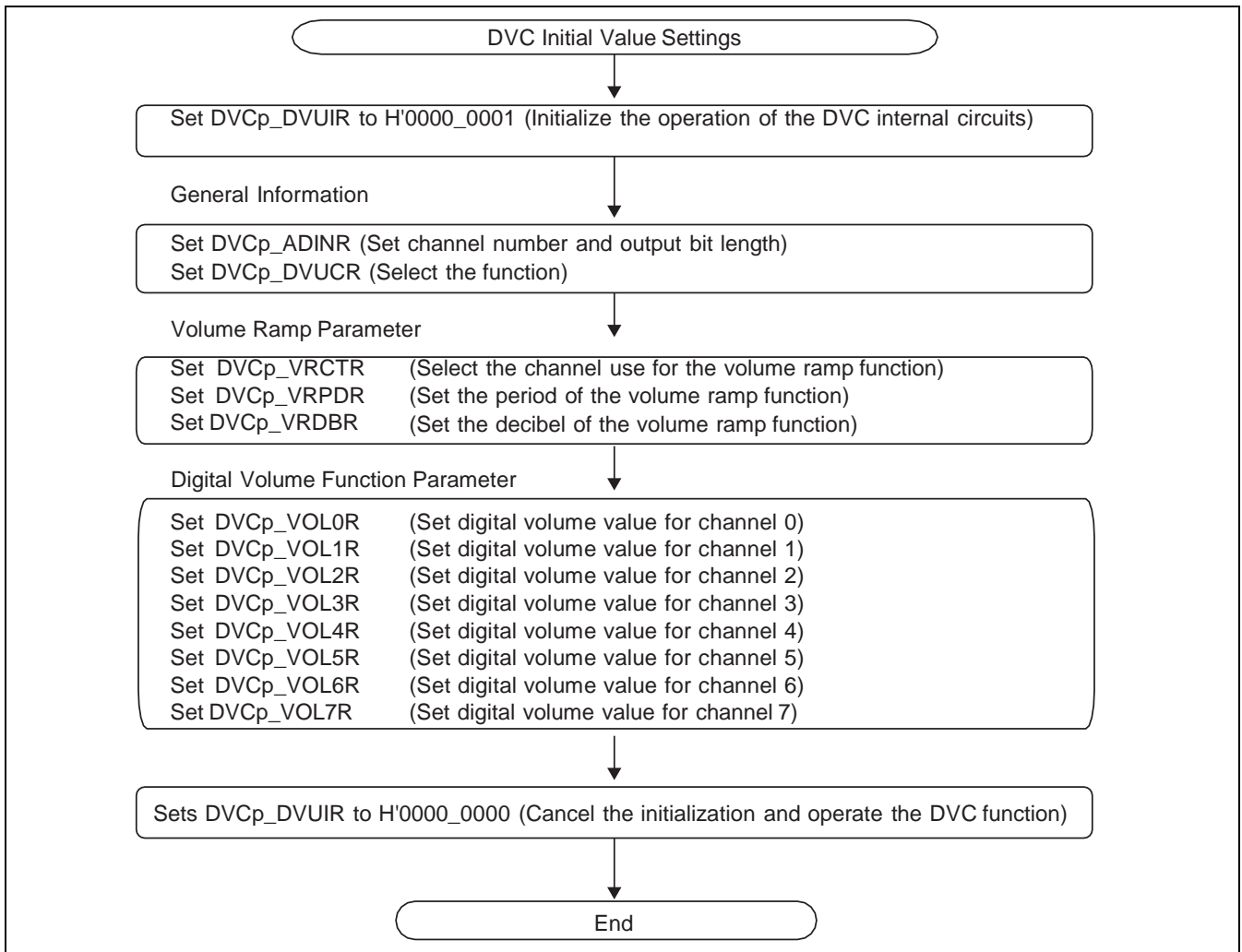


Figure 43.28 DVC Initial Value Setting Flowchart

- DVC Processing Parameter

Figure 43.29 shows the DVC processing parameter flowchart. When DVC is processing, user can control the zero cross mute and the volume ramp and the digital volume value or by changing the setting value of registers. Before change the parameter of each function, the setting of DVCp_DVUER should be disables. After that set the setting of each function and enables the DVCp_DVUER for reflect the register setting to the DVC processing.

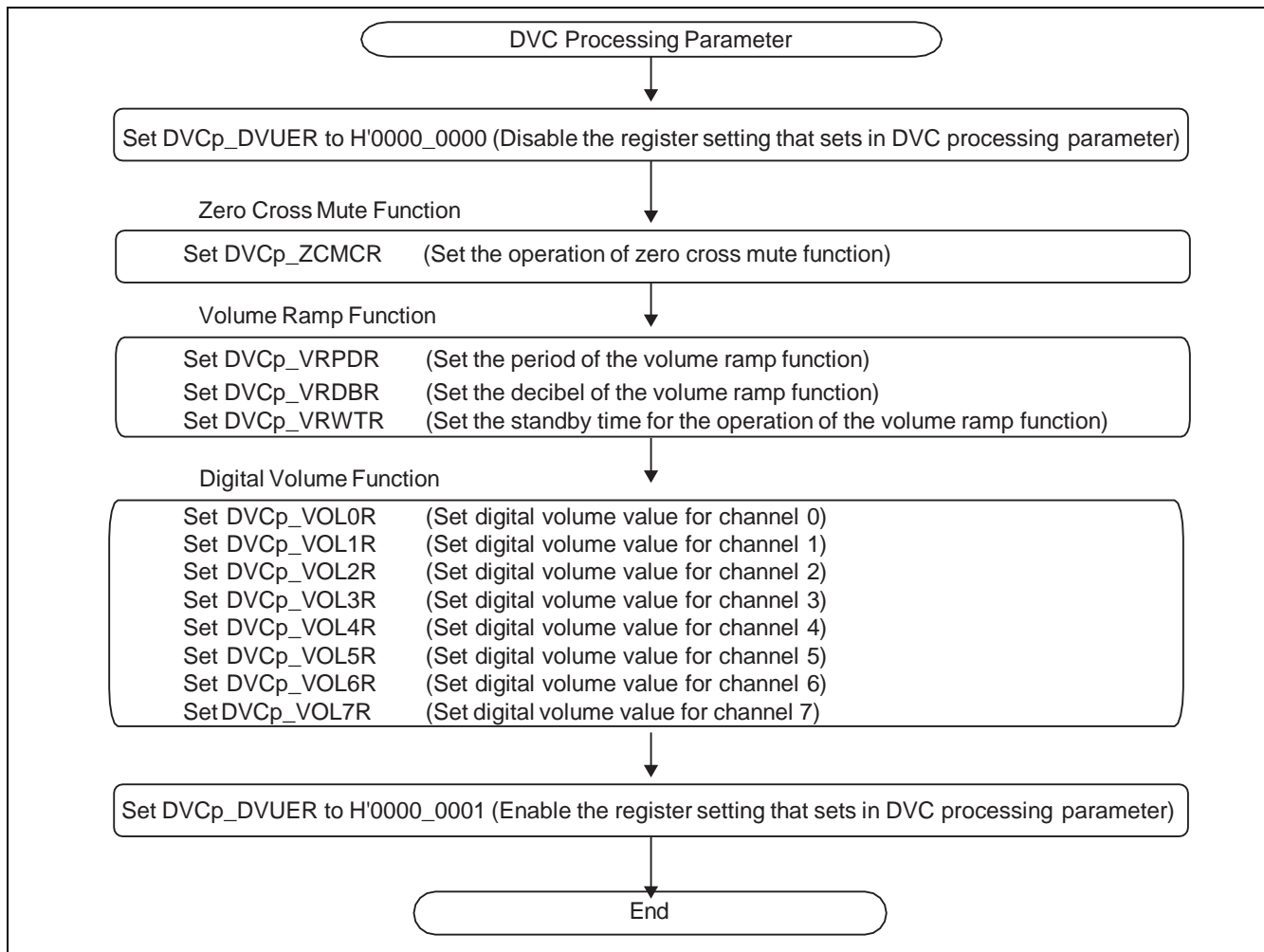


Figure 43.29 DVC Processing Parameter Flowchart

- DVC Initialization

Figure 43.30 shows the flowchart of DVC Initialization. DVCp_DVUIR register doesn't initialize the register settings and these values are maintained. Before cancel the initialization, it is necessary to set or check the register settings of peripheral IP.

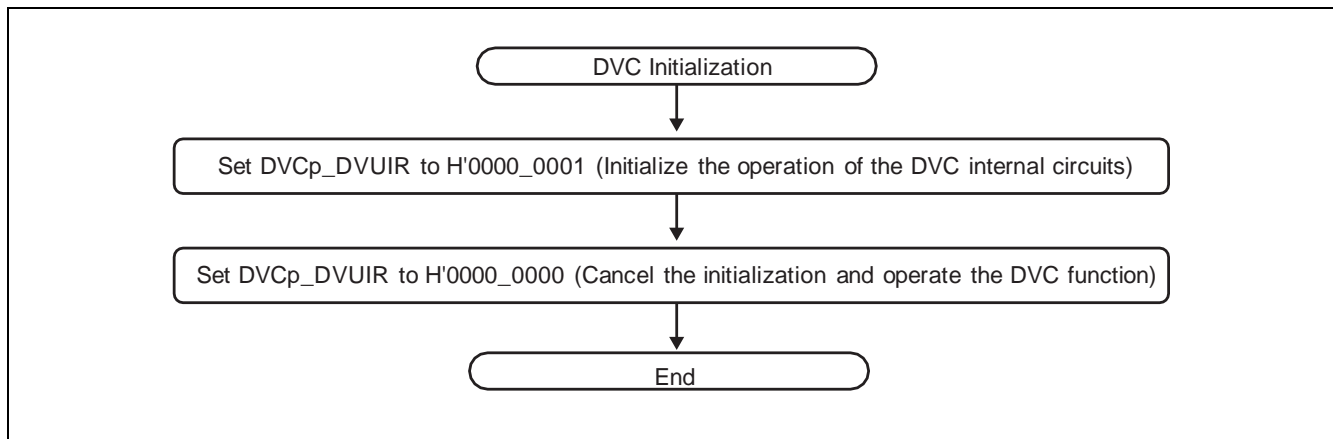


Figure 43.30 DVC Initialization Flowchart

- DVC Halting

Figure 43.31 shows the flowchart of DVC halting. When DVC is halting, it should be initialize or reset by software reset or hardware reset. Before initialize or reset the DVC function, it is necessary to confirm the register settings and operation of peripheral IP.

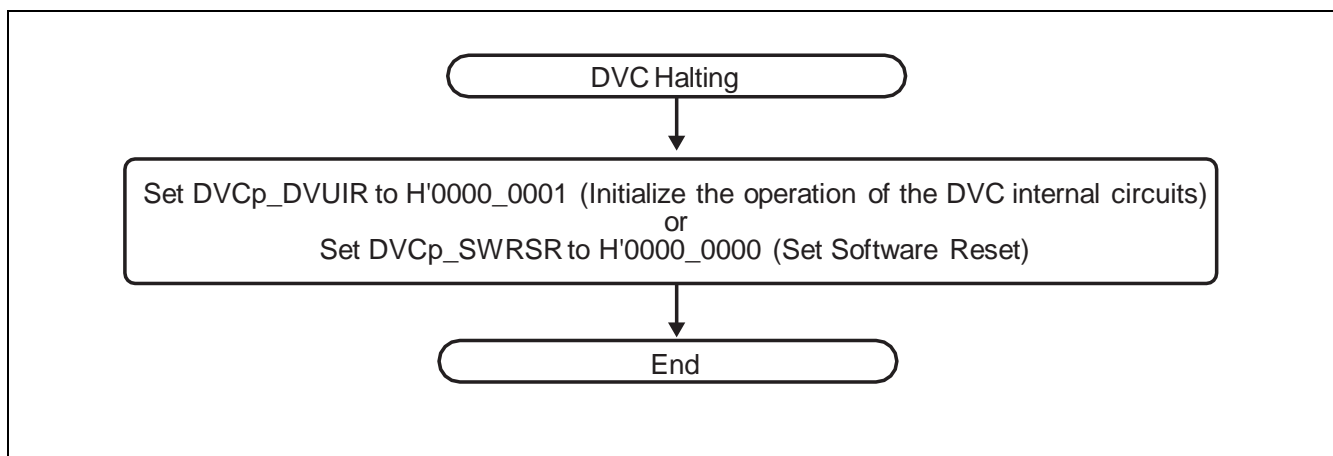


Figure 43.31 DVC Halting Flowchart

43.3.8 Input Data Timing and Output Data Timing

When using the SRC or CMD, the input data timing and output data timing should be specified. For details of the settings, refer to section 42, Audio Clock Generator (ADG).

43.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

43.4.1 Notes on Fixed Value related to SCU registers

The fixed value is needed for these bits in the SCU registers.

It is necessary to change from the initial value to the specific fixed value for the bits of the registers on Table 43.17.

Setting other value is prohibited.

Table 43.16 Fixed Value for the SCU Registers without any condition

| Section | Bit Number | Fixed Value |
|--|------------|--|
| 43.2.23 SRCm SRC Control Register | 16 | B'1 |
| | 12 | B'1 |
| | 8 | B'1 |
| | 4 | B'1 |
| 43.2.21 SRCm IFS Control Register | 0 | B'1 |
| 43.2.22 SRCm IFS Value Setting Register | 27 to 0 | The value of FSI should be calculated into them. A calculate method can be found in that section. |
| 43.2.24 SRCm Buffer Size DATA RAM Setting Register | 26 to 16 | Depends on your needs. * |
| 43.2.25 SRCn Buffer Size DATA RAM Setting Register | 26 to 16 | Depends on your needs. * |
| 43.2.26 SRCm Buffer Size IJEC RAM Setting Register | 21 to 16 | H'10 |
| | 8 to 0 | Depends on your needs. * |

Note: * The fixed value depends on your needs. Refer to the explanation in those section or related table.

Table 43.17 Fixed Value for the SCU Registers with some condition

| Section | Bit Number | Fixed Value | Condition |
|---|------------|-------------|---|
| 43.2.5 SRCm Control Register | 4 | B'0 | A condition is described in a note of that section. Refer to that note. |
| 43.2.21 SRCm IFS Control Register | 23 to 20 | H'2 | Only when fast_ijk_en = 1, as bit 31 |
| | 19 to 16 | H'4 | Only when fast_ijk_en = 1, as bit 31 |
| 43.2.80 DVCp Volume Ramp Control Register | 7 to 0 | H'FF | VRMD bit of DVCp_DVUCR register is 1. |

43.4.2 Note on generation and selection of timing signal

For details on generation and selection of timing signals, see section 42. Audio Clock Generator (ADG).

44. Direct Memory Access Controller for Audio (Audio-DMAC)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

44.1 Overview

This LSI includes a direct memory access controller for audio (audio-DMAC). The audio-DMAC can be used in place of the CPU to handle high-speed data transfer to and from an external memory, the on-chip memory, memory-mapped external devices, or on-chip peripheral modules.

44.1.1 Features

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

- Up to 32 channels are available.

[RZ/G2E]

- Up to 16 channels are available.

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

- 1-T byte physical address space
- Transfer data length: Byte, word (2 bytes), long-word (4 bytes), 8 bytes, 16 bytes, 32 bytes, and 64 bytes
- Maximum number of transfer times: 16,777,215
- Address mode: Dual address mode
- Transfer requests:

Requests from on-chip peripheral modules or auto requests can be selected. The following modules can issue on-chip peripheral module requests.

SSI0 to SSI9, SCU0 to SCU9

- Selectable bus modes:
Normal speed mode or slow mode can be selected for each channel.
- Either fixed priority or round-robin arbitration can be selected for use in arbitration among the transfer channels.
- Interrupt request:

The audio-DMAC can be set up to generate an interrupt request for the CPU upon completion of transfer under the control of one stage of the descriptor memory, at the end of the data transfer, in response to an MMU error, and in response to an address error.

- Descriptor memory function:

Up to 128 sets of the settings for the source address register, destination address register, and transfer count register are available (if use of the descriptor memory is only enabled for one channel) for use in setting up consecutive DMA transfers (the memory can hold register values for up to 256 stages of transfer when the external memory is selected). An infinite repeat mode is also available.

Figure 44.1 shows the block diagram of the audio-DMAC.

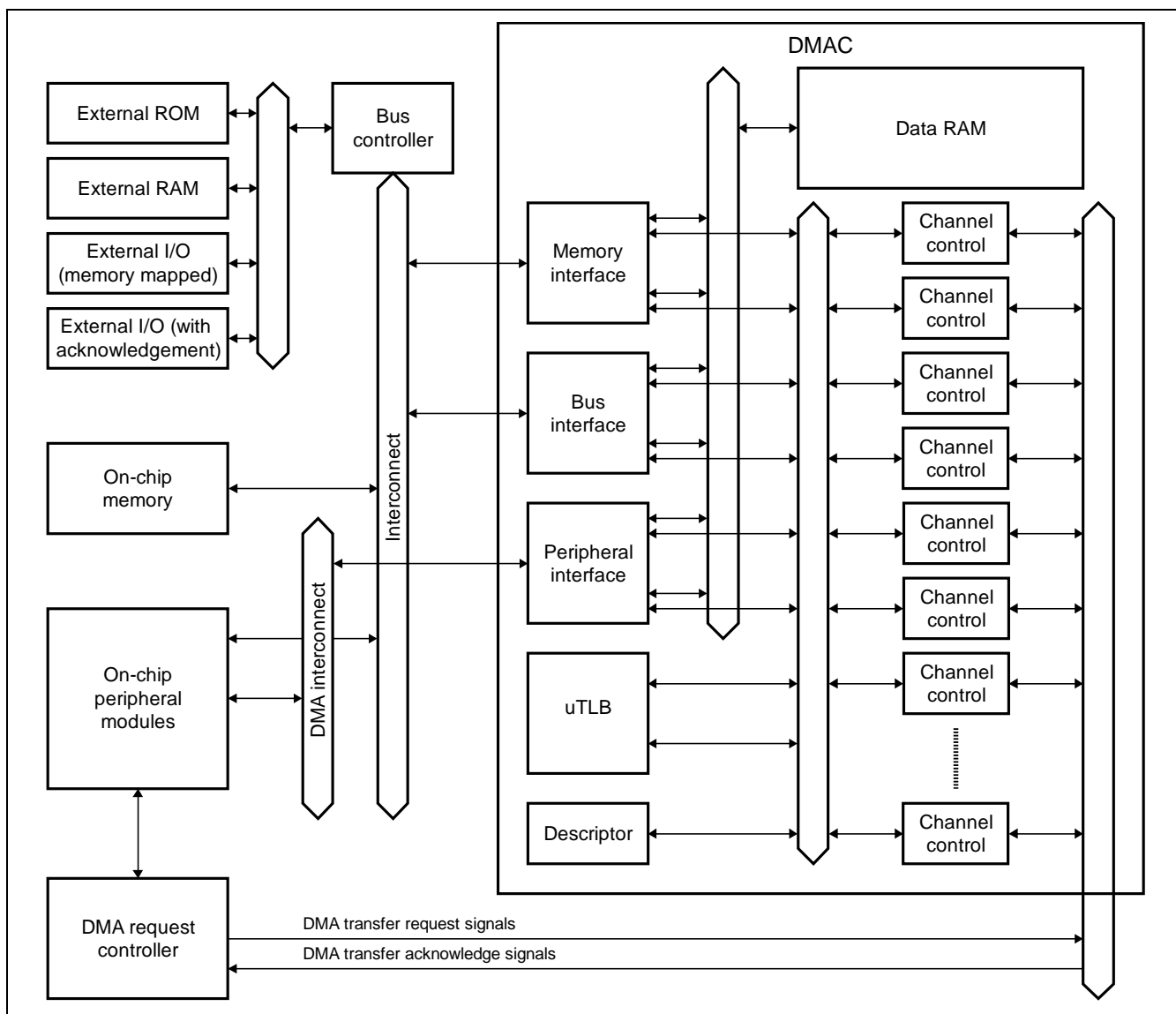


Figure 44.1 Block Diagram of the Audio-DMAC

44.1.2 External Pins

There are no external pins relevant to the audio-DMAC.

44.1.3 Register Configuration

Table 44.1 lists the registers of the audio-DMAC. Table 44.2 shows the register states of the audio-DMAC in each operating mode.

Note: The number of the audio-DMAC channels and the registers in use depend on the product:
 32 channels for the RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 (registers for both the higher- and lower-numbered sets of channels 0 to 15 and 16 to 31 are in use)

Table 44.1 Register Configuration of the Audio-DMAC

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA interrupt status register (for lower-numbered channels) | DMAISTA_L | R | H'EC70_0020 | 32 | √ | √ | √ | √ |
| DMA secure control register (for lower-numbered channels) | DMASEC_L | R/W | H'EC70_0030 | 32 | √ | √ | √ | √ |
| DMA operation register (for lower-numbered channels) | DMAOR_L | R/W | H'EC70_0060 | 16 | √ | √ | √ | √ |
| DMA channel clear register (for lower-numbered channels) | DMACHCLR_L | W | H'EC70_0080 | 32 | √ | √ | √ | √ |
| DPRAM secure control register (for lower-numbered channels) | DMADPSEC_L | R/W | H'EC70_00A0 | 32 | √ | √ | √ | √ |
| DMA source address register_0 | DMASAR_0 | R/W | H'EC70_8000 H'EC70_8020* | 32 | √ | √ | √ | √ |
| DMA destination address register_0 | DMADAR_0 | R/W | H'EC70_8004 H'EC70_8024* | 32 | √ | √ | √ | √ |
| DMA transfer count register_0 | DMATCR_0 | R/W | H'EC70_8008 | 32 | √ | √ | √ | √ |
| DMA transfer size register_0 | DMATSR_0 | R/W | H'EC70_8028* | 32 | √ | √ | √ | √ |
| DMA channel control register_0 | DMACHCR_0 | R/W | H'EC70_800C H'EC70_802C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_0 | DMATCRB_0 | R/W | H'EC70_8018 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_0 | DMATSRB_0 | R/W | H'EC70_8038 | 32 | √ | √ | √ | √ |
| DMA channel control register B_0 | DMACHCRB_0 | R/W | H'EC70_801C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_0 | DMARS_0 | R/W | H'EC70_8040 | 16 | √ | √ | √ | √ |
| DMA buffer control register_0 | DMABUFCR_0 | R/W | H'EC70_8048 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_0 | DMADPBASE_0 | R/W | H'EC70_8050 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_0 | DMADPCR_0 | R/W | H'EC70_8054 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_0 | DMAFIXSAR_0 | R/W | H'EC70_8010 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_0 | DMAFIXDAR_0 | R/W | H'EC70_8014 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_0 | DMAFIXDPBASE_0 | R/W | H'EC70_8060 | 32 | √ | √ | √ | √ |
| DMA source address register_1 | DMASAR_1 | R/W | H'EC70_8080 H'EC70_80A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_1 | DMADAR_1 | R/W | H'EC70_8084 H'EC70_80A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_1 | DMATCR_1 | R/W | H'EC70_8088 | 32 | √ | √ | √ | √ |
| DMA transfer size register_1 | DMATSR_1 | R/W | H'EC70_80A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_1 | DMACHCR_1 | R/W | H'EC70_808C H'EC70_80AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_1 | DMATCRB_1 | R/W | H'EC70_8098 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_1 | DMATSRB_1 | R/W | H'EC70_80B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_1 | DMACHCRB_1 | R/W | H'EC70_809C | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | | |
|--|----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|--|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| DMA extended resource selector_1 | DMARS_1 | R/W | H'EC70_80C0 | 16 | √ | √ | √ | √ | |
| DMA buffer control register_1 | DMABUFCR_1 | R/W | H'EC70_80C8 | 32 | √ | √ | √ | √ | |
| DMA descriptor base address register_1 | DMADPBASE_1 | R/W | H'EC70_80D0 | 32 | √ | √ | √ | √ | |
| DMA descriptor control register_1 | DMADPCR_1 | R/W | H'EC70_80D4 | 32 | √ | √ | √ | √ | |
| DMA fixed descriptor base address register_1 | DMAFIXDPBASE_1 | R/W | H'EC70_80E0 | 32 | √ | √ | √ | √ | |
| DMA fixed source address register_1 | DMAFIXSAR_1 | R/W | H'EC70_8090 | 32 | √ | √ | √ | √ | |
| DMA fixed destination address register_1 | DMAFIXDAR_1 | R/W | H'EC70_8094 | 32 | √ | √ | √ | √ | |
| DMA source address register_2 | DMASAR_2 | R/W | H'EC70_8100 H'EC70_8120* | 32 | √ | √ | √ | √ | |
| DMA destination address register_2 | DMADAR_2 | R/W | H'EC70_8104 H'EC70_8124* | 32 | √ | √ | √ | √ | |
| DMA transfer count register_2 | DMATCR_2 | R/W | H'EC70_8108 | 32 | √ | √ | √ | √ | |
| DMA transfer size register_2 | DMATSR_2 | R/W | H'EC70_8128* | 32 | √ | √ | √ | √ | |
| DMA channel control register_2 | DMACHCR_2 | R/W | H'EC70_810C H'EC70_812C* | 32 | √ | √ | √ | √ | |
| DMA transfer count register B_2 | DMATCRB_2 | R/W | H'EC70_8118 | 32 | √ | √ | √ | √ | |
| DMA transfer size register B_2 | DMATSRB_2 | R/W | H'EC70_8138* | 32 | √ | √ | √ | √ | |
| DMA channel control register B_2 | DMACHCRB_2 | R/W | H'EC70_811C | 32 | √ | √ | √ | √ | |
| DMA extended resource selector_2 | DMARS_2 | R/W | H'EC70_8140 | 16 | √ | √ | √ | √ | |
| DMA buffer control register_2 | DMABUFCR_2 | R/W | H'EC70_8148 | 32 | √ | √ | √ | √ | |
| DMA descriptor base address register_2 | DMADPBASE_2 | R/W | H'EC70_8150 | 32 | √ | √ | √ | √ | |
| DMA descriptor control register_2 | DMADPCR_2 | R/W | H'EC70_8154 | 32 | √ | √ | √ | √ | |
| DMA fixed source address register_2 | DMAFIXSAR_2 | R/W | H'EC70_8110 | 32 | √ | √ | √ | √ | |
| DMA fixed destination address register_2 | DMAFIXDAR_2 | R/W | H'EC70_8114 | 32 | √ | √ | √ | √ | |
| DMA fixed descriptor base address register_2 | DMAFIXDPBASE_2 | R/W | H'EC70_8160 | 32 | √ | √ | √ | √ | |
| DMA source address register_3 | DMASAR_3 | R/W | H'EC70_8180 H'EC70_81A0* | 32 | √ | √ | √ | √ | |
| DMA destination address register_3 | DMADAR_3 | R/W | H'EC70_8184 H'EC70_81A4* | 32 | √ | √ | √ | √ | |
| DMA transfer count register_3 | DMATCR_3 | R/W | H'EC70_8188 | 32 | √ | √ | √ | √ | |
| DMA transfer size register_3 | DMATSR_3 | R/W | H'EC70_81A8* | 32 | √ | √ | √ | √ | |
| DMA channel control register_3 | DMACHCR_3 | R/W | H'EC70_818C H'EC70_81AC* | 32 | √ | √ | √ | √ | |
| DMA transfer count register B_3 | DMATCRB_3 | R/W | H'EC70_8198 | 32 | √ | √ | √ | √ | |
| DMA transfer size register B_3 | DMATSRB_3 | R/W | H'EC70_81B8* | 32 | √ | √ | √ | √ | |
| DMA channel control register B_3 | DMACHCRB_3 | R/W | H'EC70_819C | 32 | √ | √ | √ | √ | |
| DMA extended resource selector_3 | DMARS_3 | R/W | H'EC70_81C0 | 16 | √ | √ | √ | √ | |
| DMA buffer control register_3 | DMABUFCR_3 | R/W | H'EC70_81C8 | 32 | √ | √ | √ | √ | |
| DMA descriptor base address register_3 | DMADPBASE_3 | R/W | H'EC70_81D0 | 32 | √ | √ | √ | √ | |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--|----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA descriptor control register_3 | DMADPCR_3 | R/W | H'EC70_81D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_3 | DMAFIXSAR_3 | R/W | H'EC70_8190 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_3 | DMAFIXDAR_3 | R/W | H'EC70_8194 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_3 | DMAFIXDPBASE_3 | R/W | H'EC70_81E0 | 32 | √ | √ | √ | √ |
| DMA source address register_4 | DMASAR_4 | R/W | H'EC70_8200 H'EC70_8220* | 32 | √ | √ | √ | √ |
| DMA destination address register_4 | DMADAR_4 | R/W | H'EC70_8204 H'EC70_8224* | 32 | √ | √ | √ | √ |
| DMA transfer count register_4 | DMATCR_4 | R/W | H'EC70_8208 | 32 | √ | √ | √ | √ |
| DMA transfer size register_4 | DMATSR_4 | R/W | H'EC70_8228* | 32 | √ | √ | √ | √ |
| DMA channel control register_4 | DMACHCR_4 | R/W | H'EC70_820C H'EC70_822C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_4 | DMATCRB_4 | R/W | H'EC70_8218 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_4 | DMATSRB_4 | R/W | H'EC70_8238* | 32 | √ | √ | √ | √ |
| DMA channel control register B_4 | DMACHCRB_4 | R/W | H'EC70_821C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_4 | DMARS_4 | R/W | H'EC70_8240 | 16 | √ | √ | √ | √ |
| DMA buffer control register_4 | DMABUFCR_4 | R/W | H'EC70_8248 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_4 | DMADPBASE_4 | R/W | H'EC70_8250 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_4 | DMADPCR_4 | R/W | H'EC70_8254 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_4 | DMAFIXSAR_4 | R/W | H'EC70_8210 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_4 | DMAFIXDAR_4 | R/W | H'EC70_8214 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_4 | DMAFIXDPBASE_4 | R/W | H'EC70_8260 | 32 | √ | √ | √ | √ |
| DMA source address register_5 | DMASAR_5 | R/W | H'EC70_8280 H'EC70_82A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_5 | DMADAR_5 | R/W | H'EC70_8284 H'EC70_82A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_5 | DMATCR_5 | R/W | H'EC70_8288 | 32 | √ | √ | √ | √ |
| DMA transfer size register_5 | DMATSR_5 | R/W | H'EC70_82A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_5 | DMACHCR_5 | R/W | H'EC70_828C H'EC70_82AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_5 | DMATCRB_5 | R/W | H'EC70_8298 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_5 | DMATSRB_5 | R/W | H'EC70_82B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_5 | DMACHCRB_5 | R/W | H'EC70_829C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_5 | DMARS_5 | R/W | H'EC70_82C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_5 | DMABUFCR_5 | R/W | H'EC70_82C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_5 | DMADPBASE_5 | R/W | H'EC70_82D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_5 | DMADPCR_5 | R/W | H'EC70_82D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_5 | DMAFIXSAR_5 | R/W | H'EC70_8290 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_5 | DMAFIXDAR_5 | R/W | H'EC70_8294 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--|----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA fixed descriptor base address register_5 | DMAFIXDPBASE_5 | R/W | H'EC70_82E0 | 32 | √ | √ | √ | √ |
| DMA source address register_6 | DMASAR_6 | R/W | H'EC70_8300 H'EC70_8320* | 32 | √ | √ | √ | √ |
| DMA destination address register_6 | DMADAR_6 | R/W | H'EC70_8304 H'EC70_8324* | 32 | √ | √ | √ | √ |
| DMA transfer count register_6 | DMATCR_6 | R/W | H'EC70_8308 | 32 | √ | √ | √ | √ |
| DMA transfer size register_6 | DMATSR_6 | R/W | H'EC70_8328* | 32 | √ | √ | √ | √ |
| DMA channel control register_6 | DMACHCR_6 | R/W | H'EC70_830C H'EC70_832C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_6 | DMATCRB_6 | R/W | H'EC70_8318 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_6 | DMATSRB_6 | R/W | H'EC70_8338* | 32 | √ | √ | √ | √ |
| DMA channel control register B_6 | DMACHCRB_6 | R/W | H'EC70_831C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_6 | DMARS_6 | R/W | H'EC70_8340 | 16 | √ | √ | √ | √ |
| DMA buffer control register_6 | DMABUFCR_6 | R/W | H'EC70_8348 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_6 | DMADPBASE_6 | R/W | H'EC70_8350 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_6 | DMADPCR_6 | R/W | H'EC70_8354 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_6 | DMAFIXSAR_6 | R/W | H'EC70_8310 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_6 | DMAFIXDAR_6 | R/W | H'EC70_8314 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_6 | DMAFIXDPBASE_6 | R/W | H'EC70_8360 | 32 | √ | √ | √ | √ |
| DMA source address register_7 | DMASAR_7 | R/W | H'EC70_8380 H'EC70_83A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_7 | DMADAR_7 | R/W | H'EC70_8384 H'EC70_83A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_7 | DMATCR_7 | R/W | H'EC70_8388 | 32 | √ | √ | √ | √ |
| DMA transfer size register_7 | DMATSR_7 | R/W | H'EC70_83A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_7 | DMACHCR_7 | R/W | H'EC70_838C H'EC70_83AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_7 | DMATCRB_7 | R/W | H'EC70_8398 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_7 | DMATSRB_7 | R/W | H'EC70_83B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_7 | DMACHCRB_7 | R/W | H'EC70_839C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_7 | DMARS_7 | R/W | H'EC70_83C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_7 | DMABUFCR_7 | R/W | H'EC70_83C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_7 | DMADPBASE_7 | R/W | H'EC70_83D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_7 | DMADPCR_7 | R/W | H'EC70_83D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_7 | DMAFIXSAR_7 | R/W | H'EC70_8390 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_7 | DMAFIXDAR_7 | R/W | H'EC70_8394 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_7 | DMAFIXDPBASE_7 | R/W | H'EC70_83E0 | 32 | √ | √ | √ | √ |
| DMA source address register_8 | DMASAR_8 | R/W | H'EC70_8400 H'EC70_8420* | 32 | √ | √ | √ | √ |
| DMA destination address register_8 | DMADAR_8 | R/W | H'EC70_8404 H'EC70_8424* | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--|----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA transfer count register_8 | DMATCR_8 | R/W | H'EC70_8408 | 32 | √ | √ | √ | √ |
| DMA transfer size register_8 | DMATSR_8 | R/W | H'EC70_8428* | 32 | √ | √ | √ | √ |
| DMA channel control register_8 | DMACHCR_8 | R/W | H'EC70_840C H'EC70_842C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_8 | DMATCRB_8 | R/W | H'EC70_8418 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_8 | DMATSRB_8 | R/W | H'EC70_8438* | 32 | √ | √ | √ | √ |
| DMA channel control register B_8 | DMACHCRB_8 | R/W | H'EC70_841C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_8 | DMARS_8 | R/W | H'EC70_8440 | 16 | √ | √ | √ | √ |
| DMA buffer control register_8 | DMABUFCR_8 | R/W | H'EC70_8448 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_8 | DMADPBASE_8 | R/W | H'EC70_8450 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_8 | DMADPCR_8 | R/W | H'EC70_8454 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_8 | DMAFIXSAR_8 | R/W | H'EC70_8410 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_8 | DMAFIXDAR_8 | R/W | H'EC70_8414 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_8 | DMAFIXDPBASE_8 | R/W | H'EC70_8460 | 32 | √ | √ | √ | √ |
| DMA source address register_9 | DMASAR_9 | R/W | H'EC70_8480 H'EC70_84A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_9 | DMADAR_9 | R/W | H'EC70_8484 H'EC70_84A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_9 | DMATCR_9 | R/W | H'EC70_8488 | 32 | √ | √ | √ | √ |
| DMA transfer size register_9 | DMATSR_9 | R/W | H'EC70_84A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_9 | DMACHCR_9 | R/W | H'EC70_848C H'EC70_84AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_9 | DMATCRB_9 | R/W | H'EC70_8498 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_9 | DMATSRB_9 | R/W | H'EC70_84B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_9 | DMACHCRB_9 | R/W | H'EC70_849C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_9 | DMARS_9 | R/W | H'EC70_84C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_9 | DMABUFCR_9 | R/W | H'EC70_84C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_9 | DMADPBASE_9 | R/W | H'EC70_84D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_9 | DMADPCR_9 | R/W | H'EC70_84D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_9 | DMAFIXSAR_9 | R/W | H'EC70_8490 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_9 | DMAFIXDAR_9 | R/W | H'EC70_8494 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_9 | DMAFIXDPBASE_9 | R/W | H'EC70_84E0 | 32 | √ | √ | √ | √ |
| DMA source address register_10 | DMASAR_10 | R/W | H'EC70_8500 H'EC70_8520* | 32 | √ | √ | √ | √ |
| DMA destination address register_10 | DMADAR_10 | R/W | H'EC70_8504 H'EC70_8524* | 32 | √ | √ | √ | √ |
| DMA transfer count register_10 | DMATCR_10 | R/W | H'EC70_8508 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA transfer size register_10 | DMATSR_10 | R/W | H'EC70_8528* | 32 | √ | √ | √ | √ |
| DMA channel control register_10 | DMACHCR_10 | R/W | H'EC70_850C H'EC70_852C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_10 | DMATCRB_10 | R/W | H'EC70_8518 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_10 | DMATSRB_10 | R/W | H'EC70_8538* | 32 | √ | √ | √ | √ |
| DMA channel control register B_10 | DMACHCRB_10 | R/W | H'EC70_851C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_10 | DMARS_10 | R/W | H'EC70_8540 | 16 | √ | √ | √ | √ |
| DMA buffer control register_10 | DMABUFCR_10 | R/W | H'EC70_8548 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_10 | DMADPBASE_10 | R/W | H'EC70_8550 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_10 | DMADPCR_10 | R/W | H'EC70_8554 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_10 | DMAFIXSAR_10 | R/W | H'EC70_8510 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_10 | DMAFIXDAR_10 | R/W | H'EC70_8514 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_10 | DMAFIXDPBASE_10 | R/W | H'EC70_8560 | 32 | √ | √ | √ | √ |
| DMA source address register_11 | DMASAR_11 | R/W | H'EC70_8580 H'EC70_85A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_11 | DMADAR_11 | R/W | H'EC70_8584 H'EC70_85A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_11 | DMATCR_11 | R/W | H'EC70_8588 | 32 | √ | √ | √ | √ |
| DMA transfer size register_11 | DMATSR_11 | R/W | H'EC70_85A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_11 | DMACHCR_11 | R/W | H'EC70_858C H'EC70_85AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_11 | DMATCRB_11 | R/W | H'EC70_8598 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_11 | DMATSRB_11 | R/W | H'EC70_85B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_11 | DMACHCRB_11 | R/W | H'EC70_859C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_11 | DMARS_11 | R/W | H'EC70_85C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_11 | DMABUFCR_11 | R/W | H'EC70_85C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_11 | DMADPBASE_11 | R/W | H'EC70_85D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_11 | DMADPCR_11 | R/W | H'EC70_85D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_11 | DMAFIXSAR_11 | R/W | H'EC70_8590 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_11 | DMAFIXDAR_11 | R/W | H'EC70_8594 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_11 | DMAFIXDPBASE_11 | R/W | H'EC70_85E0 | 32 | √ | √ | √ | √ |
| DMA source address register_12 | DMASAR_12 | R/W | H'EC70_8600 H'EC70_8620* | 32 | √ | √ | √ | √ |
| DMA destination address register_12 | DMADAR_12 | R/W | H'EC70_8604 H'EC70_8624* | 32 | √ | √ | √ | √ |
| DMA transfer count register_12 | DMATCR_12 | R/W | H'EC70_8608 | 32 | √ | √ | √ | √ |
| DMA transfer size register_12 | DMATSR_12 | R/W | H'EC70_8628* | 32 | √ | √ | √ | √ |
| DMA channel control register_12 | DMACHCR_12 | R/W | H'EC70_860C H'EC70_862C* | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA transfer count register B_12 | DMATCRB_12 | R/W | H'EC70_8618 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_12 | DMATSRB_12 | R/W | H'EC70_8638* | 32 | √ | √ | √ | √ |
| DMA channel control register B_12 | DMACHCRB_12 | R/W | H'EC70_861C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_12 | DMARS_12 | R/W | H'EC70_8640 | 16 | √ | √ | √ | √ |
| DMA buffer control register_12 | DMABUFCR_12 | R/W | H'EC70_8648 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_12 | DMADPBASE_12 | R/W | H'EC70_8650 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_12 | DMADPCR_12 | R/W | H'EC70_8654 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_12 | DMAFIXSAR_12 | R/W | H'EC70_8610 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_12 | DMAFIXDAR_12 | R/W | H'EC70_8614 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_12 | DMAFIXDPBASE_12 | R/W | H'EC70_8660 | 32 | √ | √ | √ | √ |
| DMA source address register_13 | DMASAR_13 | R/W | H'EC70_8680 H'EC70_86A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_13 | DMADAR_13 | R/W | H'EC70_8684 H'EC70_86A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_13 | DMATCR_13 | R/W | H'EC70_8688 | 32 | √ | √ | √ | √ |
| DMA transfer size register_13 | DMATSR_13 | R/W | H'EC70_86A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_13 | DMACHCR_13 | R/W | H'EC70_868C H'EC70_86AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_13 | DMATCRB_13 | R/W | H'EC70_8698 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_13 | DMATSRB_13 | R/W | H'EC70_86B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_13 | DMACHCRB_13 | R/W | H'EC70_869C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_13 | DMARS_13 | R/W | H'EC70_86C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_13 | DMABUFCR_13 | R/W | H'EC70_86C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_13 | DMADPBASE_13 | R/W | H'EC70_86D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_13 | DMADPCR_13 | R/W | H'EC70_86D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_13 | DMAFIXSAR_13 | R/W | H'EC70_8690 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_13 | DMAFIXDAR_13 | R/W | H'EC70_8694 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_13 | DMAFIXDPBASE_13 | R/W | H'EC70_86E0 | 32 | √ | √ | √ | √ |
| DMA source address register_14 | DMASAR_14 | R/W | H'EC70_8700 H'EC70_8720* | 32 | √ | √ | √ | √ |
| DMA destination address register_14 | DMADAR_14 | R/W | H'EC70_8704 H'EC70_8724* | 32 | √ | √ | √ | √ |
| DMA transfer count register_14 | DMATCR_14 | R/W | H'EC70_8708 | 32 | √ | √ | √ | √ |
| DMA transfer size register_14 | DMATSR_14 | R/W | H'EC70_8728* | 32 | √ | √ | √ | √ |
| DMA channel control register_14 | DMACHCR_14 | R/W | H'EC70_870C H'EC70_872C* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_14 | DMATCRB_14 | R/W | H'EC70_8718 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_14 | DMATSRB_14 | R/W | H'EC70_8738* | 32 | √ | √ | √ | √ |
| DMA channel control register B_14 | DMACHCRB_14 | R/W | H'EC70_871C | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-------------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA extended resource selector_14 | DMARS_14 | R/W | H'EC70_8740 | 16 | √ | √ | √ | √ |
| DMA buffer control register_14 | DMABUFCR_14 | R/W | H'EC70_8748 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_14 | DMADPBASE_14 | R/W | H'EC70_8750 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_14 | DMADPCR_14 | R/W | H'EC70_8754 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_14 | DMAFIXSAR_14 | R/W | H'EC70_8710 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_14 | DMAFIXDAR_14 | R/W | H'EC70_8714 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_14 | DMAFIXDPBASE_14 | R/W | H'EC70_8760 | 32 | √ | √ | √ | √ |
| DMA source address register_15 | DMASAR_15 | R/W | H'EC70_8780 H'EC70_87A0* | 32 | √ | √ | √ | √ |
| DMA destination address register_15 | DMADAR_15 | R/W | H'EC70_8784 H'EC70_87A4* | 32 | √ | √ | √ | √ |
| DMA transfer count register_15 | DMATCR_15 | R/W | H'EC70_8788 | 32 | √ | √ | √ | √ |
| DMA transfer size register_15 | DMATSR_15 | R/W | H'EC70_87A8* | 32 | √ | √ | √ | √ |
| DMA channel control register_15 | DMACHCR_15 | R/W | H'EC70_878C H'EC70_87AC* | 32 | √ | √ | √ | √ |
| DMA transfer count register B_15 | DMATCRB_15 | R/W | H'EC70_8798 | 32 | √ | √ | √ | √ |
| DMA transfer size register B_15 | DMATSRB_15 | R/W | H'EC70_87B8* | 32 | √ | √ | √ | √ |
| DMA channel control register B_15 | DMACHCRB_15 | R/W | H'EC70_879C | 32 | √ | √ | √ | √ |
| DMA extended resource selector_15 | DMARS_15 | R/W | H'EC70_87C0 | 16 | √ | √ | √ | √ |
| DMA buffer control register_15 | DMABUFCR_15 | R/W | H'EC70_87C8 | 32 | √ | √ | √ | √ |
| DMA descriptor base address register_15 | DMADPBASE_15 | R/W | H'EC70_87D0 | 32 | √ | √ | √ | √ |
| DMA descriptor control register_15 | DMADPCR_15 | R/W | H'EC70_87D4 | 32 | √ | √ | √ | √ |
| DMA fixed source address register_15 | DMAFIXSAR_15 | R/W | H'EC70_8790 | 32 | √ | √ | √ | √ |
| DMA fixed destination address register_15 | DMAFIXDAR_15 | R/W | H'EC70_8794 | 32 | √ | √ | √ | √ |
| DMA fixed descriptor base address register_15 | DMAFIXDPBASE_15 | R/W | H'EC70_87E0 | 32 | √ | √ | √ | √ |
| Descriptor memory (for lower-numbered channels) | DescriptorMEM | R/W | H'EC70_A000 to H'EC70_A7FC | 32 | √ | √ | √ | √ |
| Secure function Secure Status register (for channels 0 to 15) | DMASES_L | R/W | H'EC70_00C0 | 32 | √ | √ | √ | √ |
| Secure function Slave Error Address register (for channels 0 to 15) | DMASEDDR_L | R | H'EC70_00C4 | 32 | √ | √ | √ | √ |
| Secure function Error Master ID register (for channels 0 to 15) | DMASEMID_L | R | H'EC70_00C8 | 32 | √ | √ | √ | √ |
| DMA interrupt status register (for higher-numbered channels) | DMAISTA_U | R | H'EC72_0020 | 32 | √ | √ | √ | √ |
| DMA secure control register (for higher-numbered channels) | DMASEC_H | R/W | H'EC72_0030 | 32 | √ | √ | √ | √ |
| DMA operation register (for higher-numbered channels) | DMAOR_U | R/W | H'EC72_0060 | 16 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|--|-----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA channel clear register (for higher-numbered channels) | DMACHCLR_U | W | H'EC72_0080 | 32 | √ | √ | √ | √ |
| DPRAM secure control register (for higher-numbered channels) | DMADPSEC_U | R/W | H'EC72_00A0 | 32 | √ | √ | √ | √ |
| DMA source address register_16 | DMASAR_16 | R/W | H'EC72_8000 H'EC72_8020* | 32 | √ | √ | √ | — |
| DMA destination address register_16 | DMADAR_16 | R/W | H'EC72_8004 H'EC72_8024* | 32 | √ | √ | √ | — |
| DMA transfer count register_16 | DMATCR_16 | R/W | H'EC72_8008 | 32 | √ | √ | √ | — |
| DMA transfer size register_16 | DMATSR_16 | R/W | H'EC72_8028* | 32 | √ | √ | √ | — |
| DMA channel control register_16 | DMACHCR_16 | R/W | H'EC72_800C H'EC72_802C* | 32 | √ | √ | √ | — |
| DMA transfer count register B_16 | DMATCRB_16 | R/W | H'EC72_8018 | 32 | √ | √ | √ | — |
| DMA transfer size register B_16 | DMATSRB_16 | R/W | H'EC72_8038 | 32 | √ | √ | √ | — |
| DMA channel control register B_16 | DMACHCRB_16 | R/W | H'EC72_801C | 32 | √ | √ | √ | — |
| DMA extended resource selector_16 | DMARS_16 | R/W | H'EC72_8040 | 16 | √ | √ | √ | — |
| DMA buffer control register_16 | DMABUFCR_16 | R/W | H'EC72_8048 | 32 | √ | √ | √ | — |
| DMA descriptor base address register_16 | DMADPBASE_16 | R/W | H'EC72_8050 | 32 | √ | √ | √ | — |
| DMA descriptor control register_16 | DMADPCR_16 | R/W | H'EC72_8054 | 32 | √ | √ | √ | — |
| DMA fixed source address register_16 | DMAFIXSAR_16 | R/W | H'EC72_8010 | 32 | √ | √ | √ | — |
| DMA fixed destination address register_16 | DMAFIXDAR_16 | R/W | H'EC72_8014 | 32 | √ | √ | √ | — |
| DMA fixed descriptor base address register_16 | DMAFIXDPBASE_16 | R/W | H'EC72_8060 | 32 | √ | √ | √ | — |
| DMA source address register_17 | DMASAR_17 | R/W | H'EC72_8080 H'EC72_80A0* | 32 | √ | √ | √ | — |
| DMA destination address register_17 | DMADAR_17 | R/W | H'EC72_8084 H'EC72_80A4* | 32 | √ | √ | √ | — |
| DMA transfer count register_17 | DMATCR_17 | R/W | H'EC72_8088 | 32 | √ | √ | √ | — |
| DMA transfer size register_17 | DMATSR_17 | R/W | H'EC72_80A8* | 32 | √ | √ | √ | — |
| DMA channel control register_17 | DMACHCR_17 | R/W | H'EC72_808C H'EC72_80AC* | 32 | √ | √ | √ | — |
| DMA transfer count register B_17 | DMATCRB_17 | R/W | H'EC72_8098 | 32 | √ | √ | √ | — |
| DMA transfer size register B_17 | DMATSRB_17 | R/W | H'EC72_80B8* | 32 | √ | √ | √ | — |
| DMA channel control register B_17 | DMACHCRB_17 | R/W | H'EC72_809C | 32 | √ | √ | √ | — |
| DMA extended resource selector_17 | DMARS_17 | R/W | H'EC72_80C0 | 16 | √ | √ | √ | — |
| DMA buffer control register_17 | DMABUFCR_17 | R/W | H'EC72_80C8 | 32 | √ | √ | √ | — |
| DMA descriptor base address register_17 | DMADPBASE_17 | R/W | H'EC72_80D0 | 32 | √ | √ | √ | — |
| DMA descriptor control register_17 | DMADPCR_17 | R/W | H'EC72_80D4 | 32 | √ | √ | √ | — |
| DMA fixed source address register_17 | DMAFIXSAR_17 | R/W | H'EC72_8090 | 32 | √ | √ | √ | — |
| DMA fixed destination address register_17 | DMAFIXDAR_17 | R/W | H'EC72_8094 | 32 | √ | √ | √ | — |
| DMA fixed descriptor base address register_17 | DMAFIXDPBASE_17 | R/W | H'EC72_80E0 | 32 | √ | √ | √ | — |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|--|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| DMA source address register_18 | DMASAR_18 | R/W | H'EC72_8100 H'EC72_8120* | 32 | √ | √ | √ | — | |
| DMA destination address register_18 | DMADAR_18 | R/W | H'EC72_8104 H'EC72_8124* | 32 | √ | √ | √ | — | |
| DMA transfer count register_18 | DMATCR_18 | R/W | H'EC72_8108 | 32 | √ | √ | √ | — | |
| DMA transfer size register_18 | DMATSR_18 | R/W | H'EC72_8128* | 32 | √ | √ | √ | — | |
| DMA channel control register_18 | DMACHCR_18 | R/W | H'EC72_810C H'EC72_812C* | 32 | √ | √ | √ | — | |
| DMA transfer count register B_18 | DMATCRB_18 | R/W | H'EC72_8118 | 32 | √ | √ | √ | — | |
| DMA transfer size register B_18 | DMATSRB_18 | R/W | H'EC72_8138* | 32 | √ | √ | √ | — | |
| DMA channel control register B_18 | DMACHCRB_18 | R/W | H'EC72_811C | 32 | √ | √ | √ | — | |
| DMA extended resource selector_18 | DMARS_18 | R/W | H'EC72_8140 | 16 | √ | √ | √ | — | |
| DMA buffer control register_18 | DMABUFCR_18 | R/W | H'EC72_8148 | 32 | √ | √ | √ | — | |
| DMA descriptor base address register_18 | DMADPBASE_18 | R/W | H'EC72_8150 | 32 | √ | √ | √ | — | |
| DMA descriptor control register_18 | DMADPCR_18 | R/W | H'EC72_8154 | 32 | √ | √ | √ | — | |
| DMA fixed source address register_18 | DMAFIXSAR_18 | R/W | H'EC72_8110 | 32 | √ | √ | √ | — | |
| DMA fixed destination address register_18 | DMAFIXDAR_18 | R/W | H'EC72_8114 | 32 | √ | √ | √ | — | |
| DMA fixed descriptor base address register_18 | DMAFIXDPBASE_18 | R/W | H'EC72_8160 | 32 | √ | √ | √ | — | |
| DMA source address register_19 | DMASAR_19 | R/W | H'EC72_8180 H'EC72_81A0* | 32 | √ | √ | √ | — | |
| DMA destination address register_19 | DMADAR_19 | R/W | H'EC72_8184 H'EC72_81A4* | 32 | √ | √ | √ | — | |
| DMA transfer count register_19 | DMATCR_19 | R/W | H'EC72_8188 | 32 | √ | √ | √ | — | |
| DMA transfer size register_19 | DMATSR_19 | R/W | H'EC72_81A8* | 32 | √ | √ | √ | — | |
| DMA channel control register_19 | DMACHCR_19 | R/W | H'EC72_818C H'EC72_81AC* | 32 | √ | √ | √ | — | |
| DMA transfer count register B_19 | DMATCRB_19 | R/W | H'EC72_8198 | 32 | √ | √ | √ | — | |
| DMA transfer size register B_19 | DMATSRB_19 | R/W | H'EC72_81B8* | 32 | √ | √ | √ | — | |
| DMA channel control register B_19 | DMACHCRB_19 | R/W | H'EC72_819C | 32 | √ | √ | √ | — | |
| DMA extended resource selector_19 | DMARS_19 | R/W | H'EC72_81C0 | 16 | √ | √ | √ | — | |
| DMA buffer control register_19 | DMABUFCR_19 | R/W | H'EC72_81C8 | 32 | √ | √ | √ | — | |
| DMA descriptor base address register_19 | DMADPBASE_19 | R/W | H'EC72_81D0 | 32 | √ | √ | √ | — | |
| DMA descriptor control register_19 | DMADPCR_19 | R/W | H'EC72_81D4 | 32 | √ | √ | √ | — | |
| DMA fixed source address register_19 | DMAFIXSAR_19 | R/W | H'EC72_8190 | 32 | √ | √ | √ | — | |
| DMA fixed destination address register_19 | DMAFIXDAR_19 | R/W | H'EC72_8194 | 32 | √ | √ | √ | — | |
| DMA fixed descriptor base address register_19 | DMAFIXDPBASE_19 | R/W | H'EC72_81E0 | 32 | √ | √ | √ | — | |
| DMA source address register_20 | DMASAR_20 | R/W | H'EC72_8200 H'EC72_8220* | 32 | √ | √ | √ | — | |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA destination address register_20 | DMADAR_20 | R/W | H'EC72_8204 H'EC72_8224* | 32 | √ | √ | √ | — |
| DMA transfer count register_20 | DMATCR_20 | R/W | H'EC72_8208 | 32 | √ | √ | √ | — |
| DMA transfer size register_20 | DMATSR_20 | R/W | H'EC72_8228* | 32 | √ | √ | √ | — |
| DMA channel control register_20 | DMACHCR_20 | R/W | H'EC72_820C H'EC72_822C* | 32 | √ | √ | √ | — |
| DMA transfer count register B_20 | DMATCRB_20 | R/W | H'EC72_8218 | 32 | √ | √ | √ | — |
| DMA transfer size register B_20 | DMATSRB_20 | R/W | H'EC72_8238* | 32 | √ | √ | √ | — |
| DMA channel control register B_20 | DMACHCRB_20 | R/W | H'EC72_821C | 32 | √ | √ | √ | — |
| DMA extended resource selector_20 | DMARS_20 | R/W | H'EC72_8240 | 16 | √ | √ | √ | — |
| DMA buffer control register_20 | DMABUFCR_20 | R/W | H'EC72_8248 | 32 | √ | √ | √ | — |
| DMA descriptor base address register_20 | DMADPBASE_20 | R/W | H'EC72_8250 | 32 | √ | √ | √ | — |
| DMA descriptor control register_20 | DMADPCR_20 | R/W | H'EC72_8254 | 32 | √ | √ | √ | — |
| DMA fixed source address register_20 | DMAFIXSAR_20 | R/W | H'EC72_8210 | 32 | √ | √ | √ | — |
| DMA fixed destination address register_20 | DMAFIXDAR_20 | R/W | H'EC72_8214 | 32 | √ | √ | √ | — |
| DMA fixed descriptor base address register_20 | DMAFIXDPBASE_20 | R/W | H'EC72_8260 | 32 | √ | √ | √ | — |
| DMA source address register_21 | DMASAR_21 | R/W | H'EC72_8280 H'EC72_82A0* | 32 | √ | √ | √ | — |
| DMA destination address register_21 | DMADAR_21 | R/W | H'EC72_8284 H'EC72_82A4* | 32 | √ | √ | √ | — |
| DMA transfer count register_21 | DMATCR_21 | R/W | H'EC72_8288 | 32 | √ | √ | √ | — |
| DMA transfer size register_21 | DMATSR_21 | R/W | H'EC72_82A8* | 32 | √ | √ | √ | — |
| DMA channel control register_21 | DMACHCR_21 | R/W | H'EC72_828C H'EC72_82AC* | 32 | √ | √ | √ | — |
| DMA transfer count register B_21 | DMATCRB_21 | R/W | H'EC72_8298 | 32 | √ | √ | √ | — |
| DMA transfer size register B_21 | DMATSRB_21 | R/W | H'EC72_82B8* | 32 | √ | √ | √ | — |
| DMA channel control register B_21 | DMACHCRB_21 | R/W | H'EC72_829C | 32 | √ | √ | √ | — |
| DMA extended resource selector_21 | DMARS_21 | R/W | H'EC72_82C0 | 16 | √ | √ | √ | — |
| DMA buffer control register_21 | DMABUFCR_21 | R/W | H'EC72_82C8 | 32 | √ | √ | √ | — |
| DMA descriptor base address register_21 | DMADPBASE_21 | R/W | H'EC72_82D0 | 32 | √ | √ | √ | — |
| DMA descriptor control register_21 | DMADPCR_21 | R/W | H'EC72_82D4 | 32 | √ | √ | √ | — |
| DMA fixed source address register_21 | DMAFIXSAR_21 | R/W | H'EC72_8290 | 32 | √ | √ | √ | — |
| DMA fixed destination address register_21 | DMAFIXDAR_21 | R/W | H'EC72_8294 | 32 | √ | √ | √ | — |
| DMA fixed descriptor base address register_21 | DMAFIXDPBASE_21 | R/W | H'EC72_82E0 | 32 | √ | √ | √ | — |
| DMA source address register_22 | DMASAR_22 | R/W | H'EC72_8300 H'EC72_8320* | 32 | √ | √ | √ | — |
| DMA destination address register_22 | DMADAR_22 | R/W | H'EC72_8304 H'EC72_8324* | 32 | √ | √ | √ | — |
| DMA transfer count register_22 | DMATCR_22 | R/W | H'EC72_8308 | 32 | √ | √ | √ | — |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA transfer size register_22 | DMATSR_22 | R/W | H'EC72_8328* | 32 | √ | √ | √ | — |
| DMA channel control register_22 | DMACHCR_22 | R/W | H'EC72_830C H'EC72_832C* | 32 | √ | √ | √ | — |
| DMA transfer count register B_22 | DMATCRB_22 | R/W | H'EC72_8318 | 32 | √ | √ | √ | — |
| DMA transfer size register B_22 | DMATSRB_22 | R/W | H'EC72_8338* | 32 | √ | √ | √ | — |
| DMA channel control register B_22 | DMACHCRB_22 | R/W | H'EC72_831C | 32 | √ | √ | √ | — |
| DMA extended resource selector_22 | DMARS_22 | R/W | H'EC72_8340 | 16 | √ | √ | √ | — |
| DMA buffer control register_22 | DMABUFCR_22 | R/W | H'EC72_8348 | 32 | √ | √ | √ | — |
| DMA descriptor base address register_22 | DMADPBASE_22 | R/W | H'EC72_8350 | 32 | √ | √ | √ | — |
| DMA descriptor control register_22 | DMADPCR_22 | R/W | H'EC72_8354 | 32 | √ | √ | √ | — |
| DMA fixed source address register_22 | DMAFIXSAR_22 | R/W | H'EC72_8310 | 32 | √ | √ | √ | — |
| DMA fixed destination address register_22 | DMAFIXDAR_22 | R/W | H'EC72_8314 | 32 | √ | √ | √ | — |
| DMA fixed descriptor base address register_22 | DMAFIXDPBASE_22 | R/W | H'EC72_8360 | 32 | √ | √ | √ | — |
| DMA source address register_23 | DMASAR_23 | R/W | H'EC72_8380 H'EC72_83A0* | 32 | √ | √ | √ | — |
| DMA destination address register_23 | DMADAR_23 | R/W | H'EC72_8384 H'EC72_83A4* | 32 | √ | √ | √ | — |
| DMA transfer count register_23 | DMATCR_23 | R/W | H'EC72_8388 | 32 | √ | √ | √ | — |
| DMA transfer size register_23 | DMATSR_23 | R/W | H'EC72_83A8* | 32 | √ | √ | √ | — |
| DMA channel control register_23 | DMACHCR_23 | R/W | H'EC72_838C H'EC72_83AC* | 32 | √ | √ | √ | — |
| DMA transfer count register B_23 | DMATCRB_23 | R/W | H'EC72_8398 | 32 | √ | √ | √ | — |
| DMA transfer size register B_23 | DMATSRB_23 | R/W | H'EC72_83B8* | 32 | √ | √ | √ | — |
| DMA channel control register B_23 | DMACHCRB_23 | R/W | H'EC72_839C | 32 | √ | √ | √ | — |
| DMA extended resource selector_23 | DMARS_23 | R/W | H'EC72_83C0 | 16 | √ | √ | √ | — |
| DMA buffer control register_23 | DMABUFCR_23 | R/W | H'EC72_83C8 | 32 | √ | √ | √ | — |
| DMA descriptor base address register_23 | DMADPBASE_23 | R/W | H'EC72_83D0 | 32 | √ | √ | √ | — |
| DMA descriptor control register_23 | DMADPCR_23 | R/W | H'EC72_83D4 | 32 | √ | √ | √ | — |
| DMA fixed source address register_23 | DMAFIXSAR_23 | R/W | H'EC72_8390 | 32 | √ | √ | √ | — |
| DMA fixed destination address register_23 | DMAFIXDAR_23 | R/W | H'EC72_8394 | 32 | √ | √ | √ | — |
| DMA fixed descriptor base address register_23 | DMAFIXDPBASE_23 | R/W | H'EC72_83E0 | 32 | √ | √ | √ | — |
| DMA source address register_24 | DMASAR_24 | R/W | H'EC72_8400 H'EC72_8420* | 32 | √ | √ | √ | — |
| DMA destination address register_24 | DMADAR_24 | R/W | H'EC72_8404 H'EC72_8424* | 32 | √ | √ | √ | — |
| DMA transfer count register_24 | DMATCR_24 | R/W | H'EC72_8408 | 32 | √ | √ | √ | — |
| DMA transfer size register_24 | DMATSR_24 | R/W | H'EC72_8428* | 32 | √ | √ | √ | — |
| DMA channel control register_24 | DMACHCR_24 | R/W | H'EC72_840C H'EC72_842C* | 32 | √ | √ | √ | — |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA transfer count register B_24 | DMATCRB_24 | R/W | H'EC72_8418 | 32 | √ | √ | √ | — |
| DMA transfer size register B_24 | DMATSRB_24 | R/W | H'EC72_8438* | 32 | √ | √ | √ | — |
| DMA channel control register B_24 | DMACHCRB_24 | R/W | H'EC72_841C | 32 | √ | √ | √ | — |
| DMA extended resource selector_24 | DMARS_24 | R/W | H'EC72_8440 | 16 | √ | √ | √ | — |
| DMA buffer control register_24 | DMABUFCR_24 | R/W | H'EC72_8448 | 32 | √ | √ | √ | — |
| DMA descriptor base address register_24 | DMADPBASE_24 | R/W | H'EC72_8450 | 32 | √ | √ | √ | — |
| DMA descriptor control register_24 | DMADPCR_24 | R/W | H'EC72_8454 | 32 | √ | √ | √ | — |
| DMA fixed source address register_24 | DMAFIXSAR_24 | R/W | H'EC72_8410 | 32 | √ | √ | √ | — |
| DMA fixed destination address register_24 | DMAFIXDAR_24 | R/W | H'EC72_8414 | 32 | √ | √ | √ | — |
| DMA fixed descriptor base address register_24 | DMAFIXDPBASE_24 | R/W | H'EC72_8460 | 32 | √ | √ | √ | — |
| DMA source address register_25 | DMASAR_25 | R/W | H'EC72_8480 H'EC72_84A0* | 32 | √ | √ | √ | — |
| DMA destination address register_25 | DMADAR_25 | R/W | H'EC72_8484 H'EC72_84A4* | 32 | √ | √ | √ | — |
| DMA transfer count register_25 | DMATCR_25 | R/W | H'EC72_8488 | 32 | √ | √ | √ | — |
| DMA transfer size register_25 | DMATSR_25 | R/W | H'EC72_84A8* | 32 | √ | √ | √ | — |
| DMA channel control register_25 | DMACHCR_25 | R/W | H'EC72_848C H'EC72_84AC* | 32 | √ | √ | √ | — |
| DMA transfer count register B_25 | DMATCRB_25 | R/W | H'EC72_8498 | 32 | √ | √ | √ | — |
| DMA transfer size register B_25 | DMATSRB_25 | R/W | H'EC72_84B8* | 32 | √ | √ | √ | — |
| DMA channel control register B_25 | DMACHCRB_25 | R/W | H'EC72_849C | 32 | √ | √ | √ | — |
| DMA extended resource selector_25 | DMARS_25 | R/W | H'EC72_84C0 | 16 | √ | √ | √ | — |
| DMA buffer control register_25 | DMABUFCR_25 | R/W | H'EC72_84C8 | 32 | √ | √ | √ | — |
| DMA descriptor base address register_25 | DMADPBASE_25 | R/W | H'EC72_84D0 | 32 | √ | √ | √ | — |
| DMA descriptor control register_25 | DMADPCR_25 | R/W | H'EC72_84D4 | 32 | √ | √ | √ | — |
| DMA fixed source address register_25 | DMAFIXSAR_25 | R/W | H'EC72_8490 | 32 | √ | √ | √ | — |
| DMA fixed destination address register_25 | DMAFIXDAR_25 | R/W | H'EC72_8494 | 32 | √ | √ | √ | — |
| DMA fixed descriptor base address register_25 | DMAFIXDPBASE_25 | R/W | H'EC72_84E0 | 32 | √ | √ | √ | — |
| DMA source address register_26 | DMASAR_26 | R/W | H'EC72_8500 H'EC72_8520* | 32 | √ | √ | √ | — |
| DMA destination address register_26 | DMADAR_26 | R/W | H'EC72_8504 H'EC72_8524* | 32 | √ | √ | √ | — |
| DMA transfer count register_26 | DMATCR_26 | R/W | H'EC72_8508 | 32 | √ | √ | √ | — |
| DMA transfer size register_26 | DMATSR_26 | R/W | H'EC72_8528* | 32 | √ | √ | √ | — |
| DMA channel control register_26 | DMACHCR_26 | R/W | H'EC72_850C H'EC72_852C* | 32 | √ | √ | √ | — |
| DMA transfer count register B_26 | DMATCRB_26 | R/W | H'EC72_8518 | 32 | √ | √ | √ | — |
| DMA transfer size register B_26 | DMATSRB_26 | R/W | H'EC72_8538* | 32 | √ | √ | √ | — |
| DMA channel control register B_26 | DMACHCRB_26 | R/W | H'EC72_851C | 32 | √ | √ | √ | — |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA extended resource selector_26 | DMARS_26 | R/W | H'EC72_8540 | 16 | √ | √ | √ | — |
| DMA buffer control register_26 | DMABUFCR_26 | R/W | H'EC72_8548 | 32 | √ | √ | √ | — |
| DMA descriptor base address register_26 | DMADPBASE_26 | R/W | H'EC72_8550 | 32 | √ | √ | √ | — |
| DMA descriptor control register_26 | DMADPCR_26 | R/W | H'EC72_8554 | 32 | √ | √ | √ | — |
| DMA fixed source address register_26 | DMAFIXSAR_26 | R/W | H'EC72_8510 | 32 | √ | √ | √ | — |
| DMA fixed destination address register_26 | DMAFIXDAR_26 | R/W | H'EC72_8514 | 32 | √ | √ | √ | — |
| DMA fixed descriptor base address register_26 | DMAFIXDPBASE_26 | R/W | H'EC72_8560 | 32 | √ | √ | √ | — |
| DMA source address register_27 | DMASAR_27 | R/W | H'EC72_8580 H'EC72_85A0* | 32 | √ | √ | √ | — |
| DMA destination address register_27 | DMADAR_27 | R/W | H'EC72_8584 H'EC72_85A4* | 32 | √ | √ | √ | — |
| DMA transfer count register_27 | DMATCR_27 | R/W | H'EC72_8588 | 32 | √ | √ | √ | — |
| DMA transfer size register_27 | DMATSR_27 | R/W | H'EC72_85A8* | 32 | √ | √ | √ | — |
| DMA channel control register_27 | DMACHCR_27 | R/W | H'EC72_858C H'EC72_85AC* | 32 | √ | √ | √ | — |
| DMA transfer count register B_27 | DMATCRB_27 | R/W | H'EC72_8598 | 32 | √ | √ | √ | — |
| DMA transfer size register B_27 | DMATSRB_27 | R/W | H'EC72_85B8* | 32 | √ | √ | √ | — |
| DMA channel control register B_27 | DMACHCRB_27 | R/W | H'EC72_859C | 32 | √ | √ | √ | — |
| DMA extended resource selector_27 | DMARS_27 | R/W | H'EC72_85C0 | 16 | √ | √ | √ | — |
| DMA buffer control register_27 | DMABUFCR_27 | R/W | H'EC72_85C8 | 32 | √ | √ | √ | — |
| DMA descriptor base address register_27 | DMADPBASE_27 | R/W | H'EC72_85D0 | 32 | √ | √ | √ | — |
| DMA descriptor control register_27 | DMADPCR_27 | R/W | H'EC72_85D4 | 32 | √ | √ | √ | — |
| DMA fixed source address register_27 | DMAFIXSAR_27 | R/W | H'EC72_8590 | 32 | √ | √ | √ | — |
| DMA fixed destination address register_27 | DMAFIXDAR_27 | R/W | H'EC72_8594 | 32 | √ | √ | √ | — |
| DMA fixed descriptor base address register_27 | DMAFIXDPBASE_27 | R/W | H'EC72_85E0 | 32 | √ | √ | √ | — |
| DMA source address register_28 | DMASAR_28 | R/W | H'EC72_8600 H'EC72_8620* | 32 | √ | √ | √ | — |
| DMA destination address register_28 | DMADAR_28 | R/W | H'EC72_8604 H'EC72_8624* | 32 | √ | √ | √ | — |
| DMA transfer count register_28 | DMATCR_28 | R/W | H'EC72_8608 | 32 | √ | √ | √ | — |
| DMA transfer size register_28 | DMATSR_28 | R/W | H'EC72_8628* | 32 | √ | √ | √ | — |
| DMA channel control register_28 | DMACHCR_28 | R/W | H'EC72_860C H'EC72_862C* | 32 | √ | √ | √ | — |
| DMA transfer count register B_28 | DMATCRB_28 | R/W | H'EC72_8618 | 32 | √ | √ | √ | — |
| DMA transfer size register B_28 | DMATSRB_28 | R/W | H'EC72_8638* | 32 | √ | √ | √ | — |
| DMA channel control register B_28 | DMACHCRB_28 | R/W | H'EC72_861C | 32 | √ | √ | √ | — |
| DMA extended resource selector_28 | DMARS_28 | R/W | H'EC72_8640 | 16 | √ | √ | √ | — |
| DMA buffer control register_28 | DMABUFCR_28 | R/W | H'EC72_8648 | 32 | √ | √ | √ | — |
| DMA descriptor base address register_28 | DMADPBASE_28 | R/W | H'EC72_8650 | 32 | √ | √ | √ | — |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA descriptor control register_28 | DMADPCR_28 | R/W | H'EC72_8654 | 32 | √ | √ | √ | — |
| DMA fixed source address register_28 | DMAFIXSAR_28 | R/W | H'EC72_8610 | 32 | √ | √ | √ | — |
| DMA fixed destination address register_28 | DMAFIXDAR_28 | R/W | H'EC72_8614 | 32 | √ | √ | √ | — |
| DMA fixed descriptor base address register_28 | DMAFIXDPBASE_28 | R/W | H'EC72_8660 | 32 | √ | √ | √ | — |
| DMA source address register_29 | DMASAR_29 | R/W | H'EC72_8680 H'EC72_86A0* | 32 | √ | √ | √ | — |
| DMA destination address register_29 | DMADAR_29 | R/W | H'EC72_8684 H'EC72_86A4* | 32 | √ | √ | √ | — |
| DMA transfer count register_29 | DMATCR_29 | R/W | H'EC72_8688 | 32 | √ | √ | √ | — |
| DMA transfer size register_29 | DMATSR_29 | R/W | H'EC72_86A8* | 32 | √ | √ | √ | — |
| DMA channel control register_29 | DMACHCR_29 | R/W | H'EC72_868C H'EC72_86AC* | 32 | √ | √ | √ | — |
| DMA transfer count register B_29 | DMATCRB_29 | R/W | H'EC72_8698 | 32 | √ | √ | √ | — |
| DMA transfer size register B_29 | DMATSRB_29 | R/W | H'EC72_86B8* | 32 | √ | √ | √ | — |
| DMA channel control register B_29 | DMACHCRB_29 | R/W | H'EC72_869C | 32 | √ | √ | √ | — |
| DMA extended resource selector_29 | DMARS_29 | R/W | H'EC72_86C0 | 16 | √ | √ | √ | — |
| DMA buffer control register_29 | DMABUFCR_29 | R/W | H'EC72_86C8 | 32 | √ | √ | √ | — |
| DMA descriptor base address register_29 | DMADPBASE_29 | R/W | H'EC72_86D0 | 32 | √ | √ | √ | — |
| DMA descriptor control register_29 | DMADPCR_29 | R/W | H'EC72_86D4 | 32 | √ | √ | √ | — |
| DMA fixed source address register_29 | DMAFIXSAR_29 | R/W | H'EC72_8690 | 32 | √ | √ | √ | — |
| DMA fixed destination address register_29 | DMAFIXDAR_29 | R/W | H'EC72_8694 | 32 | √ | √ | √ | — |
| DMA fixed descriptor base address register_29 | DMAFIXDPBASE_29 | R/W | H'EC72_86E0 | 32 | √ | √ | √ | — |
| DMA source address register_30 | DMASAR_30 | R/W | H'EC72_8700 H'EC72_8720* | 32 | √ | √ | √ | — |
| DMA destination address register_30 | DMADAR_30 | R/W | H'EC72_8704 H'EC72_8724* | 32 | √ | √ | √ | — |
| DMA transfer count register_30 | DMATCR_30 | R/W | H'EC72_8708 | 32 | √ | √ | √ | — |
| DMA transfer size register_30 | DMATSR_30 | R/W | H'EC72_8728* | 32 | √ | √ | √ | — |
| DMA channel control register_30 | DMACHCR_30 | R/W | H'EC72_870C H'EC72_872C* | 32 | √ | √ | √ | — |
| DMA transfer count register B_30 | DMATCRB_30 | R/W | H'EC72_8718 | 32 | √ | √ | √ | — |
| DMA transfer size register B_30 | DMATSRB_30 | R/W | H'EC72_8738* | 32 | √ | √ | √ | — |
| DMA channel control register B_30 | DMACHCRB_30 | R/W | H'EC72_871C | 32 | √ | √ | √ | — |
| DMA extended resource selector_30 | DMARS_30 | R/W | H'EC72_8740 | 16 | √ | √ | √ | — |
| DMA buffer control register_30 | DMABUFCR_30 | R/W | H'EC72_8748 | 32 | √ | √ | √ | — |
| DMA descriptor base address register_30 | DMADPBASE_30 | R/W | H'EC72_8750 | 32 | √ | √ | √ | — |
| DMA descriptor control register_30 | DMADPCR_30 | R/W | H'EC72_8754 | 32 | √ | √ | √ | — |
| DMA fixed source address register_30 | DMAFIXSAR_30 | R/W | H'EC72_8710 | 32 | √ | √ | √ | — |
| DMA fixed destination address register_30 | DMAFIXDAR_30 | R/W | H'EC72_8714 | 32 | √ | √ | √ | — |

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DMA fixed descriptor base address register_30 | DMAFIXDPBASE_30 | R/W | H'EC72_8760 | 32 | √ | √ | √ | — |
| DMA source address register_31 | DMASAR_31 | R/W | H'EC72_8780 H'EC72_87A0* | 32 | √ | √ | √ | — |
| DMA destination address register_31 | DMADAR_31 | R/W | H'EC72_8784 H'EC72_87A4* | 32 | √ | √ | √ | — |
| DMA transfer count register_31 | DMATCR_31 | R/W | H'EC72_8788 | 32 | √ | √ | √ | — |
| DMA transfer size register_31 | DMATSR_31 | R/W | H'EC72_87A8* | 32 | √ | √ | √ | — |
| DMA channel control register_31 | DMACHCR_31 | R/W | H'EC72_878C H'EC72_87AC* | 32 | √ | √ | √ | — |
| DMA transfer count register B_31 | DMATCRB_31 | R/W | H'EC72_8798 | 32 | √ | √ | √ | — |
| DMA transfer size register B_31 | DMATSRB_31 | R/W | H'EC72_87B8* | 32 | √ | √ | √ | — |
| DMA channel control register B_31 | DMACHCRB_31 | R/W | H'EC72_879C | 32 | √ | √ | √ | — |
| DMA extended resource selector_31 | DMARS_31 | R/W | H'EC72_87C0 | 16 | √ | √ | √ | — |
| DMA buffer control register_31 | DMABUFCR_31 | R/W | H'EC72_87C8 | 32 | √ | √ | √ | — |
| DMA descriptor base address register_31 | DMADPBASE_31 | R/W | H'EC72_87D0 | 32 | √ | √ | √ | — |
| DMA descriptor control register_31 | DMADPCR_31 | R/W | H'EC72_87D4 | 32 | √ | √ | √ | — |
| DMA fixed source address register_31 | DMAFIXSAR_31 | R/W | H'EC72_8790 | 32 | √ | √ | √ | — |
| DMA fixed destination address register_31 | DMAFIXDAR_31 | R/W | H'EC72_8794 | 32 | √ | √ | √ | — |
| DMA fixed descriptor base address register_31 | DMAFIXDPBASE_31 | R/W | H'EC72_87E0 | 32 | √ | √ | √ | — |
| Secure function Secure Status register (for channels 16 to 31) | DMASES_U | R/W | H'EC72_00C0 | 32 | √ | √ | √ | — |
| Secure function Slave Error Address register (for channels 16 to 31) | DMASEDDR_U | R | H'EC72_00C4 | 32 | √ | √ | √ | — |
| Secure function Error Master ID register (for channels 16 to 31) | DMASEMID_U | R | H'EC72_00C8 | 32 | √ | √ | √ | — |

Note: For RZ/G Gen2, the base address of registers for the lower-numbered channels (0 to 15) is H'EC70_0000.

For RZ/G Gen2, the base address of registers for the higher-numbered channels (16 to 31) is H'EC72_0000.

* This address is used in total size transmission (see section 44.3.6, Total Size Transmission).

Table 44.2 States of Audio-DMAC Registers in each Operating Mode

| Abbreviation | Power-On Reset | Module Standby |
|-----------------------------------|----------------|----------------|
| DMAISTA_L/ DMAISTA_U | Initialized | Retained |
| DMASEC_L/ DMASEC_H | Initialized | Retained |
| DMAOR_L/ DMAOR_U | Initialized | Retained |
| DMACHCLR_L/ DMACHCLR_U | Initialized | Retained |
| DMADPSEC_L/ DMADPSEC_U | Initialized | Retained |
| DMASAR_0 to DMASAR_31 | Initialized | Retained |
| DMADAR_0 to DMADAR_31 | Initialized | Retained |
| DMATCR_0 to DMATCR_31 | Initialized | Retained |
| DMATSR_0 to DMATSR_31 | Initialized | Retained |
| DMACHCR_0 to DMACHCR_31 | Initialized | Retained |
| DMATCRB_0 DMATCRB_31 | Initialized | Retained |
| DMATSRB_0 to DMATSRB_31 | Initialized | Retained |
| DMACHCRB_0 to DMACHCRB_31 | Initialized | Retained |
| DMABUFCR_0 to DMABUFCR_31 | Initialized | Retained |
| DMARS_0 to DMARS_31 | Initialized | Retained |
| DMADPBASE_0 to DMADPBASE_31 | Initialized | Retained |
| DMADPCR_0 DMADPCR_31 | Initialized | Retained |
| DMAFIXSAR_0 to DMAFIXSAR_31 | Initialized | Retained |
| DMAFIXDAR_0 to DMAFIXDAR_31 | Initialized | Retained |
| DMAFIXDPBASE_0 to DMAFIXDPBASE_31 | Initialized | Retained |
| DescriptorMEM | Undefined | Retained |
| DMASES_L/ DMASES_U | Initialized | Retained |
| DMASEDDR_L/ DMASEDDR_U | Initialized | Retained |
| DMASEMID_L/ DMASEMID_U | Initialized | Retained |

44.1.4 Connected Modules

Table 44.3 shows the connected modules to Audio-DMAC. As for the feature of the connected modules, refer to the chapter of each modules.

Table 44.3 Connected Modules

| Module name | Connected module name | Function of connected module |
|--|-----------------------|--|
| Direct Memory Access Controller for Audio (Audio-DMAC) | CPG | Output clocks |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |
| | INTC-SYS | Control to interrupt |
| | IPMMU | UTLB function |
| | Audio AXI Main Bus | MEM I/F-128bit, BUS I/F-64bit connection |
| | Audio AXI DMAC Bus | PERI I/F-64bit connection |
| | D-REQ synchronous | DMA request and RACK handshake |

44.2 Register Description

44.2.1 DMA Interrupt Status Register for Lower-Numbered Channels (DMAISTA_L)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMAISTA_L is a 32-bit readable register that indicates the states of the interrupt signals for each of the lower-numbered channels.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | I15 | I14 | I13 | I12 | I11 | I10 | I9 | I8 | I7 | I6 | I5 | I4 | I3 | I2 | I1 | I0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | I15 | B'0 | R | Interrupt State in Channel 15 0: An interrupt is not present. 1: An interrupt is present. |
| 14 | I14 | B'0 | R | Interrupt State in Channel 14 0: An interrupt is not present. 1: An interrupt is present. |
| 13 | I13 | B'0 | R | Interrupt State in Channel 13 0: An interrupt is not present. 1: An interrupt is present. |
| 12 | I12 | B'0 | R | Interrupt State in Channel 12 0: An interrupt is not present. 1: An interrupt is present. |
| 11 | I11 | B'0 | R | Interrupt State in Channel 11 0: An interrupt is not present. 1: An interrupt is present. |
| 10 | I10 | B'0 | R | Interrupt State in Channel 10 0: An interrupt is not present. 1: An interrupt is present. |
| 9 | I9 | B'0 | R | Interrupt State in Channel 9 0: An interrupt is not present. 1: An interrupt is present. |
| 8 | I8 | B'0 | R | Interrupt State in Channel 8 0: An interrupt is not present. 1: An interrupt is present. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|--|
| 7 | I7 | B'0 | R | Interrupt State in Channel 7 0: An interrupt is not present. 1: An interrupt is present. |
| 6 | I6 | B'0 | R | Interrupt State in Channel 6 0: An interrupt is not present. 1: An interrupt is present. |
| 5 | I5 | B'0 | R | Interrupt State in Channel 5 0: An interrupt is not present. 1: An interrupt is present. |
| 4 | I4 | B'0 | R | Interrupt State in Channel 4 0: An interrupt is not present. 1: An interrupt is present. |
| 3 | I3 | B'0 | R | Interrupt State in Channel 3 0: An interrupt is not present. 1: An interrupt is present. |
| 2 | I2 | B'0 | R | Interrupt State in Channel 2 0: An interrupt is not present. 1: An interrupt is present. |
| 1 | I1 | B'0 | R | Interrupt State in Channel 1 0: An interrupt is not present. 1: An interrupt is present. |
| 0 | I0 | B'0 | R | Interrupt State in Channel 0 0: An interrupt is not present. 1: An interrupt is present. |

44.2.2 DMA Interrupt Status Register for Higher-Numbered Channels (DMAISTA_U)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

DMAISTA_U is a 32-bit readable register that indicates the states of the interrupt signals for each of the higher-numbered channels.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | I31 | I30 | I29 | I28 | I27 | I26 | I25 | I24 | I23 | I22 | I21 | I20 | I19 | I18 | I17 | I16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | I31 | B'0 | R | Interrupt State in Channel 31 0: An interrupt is not present. 1: An interrupt is present. |
| 14 | I30 | B'0 | R | Interrupt State in Channel 30 0: An interrupt is not present. 1: An interrupt is present. |
| 13 | I29 | B'0 | R | Interrupt State in Channel 29 0: An interrupt is not present. 1: An interrupt is present. |
| 12 | I28 | B'0 | R | Interrupt State in Channel 28 0: An interrupt is not present. 1: An interrupt is present. |
| 11 | I27 | B'0 | R | Interrupt State in Channel 27 0: An interrupt is not present. 1: An interrupt is present. |
| 10 | I26 | B'0 | R | Interrupt State in Channel 26 0: An interrupt is not present. 1: An interrupt is present. |
| 9 | I25 | B'0 | R | Interrupt State in Channel 25 0: An interrupt is not present. 1: An interrupt is present. |
| 8 | I24 | B'0 | R | Interrupt State in Channel 24 0: An interrupt is not present. 1: An interrupt is present. |
| 7 | I23 | B'0 | R | Interrupt State in Channel 23 0: An interrupt is not present. 1: An interrupt is present. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|---|
| 6 | I22 | B'0 | R | Interrupt State in Channel 22 0: An interrupt is not present. 1: An interrupt is present. |
| 5 | I21 | B'0 | R | Interrupt State in Channel 21 0: An interrupt is not present. 1: An interrupt is present. |
| 4 | I20 | B'0 | R | Interrupt State in Channel 20 0: An interrupt is not present. 1: An interrupt is present. |
| 3 | I19 | B'0 | R | Interrupt State in Channel 19 0: An interrupt is not present. 1: An interrupt is present. |
| 2 | I18 | B'0 | R | Interrupt State in Channel 18 0: An interrupt is not present. 1: An interrupt is present. |
| 1 | I17 | B'0 | R | Interrupt State in Channel 17 0: An interrupt is not present. 1: An interrupt is present. |
| 0 | I16 | B'0 | R | Interrupt State in Channel 16 0: An interrupt is not present. 1: An interrupt is present. |

44.2.3 DMA Secure Control Register for Lower-Numbered Channels (DMASEC_L)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASEC_L is a 32-bit readable/writable register that controls the security attribute of each of the lower-numbered channels. Only the initiator in the secure mode can change the setting of this register.

Only secure access is allowed to registers of channels with the secure mode setting. The following registers are protected by the secure mode.

DMASAR, DMADAR, DMATCR, DMATSR, DMACHCR, DMATCRB, DMATSRB, DMACHCRB, DMARS, DMABUFGR, DMADPBASE, DMADPCR, DMAFIXSAR, DMAFIXDAR, and DMAFIXDPBASE

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | S15 | S14 | S13 | S12 | S11 | S10 | S9 | S8 | S7 | S6 | S5 | S4 | S3 | S2 | S1 | S0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | S15 | B'0 | R/W | Secure Mode Setting for Channel 15 0: Non-secure mode 1: Secure mode |
| 14 | S14 | B'0 | R/W | Secure Mode Setting for Channel 14 0: Non-secure mode 1: Secure mode |
| 13 | S13 | B'0 | R/W | Secure Mode Setting for Channel 13 0: Non-secure mode 1: Secure mode |
| 12 | S12 | B'0 | R/W | Secure Mode Setting for Channel 12 0: Non-secure mode 1: Secure mode |
| 11 | S11 | B'0 | R/W | Secure Mode Setting for Channel 11 0: Non-secure mode 1: Secure mode |
| 10 | S10 | B'0 | R/W | Secure Mode Setting for Channel 10 0: Non-secure mode 1: Secure mode |
| 9 | S9 | B'0 | R/W | Secure Mode Setting for Channel 9 0: Non-secure mode 1: Secure mode |
| 8 | S8 | B'0 | R/W | Secure Mode Setting for Channel 8 0: Non-secure mode 1: Secure mode |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|---|
| 7 | S7 | B'0 | R/W | Secure Mode Setting for Channel 7 0: Non-secure mode 1: Secure mode |
| 6 | S6 | B'0 | R/W | Secure Mode Setting for Channel 6 0: Non-secure mode 1: Secure mode |
| 5 | S5 | B'0 | R/W | Secure Mode Setting for Channel 5 0: Non-secure mode 1: Secure mode |
| 4 | S4 | B'0 | R/W | Secure Mode Setting for Channel 4 0: Non-secure mode 1: Secure mode |
| 3 | S3 | B'0 | R/W | Secure Mode Setting for Channel 3 0: Non-secure mode 1: Secure mode |
| 2 | S2 | B'0 | R/W | Secure Mode Setting for Channel 2 0: Non-secure mode 1: Secure mode |
| 1 | S1 | B'0 | R/W | Secure Mode Setting for Channel 1 0: Non-secure mode 1: Secure mode |
| 0 | S0 | B'0 | R/W | Secure Mode Setting for Channel 0 0: Non-secure mode 1: Secure mode |

44.2.4 DMA Secure Control Register Higher-Numbered Channels (DMASEC_H)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASEC_H is a 32-bit readable/writable register that controls the security attribute of each of the higher-numbered channels. Only the initiator in the secure mode can change the setting of this register.

Only secure access is allowed to registers of channels with the secure mode setting. The following registers are protected by the secure mode.

DMASAR, DMADAR, DMATCR, DMATSR, DMACHCR, DMATCRB, DMATSRB, DMACHCRB, DMARS, DMABUFCR, DMADPBASE, DMADPCR, DMAFIXSAR, DMAFIXDAR, and DMAFIXDPBASE

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | S31 | S30 | S29 | S28 | S27 | S26 | S25 | S24 | S23 | S22 | S21 | S20 | S19 | S18 | S17 | S16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | S31 | B'0 | R/W | Secure Mode Setting for Channel 31 0: Non-secure mode 1: Secure mode |
| 14 | S30 | B'0 | R/W | Secure Mode Setting for Channel 30 0: Non-secure mode 1: Secure mode |
| 13 | S29 | B'0 | R/W | Secure Mode Setting for Channel 29 0: Non-secure mode 1: Secure mode |
| 12 | S28 | B'0 | R/W | Secure Mode Setting for Channel 28 0: Non-secure mode 1: Secure mode |
| 11 | S27 | B'0 | R/W | Secure Mode Setting for Channel 27 0: Non-secure mode 1: Secure mode |
| 10 | S26 | B'0 | R/W | Secure Mode Setting for Channel 26 0: Non-secure mode 1: Secure mode |
| 9 | S25 | B'0 | R/W | Secure Mode Setting for Channel 25 0: Non-secure mode 1: Secure mode |
| 8 | S24 | B'0 | R/W | Secure Mode Setting for Channel 24 0: Non-secure mode 1: Secure mode |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|--|
| 7 | S23 | B'0 | R/W | Secure Mode Setting for Channel 23 0: Non-secure mode 1: Secure mode |
| 6 | S22 | B'0 | R/W | Secure Mode Setting for Channel 22 0: Non-secure mode 1: Secure mode |
| 5 | S21 | B'0 | R/W | Secure Mode Setting for Channel 21 0: Non-secure mode 1: Secure mode |
| 4 | S20 | B'0 | R/W | Secure Mode Setting for Channel 20 0: Non-secure mode 1: Secure mode |
| 3 | S19 | B'0 | R/W | Secure Mode Setting for Channel 19 0: Non-secure mode 1: Secure mode |
| 2 | S18 | B'0 | R/W | Secure Mode Setting for Channel 18 0: Non-secure mode 1: Secure mode |
| 1 | S17 | B'0 | R/W | Secure Mode Setting for Channel 17 0: Non-secure mode 1: Secure mode |
| 0 | S16 | B'0 | R/W | Secure Mode Setting for Channel 16 0: Non-secure mode 1: Secure mode |

44.2.5 DMA Operation Register for Lower-Numbered Channels (DMAOR_L)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMAOR_L is a 16-bit readable/writable register that enables DMA transfer on all lower-numbered channels and specifies the method used to determine the priority levels for all lower-numbered DMA channels. This register also indicates address errors.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---------|-----|---|---|---|---|---|--------|---|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | PR[1:0] | — | — | — | — | — | — | AE | — | DME |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R/(W)* | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|--------|---|
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9, 8 | PR[1:0] | B'00 | R/W | Priority Mode Select the method for setting the order of priority of channels when transfer requests for multiple channels arrive simultaneously. B'00: Fixed CH0 > CH1 > ... > CH11 > CH15 B'11: Round-robin priority Other than above: Setting prohibited |
| 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | AE | B'0 | R/(W)* | Address Error Flag Indicates that an address error interrupt occurred during DMA transfer. This bit is set under the following conditions: The value set in DMASAR or DMADAR does not fall on a boundary corresponding to the transfer size. The source or destination for transfer is in an invalid space. The source or destination for transfer is in module stop mode. If this bit is set, DMA transfer through the channel is not possible even if the DE bit is set to 1. To clear the AE bit, write 0 to the AE bit after reading 1 from it or clear the CAE bit for each channel for which it is set. Clearing the AE bit clears the channel address error bits for all channels. 0: An audio-DMAC address error interrupt is not present. [Clearing condition] Writing CAE = 0 after reading CAE = 1 1: An audio-DMAC address error interrupt being generated during DMA transfer. |
| 1 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|---|
| 0 | DME | B'0 | R/W | <p>DMA Master Enable</p> <p>Enables or disables DMA transfer on all channels. If the DME bit and the DE bit in DMACHCR are both set to 1, DMA transfer is enabled. At this time all AE bits in DMAOR_L must have the value 0. For DMA transfer on a channel to then proceed, the TE bit in DMACHCR for the channel must also have the value 0. Clearing this bit during transfer aborts transfer on all channels.</p> <p>0: Disables DMA transfers on all channels 1: Enables DMA transfers on all channels</p> |

Note: * Writing 0 is possible to clear the flag.

44.2.6 DMA Operation Register for Higher-Numbered Channels (DMAOR_U)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

DMAOR_U is a 16-bit readable/writable register that enables DMA transfer on all higher-numbered channels and specifies the method used to determine the priority levels for all higher-numbered DMA channels. This register also indicates address errors.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---------|-----|---|---|---|---|---|--------|---|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | PR[1:0] | — | — | — | — | — | — | AE | — | DME |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R/(W)* | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|--------|---|
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9, 8 | PR[1:0] | B'00 | R/W | Priority Mode Select the method for setting the order of priority of channels when transfer requests for multiple channels arrive simultaneously. B'00: Fixed CH16 > CH17 > ... > CH30 > CH31 [in the RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] B'11: Round-robin priority Other than above: Setting prohibited |
| 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | AE | B'0 | R/(W)* | Address Error Flag Indicates that an address error interrupt occurred during DMA transfer. This bit is set under the following conditions: The value set in DMASAR or DMADAR does not fall on a boundary corresponding to the transfer size. The source or destination for transfer is in an invalid space. The source or destination for transfer is in module stop mode. If this bit is set, DMA transfer through the channel is not possible even if the DE bit is set to 1. To clear the AE bit, write 0 to the AE bit after reading 1 from it or clear the CAE bit for each channel for which it is set. Clearing the AE bit clears the channel address error bits for all channels. 0: An audio-DMAC address error interrupt is not present. [Clearing condition] Writing CAE = 0 after reading CAE = 1 1: An audio-DMAC address error interrupt being generated during DMA transfer. |
| 1 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|--|
| 0 | DME | B'0 | R/W | <p>DMA Master Enable</p> <p>Enables or disables DMA transfer on all channels. If the DME bit and the DE bit in DMACHCR are both set to 1, DMA transfer is enabled. At this time all AE bits in DMAOR_U must have the value 0. For DMA transfer on a channel to then proceed, the TE bit in DMACHCR for the channel must also have the value 0. Clearing this bit during transfer aborts transfer on all channels.</p> <p>0: Disables DMA transfers on all channels</p> <p>1: Enables DMA transfers on all channels</p> |

44.2.7 DMA Channel Clear Register for Lower-Numbered Channels (DMACHCLR_L)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMACHCLR_L is a 32-bit writable register that initializes each of the lower-numbered channels.

When a bit of this register is set, the state of the corresponding channel is completely initialized.

This includes initialization of the following registers.

DMASAR, DMADAR, DMATCR, DMATSR, DMACHCR, DMATCRB, DMATSRB, DMACHCRB, DMARS, DMABUFGR, DMADPBASE, DMADPCR, DMAFIXSAR, DMAFIXDAR, and DMAFIXDPBASE

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CLR[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | W | W | W | W | W | W | W | W | W | W | W | W | W | W | W | W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | CLR[15:0] | All 0 | W | Writing to a bit leads to clearing of all registers for the corresponding channel. CLR[0] 0: Ignored 1: All registers for channel 0 are cleared. CLR[1] 0: Ignored 1: All registers for channel 1 are cleared. CLR[2] 0: Ignored 1: All registers for channel 2 are cleared. ... CLR[12] 0: Ignored 1: All registers for channel 12 are cleared. CLR[13] 0: Ignored 1: All registers for channel 13 are cleared. CLR[14] 0: Ignored 1: All registers for channel 14 are cleared. CLR[15] 0: Ignored 1: All registers for channel 15 are cleared. |

When writing to this register, confirm that the DE bit is set to 0.

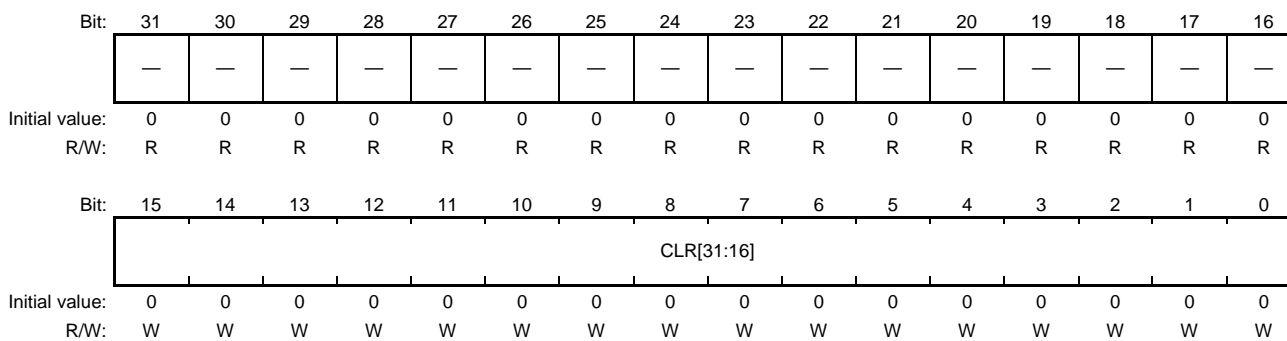
44.2.8 DMA Channel Clear Register for Higher-Numbered Channels (DMACHCLR_U)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

DMACHCLR_U is a 32-bit writable register that initializes each of the higher-numbered channels. When a bit of this register is set, the state of the corresponding channel is completely initialized.

This includes initialization of the following registers.

DMASAR, DMADAR, DMATCR, DMATSR, DMACHCR, DMATCRB, DMATSRB, DMACHCRB, DMARS, DMABUFGR, DMADPBASE, DMADPCR, DMAFIXSAR, DMAFIXDAR, and DMAFIXDPBASE



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | CLR[31:16] | All 0 | W | Writing to a bit leads to clearing of all registers for the corresponding channel. CLR[16] 0: Ignored 1: All registers for channel 16 are cleared. CLR[17] 0: Ignored 1: All registers for channel 17 are cleared. ... CLR[25] 0: Ignored 1: All registers for channel 25 are cleared. CLR[26] 0: Ignored 1: All registers for channel 26 are cleared. [in the RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] CLR[27] 0: Ignored 1: All registers for channel 27 are cleared. [in the RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] CLR[28] 0: Ignored 1: All registers for channel 28 are cleared. [in the RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] CLR[29] 0: Ignored 1: All registers for channel 29 are cleared. [in the RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] CLR[30] 0: Ignored 1: All registers for channel 30 are cleared. [in the RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] CLR[31] 0: Ignored 1: All registers for channel 31 are cleared. [in the RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] When writing to this register, confirm that the DE bit is set to 0. |

44.2.9 DPRAM Secure Control Register for Lower-Numbered Channels (DMADPSEC_L)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMADPSEC_L is a 32-bit readable/writable register that controls the security attribute of the descriptor memory. Only the initiator in the secure mode can change the setting of this register.

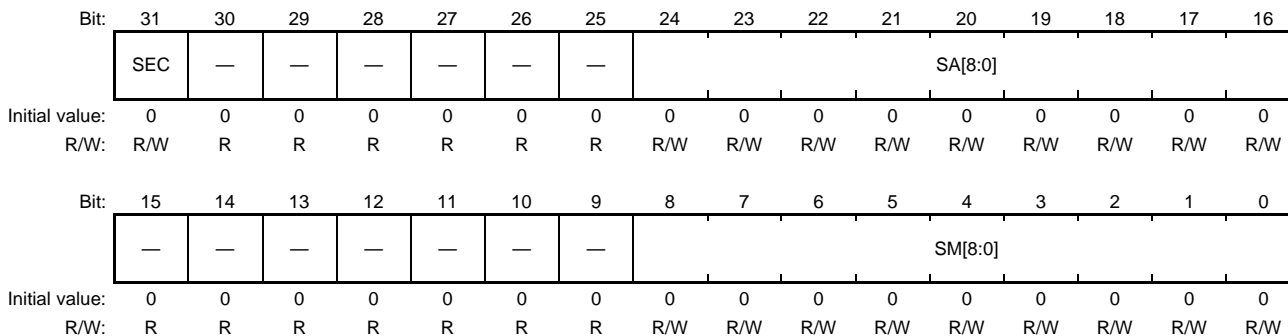
| | | | | | | | | | | | | | | | | |
|----------------|-----|----|----|----|----|----|----|---------|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SEC | — | — | — | — | — | — | SA[8:0] | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | SM[8:0] | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | SEC | B'0 | R/W | Security Attribute Setting for Descriptor Memory Specifies the security attribute of the address space used for the descriptor memory. 0: Non-secure 1: Secure |
| 30 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 to 16 | SA[8:0] | H'000 | R/W | Security Attribute Setting for Base Address of Descriptor Memory Specify the base address of the descriptor memory to be assigned the security attribute. H'000: H'A000 H'001: H'A004 ... H'1FF: H'A7FC |
| 15 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 to 0 | SM[8:0] | H'000 | R/W | Security Attribute Setting for Base Address Mask of Descriptor Memory Specify the security attribute base address mask of the descriptor memory. The range of memory to be assigned the security attribute is specified by this register. See Figure 44.4. |

44.2.10 DPRAM Secure Control Register for Higher-Numbered Channels (DMADPSEC_U)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

DMADPSEC_U is a 32-bit readable/writable register that controls the security attribute of the descriptor memory. Only the initiator in the secure mode can change the setting of this register.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | SEC | B'0 | R/W | Security Attribute Setting for Descriptor Memory Specifies the security attribute of the address space used for the descriptor memory. 0: Non-secure 1: Secure |
| 30 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 to 16 | SA[8:0] | H'000 | R/W | Security Attribute Setting for Base Address of Descriptor Memory Specify the base address of the descriptor memory to be assigned the security attribute. H'000: H'A000 H'001: H'A004 ... H'1FF: H'A7FC |
| 15 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 to 0 | SM[8:0] | H'000 | R/W | Security Attribute Setting for Base Address Mask of Descriptor Memory Specify the security attribute base address mask of the descriptor memory. The range of memory to be assigned the security attribute is specified by this register. See Figure 44.4. |

44.2.11 DMA Source Address Registers 0 to 31 (DMASAR_0 to DMASAR_31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

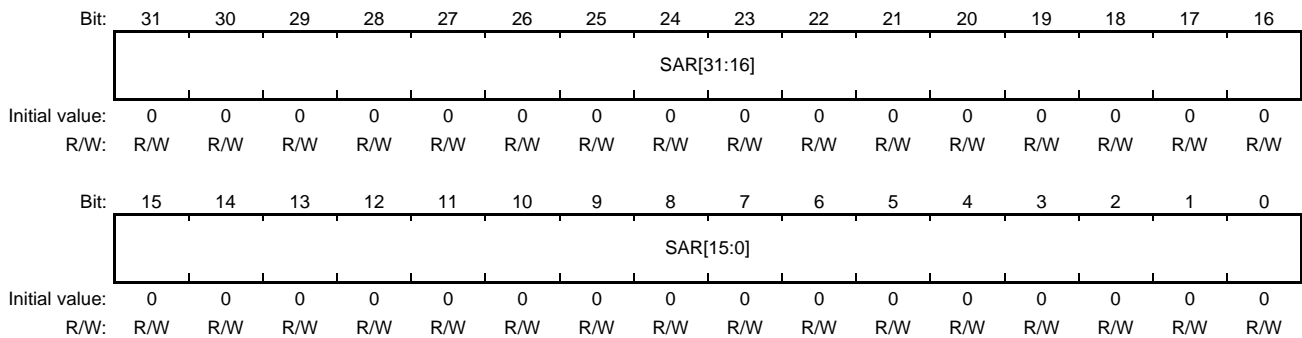
Note: The availability of channels depends on the product as follows.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)

RZ/G2E: 16 channels (channels 0 to 15)

DMASAR is a 32-bit readable/writable register that specifies the source address of a DMA transfer. While a DMA transfer is in progress, this register indicates the next source address.

When the address mode is incrementation, sources in memory only have byte boundaries. For details, refer to Table 44.4.



44.2.12 DMA Destination Address Registers 0 to 31 (DMADAR_0 to DMADAR_31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: The availability of channels depends on the product as follows.
 RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)
 RZ/G2E: 16 channels (channels 0 to 15)

DMADAR is 32-bit readable/writable register that specify the destination address of a DMA transfer. While a DMA transfer is in progress, this register indicates the next destination address. When the address mode is incremented, destinations in memory only have byte boundaries. For details, refer to Table 44.4.

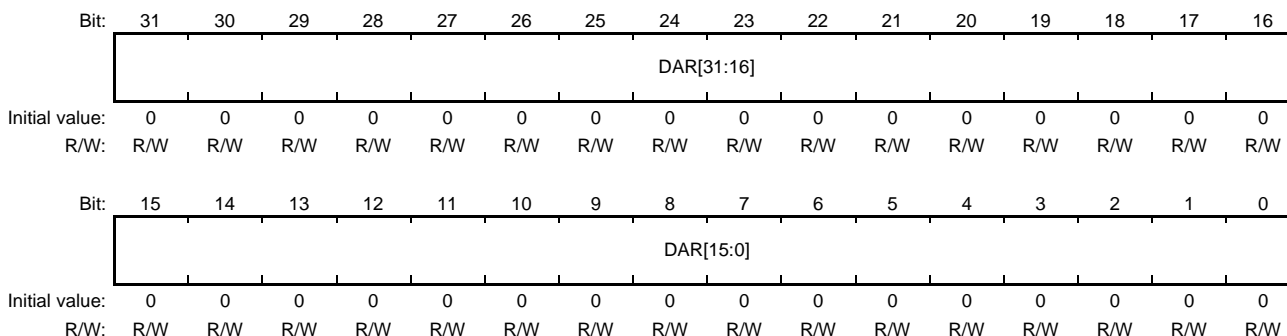


Table 44.4 SAR/DAR Address Restriction

| Resource Selection | Address Mode | Restriction |
|--|----------------|---|
| Auto request | Incrementation | No restriction (byte boundaries) |
| | Others | Boundary corresponding to the DMA transfer size |
| On-chip peripheral module request Transmission/DAR, Reception/SAR | All | Boundary corresponding to the DMA transfer size |
| On-chip peripheral module request Transmission/SAR, Reception/DAR | Incrementation | No restriction (byte boundaries) |
| | Others | Boundary corresponding to the DMA transfer size |

44.2.13 DMA Transfer Count Registers 0 to 31 (DMATCR_0 to DMATCR_31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: The availability of channels depends on the product as follows.

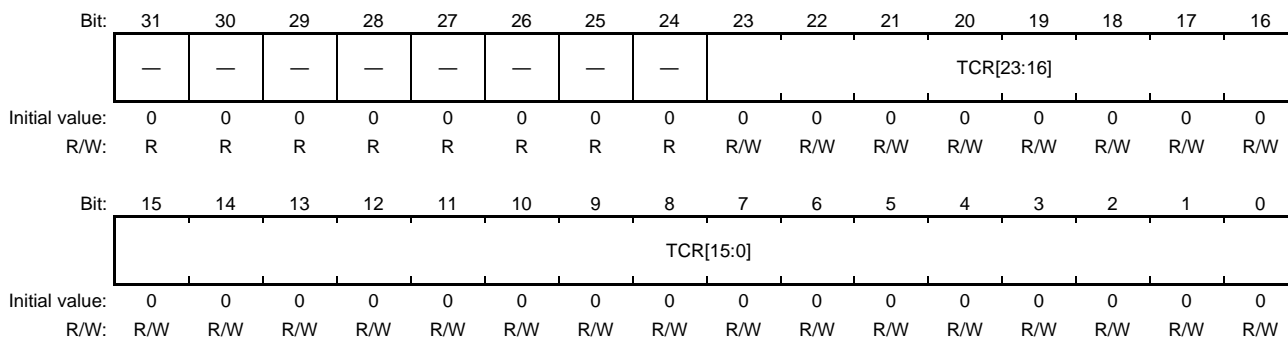
RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)

RZ/G2E: 16 channels (channels 0 to 15)

DMATCR is a 32-bit readable/writable register that specifies the number of rounds of DMA transfer. The number of rounds of DMA transfer is 1 when the setting is H'0000_0001 and 16,777,215 (the maximum) when the setting is H'00FF_FFFF. During a DMA transfer, this register indicates the remaining number of rounds of transfer.

The audio-DMAC includes independent data buffers for reading and writing. Therefore, the read transfer counter and write transfer counter have different values. This register indicates the counter value used in reading.

The eight higher-order bits of DMATCR are always read as 0, and the write value should always be 0.



44.2.14 DMA Transfer Count Registers B_0 to 31 (DMATCRB_0 to DMATCRB_31)

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

Note: The availability of channels depends on the product as follows.

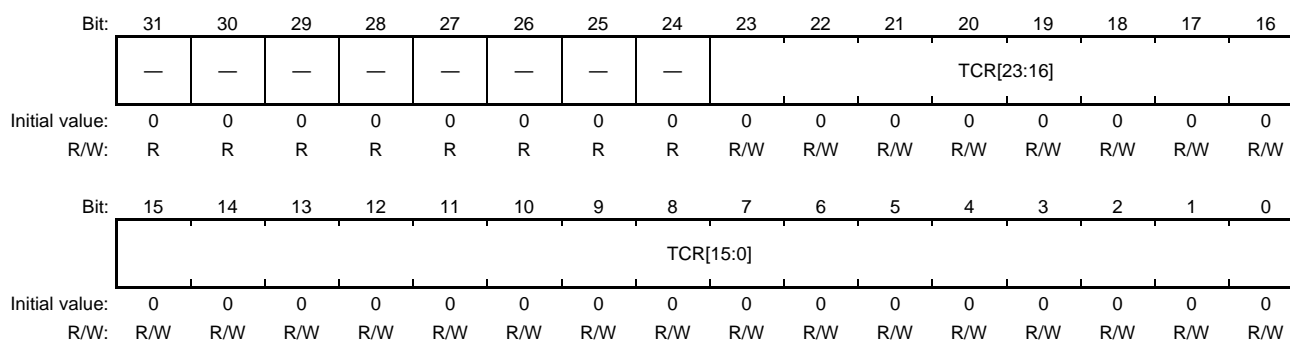
RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)

RZ/G2E: 16 channels (channels 0 to 15)

DMATCRB is a 32-bit readable/writable register that specifies the number of rounds of DMA transfer. The number of rounds of DMA transfer is 1 when the setting is H'0000_0001 and 16,777,215 (the maximum) when the setting is H'00FF_FFFF. During a DMA transfer, this register indicates the remaining number of rounds of transfer.

The audio-DMAC includes independent data buffers for reading and writing. Therefore, the read transfer counter and write transfer counter have different values. This register indicates the counter value used in writing.

The eight higher-order bits of DMATCRB are always read as 0, and the write value should always be 0.

**44.2.15 DMA Transfer Size Registers 0 to 31 (DMATSR_0 to DMATSR_31)**

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

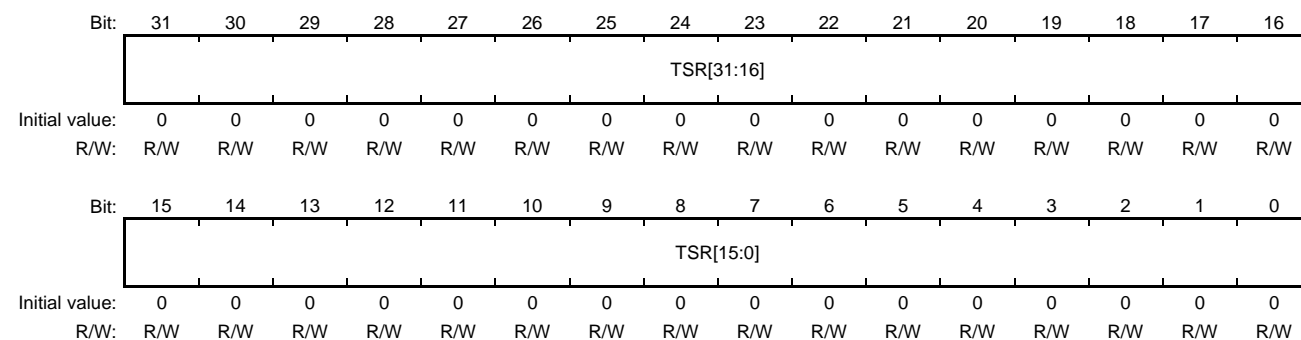
Note: The availability of channels depends on the product as follows.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)

RZ/G2E: 16 channels (channels 0 to 15)

DMATSR is a 32-bit readable/writable register that specifies a total amount of memory to be transferred. The total size of DMA transfer is 1 byte when the setting is H'0000_0001, 4,294,967,295 bytes when the setting is H'FFFF_FFFF, and 4,294,967,296 bytes (the maximum) when the setting is H'0000_0000. During a DMA transfer, this register indicates the remaining amount of memory to be transferred. This register is used in total size transmission.

The audio-DMAC includes independent data buffers for reading and writing. Therefore, reading and writing will have different transfer sizes. This register indicates the value of the read transfer size.

**44.2.16 DMA Transfer Size Registers B_0 to 31 (DMATSRB_0 to DMATSRB_31)**

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

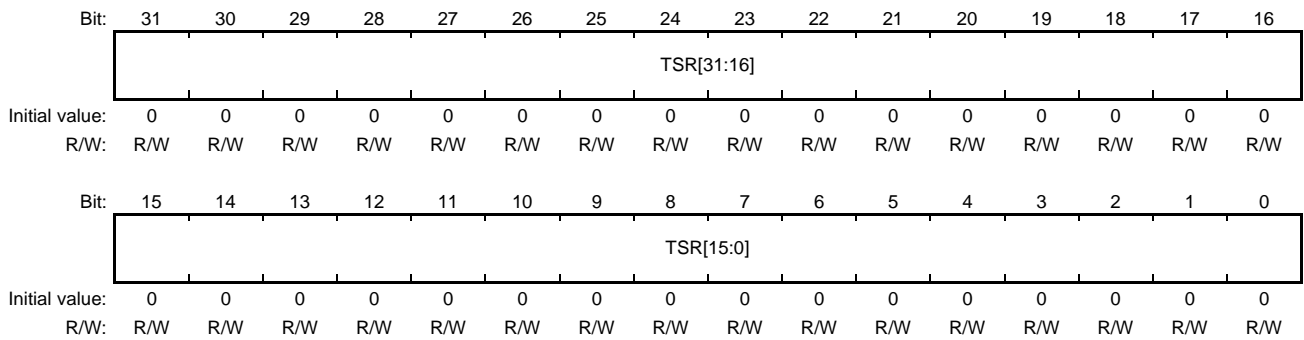
Note: The availability of channels depends on the product as follows.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)

RZ/G2E: 16 channels (channels 0 to 15)

DMATSRB is a 32-bit readable/writable register that specifies a total amount of memory to be transferred. The total size of DMA transfer is 1 byte when the setting is H'0000_0001, 4,294,967,295 bytes when the setting is H'FFFF_FFFF, and 4,294,967,296 bytes (the maximum) when the setting is H'0000_0000. During a DMA transfer, this register indicates the remaining amount of memory to be transferred. This register is used in total size transmission.

The audio-DMAC includes independent data buffers for reading and writing. Therefore, reading and writing will have different transfer sizes. This register indicates the value of the write transfer size.



44.2.17 DMA Channel Control Registers 0 to 31 (DMACHCR_0 to DMACHCR_31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: The availability of channels depends on the product as follows.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)

RZ/G2E: 16 channels (channels 0 to 15)

DMACHCR is a 32-bit readable/writable register that controls the DMA transfer mode.

| | | | | | | | | | | | | | | | | |
|----------------|---------|------|----------|-----|----------|-----|-----|-----|-----|---------|---------|-----|--------|-----|--------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CAE | CAIE | DPM[1:0] | | RPT[2:0] | | — | — | DPB | TS[3:2] | | DSE | DSIE | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/(W)* | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/(W)* | R/W | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DM[1:0] | | SM[1:0] | | RS[3:0] | | | — | — | — | TS[1:0] | | IE | TE | DE | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/(W)* | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|--------|--|
| 31 | CAE | B'0 | R/(W)* | <p>Channel Address Error Flag</p> <p>Indicates that an address error interrupt occurred during DMA transfer.</p> <p>This bit is set under the following conditions:</p> <ul style="list-style-type: none"> The value set in DMASAR or DMADAR does not fall on a boundary corresponding to the transfer size. The source or destination for transfer is in an invalid space. The source or destination for transfer is in module stop mode. <p>If this bit is set, DMA transfer through the channel is not possible even if the DE bit is set to 1.</p> <p>To clear the CAE bit, write 0 to the CAE bit after reading 1 from it or clear the AE bit in DMAOR.</p> <p>Clearing the CAE bit clears the channel address error bits for all channels.</p> <p>0: An audio-DMAC address error interrupt is not present. [Clearing condition] Writing CAE = 0 after reading CAE = 1</p> <p>1: An audio-DMAC address error interrupt being generated during DMA transfer.</p> |
| 30 | CAIE | B'0 | R/W | <p>Channel Address Error Interrupt Enable</p> <p>Enables or disables the generation of interrupt requests for the CPU when address errors occur. When the CAIE bit is set to 1, if the CAE bit is also set, an interrupt (DEI 0 to 25) from the corresponding channel will be generated for the CPU in response to address errors.</p> <p>Note: An address error interrupt (DADERR) is also asserted simultaneously. See section 19, Interrupt Controller (INTC-AP) for more details.</p> <p>0: Interrupt requests are disabled. 1: Interrupt requests are enabled.</p> |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|--------|--|
| 29, 28 | DPM[1:0] | B'00 | R/W | <p>Operating Mode of Descriptor Memory</p> <p>Enable or disable the descriptor memory and specify its operating mode.</p> <p>B'00: Disabled (normal use)</p> <p>B'01: Enabled (normal mode)</p> <p>B'10: Enabled (repeat mode)</p> <p>B'11: Enabled (read-out interrupt mode, infinite repeat mode)</p> |
| 27 to 25 | RPT[2:0] | B'000 | R/W | <p>Descriptor Setting Update</p> <p>Specify the parameters to be updated from the descriptor memory.</p> <p>RPT[2]: Enables or disables updating of the source address register</p> <p>RPT[1]: Enables or disables updating of the destination address register</p> <p>RPT[0]: Enables or disables updating of the transfer count register</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 24, 23 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 22 | DPB | B'0 | R/W | <p>Descriptor Start</p> <p>Specifies configuration to be loaded when transfer under control of the descriptor memory begins.</p> <p>This bit is cleared after the descriptor memory is read.</p> <p>0: Processing starts with the values in DMASAR, DMADAR, and DMATCR.</p> <p>1: Processing starts after the first set of descriptors is read out.</p> |
| 21, 20 | TS[3:2] | B'00 | R/W | <p>DMA Transfer Size</p> <p>In combination with TS[1:0], these bits specify the DMA transfer size. When the transfer source or transfer destination is a register of an on-chip peripheral module for which a transfer size is specified, be sure to select the specified transfer size. For the transfer source or destination address specified by DMASAR or DMADAR, an appropriate boundary address should be set according to the transfer data size.</p> <p>TS[3:2] + TS[1:0] (“+” here indicates concatenation, not addition)</p> <p>B'0000: Transfer is in byte units.</p> <p>B'0001: Transfer is in word (2-byte) units.</p> <p>B'0010: Transfer is in longword (4-byte) units.</p> <p>B'0011: Transfer is in 16-byte units.</p> <p>B'0100: Transfer is in 32-byte units.</p> <p>B'0101: Transfer is in 64-byte units.</p> <p>B'0111: Transfer is in 8-byte units.</p> <p>Other than above: Setting prohibited</p> |
| 19 | DSE | B'0 | R/(W)* | <p>Descriptor Stage End</p> <p>When the DSIE bit is set to 1 and the descriptor memory is enabled, the DSE bit is set to 1 on completion of the DMA transfer. This bit is not set when the DPM bit is set to 0 (descriptors are disabled). To clear the DSE bit, start by reading it as 1, and then write 0 to the bit.</p> <p>0: DMA transfer is still running or has been aborted.</p> <p>1: Transfer under the control of one stage of the descriptor memory has been completed.</p> |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|--------|----------|---------------|-----|---|
| 18 | DSIE | B'0 | R/W | <p>Descriptor Stage End Interrupt Enable</p> <p>Specifies whether an interrupt request is generated for the CPU on completion of transfer under the control of one stage of the descriptor memory. When this bit is set to 1, an interrupt (DEI) is generated for the CPU whenever the DSE is set to 1.</p> <p>0: Interrupt requests are disabled.</p> <p>1: Interrupt requests are enabled.</p> |
| 17, 16 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 15, 14 | DM[1:0] | B'00 | R/W | <p>Destination Address Mode</p> <p>Specify whether the DMA destination address is incremented, fixed, or decremented. The unit of transfer (transfer size) determines the size of the increment.</p> <p>B'00: Destination address is fixed.</p> <p>B'01: Destination addresses are incremented.</p> <ul style="list-style-type: none"> + 1 when transfer is in byte units. + 2 when transfer is in word units. + 4 when transfer is in longword units. + 8 when transfer is in 8-byte units. + 16 when transfer is in 16-byte units. + 32 when transfer is in 32-byte units. + 64 when transfer is in 64-byte units. <p>B'10: Destination addresses are decremented.</p> <ul style="list-style-type: none"> – 1 when transfer is in byte units. – 2 when transfer is in word units. – 4 when transfer is in longword units. <p>Setting prohibited when transfer is in 8-, 16-, 32-, or 64-byte units.</p> <p>B'11: Setting Prohibited</p> |
| 13, 12 | SM[1:0] | B'00 | R/W | <p>Source Address Mode</p> <p>Specify whether the DMA source address is incremented, fixed, or decremented. The unit of transfer (transfer size) determines the size of the increment.</p> <p>B'00: Source address is fixed.</p> <p>B'01: Source addresses are incremented.</p> <ul style="list-style-type: none"> + 1 when transfer is in byte units. + 2 when transfer is in word units. + 4 when transfer is in longword units. + 8 when transfer is in 8-byte units. + 16 when transfer is in 16-byte units. + 32 when transfer is in 32-byte units. + 64 when transfer is in 64-byte units. <p>B'10: Source addresses are decremented.</p> <ul style="list-style-type: none"> – 1 when transfer is in byte units. – 2 when transfer is in word units. – 4 when transfer is in longword units. <p>Setting prohibited when transfer is in 8-, 16-, 32-, or 64-byte units.</p> <p>B'11: Setting Prohibited</p> |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|----------|---------------|--------|--|
| 11 to 8 | RS[3:0] | B'0000 | R/W | <p>Resource Selection</p> <p>Specify the source of transfer requests. Only change the transfer request source while the DMA enable bit (DE) is set to 0.</p> <p>B'0100: Auto request</p> <p>B'1000: Source is selected by the DMA extended resource selector.</p> <p>Other than above: Settings prohibited</p> |
| 7 to 5 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 4, 3 | TS[1:0] | B'00 | R/W | <p>DMA Transfer Size</p> <p>See the description of TS[3:2] (bits 21 and 20).</p> |
| 2 | IE | B'0 | R/W | <p>Interrupt Enable</p> <p>Specifies whether or not an interrupt request is generated for the CPU on completion of DMA transfer. When this bit is set to 1, an interrupt request (DEI) for the CPU is generated whenever the TE bit is set to 1.</p> <p>0: Interrupt request is disabled.</p> <p>1: Interrupt request is enabled.</p> |
| 1 | TE | B'0 | R/(W)* | <p>Transfer End Flag</p> <p>When the descriptor memory is not in use, the TE bit is set to 1 when DMATCR becomes 0 on completion of the DMA transfer.</p> <p>When the descriptor memory is in use, the TE bit is set to 1 on completion of all transfers set up in the descriptor memory. The TE bit is not set to 1 in the following cases.</p> <ul style="list-style-type: none"> DMA transfer ends due to a DMA address error before DMATCR becomes 0. DMA transfer is aborted by clearing the DE and DME bits in DMAOR. <p>To clear the TE bit, start by reading it as 1, and then write 0 to it.</p> <p>When the TE bit is set to 1, transfer is not possible even if the DE bit is set to 1.</p> <p>0: DMA transfer is in progress or was aborted [Clearing condition] Writing of 0 after reading of 1</p> <p>1: DMA transfer ended on the specified count (TCR = 0)</p> |
| 0 | DE | B'0 | R/W | <p>DMA Enable</p> <p>Enables or disables DMA transfer. In the auto request mode, a DMA transfer is started by setting the DE and DME bits in DMAOR to 1. At this time, the setting of both the AE and TE bits in DMAOR must be 0. In a peripheral module request, a DMA transfer starts if the transfer request is generated by the selected device or on-chip peripheral module after setting the DE and DME bits to 1. In this case too, the settings of both the TE and AE bits must be 0. Clearing the DE bit to 0 aborts all DMA transfer.</p> <p>Note: Ensure that the setting of the DE bit is actually 0 after clearing it.</p> <p>0: DMA transfer is disabled.</p> <p>1: DMA transfer is enabled.</p> |

Note: * Writing 0 is possible to clear the flag.

44.2.18 DMA Channel Control Register B_0 to 31 (DMACHCRB_0 to DMACHCRB_31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: The availability of channels depends on the product as follows.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)

RZ/G2E: 16 channels (channels 0 to 15)

DMACHCRB is a 32-bit readable/writable register that controls the DMA transfer mode.

| | | | | | | | | | | | | | | | | |
|----------------|-----------|-----|-----|-----|-----|-----|-----|-----|-----------|-----|-----|----------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | DCNT[7:0] | | | | | | | | DPTR[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DRST | — | — | — | — | — | — | DTS | SLM[3:0] | | | PRI[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | W | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|-----------|---------------|-----|---|
| 31 to 24 | DCNT[7:0] | H'00 | R/W | Number of Stages of Descriptor Memory Specify the number of stages of the descriptor memory as DCNT + 1. When the descriptor memory is enabled, a transfer end (TE) interrupt is only generated for the CPU on completion of transfer under control of the specified number of stages. |
| 23 to 16 | DPTR[7:0] | H'00 | R | Descriptor Pointer This bit indicates the pointer to the next descriptor to be read. It is cleared to 0 when the last descriptor of the number of stages specified by DCNT[7:0] is read. It is also cleared to 0 when 1 is written to DRST. |
| 15 | DRST | B'0 | W | Descriptor Reset Resets the descriptor pointer. Before the descriptor memory is used, the pointer must be reset by writing 1 to this bit. This bit is always read as 0. |
| 14 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | DTS | B'0 | R/W | Total Size Transmission under Descriptor Control This bit is only effective when total size transmission is selected. 0: The TCR fields of the descriptors are used as transfer count settings. 1: The TCR fields of the descriptors are used as total size settings. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|--------|----------|---------------|-----|---|
| 7 to 4 | SLM[3:0] | B'0000 | R/W | <p>DMA Transfer Low-Speed Mode</p> <p>Specify the number of cycles of the AXI-bus clock (256.66 MHz) for the DMA transfer. One round of DMA transfer is executed in the number of cycles of the clock specified by this bit.</p> <p>B'0000: Normal mode</p> <p>B'1000: On round in 256 cycles of the clock.</p> <p>B'1001: On round in 512 cycles of the clock.</p> <p>B'1010: On round in 1024 cycles of the clock.</p> <p>:</p> <p>B'1111: On round in 32768 cycles of the clock.</p> <p>Other than above: Setting prohibited</p> |
| 3 to 0 | PRI[3:0] | B'0000 | R/W | <p>Channel Request Priority Setting</p> <p>These bits should be written by 0.</p> |

44.2.19 DMA Buffer Control Registers 0 to 31 (DMABUFCR_0 to DMABUFCR_31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: The availability of channels depends on the product as follows.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)

RZ/G2E: 16 channels (channels 0 to 15)

DMABUFCR is a 32-bit readable/writable register that controls the upper limit on buffer size in and burst unit for the SDRAM.

Use this register when the upper limit on buffering requires control.

| | | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----------|----------|-----|-----|-----|-----|-----|-----|-----|-----|---|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | | |
| | — | — | — | — | — | — | — | MBU[8:0] | | | | | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | |
| | — | — | — | — | — | — | ULB[9:0] | | | | | | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|---|
| 31 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 to 16 | MBU[8:0] | H'080 | R/W | Maximum Burst Unit for SDRAM This register is only effective for SDRAM access, and everything other than that is under control of the transfer size (unit). Settings bigger than ULB are prohibited. Power-of-two settings are recommended. Maximum value is 256 (bytes). |
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 to 0 | ULB[9:0] | H'100 | R/W | Upper Limit on Buffer Size This register controls the upper limit value for buffering. Power-of-two settings are recommended. Maximum value is 512 (bytes). |

44.2.20 DMA Extended Resource Selectors 0 to 31 (DMARS_0 to DMARS_31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: The availability of channels depends on the product as follows.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)

RZ/G2E: 16 channels (channels 0 to 15)

DMARS is a 16-bit readable/writable register that specifies the on-chip peripheral module to be the source of the DMA transfer request for the given channel. DMARS_0 specifies the source for channel 0, DMARS_1 specifies the source for channel 1 and so on.

When bits MID and RID are set to a value other than the values listed in Table 44.5, the operation of this LSI is not guaranteed. Transfer requests from the source selected in DMARS are only valid when the resource selection bits (RS[3:0]) in DMACHCR have been set to B'1000. Otherwise, even if DMARS has been set, requests from the corresponding transfer request source are not accepted.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|----------|-----|-----|-----|-----|-----|----------|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | MID[5:0] | | | | | | RID[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

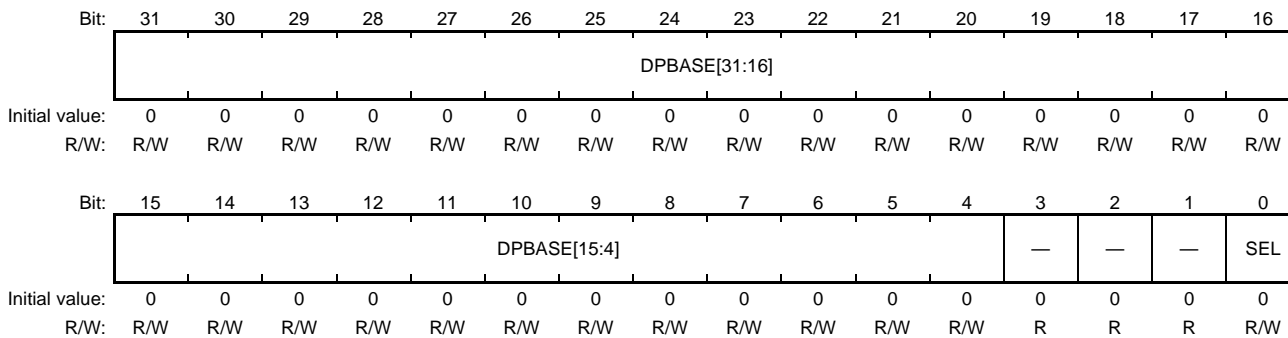
| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|----------|---------------|-----|--|
| 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 2 | MID[5:0] | All 0 | R/W | DMA Request Source Adoption MID[5] to MID[0] (MID) See Table 44.5. |
| 1, 0 | RID[1:0] | B'00 | R/W | DMA Request Source Adoption RID[5] and RID[0] (RID) See Table 44.5. |

44.2.21 DMA Descriptor Base Address Registers 0 to 31 (DMADPBASE_0 to DMADPBASE_31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: The availability of channels depends on the product as follows.
 RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)
 RZ/G2E: 16 channels (channels 0 to 15)

DMADPBASE specifies the base address of the descriptor memory. The address range of the descriptor memory is specified by setting this register.



| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|--------------|---------------|-----|---|
| 31 to 4 | DPBASE[31:4] | All 0 | R/W | Base Address of Descriptor Memory Setting example: When Built-in memory is used, [Audio-DMAC Lower]: H'EC70 A00 to H'EC70 A7F [Audio-DMAC Higher]: H'EC72 A00 to H'EC72 A7F When External memory is used, Other memory area on a 16-byte boundary |
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | SEL | B'0 | R/W | Descriptor Memory Selection Selects the memory to be used as descriptor memory. 0: Setting Prohibited 1: Built-in memory or External memory is used. |

44.2.22 DMA Descriptor Control Registers 0 to 31 (DMADPCR_0 to DMADPCR_31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: The availability of channels depends on the product as follows.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)

RZ/G2E: 16 channels (channels 0 to 15)

DMADPCR is a 32-bit readable/writable register that controls the timing with which interrupts are output in read-out interrupt mode (descriptor mode 3).

| | | | | | | | | | | | | | | | | |
|----------------|-----------|-----|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | DIPT[7:0] | | | | | | | | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|-----------|---------------|-----|--|
| 31 to 24 | DIPT[7:0] | H'00 | R/W | Descriptor Read-out Interrupt Pointer The number of stages for which descriptor read-out interrupts are generated in descriptor mode 3. DIPT + 1 specifies the number of descriptor stages. |
| 23 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

44.2.23 DMA Fixed Source Address Registers 0 to 31 (DMAFIXSAR_0 to DMAFIXSAR_31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: The availability of channels depends on the product as follows.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)

RZ/G2E: 16 channels (channels 0 to 15)

DMAFIXSAR is a 32-bit readable/writable register that specifies the most significant 8 bits of the 40-bit source address for a DMA transfer.

This register is not incremented by carrying when DMASAR overflows. And when uTLB function is effective, this register is invalid.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | SAR[39:32] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

44.2.24 DMA Fixed Destination Address Registers 0 to 31 (DMAFIXDAR_0 to DMAFIXDAR_31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: The availability of channels depends on the product as follows.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)

RZ/G2E: 16 channels (channels 0 to 15)

DMAFIXDAR is a 32-bit readable/writable register that specifies the most significant 8 bits of the 40-bit destination address for a DMA transfer.

This register is not incremented by carrying when DMADAR overflows. And when uTLB function is effective, this register is invalid.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | DAR[39:32] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

44.2.25 DMA Fixed Descriptor Base Address Registers 0 to 31 (DMAFIXDPBASE_0 to DMAFIXDPBASE_31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Note: The availability of channels depends on the product as follows.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: 32 channels (channels 0 to 31)

RZ/G2E: 16 channels (channels 0 to 15)

DMAFIXDPBASE is a 32-bit readable/writable register that specifies the most significant 8 bits of the 40-bit descriptor base address for a DMA transfer. And when uTLB function is effective, this register is invalid.

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------------|-----|-----|-----|-----|-----|-----|-----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | DPBASE[39:32] | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |

44.2.26 Descriptor Memory (DescriptorMEM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

See section 44.3.4, Descriptor Memory.

44.2.27 Secure function Secure Status register for channel 0 to 15 (DMASES_L)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASES_L is a 32-bit readable/writable register that contain error status of secure function of channel 0 to 15.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | Error |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | Error | B'0 | R/W | Error status of channel 0 to 15 Read: 0: No Error detected. 1: Error detected. Write*: 0: No operation 1: Clear DMASES_L, DMASEMID_L, DMASEDDR_L register. |

44.2.28 Secure function Secure Status register for channel 16 to 31 (DMASES_U)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

DMASES_U is a 32-bit readable/writable register that contain error status of secure function of channel 16 to 31.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | Error |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | Error | B'0 | R/W | Error status of channel 16 to 31 Read: 0: No Error detected. 1: Error detected. Write*: 0: No operation 1: Clear DMASES_U, DMASEMID_U, DMASEDDR_U register. |

44.2.29 Secure function Salve Error Address register for channel 0 to 15 (DMASEDDR_L)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASEDDR_L is a 32-bit readable register that contain first error address when access to channel 0 to 15 was denied.

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | EA[31:16] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | EA[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|----------|---------------|-----|--|
| 31 to 0 | EA[31:0] | All 0 | R | First error address when access to channel 0 to 15 was denied. |

44.2.30 Secure function Salve Error Address register for channel 16 to 31 (DMASEDDR_U)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

DMASEDDR_U is a 32-bit readable register that contain first error address when access to channel 16 to 31 was denied.

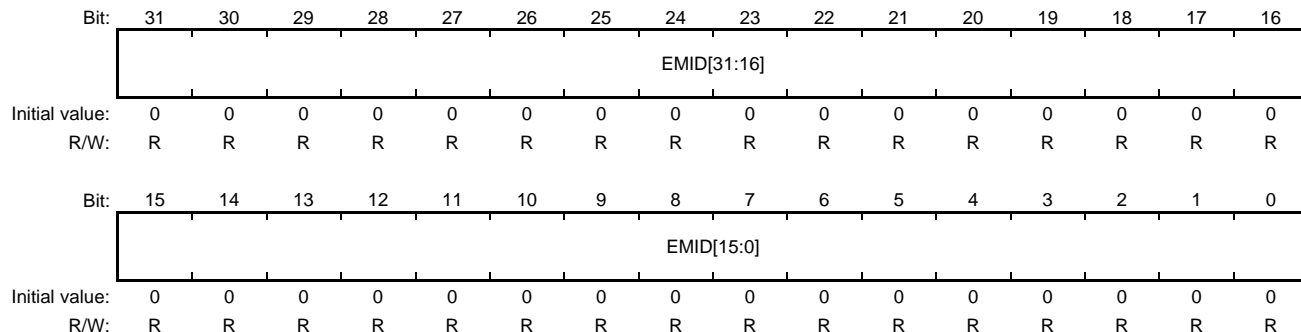
| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | EA[31:16] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | EA[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|----------|---------------|-----|---|
| 31 to 0 | EA[31:0] | All 0 | R | First error address when access to channel 16 to 31 was denied. |

44.2.31 Secure function Error Master ID register for channel 0 to 15 (DMASEMID_L)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DMASEMID_L is a 32-bit readable register that contain first Master ID when access to channel 0 to 15 was denied.

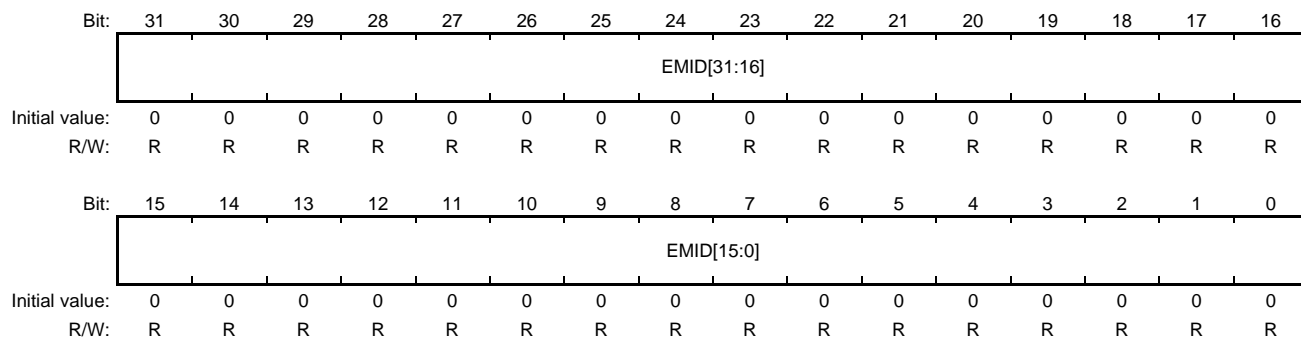


| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|------------|---------------|-----|--|
| 31 to 0 | EMID[31:0] | All 0 | R | First Master ID when access to channel 0 to 15 was denied. |

44.2.32 Secure function Error Master ID register for channel 16 to 31 (DMASEMID_U)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

DMASEMID_U is a 32-bit readable register that contain first Master ID when access to channel 16 to 31 was denied.



| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|------------|---------------|-----|---|
| 31 to 0 | EMID[31:0] | All 0 | R | First Master ID when access to channel 16 to 31 was denied. |

44.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

When there is a DMA transfer request, the DMAC starts the transfer according to the predetermined channel priority, when the transfer end conditions are satisfied, it ends the transfer. Transfers can be requested in two modes: auto request and on-chip peripheral module request. The bus mode can be selected from normal speed mode and slow speed mode).

44.3.1 DMA Transfer Requests

Most commonly, DMA transfer requests are generated by either the source or destination for transfer, but they can also be generated by on-chip peripheral modules that are neither the source nor the destination. Transfers can be requested in two modes: auto request, and on-chip peripheral module request. The request mode is selected for each channel by DMARS.

(1) Auto-Request Mode

When there is no transfer request signal from an external source, as in a memory-to-memory transfer or a transfer between memory and an on-chip peripheral module unable to request a transfer, auto-request mode allows the audio-DMAC to automatically generate a transfer request signal internally. When the DE bit in DMACHCR and the DME bit in DMAOR are set to 1 for the target channel, the transfer begins so long as the CAE bit in DMACHCR is 0.

(2) On-Chip Peripheral Module Request Mode

This mode is in case of CHCR.RS = 1000.

In this mode, a transfer is performed at the transfer request signal of an on-chip peripheral module. The source (on-chip peripheral module) of the DMA transfer request is specified by DMARS.

When this mode is selected, if the DMA transfer is enabled (DE = 1, DME = 1, TE = 0, CAE = 0), a transfer is performed upon the input of a transfer request signal.

When a transmit data empty transfer request of the SCIF is set as the transfer request, the transfer destination must be the SCIF's transmit data register. Likewise, when receive data full transfer request of the SCIF is set as the transfer request, the transfer source must be the SCIF's receive data register. These conditions also apply to the other on-chip peripheral modules.

The number of the receive FIFO triggers can be set as a transfer request depending on an on-chip peripheral module. Data needs to be read after the DMA transfer is ended, because data may be left in the receive FIFO when the receive FIFO trigger condition is not satisfied.

Table 44.5 Selecting On-Chip Peripheral Module Request Modes

| DMARS MID + RID | DMA Transfer Request Source | DMA Transfer Request Signal | Source | Destination | Second Generation RZ/G Series Products | | | |
|--------------------|--------------------------------|--------------------------------|-----------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| H'85 | SCU0 (SRC0in_BUSIF) | Transmit | Arbitrary | — | √ | √ | √ | √ |
| H'87 | SCU1 (SRC1in_BUSIF) | Transmit | Arbitrary | — | √ | √ | √ | √ |
| H'89 | SCU2 (SRC2in_BUSIF) | Transmit | Arbitrary | — | √ | √ | √ | √ |
| H'8B | SCU3 (SRC3in_BUSIF) | Transmit | Arbitrary | — | √ | √ | √ | √ |
| H'8D | SCU4 (SRC4in_BUSIF) | Transmit | Arbitrary | — | √ | √ | √ | √ |
| H'8F | SCU5 (SRC5in_BUSIF) | Transmit | Arbitrary | — | √ | √ | √ | √ |
| H'91 | SCU6 (SRC6in_BUSIF) | Transmit | Arbitrary | — | √ | √ | √ | √ |
| H'93 | SCU7 (SRC7in_BUSIF) | Transmit | Arbitrary | — | √ | √ | √ | √ |
| H'95 | SCU8 (SRC8in_BUSIF) | Transmit | Arbitrary | — | √ | √ | √ | √ |
| H'97 | SCU9 (SRC9in_BUSIF) | Transmit | Arbitrary | — | √ | √ | √ | √ |
| H'BC | SCUCMD0 (CMD0out_BUSIF) | Receive | — | Arbitrary | √ | √ | √ | √ |
| H'BE | SCUCMD1 (CMD1out_BUSIF) | Receive | — | Arbitrary | √ | √ | √ | √ |
| H'9A | SCUOUT0 (SRC0out_BUSIF) | Receive | — | Arbitrary | √ | √ | √ | √ |
| H'9C | SCUOUT1 (SRC1out_BUSIF) | Receive | — | Arbitrary | √ | √ | √ | √ |
| H'9E | SCUOUT2 (SRC2out_BUSIF) | Receive | — | Arbitrary | √ | √ | √ | √ |
| H'A0 | SCUOUT3 (SRC3out_BUSIF) | Receive | — | Arbitrary | √ | √ | √ | √ |
| H'B0 | SCUOUT4 (SRC4out_BUSIF) | Receive | — | Arbitrary | √ | √ | √ | √ |
| H'B2 | SCUOUT5 (SRC5out_BUSIF) | Receive | — | Arbitrary | √ | √ | √ | √ |
| H'B4 | SCUOUT6 (SRC6out_BUSIF) | Receive | — | Arbitrary | √ | √ | √ | √ |
| H'B6 | SCUOUT7 (SRC7out_BUSIF) | Receive | — | Arbitrary | √ | √ | √ | √ |
| H'B8 | SCUOUT8 (SRC8out_BUSIF) | Receive | — | Arbitrary | √ | √ | √ | √ |
| H'BA | SCUOUT9 (SRC9out_BUSIF) | Receive | — | Arbitrary | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| DMARS MID + RID | DMA Transfer Request Source | DMA Transfer Request Signal | Source | Destination | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------------------|--------------------------------|--------------------------------|-----------|-------------|--------|----------------------------|--------|--------|
| | | | | | √ | √ | √ | √ |
| H'15 | SSI0_0 transmitter | TXI (Transmit data request) | Arbitrary | ssip00 | √ | √ | √ | √ |
| H'16 | SSI0_0 receiver | RXI (Receive data request) | ssip00 | Arbitrary | √ | √ | √ | √ |
| H'35 | SSI0_1 transmitter | TXI (Transmit data request) | Arbitrary | ssip01 | √ | √ | √ | √ |
| H'36 | SSI0_1 receiver | RXI (Receive data request) | ssip01 | Arbitrary | √ | √ | √ | √ |
| H'37 | SSI0_2 transmitter | TXI (Transmit data request) | Arbitrary | ssip02 | √ | √ | √ | √ |
| H'38 | SSI0_2 receiver | RXI (Receive data request) | ssip02 | Arbitrary | √ | √ | √ | √ |
| H'47 | SSI0_3 transmitter | TXI (Transmit data request) | Arbitrary | ssip03 | √ | √ | √ | √ |
| H'48 | SSI0_3 receiver | RXI (Receive data request) | ssip03 | Arbitrary | √ | √ | √ | √ |
| H'3F | SSI0_4 transmitter | TXI (Transmit data request) | Arbitrary | ssip04 | √ | √ | √ | — |
| H'40 | SSI0_4 receiver | RXI (Receive data request) | ssip04 | Arbitrary | √ | √ | √ | — |
| H'43 | SSI0_5 transmitter | TXI (Transmit data request) | Arbitrary | ssip05 | √ | √ | √ | — |
| H'44 | SSI0_5 receiver | RXI (Receive data request) | ssip05 | Arbitrary | √ | √ | √ | — |
| H'4F | SSI0_6 transmitter | TXI (Transmit data request) | Arbitrary | ssip06 | √ | √ | √ | — |
| H'50 | SSI0_6 receiver | RXI (Receive data request) | ssip06 | Arbitrary | √ | √ | √ | — |
| H'53 | SSI0_7 transmitter | TXI (Transmit data request) | Arbitrary | ssip07 | √ | √ | √ | — |
| H'54 | SSI0_7 receiver | RXI (Receive data request) | ssip07 | Arbitrary | √ | √ | √ | — |
| H'49 | SSI1_0 transmitter | TXI (Transmit data request) | Arbitrary | ssip10 | √ | √ | √ | √ |
| H'4A | SSI1_0 receiver | RXI (Receive data request) | ssip10 | Arbitrary | √ | √ | √ | √ |
| H'4B | SSI1_1 transmitter | TXI (Transmit data request) | Arbitrary | ssip11 | √ | √ | √ | √ |
| H'4C | SSI1_1 receiver | RXI (Receive data request) | ssip11 | Arbitrary | √ | √ | √ | √ |
| H'57 | SSI1_2 transmitter | TXI (Transmit data request) | Arbitrary | ssip12 | √ | √ | √ | √ |
| H'58 | SSI1_2 receiver | RXI (Receive data request) | ssip12 | Arbitrary | √ | √ | √ | √ |
| H'59 | SSI1_3 transmitter | TXI (Transmit data request) | Arbitrary | ssip13 | √ | √ | √ | √ |
| H'5A | SSI1_3 receiver | RXI (Receive data request) | ssip13 | Arbitrary | √ | √ | √ | √ |
| H'5F | SSI1_4 transmitter | TXI (Transmit data request) | Arbitrary | ssip14 | √ | √ | √ | — |
| H'60 | SSI1_4 receiver | RXI (Receive data request) | ssip14 | Arbitrary | √ | √ | √ | — |
| H'C3 | SSI1_5 transmitter | TXI (Transmit data request) | Arbitrary | ssip15 | √ | √ | √ | — |
| H'C4 | SSI1_5 receiver | RXI (Receive data request) | ssip15 | Arbitrary | √ | √ | √ | — |
| H'C7 | SSI1_6 transmitter | TXI (Transmit data request) | Arbitrary | ssip16 | √ | √ | √ | — |
| H'C8 | SSI1_6 receiver | RXI (Receive data request) | ssip16 | Arbitrary | √ | √ | √ | — |
| H'CB | SSI1_7 transmitter | TXI (Transmit data request) | Arbitrary | ssip17 | √ | √ | √ | — |
| H'CC | SSI1_7 receiver | RXI (Receive data request) | ssip17 | Arbitrary | √ | √ | √ | — |
| H'63 | SSI2_0 transmitter | TXI (Transmit data request) | Arbitrary | ssip20 | √ | √ | √ | √ |
| H'64 | SSI2_0 receiver | RXI (Receive data request) | ssip20 | Arbitrary | √ | √ | √ | √ |
| H'67 | SSI2_1 transmitter | TXI (Transmit data request) | Arbitrary | ssip21 | √ | √ | √ | √ |
| H'68 | SSI2_1 receiver | RXI (Receive data request) | ssip21 | Arbitrary | √ | √ | √ | √ |
| H'6B | SSI2_2 transmitter | TXI (Transmit data request) | Arbitrary | ssip22 | √ | √ | √ | √ |
| H'6C | SSI2_2 receiver | RXI (Receive data request) | ssip22 | Arbitrary | √ | √ | √ | √ |
| H'6D | SSI2_3 transmitter | TXI (Transmit data request) | Arbitrary | ssip23 | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| DMARS MID + RID | DMA Transfer Request Source | DMA Transfer Request Signal | Source | Destination | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------------------|--------------------------------|--------------------------------|-----------|-------------|--------|----------------------------|--------|--------|
| | | | | | √ | √ | √ | √ |
| H'6E | SSI2_3 receiver | RXI (Receive data request) | ssip23 | Arbitrary | √ | √ | √ | √ |
| H'CF | SSI2_4 transmitter | TXI (Transmit data request) | Arbitrary | ssip24 | √ | √ | √ | — |
| H'CE | SSI2_4 receiver | RXI (Receive data request) | ssip24 | Arbitrary | √ | √ | √ | — |
| H'EB | SSI2_5 transmitter | TXI (Transmit data request) | Arbitrary | ssip25 | √ | √ | √ | — |
| H'EC | SSI2_5 receiver | RXI (Receive data request) | ssip25 | Arbitrary | √ | √ | √ | — |
| H'ED | SSI2_6 transmitter | TXI (Transmit data request) | Arbitrary | ssip26 | √ | √ | √ | — |
| H'EE | SSI2_6 receiver | RXI (Receive data request) | ssip26 | Arbitrary | √ | √ | √ | — |
| H'EF | SSI2_7 transmitter | TXI (Transmit data request) | Arbitrary | ssip27 | √ | √ | √ | — |
| H'F0 | SSI2_7 receiver | RXI (Receive data request) | ssip27 | Arbitrary | √ | √ | √ | — |
| H'6F | SSI3_0 transmitter | TXI (Transmit data request) | Arbitrary | ssip30 | √ | √ | √ | √ |
| H'70 | SSI3_0 receiver | RXI (Receive data request) | ssip30 | Arbitrary | √ | √ | √ | √ |
| H'21 | SSI3_1 transmitter | TXI (Transmit data request) | Arbitrary | ssip31 | √ | √ | √ | √ |
| H'22 | SSI3_1 receiver | RXI (Receive data request) | ssip31 | Arbitrary | √ | √ | √ | √ |
| H'23 | SSI3_2 transmitter | TXI (Transmit data request) | Arbitrary | ssip32 | √ | √ | √ | √ |
| H'24 | SSI3_2 receiver | RXI (Receive data request) | ssip32 | Arbitrary | √ | √ | √ | √ |
| H'25 | SSI3_3 transmitter | TXI (Transmit data request) | Arbitrary | ssip33 | √ | √ | √ | √ |
| H'26 | SSI3_3 receiver | RXI (Receive data request) | ssip33 | Arbitrary | √ | √ | √ | √ |
| H'27 | SSI3_4 transmitter | TXI (Transmit data request) | Arbitrary | ssip34 | √ | √ | √ | — |
| H'28 | SSI3_4 receiver | RXI (Receive data request) | ssip34 | Arbitrary | √ | √ | √ | — |
| H'29 | SSI3_5 transmitter | TXI (Transmit data request) | Arbitrary | ssip35 | √ | √ | √ | — |
| H'2A | SSI3_5 receiver | RXI (Receive data request) | ssip35 | Arbitrary | √ | √ | √ | — |
| H'2B | SSI3_6 transmitter | TXI (Transmit data request) | Arbitrary | ssip36 | √ | √ | √ | — |
| H'2C | SSI3_6 receiver | RXI (Receive data request) | ssip36 | Arbitrary | √ | √ | √ | — |
| H'2D | SSI3_7 transmitter | TXI (Transmit data request) | Arbitrary | ssip37 | √ | √ | √ | — |
| H'2E | SSI3_7 receiver | RXI (Receive data request) | ssip37 | Arbitrary | √ | √ | √ | — |
| H'71 | SSI4_0 transmitter | TXI (Transmit data request) | Arbitrary | ssip40 | √ | √ | √ | √ |
| H'72 | SSI4_0 receiver | RXI (Receive data request) | ssip40 | Arbitrary | √ | √ | √ | √ |
| H'17 | SSI4_1 transmitter | TXI (Transmit data request) | Arbitrary | ssip41 | √ | √ | √ | √ |
| H'18 | SSI4_1 receiver | RXI (Receive data request) | ssip41 | Arbitrary | √ | √ | √ | √ |
| H'19 | SSI4_2 transmitter | TXI (Transmit data request) | Arbitrary | ssip42 | √ | √ | √ | √ |
| H'1A | SSI4_2 receiver | RXI (Receive data request) | ssip42 | Arbitrary | √ | √ | √ | √ |
| H'1B | SSI4_3 transmitter | TXI (Transmit data request) | Arbitrary | ssip43 | √ | √ | √ | √ |
| H'1C | SSI4_3 receiver | RXI (Receive data request) | ssip43 | Arbitrary | √ | √ | √ | √ |
| H'1D | SSI4_4 transmitter | TXI (Transmit data request) | Arbitrary | ssip44 | √ | √ | √ | — |
| H'1E | SSI4_4 receiver | RXI (Receive data request) | ssip44 | Arbitrary | √ | √ | √ | — |
| H'1F | SSI4_5 transmitter | TXI (Transmit data request) | Arbitrary | ssip45 | √ | √ | √ | — |
| H'20 | SSI4_5 receiver | RXI (Receive data request) | ssip45 | Arbitrary | √ | √ | √ | — |
| H'31 | SSI4_6 transmitter | TXI (Transmit data request) | Arbitrary | ssip46 | √ | √ | √ | — |
| H'32 | SSI4_6 receiver | RXI (Receive data request) | ssip46 | Arbitrary | √ | √ | √ | — |

| | | | | | Second Generation RZ/G Series Products | | | |
|--------------------|---------------------------------|--------------------------------|-----------|-------------|---|----------------------------|--------|--------|
| DMARS MID + RID | DMA Transfer Request Source | DMA Transfer Request Signal | Source | Destination | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| H'33 | SSI4_7 transmitter | TXI (Transmit data request) | Arbitrary | ssip47 | √ | √ | √ | — |
| H'34 | SSI4_7 receiver | RXI (Receive data request) | ssip47 | Arbitrary | √ | √ | √ | — |
| H'73 | SSI5 transmitter | TXI (Transmit data request) | Arbitrary | ssip5 | √ | √ | √ | √ |
| H'74 | SSI5 receiver | RXI (Receive data request) | ssip5 | Arbitrary | √ | √ | √ | √ |
| H'75 | SSI6 transmitter | TXI (Transmit data request) | Arbitrary | ssip6 | √ | √ | √ | √ |
| H'76 | SSI6 receiver | RXI (Receive data request) | ssip6 | Arbitrary | √ | √ | √ | √ |
| H'79 | SSI7 transmitter | TXI (Transmit data request) | Arbitrary | ssip7 | √ | √ | √ | √ |
| H'7A | SSI7 receiver | RXI (Receive data request) | ssip7 | Arbitrary | √ | √ | √ | √ |
| H'7B | SSI8 transmitter | TXI (Transmit data request) | Arbitrary | ssip8 | √ | √ | √ | √ |
| H'7C | SSI8 receiver | RXI (Receive data request) | ssip8 | Arbitrary | √ | √ | √ | √ |
| H'7D | SSI9_0 transmitter | TXI (Transmit data request) | Arbitrary | ssip90 | √ | √ | √ | √ |
| H'7E | SSI9_0 receiver | RXI (Receive data request) | ssip90 | Arbitrary | √ | √ | √ | √ |
| H'7F | SSI9_1 transmitter | TXI (Transmit data request) | Arbitrary | ssip91 | √ | √ | √ | √ |
| H'80 | SSI9_1 receiver | RXI (Receive data request) | ssip91 | Arbitrary | √ | √ | √ | √ |
| H'81 | SSI9_2 transmitter | TXI (Transmit data request) | Arbitrary | ssip92 | √ | √ | √ | √ |
| H'82 | SSI9_2 receiver | RXI (Receive data request) | ssip92 | Arbitrary | √ | √ | √ | √ |
| H'83 | SSI9_3 transmitter | TXI (Transmit data request) | Arbitrary | ssip93 | √ | √ | √ | √ |
| H'84 | SSI9_3 receiver | RXI (Receive data request) | ssip93 | Arbitrary | √ | √ | √ | √ |
| H'A3 | SSI9_4 transmitter | TXI (Transmit data request) | Arbitrary | ssip94 | √ | √ | √ | — |
| H'A4 | SSI9_4 receiver | RXI (Receive data request) | ssip94 | Arbitrary | √ | √ | √ | — |
| H'A5 | SSI9_5 transmitter | TXI (Transmit data request) | Arbitrary | ssip95 | √ | √ | √ | — |
| H'A6 | SSI9_5 receiver | RXI (Receive data request) | ssip95 | Arbitrary | √ | √ | √ | — |
| H'A7 | SSI9_6 transmitter | TXI (Transmit data request) | Arbitrary | ssip96 | √ | √ | √ | — |
| H'A8 | SSI9_6 receiver | RXI (Receive data request) | ssip96 | Arbitrary | √ | √ | √ | — |
| H'A9 | SSI9_7 transmitter | TXI (Transmit data request) | Arbitrary | ssip97 | √ | √ | √ | — |
| H'AA | SSI9_7 receiver | RXI (Receive data request) | ssip97 | Arbitrary | √ | √ | √ | — |
| H'01 | SSIND0 transmitter (SSITDR0) | TXI (Transmit data request) | Arbitrary | ssindd0 | √ | √ | √ | √ |
| H'02 | SSIND0 receiver (SSIRDR0) | RXI (Receive data request) | ssindd0 | Arbitrary | √ | √ | √ | √ |
| H'03 | SSIND1 transmitter (SSITDR1) | TXI (Transmit data request) | Arbitrary | ssindd1 | √ | √ | √ | √ |
| H'04 | SSIND1 receiver (SSIRDR1) | RXI (Receive data request) | ssindd1 | Arbitrary | √ | √ | √ | √ |
| H'05 | SSIND2 transmitter (SSITDR2) | TXI (Transmit data request) | Arbitrary | ssindd2 | √ | √ | √ | √ |
| H'06 | SSIND2 receiver (SSIRDR2) | RXI (Receive data request) | ssindd2 | Arbitrary | √ | √ | √ | √ |
| H'07 | SSIND3 transmitter (SSITDR3) | TXI (Transmit data request) | Arbitrary | ssindd3 | √ | √ | √ | √ |

| DMARS MID + RID | DMA Transfer Request Source | DMA Transfer Request Signal | Source | Destination | Second Generation RZ/G Series Products | | | |
|--------------------|---------------------------------|--------------------------------|-----------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| H'08 | SSIND3 receiver (SSIRDR3) | RXI (Receive data request) | ssindd3 | Arbitrary | √ | √ | √ | √ |
| H'09 | SSIND4 transmitter (SSITDR4) | TXI (Transmit data request) | Arbitrary | ssindd4 | √ | √ | √ | √ |
| H'0A | SSIND4 receiver (SSIRDR4) | RXI (Receive data request) | ssindd4 | Arbitrary | √ | √ | √ | √ |
| H'0B | SSIND5 transmitter (SSITDR5) | TXI (Transmit data request) | Arbitrary | ssindd5 | √ | √ | √ | √ |
| H'0C | SSIND5 receiver (SSIRDR5) | RXI (Receive data request) | ssindd5 | Arbitrary | √ | √ | √ | √ |
| H'0D | SSIND6 transmitter (SSITDR6) | TXI (Transmit data request) | Arbitrary | ssindd6 | √ | √ | √ | √ |
| H'0E | SSIND6 receiver (SSIRDR6) | RXI (Receive data request) | ssindd6 | Arbitrary | √ | √ | √ | √ |
| H'0F | SSIND7 transmitter (SSITDR7) | TXI (Transmit data request) | Arbitrary | ssindd7 | √ | √ | √ | √ |
| H'10 | SSIND7 receiver (SSIRDR7) | RXI (Receive data request) | ssindd7 | Arbitrary | √ | √ | √ | √ |
| H'11 | SSIND8 transmitter (SSITDR8) | TXI (Transmit data request) | Arbitrary | ssindd8 | √ | √ | √ | √ |
| H'12 | SSIND8 receiver (SSIRDR8) | RXI (Receive data request) | ssindd8 | Arbitrary | √ | √ | √ | √ |
| H'13 | SSIND9 transmitter (SSITDR9) | TXI (Transmit data request) | Arbitrary | ssindd9 | √ | √ | √ | √ |
| H'14 | SSIND9 receiver (SSIRDR9) | RXI (Receive data request) | ssindd9 | Arbitrary | √ | √ | √ | √ |

Table 44.6 Data Length of DMA Transfer for Each of the On-Chip Peripheral Modules

| Module | 1 Byte | 2 Bytes | 4 Bytes | 8 Bytes | 16 Bytes | 32 Bytes | Second Generation RZ/G Series Products | | | |
|--|--------|---------|---------|---------|----------|----------|---|----------------------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SCU0/SCU1/SCU2/ SCU3/SCU4/SCU5/ SCU6/SCU7/SCU8/ SCU9 (SRC0in_BUSIF/SRC1in_B USIF/SRC2in_BUSIF/ SRC3in_BUSIF/SRC4in_ BUSIF/SRC5in_BUSIF/ SRC6in_BUSIF/ SRC7in_BUSIF/ SRC8in_BUSIF/ SRC9in_BUSIF) | — | — | √ | — | — | — | √ | √ | √ | √ |
| SCUCMD0/SCUCMD1 (CMD0out_BUSIF/ CMD1out_BUSIF) | — | — | √ | — | — | — | √ | √ | √ | √ |
| SCUOUT0/SCUOUT1/ SCUOUT2/SCUOUT3/ SCUOUT4/SCUOUT5/ SCUOUT6/SCUOUT7/ SCUOUT8/SCUOUT9 (SRC0out_BUSIF/ SRC1out_BUSIF/ SRC2out_BUSIF/ SRC3out_BUSIF/ SRC4out_BUSIF/ SRC5out_BUSIF/ SRC6out_BUSIF/ SRC7out_BUSIF/ SRC8out_BUSIF/ SRC9out_BUSIF) | — | — | √ | — | — | — | √ | √ | √ | √ |
| SSI0_0/SSI0_1/SSI0_2/ SSI0_3/SSI1_0/SSI1_1/ SSI1_2/SSI1_3/SSI2_0/ SSI2_1/SSI2_2/SSI2_3/ SSI2_4/SSI2_5/SSI2_6/ SSI2_7/SSI3_0/SSI4_0/ SSI5/SSI6/SSI7/SSI8/ SSI9_0/SSI9_1/SSI9_2/ SSI9_3 | — | — | √ | — | — | — | √ | √ | √ | √ |
| SSI0_4/SSI0_5/SSI0_6/ SSI0_7/SSI1_4/SSI1_5/ SSI1_6/SSI1_7/SSI3_0/ SSI3_1/SSI3_2/SSI3_3/ SSI3_4/SSI3_5/SSI3_6/ SSI3_7/SSI4_0/SSI4_1/ SSI4_2/SSI4_3/SSI4_4/ SSI4_5/SSI4_6/SSI4_7/ SSI9_4/SSI9_5/SSI9_6/ SSI9_7 | — | — | √ | — | — | — | √ | √ | √ | — |
| SSIND0/1/2/3/4/5/6/7/8/9 (SSITDR0/1/2/3/4/5/6/7/8/9, SSIRD0/1/2/3/4/5/6/7/8/9) | — | — | √ | — | — | — | √ | √ | √ | √ |

44.3.2 Channel Priority

When the audio-DMAC receives simultaneous transfer requests on two or more channels, it transfers data according to a predetermined priority. Two modes (fixed mode and round-robin mode) are selected by the PR[1:0] bits in DMAOR.

(1) Fixed Mode

In this mode, the priority levels among the channels remain fixed.

CH0 > CH1 > ... > CH11 > CH12, CH13 > CH14 > ... > CH25 > CH26 > CH27 > CH28 > CH29 > CH30 > CH31

(2) Round-Robin Mode

In round-robin mode, each time data of one transfer unit (byte, word, long-word, 8-byte or 16-byte units) is transferred on one channel, the priority is rotated. The channel on which the transfer was just finished rotates to the bottom of the priority. The priority of round-robin mode is CH0 > CH1 > ... > CH11 > CH12 > ... > CH24 > CH25 > ... > CH30 > CH31 immediately after reset.

44.3.3 Slow Speed Mode

In the low-speed mode, a single round of DMA transfer is performed every time the number of clock cycles specified by the SLM bits in DMACHCRB elapse. This mode can be selected per DMA channel. Transfer on other channels can proceed after each round of transfer for a channel in the low-speed mode is completed.

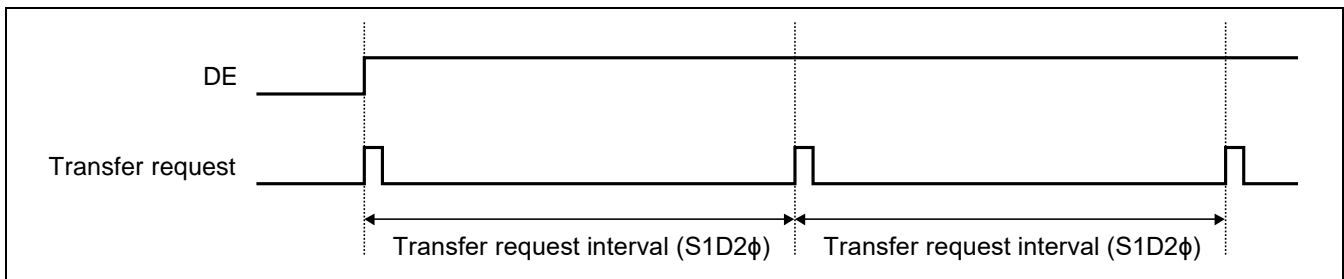


Figure 44.2 Slow Speed Mode

44.3.4 Descriptor Memory

The descriptor memory function is selected by setting the DPM[1:0] bits in DMACHCR to B'01, B'10, or B'11. When DMATCR is set to 0 and the DMA transfer is completed, the next set of settings is read, and if only a single channel is enabled, the contents defined by up to 128 stages of descriptor memory can be consecutively transferred when the built-in descriptor memory is used. External memory can also be used as the descriptor memory. In that case, the contents defined by up to 256 stages of descriptor memory can be transferred.

The following initial settings are required to use the descriptor memory.

- Set the base address of the descriptor memory for the DMA transfer in DMADPBASE.
- Set the DRST bit in DMACHCRB to reset the descriptor memory.
- Set the number of stages of the descriptor memory in the DCNT bits of DMACHCRB.

The descriptor memory is shared between all channels. Ensure that the areas of descriptor memory for use by each of the channels do not overlap.

It is necessary to arrange each stage of the descriptor memory on a 16-byte boundary.

There are two methods to activate the descriptor memory as follows.

- Specify the first DMA transfer settings in DMASAR, DMADAR, and DMATCR, and specify the subsequent settings in the descriptor memory. Then, set the DPB bit in DMACHCR to 0 to activate the descriptor memory. In this case, after completion of the transfer specified in DMASAR, DMADAR, and DMATCR, transfer continues after new settings are read from the descriptor memory. Note, however, that, when the operating mode of the descriptor memory is set to the repeat mode, the values specified in DMASAR, DMADAR, and DMATCR are not read, and the transfer starts and is repeated from the head of the descriptor memory.
- Write the DMA transfer settings to the descriptor memory, and write 1 to the DPB bit in DMACHCR to activate the descriptor memory. In this case, the DMA transfer starts from the first settings in the descriptor memory.

There are three operating modes of the descriptor memory, which can be selected by setting the DPM bits in DMACHCR.

For details on these operating modes, see the descriptions of each operating mode in this section.

(1) Configuration of Descriptor Memory

Figure 44.3 shows the configuration of the built-in descriptor memory.

The capacity of the built-in descriptor memory is 16 bytes per stage × 128 stages.

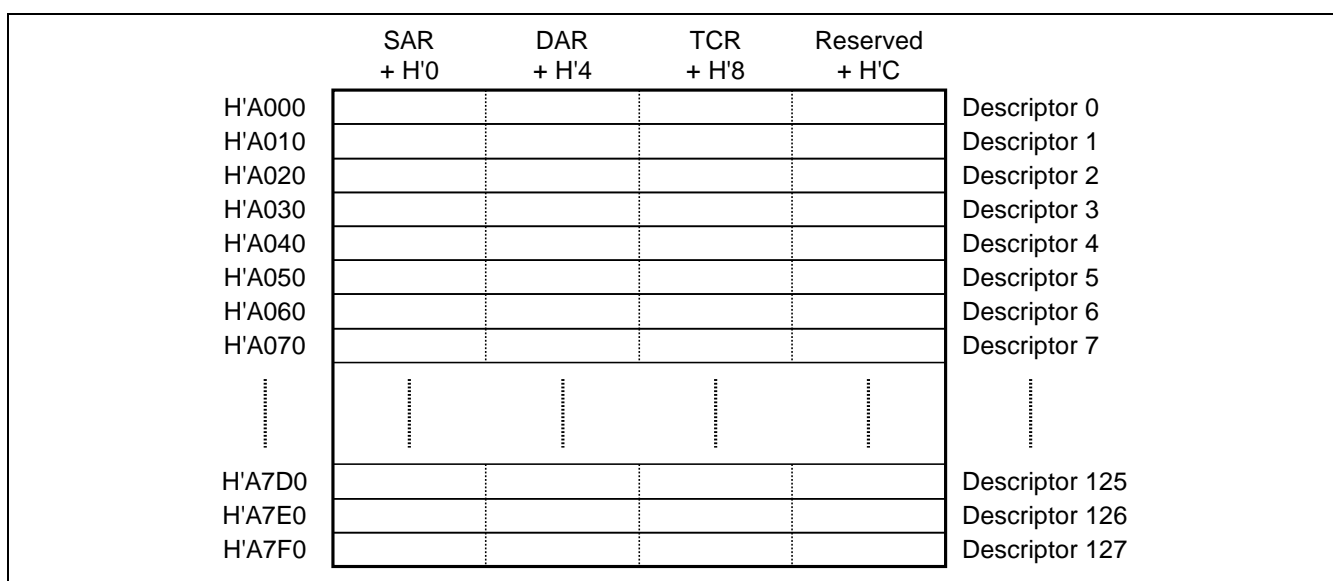


Figure 44.3 Configuration of Built-in Descriptor Memory

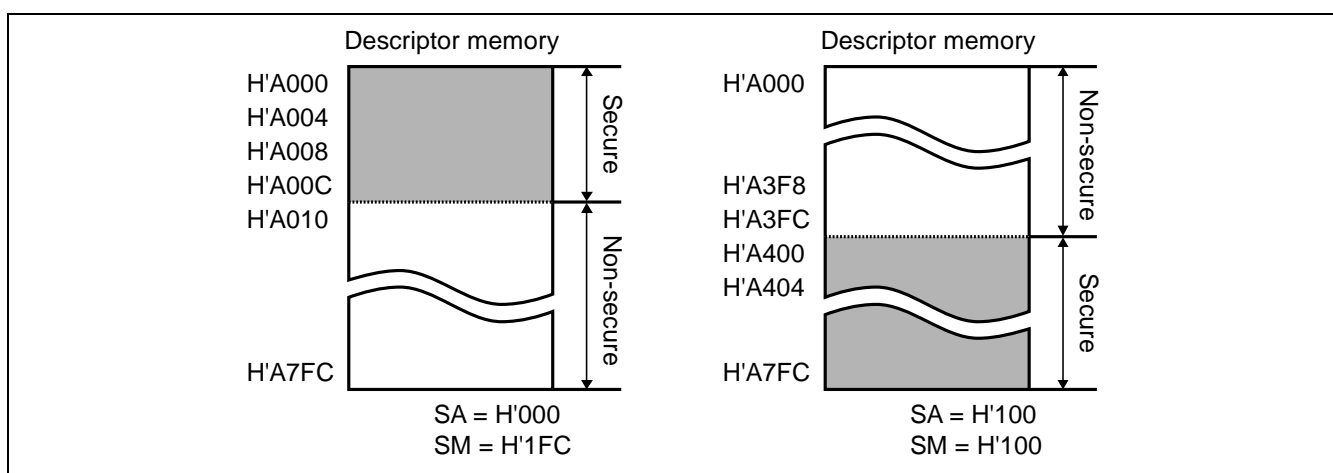


Figure 44.4 Example of DMADPSEC Setting

(2) Flow of Updating from Descriptor Memory

The RPT bits in DMACHCR can be used to specify which registers are to be updated from the descriptor memory.

The DPTR bits in DMACHCRB are incremented when updating from the descriptor memory is completed. If the DPTR value matches the DCNT value, the DPTR value is reset to 0.

This flow is automatically processed by hardware.

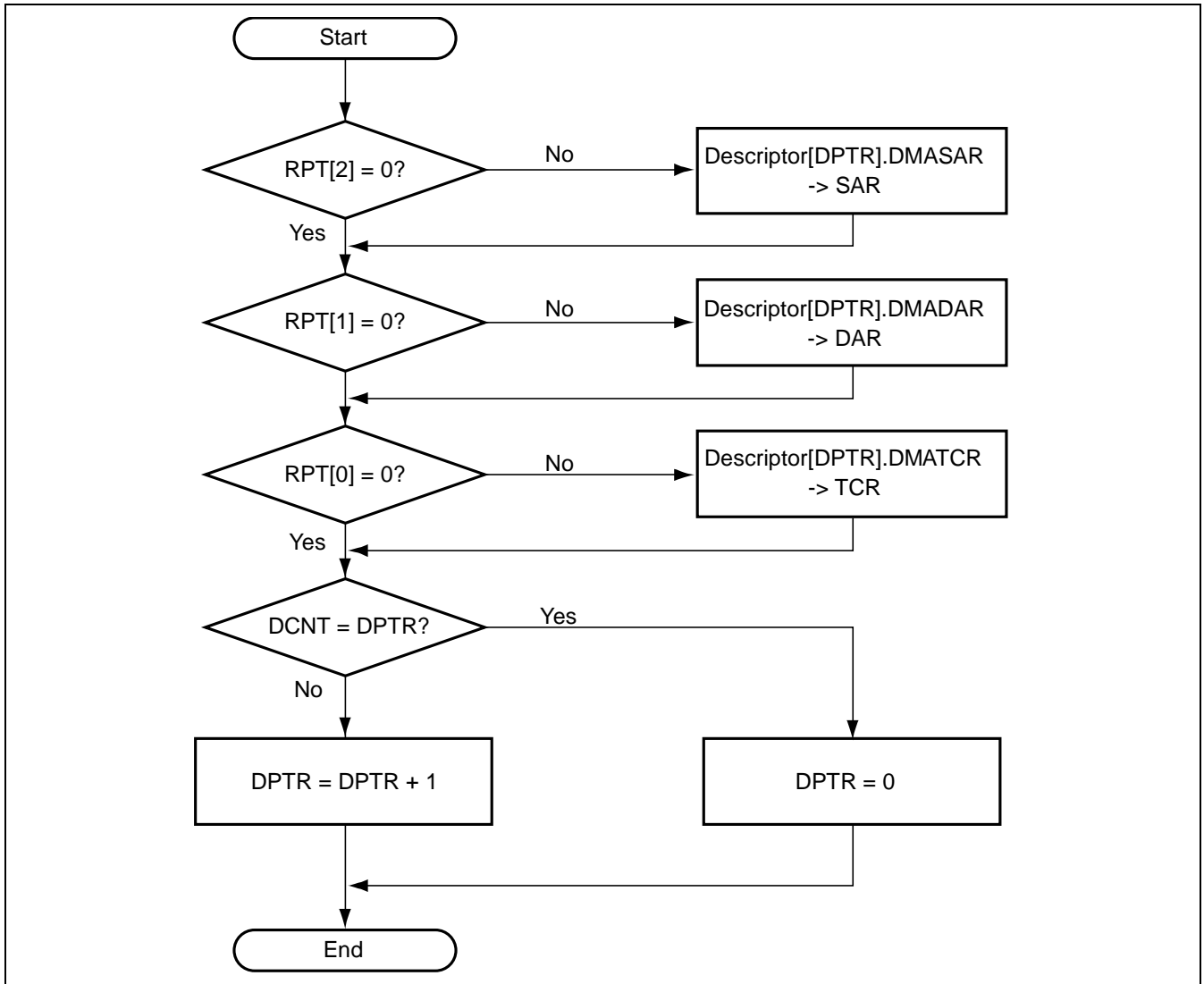


Figure 44.5 Flow of Updating from Descriptor Memory

(3) Operating Mode 1 of Descriptor Memory

Set the DPM bits in DMACHCR to B'01 to select operating mode 1 (normal mode). This mode allows consecutive transfer under control of the descriptor memory.

In this mode, the DMA transfer is complete when the TE bit in DMACHCR is set to 1 after transfer under control of the number of stages of the descriptor memory specified in the DCNT bits in DMACHCRB.

When the DSIE bit in DMACHCR is set to 1, a DSE interrupt is generated every time transfer under the control of one stage of descriptor memory is complete. If a first DSE interrupt has not been processed when a further DSE interrupt is generated, the contents of the next stage of the descriptor memory are not read and the DMA transfer is aborted. Clearing the DSE bit only causes clearing of the first interrupt and reading of the next stage of the descriptor memory.

Figure 44.6 is an example of transfer when operating mode 1 is selected and the TE and DSE bits are set to 1.

Figure 44.7 is an example of transfer when there is a delay in processing of the DSE interrupt when operating mode 1 is selected and the TE and DSE bits are set to 1.

Figure 44.8 is an example of transfer when operating mode 1 is selected and the TE bit is set to 1.

In each example, there are four descriptor stages.

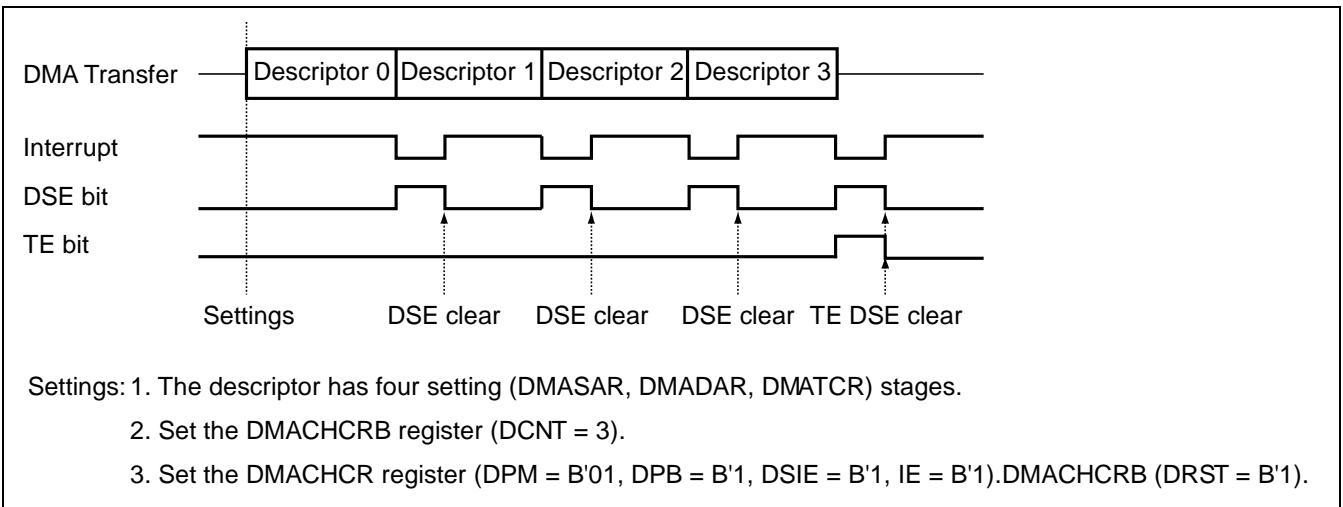


Figure 44.6 Operating Mode 1 (Example 1)

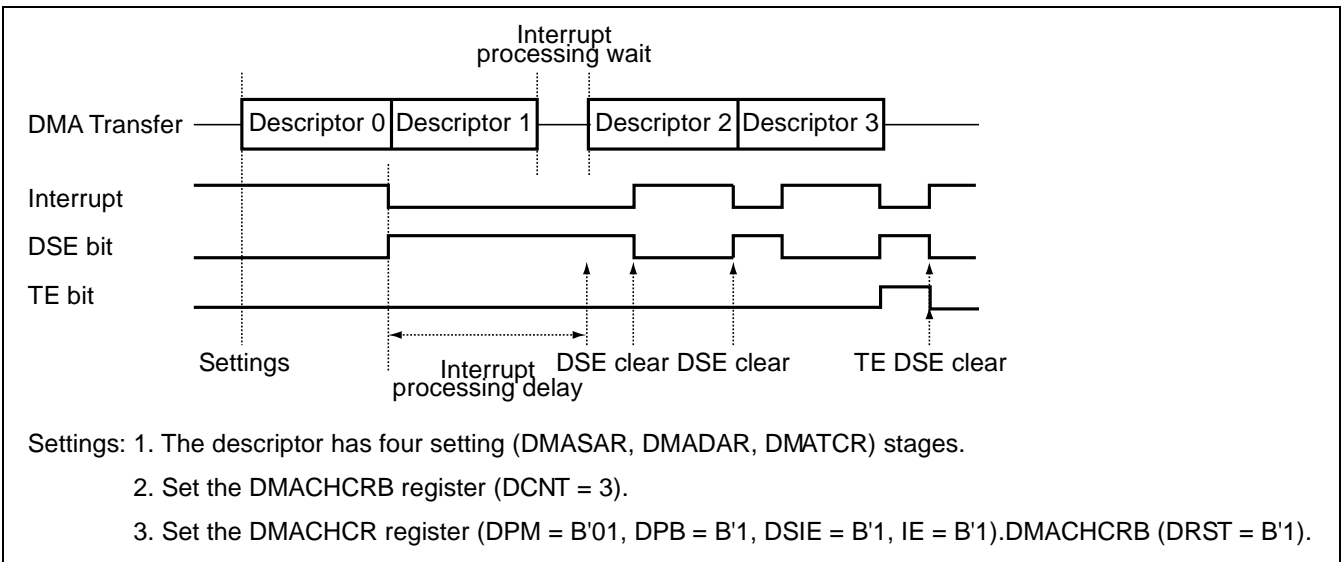


Figure 44.7 Operating Mode 1 (Example 2)

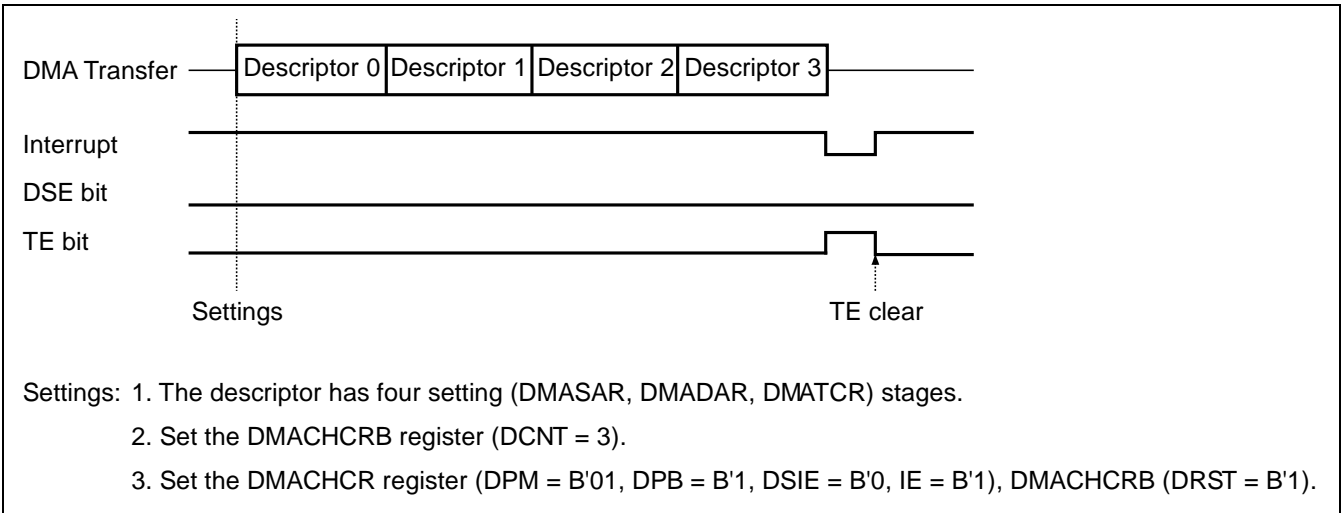


Figure 44.8 Operating Mode 1 (Example 3)

(4) Operating Mode 2 of Descriptor Memory

Set the DPM bits in DMACHCR to B'10 to select operating mode 2 (repeat mode). This mode allows consecutive transfer under control of the descriptor memory.

In this mode, after transfer under control of the number of stages of descriptor memory specified in the DCNT bits in DMACHCRB, the TE bit in DMACHCR is set to 1. This operation is then repeated from the head of the descriptor memory.

When the DSIE bit in DMACHCR is set to 1, a DSE interrupt is generated every time transfer under the control of one stage of descriptor memory is complete. If a first DSE interrupt has not been processed when a further DSE interrupt is generated, the contents of the next stage of the descriptor memory are not read and the DMA transfer is aborted. Clearing the DSE bit only causes clearing of the first interrupt and reading of the next stage of the descriptor memory.

When the DSIE bit in DMACHCR is set to 0, after transfer under control of all stages of the descriptor memory is complete, the TE bit in DMACHCR is set to 1 and a TE interrupt is generated. If a TE interrupt has not been processed when a further interrupt is generated, the contents of the next stage of the descriptor memory are not read and the DMA transfer is aborted. Clearing the TE bit only causes clearing of the first interrupt and reading of the next stage of the descriptor memory.

To end operation in mode 2, change the mode to mode 1 by using the TE interrupt processing. When the mode is changed to mode 1, the DMA transfer is completed when the next TE interrupt is generated.

Figure 44.9 is an example of transfer when operating mode 2 is selected and the TE and DSE bits are set to 1.

Figure 44.10 is an example of transfer when there is a delay in processing of the DSE interrupt when operating mode 2 is selected and the TE and DSE bits are set to 1.

Figure 44.11 is an example of transfer when operating mode 2 is selected and the TE bit is set to 1.

Figure 44.12 is an example of transfer when there is a delay in processing of the DSE interrupt when operating mode 2 is selected and the TE is set to 1.

In each example, there are four descriptor stages.

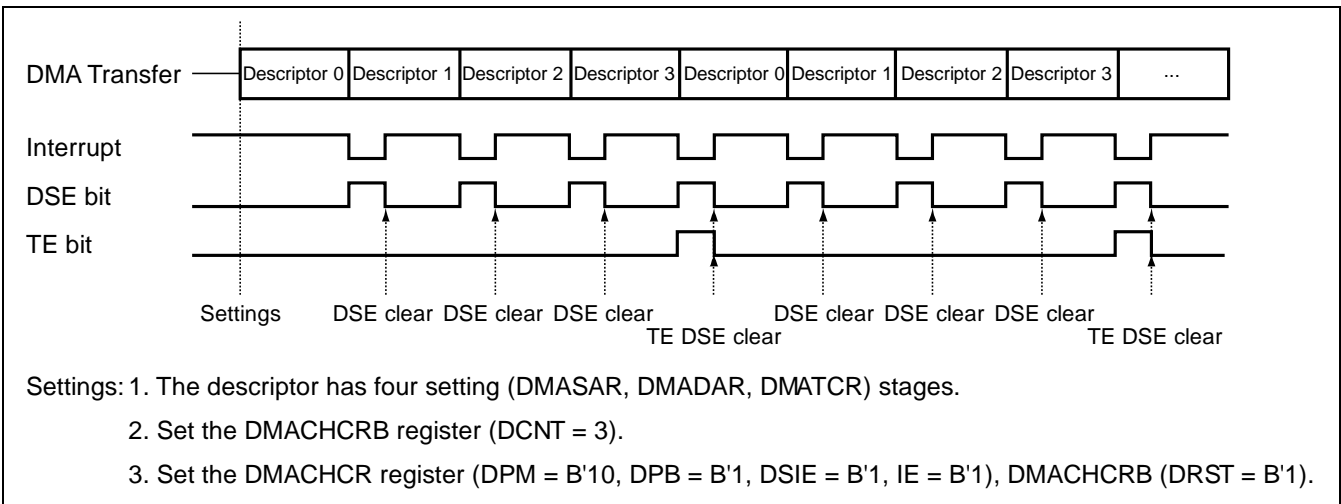


Figure 44.9 Operating Mode 2 (Example 1)

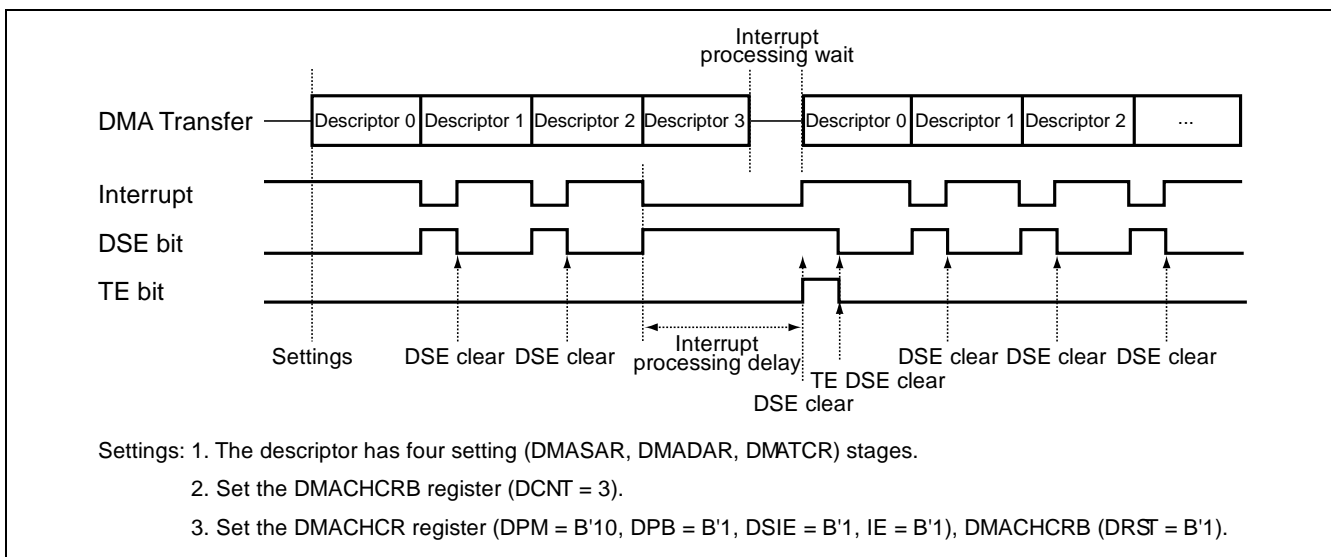


Figure 44.10 Operating Mode 2 (Example 2)

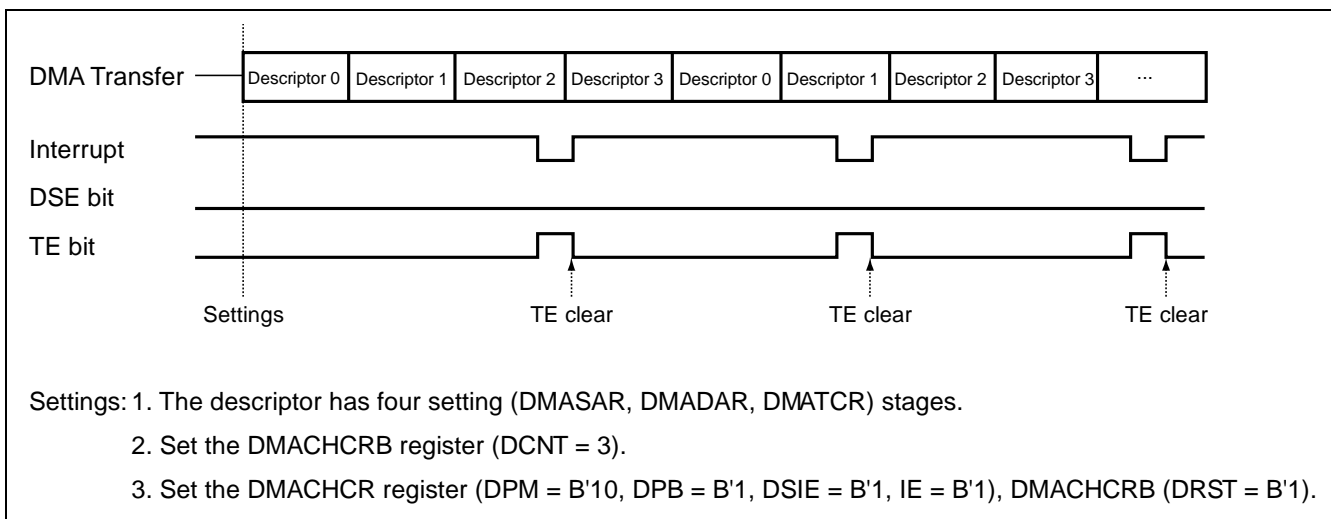


Figure 44.11 Operating Mode 2 (Example 3)

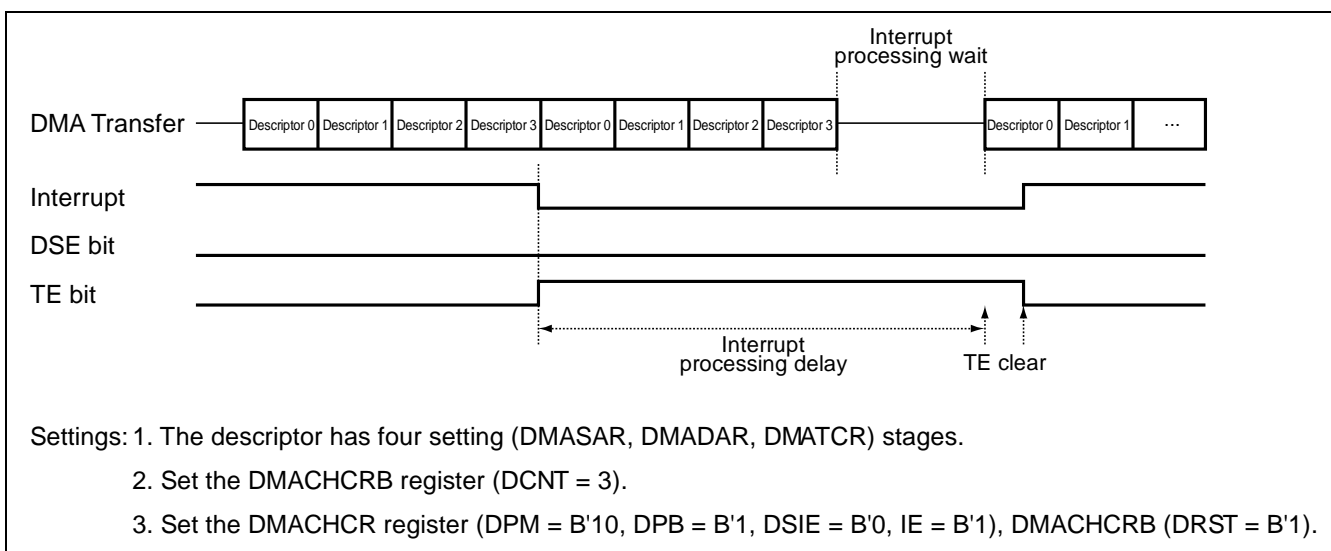


Figure 44.12 Operating Mode 2 (Example 4)

(5) Operating Mode 3 of Descriptor Memory

Set the DPM bits in DMACHCR to B'11 to select operating mode 3 (infinite repeat mode). This mode allows consecutive transfer under control of the descriptor memory.

In this mode, after transfer under control of the number of stages of descriptor memory specified in the DCNT bits in DMACHCRB, the TE bit in DMACHCR is set to 1. This operation is then repeated from the head of the descriptor memory.

When the DSIE bit in DMACHCR is set to 1, a DSE interrupt is generated every time transfer under the control of one stage of descriptor memory is complete. Even if a first DSE interrupt has not been processed when a further DSE interrupt is generated, the DMA transfer is not aborted. Regardless of the number of DSE interrupts that have been generated, the DSE bit can be cleared by writing to it once. Similarly, even if a first TE interrupt has not been processed when a further TE interrupt is generated, the DMA transfer is not aborted. Regardless of the number of TE interrupts that have been generated, the TE bit can be cleared by writing to it once.

Figure 44.13 is an example of transfer when infinite repeat mode is selected.

Figure 44.14 is an example of transfer when read-out interrupt mode is selected.

In each example, there are four descriptor stages.

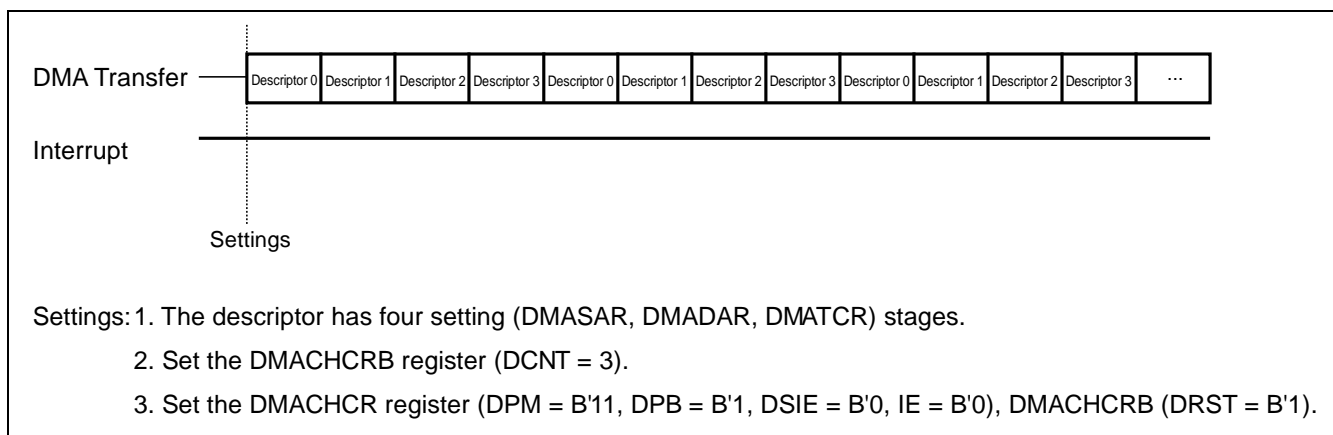


Figure 44.13 Operating Mode 3 (Infinite Repeat Mode)

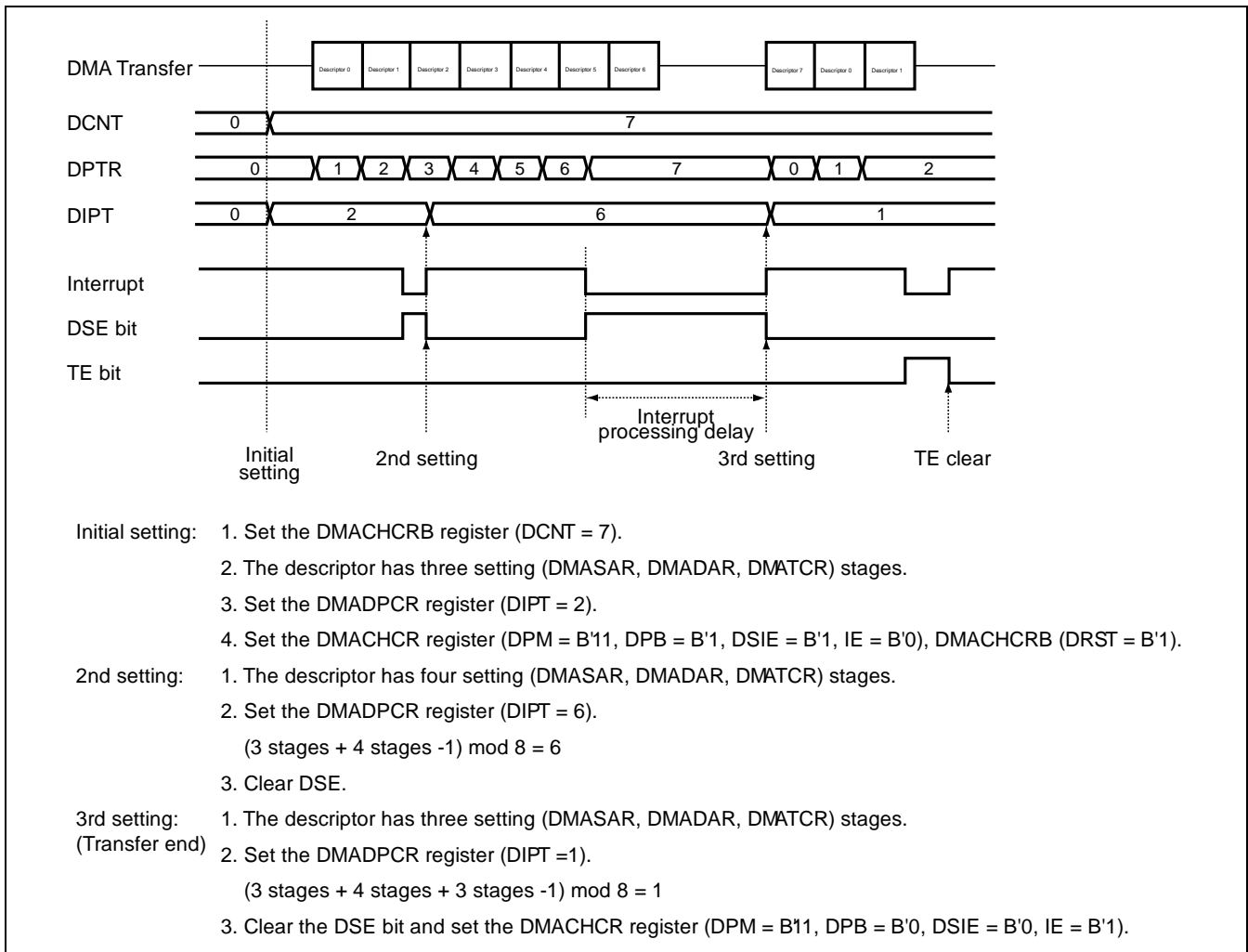


Figure 44.14 Operating Mode 3 (Read-Out Interrupt Mode)

(6) Using the Descriptor Memory for Double Buffering

To use the descriptor memory for double buffering, set the number of stages of descriptor memory to 2, set the buffer configuration to the descriptor memory, and activate the memory in operating mode 2. The DSE interrupt is used in double buffering. To end the use of double buffering, disable the descriptor operating mode, which stops the transfer on completion of the transfer currently in progress.

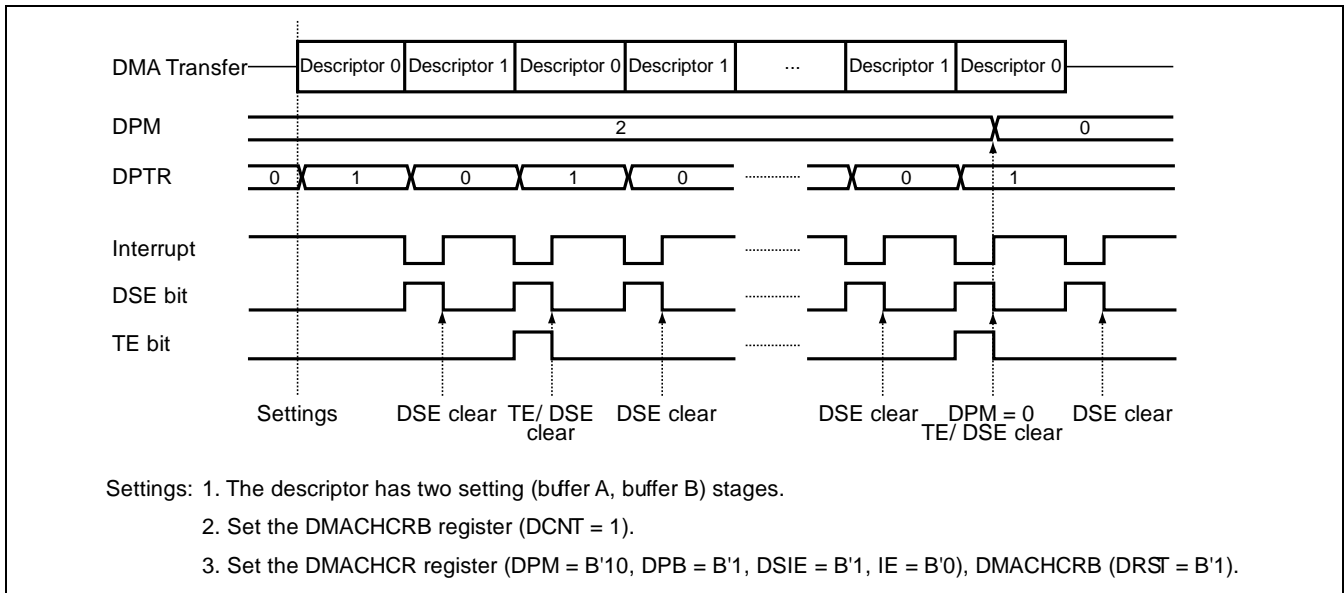


Figure 44.15 Using the Descriptor Memory for Double Buffering

44.3.5 Transmission Flow

Set the transfer conditions as required in the following registers:

DMA source address register (DMASAR), DMA destination address register (DMADAR), DMA transfer count register (DMATCR), DMA Channel control register (DMACHCR), DMA operation register (DMAOR) and DMA extended resource selector (DMARS)

The DMAC then transmits data in the following order.

- A transfer request is generated and the controller checks whether the transfer is allowed (DE = 1, DME = 1, TE = 0, DSE = 0, CAE = 0). When a channel is in the auto request mode, the transfer starts automatically.
- The controller checks whether updating from the descriptor memory is required.
Updating from the descriptor memory proceeds when the DPB bit in DMACHCR is set to 1 or the descriptor memory is enabled, if DMATCR is set to 0.
For updating by using the descriptor memory, see section 44.3.4, Descriptor Memory.
- Check whether address translation by the IPMMU is required.
Address translation by the IPMMU proceeds if the address exceeds the effective size for address translation when the DE bit in DMACHCR is enabled and updating of transfer settings from the descriptor memory is executed.
- Each time a transfer request is generated, the amount of data for a single round of transfer (specified by the TS[3:0] bits) is transmitted. The value in DMATCR is decremented by 1 every time the DMA transfer is completed.
- When the specified number of rounds of transfer are completed (the value in DMATCR is set to 0), the transfer ends normally. A TE interrupt is generated for the CPU upon the end of the transfer if the IE bit in DMACHCR is set to 1. If the descriptor memory is enabled, the processing differs with the mode of the descriptor memory. For more details, see section 44.3.4, Descriptor Memory.
- Transfer is aborted when the DMAC encounters an address error. Transfer is also aborted when the DE bit in DMACHCR or the DME bit in DMAOR is set to 0.

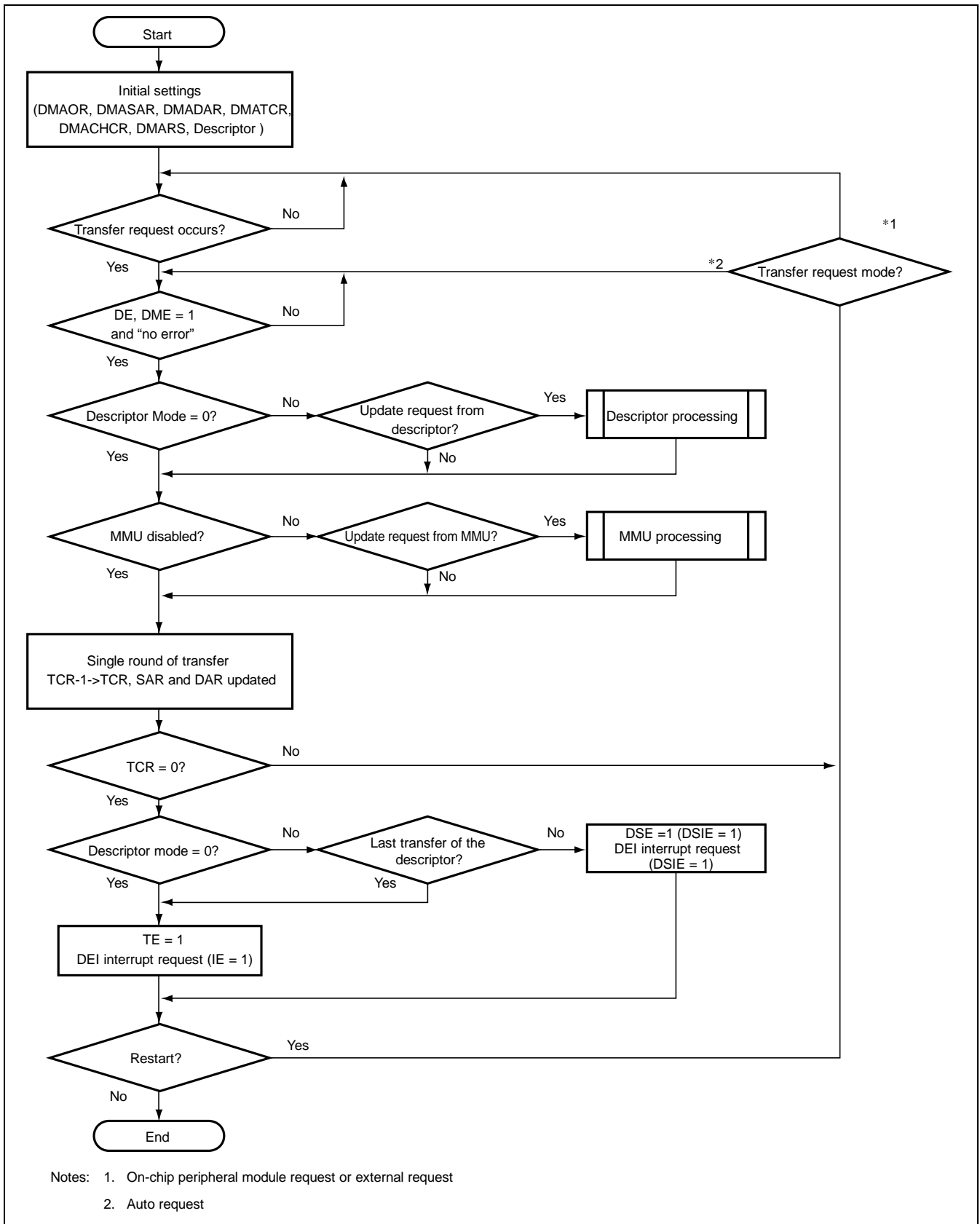


Figure 44.16 Transmission Flow

44.3.6 Total Size Transmission

The amount of data for total size transmission can be set in DMATSR (in bytes), and this setting is effective regardless of the size specified in the TS bits in DMACHCR. Thereby, the desired size can be transmitted in a single round of DMA transfer.

To use this function, make the settings for total size transmission in DMASAR, DMADAR, DMATSR, DMATSRB and DMACHCR.

Total size is set in the TCR (TSR) field of descriptors and 1 is set in DMACHCRB.DTS when total size transmission and descriptors are in use.

44.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Pay attention to the following notes when using the audio-DMAC.

(1) DMA Transfer for Peripheral Modules

When executing DMA transfer for an on-chip peripheral module, set addresses on the appropriate boundary (in terms of the amount of data for each round of transfer) for the transfer source and destination addresses. Otherwise, an address error may occur.

(2) Module Stop

While the audio-DMAC is operating, the module stop register (MSTPCR5) should not be set to stop the audio-DMAC. If the audio-DMAC is stopped in this way, results of the transfer that was in progress cannot be guaranteed.

(3) Address Error

When a DMA address error is generated, reset the registers of the channel on which the error has occurred and then start transfer anew.

(4) Aborting DMA Transfer

To abort a DMA transfer, disable the interrupt signal and set the DE bit in the DMA channel control register (DMACHCR) to 0 to disable the DMA transfer. If the TE and DSE bits are set when DMA transfer is aborted, these bits should be initialized. There is a possibility that TE and DSE will not be set with synchronized timing after transmission, so it's necessary to recognize the following three possibilities and take measures accordingly.

1. The DSE and TE bits are set to 1 before initialization of DMACHCR, but after the interrupt was disabled.
The TE and DSE bits are initialized within the DMA transfer initialization sequence.
2. The DSE and TE bits are set to 1 and the controller fails to abort the transfer, after the interrupt was disabled.
When the DE bit has become 0 after DMA transfer initialization, check the TE and DSE bits. If the TE or DSE bit is 1, go through the DMA transfer initialization process.
3. The TE and DSE bits are not cleared to 0 but the transfer is aborted, after the interrupt is disabled.
The TE or DSE bits are not set because data for transfer still remain after transfer is aborted.

Note: Initialization of DMA transfer during execution of the last transfer leads to a delay in setting of the TE and DSE bits, so include a dummy read.

Figure 44.17 shows an example of processing to abort DMA transfer.

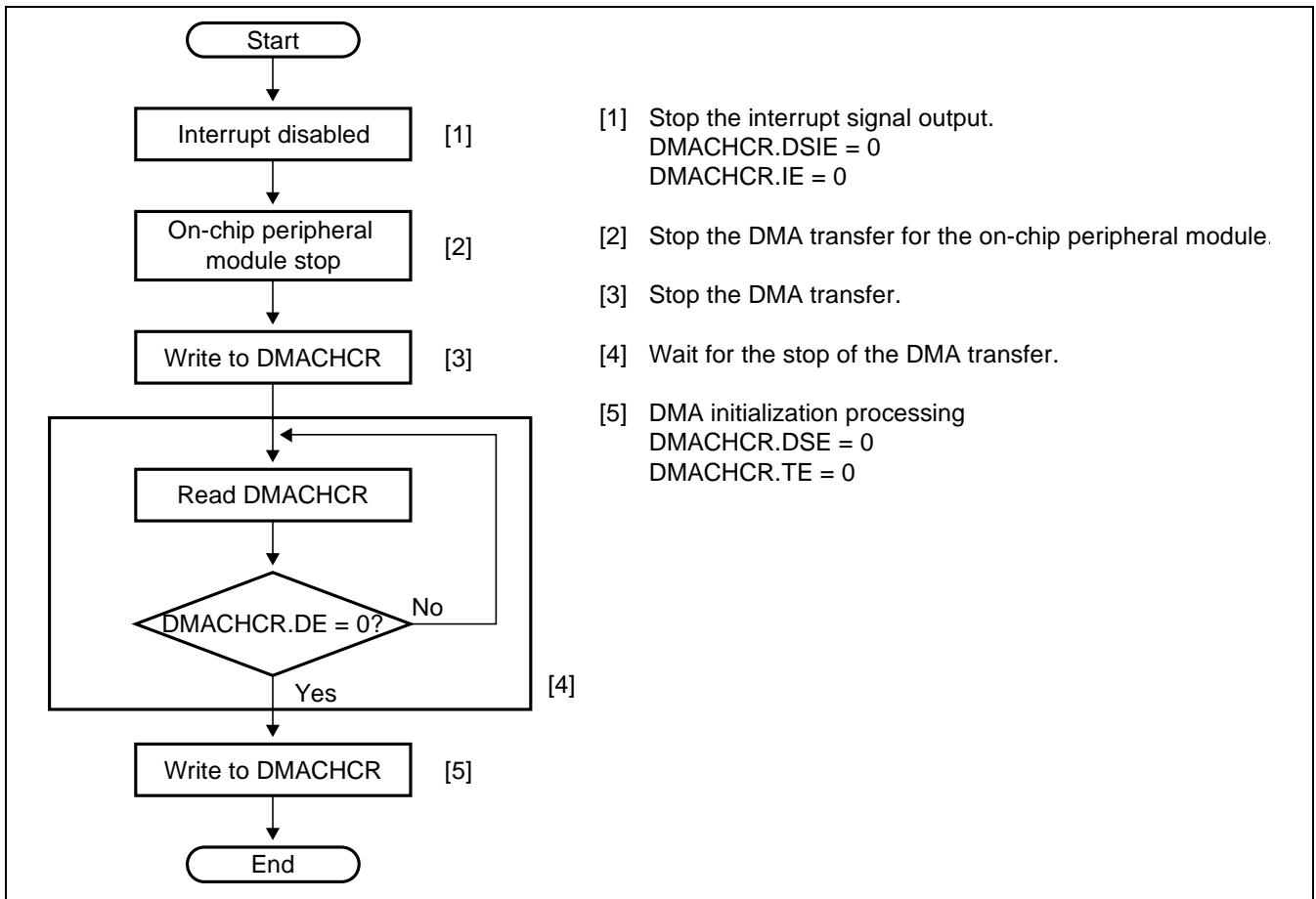


Figure 44.17 Example of Processing to Abort DMA Transfer

45. Audio-DMAC-Peripheral-Peripheral (Audio-DMACpp)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

45.1 Overview

The Audio DMAC-Peripheral-Peripheral module controls data transfer between the audio modules connected to the audio local bus.

45.1.1 Features

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

- Number of channels: 29 channels + 29 channels(extended)
- Data transfer size: Longword (4 bytes)
- Addressing mode: Dual addressing; fixed access size
- Transfer count: Not programmable
- Interrupt processing: None

[RZ/G2E]

- Number of channels: 29 channels (extended)
- Data transfer size: Longword (4 bytes)
- Addressing mode: Dual addressing; fixed access size
- Transfer count: Not programmable
- Interrupt processing: None

45.1.2 Block Diagram

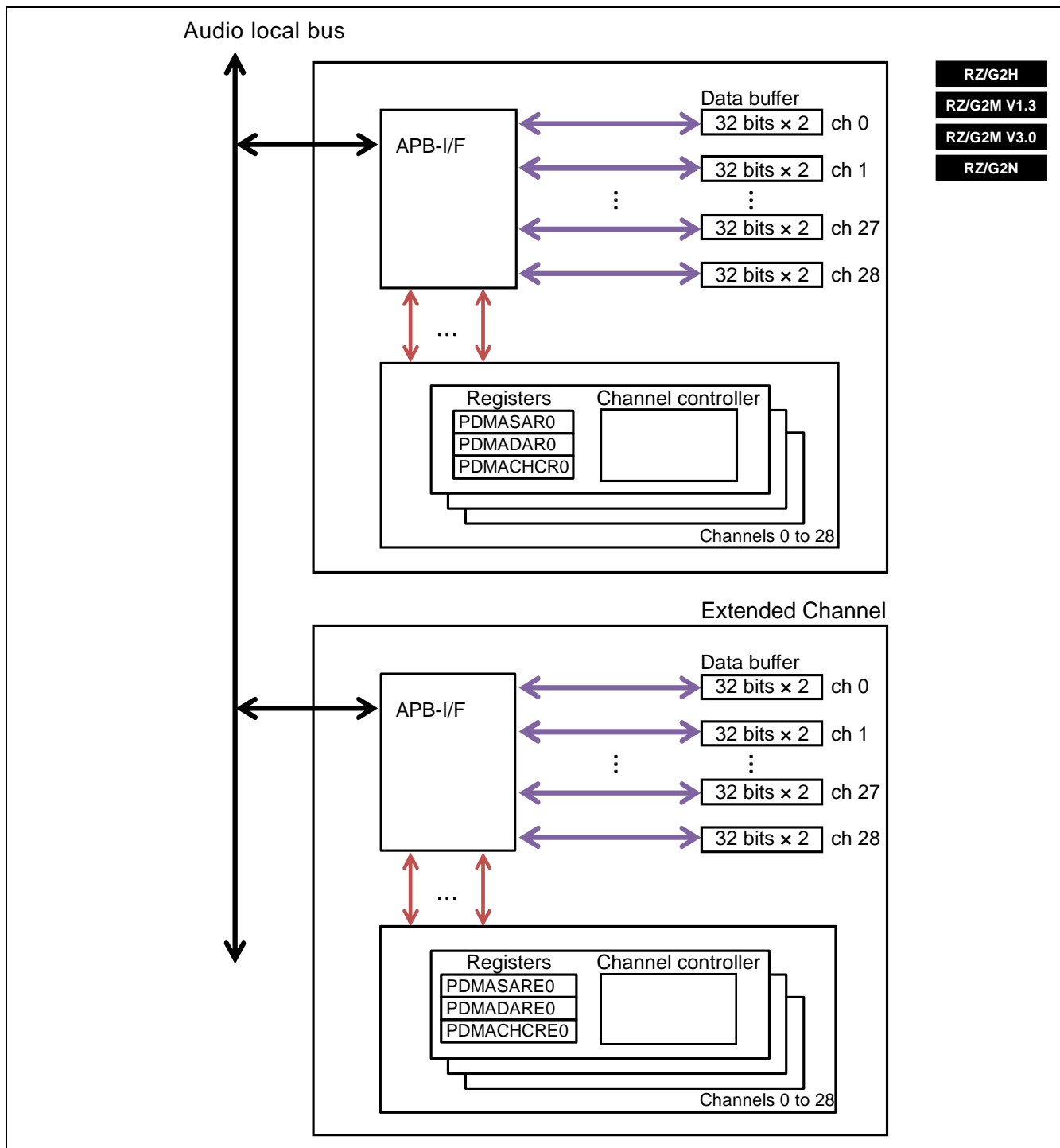


Figure 45.1 Block Diagram [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

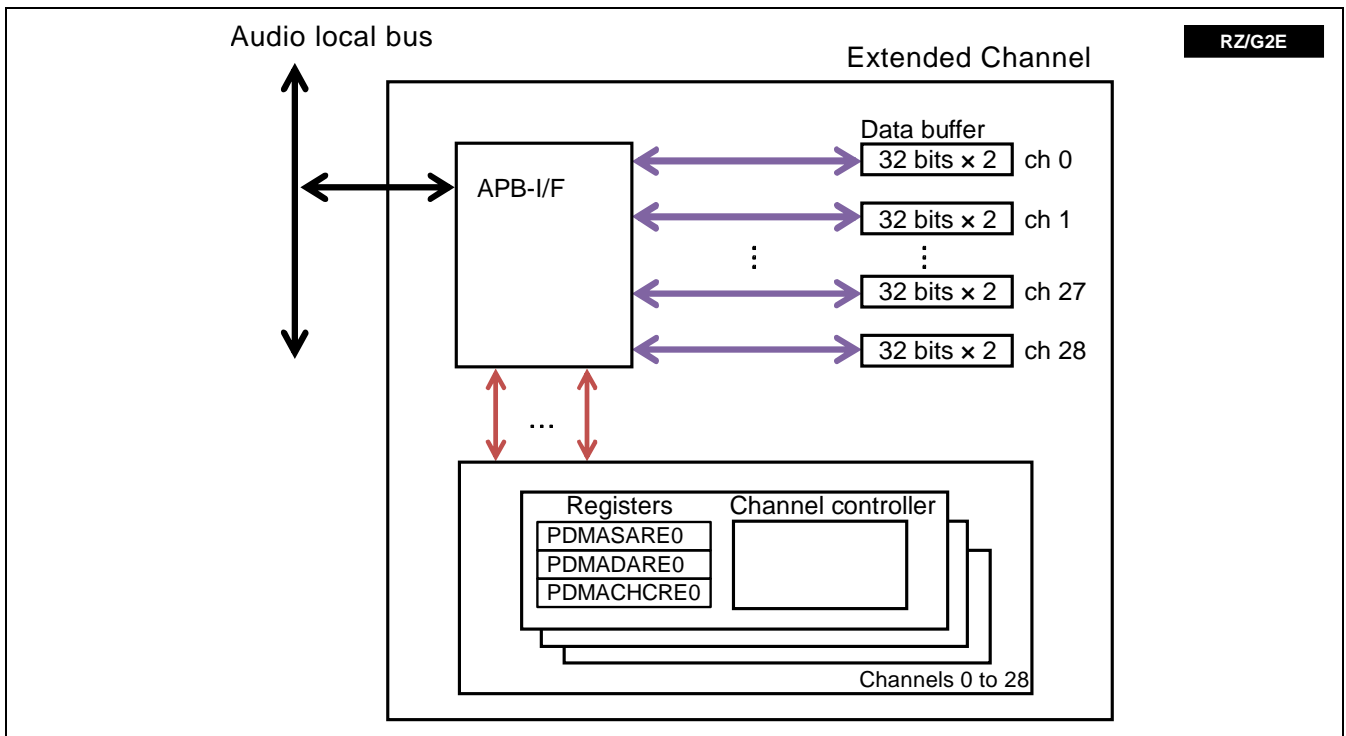


Figure 45.2 Block Diagram [RZ/G2E]

45.1.3 External Pins

No external pins are provided.

45.1.4 Register Configuration

Table 45.1 List of Registers

| Register Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|--|--------------|-----|--------------------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| PDMA Source Address n Register | PDMASARn | R/W | H'EC74_0020 + H'10 × [n] | H'0000_0000 | 32 | √ | √ | √ | — |
| PDMA Destination Address n Register | PDMADARn | R/W | H'EC74_0024 + H'10 × [n] | H'0000_0000 | 32 | √ | √ | √ | — |
| PDMA Channel Control n Register | PDMACHCRn | R/W | H'EC74_002C + H'10 × [n] | H'0000_0000 | 32 | √ | √ | √ | — |
| PDMA Source Address n Register (extended) | PDMASAREn | R/W | H'EC76_0020 + H'10 × [n] | H'0000_0000 | 32 | √ | √ | √ | √ |
| PDMA Destination Address n Register (extended) | PDMADAREn | R/W | H'EC76_0024 + H'10 × [n] | H'0000_0000 | 32 | √ | √ | √ | √ |
| PDMA Channel Control n Register (extended) | PDMACHCREn | R/W | H'EC76_002C + H'10 × [n] | H'0000_0000 | 32 | √ | √ | √ | √ |

- Notes:
1. N indicates the DMAC channel number (n = 0 to 28)
 2. Access the listed registers from the CPU in longword (32-bit) units; byte or word access is prohibited.
 3. Do not write to any addresses other than listed above. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed above are undefined.

45.1.5 Connected Module

Table 45.2 Connected module

| Module name | Connected module name | Function of connected module |
|----------------------------------|------------------------------|-------------------------------------|
| Audio DMAC-Peripheral-Peripheral | APMU | Access the Register |
| | CPG | Output Clocks |
| | SSIU | Serial Sound Interface Unit |
| | SSI | Serial Sound Interface |
| | SCU | Sampling Rate Converter Unit |

45.2 Register Description

[Legend]

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC1: Readable/writable. Writing 1 initializes the bit, and writing 0 is ignored.

R: Read-only. The write value should always be 0.

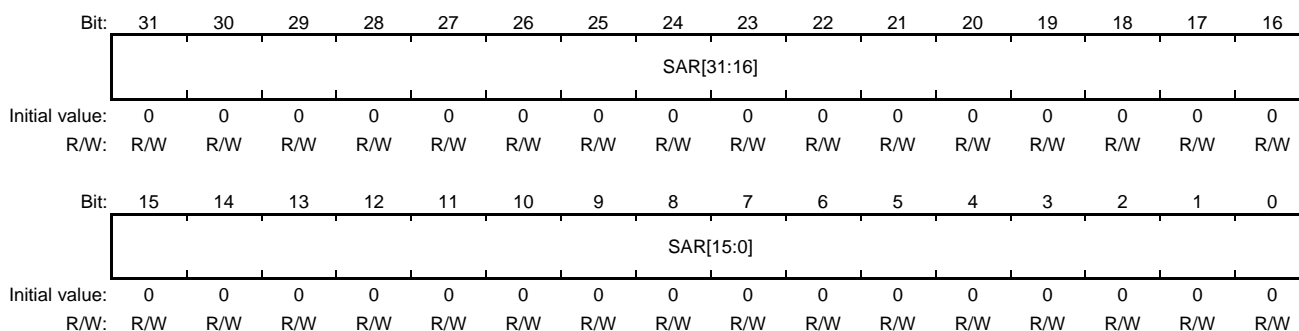
—/W: Write-only. The read value is undefined.

45.2.1 PDMA Source Address n Register (PDMASARn)

Note: n = 0 to 28

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: Sets the DMA transfer source address.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---------------------------------------|
| 31 to 0 | SAR | All 0 | R/W | Sets the DMA transfer source address. |

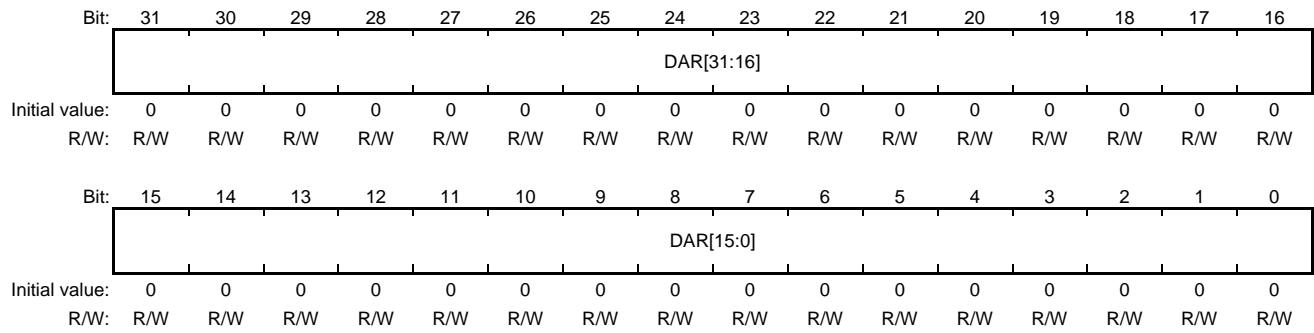
Note: It's prohibited to set same source address in other channels including PDMASARn and PDMASAREn.

45.2.2 PDMA Destination Address n Register (PDMADARn)

Note: n = 0 to 28

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: Sets the DMA transfer destination address.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | DAR | All 0 | R/W | Sets the DMA transfer destination address. |

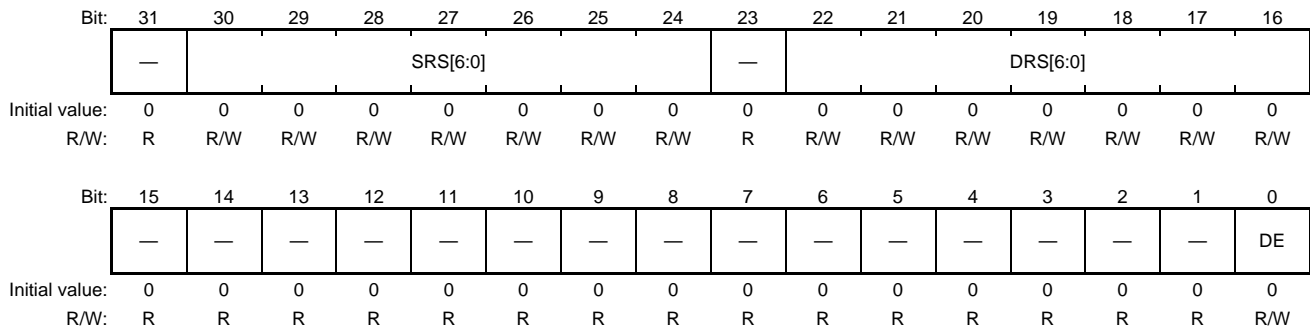
Note: It's prohibited to set same destination address in other channels including PDMADARn and PDMADAREn.

45.2.3 PDMA Channel Control n Register (PDMACHCRn)

Note: n = 0 to 28

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Function: Selects the request sources of DMA transfer source and destination, and also starts or stops DMA.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 30 to 24 | SRS | H'00 | R/W | Transfer Source Request Source Select (setting H'00 to H'38 and the values with “—” are prohibited) Refer to Table 45.3. |
| 23 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 22 to 16 | DRS | H'00 | R/W | Transfer Destination Request Source Select (setting H'00 to H'36 and the values with “—” are prohibited) Refer to Table 45.3. |
| 15 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | DE | B'0 | R/W | DMA Enable To enable DMA transfer function, set this bit to 1. Clearing this bit to 0 stops transfer. Note: Even if 0 is written to this bit while it is 1, 1 is read out until the transfer is completed. |

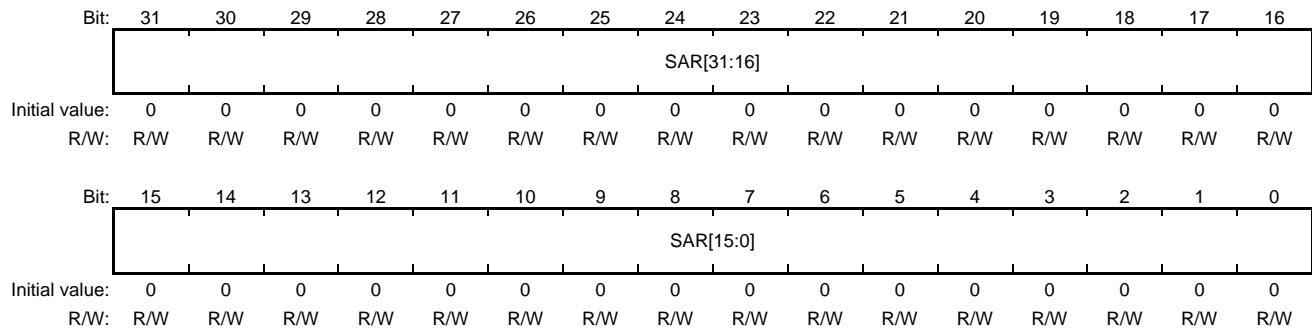
- Notes:
1. It's prohibited to set same source/destination request source in other channels including PDMACHCRn.SRS/DRS and PDMACHCREn.SRS/DRS.
 2. When clearing DE bit, do not change other bits.

45.2.4 PDMA Source Address n Register (extended) (PDMASAREn)

Note: n = 0 to 28

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: Sets the DMA transfer source address.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---------------------------------------|
| 31 to 0 | SAR | All 0 | R/W | Sets the DMA transfer source address. |

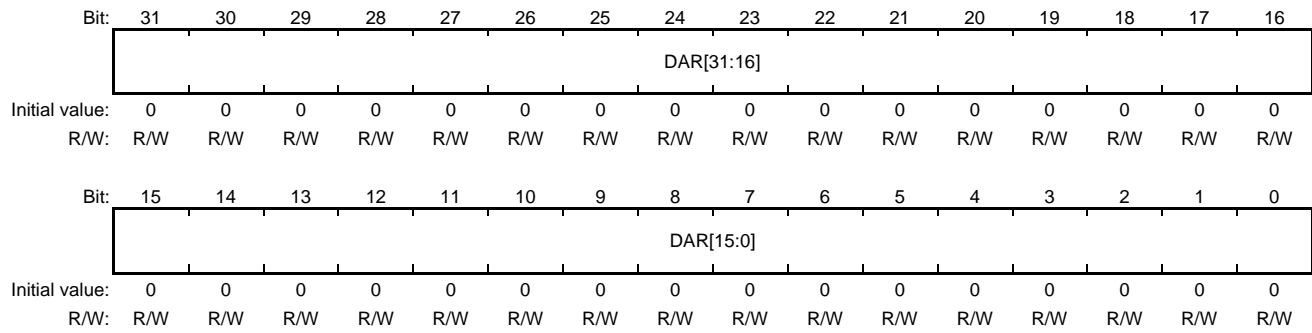
Note: It's prohibited to set same source address in other channels including PDMASARn and PDMASAREn.

45.2.5 PDMA Destination Address n Register (extended) (PDMADAREn)

Note: n = 0 to 28

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: Sets the DMA transfer destination address.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | DAR | All 0 | R/W | Sets the DMA transfer destination address. |

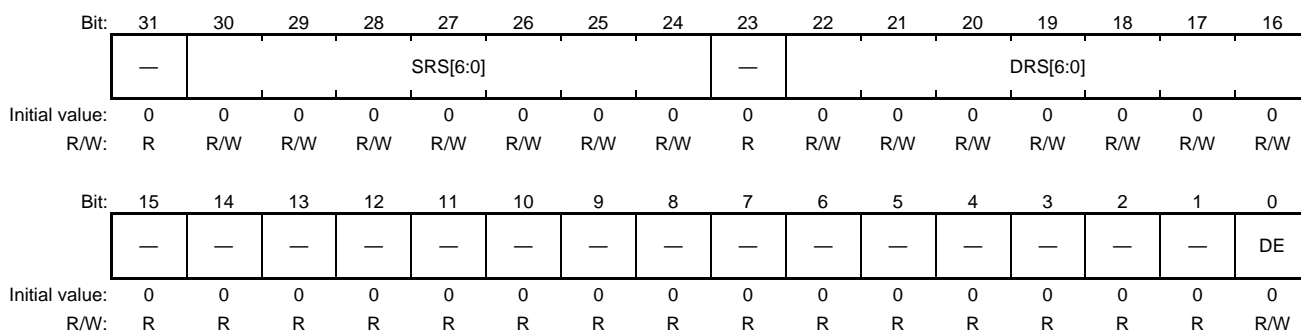
Note: It's prohibited to set same destination address in other channels including PDMADARn and PDMADAREn.

45.2.6 PDMA Channel Control n Register (extended) (PDMACHCREn)

Note: n = 0 to 28

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Function: Selects the request sources of DMA transfer source and destination, and also starts or stops DMA.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 30 to 24 | SRS | H'00 | R/W | Transfer Source Request Source Select (setting H'00 to H'5B and the values with “—” are prohibited) Refer to Table 45.4(extended). |
| 23 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 22 to 16 | DRS | H'00 | R/W | Transfer Destination Request Source Select (setting H'00 to H'5C and the values with “—” are prohibited) Refer to Table 45.4(extended). |
| 15 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | DE | B'0 | R/W | DMA Enable To enable DMA transfer function, set this bit to 1. Clearing this bit to 0 stops transfer. Note: Even if 0 is written to this bit while it is 1, 1 is read out until the transfer is completed. |

- Notes:
1. It's prohibited to set same source/destination request source in other channels including PDMACHCRn.SRS/DRS and PDMACHCREn.SRS/DRS.
 2. When clearing DE bit, do not change other bits.

45.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

45.3.1 Register Settings

Table 45.3 PDMASARn, PDMADARn and PDMACHCRn Register Settings

| | | | | | | | Second Generation RZ/G Series Products | | | |
|-----------------------|----------|------|-------------------|----------|------|-------------------|---|----------------------------|--------|--------|
| Source Destination | PDMASARn | | PDMACHC Rn.SRS | PDMADARn | | PDMACHC Rn.DRS | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI0-0_BUSIF | H'EC40 | 0000 | H'00 | H'EC40 | 0000 | H'00 | √ | √ | √ | — |
| SSI0-1_BUSIF | | 0400 | H'01 | | 0400 | H'01 | √ | √ | √ | — |
| SSI0-2_BUSIF | | 0800 | H'02 | | 0800 | H'02 | √ | √ | √ | — |
| SSI0-3_BUSIF | | 0C00 | H'03 | | 0C00 | H'03 | √ | √ | √ | — |
| SSI0-4_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI0-5_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI0-6_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI0-7_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI1-0_BUSIF | | 1000 | H'04 | | 1000 | H'04 | √ | √ | √ | — |
| SSI1-1_BUSIF | | 1400 | H'05 | | 1400 | H'05 | √ | √ | √ | — |
| SSI1-2_BUSIF | | 1800 | H'06 | | 1800 | H'06 | √ | √ | √ | — |
| SSI1-3_BUSIF | | 1C00 | H'07 | | 1C00 | H'07 | √ | √ | √ | — |
| SSI1-4_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI1-5_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI1-6_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI1-7_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI2-0_BUSIF | | 2000 | H'08 | | 2000 | H'08 | √ | √ | √ | — |
| SSI2-1_BUSIF | | 2400 | H'09 | | 2400 | H'09 | √ | √ | √ | — |
| SSI2-2_BUSIF | | 2800 | H'0A | | 2800 | H'0A | √ | √ | √ | — |
| SSI2-3_BUSIF | | 2C00 | H'0B | | 2C00 | H'0B | √ | √ | √ | — |
| SSI2-4_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI2-5_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI2-6_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI2-7_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI3-0_BUSIF | | 3000 | H'0C | | 3000 | H'0C | √ | √ | √ | — |
| SSI3-1_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI3-2_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI3-3_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI3-4_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI3-5_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI3-6_BUSIF | | — | — | | — | — | √ | √ | √ | — |
| SSI3-7_BUSIF | | — | — | | — | — | √ | √ | √ | — |

**Second Generation
RZ/G Series Products**

| Source Destination | PDMASARn | PDMACHC Rn.SRS | PDMACHC Rn.DRS | PDMADARn | PDMACHC Rn.DRS | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
|-----------------------|----------|-------------------|-------------------|----------|-------------------|--------|----------------------------|--------|--------|---|
| SSI4-0_BUSIF | H'EC40 | 4000 | H'0D | H'EC40 | 4000 | H'0D | √ | √ | √ | — |
| SSI4-1_BUSIF | — | — | — | — | — | — | √ | √ | √ | — |
| SSI4-2_BUSIF | — | — | — | — | — | — | √ | √ | √ | — |
| SSI4-3_BUSIF | — | — | — | — | — | — | √ | √ | √ | — |
| SSI4-4_BUSIF | — | — | — | — | — | — | √ | √ | √ | — |
| SSI4-5_BUSIF | — | — | — | — | — | — | √ | √ | √ | — |
| SSI4-6_BUSIF | — | — | — | — | — | — | √ | √ | √ | — |
| SSI4-7_BUSIF | — | — | — | — | — | — | √ | √ | √ | — |
| SSI5_BUSIF | 5000 | H'0E | — | 5000 | H'0E | — | √ | √ | √ | — |
| SSI6_BUSIF | 6000 | H'0F | — | 6000 | H'0F | — | √ | √ | √ | — |
| SSI7_BUSIF | 7000 | H'10 | — | 7000 | H'10 | — | √ | √ | √ | — |
| SSI8_BUSIF | 8000 | H'11 | — | 8000 | H'11 | — | √ | √ | √ | — |
| SSI9-0_BUSIF | 9000 | H'12 | — | 9000 | H'12 | — | √ | √ | √ | — |
| SSI9-1_BUSIF | 9400 | H'13 | — | 9400 | H'13 | — | √ | √ | √ | — |
| SSI9-2_BUSIF | 9800 | H'14 | — | 9800 | H'14 | — | √ | √ | √ | — |
| SSI9-3_BUSIF | 9C00 | H'15 | — | 9C00 | H'15 | — | √ | √ | √ | — |
| SSI9-4_BUSIF | — | — | — | — | — | — | √ | √ | √ | — |
| SSI9-5_BUSIF | — | — | — | — | — | — | √ | √ | √ | — |
| SSI9-6_BUSIF | — | — | — | — | — | — | √ | √ | √ | — |
| SSI9-7_BUSIF | — | — | — | — | — | — | √ | √ | √ | — |

| | | Second Generation RZ/G Series Products | | | | | | | | |
|-----------------------|----------|---|----------|-------------------|--------|----------------------------|--------|--------|---|---|
| Source Destination | PDMASARn | PDMACHC Rn.SRS | PDMADARn | PDMACHC Rn.DRS | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | | |
| SRC0in_BUSIF | H'EC30 | — | — | H'EC30 | 0000 | H'2D | √ | √ | √ | — |
| SRC1in_BUSIF | | — | — | | 0400 | H'2E | √ | √ | √ | — |
| SRC2in_BUSIF | | — | — | | 0800 | H'2F | √ | √ | √ | — |
| SRC3in_BUSIF | | — | — | | 0C00 | H'30 | √ | √ | √ | — |
| SRC4in_BUSIF | | — | — | | 1000 | H'31 | √ | √ | √ | — |
| SRC5in_BUSIF | H'EC30 | — | — | H'EC30 | 1400 | H'32 | √ | √ | √ | — |
| SRC6in_BUSIF | | — | — | | 1800 | H'33 | √ | √ | √ | — |
| SRC7in_BUSIF | | — | — | | 1C00 | H'34 | √ | √ | √ | — |
| SRC8in_BUSIF | | — | — | | 2000 | H'35 | √ | √ | √ | — |
| SRC9in_BUSIF | | — | — | | 2400 | H'36 | √ | √ | √ | — |
| SRC0out_BUSIF | | 4000 | H'2D | | — | — | √ | √ | √ | — |
| SRC1out_BUSIF | | 4400 | H'2E | | — | — | √ | √ | √ | — |
| SRC2out_BUSIF | | 4800 | H'2F | | — | — | √ | √ | √ | — |
| SRC3out_BUSIF | | 4C00 | H'30 | | — | — | √ | √ | √ | — |
| SRC4out_BUSIF | | 5000 | H'31 | | — | — | √ | √ | √ | — |
| SRC5out_BUSIF | | 5400 | H'32 | | — | — | √ | √ | √ | — |
| SRC6out_BUSIF | | 5800 | H'33 | | — | — | √ | √ | √ | — |
| SRC7out_BUSIF | | 5C00 | H'34 | | — | — | √ | √ | √ | — |
| SRC8out_BUSIF | | 6000 | H'35 | | — | — | √ | √ | √ | — |
| SRC9out_BUSIF | | 6400 | H'36 | | — | — | √ | √ | √ | — |
| CMD0out_BUSIF | | 8000 | H'37 | | — | — | √ | √ | √ | — |
| CMD1out_BUSIF | | 8400 | H'38 | | — | — | √ | √ | √ | — |

Table 45.4 PDMASAREn, PDMADAREn and PDMACHCREn Register Settings

| Source Destination | | | | | | | Second Generation RZ/G Series Products | | | |
|-----------------------|-----------|------|--------------------|-----------|------|--------------------|---|----------------------------|--------|--------|
| | PDMASAREn | | PDMACHC REn.SRS | PDMADAREn | | PDMACHC REn.DRS | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI0-0_BUSIF | H'EC40 | 0000 | H'00 | H'EC40 | 0000 | H'00 | √ | √ | √ | √ |
| SSI0-1_BUSIF | | 0400 | H'01 | | 0400 | H'01 | √ | √ | √ | √ |
| SSI0-2_BUSIF | | 0800 | H'02 | | 0800 | H'02 | √ | √ | √ | √ |
| SSI0-3_BUSIF | | 0C00 | H'03 | | 0C00 | H'03 | √ | √ | √ | √ |
| SSI0-4_BUSIF | | A000 | H'39 | | A000 | H'39 | √ | √ | √ | — |
| SSI0-5_BUSIF | | A400 | H'3A | | A400 | H'3A | √ | √ | √ | — |
| SSI0-6_BUSIF | | A800 | H'3B | | A800 | H'3B | √ | √ | √ | — |
| SSI0-7_BUSIF | | AC00 | H'3C | | AC00 | H'3C | √ | √ | √ | — |
| SSI1-0_BUSIF | | 1000 | H'04 | | 1000 | H'04 | √ | √ | √ | √ |
| SSI1-1_BUSIF | | 1400 | H'05 | | 1400 | H'05 | √ | √ | √ | √ |
| SSI1-2_BUSIF | | 1800 | H'06 | | 1800 | H'06 | √ | √ | √ | √ |
| SSI1-3_BUSIF | | 1C00 | H'07 | | 1C00 | H'07 | √ | √ | √ | √ |
| SSI1-4_BUSIF | | B000 | H'3D | | B000 | H'3D | √ | √ | √ | — |
| SSI1-5_BUSIF | | B400 | H'3E | | B400 | H'3E | √ | √ | √ | — |
| SSI1-6_BUSIF | | B800 | H'3F | | B800 | H'3F | √ | √ | √ | — |
| SSI1-7_BUSIF | | BC00 | H'40 | | BC00 | H'40 | √ | √ | √ | — |
| SSI2-0_BUSIF | | 2000 | H'08 | | 2000 | H'08 | √ | √ | √ | √ |
| SSI2-1_BUSIF | | 2400 | H'09 | | 2400 | H'09 | √ | √ | √ | √ |
| SSI2-2_BUSIF | | 2800 | H'0A | | 2800 | H'0A | √ | √ | √ | √ |
| SSI2-3_BUSIF | | 2C00 | H'0B | | 2C00 | H'0B | √ | √ | √ | √ |
| SSI2-4_BUSIF | | C000 | H'41 | | C000 | H'41 | √ | √ | √ | — |
| SSI2-5_BUSIF | | C400 | H'42 | | C400 | H'42 | √ | √ | √ | — |
| SSI2-6_BUSIF | | C800 | H'43 | | C800 | H'43 | √ | √ | √ | — |
| SSI2-7_BUSIF | | CC00 | H'44 | | CC00 | H'44 | √ | √ | √ | — |
| SSI3-0_BUSIF | | 3000 | H'0C | | 3000 | H'0C | √ | √ | √ | √ |
| SSI3-1_BUSIF | | 3400 | H'45 | | 3400 | H'45 | √ | √ | √ | √ |
| SSI3-2_BUSIF | | 3800 | H'46 | | 3800 | H'46 | √ | √ | √ | √ |
| SSI3-3_BUSIF | | 3C00 | H'47 | | 3C00 | H'47 | √ | √ | √ | √ |
| SSI3-4_BUSIF | | D000 | H'48 | | D000 | H'48 | √ | √ | √ | — |
| SSI3-5_BUSIF | | D400 | H'49 | | D400 | H'49 | √ | √ | √ | — |
| SSI3-6_BUSIF | | D800 | H'4A | | D800 | H'4A | √ | √ | √ | — |
| SSI3-7_BUSIF | | DC00 | H'4B | | DC00 | H'4B | √ | √ | √ | — |
| SSI4-0_BUSIF | | 4000 | H'0D | | 4000 | H'0D | √ | √ | √ | √ |
| SSI4-1_BUSIF | | 4400 | H'4C | | 4400 | H'4C | √ | √ | √ | √ |
| SSI4-2_BUSIF | | 4800 | H'4D | | 4800 | H'4D | √ | √ | √ | √ |

| Source Destination | Second Generation RZ/G Series Products | | | | | | | | | |
|-----------------------|---|------|--------------------|-----------|------|--------------------|--------|----------------------------|--------|--------|
| | PDMASAREn | | PDMACHC REn.SRS | PDMADAREn | | PDMACHC REn.DRS | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SSI4-3_BUSIF | H'EC40 | 4C00 | H'4E | H'EC40 | 4C00 | H'4E | √ | √ | √ | √ |
| SSI4-4_BUSIF | | E000 | H'4F | | E000 | H'4F | √ | √ | √ | — |
| SSI4-5_BUSIF | | E400 | H'50 | | E400 | H'50 | √ | √ | √ | — |
| SSI4-6_BUSIF | | E800 | H'51 | | E800 | H'51 | √ | √ | √ | — |
| SSI4-7_BUSIF | | EC00 | H'52 | | EC00 | H'52 | √ | √ | √ | — |
| SSI5_BUSIF | | 5000 | H'0E | | 5000 | H'0E | √ | √ | √ | √ |
| SSI6_BUSIF | | 6000 | H'0F | | 6000 | H'0F | √ | √ | √ | √ |
| SSI7_BUSIF | | 7000 | H'10 | | 7000 | H'10 | √ | √ | √ | √ |
| SSI8_BUSIF | | 8000 | H'11 | | 8000 | H'11 | √ | √ | √ | √ |
| SSI9-0_BUSIF | | 9000 | H'12 | | 9000 | H'12 | √ | √ | √ | √ |
| SSI9-1_BUSIF | | 9400 | H'13 | | 9400 | H'13 | √ | √ | √ | √ |
| SSI9-2_BUSIF | | 9800 | H'14 | | 9800 | H'14 | √ | √ | √ | √ |
| SSI9-3_BUSIF | | 9C00 | H'15 | | 9C00 | H'15 | √ | √ | √ | √ |
| SSI9-4_BUSIF | | F000 | H'53 | | F000 | H'53 | √ | √ | √ | — |
| SSI9-5_BUSIF | | F400 | H'54 | | F400 | H'54 | √ | √ | √ | — |
| SSI9-6_BUSIF | | F800 | H'55 | | F800 | H'55 | √ | √ | √ | — |
| SSI9-7_BUSIF | | FC00 | H'56 | | FC00 | H'56 | √ | √ | √ | — |
| SRC0in_BUSIF | H'EC30 | — | — | H'EC30 | 0000 | H'2D | √ | √ | √ | √ |
| SRC1in_BUSIF | | — | — | | 0400 | H'2E | √ | √ | √ | √ |
| SRC2in_BUSIF | | — | — | | 0800 | H'2F | √ | √ | √ | √ |
| SRC3in_BUSIF | | — | — | | 0C00 | H'30 | √ | √ | √ | √ |
| SRC4in_BUSIF | | — | — | | 1000 | H'31 | √ | √ | √ | √ |
| SRC5in_BUSIF | | — | — | | 1400 | H'32 | √ | √ | √ | √ |
| SRC6in_BUSIF | | — | — | | 1800 | H'33 | √ | √ | √ | √ |
| SRC7in_BUSIF | | — | — | | 1C00 | H'34 | √ | √ | √ | √ |

| Source Destination | PDMASAREn | | PDMACHC REn.SRS | | PDMADAREn | | PDMACHC REn.DRS | | Second Generation RZ/G Series Products | | | |
|-----------------------|-----------|------|--------------------|--------|-----------|------|--------------------|---|---|----------------------------|--------|--------|
| | | | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SRC8in_BUSIF | H'EC30 | — | — | H'EC30 | 2000 | H'35 | — | — | √ | √ | √ | √ |
| SRC9in_BUSIF | — | — | — | — | 2400 | H'36 | — | — | √ | √ | √ | √ |
| SRC0out_BUSIF | — | 4000 | H'2D | — | — | — | — | — | √ | √ | √ | √ |
| SRC1out_BUSIF | — | 4400 | H'2E | — | — | — | — | — | √ | √ | √ | √ |
| SRC2out_BUSIF | — | 4800 | H'2F | — | — | — | — | — | √ | √ | √ | √ |
| SRC3out_BUSIF | — | 4C00 | H'30 | — | — | — | — | — | √ | √ | √ | √ |
| SRC4out_BUSIF | — | 5000 | H'31 | — | — | — | — | — | √ | √ | √ | √ |
| SRC5out_BUSIF | — | 5400 | H'32 | — | — | — | — | — | √ | √ | √ | √ |
| SRC6out_BUSIF | — | 5800 | H'33 | — | — | — | — | — | √ | √ | √ | √ |
| SRC7out_BUSIF | — | 5C00 | H'34 | — | — | — | — | — | √ | √ | √ | √ |
| SRC8out_BUSIF | — | 6000 | H'35 | — | — | — | — | — | √ | √ | √ | √ |
| SRC9out_BUSIF | — | 6400 | H'36 | — | — | — | — | — | √ | √ | √ | √ |
| CMD0out_BUSI F | — | 8000 | H'37 | — | — | — | — | — | √ | √ | √ | √ |
| CMD1out_BUSI F | — | 8400 | H'38 | — | — | — | — | — | √ | √ | √ | √ |

45.3.2 Setting Flow

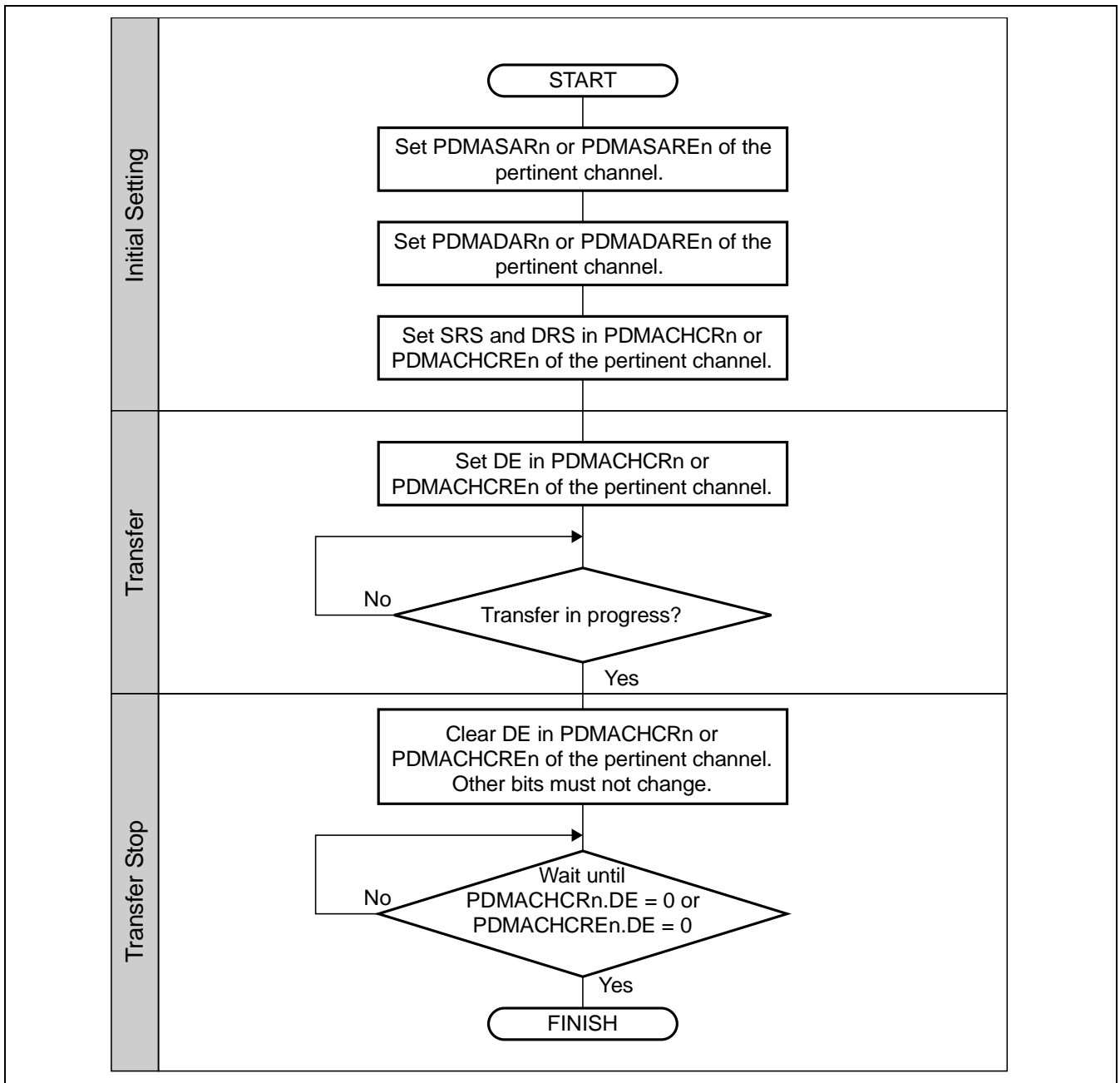


Figure 45.3 Setting Flow

45.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

There is no attention of the usage.

46. EthernetAVB-IF

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

46.1 Overview

The EthernetAVB-IF includes an Ethernet controller (E-MAC) that conforms to the definition of the MAC (Media Access Control) layer for Ethernet in the IEEE 802.3 standard.

When connected with a physical-layer LSI chip (PHY-LSI) that complies with the standard, the E-MAC is able to transmit and receive Ethernet (IEEE 802.3) frames. The E-MAC has a single MAC layer interface.

The EthernetAVB-IF has a dedicated direct memory access controller (AVB-DMAC) for transferring transmitted Ethernet frames to and received Ethernet frames from respective storage areas in the User RAM(URAM) at high speed.

The AVB-DMAC is compliant with the following three standards formulated for IEEE 802.1BA: the IEEE 802.1AS timing and synchronization protocol, the IEEE 802.1Qav real-time transfer, and the IEEE 802.1Qat stream reservation protocol.

In this section, URAM refers to the local RAM and external memory for the LSI.

46.1.1 Features

Table 46.1 lists the specifications of the EthernetAVB-IF module.

Table 46.1 Specifications (Functions)

| Item | Description |
|--|---|
| Protocol | Flow control conforming with the IEEE 802.3x standard |
| Data transmission and reception | Transmission and reception of Ethernet (IEEE 802.3) frames |
| Transfer speed | Supports transfer at 100 and 1000 Mbps |
| Mode | Full-duplex mode |
| Interface | Supports the RGMII (Reduced Gigabit Media Independent Interface) |
| Summary of the EthernetAVB-IF function | An intelligent frame separation DMAC (AVB-DMAC) conforming with the following standards stipulated for IEEE 802.1BA: IEEE 802.1AS (time synchronization protocol) IEEE 802.1Qav (real-time transfer) IEEE 1722 (AVTP presentation timestamp) IEEE 802.1Qat is supported by software. Descriptor management system Identification and sorting of frame data, and extraction and gathering of valid data Controllable interrupt frequency (reducing the load on the CPU) |
| Transmit/Receive FIFO | For transmission: 16Kbytes For reception: 8Kbytes |
| Max frame size | For transmission: 2047bytes For reception: 4092bytes |
| Magic Packet™ | Detection of Magic Packets™* and output of a detected signal |

Note: * Magic Packet™ is a trademark of Advanced Micro Devices, Inc.

46.1.2 Block Diagram

Figure 46.1, Figure 46.2, Figure 46.3 are block diagram of the EthernetAVB-IF.

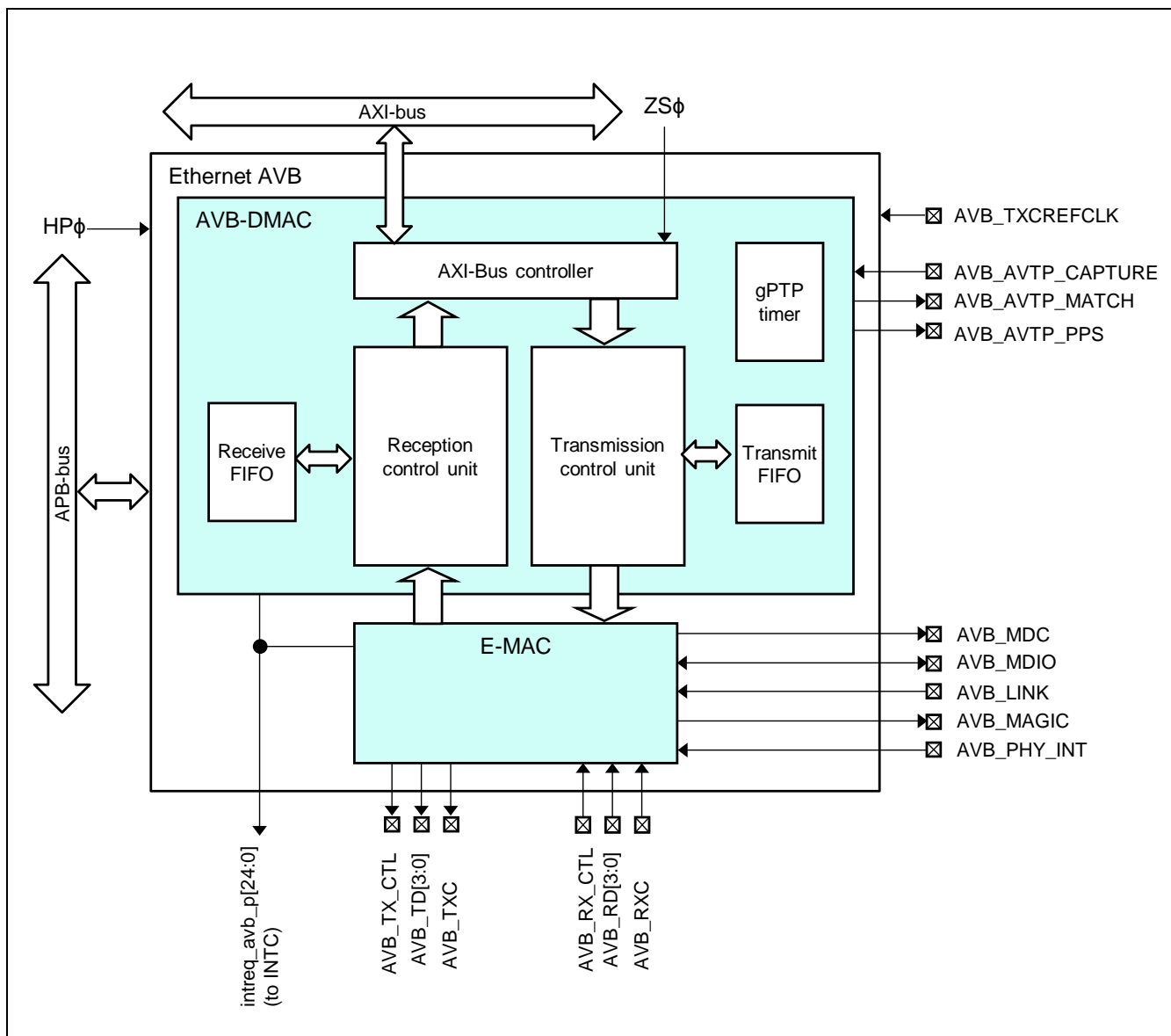


Figure 46.1 Block Diagram of EthernetAVB-IF

Table 46.2 Clock Specification

| Name | Function | Clock | Second Generation RZ/G Series Products | | | |
|-----------|-----------------------------|-------------|---|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| ZS ϕ | AXI bus clock | S0D3 ϕ | √ | √ | √ | — |
| | | S3D1 ϕ | — | — | — | √ |
| HP ϕ | High speed Peripheral clock | S0D6 ϕ | √ | √ | √ | — |
| | | S3D2 ϕ | — | — | — | √ |

46.1.3 External Pins

Table 46.3 Pin Configuration lists the pins for use by EthernetAVB-IF.

Table 46.3 Pin Configuration

| Name | I/O | Function | Second Generation RZ/G Series Products | | | |
|------------------|-----|--|---|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| AVB_TD[3:0] | O | Transmit data signal | √ | √ | √ | √ |
| AVB_TX_CTL | O | Transmit control signal | √ | √ | √ | √ |
| AVB_RD[3:0] | I | Receive data signal | √ | √ | √ | √ |
| AVB_RX_CTL | I | Receive control signal | √ | √ | √ | √ |
| AVB_RXC | I | Receive clock signal | √ | √ | √ | √ |
| AVB_TXC | O | transmit clock signal | √ | √ | √ | √ |
| AVB_TXCREFCLK | I | reference clock signal | √ | √ | √ | √ |
| AVB_MDC | O | Management information transfer clock signal | √ | √ | √ | √ |
| AVB_MDIO | I/O | Management information transmit/receive data | √ | √ | √ | √ |
| AVB_LINK | I | Link status signal | √ | √ | √ | √ |
| AVB_MAGIC | O | Magic packet signal | √ | √ | √ | √ |
| AVB_PHY_INT | I | PHY interrupt signal | √ | √ | √ | √ |
| AVB_AVTP_CAPTURE | I | AVTP capture signal | √ | √ | √ | √ |
| AVB_AVTP_MATCH | O | AVTP match signal | √ | √ | √ | √ |
| AVB_AVTP_PPS | O | AVTP Pulse-per-second signal | √ | √ | √ | √ |

46.1.4 Register Configuration

Table 46.4 and Table 46.5 lists the EthernetAVB-IF related registers and their configurations.

Table 46.4 Configurations of AVB-DMAC-related Registers

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|--|--------------|-----|------------------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| AVB-DMAC mode register | CCC | R/W | H'E680_0000 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Descriptor base address table register | DBAT | R/W | H'E680_0004 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Descriptor base address load request register | DLR | R/W | H'E680_0008 | H'003F_FFFF | 32 | √ | √ | √ | √ |
| AVB-DMAC status register | CSR | R | H'E680_000C | H'0000_0001 | 32 | √ | √ | √ | √ |
| Current descriptor address register q (q = 0 to 21) | CDARq | R | H'E680_0010 + q × 4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Error status register | ESR | R | H'E680_0088 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AVB-DMAC Product Specific Register | APSR | R/W | H'E680_008C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Receive configuration register | RCR | R/W | H'E680_0090 | H'1800_0000 | 32 | √ | √ | √ | √ |
| Receive queue configuration register i (i = 0 to 4) | RQCi | R/W | H'E680_0094 + i × 4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Receive padding configuration register | RPC | R/W | H'E680_00B0 | H'0000_0100 | 32 | √ | √ | √ | √ |
| Reception Truncation Configuration register | RTC | R/W | H'E680_00B4 | H'0FFC_0FFC | 32 | √ | √ | √ | √ |
| Unread frame counter warning level register | UFCW | R/W | H'E680_00BC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Unread frame counter stop level register | UFCS | R/W | H'E680_00C0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Unread frame counter register i (i = 0 to 4) | UFVi | R | H'E680_00C4 + i × 4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Unread frame counter decrement register i (i = 0 to 4) | UFCDi | R/W | H'E680_00E0 + i × 4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Separation filter offset register | SFO | R/W | H'E680_00FC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Separation filter pattern register i (i = 0 to 31) | SFPi | R/W | H'E680_0100 + i × 4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Separation Filter Value register i (i = 0 to 1) | SFVi | R/W | H'E680_01B8 + i × 4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Separation filter mask register i (i = 0, 1) | SFM i | R/W | H'E680_01C0 + i × 4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Separation Filter Load register | SFL | R/W | H'E680_01C8 | H'0000_001F | 32 | √ | √ | √ | √ |
| Current Incremental Address Register r (r = 0 to 17) | CIARr | R | H'E680_0200 + r × 4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Last Incremental Address Register r (r = 0 to 17) | LIARr | R | H'E680_0280 + r × 4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Transmit configuration register | TGC | R/W | H'E680_0300 | H'0022_2200 | 32 | √ | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|--------------|-----|------------------------|---------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Transmit configuration control register | TCCR | R/W | H'E680_0304 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Transmit status register | TSR | R | H'E680_0308 | H'0000_0000 | 32 | √ | √ | √ | √ |
| E-MAC status FIFO Access register | MFA | R | H'E680_030C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Time stamp FIFO access register 0 | TFA0 | R | H'E680_0310 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Time stamp FIFO access register 1 | TFA1 | R | H'E680_0314 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Time stamp FIFO access register 2 | TFA2 | R | H'E680_0318 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Version and Release Register | VRR | R | H'E680_031C | H'0000_E300 | 32 | √ | √ | √ | √ |
| CBS increment value register c (c = 0, 1) | CIVRc | R/W | H'E680_0320 + c × 4 | H'0000_0001 | 32 | √ | √ | √ | √ |
| CBS decrement value register c (c = 0, 1) | CDVRc | R/W | H'E680_0328 + c × 4 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |
| CBS upper limit register c (c = 0, 1) | CULc | R/W | H'E680_0330 + c × 4 | H'7FFF_FFFF | 32 | √ | √ | √ | √ |
| CBS lower limit register c (c = 0, 1) | CLLc | R/W | H'E680_0338 + c × 4 | H'8000_0001 | 32 | √ | √ | √ | √ |
| Descriptor interrupt control register | DIC | R/W | H'E680_0350 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Descriptor interrupt status register | DIS | R/W | H'E680_0354 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Error interrupt control register | EIC | R/W | H'E680_0358 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Error interrupt status register | EIS | R/W | H'E680_035C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Receive interrupt control register 0 | RIC0 | R/W | H'E680_0360 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Receive interrupt status register 0 | RIS0 | R/W | H'E680_0364 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Receive interrupt control register 1 | RIC1 | R/W | H'E680_0368 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Receive interrupt status register 1 | RIS1 | R/W | H'E680_036C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Receive interrupt control register 2 | RIC2 | R/W | H'E680_0370 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Receive interrupt status register 2 | RIS2 | R/W | H'E680_0374 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Transmit interrupt control register | TIC | R/W | H'E680_0378 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Transmit interrupt status register | TIS | R/W | H'E680_037C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Interrupt summary status register | ISS | R | H'E680_0380 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Common Interrupt Enable register | CIE | R/W | H'E680_0384 | H'0000_0000 | 32 | √ | √ | √ | √ |

| | | | | | | Second Generation RZ/G Series Products | | | |
|--|--------------|-----|------------------------|---------------|-------------|---|----------------------------|--------|--------|
| Name | Abbreviation | R/W | Address | Initial Value | Access Size | | | | |
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Reception Interrupt Control register 3 | RIC3 | R/W | H'E680_0388 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Reception Interrupt Status register 3 | RIS3 | R/W | H'E680_038C | H'0000_0000 | 32 | √ | √ | √ | √ |
| gPTP configuration control register | GCCR | R/W | H'E680_0390 | H'0000_002C | 32 | √ | √ | √ | √ |
| gPTP maximum transit time configuration register | GMTT | R/W | H'E680_0394 | H'0000_0000 | 32 | √ | √ | √ | √ |
| gPTP presentation time comparison register | GPTC | R/W | H'E680_0398 | H'0000_0000 | 32 | √ | √ | √ | √ |
| gPTP timer increment configuration register | GTI | R/W | H'E680_039C | H'0000_0001 | 32 | √ | √ | √ | √ |
| gPTP timer offset register i (i = 0 to 2) | GTOi | R/W | H'E680_03A0 + i × 4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| gPTP interrupt control register | GIC | R/W | H'E680_03AC | H'0000_0000 | 32 | √ | √ | √ | √ |
| gPTP interrupt status register | GIS | R/W | H'E680_03B0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| gPTP Captured Presentation Time register | GCPT | R | H'E680_03B4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| gPTP timer capture register i (i = 0 to 2) | GCTi | R/W | H'E680_03B8 + i × 4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Gtp Status Register | GSR | R | H'E680_03C4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Gtp Interrupt Enable register | GIE | R/W | H'E680_03CC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Gtp Interrupt Disable register | GID | R/W | H'E680_03D0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Gtp Interrupt Line selection register | GIL | R/W | H'E680_03D4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Gtp Avtp Capture Prescaler register | GACP | R/W | H'E680_03DC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Gtp Presentation Time Fifo register i (i = 0 to 3) | GPTFi | R/W | H'E680_03E0 + i × 4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Gtp Captured Avtp Time register i (i = 0 to 15) | GCATi | R | H'E680_0400 + i × 4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Descriptor Interrupt Line selection register | DIL | R/W | H'E680_0440 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Error Interrupt Line selection register | EIL | R/W | H'E680_0444 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Transmission Interrupt Line selection register | TIL | R/W | H'E680_0448 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Descriptor Interrupt Enable register | DIE | R/W | H'E680_0450 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Descriptor Interrupt Disable register | DID | R/W | H'E680_0454 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Error Interrupt Enable register | EIE | R/W | H'E680_0458 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Error Interrupt Disable register | EID | R/W | H'E680_045C | H'0000_0000 | 32 | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|--------------|-----|-------------|---------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Reception Interrupt Enable register 0 | RIE0 | R/W | H'E680_0460 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Reception Interrupt Disable register 0 | RID0 | R/W | H'E680_0464 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Reception Interrupt Enable register 1 | RIE1 | R/W | H'E680_0468 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Reception Interrupt Disable register 1 | RID1 | R/W | H'E680_046C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Reception Interrupt Enable register 2 | RIE2 | R/W | H'E680_0470 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Reception Interrupt Disable register 2 | RID2 | R/W | H'E680_0474 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Transmission Interrupt Enable register | TIE | R/W | H'E680_0478 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Transmission Interrupt Disable register | TID | R/W | H'E680_047C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Reception Interrupt Enable register 3 | RIE3 | R/W | H'E680_0488 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Reception Interrupt Disable register 3 | RID3 | R/W | H'E680_048C | H'0000_0000 | 32 | √ | √ | √ | √ |

Table 46.5 Configuration of E-MAC-related Registers

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|--|--------------|-----|-------------|---------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| E-MAC mode register | ECMR | R/W | H'E680_0500 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Receive frame length register | RFLR | R/W | H'E680_0508 | H'0000_0000 | 32 | √ | √ | √ | √ |
| E-MAC status register | ECSR | R/W | H'E680_0510 | H'0000_0000 | 32 | √ | √ | √ | √ |
| E-MAC interrupt permission register | ECSIPR | R/W | H'E680_0518 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PHY interface register | PIR | R/W | H'E680_0520 | H'0000_000X | 32 | √ | √ | √ | √ |
| PHY Status Register | PSR | R | H'E680_0528 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PHY_INT Polarity Register | PIPR | R/W | H'E680_052C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Automatic PAUSE frame register | APR | R/W | H'E680_0554 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Manual PAUSE frame register | MPR | R/W | H'E680_0558 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PAUSE frame transmit counter | PFTCR | R | H'E680_055C | H'0000_0000 | 32 | √ | √ | √ | √ |
| PAUSE frame receive counter | PFRCR | R | H'E680_0560 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Automatic PAUSE frame retransmit count register | TPAUSER | R/W | H'E680_0564 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PAUSE frame transmit times counter | PFTTCCR | R | H'E680_0568 | H'0000_0000 | 32 | √ | √ | √ | √ |
| E-MAC Mode Register 2 | GECMR | R/W | H'E680_05B0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| E-MAC address high register | MAHR | R/W | H'E680_05C0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| E-MAC address low register | MALR | R/W | H'E680_05C8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Transmit retry over counter register | TROCR | R/W | H'E680_0700 | H'0000_0000 | 32 | √ | √ | √ | √ |
| CRC error frame receive counter register | CEFCR | R/W | H'E680_0740 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Frame receive error counter register | FRECR | R/W | H'E680_0748 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Too-short frame receive counter register | TSFRCCR | R/W | H'E680_0750 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Too-long frame receive counter register | TLFRCCR | R/W | H'E680_0758 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Residual-bit frame receive counter register | RFCR | R/W | H'E680_0760 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Multicast address frame receive counter register | MAFCR | R/W | H'E680_0778 | H'0000_0000 | 32 | √ | √ | √ | √ |

46.1.5 Connected Module

Table 46.6 Connected module

| Module name | Connected module name | Function of connected module |
|--------------------|------------------------------|-------------------------------------|
| EthernetAVB-IF | AP-System Core | Access the Registers |
| | CPG | Output clocks |
| | PFC | Select the External pins |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |
| | INTC-SYS | Control to interrupt |
| | ADG | Output capture signal |
| | DU | Output capture signal |

46.2 Register Description

46.2.1 AVB-DMAC Mode Register (CCC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CCC register specifies the operating mode of the AVB-DMAC.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|------|-----|-----|-----|-----|-----|-----|-----|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | FCE | LBME | — | — | — | — | — | — | — | CSEL[1:0] |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | DTSR | GAC | — | — | — | — | — | — | OPC[1:0] |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

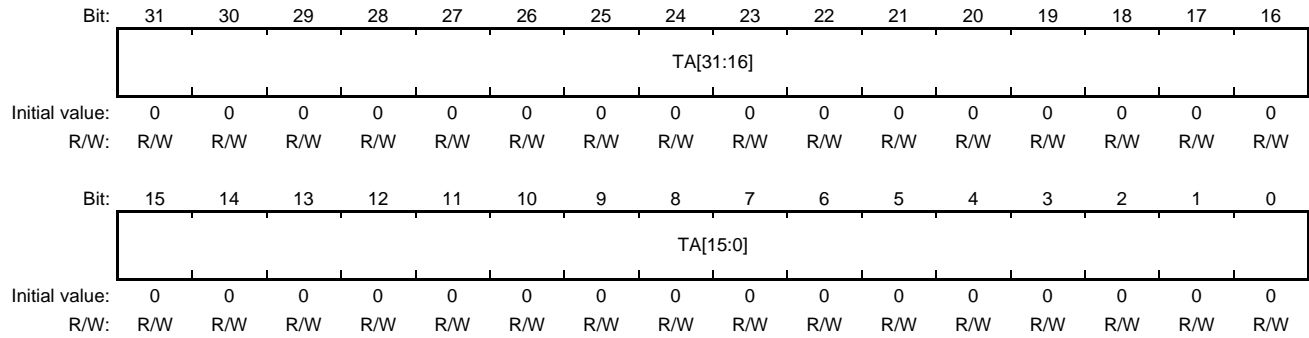
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 26 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 25 | FCE | B'0 | R/W | Flow Control Enable This bit enables the flow control support of E-MAC. When flow control is enabled, the E-MAC gets informed about the Rx-FIFO level (Rx-FIFO fill level reached RCR.RFCL). 0: Flow control disabled 1: Flow control enabled |
| 24 | LBME | B'0 | R/W | Loopback Mode Enable This bit enables loopback mode. In loopback mode, the transmission lines are internally connected to the reception lines. When loopback mode is to be used, the Ethernet transmission clock must be supplied to the RGMII interface. A received clock signal is not required. Writing to this bit is only possible when the current operating mode is configuration mode. 0: Normal operation 1: Loopback mode is enabled. Note: Data for transmission are still output normally. To eliminate effects on external modules, pin control should be applied to block the output of data. |
| 23 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 17, 16 | CSEL[1:0] | B'00 | R/W | <p>gPTP Clock Select</p> <p>These bits select the clock source for the gPTP timer.</p> <p>Writing to these bits is only possible when the current operating mode is configuration mode and CCC.GAC is 0.</p> <p>Writing to these bits is only possible when the current operating mode is Reset mode when writing CCC.GAC to 1 and CCC.OPC to B'01 by same write access.</p> <p>B'00: gPTP is disabled</p> <p>B'01: High-speed peripheral bus clock</p> <p>B'10 and B'11: Setting prohibited</p> |
| 15 to 9 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 8 | DTSR | B'0 | R/W | <p>Data Transmission Suspend Request</p> <p>This bit can suspend access to the URAM.</p> <p>The access is suspended on completion of the transfer of the AXI transaction currently being transferred.</p> <p>This function disables access to the URAM without affecting normal operation of the AVB-DMAC. Use this bit when exclusive control over the contents of the URAM is necessary, for example, in checking its integrity.</p> <p>Note that the transmission and reception queues are not processed while access is suspended.</p> <p>Change neither the AVB-DMAC settings nor the mode while access is suspended.</p> <p>0: Normal operation</p> <p>1: Requests suspension</p> |
| 7 | GAC | B'0 | R/W | <p>Gptp Active in Config</p> <p>This bit enables the gPTP support of EthernetAVB-IF in CONFIG mode. Function of gPTP support in OPERATION and STANDBY is not influenced by this bit. When gPTP support is active in CONFIG, CCC.CSEL defines the timer clock source.</p> <p>The CPU can only write 1 to this bit if CSR.OPS is RESET when writing CCC.OPC to B'01 by same write access.</p> <p>0: Normal operation</p> <p>1: gPTP support active in CONFIG mode</p> |
| 6 to 2 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 1, 0 | OPC[1:0] | B'00 | R/W | <p>Operating Mode Configuration</p> <p>These bits specify the operating mode.</p> <p>For the operating modes, see section 46.3.1(1) Operating Modes.</p> <p>Writing to these bits is possible in any of the operating modes, but should not be done after the application system has issued a Power Off request.</p> <p>When CCC.GAC is 1 CPU should not write B'11 to these bits.</p> <p>[Changing condition]</p> <p>These bits are set to B'00 when RESET mode is entered due to 'enter power off' request.</p> <p>B'00: Reset mode</p> <p>B'01: Configuration mode</p> <p>B'10: Operation mode</p> <p>B'11: standby mode</p> |

46.2.2 Descriptor Base Address Table Register (DBAT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The DBAT register specifies the base address of the descriptor table in the URAM. For the structure of this table, see section 46.3.3 Descriptors. Writing to this bit is only possible when the current operating mode is configuration mode.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | TA[31:0] | H'0000_0000 | R/W | Descriptor Base Table Address Base address of the descriptor table in the URAM Note: The setting of this bit must be a multiple of four (i.e. bit 0 and bit 1 must be set to 0). |

46.2.3 Descriptor Base Address Load Request Register (DLR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The DLR register is used to issue a request to load the values from the current descriptor address register q (CDAR $_q$) for each queue to the descriptor base address table register (DBAT).

Setting a bit to 1 issues a request for loading the descriptor base address for the queue q . If transfer is currently in progress, loading is executed on completion of transfer for the current frame. Completion of loading leads to automatic setting of the corresponding bit to 0.

For the transmission queues, base address load requests are executed even while fetching is in progress (the transmit start request bit in the transmit configuration control register (TCCR.TSRQt) is 1). Therefore, be sure to check that fetching is not in progress before issuing a request.

Writing to a bit of this register is only possible when the current operating mode is operation mode. Only 1 can be written to this bit.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | LBA21 | LBA20 | LBA19 | LBA18 | LBA17 | LBA16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | LBA15 | LBA14 | LBA13 | LBA12 | LBA11 | LBA10 | LBA9 | LBA8 | LBA7 | LBA6 | LBA5 | LBA4 | LBA3 | LBA2 | LBA1 | LBA0 |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 22 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 21 | LBA21 | B'1 | R/W | Base Address Load Request (Rx17: Stream 15) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 20 | LBA20 | B'1 | R/W | Base Address Load Request (Rx16: Stream 14) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 19 | LBA19 | B'1 | R/W | Base Address Load Request (Rx15: Stream 13) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 18 | LBA18 | B'1 | R/W | Base Address Load Request (Rx14: Stream 12) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 17 | LBA17 | B'1 | R/W | Base Address Load Request (Rx13: Stream 11) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 16 | LBA16 | B'1 | R/W | Base Address Load Request (Rx12: Stream 10) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 15 | LBA15 | B'1 | R/W | Base Address Load Request (Rx11: Stream 9) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 14 | LBA14 | B'1 | R/W | Base Address Load Request (Rx10: Stream 8) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 13 | LBA13 | B'1 | R/W | Base Address Load Request (Rx9: Stream 7) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 12 | LBA12 | B'1 | R/W | Base Address Load Request (Rx8: Stream 6) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 11 | LBA11 | B'1 | R/W | Base Address Load Request (Rx7: Stream 5) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 10 | LBA10 | B'1 | R/W | Base Address Load Request (Rx6: Stream 4) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 9 | LBA9 | B'1 | R/W | Base Address Load Request (Rx5: Stream 3) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 8 | LBA8 | B'1 | R/W | Base Address Load Request (Rx4: Stream 2) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | LBA7 | B'1 | R/W | Base Address Load Request (Rx3: Stream 1) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 6 | LBA6 | B'1 | R/W | Base Address Load Request (Rx2: Stream 0) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 5 | LBA5 | B'1 | R/W | Base Address Load Request (Rx1: Network Control) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 4 | LBA4 | B'1 | R/W | Base Address Load Request (Rx0: Best Effort) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 3 | LBA3 | B'1 | R/W | Base Address Load Request (Tx3: Stream Class A) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 2 | LBA2 | B'1 | R/W | Base Address Load Request (Tx2: Stream Class B) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 1 | LBA1 | B'1 | R/W | Base Address Load Request (Tx1: Network Control) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |
| 0 | LBA0 | B'1 | R/W | Base Address Load Request (Tx0: Best Effort) 0: No load request is issued. 1: When written: A request for loading the corresponding base address is issued. When read: The given base address is being loaded. |

46.2.4 AVB-DMAC Status Register (CSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CSR register is used to indicate the operating mode in which the AVB-DMAC is running and the individual communications states.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----|----|----|------|-----|----------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | TDUO | RPO | TPO3 | TPO2 | TPO1 | TPO0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | DTS | — | — | — | — | OPS[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 22 | — | All 0 | R | Reserved These bits are read as 0. |
| 21 | TDUO | B'0 | R | <p>Transmission Descriptor Update On-going</p> <p>This bit indicates that there is pending descriptor update for one or more transmit queues.</p> <p>When this bit is 1, there are descriptors of already or currently processed transmit storage elements not updated in URAM.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when all transmit queue related storage elements processed.</p> <p>This bit is set to 1 when descriptor related status update procedure starts (descriptor update and register flagging).</p> <p>0: No pending descriptor update 1: Pending descriptor update in URAM</p> |
| 20 | RPO | B'0 | R | <p>Receive Process Status</p> <p>This bit indicates whether a reception queue contains an unread received frame.</p> <p>This bit being set to 1 indicates that a received frame is yet to be stored in the URAM.</p> <p>0: Normal operation</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> — The current operating mode is not operation mode. — Received frames in the reception FIFO all being stored in the URAM. <p>1: Reception is in progress.</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> — A received frame being stored in the reception FIFO (but not yet in the URAM). |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 19 | TPO3 | B'0 | R | <p>Transmit Process Status 3 (Stream Class A)</p> <p>This bit indicates whether a class A stream is being transmitted. This bit being set to 1 indicates that the AVB-DMAC is fetching data for transmission from the URAM, or the E-MAC is transmitting data.</p> <p>0: Normal operation [Clearing conditions]</p> <ul style="list-style-type: none"> — The current operating mode is not operation mode. — Completion of transfer of all frames for transmission from the transmission FIFO and TCCR.TSRQ3 is 0. <p>1: Transmission is in suspended. [Setting condition]</p> <ul style="list-style-type: none"> — Transmission being started (by writing 1 to the transmit start request bit in the transmit configuration control register (TCCR.TSRQ3)). |
| 18 | TPO2 | B'0 | R | <p>Transmit Process Status 2 (Stream Class B)</p> <p>This bit indicates whether a class B stream is being transmitted. This bit being set to 1 indicates that the AVB-DMAC is fetching data for transmission from the URAM, or the E-MAC is transmitting data.</p> <p>0: Normal operation [Clearing conditions]</p> <ul style="list-style-type: none"> — The current operating mode is not operation mode. — Completion of transfer of all frames for transmission from the transmission FIFO and TCCR.TSRQ2 is 0. <p>1: Transmission is in suspended. [Setting condition]</p> <ul style="list-style-type: none"> — Transmission being started (by writing 1 to the transmit start request bit in the transmit configuration control register (TCCR.TSRQ2)). |
| 17 | TPO1 | B'0 | R | <p>Transmit Process Status 1 (Network Control)</p> <p>This bit indicates whether a network control is being transmitted. This bit being set to 1 indicates that the AVB-DMAC is fetching data for transmission from the URAM, or the E-MAC is transmitting data.</p> <p>0: Normal operation [Clearing conditions]</p> <ul style="list-style-type: none"> — The current operating mode is not operation mode. — Completion of transfer of all frames for transmission from the transmission FIFO and TCCR.TSRQ1 is 0. <p>1: Transmission is in suspended. [Setting condition]</p> <ul style="list-style-type: none"> — Transmission being started (by writing 1 to the transmit start request bit in the transmit configuration control register (TCCR.TSRQ1)). |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 16 | TPO0 | B'0 | R | <p>Transmit Process Status 0 (Best Effort)</p> <p>This bit indicates whether a best effort is being transmitted.</p> <p>This bit being set to 1 indicates that the AVB-DMAC is fetching data for transmission from the URAM, or the E-MAC is transmitting data.</p> <p>0: Normal operation [Clearing conditions]</p> <ul style="list-style-type: none"> — The current operating mode is not operation mode. — Completion of transfer of all frames for transmission from the transmission FIFO and TCCR.TSRQ0 is 0. <p>1: Transmission is in suspended. [Setting condition]</p> <ul style="list-style-type: none"> — Transmission being started (by writing 1 to the transmit start request bit in the transmit configuration control register (TCCR.TSRQ0)). |
| 15 to 9 | — | All 0 | R | <p>Reserved</p> <p>These bits are read as 0.</p> |
| 8 | DTS | B'0 | R | <p>Data Transmission Suspended Status</p> <p>This bit indicates whether access to the URAM is enabled.</p> <p>0: Normal operation [Clearing conditions]</p> <ul style="list-style-type: none"> — The data transmission suspend request bit in the AVB-DMAC mode register (CCC.DTSR) being 0. <p>1: Transmission is in suspended. [Setting condition]</p> <ul style="list-style-type: none"> — Access to the URAM not proceeding while the data transmission suspend request bit (CCC.DTSR) in the AVB-DMAC mode register (CCC) is 1 (if the URAM is being accessed, this bit is set to 1 on completion of access). |
| 7 to 4 | — | All 0 | R | <p>Reserved</p> <p>These bits are read as 0.</p> |
| 3 to 0 | OPS[3:0] | B'0001 | R | <p>Operating Mode Status</p> <p>These bits indicate the current operating mode.</p> <p>For the operating modes, see section 46.3.1(1) Operating Modes.</p> <p>B'0001: Reset mode B'0010: Configuration mode B'0100: Operation mode B'1000: standby mode</p> <p>Other settings are reserved.</p> |

46.2.5 Current Descriptor Address Register q (CDARq) (q = 0 to 21)

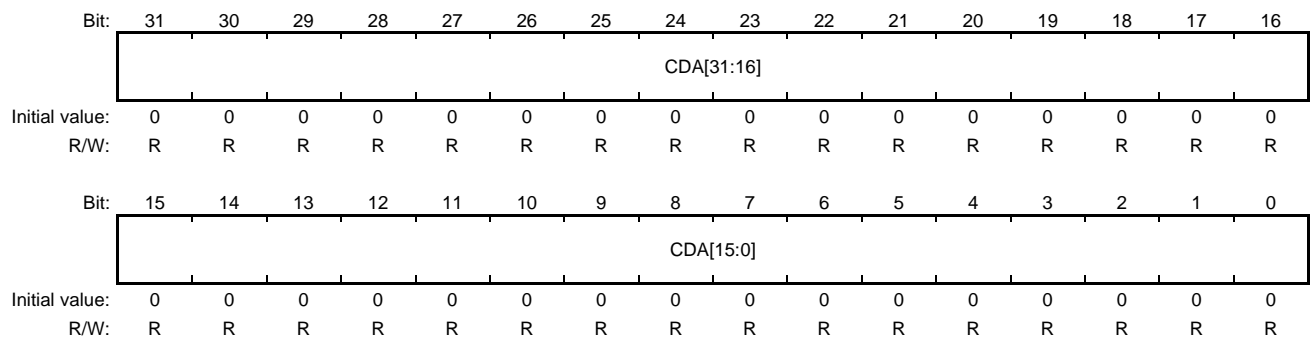
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CDARq register indicates the current descriptor address.

CDAR0 to CDAR3 indicate the addresses of the current descriptors for the corresponding transmission queues while CDAR4 to CDAR21 indicate the addresses of the current descriptors for the corresponding reception queues.

If the operating mode is changed to operation mode, the contents of the register for the queue to be used are set in the descriptor base address table register (DBAT).

Also, when the descriptor base address load request register (DLR) issues a load request, the contents of the descriptor base address table register (DBAT) are set in to the corresponding CDAR registers.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | CDA[31:0] | H'0000_0000 | R | <p>Current Descriptor Address</p> <p>The address of the current descriptors for the transmission queues.</p> <p>Conditions for updating:</p> <ul style="list-style-type: none"> These bits are set to 0 when the operating mode is not operation mode. This register is updated in response to processing of the descriptor for a queue. |

46.2.6 Error Status Register (ESR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----|---------|----|----|----|----|----|----------|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | EIL | ET[3:0] | | | — | — | — | EQN[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 13 | — | All 0 | R | Reserved These bits are read as 0. |
| 12 | EIL | B'0 | R | <p>Error Information Lost</p> <p>This bit indicates that error information detect by EthernetAVB-IF is lost, because the previous reported error has not been processed by CPU.</p> <p>[Changing conditions]</p> <ul style="list-style-type: none"> — This bit is set to 0 when leaving operation mode. — This bit is set to 0 when CPU writes 0 to EIS.QEF. — This bit is set to 1 when the set condition of EIS.QEF is fulfilled while EIS.QEF is 1. <p>0: No loss of error information 1: Lost of error information detected</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 11 to 8 | ET[3:0] | B'0000 | R | <p>Error Type</p> <p>These bits indicate details about the transfer stage which was handled when EthernetAVB-IF has detected an error.</p> <p>When the fault is related to the read descriptor (ESR.ET = B'0000 or B'0010), CPU needs to correct the faulty descriptor before the related queue can continue processing. Because the queue halts at the faulty descriptor CDARq.CDA (q = ESR.EQN) is identifying the faulty descriptor directly.</p> <p>When the fault is related to descriptor writing (ESR.ET = B'0001), CPU needs to recognize the not-updated or incorrectly updated descriptor in queue ESR.EQN. The write problem is not influencing how EthernetAVB-IF processes the descriptor chain.</p> <p>When the fault is related to the Tx-buffer (ESR.ET = B'0011), CPU needs to clean-up the Tx-buffer to correct the buffer control structures.</p> <p>All other errors are transient in nature and may be corrected by continuation of hardware or software operation; so there is no strong demand on CPU interaction.</p> <p>Refer to Section 46.3.2(2) Checking Integrity for details of error handling.</p> <p>The CPU should only evaluate these bits when the EIS.QEF bit is 1.</p> <p>[Changing condition]</p> <ul style="list-style-type: none"> — These bits are updated when the set condition of the EIS.QEF bit is fulfilled and the EIS.QEF bit is 0. <p>B'0000: Read descriptor from URAM B'0001: Write descriptor to URAM B'0010: Interpret data descriptor B'0011: Tx-buffer is corrupted B'0100: Read data from URAM B'0101: Write data or timestamp to URAM B'0110: Reading from Rx-FIFO B'0111: Rx-FIFO is corrupted B'1000: Frame size error during reception detected B'1001: Frame size error during transmission detected B'1010: Tx-buffer overflow B'1011: AVTP FIFO error</p> |
| 7 to 5 | — | All 0 | R | <p>Reserved</p> <p>These bits are read as 0.</p> |
| 4 to 0 | EQN[4:0] | B'0_0000 | R | <p>Error Queue Number</p> <p>These bits indicate the queue number which was handled when EthernetAVB-IF has detected an error.</p> <p>A fault reported for ESR.EQN = 0 to 3 is related to transmit queue t (t = 0 to 3).</p> <p>From ESR.EQN = 4, the fault is related to receive queue r (r = ESR.EQN – 4).</p> <p>The CPU should only evaluate these bits when the EIS.QEF bit is 1. The CPU should not evaluate these bits when the ESR.ET bit is B'0011, B'0111 or B'1011.</p> <p>[Changing condition]</p> <ul style="list-style-type: none"> — These bits are updated when the set condition of the EIS.QEF bit is fulfilled and the EIS.QEF bit is 0. |

46.2.7 AVB-DMAC Product Specific Register (APSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Writing to this bit is only possible when the current operating mode is configuration mode.

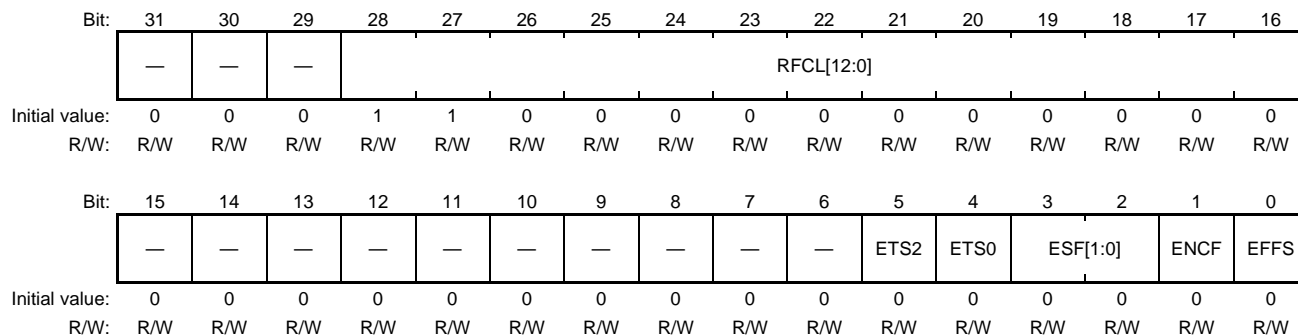
| | | | | | | | | | | | | | | | | |
|----------------|-----|-------|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | —/TDM | RDM | — | — | — | — | — | — | — | — | CMSW | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 20 | — | All 0 | R/W | Reserved These bits are read as 0. |
| 19 to 15 | — | All 0 | R/W | Reserved These bits are read as 0. |
| 14 | — | B'0 | R/W | Reserved [RZ/G2E] These bits are read as 0. |
| | TDM | B'0 | R/W | Tx clock internal Delay Mode [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] This bit can add internal Tx clock delay typ 2.0ns*. 0: normal mode 1: Setting prohibited * Refer to Electrical Characteristics for details. |
| 13 | RDM | B'0 | R/W | Rx clock internal Delay Mode This bit can add internal Rx clock delay typ 1.8ns*. 0: normal mode 1: delayed mode * Refer to Electrical Characteristics for details. |
| 12 to 5 | — | All 0 | R/W | Reserved These bits are read as 0. |
| 4 | CMSW | B'0 | R/W | select width of internal compare match signal 0: 1clock period pulse 1: 4clock period pulse |
| 3 to 0 | — | B'0000 | R/W | Reserved These bits are read as 0. |

46.2.8 Receive Configuration Register (RCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RCR register is used to make settings related to reception for the AVB-DMAC.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 29 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 28 to 16 | RFCL[12:0] | H'1800 | R/W | Receive FIFO Caution Level These bits set the caution level for the reception FIFO and are used to maintain the priority order of the storage of received data and the fetching of data for transmission. If the reception FIFO contains less data than this level, processing of both transmission and reception queues becomes pending. If the reception FIFO contains more data than this level, only data in the reception queue are transferred, and processing of the transmission queue becomes pending. Writing to this bit is only possible when the current operating mode is configuration mode. Recommended value: H'1800 Notes: <ul style="list-style-type: none"> The setting of this bit must be a multiple of four (i.e. set RFCL[1:0] = B'00). In the case of this LSI chip, set these bits to H'1800. |
| 15 to 6 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 5 | ETS2 | B'0 | R/W | Time Stamp Enable (Stream) Enables the inclusion of time-stamp information in reception queues 2 to 17. Writing to this bit is only possible when the current operating mode is configuration mode. 0: Time stamping is disabled. 1: Time stamping is enabled. Recommended value: 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 4 | ETS0 | B'0 | R/W | <p>Time Stamp Enable (Best Effort)</p> <p>Enables the inclusion of time-stamp information in reception queue 0.</p> <p>Writing to this bit is only possible when the current operating mode is configuration mode.</p> <p>0: Time stamping is disabled.</p> <p>1: Time stamping is enabled.</p> <p>Recommended value: 0</p> |
| 3, 2 | ESF[1:0] | B'00 | R/W | <p>Stream Filtering Select</p> <p>Settings for reception queues 2 to 17.</p> <p>These bits select separation filtering for reception queues 2 to 17. The queue-dependent separation filter can be used in combination with the identification of AVB stream frames.</p> <p>When the value is B'00, filtering is disabled and frames from streams are processed in reception queue 0 (best effort).</p> <p>When the value is B'01, the separation filter is enabled for both AVB stream frames and non-AVB stream frames; frames from non-matching streams are processed in reception queue 0 (best effort).</p> <p>When the value is B'10, the separation filter is enabled for AVB stream frames; frames from non-matching streams are discarded.</p> <p>When the value is B'11, the separation filter is enabled for AVB stream frames; frames from non-matching streams are processed in reception queue 0 (best effort).</p> <p>For separation filtering, see section 46.3.4(1)(a) Separation Filtering.</p> <p>Writing to this bit is only possible when the current operating mode is configuration mode.</p> <p>B'00: Filtering is disabled. Frames are processed in queue 0 (best effort).</p> <p>B'01: The filter for both AVB stream frames and non-AVB stream frames is enabled; non-matching frames from the stream are processed in queue 0 (best effort).</p> <p>B'10: The filter for separating AVB stream frames from non-AVB stream frames is enabled; non-matching frames are discarded.</p> <p>B'11: The filter for separating AVB stream frames from non-AVB stream frames is enabled; non-matching frames from a stream are processed in queue 0 (best effort).</p> <p>Recommended value: B'10 or B'11</p> |
| 1 | ENCF | B'0 | R/W | <p>Network Control Filtering Enable</p> <p>Setting for reception queue 1 (network control)</p> <p>Enables the AVB network control frame for reception queue 1.</p> <p>When reception queue 1 is disabled, a received frame is stored in reception queue 0 (best effort).</p> <p>Writing to this bit is only possible when the current operating mode is configuration mode.</p> <p>0: Network control is disabled.</p> <p>1: Network control is enabled.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | EFFS | B'0 | R/W | <p>Error Frame Enable</p> <p>Enables or disables the reception of frames that have been classified as error frames by the E-MAC.</p> <p>Received error frames are stored in reception queue 0 (best effort).</p> <p>An indicator of error detection by the E-MAC during reception is stored in the descriptor (DESCR.MS).</p> <p>Writing to this bit is only possible when the current operating mode is configuration mode.</p> <p>0: Error frames are disabled.</p> <p>1: Error frames are enabled.</p> <p>Recommended value: 0</p> |

46.2.9 Receive Queue Configuration Register i (RQCi) (i = 0 to 4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

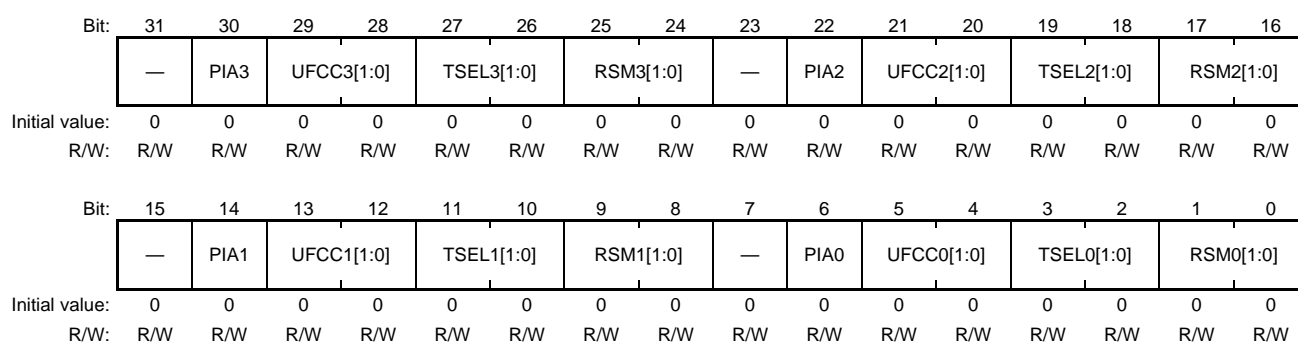
The RQC0 register is used to set up reception queues 0 to 3.

The RQC1 register is used to set up reception queues 4 to 7.

The RQC2 register is used to set up reception queues 8 to 11.

The RQC3 register is used to set up reception queues 12 to 15.

The RQC4 register is used to set up reception queues 16 to 17.



| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|--|
| 31 | — | B'0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 30 | PIA3 | B'0 | R/W | Packed Incremental data Area (Receive Queue 3 + i × 4) This bit defines how addresses inside the incremental data area of queue 3 + i × 4 are handled. When this bit is 0, frame data in incremental data area starts always at a 32 bit address. Depending on received frame length there are up to 3 undefined bytes between two frames in URAM. When this bit is 1, there will no gaps generated between two frames in incremental data area. The CPU can only write to this bit if CSR.OPS is CONFIG. 0: Frame data starts always 32 bit aligned 1: No gaps between frame data in incremental data area |
| 29, 28 | UFCC3[1:0] | B'00 | R/W | Unread Frame Counter Configuration (Receive Queue 3 + i × 4) These bits set the unread frame counter used in reception queue 3 + i × 4. With the AVB-DMAC, four patterns of settings are available for the unread frame counter. Use the unread frame counter warning level configuration register (UFCW) and the unread frame counter stop level configuration register (UFCS) to set the warning level and stop level of the unread frame counter. Set the pattern number 3 set in the unread frame counter warning level configuration register (UFCW) and unread frame counter stop level configuration register (UFCS) in this bit. When the value is B'00, the stop function is disabled. Writing to the bits is only possible when the current operating mode is configuration mode. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|---|
| 27, 26 | TSEL3 | B'00 | R/W | <p>Truncation SElection (Receive Queue 3 + i × 4)</p> <p>This bit defines a maximum number of bytes for storing frames in receive queue 3 + i × 4. Frames longer than the selected maximum are truncated.</p> <p>EthernetAVB-IF provides 2 common configurations for the maximum frame length. This bit selects one of the configurations j defined by RTCi.MFLj.</p> <p>B'00: Maximum frame length defined by RTC0.MFL0 B'01: Maximum frame length defined by RTC0.MFL1 Other configurations are Invalid The CPU can only write to these bits if CSR.OPS is CONFIG.</p> |
| 25, 24 | RSM3[1:0] | B'00 | R/W | <p>Receive Synchronous Mode (Receive Queue 3 + i × 4)</p> <p>These bits set receive synchronous mode.</p> <p>Set B'00 in this bit.</p> <p>For receive synchronous mode, see section 46.3.4(3)(c) Mode with Write-Back. Writing to the bits is only possible when the current operating mode is configuration mode.</p> <p>B'00: Mode with write-back B'01: Keep DT mode (no update of DT field at descriptor write back) B'10: No write-back (no descriptor write back) B'11: Invalid</p> |
| 23 | — | B'0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 22 | PIA2 | B'0 | R/W | <p>Packed Incremental data Area (Receive Queue 2 + i × 4)</p> <p>This bit defines how addresses inside the incremental data area of queue 2 + i × 4 are handled.</p> <p>When this bit is 0, frame data in incremental data area starts always at a 32 bit address. Depending on received frame length there are up to 3 undefined bytes between two frames in URAM.</p> <p>When this bit is 1, there will no gaps generated between two frames in incremental data area.</p> <p>The CPU can only write to this bit if CSR.OPS is CONFIG.</p> <p>0: Frame data starts always 32 bit aligned 1: No gaps between frame data in incremental data area</p> |
| 21, 20 | UFCC2[1:0] | B'00 | R/W | <p>Unread Frame Counter Configuration (Receive Queue 2 + i × 4)</p> <p>These bits set the unread frame counter used in reception queue 2 + i × 4.</p> <p>With the AVB-DMAC, four patterns of settings are available for the unread frame counter. Use the unread frame counter warning level configuration register (UFCW) and the unread frame counter stop level configuration register (UFCS) to set the warning level and stop level of the unread frame counter.</p> <p>Set the pattern number 2 set in the unread frame counter warning level configuration register (UFCW) and unread frame counter stop level configuration register (UFCS) in this bit.</p> <p>When the value is B'00, the stop function is disabled.</p> <p>Writing to the bits is only possible when the current operating mode is configuration mode.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|---|
| 19, 18 | TSEL2 | B'00 | R/W | <p>Truncation SElection (Receive Queue $2 + i \times 4$)</p> <p>This bit defines a maximum number of bytes for storing frames in receive queue $2 + i \times 4$. Frames longer than the selected maximum are truncated.</p> <p>EthernetAVB provides 2 common configurations for the maximum frame length. This bit selects one of the configurations j defined by $RTCi.MFLj$.</p> <p>B'00: Maximum frame length defined by $RTC0.MFL0$</p> <p>B'01: Maximum frame length defined by $RTC0.MFL1$</p> <p>Other configurations are Invalid</p> <p>The CPU can only write to these bits if $CSR.OPIS$ is CONFIG.</p> |
| 17, 16 | RSM2[1:0] | B'00 | R/W | <p>Receive Synchronous Mode (Receive Queue $2 + i \times 4$)</p> <p>These bits set receive synchronous mode.</p> <p>Set B'00 in this bit.</p> <p>For receive synchronous mode, see section 46.3.4(3)(c) Mode with Write-Back. Writing to the bits is only possible when the current operating mode is configuration mode.</p> <p>B'00: Mode with write-back</p> <p>B'01: Keep DT mode (no update of DT field at descriptor write back)</p> <p>B'10: No write-back (no descriptor write back)</p> <p>B'11: Invalid</p> |
| 15 | — | B'0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 14 | PIA1 | B'0 | R/W | <p>Packed Incremental data Area (Receive Queue $1 + i \times 4$)</p> <p>This bit defines how addresses inside the incremental data area of queue $1 + i \times 4$ are handled.</p> <p>When this bit is 0, frame data in incremental data area starts always at a 32 bit address. Depending on received frame length there are up to 3 undefined bytes between two frames in URAM.</p> <p>When this bit is 1, there will no gaps generated between two frames in incremental data area.</p> <p>The CPU can only write to this bit if $CSR.OPIS$ is CONFIG.</p> <p>0: Frame data starts always 32 bit aligned</p> <p>1: No gaps between frame data in incremental data area</p> |
| 13, 12 | UFCC1[1:0] | B'00 | R/W | <p>Unread Frame Counter Configuration (Receive Queue $1 + i \times 4$)</p> <p>These bits set the unread frame counter used in reception queue $1 + i \times 4$.</p> <p>With the AVB-DMAC, four patterns of settings are available for the unread frame counter. Use the unread frame counter warning level configuration register (UFCW) and the unread frame counter stop level configuration register (UFCS) to set the warning level and stop level of the unread frame counter.</p> <p>Set the pattern number 1 set in the unread frame counter warning level configuration register (UFCW) and unread frame counter stop level configuration register (UFCS) in this bit.</p> <p>When the value is B'00, the stop function is disabled.</p> <p>Writing to the bits is only possible when the current operating mode is configuration mode.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|---|
| 11, 10 | TSEL1 | B'00 | R/W | <p>Truncation SElection (Receive Queue $1 + i \times 4$)</p> <p>This bit defines a maximum number of bytes for storing frames in receive queue $1 + i \times 4$. Frames longer than the selected maximum are truncated.</p> <p>EthernetAVB-IF provides 2 common configurations for the maximum frame length. This bit selects one of the configurations j defined by RTCi.MFLj.</p> <p>B'00: Maximum frame length defined by RTC0.MFL0 B'01: Maximum frame length defined by RTC0.MFL1 Other configurations are Invalid</p> <p>The CPU can only write to these bits if CSR.OPS is CONFIG.</p> |
| 9, 8 | RSM1[1:0] | B'00 | R/W | <p>Receive Synchronous Mode (Receive Queue $1 + i \times 4$)</p> <p>These bits set receive synchronous mode.</p> <p>Set B'00 in this bit.</p> <p>For receive synchronous mode, see section 46.3.4(3)(c) Mode with Write-Back. Writing to the bits is only possible when the current operating mode is configuration mode.</p> <p>B'00: Mode with write-back B'01: Keep DT mode (no update of DT field at descriptor write back) B'10: No write-back (no descriptor write back) B'11: Invalid</p> |
| 7 | — | B'0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 6 | PIA0 | B'0 | R/W | <p>Packed Incremental data Area (Receive Queue $0 + i \times 4$)</p> <p>This bit defines how addresses inside the incremental data area of queue $0 + i \times 4$ are handled.</p> <p>When this bit is 0, frame data in incremental data area starts always at a 32 bit address. Depending on received frame length there are up to 3 undefined bytes between two frames in URAM.</p> <p>When this bit is 1, there will no gaps generated between two frames in incremental data area.</p> <p>The CPU can only write to this bit if CSR.OPS is CONFIG.</p> <p>0: Frame data starts always 32 bit aligned 1: No gaps between frame data in incremental data area</p> |
| 5, 4 | UFCC0[1:0] | B'00 | R/W | <p>Unread Frame Counter Configuration (Receive Queue $0 + i \times 4$)</p> <p>These bits set the unread frame counter used in reception queue $0 + i \times 4$.</p> <p>With the AVB-DMAC, four patterns of settings are available for the unread frame counter. Use the unread frame counter warning level configuration register (UFCW) and the unread frame counter stop level configuration register (UFCS) to set the warning level and stop level of the unread frame counter.</p> <p>Set the pattern number 0 set in the unread frame counter warning level configuration register (UFCW) and unread frame counter stop level configuration register (UFCS) in this bit.</p> <p>When the value is B'00, the stop function is disabled.</p> <p>Writing to the bits is only possible when the current operating mode is configuration mode.</p> |

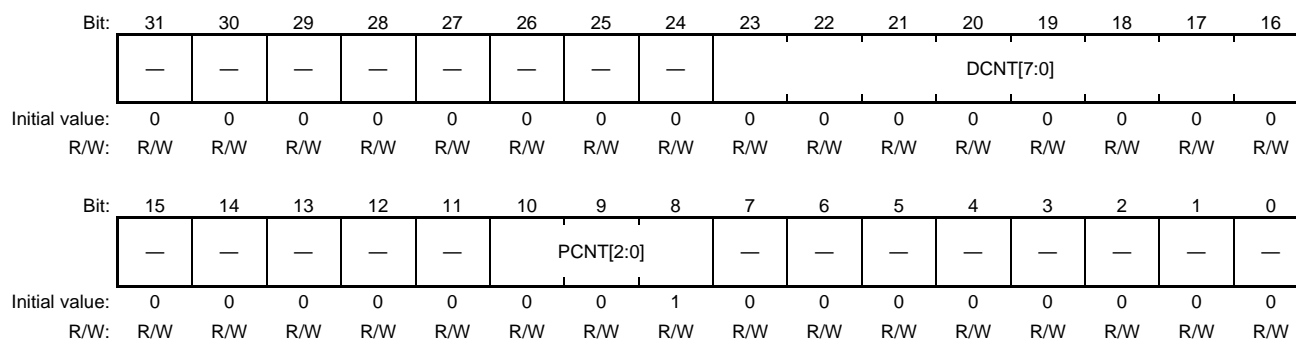
| Bit | Bit Name | Initial Value | R/W | Description |
|------|-----------|---------------|-----|---|
| 3, 2 | TSELO | B'00 | R/W | <p>Truncation SElection (Receive Queue 0 + i × 4)</p> <p>This bit defines a maximum number of bytes for storing frames in receive queue 0 + i × 4. Frames longer than the selected maximum are truncated.</p> <p>EthernetAVB-IF provides 2 common configurations for the maximum frame length. This bit selects one of the configurations j defined by RTCi.MFLj.</p> <p>B'00: Maximum frame length defined by RTC0.MFL0 B'01: Maximum frame length defined by RTC0.MFL1 Other configurations are Invalid The CPU can only write to these bits if CSR.OPS is CONFIG.</p> |
| 1, 0 | RSM0[1:0] | B'00 | R/W | <p>Receive Synchronous Mode (Receive Queue 0 + i × 4)</p> <p>These bits set receive synchronous mode.</p> <p>Set B'00 in this bit.</p> <p>For receive synchronous mode, see section 46.3.4(3)(c) Mode with Write-Back. Writing to the bits is only possible when the current operating mode is configuration mode.</p> <p>B'00: Mode with write-back B'01: Keep DT mode (no update of DT field at descriptor write back) B'10: No write-back (no descriptor write back) B'11: Invalid</p> |

46.2.10 Receive Padding Configuration Register (RPC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RPC register is used to set padding for received frames.

Note: Padding can be used to extend frame lengths, but frame lengths should not exceed 4 Kbytes.

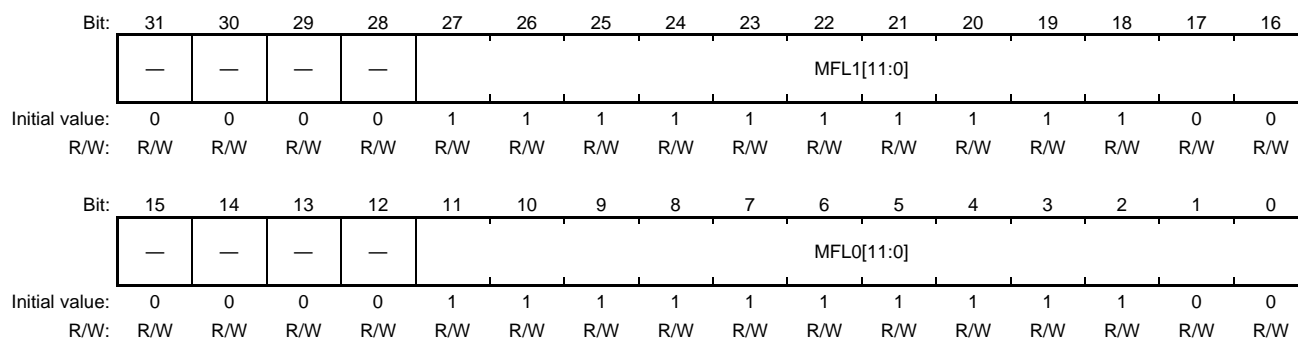


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 24 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 23 to 16 | DCNT[7:0] | H'00 | R/W | Stored Data Counter These bits specify the amount of the frame data (1 to 255) to be stored following the padding. Counting by one indicates one word (4 bytes). For example, when these bits are set to 47, the amount of data is 47 words (= 188 bytes). When these bits are 0, all received data have been stored following the initial padding. Writing to the bits is only possible when the current operating mode is configuration mode. For details on padding, see section 46.3.4(3)(b) Incremental Data Areas. |
| 15 to 11 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 10 to 8 | PCNT[2:0] | B'001 | R/W | Stored Padding Counter These bits specify the amount of padding to be appended to the URAM. Counting by one indicates one word (4 bytes). For example, when these bits are set to 1, the amount of padding is one word (= 4 bytes). Writing to the bits is only possible when the current operating mode is configuration mode. For details on padding, see section 46.3.4(3)(b) Incremental Data Areas. The CPU should not write 0 to these bits. |
| 7 to 0 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |

46.2.11 Reception Truncation Configuration register (RTC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RTC register is used to set Maximum Number of bytes stored per received frame.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 28 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 27 to 16 | MFL1[11:0] | H'FFC | R/W | Maximum Frame Length 1 These bits define the maximum frame length (64 to 4092 bytes) stored to URAM. For each receive queue one parameter 1 used for truncation is selected. When a frame received by E-MAC is truncated, CPU gets informed by EIS.QEF and frame size error (ESR.ET is B'1000). Additionally the DESCR.TI in the updated descriptor is set to 1. The CPU can only write to these bits if CSR.OPS is CONFIG. The CPU should only write values which are a multiple of 4 to these bits. The CPU should not write values less than 64 to these bits. The CPU should not write values less then SFO.FBP + 8 to these bits when RCR.ESF is not equal to B'00. |
| 15 to 12 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 11 to 0 | MFL0[11:0] | H'FFC | R/W | Maximum Frame Length 0 These bits define the maximum frame length (64 to 4092 bytes) stored to URAM. For each receive queue one parameter 0 used for truncation is selected. When a frame received by E-MAC is truncated, CPU gets informed by EIS.QEF and frame size error (ESR.ET is B'1000). Additionally the DESCR.TI in the updated descriptor is set to 1. The CPU can only write to these bits if CSR.OPS is CONFIG. The CPU should only write values which are a multiple of 4 to these bits. The CPU should not write values less than 64 to these bits. The CPU should not write values less then SFO.FBP + 8 to these bits when RCR.ESF is not equal to B'00. |

46.2.12 Unread frame counter warning level register (UFCW)

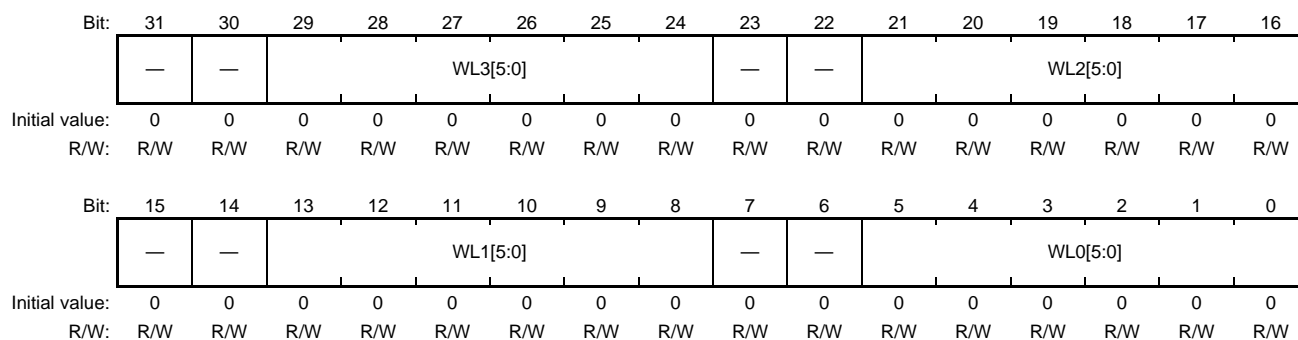
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The UFCW register sets the warning levels for the number of unread frames.

One of the four warning levels from 0 to 3 can be set for each reception queue. When these bits are set to 0, the stop function is disabled. The level to be used is specified by the receive queue configuration register i (RQCi) (i = 0 to 4).

Writing to the bits is only possible when the current operating mode is configuration mode.

The CPU should not write 63 to these bits.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31, 30 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 29 to 24 | WL3[5:0] | B'00_0000 | R/W | Warning Level 3 Unread frame count warning level 3 |
| 23, 22 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 21 to 16 | WL2[5:0] | B'00_0000 | R/W | Warning Level 2 Unread frame count warning level 2 |
| 15, 14 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 13 to 8 | WL1[5:0] | B'00_0000 | R/W | Warning Level 1 Unread frame count warning level 1 |
| 7, 6 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 5 to 0 | WL0[5:0] | B'00_0000 | R/W | Warning Level 0 Unread frame count warning level 0 |

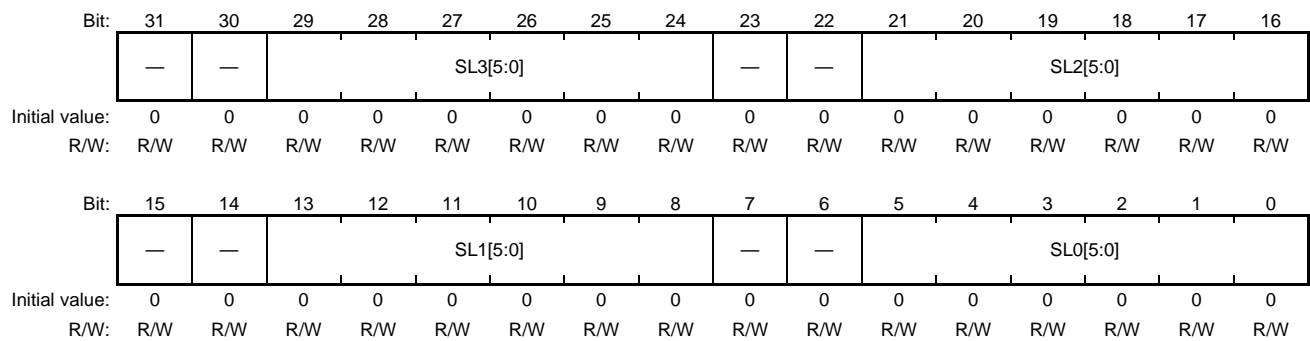
46.2.13 Unread frame counter stop level register (UFCS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The UFCS register sets the stop levels for unread frames.

One of the four stop levels from 0 to 3 can be set for each reception queue. When these bits are set to 0, the stop function is disabled. The level to be used is specified by the receive queue configuration register i (RQCi) ($i = 0$ to 4).

Writing to the bits is only possible when the current operating mode is configuration mode.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31, 30 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 29 to 24 | SL3[5:0] | B'00_0000 | R/W | Stop Level 3 Unread frame count stop level 3 |
| 23, 22 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 21 to 16 | SL2[5:0] | B'00_0000 | R/W | Stop Level 2 Unread frame count stop level 2 |
| 15, 14 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 13 to 8 | SL1[5:0] | B'00_0000 | R/W | Stop Level 1 Unread frame count stop level 1 |
| 7, 6 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 5 to 0 | SL0[5:0] | B'00_0000 | R/W | Stop Level 0 Unread frame count stop level 0 The CPU can only write 0 |

46.2.14 Unread Frame Counter Register i (UFCVi) (i = 0 to 4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The UFCV0 register indicates the number of unread frames in reception queues 0 to 3.

The UFCV1 register indicates the number of unread frames in reception queues 4 to 7.

The UFCV2 register indicates the number of unread frames in reception queues 8 to 11.

The UFCV3 register indicates the number of unread frames in reception queues 12 to 15.

The UFCV4 register indicates the number of unread frames in reception queues 16 and 17.

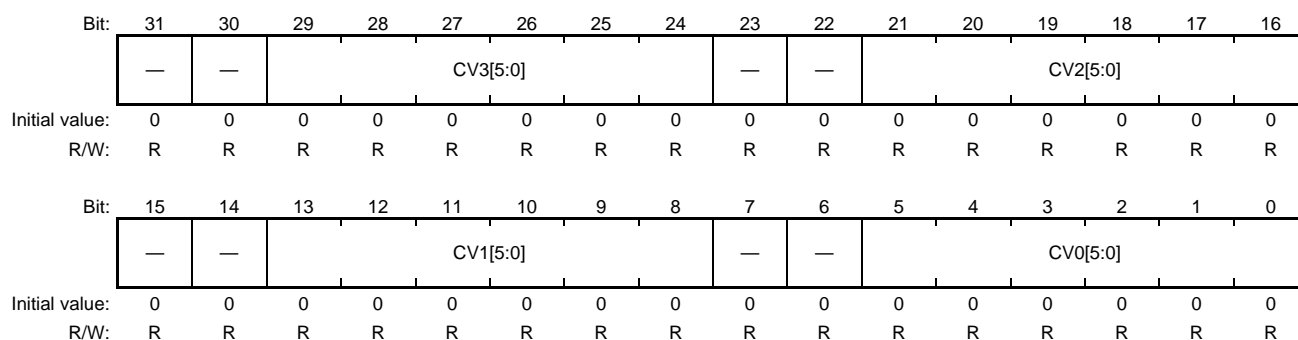
For a description of how to use unread frames, refer to section 46.3.4 (4), Unread Frame Counters.

Conditions for updating:

The bits are set to 0 when the operating mode is not operation mode and when the descriptor base address load request register (DLR) issues a base address load request.

The number is incremented when data received in reception queue r are stored normally. The maximum increment is H'3F. If the value exceeds H'3F, incrementation will not proceed.)

The number is decremented by the value written to the unread frame counter decrement register i (UFCDi).



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31, 30 | — | All 0 | R | Reserved These bits are read as 0. |
| 29 to 24 | CV3[5:0] | B'00_0000 | R | Unread Frame Count 3 + 4 × i Number of unread frames in reception queue 3 + 4 × i |
| 23, 22 | — | All 0 | R | Reserved These bits are read as 0. |
| 21 to 16 | CV2[5:0] | B'00_0000 | R | Unread Frame Count 2 + 4 × i Number of unread frames in reception queue 2 + 4 × i |
| 15, 14 | — | All 0 | R | Reserved These bits are read as 0. |
| 13 to 8 | CV1[5:0] | B'00_0000 | R | Unread Frame Count 1 + 4 × i Number of unread frames in reception queue 1 + 4 × i |
| 7, 6 | — | All 0 | R | Reserved These bits are read as 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 5 to 0 | CV0[5:0] | B'00_0000 | R | Unread Frame Count 0 + 4 × i Number of unread frames in reception queue 0 + 4 × i |

46.2.15 Unread Frame Counter Decrement Register i (UFCDi) (i = 0 to 4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The UFCD0 register is used to decrement unread counters in reception queues 0 to 3.

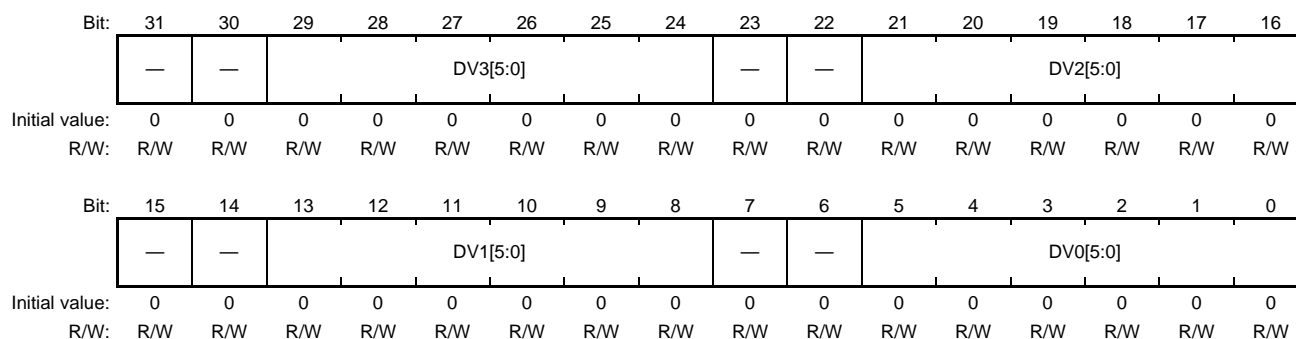
The UFCD1 register is used to decrement unread counters in reception queues 4 to 7.

The UFCD2 register is used to decrement unread counters in reception queues 8 to 11.

The UFC register is used to decrement unread counters in reception queues 12 to 15.

The UFCD4 register is used to decrement unread counters in reception queues 16 and 17.

Write H'3F to these bits to reset the unread counters in reception queue r (r = 0 to 17). These bits are always read as 0.



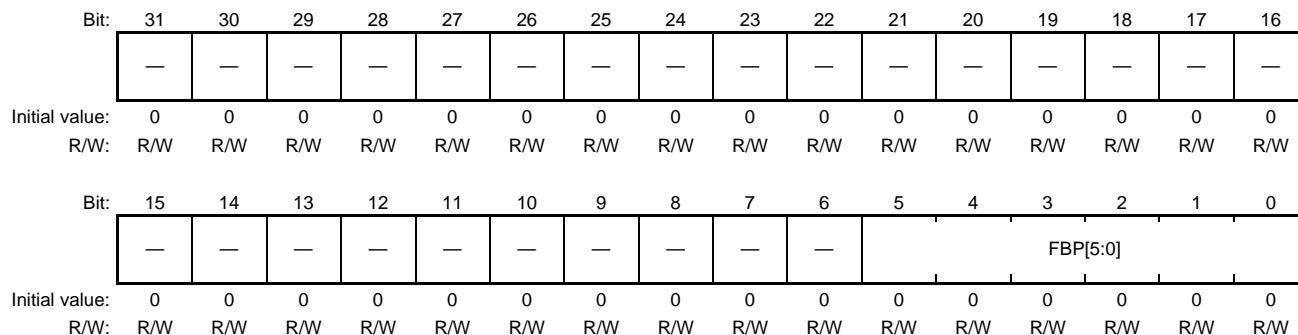
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31, 30 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 29 to 24 | DV3[5:0] | B'00_0000 | R/W | Unread Frame Decrement Value $3 + 4 \times i$ Unread frame decrement value for reception queue $3 + 4 \times i$ |
| 23, 22 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 21 to 16 | DV2[5:0] | B'00_0000 | R/W | Unread Frame Decrement Value $2 + 4 \times i$ Unread frame decrement value for reception queue $2 + 4 \times i$ |
| 15, 14 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 13 to 8 | DV1[5:0] | B'00_0000 | R/W | Unread Frame Decrement Value $1 + 4 \times i$ Unread frame decrement value for reception queue $1 + 4 \times i$ |
| 7, 6 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 5 to 0 | DV0[5:0] | B'00_0000 | R/W | Unread Frame Decrement Value $0 + 4 \times i$ Unread frame decrement value for reception queue $0 + 4 \times i$ |

46.2.16 Separation Filter Offset Register (SFO)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The SFO register sets an offset into frames for use by the separation filter.

Note: Received frames having fewer bytes than the setting of these bits + 8 bytes are judged to be non-matching by the separation filter. In this case, the data will either be sorted into a reception queue or discarded in accord with the setting of the separation filtering select bits in the receive configuration register (RCR.ESF).



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 6 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 5 to 0 | FBP[5:0] | B'00_0000 | R/W | First Byte Position These bits set the position in Ethernet frames of the first byte of the bytes to be used by the separation filter. When these bits are 0, the separation filter starts from the start of each Ethernet frame (first byte of the destination address). For bytes in Ethernet frames, see Figure 46.2 Operating Mode of AVB-DMAC, in section 46.3.1(1) Operating Modes. Writing to the bits is only possible when the current operating mode is configuration mode. [Changing condition] These bits are updated to the value of SFVi.LV[5:0] when SFL.LC changes from 30 (Load SFO.FBP) to 31 while CSR.OPS is OPERATION. For separation filtering, see section 46.3.4(1)(a) Separation Filtering. |

46.2.17 Separation Filter Pattern Register i (SFPi) (i = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

A pair of SFPi registers set the pattern for the separation filters to be used by the corresponding reception queues 2 to 17 (for streams 0 to 15).

Each queue shares a 64-bit setting; reception queue 2 (for stream 0) uses SFP0 and SFP1, reception queue 17 (for stream 15) uses SFP30 and SFP31, and so on.

The separation filter passes a frame when, after masking by the mask value set in the separation filter mask register (SFMi), data from received frames match the value defined in these bits.

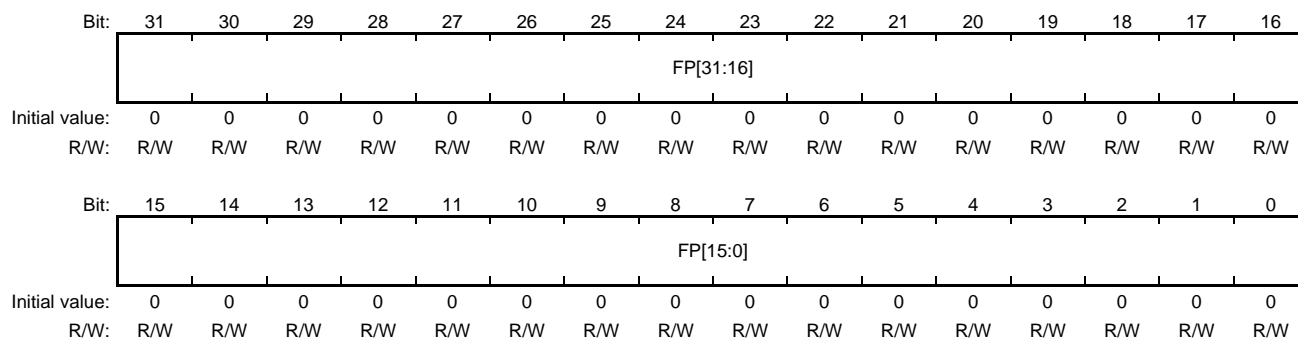
SFPi.FP[7:0] (where i is an even number) are used for the byte of Ethernet frame data specified by the separation filter offset register, while SFPi.FP[63:56] (where i is the corresponding odd number) are used for the byte at the address specified by the separation filter offset register (SFO) + 7.

Writing to the bits is only possible when the current operating mode is configuration mode.

[Changing condition]

These bits are updated to the value of SFVi.LV[63:0] when SFL.LC changes from s (Load SFPi.FPs) to 31 while CSR.OPS is OPERATION.

For separation filtering, see section 46.3.4(1)(a) Separation Filtering.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | FP[31:0] | H'0000_0000 | R/W | Separation Filter Pattern These bits set the pattern of the separation filter. The 64-bit filter pattern is set for each queue. |

46.2.18 Separation Filter Value register i (SFVi) (i = 0 to 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

These bits define the 64 bit value to be loaded for separation filtering.

When loading the first by position (SFO.FBP), SFVi.LV[63:6] are ignored.

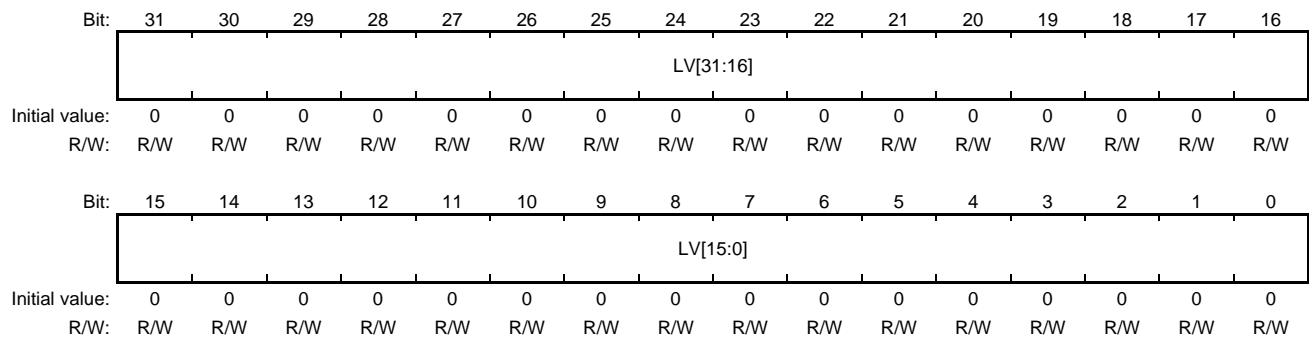
The CPU can only write to these bits if CSR.OPS is OPERATION.

The CPU can only write to these bits if SFL.LC is 31.

[Changing condition]

These bits are set to 0 when leaving OPERATION mode.

For separation filtering, see section 46.3.4(1)(a) Separation Filtering.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | LV[31:0] | H'0000_0000 | R/W | Separation Filter Value These bits set the Load value of the separation filter. |

46.2.19 Separation Filter Mask Register i (SFMi) (i = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

A pair of SFMi registers sets the mask value for the separation filter used by the corresponding reception queue 2 to 17 (stream 0 to 15).

SFM0.CFM[7:0] are used for bytes of Ethernet frame data specified by the separation filter offset register, while SFM1.CFM[63:56] are used for the separation filter offset register (SFO) + 7.

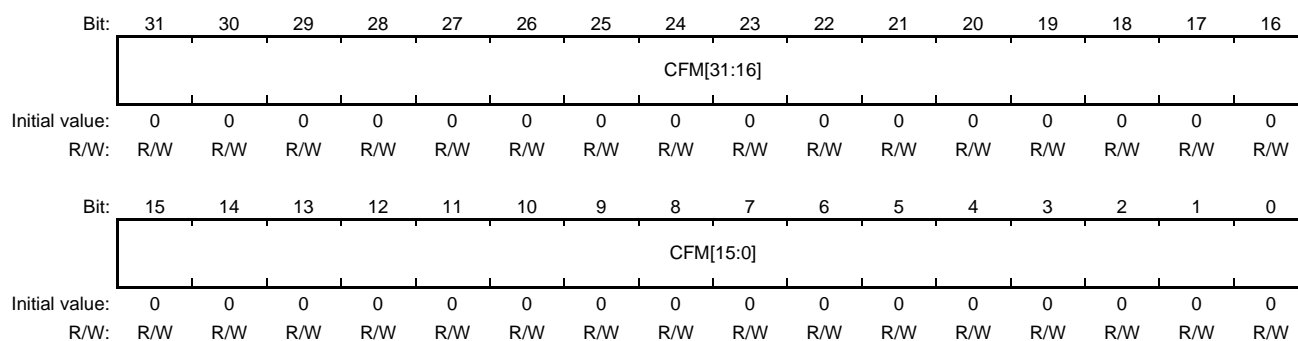
Frame data at the positions of mask bits that are set to 0 are masked; that is, they do not affect pattern-matching by the separation filter.

Writing to the bits is only possible when the current operating mode is configuration mode.

[Changing condition]

These bits are updated to the value of SFVi.LV[63:0] when SFL.LC changes from 29 (Load SFMi.CFM) to 31 while CSR.OPS is OPERATION.

For separation filtering, see section 46.3.4(1)(a) Separation Filtering.

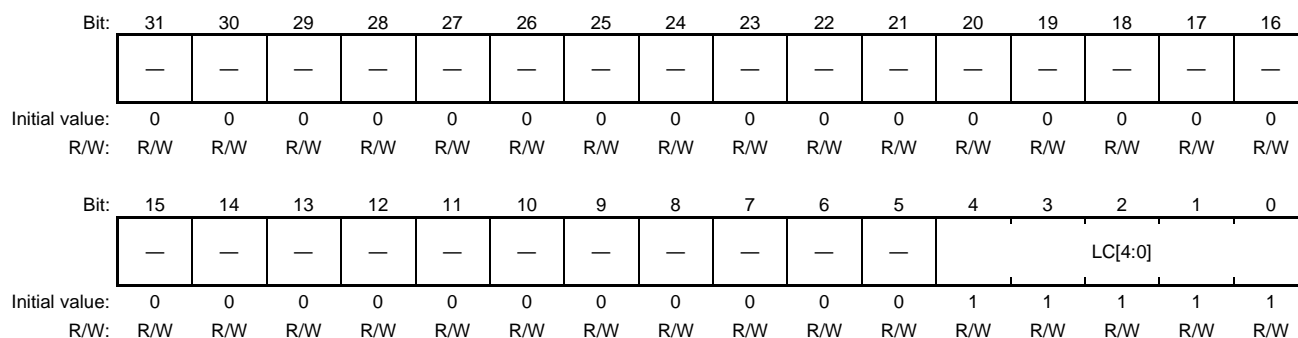


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 0 | CFM[31:0] | H'0000_0000 | R/W | Separation Filter Mask These bits set the mask value for the separation filter. |

46.2.20 Separation Filter Load register (SFL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The SFL register is used to separation filter load.

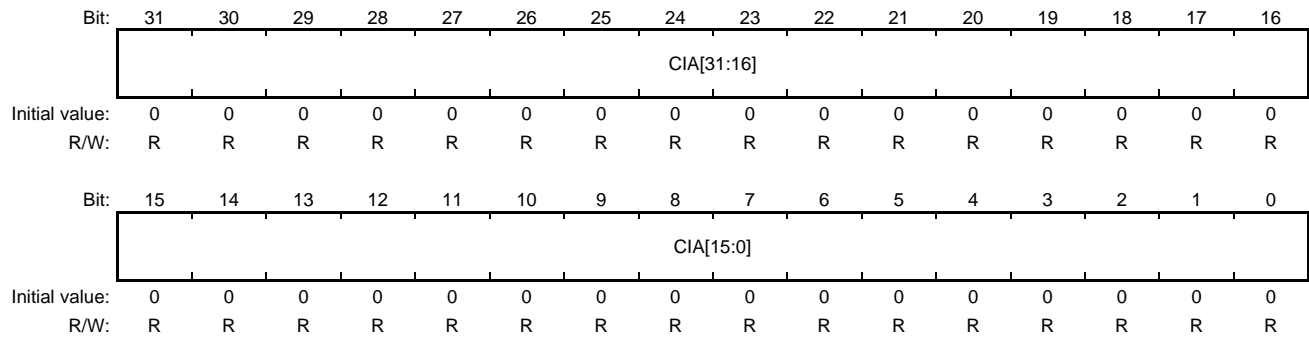


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 5 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 4 to 0 | LC[4:0] | H'1F | R/W | Load Command These bits control update of separation filter configuration in OPERATION mode. By writing 30 to these bits CPU triggers an update of First Byte Position (SFO.FBP) value. By writing 29 to these bits CPU triggers an update of Common Filter Mask (SFMi.CFM) value. By writing number between 0 and 15 to these bits CPU triggers an update of Filter Pattern s (SFPI.FPs) value. The number written to SFL.LC defines the index s of the updated Filter Pattern. The CPU can only write to these bits if CSR.OPS is OPERATION. The CPU can only write to these bits if SFL.LC is 31. [Changing condition] These bits are set to 31 when leaving OPERATION mode. These bits are set to 31 when the requested update has been processed. 0-15: Load of SFPI.FPs with s = SFL.LC 16-28: Invalid 29: Load of SFMi.CFM 30: Load of SFO.FBP 31: No pending load request |

46.2.21 Current Incremental Address Register r (CIARr) (r = 0 to 17)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CIARr register is used to check queue address.

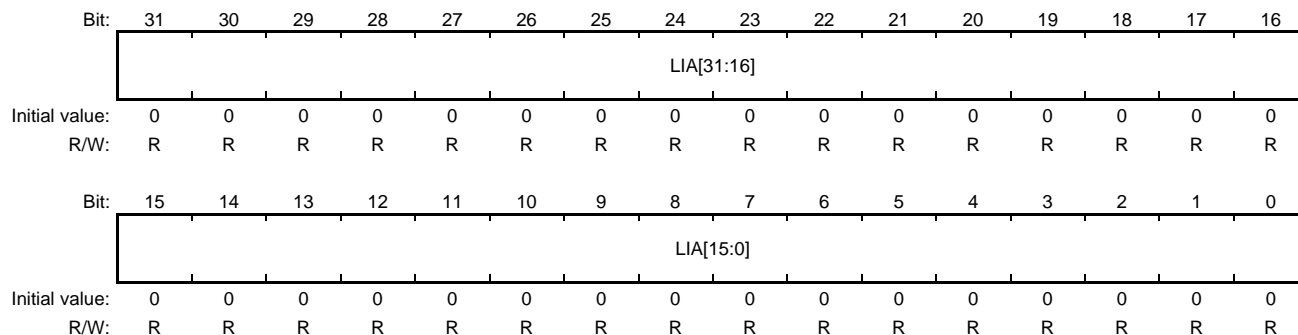


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 0 | CIA[31:0] | H'0000_0000 | R | <p>Current incremental address used by receive queue</p> <p>These bits indicate the incremental address of receive queue r, where currently frame data is stored or where frame data will be stored when queue r gets active.</p> <p>Refer to section 46.3.4(3)(b) Incremental Data Areas.</p> <p>[Changing condition]</p> <p>These bits are set to 0 when leaving OPERATION mode.</p> <p>These bits are updated to next address in incremental data area when processing a FEMPTY_IS or FEMPTY_IC descriptor.</p> |

46.2.22 Last Incremental Address Register r (LIARr) (r = 0 to 17)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The LIARr register is used to check queue address.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | LIA[31:0] | H'0000_0000 | R | <p>Last incremental address used by receive queue</p> <p>These bits indicate the current incremental address of receive queue r (CIARr.CIA) before its address was updated due to processing a FEMPTY_IS descriptor. So it describes the first not used address inside the incremental data area of receive queue r. Refer to section 46.3.4(3)(b) Incremental Data Areas.</p> <p>[Changing condition]</p> <p>These bits are set to 0 when leaving OPERATION mode.</p> <p>These bits are set to CIARr.CIA when processing a FEMPTY_IS descriptor.</p> |

46.2.23 Transmit Configuration Register (TGC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The TGC register is used to make settings related to transmission for the AVB-DMAC.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----------|-----|-----|-----|-----------|-----|-----|-----|-----------|------|------|------|-----------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | TBD3[1:0] | | — | — | TBD2[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | TBD1[1:0] | | — | — | TBD0[1:0] | | — | — | TQP | ECBS | TSM3 | TSM2 | TSM1 | TSM0 |
| Initial value: | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 22 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 21, 20 | TBD3[1:0] | B'10 | R/W | Transmit FIFO Size (Stream Class A) These bits set the sizes of the transmission FIFO buffers for use with each of the transmission queue 3 (for stream class A). Writing to these bits is only possible when the current operating mode is configuration mode. |
| 19, 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17, 16 | TBD2[1:0] | B'10 | R/W | Transmit FIFO Size (Stream Class B) These bits set the sizes of the transmission FIFO buffers for use with each of the transmission queue 2 (for stream class B). Writing to these bits is only possible when the current operating mode is configuration mode. |
| 15, 14 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 13, 12 | TBD1[1:0] | B'10 | R/W | Transmit FIFO Size (Network Control) These bits set the sizes of the transmission FIFO buffers for use with each of the transmission queue 1 (for network control). Writing to these bits is only possible when the current operating mode is configuration mode. |
| 11, 10 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 9, 8 | TBD0[1:0] | B'10 | R/W | Transmit FIFO Size (Best Effort) These bits set the sizes of the transmission FIFO buffers for use with each of the transmission queue 0 (for best effort). Writing to these bits is only possible when the current operating mode is configuration mode. |
| 7, 6 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 5 | TQP | B'0 | R/W | <p>Transmission Queues Priority</p> <p>This bit defines how EthernetAVB-IF prioritises between transmit queues when scheduling transmission. For fetching always the queue with highest priority is selected, for transmission the queue with highest priority where data is available in Tx-Buffer.</p> <p>Transmission priority of queue 2 and 3 can be influenced by the credit based shaping algorithm.</p> <p>The priority of the transmit queues is given from high to low.</p> <p>0: Default priority: Q3(CBS), Q2(CBS), Q1, Q0 1: Alternate priority: Q1, Q3(CBS), Q2(CBS), Q0</p> <p>For the credit-based shaping (CBS) algorithm, see section 46.3.6 CBS (Credit-Based Shaping).</p> <p>Writing to the bits is only possible when the current operating mode is configuration mode.</p> |
| 4 | ECBS | B'0 | R/W | <p>Enable Credit Based Shaping</p> <p>This bit defines if CBS algorithm is used for transmit queue 2 and 3 to prioritise between transmit queues when scheduling transmission. When CBS is enabled, the shaping bases on queue specific configuration in CIVRc, CDVRc, CULc and CLLc.</p> <p>0: CBS globally disabled 1: CBS enabled based on queue specific configuration</p> <p>For the credit-based shaping (CBS) algorithm, see section 46.3.6 CBS (Credit-Based Shaping).</p> <p>Writing to the bits is only possible when the current operating mode is configuration mode.</p> |
| 3 | TSM3 | B'0 | R/W | <p>Transmit Synchronous Mode (Stream Class A)</p> <p>Set these bits to 0.</p> <p>0: With write-back 1: Setting prohibited</p> |
| 2 | TSM2 | B'0 | R/W | <p>Transmit Synchronous Mode (Stream Class B)</p> <p>Set these bits to 0.</p> <p>0: With write-back 1: Setting prohibited</p> |
| 1 | TSM1 | B'0 | R/W | <p>Transmit Synchronous Mode (Network Control)</p> <p>Set these bits to 0.</p> <p>0: With write-back 1: Setting prohibited</p> |
| 0 | TSM0 | B'0 | R/W | <p>Transmit Synchronous Mode (Best Effort)</p> <p>Set these bits to 0.</p> <p>0: With write-back 1: Setting prohibited</p> |

46.2.24 Transmit Configuration Control Register (TCCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The TCCR register controls transmission by the AVB-DMAC and is used to make related settings.

Conditions for updating:

The bit is set to 0 when the operating mode is not operation mode, when a descriptor of type EEMPTY, FEMPTY or LEMPTY (no usable data) is processed, when an EOS descriptor is processed, and when a descriptor with defective data is processed.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|------|-----|-----|-----|-----|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | MFR | MFEN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | TFR | TFEN | — | — | — | — | TSRQ3 | TSRQ2 | TSRQ1 | TSRQ0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 | MFR | B'0 | R/W | E-MAC status FIFO Release This bit releases the oldest entry of the E-MAC status FIFO. When 1 is written to this bit, EthernetAVB-IF gets informed that CPU has processed the oldest E-MAC status FIFO entry as visible in MFA register. The oldest entry is removed from E-MAC status FIFO. This bit is always read as 0. 0: No request to E-MAC status FIFO 1: Release oldest entry of E-MAC status FIFO |
| 16 | MFEN | B'0 | R/W | E-MAC status FIFO ENable This bit enables the storage of transmission status information (the MAC flags for frame transmission) in the E-MAC status FIFO. Additionally CPU can control the status storage for each frame provided for transmission using DESCR.MSR. When this bit is set to 0 all pending FIFO entries are removed. 0: Disabled 1: Enabled |
| 15 to 10 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 9 | TFR | B'0 | R/W | Time Stamp FIFO Release This bit releases the oldest entry in the time-stamp FIFO. For a description of how to use the time-stamp FIFO, see section 46.3.5(4) Time Stamping in Transmission. 0: (Not operating) 1: Releases the oldest entry in the time-stamp FIFO. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 8 | TFEN | B'0 | R/W | <p>Time Stamp FIFO Enable</p> <p>This bit enables storage in the time-stamp FIFO.</p> <p>When it is set, time-stamp information is stored for descriptors with DESCR.TSR set to 1 (for DESCR.TSR, see section 46.3.5(2)(b) Configuration of Transmission Frame Data Descriptors.</p> <p>When 0 is set in this bit, no entries are made in the time-stamp FIFO.</p> <p>For a description of how to use the time-stamp FIFO, see section 46.3.5(4) Time Stamping in Transmission.</p> <p>0: Recording of transmission time stamps in the time-stamp FIFO is disabled.</p> <p>1: Recording of transmission time stamps in the time-stamp FIFO is enabled.</p> |
| 7 to 4 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 3 | TSRQ3 | B'0 | R/W | <p>Transmit Start Request (Queue 3 (Stream Class A))</p> <p>This bit issues a request to start transmission for transmission queue 3.</p> <p>When read, this bit being set to 1 indicates that transmission queue t has a frame that has not yet been fetched to the transmission FIFO.</p> <p>Frame transmission by the E-MAC is processed independently from fetching to the transmission FIFO. The timing of transmission from a queue depends on the priority order of transmission.</p> <p>For the scheduling of transmission queues, see section 46.3.5(1) Transmission Modes.</p> <p>Writing to this bit is only possible when the current operating mode is operation mode.</p> <p>Only 1 can be written to the bit. Writing 0 to the bit has no effect.</p> <p>0: Transmission queue is empty or stopped.</p> <p>1: When written: A transmission start request is issued.</p> <p>When read: Fetching of data for transmission is pending.</p> |
| 2 | TSRQ2 | B'0 | R/W | <p>Transmit Start Request (Queue 2 (Stream Class B))</p> <p>This bit issues a request to start transmission for transmission queue 2.</p> <p>When read, this bit being set to 1 indicates that transmission queue t has a frame that has not yet been fetched to the transmission FIFO.</p> <p>Frame transmission by the E-MAC is processed independently from fetching to the transmission FIFO. The timing of transmission from a queue depends on the priority order of transmission.</p> <p>For the scheduling of transmission queues, see section 46.3.5(1) Transmission Modes.</p> <p>Writing to this bit is only possible when the current operating mode is operation mode.</p> <p>Only 1 can be written to the bit. Writing 0 to the bit has no effect.</p> <p>0: Transmission queue is empty or stopped.</p> <p>1: When written: A transmission start request is issued.</p> <p>When read: Fetching of data for transmission is pending.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | TSRQ1 | B'0 | R/W | <p>Transmit Start Request (Queue 1 (Network Control))</p> <p>This bit issues a request to start transmission for transmission queue 1.</p> <p>When read, this bit being set to 1 indicates that transmission queue t has a frame that has not yet been fetched to the transmission FIFO.</p> <p>Frame transmission by the E-MAC is processed independently from fetching to the transmission FIFO. The timing of transmission from a queue depends on the priority order of transmission.</p> <p>For the scheduling of transmission queues, see section 46.3.5(1) Transmission Modes.</p> <p>Writing to this bit is only possible when the current operating mode is operation mode.</p> <p>Only 1 can be written to the bit. Writing 0 to the bit has no effect.</p> <p>0: Transmission queue is empty or stopped.</p> <p>1: When written: A transmission start request is issued.</p> <p>When read: Fetching of data for transmission is pending.</p> |
| 0 | TSRQ0 | B'0 | R/W | <p>Transmit Start Request (Queue 0 (Best Effort))</p> <p>This bit issues a request to start transmission for transmission queue 0.</p> <p>When read, this bit being set to 1 indicates that transmission queue t has a frame that has not yet been fetched to the transmission FIFO.</p> <p>Frame transmission by the E-MAC is processed independently from fetching to the transmission FIFO. The timing of transmission from a queue depends on the priority order of transmission.</p> <p>For the scheduling of transmission queues, see section 46.3.5(1) Transmission Modes.</p> <p>Writing to this bit is only possible when the current operating mode is operation mode.</p> <p>Only 1 can be written to the bit. Writing 0 to the bit has no effect.</p> <p>0: Transmission queue is empty or stopped.</p> <p>1: When written: A transmission start request is issued.</p> <p>When read: Fetching of data for transmission is pending.</p> |

46.2.25 Transmit Status Register (TSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The TSR register indicates the state of transmission by the AVB-DMAC.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|-----------|----|----|----|----|----|-----------|----|-----------|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | MFFL[4:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | TFFL[2:0] | | — | — | — | — | CCS1[1:0] | | CCS0[1:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 21 | — | All 0 | R | Reserved These bits are read as 0. The write value should be 0. |
| 20 to 16 | MFFL[4:0] | H'00 | R | Number of entries stored in the E-MAC status FIFO E-MAC status FIFO Fill Level These bits indicate the number of entries available in the MAC status FIFO. If this value is 0, the FIFO is empty. If this value is 16, the FIFO is full. The values 17 to 31 are reserved. [Changing condition] These bits are set to 0 when leaving OPERATION mode. These bits are set to 0 when TCCR.MFEN is 0. This value is incremented when frame with DESCR.MSR has been transmitted by E-MAC, TCCR.MFEN is 1 and TSR.MFFL is not 16. This value is incremented when E-MAC detects an error during transmission, TCCR.MFEN is 1 and TSR.MFFL is not 16. This value is decremented when 1 is written to TCCR.MFR and TSR.MFFL is not 0. |
| 15 to 11 | — | All 0 | R | Reserved These bits are read as 0. The write value should be 0. |

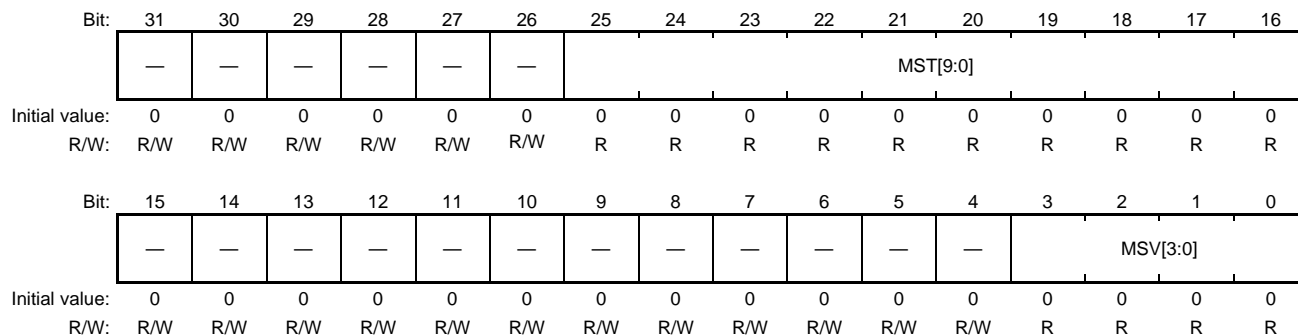
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 10 to 8 | TFFL[2:0] | B'00 | R | <p>Time Stamp FIFO Count</p> <p>These bits indicate the number of time stamps in the time-stamp FIFO.</p> <p>The values 0 and 3 indicate that the time-stamp FIFO is empty and full, respectively (values 4 to 7 are reserved).</p> <p>Conditions for updating:</p> <ul style="list-style-type: none"> The bits are set to 0 when the operating mode is not operation mode and when the time stamp FIFO enable bit in the transmit configuration control register (TCCR.TFEN) = 0. When the time stamp FIFO enable bit (TCCR.TFEN) is 1 and these bits are not 3, the value of these bits is incremented after a frame with DESCR.TSR set has been transmitted by the E-MAC (for DESCR.TSR, see section 46.3.5(2)(b) Configuration of Transmission Frame Data Descriptors. The value of these bits is decremented if it is not 0 when 1 is written to the time stamp FIFO release bit in the transmit configuration control register (TCCR.TFR). |
| 7 to 4 | — | All 0 | R | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 3, 2 | CCS1[1:0] | B'00 | R | <p>CBS Counter Status 1 (Class A)</p> <p>These bits indicate the CBS (credit-based shaping) state of stream data transmission queue 1. If the calculated credit value is outside the range specified by CBS upper limit register c (CULc) and CBS lower limit register c (CLLc), it falls outside the range for CBS.</p> <p>Conditions for updating:</p> <ul style="list-style-type: none"> The bits are set to B'00 when the operating mode is not operation mode. The bits are set to B'00 when the CBS is inside the condition limits. The bits are set to B'01 if the credit value calculated by the CBS is lower than the value in CBS lower limit register c (CLLc). The bits are set to B'10 if the credit value calculated by the CBS is higher than the value in CBS upper limit register c (CULc). <p>B'00: The current credit value is within the limit. B'01: The current credit value is less than or equal to the lower limit. B'10: The current credit value is greater than or equal to the upper limit. B'11: Setting prohibited</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|-----------|---------------|-----|---|
| 1, 0 | CCS0[1:0] | B'00 | R | <p>CBS Counter Status 0 (Class B)</p> <p>These bits indicate the CBS (credit-based shaping) state of stream data transmission queue 0. If the calculated credit value is outside the range specified by CBS upper limit register c (CULc) and CBS lower limit register c (CLLc), it falls outside the range for CBS.</p> <p>Conditions for updating:</p> <ul style="list-style-type: none"> • The bits are set to B'00 when the operating mode is not operation mode. The bits are set to B'00 when the CBS is inside the condition limits. • The bits are set to B'01 if the credit value calculated by the CBS is lower than the value in CBS lower limit register c (CLLc). • The bits are set to B'10 if the credit value calculated by the CBS is higher than the value in CBS upper limit register c (CULc). <p>B'00: The current credit value is within the limit. B'01: The current credit value is less than or equal to the lower limit. B'10: The current credit value is greater than or equal to the upper limit. B'11: Setting prohibited</p> |

46.2.26 E-MAC status FIFO Access register (MFA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The MFA register indicates the state of MAC.



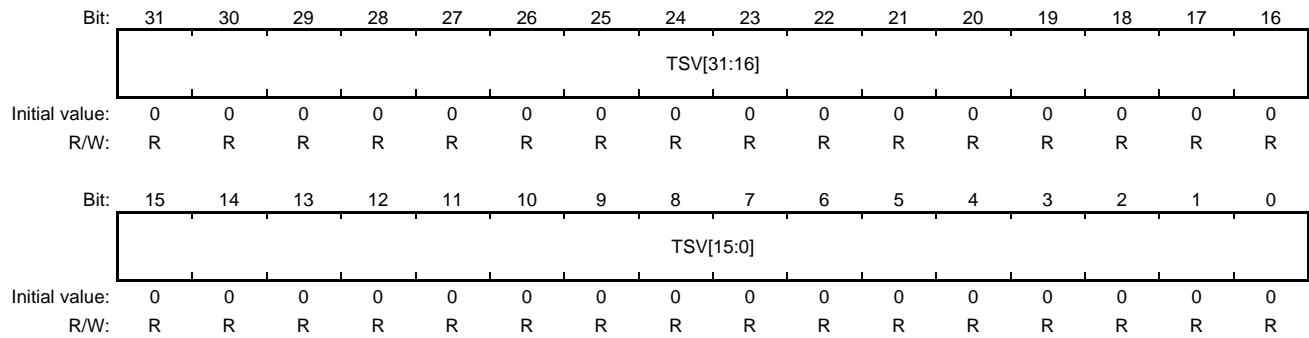
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 26 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 25 to 16 | MST[9:0] | H'000 | R | Tag number from descriptor identifying the frame E-MAC status relation E-MAC Status Tag These bits represent the DESCR.TAG bits from the descriptor defining the data for frame transmission. The tag is used to get the relation between the frame inside a transmit queue and the E-MAC status available in the FIFO (MFA.MSV). The CPU should not read these bits when TSR.MFFL is 0. [Changing condition] These bits are updated when the first entry is stored in FIFO (TSR.MFFL changes from 0 to 1). These bits are updated when the oldest entry is released (writing 1 to TCCR.MFR). These bits are updated when the FIFO is overwritten (a frame with DESCR.MSR in descriptor has been transmitted by MAC, TCCR.MFEN is 1 and TSR.MFFL is 16). These bits are updated when the FIFO is overwritten (E-MAC detects an error during transmission and TSR.MFFL is 16). <i>Note for verification: These bits are set to 0 when leaving OPERATION mode.</i> |
| 15 to 4 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 3 to 0 | MSV[3:0] | H'0 | R | <p>E-MAC Status Value</p> <p>These bits represent the E-MAC status bits as stored in the oldest E-MAC status FIFO entry.</p> <p><i>MSV[3] Reserved</i></p> <p><i>MSV[2] Carrier lost during frame transmission</i></p> <p><i>MSV[1] Delay collision</i></p> <p><i>MSV[0] Transmission time out</i></p> <p>The CPU should not read these bits when TSR.MFFL is 0.</p> <p>[Changing condition]</p> <p>These bits are updated when the first entry is stored in FIFO (TSR.MFFL changes from 0 to 1).</p> <p>These bits are updated when the oldest entry is released (writing 1 to TCCR.MFR).</p> <p>These bits are updated when the FIFO overwritten (a frame with DESCR.MSR in descriptor has been transmitted by E-MAC, TCCR.MFEN is 1 and TSR.MFFL is 16).</p> <p>These bits are updated when the FIFO is overwritten (E-MAC detects an error during transmission and TSR.MFFL is 16).</p> <p><i>Note for verification: These bits are set to 0 when leaving OPERATION mode.</i></p> |

46.2.27 Time Stamp FIFO Access Register 0 (TFA0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TFA0 indicates the nano seconds portion of the timestamp value.

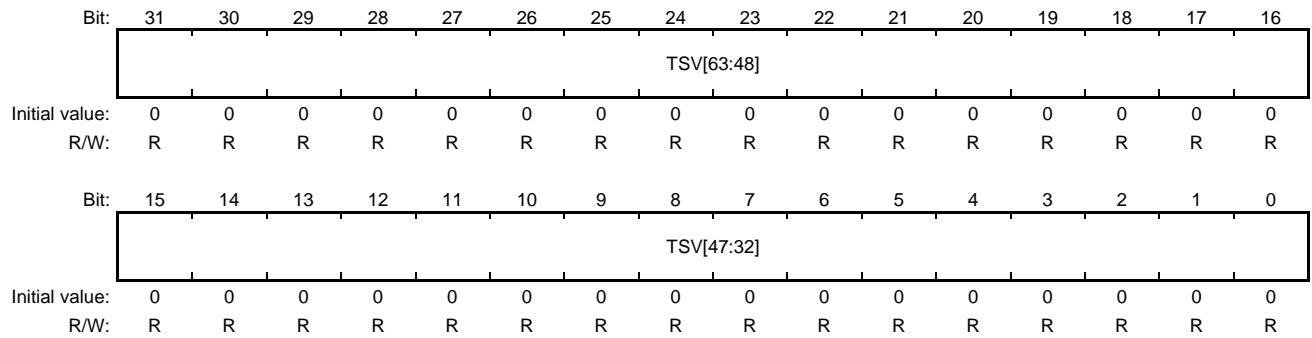


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 0 | TSV[31:0] | H'0000_0000 | R | <p>Time Stamp Value</p> <p>These 80 bits consist of TFA0.TSV[31:0], TFA1.TSV[63:32], and TFA2.TSV[79:64], which together indicate the oldest time stamp value stored in the time-stamp FIFO.</p> <p>Once the time-stamp FIFO is full, no further time-stamp values are stored.</p> <p>Conditions for updating:</p> <ul style="list-style-type: none"> The bits are set to H'0000_0000 when the operating mode is not operation mode. The register is updated whenever a value is stored in the time-stamp FIFO (when the time-stamp FIFO count bit in the transmit status register (TSR.TFFL) changes from 0 to 1). The register is updated when the oldest entry is released (when the time stamp FIFO release bit in the transmit configuration control register (TCCR.TFR) is set to 1). |

46.2.28 Time Stamp FIFO Access Register 1 (TFA1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The TFA1 register indicates the lower seconds portion of the timestamp value.

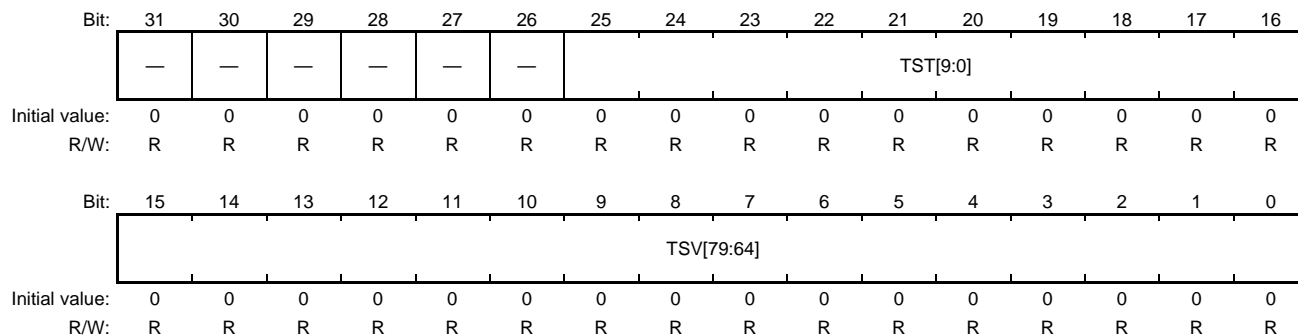


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 0 | TSV[63:32] | H'0000_0000 | R | Time Stamp Value For details, see section 46.2.28 Time Stamp FIFO Access Register 0 (TFA0). |

46.2.29 Time Stamp FIFO Access Register 2 (TFA2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

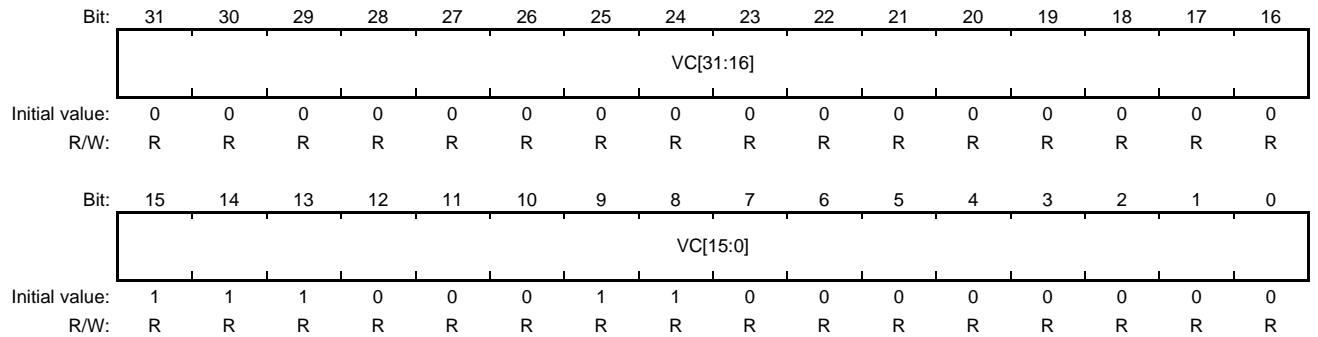
The TFA2 register indicates the timestamp tag and the higher seconds portion of the timestamp value.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 26 | — | All 0 | R | Reserved These bits are read as 0. |
| 25 to 16 | TST[9:0] | H'000 | R | Time Stamp Tag These bits indicate the contents of the DESCR.TAG bit within the descriptor for frame transmission. These values are used to check the correlation between frames within the transmission queue and the time-stamp values (accessible through time stamp FIFO access register i (TFAi)) which can be placed in the FIFO. For the tagging of frames in transmission, see section 46.3.5(4) Time Stamping in Transmission. Conditions for updating: <ul style="list-style-type: none"> The bits are set to H'000 when the operating mode is not operation mode. Updated when a value is stored in the time-stamp FIFO (when the value of the time stamp FIFO count bit in the transmit status register (TSR.TFFL) changes from 0 to 1). Updated when the oldest entry has been released (1 is set in the time stamp FIFO release bit in the transmit configuration control register (TCCR.TFR)). |
| 15 to 0 | TSV[79:64] | H'0000 | R | Time Stamp Value For details, see section 46.2.28 Time Stamp FIFO Access Register 0 (TFA0). |

46.2.30 Version and Release Register (VRR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



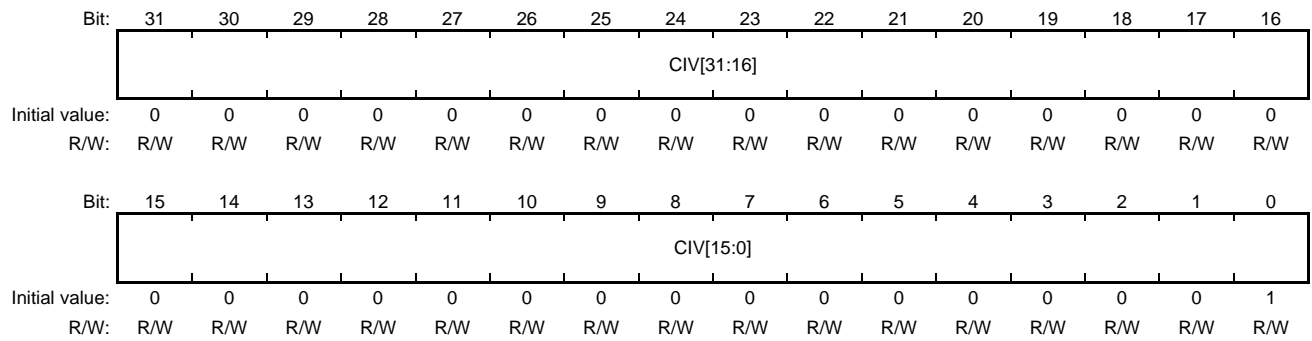
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | VC[31:0] | H'0000_E300 | R | Version Code These bits identify the version of used EthernetAVB-IF. |
| | | | | H'0000_E300: RAVBES3 |

46.2.31 CBS Increment Value Register c (CIVRc) (c = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CIVR0 register sets the increment in the CBS algorithm for transmission queue 2 (for stream class B).

The CIVR1 register sets the increment in the CBS algorithm for transmission queue 3 (for stream class A).



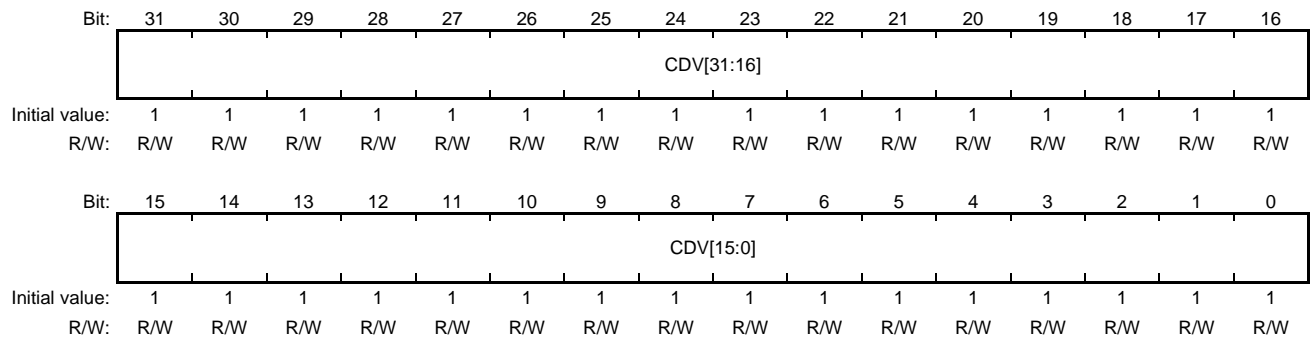
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 0 | CIV[31:0] | H'0000_0001 | R/W | <p>CBS Increment Value</p> <p>CBS increment value (1 to H'FFFF)</p> <p>These bits define the increment value related to idleSlope for the CBS algorithm.</p> <p>Set a value in the range from H'0000_0001 to H'0000_FFFF.</p> <p>The value to be written to these bits depends on the Ethernet bit rate and HPφ (high-speed peripheral clock). For details, see section 46.3.6 CBS (Credit-Based Shaping).</p> <p>Note: When write to these bits, corresponding queue's credits are cleared to 0 respectively.</p> |

46.2.32 CBS Decrement Value Register c (CDVRC) (c = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CDVR0 register sets the decrement in the CBS algorithm for transmission queue 2 (for stream class B).

The CDVR1 register sets the decrement in the CBS algorithm for transmission queue 3 (for stream class A).



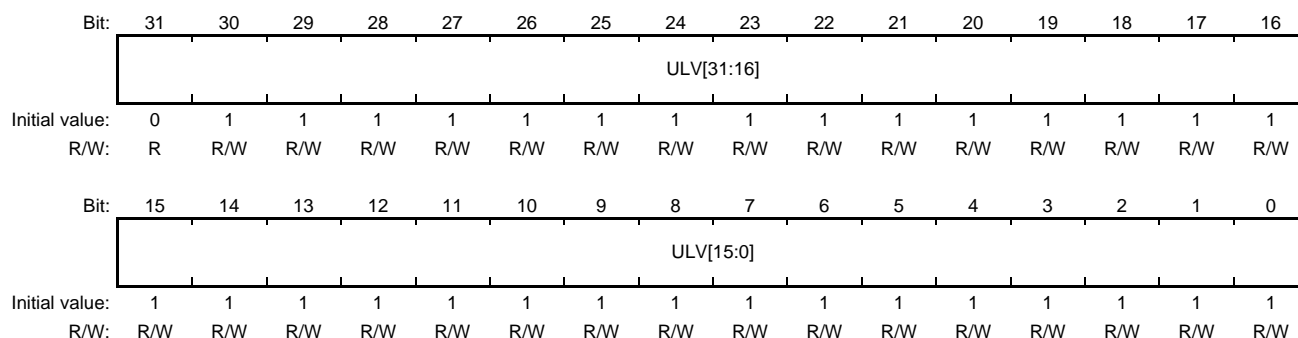
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | CDV[31:0] | H'FFFF_FFFF | R/W | <p>CBS Decrement Value</p> <p>Setting value: -1 to -65536 (H'FFFF_FFFF to H'FFFF_0000)</p> <p>These bits set the decrement for the CBS algorithm.</p> <p>These bits define the decrement value related to sendSlope for CBS algorithm.</p> <p>The value to be written to these bits depends on the Ethernet bit rate and HPφ (high-speed peripheral clock). For details, see section 46.3.6 CBS (Credit-Based Shaping).</p> <p>Note: When write to these bits, corresponding queue's credits are cleared to 0 respectively.</p> |

46.2.33 CBS Upper Limit Register c (CULc) (c = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CUL0 register sets the upper limit for credit values calculated by using the CBS algorithm for transmission queue 2 (for stream class B).

The CUL1 register sets the upper limit for credit values calculated by using the CBS algorithm for transmission queue 3 (for stream class A).



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|----------|---|
| 31 to 0 | ULV[31:0] | H'7FFF_FFFF | R/W * | <p>CBS Upper Limit</p> <p>These bits set the upper limit for credit values calculated by using the CBS algorithm.</p> <p>The setting is a limiting value for error detection and does not normally affect operation of the algorithm.</p> <p>Write a positive value to these bits.</p> <p>For details, see section 46.3.6 CBS (Credit-Based Shaping).</p> |

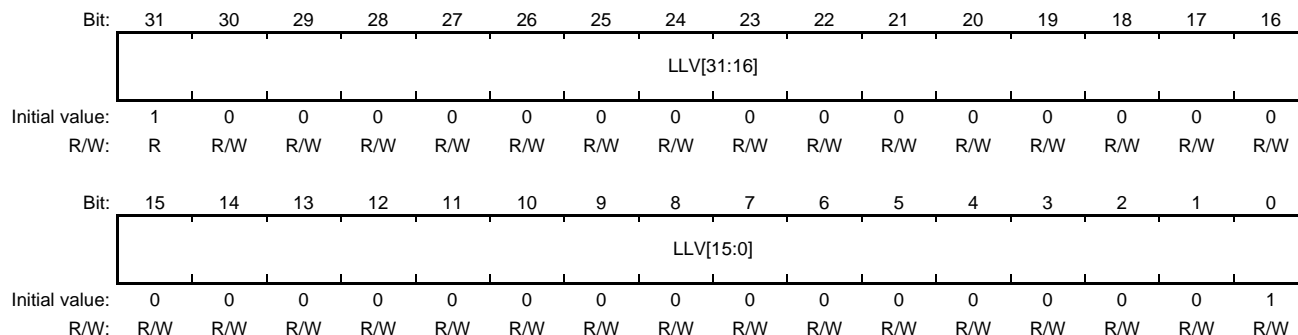
Note: * Bit 31 is read only.

46.2.34 CBS Lower Limit Register c (CLLc) (c = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CUL0 register sets the lower limit for credit values calculated by using the CBS algorithm for transmission queue 2 (for stream class B).

The CUL1 register sets the lower limit for credit values calculated by using the CBS algorithm for transmission queue 3 (for stream class A).



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|------|---|
| 31 to 0 | LLV[31:0] | H'8000_0001 | R/W* | <p>CBS Lower Limit</p> <p>These bits set the lower limit for credit values calculated by using the CBS algorithm.</p> <p>The setting is a limiting value for error detection and does not normally affect operation of the algorithm.</p> <p>Write a negative value to these bits.</p> <p>For details, see section 46.3.6 CBS (Credit-Based Shaping).</p> |

Note: * Bit 31 is read only.

46.2.35 Descriptor Interrupt Control Register (DIC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The DIC register is used to control descriptor interrupts 1 to 15.

When an interrupt source flag is set (a bit from DPF1 to DPF15 bits in the descriptor interrupt status register (DIS) = 1) while the interrupt is enabled, the interrupt is issued.

[Changing condition]

This bit is set to 0 when writing 1 to DID.DPDi.

This bit is set to 1 when writing 1 to DIE.DPSi.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DPE15 | DPE14 | DPE13 | DPE12 | DPE11 | DPE10 | DPE9 | DPE8 | DPE7 | DPE6 | DPE5 | DPE4 | DPE3 | DPE2 | DPE1 | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 | DPE15 | B'0 | R/W | Descriptor Interrupt Enable 15 0: Disabled 1: Enabled |
| 14 | DPE14 | B'0 | R/W | Descriptor Interrupt Enable 14 0: Disabled 1: Enabled |
| 13 | DPE13 | B'0 | R/W | Descriptor Interrupt Enable 13 0: Disabled 1: Enabled |
| 12 | DPE12 | B'0 | R/W | Descriptor Interrupt Enable 12 0: Disabled 1: Enabled |
| 11 | DPE11 | B'0 | R/W | Descriptor Interrupt Enable 11 0: Disabled 1: Enabled |
| 10 | DPE10 | B'0 | R/W | Descriptor Interrupt Enable 10 0: Disabled 1: Enabled |
| 9 | DPE9 | B'0 | R/W | Descriptor Interrupt Enable 9 0: Disabled 1: Enabled |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 8 | DPE8 | B'0 | R/W | Descriptor Interrupt Enable 8 0: Disabled 1: Enabled |
| 7 | DPE7 | B'0 | R/W | Descriptor Interrupt Enable 7 0: Disabled 1: Enabled |
| 6 | DPE6 | B'0 | R/W | Descriptor Interrupt Enable 6 0: Disabled 1: Enabled |
| 5 | DPE5 | B'0 | R/W | Descriptor Interrupt Enable 5 0: Disabled 1: Enabled |
| 4 | DPE4 | B'0 | R/W | Descriptor Interrupt Enable 4 0: Disabled 1: Enabled |
| 3 | DPE3 | B'0 | R/W | Descriptor Interrupt Enable 3 0: Disabled 1: Enabled |
| 2 | DPE2 | B'0 | R/W | Descriptor Interrupt Enable 2 0: Disabled 1: Enabled |
| 1 | DPE1 | B'0 | R/W | Descriptor Interrupt Enable 1 0: Disabled 1: Enabled |
| 0 | — | B'0 | R/W | Reserved These bits are read as 0. The write value should be 0. |

46.2.36 Descriptor Interrupt Status Register (DIS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The DIS register indicates the state of descriptor interrupts.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DPF15 | DPF14 | DPF13 | DPF12 | DPF11 | DPF10 | DPF9 | DPF8 | DPF7 | DPF6 | DPF5 | DPF4 | DPF3 | DPF2 | DPF1 | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 | DPF15 | B'0 | R/W | Descriptor Interrupt Status15 0: The interrupt is not pending. 1: The interrupt is pending. |
| 14 | DPF14 | B'0 | R/W | Descriptor Interrupt Status14 0: The interrupt is not pending. 1: The interrupt is pending. |
| 13 | DPF13 | B'0 | R/W | Descriptor Interrupt Status13 0: The interrupt is not pending. 1: The interrupt is pending. |
| 12 | DPF12 | B'0 | R/W | Descriptor Interrupt Status12 0: The interrupt is not pending. 1: The interrupt is pending. |
| 11 | DPF11 | B'0 | R/W | Descriptor Interrupt Status11 0: The interrupt is not pending. 1: The interrupt is pending. |
| 10 | DPF10 | B'0 | R/W | Descriptor Interrupt Status10 0: The interrupt is not pending. 1: The interrupt is pending. |
| 9 | DPF9 | B'0 | R/W | Descriptor Interrupt Status9 0: The interrupt is not pending. 1: The interrupt is pending. |
| 8 | DPF8 | B'0 | R/W | Descriptor Interrupt Status8 0: The interrupt is not pending. 1: The interrupt is pending. |
| 7 | DPF7 | B'0 | R/W | Descriptor Interrupt Status7 0: The interrupt is not pending. 1: The interrupt is pending. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 6 | DPF6 | B'0 | R/W | Descriptor Interrupt Status6 0: The interrupt is not pending. 1: The interrupt is pending. |
| 5 | DPF5 | B'0 | R/W | Descriptor Interrupt Status5 0: The interrupt is not pending. 1: The interrupt is pending. |
| 4 | DPF4 | B'0 | R/W | Descriptor Interrupt Status4 0: The interrupt is not pending. 1: The interrupt is pending. |
| 3 | DPF3 | B'0 | R/W | Descriptor Interrupt Status3 0: The interrupt is not pending. 1: The interrupt is pending. |
| 2 | DPF2 | B'0 | R/W | Descriptor Interrupt Status2 0: The interrupt is not pending. 1: The interrupt is pending. |
| 1 | DPF1 | B'0 | R/W | Descriptor Interrupt Status1 0: The interrupt is not pending. 1: The interrupt is pending. |
| 0 | — | B'0 | R/W | Reserved These bits are read as 0. The write value should be 0. |

DPF1 to DPF15 Descriptor Interrupt Status Bits:

When DESC.R.DIE is 1 to 15, the corresponding bit indicates completion of the processing of a descriptor within the reception or transmission queue.

When DESC.R.DIE is 0, the descriptor interrupt is not generated.

When DESC.R.DIE is 0001b, in addition a queue specific descriptor interrupt is generated.

Only 0 can be written to these bits.

[Conditions for Changing]

- A bit is set to 0 when the operating mode is not operation mode.
- A bit is set to 1 when a descriptor with DESC.R.DIE set to the corresponding number from 1 to 15 is processed.

46.2.37 Error Interrupt Control Register (EIC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The EIC register controls the AVB-DMAC-related error interrupts.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|------|------|------|-------|-------|-------|-------|-----|-----|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | TBFE | MFFE | TFFE | CULE1 | CULE0 | CLLE1 | CLLE0 | SEE | QEE | MTEE | MREE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 11 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 10 | TBFE | B'0 | R/W | Tx-Buffer Full interrupt Enable While this bit is 1 an interrupt will be generated when EIS.TBFF is 1. [Changing condition] This bit is set to 0 when writing 1 to EID.TBFD. This bit is set to 1 when writing 1 to EIE.TBFS. 0: Disabled 1: Enabled |
| 9 | MFFE | B'0 | R/W | MAC status FIFO Full interrupt Enable While this bit is 1 an interrupt will be generated when EIS.MFFF is 1. [Changing condition] This bit is set to 0 when writing 1 to EID.MFFD. This bit is set to 1 when writing 1 to EIE.MFFS. 0: Disabled 1: Enabled |
| 8 | TFFE | B'0 | R/W | Time Stamp FIFO Full-Error Interrupt Enable When the time stamp FIFO is full (TFFF in the error interrupt status register (EIS) = 1) and the interrupt is enabled, the interrupt is issued. [Changing condition] This bit is set to 0 when writing 1 to EID.TFFD. This bit is set to 1 when writing 1 to EIE.TFFS. 0: Disabled 1: Enabled |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | CULE1 | B'0 | R/W | <p>CBS Upper Limit Error Interrupt Enable (Class A)</p> <p>When the Class A CBS reaches its upper limit (CULF1 in the error interrupt status register (EIS) = 1), the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to EID.CULD1. This bit is set to 1 when writing 1 to EIE.CULS1.</p> <p>0: Disabled 1: Enabled</p> |
| 6 | CULE0 | B'0 | R/W | <p>CBS Upper Limit Error Interrupt Enable (Class B)</p> <p>When the Class B CBS reaches its upper limit (CULF0 in the error interrupt status register (EIS) = 1), the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to EID.CULD0. This bit is set to 1 when writing 1 to EIE.CULS0.</p> <p>0: Disabled 1: Enabled</p> |
| 5 | CLLE1 | B'0 | R/W | <p>CBS Lower Limit Error Interrupt Enable (Class A)</p> <p>When the Class A CBS reaches its lower limit (CLLF1 in the error interrupt status register (EIS) = 1), the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to EID.CLLD1. This bit is set to 1 when writing 1 to EIE.CLLS1.</p> <p>0: Disabled 1: Enabled</p> |
| 4 | CLLE0 | B'0 | R/W | <p>CBS Lower Limit Error Interrupt Enable (Class B)</p> <p>When the Class B CBS reaches its lower limit (CLLF0 in the error interrupt status register (EIS) = 1), the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to EID.CLLD0. This bit is set to 1 when writing 1 to EIE.CLLS0.</p> <p>0: Disabled 1: Enabled</p> |
| 3 | SEE | B'0 | R/W | <p>Separation Error Interrupt Enable</p> <p>While this bit is 1, an interrupt will be generated when the EIS.SEF bit is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to EID.SED. This bit is set to 1 when writing 1 to EIE.SES.</p> <p>0: Disabled 1: Enabled</p> |
| 2 | QEE | B'0 | R/W | <p>Queue Error Interrupt Enable</p> <p>While this bit is 1, an interrupt will be generated when the EIS.QEF bit is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to EID.QED. This bit is set to 1 when writing 1 to EIE.QES.</p> <p>0: Disabled 1: Enabled</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | MTEE | B'0 | R/W | <p>E-MAC Transmission Error Interrupt Enable</p> <p>While this bit is 1, an interrupt will be generated when the EIS.MTEF bit is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to EID.MTED.</p> <p>This bit is set to 1 when writing 1 to EIE.MTES.</p> <p>0: Disabled 1: Enabled</p> |
| 0 | MREE | B'0 | R/W | <p>E-MAC Reception Error Interrupt Enable</p> <p>While this bit is 1, an interrupt will be generated when the EIS.MREF bit is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to EID.MRED.</p> <p>This bit is set to 1 when writing 1 to EIE.MRES.</p> <p>0: Disabled 1: Enabled</p> |

46.2.38 Error Interrupt Status Register (EIS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The EIS register indicates the states of AVB-DMAC-related error interrupts.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|------|------|------|-------|-------|-------|-------|-----|-----|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | QFS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | TBFF | MFFF | TFFF | CULF1 | CULF0 | CLLF1 | CLLF0 | SEF | QEF | MTEF | MREF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 17 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 16 | QFS | B'0 | R/W | Queue Full Error Interrupt Status With the interrupts enabled, this bit indicates that a queue is full (the receive queue r full interrupt status bit (QFFr) or the receive FIFO full interrupt status bit (RFFF) in receive interrupt status register 2 (RIS2) = 1). [Conditions for Changing] — If the receive queue r full interrupt status bit (RIS2.QFFr) and/or the receive queue r full interrupt enable bit in the - receive interrupt control register 2 (RIC2.QFEr) are updated, this bit is also updated. — If the receive FIFO full interrupt status bit (RIS2.RFFF) and/or the receive FIFO full interrupt enable bit (RIC2.RFFE) are updated, this bit is also updated. 0: The interrupt is not pending. 1: The interrupt is pending. |
| 15 to 11 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 10 | TBFF | B'0 | R/W | Tx-Buffer Full Flag This bit indicates that fetching transmit data was delayed because the Tx-Buffer has no sufficient storage available to store fetched frame. It depends on configuration of TGC.TBDt (refer to section 46.3.5(1)) if a full Tx-Buffer is expected during normal operation. No frame gets lost but transmission priority may be influenced. The CPU can only write 0 to this bit. [Changing condition] This bit is set to 0 when leaving OPERATION mode. This bit is set to 1 when frame data has been fetched from URAM but there is no storage available in Tx-Buffer. 0: No interrupt pending 1: Tx-Buffer full condition detected |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 9 | MFFF | B'0 | R/W | <p>E-MAC status FIFO Full Flag</p> <p>This bit indicates that a transmission E-MAC status is overwritten because the E-MAC status FIFO is full (overwrite condition). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a frame with DESCR.MSR has been transmitted by E-MAC, TCCR.MFEN is 1 and TSR.MFFL is 16.</p> <p>This bit is set to 1 when E-MAC detects an error during transmission and TSR.MFFL is 16.</p> <p>0: No interrupt pending 1: E-MAC status FIFO full, oldest E-MAC status lost</p> |
| 8 | TFFF | B'0 | R/W | <p>Time Stamp FIFO Full Error Interrupt Status</p> <p>This bit indicates that a new transmission time stamp has been discarded due to the time-stamp FIFO being full (i.e. has reached the overflow state).</p> <p>Only 0 can be written to the bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — The bit is set to 0 when the operating mode is not operation mode. — The bit is set to 1 when a frame with DESCR.TSR set is transmitted while the time stamp FIFO enable bit in the transmit configuration control register (TCCR.TFEN) is set to 1 and the time stamp FIFO count bit in the transmit status register (TSR.TFFL) is set to 3. <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 7 | CULF1 | B'0 | R/W | <p>CBS Upper Limit Error Interrupt Status (Class A)</p> <p>This bit indicates that CBS counter 1 has exceeded the set upper limit (CUL1.ULV in the CBS upper limit register c (CULc)).</p> <p>Only 0 can be written to the bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — This bit is set to 0 when the operating mode is not operation mode. — This bit is set to 1 when the value of the CBS counter status 1 (Class A) bits in the transmit status register (TSR.CCS1) change from B'00 (indicating a value within the range between the limits) to B'10 (indicating a value over the upper limit). <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 6 | CULF0 | B'0 | R/W | <p>CBS Upper Limit Error Interrupt Status (Class B)</p> <p>This bit indicates that CBS counter 0 has exceeded the set upper limit (CUL0.ULV in the CBS upper limit register c (CULc)).</p> <p>Only 0 can be written to the bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — The bit is set to 0 when the operating mode is not operation mode. — The bit is set to 1 when the value of the CBS counter status 0 (Class B) bit in the transmit status register changes from B'00 (indicating a value within the range between the limits) to B'10 (indicating a value over the upper limit). <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 5 | CLLF1 | B'0 | R/W | <p>CBS Lower Limit Error Interrupt Status (Class A)</p> <p>This bit indicates that CBS counter 1 has fallen below the set lower limit (CLL1.LLV in CBS lower limit register c (CLLc)).</p> <p>Only 0 can be written to the bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — The bit is set to 0 when the operating mode is not operation mode. — The bit is set to 1 when the value of the CBS counter status 1 (Class A) bit in the transmit status register (TSR.CCS1) changes from B'00 (indicating a value within the range between the limits) to B'01 (indicating a value less than the lower limit). <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 4 | CLLF0 | B'0 | R/W | <p>CBS Lower Limit Error Interrupt Status (Class B)</p> <p>This bit indicates that CBS counter 0 has fallen below the set lower limit (CLL0.LLV in the CBS lower limit register c (CLLc)).</p> <p>Only 0 can be written to the bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — The bit is set to 0 when the operating mode is not operation mode. — The bit is set to 1 when the value of the CBS counter status 0 (Class B) bit in the transmit status register (TSR.CCS0) changes from B'00 (indicating a value within the range between the limits) to B'01 (indicating a value less than the lower limit). <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | SEF | B'0 | R/W | <p>Separation Error Flag</p> <p>This bit indicates that a received frame was discarded because it has not matched any configured separation filter for AVB stream data frames.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — This bit is set to 0 when leaving operation mode. — This bit is set to 1 when a valid AVB stream data frame was received by E-MAC but discarded because the RCR.ESF bits are B'10 and no separation filter has matched. <p>0: No interrupt pending. 1: AVB stream data frame has discarded.</p> |
| 2 | QEF | B'0 | R/W | <p>Queue Error Flag</p> <p>This bit indicates that an error has been detected while processing reception or transmit queue or while processing AVTP FIFO entries.</p> <p>Details about the detected error is indicated by ESR.</p> <p>Section 46.3.2(2) gives an overview of detail error conditions and the required interaction.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — This bit is set to 0 when leaving operating mode. — This bit is set to 1 when an error condition is detected. <p>0: No interrupt pending. 1: Interrupt pending.</p> |
| 1 | MTEF | B'0 | R/W | <p>E-MAC Transmission Error Flag</p> <p>This bit indicates that the E-MAC has detected a fault during transmission.</p> <p>For detail, the E-MAC registers have to be checked.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — This bit is set to 0 when leaving operating mode. — This bit is set to 1 when E-MAC detects an error during frame transmission. <p>0: No interrupt pending. 1: E-MAC has reported an error during transmission.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | MREF | B'0 | R/W | <p>E-MAC Reception Error Flag</p> <p>This bit indicates that the E-MAC has detected a fault during reception.</p> <p>For detail, the E-MAC registers have to be checked.</p> <p>Note: When the storage of faulty received frames (RCR.EFFS) is enabled, the E-MAC error code (DESCR.MSC) is stored in the descriptor. By evaluating this information, CPU can identify corrupted frames in URAM.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none">— This bit is set to 0 when leaving operating mode.— This bit is set to 1 when E-MAC detects an error during frame reception. <p>0: No interrupt pending.</p> <p>1: E-MAC has reported an error during reception.</p> |

46.2.39 Receive Interrupt Control Register 0 (RIC0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RIC0 register controls the AVB-DMAC receive interrupts.

When an interrupt source flag is set (a bit from among the receive interrupt status bits in the receive interrupt status register (RIS0.FRf0 to 17) = 1) while the interrupt is enabled, the interrupt is issued.

[Changing condition]

This bit is set to 0 when writing 1 to RID0.FRDr.

This bit is set to 1 when writing 1 to RIE0.FRSr.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | FRE17 | FRE16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | FRE15 | FRE14 | FRE13 | FRE12 | FRE11 | FRE10 | FRE9 | FRE8 | FRE7 | FRE6 | FRE5 | FRE4 | FRE3 | FRE2 | FRE1 | FRE0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 | FRE17 | B'0 | R/W | Receive Frame Enable 17 (Stream) 0: Disabled 1: Enabled |
| 16 | FRE16 | B'0 | R/W | Receive Frame Enable 16 (Stream) 0: Disabled 1: Enabled |
| 15 | FRE15 | B'0 | R/W | Receive Frame Enable 15 (Stream) 0: Disabled 1: Enabled |
| 14 | FRE14 | B'0 | R/W | Receive Frame Enable 14 (Stream) 0: Disabled 1: Enabled |
| 13 | FRE13 | B'0 | R/W | Receive Frame Enable 13 (Stream) 0: Disabled 1: Enabled |
| 12 | FRE12 | B'0 | R/W | Receive Frame Enable 12 (Stream) 0: Disabled 1: Enabled |
| 11 | FRE11 | B'0 | R/W | Receive Frame Enable 11 (Stream) 0: Disabled 1: Enabled |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 10 | FRE10 | B'0 | R/W | Receive Frame Enable 10 (Stream) 0: Disabled 1: Enabled |
| 9 | FRE9 | B'0 | R/W | Receive Frame Enable 9 (Stream) 0: Disabled 1: Enabled |
| 8 | FRE8 | B'0 | R/W | Receive Frame Enable 8 (Stream) 0: Disabled 1: Enabled |
| 7 | FRE7 | B'0 | R/W | Receive Frame Enable 7 (Stream) 0: Disabled 1: Enabled |
| 6 | FRE6 | B'0 | R/W | Receive Frame Enable 6 (Stream) 0: Disabled 1: Enabled |
| 5 | FRE5 | B'0 | R/W | Receive Frame Enable 5 (Stream) 0: Disabled 1: Enabled |
| 4 | FRE4 | B'0 | R/W | Receive Frame Enable 4 (Stream) 0: Disabled 1: Enabled |
| 3 | FRE3 | B'0 | R/W | Receive Frame Enable 3 (Stream) 0: Disabled 1: Enabled |
| 2 | FRE2 | B'0 | R/W | Receive Frame Enable 2 (Stream) 0: Disabled 1: Enabled |
| 1 | FRE1 | B'0 | R/W | Receive Frame Enable 1 (Network Control) 0: Disabled 1: Enabled |
| 0 | FRE0 | B'0 | R/W | Receive Frame Enable 0 (Best Effort) 0: Disabled 1: Enabled |

46.2.40 Receive Interrupt Status Register 0 (RIS0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RIS0 register indicates the states of the AVB-DMAC receive interrupts.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | FRF17 | FRF16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | FRF15 | FRF14 | FRF13 | FRF12 | FRF11 | FRF10 | FRF9 | FRF8 | FRF7 | FRF6 | FRF5 | FRF4 | FRF3 | FRF2 | FRF1 | FRF0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 | FRF17 | B'0 | R/W | Receive Frame Interrupt Status 17 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 16 | FRF16 | B'0 | R/W | Receive Frame Interrupt Status 16 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 15 | FRF15 | B'0 | R/W | Receive Frame Interrupt Status 15 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 14 | FRF14 | B'0 | R/W | Receive Frame Interrupt Status 14 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 13 | FRF13 | B'0 | R/W | Receive Frame Interrupt Status 13 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 12 | FRF12 | B'0 | R/W | Receive Frame Interrupt Status 12 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 11 | FRF11 | B'0 | R/W | Receive Frame Interrupt Status 11 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 10 | FRF10 | B'0 | R/W | Receive Frame Interrupt Status 10 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 9 | FRF9 | B'0 | R/W | Receive Frame Interrupt Status 9 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 8 | FRF8 | B'0 | R/W | Receive Frame Interrupt Status 8 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 7 | FRF7 | B'0 | R/W | Receive Frame Interrupt Status 7 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 6 | FRF6 | B'0 | R/W | Receive Frame Interrupt Status 6 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 5 | FRF5 | B'0 | R/W | Receive Frame Interrupt Status 5 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 4 | FRF4 | B'0 | R/W | Receive Frame Interrupt Status 4 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 3 | FRF3 | B'0 | R/W | Receive Frame Interrupt Status 3 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 2 | FRF2 | B'0 | R/W | Receive Frame Interrupt Status 2 (Stream) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 1 | FRF1 | B'0 | R/W | Receive Frame Interrupt Status 1 (Network Control) 0: The interrupt is not pending. 1: The interrupt is pending. |
| 0 | FRF0 | B'0 | R/W | Receive Frame Interrupt Status 0 (Best Effort) 0: The interrupt is not pending. 1: The interrupt is pending. |

FRF0 to FRF17 Receive Frame Interrupt Status Bits 0 to 17:

Each bit indicates that a corresponding frame has been stored normally in reception queues 0 to 17 and that data are ready for CPU processing.

Only 0 can be written to the bit.

[Conditions for Changing]

- A bit is set to 0 when the operating mode is not operation mode.
- A bit is set to 0 when a value is written to the unread frame counter decrement register i (UFCD i) ($i = 0$ to 4), and this decrements the value of unread frame counter register i (UFCV i) ($i = 0$ to 4) to 0.
- When a frame is stored normally in a reception queue, the corresponding bit is set to 1.

46.2.41 Receive Interrupt Control Register 1 (RIC1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RIC1 register controls the AVB-DMAC receive interrupts.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RFWE | — | — | — | — | — | — | — | — | — | — | — | — | — | RWE17 | RWE16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RWE15 | RWE14 | RWE13 | RWE12 | RWE11 | RWE10 | RWE9 | RWE8 | RWE7 | RWE6 | RWE5 | RWE4 | RWE3 | RWE2 | RWE1 | RWE0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | RFWE | B'0 | R/W | <p>Receive FIFO Warning Interrupt Enable</p> <p>If the reception FIFO reaches the caution level (the value set in the receive FIFO caution level bits in the receive configuration register (RCR.RFCL)) with the corresponding interrupt enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RFWD.</p> <p>This bit is set to 1 when writing 1 to RIE1.RFWS.</p> <p>0: Disabled 1: Enabled</p> |
| 30 to 18 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 17 | RWE17 | B'0 | R/W | <p>Reception Warning interrupt Enable 17</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF17 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD17.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS17.</p> <p>0: Disabled 1: Enabled</p> |
| 16 | RWE16 | B'0 | R/W | <p>Reception Warning interrupt Enable 16</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF16 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD16.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS16.</p> <p>0: Disabled 1: Enabled</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 15 | RWE15 | B'0 | R/W | <p>Reception Warning interrupt Enable 15</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF15 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD15.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS15.</p> <p>0: Disabled 1: Enabled</p> |
| 14 | RWE14 | B'0 | R/W | <p>Reception Warning interrupt Enable 14</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF14 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD14.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS14.</p> <p>0: Disabled 1: Enabled</p> |
| 13 | RWE13 | B'0 | R/W | <p>Reception Warning interrupt Enable 13</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF13 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD13.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS13.</p> <p>0: Disabled 1: Enabled</p> |
| 12 | RWE12 | B'0 | R/W | <p>Reception Warning interrupt Enable 12</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF12 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD12.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS12.</p> <p>0: Disabled 1: Enabled</p> |
| 11 | RWE11 | B'0 | R/W | <p>Reception Warning interrupt Enable 11</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF11 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD11.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS11.</p> <p>0: Disabled 1: Enabled</p> |
| 10 | RWE10 | B'0 | R/W | <p>Reception Warning interrupt Enable 10</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF10 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD10.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS10.</p> <p>0: Disabled 1: Enabled</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 9 | RWE9 | B'0 | R/W | <p>Reception Warning interrupt Enable 9</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF9 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD9.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS9.</p> <p>0: Disabled 1: Enabled</p> |
| 8 | RWE8 | B'0 | R/W | <p>Reception Warning interrupt Enable 8</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF8 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD18.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS18.</p> <p>0: Disabled 1: Enabled</p> |
| 7 | RWE7 | B'0 | R/W | <p>Reception Warning interrupt Enable 7</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF7 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD7.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS7.</p> <p>0: Disabled 1: Enabled</p> |
| 6 | RWE6 | B'0 | R/W | <p>Reception Warning interrupt Enable 6</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF6 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD6.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS6.</p> <p>0: Disabled 1: Enabled</p> |
| 5 | RWE5 | B'0 | R/W | <p>Reception Warning interrupt Enable 5</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF5 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD5.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS5.</p> <p>0: Disabled 1: Enabled</p> |
| 4 | RWE4 | B'0 | R/W | <p>Reception Warning interrupt Enable 4</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF4 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD4.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS4.</p> <p>0: Disabled 1: Enabled</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | RWE3 | B'0 | R/W | <p>Reception Warning interrupt Enable 3</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF3 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD3.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS3.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 2 | RWE2 | B'0 | R/W | <p>Reception Warning interrupt Enable 2</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF2 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD2.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS2.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 1 | RWE1 | B'0 | R/W | <p>Reception Warning interrupt Enable 1</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF1 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD1.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS1.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 0 | RWE0 | B'0 | R/W | <p>Reception Warning interrupt Enable 0</p> <p>While this bit is 1 an interrupt will be generated when RIS1.RWF0 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID1.RWD0.</p> <p>This bit is set to 1 when writing 1 to RIE1.RWS0.</p> <p>0: Disabled</p> <p>1: Enabled</p> |

46.2.42 Receive Interrupt Status Register 1 (RIS1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RIS1 register indicates the states of the AVB-DMAC receive interrupts.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RFWF | — | — | — | — | — | — | — | — | — | — | — | — | — | RWF17 | RWF16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RWF15 | RWF14 | RWF13 | RWF12 | RWF11 | RWF10 | RWF9 | RWF8 | RWF7 | RWF6 | RWF5 | RWF4 | RWF3 | RWF2 | RWF1 | RWF0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | RFWF | B'0 | R/W | <p>Receive FIFO Warning Interrupt Status</p> <p>This bit indicates that the reception FIFO has exceeded the set caution level (the value set in the receive FIFO caution level bits in the receive configuration register (RCR.RFCL)).</p> <p>Only 0 can be written to the bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — The bit is set to 0 when the operating mode is not operation mode. — The bit is set to 1 when the reception FIFO exceeded the set caution level (the value set in the receive FIFO caution level bits in the receive configuration register (RCR.RFCL)). <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 30 to 18 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 17 | RWF17 | B'0 | R/W | <p>Reception Warning Flag 17</p> <p>This bit indicates that the unread frame counter of receive queue 17 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA21 is 1).</p> <p>This bit is set to 0 when CPU writes to UFC4.DV1 and UFCV4.CV1 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j (UFCV4.CV1 + 1 == UFCW.WLj).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 16 | RWF16 | B'0 | R/W | <p>Reception Warning Flag 16</p> <p>This bit indicates that the unread frame counter of receive queue 16 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA20 is 1).</p> <p>This bit is set to 0 when CPU writes to UFC4.DV0 and UFCV4.CV0 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV4.CV0 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |
| 15 | RWF15 | B'0 | R/W | <p>Reception Warning Flag 15</p> <p>This bit indicates that the unread frame counter of receive queue 15 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA19 is 1).</p> <p>This bit is set to 0 when CPU writes to UFC3.DV3 and UFCV3.CV3 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV3.CV3 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |
| 14 | RWF14 | B'0 | R/W | <p>Reception Warning Flag 14</p> <p>This bit indicates that the unread frame counter of receive queue 14 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA18 is 1).</p> <p>This bit is set to 0 when CPU writes to UFC3.DV2 and UFCV3.CV2 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV3.CV2 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 13 | RWF13 | B'0 | R/W | <p>Reception Warning Flag 13</p> <p>This bit indicates that the unread frame counter of receive queue 13 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA17 is 1).</p> <p>This bit is set to 0 when CPU writes to UFC3.DV1 and UFCV3.CV1 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV3.CV1 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |
| 12 | RWF12 | B'0 | R/W | <p>Reception Warning Flag 12</p> <p>This bit indicates that the unread frame counter of receive queue 12 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA16 is 1).</p> <p>This bit is set to 0 when CPU writes to UFC3.DV0 and UFCV3.CV0 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV3.CV0 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |
| 11 | RWF11 | B'0 | R/W | <p>Reception Warning Flag 11</p> <p>This bit indicates that the unread frame counter of receive queue 11 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA15 is 1).</p> <p>This bit is set to 0 when CPU writes to UFC2.DV3 and UFCV2.CV3 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV2.CV3 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 10 | RWF10 | B'0 | R/W | <p>Reception Warning Flag 10</p> <p>This bit indicates that the unread frame counter of receive queue 10 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA14 is 1).</p> <p>This bit is set to 0 when CPU writes to UFCV2.DV2 and UFCV2.CV2 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV2.CV2 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |
| 9 | RWF9 | B'0 | R/W | <p>Reception Warning Flag 9</p> <p>This bit indicates that the unread frame counter of receive queue 9 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA13 is 1).</p> <p>This bit is set to 0 when CPU writes to UFCV2.DV1 and UFCV2.CV1 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV2.CV1 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |
| 8 | RWF8 | B'0 | R/W | <p>Reception Warning Flag 8</p> <p>This bit indicates that the unread frame counter of receive queue 8 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA12 is 1).</p> <p>This bit is set to 0 when CPU writes to UFCV2.DV0 and UFCV2.CV0 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV2.CV0 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | RWF7 | B'0 | R/W | <p>Reception Warning Flag 7</p> <p>This bit indicates that the unread frame counter of receive queue 7 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA11 is 1).</p> <p>This bit is set to 0 when CPU writes to UFCV1.DV3 and UFCV1.CV3 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV1.CV3 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |
| 6 | RWF6 | B'0 | R/W | <p>Reception Warning Flag 6</p> <p>This bit indicates that the unread frame counter of receive queue 6 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA10 is 1).</p> <p>This bit is set to 0 when CPU writes to UFCV1.DV2 and UFCV1.CV2 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV1.CV2 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |
| 5 | RWF5 | B'0 | R/W | <p>Reception Warning Flag 5</p> <p>This bit indicates that the unread frame counter of receive queue 5 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA9 is 1).</p> <p>This bit is set to 0 when CPU writes to UFCV1.DV1 and UFCV1.CV1 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV1.CV1 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 4 | RWF4 | B'0 | R/W | <p>Reception Warning Flag 4</p> <p>This bit indicates that the unread frame counter of receive queue 4 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA8 is 1).</p> <p>This bit is set to 0 when CPU writes to UFCV1.DV0 and UFCV1.CV0 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV1.CV0 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |
| 3 | RWF3 | B'0 | R/W | <p>Reception Warning Flag 3</p> <p>This bit indicates that the unread frame counter of receive queue 3 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA7 is 1).</p> <p>This bit is set to 0 when CPU writes to UFCV0.DV3 and UFCV0.CV3 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV0.CV3 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |
| 2 | RWF2 | B'0 | R/W | <p>Reception Warning Flag 2</p> <p>This bit indicates that the unread frame counter of receive queue 2 has reached the configured warning level.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when CPU request a reload of the base address (DLR.LBA6 is 1).</p> <p>This bit is set to 0 when CPU writes to UFCV0.DV2 and UFCV0.CV2 is decremented to a value less than the selected warning level UFCW.WLj.</p> <p>This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV0.CV2 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | RWF1 | B'0 | R/W | <p>Reception Warning Flag 1</p> <p>This bit indicates that the unread frame counter of receive queue 1 has reached the configured warning level. The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode. This bit is set to 0 when CPU request a reload of the base address (DLR.LBA5 is 1). This bit is set to 0 when CPU writes to UFC0.DV1 and UFCV0.CV1 is decremented to a value less than the selected warning level UFCW.WLj. This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV0.CV1 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |
| 0 | RWF0 | B'0 | R/W | <p>Reception Warning Flag 0</p> <p>This bit indicates that the unread frame counter of receive queue 0 has reached the configured warning level. The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode. This bit is set to 0 when CPU request a reload of the base address (DLR.LBA4 is 1). This bit is set to 0 when CPU writes to UFC0.DV0 and UFCV0.CV0 is decremented to a value less than the selected warning level UFCW.WLj. This bit is set to 1 when a frame is stored in queue r and the unread frame counter value r is incremented to the selected warning level j ($UFCV0.CV0 + 1 == UFCW.WLj$).</p> <p>0: No interrupt pending 1: Unread frame counter warning level reached</p> |

46.2.43 Receive Interrupt Control Register 2 (RIC2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RIC2 register controls the AVB-DMAC receive interrupts.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RFFE | — | — | — | — | — | — | — | — | — | — | — | — | — | QFE17 | QFE16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | QFE15 | QFE14 | QFE13 | QFE12 | QFE11 | QFE10 | QFE9 | QFE8 | QFE7 | QFE6 | QFE5 | QFE4 | QFE3 | QFE2 | QFE1 | QFE0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | RFFE | B'0 | R/W | Receive FIFO Full Interrupt Enable When the reception FIFO is full and the interrupt is enabled, the interrupt is issued. [Changing condition] This bit is set to 0 when writing 1 to RID2.RFFD. This bit is set to 1 when writing 1 to RIE2.RFFS. 0: Disabled 1: Enabled |
| 30 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 | QFE17 | B'0 | R/W | Receive Queue 17 (Stream) Full Interrupt Enable When a reception queue is full and the interrupt is enabled, the interrupt is issued. [Changing condition] This bit is set to 0 when writing 1 to RID2.QFD17. This bit is set to 1 when writing 1 to RIE2.QFS17. 0: Disabled 1: Enabled |
| 16 | QFE16 | B'0 | R/W | Receive Queue 16 (Stream) Full Interrupt Enable When a reception queue is full and the interrupt is enabled, the interrupt is issued. [Changing condition] This bit is set to 0 when writing 1 to RID2.QFD16. This bit is set to 1 when writing 1 to RIE2.QFS16. 0: Disabled 1: Enabled |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 15 | QFE15 | B'0 | R/W | <p>Receive Queue 15 (Stream) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD15.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS15.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 14 | QFE14 | B'0 | R/W | <p>Receive Queue 14 (Stream) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD14.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS14.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 13 | QFE13 | B'0 | R/W | <p>Receive Queue 13 (Stream) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD13.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS13.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 12 | QFE12 | B'0 | R/W | <p>Receive Queue 12 (Stream) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD12.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS12.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 11 | QFE11 | B'0 | R/W | <p>Receive Queue 11 (Stream) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD11.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS11.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 10 | QFE10 | B'0 | R/W | <p>Receive Queue 10 (Stream) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD10.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS10.</p> <p>0: Disabled</p> <p>1: Enabled</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 9 | QFE9 | B'0 | R/W | <p>Receive Queue 9 (Stream) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD9.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS9.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 8 | QFE8 | B'0 | R/W | <p>Receive Queue 8 (Stream) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD8.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS8.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 7 | QFE7 | B'0 | R/W | <p>Receive Queue 7 (Stream) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD7.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS7.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 6 | QFE6 | B'0 | R/W | <p>Receive Queue 6 (Stream) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD6.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS6.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 5 | QFE5 | B'0 | R/W | <p>Receive Queue 5 (Stream) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD5.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS5.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 4 | QFE4 | B'0 | R/W | <p>Receive Queue 4 (Stream) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD4.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS4.</p> <p>0: Disabled</p> <p>1: Enabled</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 3 | QFE3 | B'0 | R/W | <p>Receive Queue 3 (Stream) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD3.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS3.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 2 | QFE2 | B'0 | R/W | <p>Receive Queue 2 (Stream) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD2.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS2.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 1 | QFE1 | B'0 | R/W | <p>Receive Queue 1 (Network Control) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD1.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS1.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 0 | QFE0 | B'0 | R/W | <p>Receive Queue 0 (Best Effort) Full Interrupt Enable</p> <p>When a reception queue is full and the interrupt is enabled, the interrupt is issued.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID2.QFD0.</p> <p>This bit is set to 1 when writing 1 to RIE2.QFS0.</p> <p>0: Disabled</p> <p>1: Enabled</p> |

46.2.44 Receive Interrupt Status Register 2 (RIS2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RIS2 register indicates the states of the AVB-DMAC receive interrupts.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RFFF | — | — | — | — | — | — | — | — | — | — | — | — | — | QFF17 | QFF16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | QFF15 | QFF14 | QFF13 | QFF12 | QFF11 | QFF10 | QFF9 | QFF8 | QFF7 | QFF6 | QFF5 | QFF4 | QFF3 | QFF2 | QFF1 | QFF0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | RFFF | B'0 | R/W | <p>Receive FIFO Full Interrupt Status</p> <p>This bit indicates that a frame was received but storing it was not possible due to the reception FIFO being full.</p> <p>When receiving a frame is not possible, the frame will be discarded.</p> <p>Other information regarding discarded frames is not retained. Even if the frame is discarded, this bit may also be set to 1 if the E-MAC determines that the frame is an error frame</p> <p>Only 0 can be written to the bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — The bit is set to 0 when the operating mode is not operation mode. — The bit is set to 1 when frame data provided by the E-MAC cannot be stored in the reception FIFO. <p>0: The interrupt is not pending. 1: The interrupt is pending</p> |
| 30 to 18 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 17 | QFF17 | B'0 | R/W | <p>Receive Queue 17 (Stream) Full Interrupt Status</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 16 | QFF16 | B'0 | R/W | <p>Receive Queue 16 (Stream) Full Interrupt Status</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 15 | QFF15 | B'0 | R/W | <p>Receive Queue 15 (Stream) Full Interrupt Status</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 14 | QFF14 | B'0 | R/W | <p>Receive Queue 14 (Stream) Full Interrupt Status</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 13 | QFF13 | B'0 | R/W | Receive Queue 13 (Stream) Full Interrupt Status 0: The interrupt is not pending. 1: The interrupt is pending. |
| 12 | QFF12 | B'0 | R/W | Receive Queue 12 (Stream) Full Interrupt Status 0: The interrupt is not pending. 1: The interrupt is pending. |
| 11 | QFF11 | B'0 | R/W | Receive Queue 11 (Stream) Full Interrupt Status 0: The interrupt is not pending. 1: The interrupt is pending. |
| 10 | QFF10 | B'0 | R/W | Receive Queue 10 (Stream) Full Interrupt Status 0: The interrupt is not pending. 1: The interrupt is pending. |
| 9 | QFF9 | B'0 | R/W | Receive Queue 9 (Stream) Full Interrupt Status 0: The interrupt is not pending. 1: The interrupt is pending. |
| 8 | QFF8 | B'0 | R/W | Receive Queue 8 (Stream) Full Interrupt Status 0: The interrupt is not pending. 1: The interrupt is pending. |
| 7 | QFF7 | B'0 | R/W | Receive Queue 7 (Stream) Full Interrupt Status 0: The interrupt is not pending. 1: The interrupt is pending. |
| 6 | QFF6 | B'0 | R/W | Receive Queue 6 (Stream) Full Interrupt Status 0: The interrupt is not pending. 1: The interrupt is pending. |
| 5 | QFF5 | B'0 | R/W | Receive Queue 5 (Stream) Full Interrupt Status 0: The interrupt is not pending. 1: The interrupt is pending. |
| 4 | QFF4 | B'0 | R/W | Receive Queue 4 (Stream) Full Interrupt Status 0: The interrupt is not pending. 1: The interrupt is pending. |
| 3 | QFF3 | B'0 | R/W | Receive Queue 3 (Stream) Full Interrupt Status 0: The interrupt is not pending. 1: The interrupt is pending. |
| 2 | QFF2 | B'0 | R/W | Receive Queue 2 (Stream) Full Interrupt Status 0: The interrupt is not pending. 1: The interrupt is pending. |
| 1 | QFF1 | B'0 | R/W | Receive Queue 1 (Network Control) Full Interrupt Status 0: The interrupt is not pending. 1: The interrupt is pending. |
| 0 | QFF0 | B'0 | R/W | Receive Queue 0 (Best Effort) Full Interrupt Status 0: The interrupt is not pending. 1: The interrupt is pending. |

QFF0 to 17 Receive 0 to 17 Full Interrupt Status Bits:

These bits indicate that reception queue r did not have space for storing a received frame.

A reception queue is treated as full when it has no descriptors (descriptor type (DESCR.DT) = FEMPTY, FEMPTY_IS, FEMPTY_IC, or FEMPTY_ND) available or reaches the set level for stopping.

Note: If no FEMPTY descriptors or no empty space for descriptors remains in the queue during storing of a divided frame (see section 46.3.4(3)(a) Storing Frame Data in the Descriptor Data Area, for storing a frame as a divided frame), an error frame is stored in the queue. Such error frames are treated as descriptor sequence errors.

[Conditions for Changing]

- A bit is set to 0 when the operating mode is not operation mode.
- A bit is set to 1 when reception queue r has no space available for storage.
- A bit is set to 1 when a SW defined stop point (EOS descriptor) reached within a split frame.
- A bit is set to 1 when the unread frame counter (unread frame counter register i (UFCVi) (i = 0 to 4)) reaches the set level for stopping.

46.2.45 Transmit Interrupt Control Register (TIC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The TIC register controls the AVB-DMAC transmit interrupts.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|------|------|------|------|-----|-----|-----|-----|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | TDPE3 | TDPE2 | TDPE1 | TDPE0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | MFWE | MFUE | TFWE | TFUE | — | — | — | — | FTE3 | FTE2 | FTE1 | FTE0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 20 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 19 | TDPE3 | B'0 | R/W | Transmit Descriptor Processed interrupt Enable 3 While this bit is 1 an interrupt will be generated when TIS.TDPF3 is 1. [Changing condition] This bit is set to 0 when writing 1 to TID.TDPD3. This bit is set to 1 when writing 1 to TIE.TDPS3. 0: Disabled 1: Enabled |
| 18 | TDPE2 | B'0 | R/W | Transmit Descriptor Processed interrupt Enable 2 While this bit is 1 an interrupt will be generated when TIS.TDPF2 is 1. [Changing condition] This bit is set to 0 when writing 1 to TID.TDPD2. This bit is set to 1 when writing 1 to TIE.TDPS2. 0: Disabled 1: Enabled |
| 17 | TDPE1 | B'0 | R/W | Transmit Descriptor Processed interrupt Enable 1 While this bit is 1 an interrupt will be generated when TIS.TDPF1 is 1. [Changing condition] This bit is set to 0 when writing 1 to TID.TDPD1. This bit is set to 1 when writing 1 to TIE.TDPS1. 0: Disabled 1: Enabled |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 16 | TDPE0 | B'0 | R/W | <p>Transmit Descriptor Processed interrupt Enable 0</p> <p>While this bit is 1 an interrupt will be generated when TIS.TDPF0 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to TID.TDPD0.</p> <p>This bit is set to 1 when writing 1 to TIE.TDPS0.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 15 to 12 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 11 | MFWE | B'0 | R/W | <p>MAC status FIFO Warning interrupt Enable</p> <p>While this bit is 1 an interrupt will be generated when TIS.MFWF is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to TID.MFWD.</p> <p>This bit is set to 1 when writing 1 to TIE.MFWS.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 10 | MFUE | B'0 | R/W | <p>MAC status FIFO Updated interrupt Enable</p> <p>While this bit is 1 an interrupt will be generated when TIS.MFUF is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to TID.MFUD.</p> <p>This bit is set to 1 when writing 1 to TIE.MFUS.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 9 | TFWE | B'0 | R/W | <p>Time Stamp FIFO Warning Interrupt Enable</p> <p>When the time-stamp FIFO reaches the warning level while the interrupt is enabled, the interrupt is issued.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 8 | TFUE | B'0 | R/W | <p>Time Stamp FIFO Update Interrupt Enable</p> <p>When the time-stamp FIFO is updated while the interrupt is enabled, the interrupt is issued.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 7 to 4 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 3 | FTE3 | B'0 | R/W | <p>Frame Transmitted interrupt Enable 3</p> <p>While this bit is 1 an interrupt will be generated when TIS.FTF3 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to TID.FTD3.</p> <p>This bit is set to 1 when writing 1 to TIE.FTS3.</p> <p>0: Disabled</p> <p>1: Enabled</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 2 | FTE2 | B'0 | R/W | <p>Frame Transmitted interrupt Enable 2</p> <p>While this bit is 1 an interrupt will be generated when TIS.FTF2 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to TID.FTD2.</p> <p>This bit is set to 1 when writing 1 to TIE.FTS2.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 1 | FTE1 | B'0 | R/W | <p>Frame Transmitted interrupt Enable 1</p> <p>While this bit is 1 an interrupt will be generated when TIS.FTF1 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to TID.FTD1.</p> <p>This bit is set to 1 when writing 1 to TIE.FTS1.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 0 | FTE0 | B'0 | R/W | <p>Frame Transmitted interrupt Enable 0</p> <p>While this bit is 1 an interrupt will be generated when TIS.FTF0 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to TID.FTD0.</p> <p>This bit is set to 1 when writing 1 to TIE.FTS0.</p> <p>0: Disabled</p> <p>1: Enabled</p> |

46.2.46 Transmit Interrupt Status Register (TIS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The TIS register indicates the states of the AVB-DMAC transmit interrupts.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|------|------|------|------|-----|-----|-----|-----|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | TDPF3 | TDPF2 | TDPF1 | TDPF0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | MFWF | MFUF | TFWF | TFUF | — | — | — | — | FTF3 | FTF2 | FTF1 | FTF0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 20 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 19 | TDPF3 | B'0 | R/W | Transmit Descriptor Processed Flag 3 This bit indicates that a descriptor in transmit queue 3 has been processed where DESCR.DIE is B'0001. Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1. The CPU can only write 0 to this bit. [Changing condition] This bit is set to 0 when leaving OPERATION mode. This bit is set to 1 when a descriptor in transmit queue 3 has been processed where DESCR.DIE is 1. 0: No interrupt pending 1: Descriptor interrupt pending |
| 18 | TDPF2 | B'0 | R/W | Transmit Descriptor Processed Flag 2 This bit indicates that a descriptor in transmit queue 2 has been processed where DESCR.DIE is B'0001. Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1. The CPU can only write 0 to this bit. [Changing condition] This bit is set to 0 when leaving OPERATION mode. This bit is set to 1 when a descriptor in transmit queue 2 has been processed where DESCR.DIE is 1. 0: No interrupt pending 1: Descriptor interrupt pending |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 17 | TDPF1 | B'0 | R/W | <p>Transmit Descriptor Processed Flag 1</p> <p>This bit indicates that a descriptor in transmit queue 1 has been processed where DESC.R.DIE is B'0001.</p> <p>Note: The descriptor with DESC.R.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in transmit queue 1 has been processed where DESC.R.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |
| 16 | TDPF0 | B'0 | R/W | <p>Transmit Descriptor Processed Flag 0</p> <p>This bit indicates that a descriptor in transmit queue 0 has been processed where DESC.R.DIE is B'0001.</p> <p>Note: The descriptor with DESC.R.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in transmit queue 0 has been processed where DESC.R.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |
| 15 to 12 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 11 | MFWF | B'0 | R/W | <p>E-MAC status FIFO Warning Flag</p> <p>This bit indicates that the warning level of the E-MAC status FIFO (12 out of 16 entries) has been reached.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when TCCR.MFEN is 0.</p> <p>This bit is set to 0 when writing 1 to TCCR.MFR.</p> <p>This bit is set to 1 when a frame with DESC.MSR has been transmitted by E-MAC and the E-MAC Status FIFO contains already 11 entries (TSR.MFFL is 11).</p> <p>This bit is set to 1 when E-MAC detects an error during transmission and the E-MAC Status FIFO contains already 11 entries (TSR.MFFL is 11).</p> <p>0: No interrupt pending 1: Tx Status FIFO warning level has been reached</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 10 | MFUF | B'0 | R/W | <p>E-MAC status FIFO Updated Flag</p> <p>This bit indicates that the E-MAC status FIFO has been updated after the E-MAC has transmitted a frame.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 0 when TCCR.MFEN is 0.</p> <p>This bit is set to 0 when writing 1 to TCCR.MFR.</p> <p>This bit is set to 1 when a frame with DESCR.MSR has been transmitted by E-MAC and TCCR.MFEN is 1.</p> <p>This bit is set to 1 when E-MAC detects an error during transmission and TCCR.MFEN is 1.</p> <p>0: No interrupt pending 1: Tx Status FIFO has been updated</p> |
| 9 | TFWF | B'0 | R/W | <p>Time Stamp FIFO Warning Interrupt Status</p> <p>This bit indicates that the transmission time-stamp FIFO has reached the warning level.</p> <p>Only 0 can be written to the bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — The bit is set to 0 when the operating mode is not operation mode and when the time stamp FIFO enable bit in the transmit configuration control register (TCCR.TFEN) is 0. — The bit is set to 1 after a frame including DESCR.TSR set has been transmitted and one entry has already been stored in the time-stamp FIFO. <p>0: The interrupt is not pending. 1: The time-stamp FIFO has reached the warning level.</p> |
| 8 | TFUF | B'0 | R/W | <p>Time Stamp FIFO Update Interrupt Status</p> <p>This bit indicates that the transmission time-stamp FIFO has been updated.</p> <p>Only 0 can be written to the bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — The bit is set to 0 when the operating mode is not operation mode, when the time stamp FIFO enable bit in the transmit configuration control register (TCCR.TFEN) is 0, and when 1 is written to the time stamp FIFO release bit in the transmit configuration control register (TCCR.TFR). — The bit is set to 1 when the time stamp FIFO enable bit (TCCR.TFEN) is 1 after a frame including DESCR.TSR set has been transmitted. <p>0: The interrupt is not pending. 1: The time-stamp FIFO has been updated.</p> |
| 7 to 4 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | FTF3 | B'0 | R/W | <p>Frame Transmitted Flag 3</p> <p>This bit indicates that from transmit queue a frame is transmitted by E-MAC.</p> <p>Note: This interrupt flag refers to the end of frame transmission by E-MAC whereas the descriptor interrupt (DIS.DPFI) refers to the end of processing storage element.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a frame from transmit queue 3 has been transmitted by the E-MAC.</p> <p>0: No interrupt pending 1: Frame transmitted by E-MAC</p> |
| 2 | FTF2 | B'0 | R/W | <p>Frame Transmitted Flag 2</p> <p>This bit indicates that from transmit queue a frame is transmitted by E-MAC.</p> <p>Note: This interrupt flag refers to the end of frame transmission by E-MAC whereas the descriptor interrupt (DIS.DPFI) refers to the end of processing storage element.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a frame from transmit queue 2 has been transmitted by the E-MAC.</p> <p>0: No interrupt pending 1: Frame transmitted by E-MAC</p> |
| 1 | FTF1 | B'0 | R/W | <p>Frame Transmitted Flag 1</p> <p>This bit indicates that from transmit queue a frame is transmitted by E-MAC.</p> <p>Note: This interrupt flag refers to the end of frame transmission by E-MAC whereas the descriptor interrupt (DIS.DPFI) refers to the end of processing storage element.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a frame from transmit queue 1 has been transmitted by the E-MAC.</p> <p>0: No interrupt pending 1: Frame transmitted by E-MAC</p> |
| 0 | FTF0 | B'0 | R/W | <p>Frame Transmitted Flag 0</p> <p>This bit indicates that from transmit queue a frame is transmitted by E-MAC.</p> <p>Note: This interrupt flag refers to the end of frame transmission by E-MAC whereas the descriptor interrupt (DIS.DPFI) refers to the end of processing storage element.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a frame from transmit queue 0 has been transmitted by the E-MAC.</p> <p>0: No interrupt pending 1: Frame transmitted by E-MAC</p> |

46.2.47 Interrupt Summary Status Register (ISS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The ISS register gives a summary of the states of AVB-DMAC-related interrupts.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | DPM15 | DPM14 | DPM13 | DPM12 | DPM11 | DPM10 | DPM9 | DPM8 | DPM7 | DPM6 | DPM5 | DPM4 | DPM3 | DPM2 | DPM1 | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | CGIM | RFWM | MFWM | MFUM | TFWM | TFUM | MM | EM | — | — | — | FTM | RWM | FRM |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 31 | DPM15 | B'0 | R | Descriptor Interrupt 15 Mirror This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE15) and descriptor interrupt status flag (DIS.DPF15) are both 1. 0: The interrupt is not pending. 1: The interrupt is pending. |
| 30 | DPM14 | B'0 | R | Descriptor Interrupt 14 Mirror This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE14) and descriptor interrupt status flag (DIS.DPF14) are both 1. 0: The interrupt is not pending. 1: The interrupt is pending. |
| 29 | DPM13 | B'0 | R | Descriptor Interrupt 13 Mirror This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE13) and descriptor interrupt status flag (DIS.DPF13) are both 1. 0: The interrupt is not pending. 1: The interrupt is pending. |
| 28 | DPM12 | B'0 | R | Descriptor Interrupt 12 Mirror This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE12) and descriptor interrupt status flag (DIS.DPF12) are both 1. 0: The interrupt is not pending. 1: The interrupt is pending. |
| 27 | DPM11 | B'0 | R | Descriptor Interrupt 11 Mirror This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE11) and descriptor interrupt status flag (DIS.DPF11) are both 1. 0: The interrupt is not pending. 1: The interrupt is pending. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 26 | DPM10 | B'0 | R | <p>Descriptor Interrupt 10 Mirror</p> <p>This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE10) and descriptor interrupt status flag (DIS.DPF10) are both 1.</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 25 | DPM9 | B'0 | R | <p>Descriptor Interrupt 9 Mirror</p> <p>This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE9) and descriptor interrupt status flag (DIS.DPF9) are both 1.</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 24 | DPM8 | B'0 | R | <p>Descriptor Interrupt 8 Mirror</p> <p>This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE8) and descriptor interrupt status flag (DIS.DPF8) are both 1.</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 23 | DPM7 | B'0 | R | <p>Descriptor Interrupt 7 Mirror</p> <p>This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE7) and descriptor interrupt status flag (DIS.DPF7) are both 1.</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 22 | DPM6 | B'0 | R | <p>Descriptor Interrupt 6 Mirror</p> <p>This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE6) and descriptor interrupt status flag (DIS.DPF6) are both 1.</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 21 | DPM5 | B'0 | R | <p>Descriptor Interrupt 5 Mirror</p> <p>This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE5) and descriptor interrupt status flag (DIS.DPF5) are both 1.</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 20 | DPM4 | B'0 | R | <p>Descriptor Interrupt 4 Mirror</p> <p>This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE4) and descriptor interrupt status flag (DIS.DPF4) are both 1.</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 19 | DPM3 | B'0 | R | <p>Descriptor Interrupt 3 Mirror</p> <p>This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE3) and descriptor interrupt status flag (DIS.DPF3) are both 1.</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 18 | DPM2 | B'0 | R | <p>Descriptor Interrupt 2 Mirror</p> <p>This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE2) and descriptor interrupt status flag (DIS.DPF2) are both 1.</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 17 | DPM1 | B'0 | R | <p>Descriptor Interrupt 1 Mirror</p> <p>This bit is set to 1 when the given descriptor interrupt enable bit (DIC.DPE1) and descriptor interrupt status flag (DIS.DPF1) are both 1.</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 16 to 14 | — | All 0 | R | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 13 | CGIM | B'0 | R | <p>gPTP Interrupt Mirror</p> <p>This bit is set to 1 when either interrupt-related bit in the two gPTP-related interrupt registers (GIC and GIS) is 1.</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 12 | RFWM | B'0 | R | <p>Receive FIFO Warning Interrupt Mirror</p> <p>This bit is set to 1 when the receive FIFO warning interrupt enable bit (RIC1.RFWE) and receive FIFO warning interrupt status flag (RIS1.RFWF) are both 1.</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 11 | MFWM | B'0 | R | <p>E-MAC status FIFO Warning Mirror</p> <p>This bit indicates that TIS.MFWF is 1 and TIC.MFWE is 1. [Changing condition]</p> <p>This bit is updated when TIS.MFWF or TIC.MFWE changes.</p> <p>0: No interrupt pending 1: E-MAC status FIFO warning interrupt pending</p> |
| 10 | MFUM | B'0 | R | <p>E-MAC status FIFO Updated Mirror</p> <p>This bit indicates that TIS.MFUF is 1 and TIC.MFUE is 1. [Changing condition]</p> <p>This bit is updated when TIS.MFUF or TIC.MFUE changes.</p> <p>0: No interrupt pending 1: E-MAC status FIFO updated interrupt pending</p> |
| 9 | TFWM | B'0 | R | <p>Time Stamp FIFO Warning Interrupt Mirror</p> <p>This bit is set to 1 when the time stamp FIFO warning interrupt enable bit (TIC.TFWE) and time stamp FIFO warning interrupt status flag (TIS.TFWF) are both 1.</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 8 | TFUM | B'0 | R | <p>Time Stamp FIFO Update Mirror</p> <p>This bit is set to 1 when the time stamp FIFO update interrupt enable bit (TIC.TFUE) and time stamp FIFO update interrupt status flag (TIS.TFUF) are both 1.</p> <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 | MM | B'0 | R | <p>E-MAC Interrupt Mirror</p> <p>This bit is set to 1 when an E-MAC interrupt is issued.</p> <p>0: The interrupt is not pending.</p> <p>1: The interrupt is pending.</p> |
| 6 | EM | B'0 | R | <p>Error Interrupt Mirror</p> <p>This bit is set to 1 when an error interrupt is issued (both of a bit in EIS and corresponding to the bit in EIC are 1).</p> <p>0: The interrupt is not pending.</p> <p>1: The interrupt is pending.</p> |
| 5 to 3 | — | All 0 | R | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 2 | FTM | B'0 | R | <p>Frame Transmitted Mirror</p> <p>[Changing condition]</p> <p>This bit is set when any matching pair of TIC.FTEt enable and TIS.FTFt flag are both 1.</p> <p>This bit is set when any matching pair of TIC.TDPEt enable and TIS.TDPFt flag are both 1.</p> <p>0: No interrupt pending</p> <p>1: Frame transmitted interrupt pending</p> |
| 1 | RWM | B'0 | R | <p>Reception Warning Mirror</p> <p>[Changing condition]</p> <p>This bit is set when any matching pair of RIC1.RWEr enable and RIS1.RWFr flag are both 1.</p> <p>0: No interrupt pending</p> <p>1: Frame transmitted interrupt pending</p> |
| 0 | FRM | B'0 | R | <p>Frame Received Mirror</p> <p>[Changing condition]</p> <p>This bit is set when any matching pair of RIC0.FREr enable and RIS0.FRFr flag are both 1.</p> <p>This bit is set when any matching pair of RIC3.RDPEr enable and RIS3.RDPFr flag are both 1.</p> <p>0: No interrupt pending</p> <p>1: Frame received interrupt pending</p> |

46.2.48 Common Interrupt Enable register (CIE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CIE register is used to control the Common Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|------|-----|-----|-----|-----|------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | RFFL | RFWL | CL0M | RQFM |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | CTIE | — | — | — | — | — | — | — | CRIE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 20 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 19 | RFFL | B'0 | R/W | Rx-FIFO Full interrupt Line select This bit selects the interrupt line of notification flagged by RIS2.RFFF. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 18 | RFWL | B'0 | R/W | Rx-FIFO Warning interrupt Line select This bit selects the interrupt line of notification flagged by RIS1.RFWF. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 17 | CL0M | B'0 | R/W | Common Line 0 Mode This bit controls if all data related interrupts using a common interrupt line or if queue specific interrupt lines for data related interrupts are used. The selected mode has no influence to internal flagging in EthernetAVB-IF register (e.g. ISS). To prevent interrupt notifications on different lines, it is recommended to set all CIE.CTIE, CIE.CRIE, and DIL.DPLi (i = 2 to 15) to 0 when this bit is 0. 0: Use common interrupt line 0 1: Use queue specific interrupt line 0 |
| 16 | RQFM | B'0 | R/W | Reception Queue Full Mode This bit controls if queue full notification is mapped to error interrupt line or to queue data interrupt line. The selected mode has no influence to internal flagging in EthernetAVB-IF register (e.g. ISS). 0: Use for error interrupt line 1: Use for queue specific interrupt line |
| 15 to 9 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 8 | CTIE | B'0 | R/W | <p>Common Transmit Interrupt Enable</p> <p>This bit controls transmit related interrupt line outputs of EthernetAVB-IF. It has no influence to internal flagging in EthernetAVB-IF register (e.g. ISS).</p> <p>These lines are controlled by this bit:</p> <p>line0_Tx[3:0] -- individual transmit queue interrupts</p> <p>Transmission interrupts mapped on other lines are not influenced.</p> <p>0: Disabled 1: Enabled</p> |
| 7 to 1 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 0 | CRIE | B'0 | R/W | <p>Common Receive Interrupt enable</p> <p>This bit controls receive related interrupt line outputs of EthernetAVB-IF. It has no influence to internal flagging in EthernetAVB-IF register (e.g. ISS).</p> <p>These lines are controlled by this bit:</p> <p>line0_Rx[17:0] -- individual receive queue interrupts</p> <p>Receive interrupts mapped on other lines are not influenced.</p> <p>0: Disabled 1: Enabled</p> |

46.2.49 Reception Interrupt Control register 3 (RIC3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RIC3 register controls the AVB-DMAC receive descriptor interrupts.

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | RDPE 17 | RDPE 16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RDPE 15 | RDPE 14 | RDPE 13 | RDPE 12 | RDPE 11 | RDPE 10 | RDPE9 8 | RDPE7 8 | RDPE6 7 | RDPE5 6 | RDPE4 5 | RDPE3 4 | RDPE2 3 | RDPE1 2 | RDPE 1 | RDPE 0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 | RDPE17 | B'0 | R/W | Receive Descriptor Processed interrupt Enable 17 While this bit is 1 an interrupt will be generated when RIS3.RDPF17 is 1. [Changing condition] This bit is set to 0 when writing 1 to RID3.RDPD17. This bit is set to 1 when writing 1 to RIE3.RDPS17. 0: Disabled 1: Enabled |
| 16 | RDPE16 | B'0 | R/W | Receive Descriptor Processed interrupt Enable 16 While this bit is 1 an interrupt will be generated when RIS3.RDPF16 is 1. [Changing condition] This bit is set to 0 when writing 1 to RID3.RDPD16. This bit is set to 1 when writing 1 to RIE3.RDPS16. 0: Disabled 1: Enabled |
| 15 | RDPE15 | B'0 | R/W | Receive Descriptor Processed interrupt Enable 15 While this bit is 1 an interrupt will be generated when RIS3.RDPF15 is 1. [Changing condition] This bit is set to 0 when writing 1 to RID3.RDPD15. This bit is set to 1 when writing 1 to RIE3.RDPS15. 0: Disabled 1: Enabled |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 14 | RDPE14 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 14</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF14 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD14.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS14.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 13 | RDPE13 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 13</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF13 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD13.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS13.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 12 | RDPE12 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 12</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF12 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD12.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS12.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 11 | RDPE11 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 11</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF11 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD11.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS11.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 10 | RDPE10 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 10</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF10 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD10.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS10.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 9 | RDPE9 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 9</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF9 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD9.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS9.</p> <p>0: Disabled</p> <p>1: Enabled</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 8 | RDPE8 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 8</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF8 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD8.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS8.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 7 | RDPE7 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 7</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF7 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD7.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS7.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 6 | RDPE6 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 6</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF6 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD6.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS6.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 5 | RDPE5 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 5</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF5 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD5.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS5.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 4 | RDPE4 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 4</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF4 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD4.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS4.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 3 | RDPE3 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 3</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF3 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD3.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS3.</p> <p>0: Disabled</p> <p>1: Enabled</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | RDPE2 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 2</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF2 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD2.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS2.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 1 | RDPE1 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 1</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF1 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD1.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS1.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 0 | RDPE0 | B'0 | R/W | <p>Receive Descriptor Processed interrupt Enable 0</p> <p>While this bit is 1 an interrupt will be generated when RIS3.RDPF0 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to RID3.RDPD0.</p> <p>This bit is set to 1 when writing 1 to RIE3.RDPS0.</p> <p>0: Disabled</p> <p>1: Enabled</p> |

46.2.50 Reception Interrupt Status register 3 (RIS3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RIS3 register indicates the states of the AVB-DMAC receive descriptor interrupts.

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | RDPF 17 | RDPF 16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RDPF 15 | RDPF 14 | RDPF 13 | RDPF 12 | RDPF 11 | RDPF 10 | RDPF9 8 | RDPF8 7 | RDPF7 6 | RDPF6 5 | RDPF5 4 | RDPF4 3 | RDPF3 2 | RDPF2 1 | RDPF1 0 | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 | RDPF17 | B'0 | R/W | Receive Descriptor Processed Flag 17 This bit indicates that a descriptor in reception queue 17 has been processed where DESCR.DIE is B'0001. Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1. The CPU can only write 0 to this bit. [Changing condition] This bit is set to 0 when leaving OPERATION mode. This bit is set to 1 when a descriptor in reception queue 17 has been processed where DESCR.DIE is 1. 0: No interrupt pending 1: Descriptor interrupt pending |
| 16 | RDPF16 | B'0 | R/W | Receive Descriptor Processed Flag 16 This bit indicates that a descriptor in reception queue 16 has been processed where DESCR.DIE is B'0001. Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1. The CPU can only write 0 to this bit. [Changing condition] This bit is set to 0 when leaving OPERATION mode. This bit is set to 1 when a descriptor in reception queue 16 has been processed where DESCR.DIE is 1. 0: No interrupt pending 1: Descriptor interrupt pending |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 15 | RDPF15 | B'0 | R/W | <p>Receive Descriptor Processed Flag 15</p> <p>This bit indicates that a descriptor in reception queue 15 has been processed where DESCR.DIE is B'0001.</p> <p>Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 15 has been processed where DESCR.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |
| 14 | RDPF14 | B'0 | R/W | <p>Receive Descriptor Processed Flag 14</p> <p>This bit indicates that a descriptor in reception queue 14 has been processed where DESCR.DIE is B'0001.</p> <p>Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 14 has been processed where DESCR.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |
| 13 | RDPF13 | B'0 | R/W | <p>Receive Descriptor Processed Flag 13</p> <p>This bit indicates that a descriptor in reception queue 13 has been processed where DESCR.DIE is B'0001.</p> <p>Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 13 has been processed where DESCR.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |
| 12 | RDPF12 | B'0 | R/W | <p>Receive Descriptor Processed Flag 12</p> <p>This bit indicates that a descriptor in reception queue 12 has been processed where DESCR.DIE is B'0001.</p> <p>Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 12 has been processed where DESCR.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 11 | RDPF11 | B'0 | R/W | <p>Receive Descriptor Processed Flag 11</p> <p>This bit indicates that a descriptor in reception queue 11 has been processed where DESCR.DIE is B'0001.</p> <p>Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 11 has been processed where DESCR.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |
| 10 | RDPF10 | B'0 | R/W | <p>Receive Descriptor Processed Flag 10</p> <p>This bit indicates that a descriptor in reception queue 10 has been processed where DESCR.DIE is B'0001.</p> <p>Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 10 has been processed where DESCR.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |
| 9 | RDPF9 | B'0 | R/W | <p>Receive Descriptor Processed Flag 9</p> <p>This bit indicates that a descriptor in reception queue 9 has been processed where DESCR.DIE is B'0001.</p> <p>Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 9 has been processed where DESCR.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |
| 8 | RDPF8 | B'0 | R/W | <p>Receive Descriptor Processed Flag 8</p> <p>This bit indicates that a descriptor in reception queue 8 has been processed where DESCR.DIE is B'0001.</p> <p>Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 8 has been processed where DESCR.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | RDPF7 | B'0 | R/W | <p>Receive Descriptor Processed Flag 7</p> <p>This bit indicates that a descriptor in reception queue 7 has been processed where DESC.R.DIE is B'0001.</p> <p>Note: The descriptor with DESC.R.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 7 has been processed where DESC.R.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |
| 6 | RDPF6 | B'0 | R/W | <p>Receive Descriptor Processed Flag 6</p> <p>This bit indicates that a descriptor in reception queue 6 has been processed where DESC.R.DIE is B'0001.</p> <p>Note: The descriptor with DESC.R.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 6 has been processed where DESC.R.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |
| 5 | RDPF5 | B'0 | R/W | <p>Receive Descriptor Processed Flag 5</p> <p>This bit indicates that a descriptor in reception queue 5 has been processed where DESC.R.DIE is B'0001.</p> <p>Note: The descriptor with DESC.R.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 5 has been processed where DESC.R.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |
| 4 | RDPF4 | B'0 | R/W | <p>Receive Descriptor Processed Flag 4</p> <p>This bit indicates that a descriptor in reception queue 4 has been processed where DESC.R.DIE is B'0001.</p> <p>Note: The descriptor with DESC.R.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 4 has been processed where DESC.R.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 3 | RDPF3 | B'0 | R/W | <p>Receive Descriptor Processed Flag 3</p> <p>This bit indicates that a descriptor in reception queue 3 has been processed where DESCR.DIE is B'0001.</p> <p>Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 3 has been processed where DESCR.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |
| 2 | RDPF2 | B'0 | R/W | <p>Receive Descriptor Processed Flag 2</p> <p>This bit indicates that a descriptor in reception queue 2 has been processed where DESCR.DIE is B'0001.</p> <p>Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 2 has been processed where DESCR.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |
| 1 | RDPF1 | B'0 | R/W | <p>Receive Descriptor Processed Flag 1</p> <p>This bit indicates that a descriptor in reception queue 1 has been processed where DESCR.DIE is B'0001.</p> <p>Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 1 has been processed where DESCR.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |
| 0 | RDPF0 | B'0 | R/W | <p>Receive Descriptor Processed Flag 0</p> <p>This bit indicates that a descriptor in reception queue 0 has been processed where DESCR.DIE is B'0001.</p> <p>Note: The descriptor with DESCR.DIE is 1, which sets this bit, sets in addition the universal descriptor interrupt DIS.DPF1.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode.</p> <p>This bit is set to 1 when a descriptor in reception queue 0 has been processed where DESCR.DIE is 1.</p> <p>0: No interrupt pending 1: Descriptor interrupt pending</p> |

46.2.51 gPTP Configuration Control Register (GCCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GCCR register is used to set and control the gPTP (generalized precision time protocol).

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----------|-----|-----|---------|------|------|-----|-----|----------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | LI[2:0] | | SPC | — | — | PGM | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | TCSS[1:0] | | — | — | LMTT | LPTC | LTI | LTO | TCR[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 23 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 22 to 20 | LI[2:0] | B'000 | R/W | Index number of compare unit to be loaded Load Index These bits define the target unit to be updated by GCCR.LPTC. The CPU can only write to these bits if GCCR.LPTC is 0b. |
| 19 | SPC | B'0 | R/W | Start Periodic Comparison This bit defines if the absolute comparison value or the periodicity value is updated to the value of GPTC.PTCV on GCCR.LPTC request. Periodic comparison starts when updating the periodic comparison value; periodic comparison stops when updating the absolute comparison value. The CPU can only write to these bits if GCCR.LPTC is 0b. 0: Absolute comparison value updated by GCCR.LPTC request 1: Periodic comparison value updated by GCCR.LPTC request |
| 18 to 17 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 16 | PGM | B'0 | R/W | Pulse Generation Mode This bit configures if the pulse per second signal (AVB_AVTP_PPS) is derived from the gPTP timer value or the corrected gPTP timer value. A pulse is generated when the second part of the selected timer is incremented. See section 46.3.8(3) Pulse per second. The CPU can only write to this bit if CSR.OPS is CONFIG. The CPU should only change this bit if CCC.GAC is 0b or if increment value used by gPTP timer is 0. 0: Pulse per second is related to gPTP timer value 1: Pulse per second is related to corrected gPTP timer value |
| 15 to 10 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |

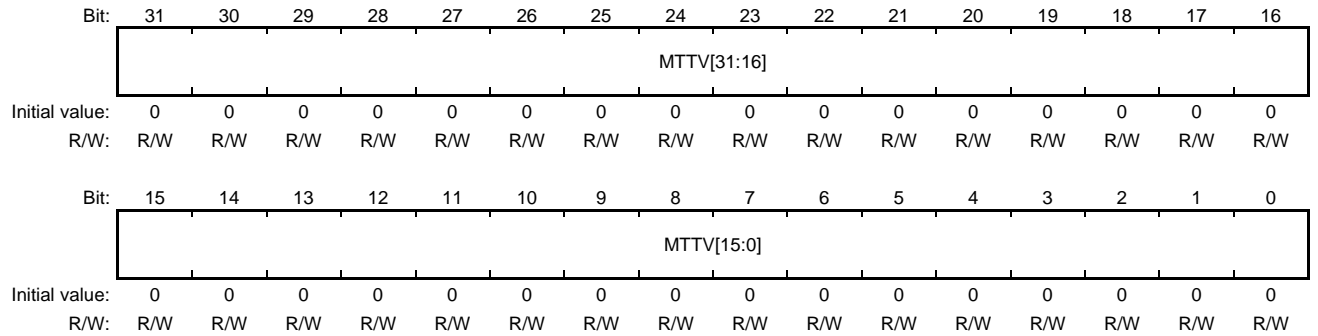
| Bit | Bit Name | Initial Value | R/W | Description |
|------|-----------|---------------|-----|--|
| 9, 8 | TCSS[1:0] | B'00 | R/W | <p>Timer Capture Source Select</p> <p>These bits select the source used for updating the captured timer register (gPTP timer capture register (GCTi.CTV)).</p> <p>These bits should still be controlled when timer control is not being requested (GCCR.TCR = 0).</p> <p>B'00: gPTP timer value B'01: Adjusted gPTP timer value B'10: AVTP presentation time B'11: Setting prohibited</p> |
| 7, 6 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 5 | LMTT | B'1 | R/W | <p>Maximum Transit Time Configuration Request</p> <p>This bit issues requests for configuring the gPTP maximum transit time configuration register (GMTT).</p> <p>Only 1 can be written to the bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — The bit is set to 1 when the operating mode is not operation mode. — The bit is set to 0 when the value of the gPTP maximum transit time configuration register (GMTT) is loaded. <p>0: Setting completed 1: When written: Issue a configuration request. When read: Completion of settings is pending.</p> |
| 4 | LPTC | B'0 | R/W | <p>Presentation Time Compare Value Configuration Request</p> <p>This bit issues requests for configuring the gPTP presentation time comparison register (GPTC).</p> <p>Only 1 can be written to the bit.</p> <p>The CPU cannot write to this bit if CSR.OPS is CONFIG and CCC.GAC is 0.</p> <p>The CPU should not write 1 to this bit when AVTP FIFO of selected compare unit is in use.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — The bit is set to 1 when the operating mode is not operation mode and CCC.GAC is 0b. — The bit is set to 0 when the value of the gPTP presentation time comparison register (GPTC) is loaded. <p>0: Setting completed 1: When written: Issue a configuration request. When read: Completion of settings is pending.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 3 | LTI | B'1 | R/W | <p>Timer Increment Value Configuration Request</p> <p>This bit issues requests for configuring the gPTP timer increment configuration register (GTI).</p> <p>Only 1 can be written to the bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — The bit is set to 1 when the operating mode is not operation mode and CCC.GAC is 0b. — The bit is set to 0 when the value of the gPTP timer increment configuration register (GTI) is loaded. <p>0: Setting completed 1: When written: Issue a configuration request. When read: Completion of settings is pending.</p> |
| 2 | LTO | B'1 | R/W | <p>Timer Offset Value Configuration Request</p> <p>This bit issues requests for configuring gPTP timer offset configuration register i (GTOi).</p> <p>Only 1 can be written to the bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — The bit is set to 1 when the operating mode is not operation mode and CCC.GAC is 0b. — The bit is set to 0 when the value of gPTP timer offset configuration register i (GTOi) is loaded. <p>0: Setting completed 1: When written: Issue a configuration request. When read: Completion of settings is pending.</p> |
| 1, 0 | TCR[1:0] | B'00 | R/W | <p>Timer Control Request</p> <p>These bits issue requests for controlling the gPTP timer.</p> <p>The source selection (GCCR.TCSS) can be done by same write access.</p> <p>Writing to the bits is only possible when the current operating mode is operation mode, or if CSR.OPS is CONFIG and CCC.GAC is 1.</p> <p>Do not write to the bit when the gPTP timer clock select bit in the AVB-DMAC mode register is B'00. Write to these bits while GCCR.TCR is B'00 or B'10.</p> <p>The CPU should not write other values than B'11 to these bits if GCCR.TCR is B'10.</p> <p>The CPU should only use continuous capture (B'10) for AVTP presentation time value (GCCR.TCSS is B'10).</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — The bits are set to B'00 when the operating mode is not operation mode and CCC.GAC is 0b. — The bits are also set to B'00 on completion of the requested processing. <p>B'00: Timer control is not requested. B'01: gPTP/AVTP presentation time reset B'10: Continuous capture of AVTP to GCTt.CTV B'11: Captures the value set in the TCSS bit.</p> |

46.2.52 gPTP Maximum Transit Time Configuration Register (GMTT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GMTT register sets the maximum time for transitions of the gPTP timer.



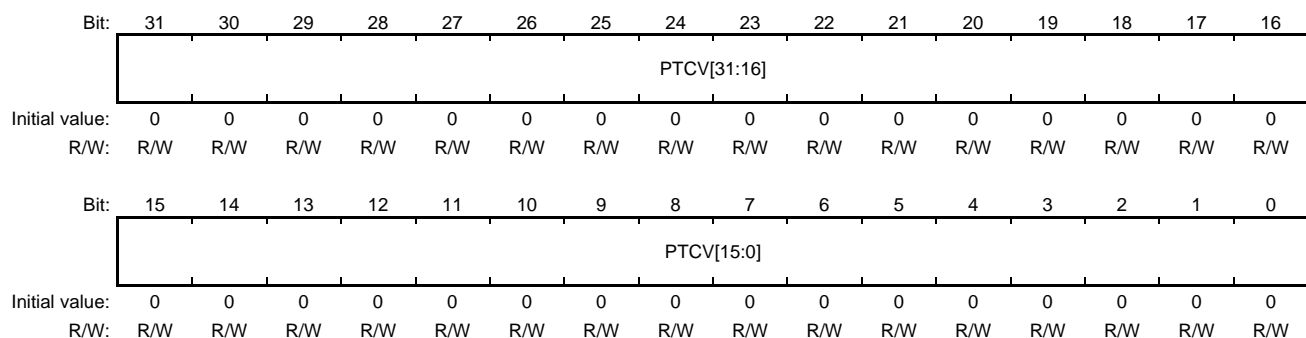
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 0 | MTTV[31:0] | H'0000_0000 | R/W | <p>Maximum Transit Time</p> <p>These bits set the maximum transition time for use in calculating AVTP presentation time.</p> <p>Write the desired setting to the bits, then issue the configuration request by setting the maximum transit time configuration request bit in the gPTP configuration control register (GCCR.LMTT) to 1.</p> |

Note: Do not write a value to these bits when the operating mode is operation mode or CCC.GAC is 1 and the maximum transit time configuration request bit (GCCR.LMTT) is 1.

46.2.53 gPTP Presentation Time Comparison Register (GPTC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GPTC register sets a value for comparison with presentation times in the gPTP timer.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 0 | PTCV[31:0] | H'0000_0000 | R/W | <p>Presentation Time Comparison Value</p> <p>These bits set a value for comparison with AVTP timer values to which a maximum transit time is not appended.</p> <p>Write the desired setting to the bits, then issue the configuration request by setting the presentation time comparison value configuration request bit in the gPTP configuration control register (GCCR.LPTC) to 1.</p> |

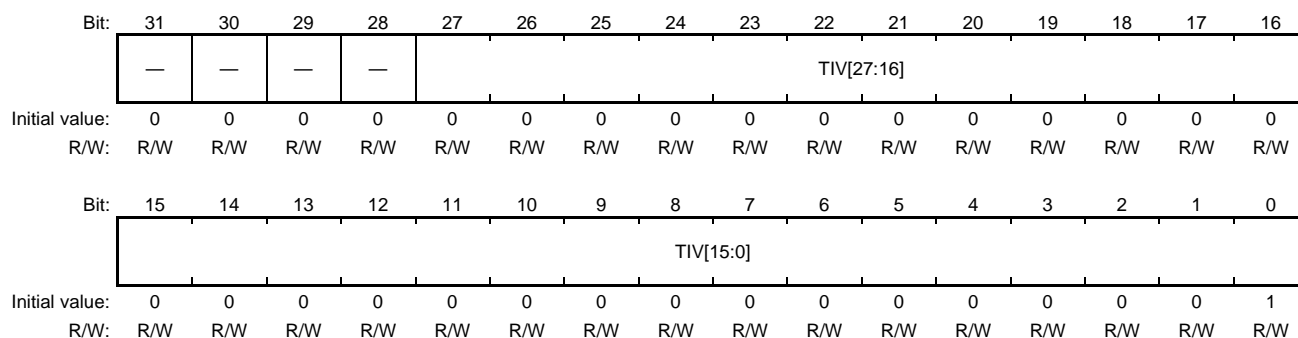
Note: Do not write a value to these bits when the presentation time comparison value configuration request bit (GCCR.LPTC) is 1.

The CPU should only write values in range of 0 to H'3FFF_FFFF to these bits when they are defining the period value of AVTP comparison unit (controlled by GCCR.SPC).

46.2.54 gPTP Timer Increment Configuration Register (GTI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GTI register sets the increment for the gPTP timer.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 28 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 27 to 0 | TIV[27:0] | H'000_0001 | R/W | gPTP Timer Increment Value When the gPTP clock select bits in the AVB-DMAC mode register (CCC.CSEL) are selecting a clock signal, these bits set the value by which the timer is incremented each time a cycle of that clock signal elapses. Write the desired setting to the bits, then issue the configuration request by setting the timer increment value configuration request bit in the gPTP configuration control register (GCCR.LTI) to 1. |

Note: Do not write a value to these bits when the operating mode is operation mode and CCC.GAC is 1 and the timer increment value configuration request bit (GCCR.LTI) is 1.

Do not write 0 to the bits.

Keep comparison value in the following range

$$x \leq \text{comparison_value} \leq 2^{32} - x$$

, where x is amount of nanoseconds of the configured increment value in GTI.TIV

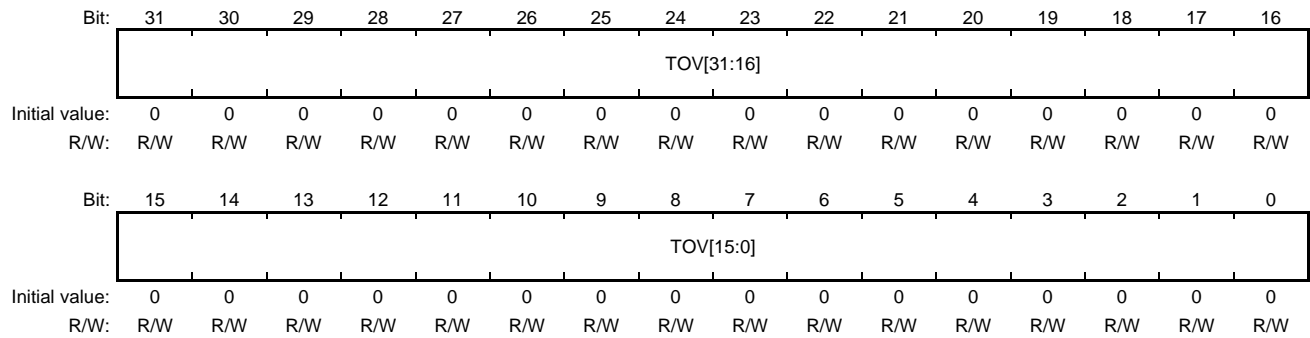
The CPU should not program increment values less than H'0010_0000

46.2.55 gPTP timer offset register i (GTOi) (i = 0 to 2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GTOi register sets an offset value for the gPTP timer.

The offset value is added to the combination of bits 0 to 31 in GTO0, 32 to 63 in GTO1, and 64 to 79 in GTO2, which together make up the gPTP timer.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | TOV[31:0] | H'0000_0000 | R/W | <p>Timer Offset Value</p> <p>This is an 80-bit value consisting of the settings in GTO0.TOV[31:0], GTO1.TOV[63:32], and GTO2.TOV[79:64], and is used to set an offset for adding to the value of the gPTP timer.</p> <p>Write the desired setting to the bits, then issue the configuration request by setting the timer offset value configuration request bit in the gPTP configuration control register (GCCR.LTO) to 1.</p> |

Note: Do not write a value to these bits when the operating mode is operation mode and CCC.GAC is 1 and the timer offset value configuration request bit (GCCR.LTO) is 1. Write H'0000 to GTO2.TOV[95:80]. Set a value in the range from 0 to 10⁹-1 (H'0000_0000 to H'3B9A_C9FF) in GTOi.TOV[31:0].

46.2.56 gPTP Interrupt Control Register (GIC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GIC register is used to control gPTP-related interrupts.

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | ATCE 15 | ATCE 14 | ATCE 13 | ATCE 12 | ATCE 11 | ATCE 10 | ATCE9 | ATCE8 | ATCE7 | ATCE6 | ATCE5 | ATCE4 | ATCE3 | ATCE2 | ATCE1 | ATCE0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | PTME7 | PTME6 | PTME5 | PTME4 | PTME3 | PTME2 | PTME1 | PTME0 | PTOE | PTCE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 31 | ATCE15 | B'0 | R/W | Avtp Time Captured interrupt Enable 15 While this bit is 1 an interrupt will be generated when GIS.ATCFi15 is 1. [Changing condition] This bit is set to 0 when writing 1 to GID.ATCD15. This bit is set to 1 when writing 1 to GIE.ATCS15. 0: Disabled 1: Enabled |
| 30 | ATCE14 | B'0 | R/W | Avtp Time Captured interrupt Enable 14 While this bit is 1 an interrupt will be generated when GIS.ATCF14 is 1. [Changing condition] This bit is set to 0 when writing 1 to GID.ATCD14. This bit is set to 1 when writing 1 to GIE.ATCS14. 0: Disabled 1: Enabled |
| 29 | ATCE13 | B'0 | R/W | Avtp Time Captured interrupt Enable 13 While this bit is 1 an interrupt will be generated when GIS.ATCF13 is 1. [Changing condition] This bit is set to 0 when writing 1 to GID.ATCD13. This bit is set to 1 when writing 1 to GIE.ATCS13. 0: Disabled 1: Enabled |
| 28 | ATCE12 | B'0 | R/W | Avtp Time Captured interrupt Enable 12 While this bit is 1 an interrupt will be generated when GIS.ATCF12 is 1. [Changing condition] This bit is set to 0 when writing 1 to GID.ATCD12. This bit is set to 1 when writing 1 to GIE.ATCS12. 0: Disabled 1: Enabled |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 27 | ATCE11 | B'0 | R/W | <p>Avtp Time Captured interrupt Enable 11</p> <p>While this bit is 1 an interrupt will be generated when GIS.ATCF11 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.ATCD11.</p> <p>This bit is set to 1 when writing 1 to GIE.ATCS11.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 26 | ATCE10 | B'0 | R/W | <p>Avtp Time Captured interrupt Enable 10</p> <p>While this bit is 1 an interrupt will be generated when GIS.ATCF10 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.ATCD10.</p> <p>This bit is set to 1 when writing 1 to GIE.ATCS10.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 25 | ATCE9 | B'0 | R/W | <p>Avtp Time Captured interrupt Enable 9</p> <p>While this bit is 1 an interrupt will be generated when GIS.ATCF9 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.ATCD9.</p> <p>This bit is set to 1 when writing 1 to GIE.ATCS9.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 24 | ATCE8 | B'0 | R/W | <p>Avtp Time Captured interrupt Enable 8</p> <p>While this bit is 1 an interrupt will be generated when GIS.ATCF8 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.ATCD8.</p> <p>This bit is set to 1 when writing 1 to GIE.ATCS8.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 23 | ATCE7 | B'0 | R/W | <p>Avtp Time Captured interrupt Enable 7</p> <p>While this bit is 1 an interrupt will be generated when GIS.ATCF7 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.ATCD7.</p> <p>This bit is set to 1 when writing 1 to GIE.ATCS7.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 22 | ATCE6 | B'0 | R/W | <p>Avtp Time Captured interrupt Enable 6</p> <p>While this bit is 1 an interrupt will be generated when GIS.ATCF6 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.ATCD6.</p> <p>This bit is set to 1 when writing 1 to GIE.ATCS6.</p> <p>0: Disabled</p> <p>1: Enabled</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 21 | ATCE5 | B'0 | R/W | <p>Avtp Time Captured interrupt Enable 5</p> <p>While this bit is 1 an interrupt will be generated when GIS.ATCF5 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.ATCD5.</p> <p>This bit is set to 1 when writing 1 to GIE.ATCS5.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 20 | ATCE4 | B'0 | R/W | <p>Avtp Time Captured interrupt Enable 4</p> <p>While this bit is 1 an interrupt will be generated when GIS.ATCF4 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.ATCD4.</p> <p>This bit is set to 1 when writing 1 to GIE.ATCS4.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 19 | ATCE3 | B'0 | R/W | <p>Avtp Time Captured interrupt Enable 3</p> <p>While this bit is 1 an interrupt will be generated when GIS.ATCF3 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.ATCD3.</p> <p>This bit is set to 1 when writing 1 to GIE.ATCS3.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 18 | ATCE2 | B'0 | R/W | <p>Avtp Time Captured interrupt Enable 2</p> <p>While this bit is 1 an interrupt will be generated when GIS.ATCF2 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.ATCD2.</p> <p>This bit is set to 1 when writing 1 to GIE.ATCS2.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 17 | ATCE1 | B'0 | R/W | <p>Avtp Time Captured interrupt Enable 1</p> <p>While this bit is 1 an interrupt will be generated when GIS.ATCF1 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.ATCD1.</p> <p>This bit is set to 1 when writing 1 to GIE.ATCS1.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 16 | ATCE0 | B'0 | R/W | <p>Avtp Time Captured interrupt Enable 0</p> <p>While this bit is 1 an interrupt will be generated when GIS.ATCF0 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.ATCD0.</p> <p>This bit is set to 1 when writing 1 to GIE.ATCS0.</p> <p>0: Disabled</p> <p>1: Enabled</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 15 to 10 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 9 | PTME7 | B'0 | R/W | Presentation Time Matched interrupt Enable 7 While this bit is 1 an interrupt will be generated when GIS.PTMF7 is 1. [Changing condition] This bit is set to 0 when writing 1 to GID.PTMD7. This bit is set to 1 when writing 1 to GIE.PTMS7. 0: Disabled 1: Enabled |
| 8 | PTME6 | B'0 | R/W | Presentation Time Matched interrupt Enable 6 While this bit is 1 an interrupt will be generated when GIS.PTMF6 is 1. [Changing condition] This bit is set to 0 when writing 1 to GID.PTMD6. This bit is set to 1 when writing 1 to GIE.PTMS6. 0: Disabled 1: Enabled |
| 7 | PTME5 | B'0 | R/W | Presentation Time Matched interrupt Enable 5 While this bit is 1 an interrupt will be generated when GIS.PTMF5 is 1. [Changing condition] This bit is set to 0 when writing 1 to GID.PTMD5. This bit is set to 1 when writing 1 to GIE.PTMS5. 0: Disabled 1: Enabled |
| 6 | PTME4 | B'0 | R/W | Presentation Time Matched interrupt Enable 4 While this bit is 1 an interrupt will be generated when GIS.PTMF4 is 1. [Changing condition] This bit is set to 0 when writing 1 to GID.PTMD4. This bit is set to 1 when writing 1 to GIE.PTMS4. 0: Disabled 1: Enabled |
| 5 | PTME3 | B'0 | R/W | Presentation Time Matched interrupt Enable 3 While this bit is 1 an interrupt will be generated when GIS.PTMF3 is 1. [Changing condition] This bit is set to 0 when writing 1 to GID.PTMD3. This bit is set to 1 when writing 1 to GIE.PTMS3. 0: Disabled 1: Enabled |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 4 | PTME2 | B'0 | R/W | <p>Presentation Time Matched interrupt Enable 2</p> <p>While this bit is 1 an interrupt will be generated when GIS.PTMF2 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.PTMD2.</p> <p>This bit is set to 1 when writing 1 to GIE.PTMS2.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 3 | PTME1 | B'0 | R/W | <p>Presentation Time Matched interrupt Enable 1</p> <p>While this bit is 1 an interrupt will be generated when GIS.PTMF1 is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.PTMD1.</p> <p>This bit is set to 1 when writing 1 to GIE.PTMS1.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 2 | PTME0 | B'0 | R/W | <p>Presentation Time Match Interrupt Enable 0</p> <p>When this bit is 1, setting of the presentation time match interrupt flag in the gPTP interrupt status register (GIS.PTMF) to 1 leads to generation of that interrupt.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.PTMD0.</p> <p>This bit is set to 1 when writing 1 to GIE.PTMS0.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 1 | PTOE | B'0 | R/W | <p>Presentation Time Overrun interrupt Enable</p> <p>While this bit is 1 an interrupt will be generated when GIS.PTOF is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.PTOD.</p> <p>This bit is set to 1 when writing 1 to GIE.PTOS.</p> <p>0: Disabled</p> <p>1: Enabled</p> |
| 0 | PTCE | B'0 | R/W | <p>Presentation Time Captured interrupt Enable</p> <p>While this bit is 1 an interrupt will be generated when GIS.PTCF is 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when writing 1 to GID.PTCD.</p> <p>This bit is set to 1 when writing 1 to GIE.PTCS.</p> <p>0: Disabled</p> <p>1: Enabled</p> |

46.2.57 gPTP interrupt status register (GIS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GIS register indicates the state of the gPTP-related interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | ATCF 15 | ATCF 14 | ATCF 13 | ATCF 12 | ATCF 11 | ATCF 10 | ATCF9 | ATCF8 | ATCF7 | ATCF6 | ATCF5 | ATCF4 | ATCF3 | ATCF2 | ATCF1 | ATCF0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | PTMF7 | PTMF6 | PTMF5 | PTMF4 | PTMF3 | PTMF2 | PTMF1 | PTMF0 | PTOF | PTCF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 31 | ATCF15 | B'0 | R/W | <p>Avtp Time Capture Flag 15</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[16] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |
| 30 | ATCF14 | B'0 | R/W | <p>Avtp Time Capture Flag 14</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[15] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |
| 29 | ATCF13 | B'0 | R/W | <p>Avtp Time Capture Flag 13</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[14] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 28 | ATCF12 | B'0 | R/W | <p>Avtp Time Capture Flag 12</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[13] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |
| 27 | ATCF11 | B'0 | R/W | <p>Avtp Time Capture Flag 11</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[12] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |
| 26 | ATCF10 | B'0 | R/W | <p>Avtp Time Capture Flag 10</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[11] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |
| 25 | ATCF9 | B'0 | R/W | <p>Avtp Time Capture Flag 9</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[10] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 24 | ATCF8 | B'0 | R/W | <p>Avtp Time Capture Flag 8</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[9] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |
| 23 | ATCF7 | B'0 | R/W | <p>Avtp Time Capture Flag 7</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[8] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |
| 22 | ATCF6 | B'0 | R/W | <p>Avtp Time Capture Flag 6</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[7] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |
| 21 | ATCF5 | B'0 | R/W | <p>Avtp Time Capture Flag 5</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[6] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 20 | ATCF4 | B'0 | R/W | <p>Avtp Time Capture Flag 4</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[5] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |
| 19 | ATCF3 | B'0 | R/W | <p>Avtp Time Capture Flag 3</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[4] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |
| 18 | ATCF2 | B'0 | R/W | <p>Avtp Time Capture Flag 2</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[3] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |
| 17 | ATCF1 | B'0 | R/W | <p>Avtp Time Capture Flag 1</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[2] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 16 | ATCF0 | B'0 | R/W | <p>Avtp Time Capture Flag 0</p> <p>This bit indicates that the AVTP time has been captured based on external capture event (rising edge of avb_pt_capture[1] signal). The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: AVTP time captured</p> |
| 15 to 10 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 9 | PTMF7 | B'0 | R/W | <p>Presentation Time Matched Flag 7</p> <p>This bit indicates that the AVTP timer value has matched the configured comparison value of AVTP comparator 7. The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when the AVTP timer value reaches or exceeds the comparison value of AVTP comparator 7.</p> <p>0: No interrupt pending 1: Presentation time exceeded</p> |
| 8 | PTMF6 | B'0 | R/W | <p>Presentation Time Matched Flag 6</p> <p>This bit indicates that the AVTP timer value has matched the configured comparison value of AVTP comparator 6. The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when the AVTP timer value reaches or exceeds the comparison value of AVTP comparator 6.</p> <p>0: No interrupt pending 1: Presentation time exceeded</p> |
| 7 | PTMF5 | B'0 | R/W | <p>Presentation Time Matched Flag 5</p> <p>This bit indicates that the AVTP timer value has matched the configured comparison value of AVTP comparator 5. The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when the AVTP timer value reaches or exceeds the comparison value of AVTP comparator 5.</p> <p>0: No interrupt pending 1: Presentation time exceeded</p> |

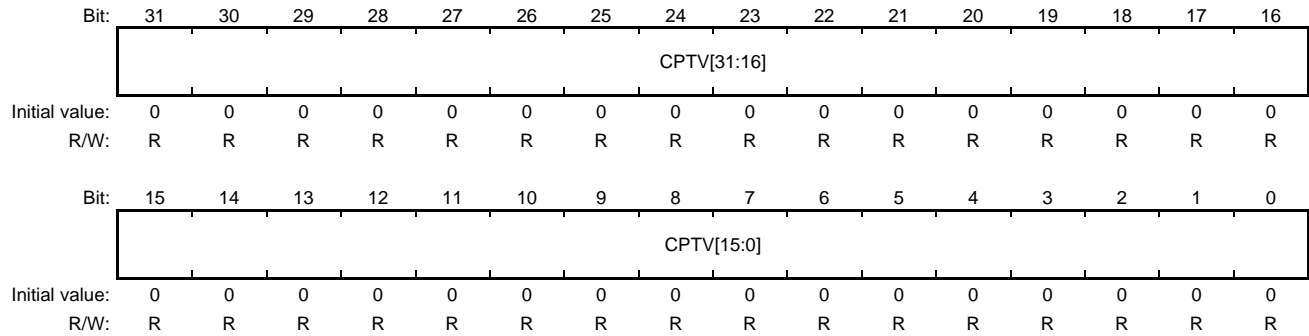
| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 6 | PTMF4 | B'0 | R/W | <p>Presentation Time Matched Flag 4</p> <p>This bit indicates that the AVTP timer value has matched the configured comparison value of AVTP comparator 4. The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when the AVTP timer value reaches or exceeds the comparison value of AVTP comparator 4.</p> <p>0: No interrupt pending 1: Presentation time exceeded</p> |
| 5 | PTMF3 | B'0 | R/W | <p>Presentation Time Matched Flag 3</p> <p>This bit indicates that the AVTP timer value has matched the configured comparison value of AVTP comparator 3. The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when the AVTP timer value reaches or exceeds the comparison value of AVTP comparator 3.</p> <p>0: No interrupt pending 1: Presentation time exceeded</p> |
| 4 | PTMF2 | B'0 | R/W | <p>Presentation Time Matched Flag 2</p> <p>This bit indicates that the AVTP timer value has matched the configured comparison value of AVTP comparator 2. The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when the AVTP timer value reaches or exceeds the comparison value of AVTP comparator 2.</p> <p>0: No interrupt pending 1: Presentation time exceeded</p> |
| 3 | PTMF1 | B'0 | R/W | <p>Presentation Time Matched Flag 1</p> <p>This bit indicates that the AVTP timer value has matched the configured comparison value of AVTP comparator 1. The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when the AVTP timer value reaches or exceeds the comparison value of AVTP comparator 1.</p> <p>0: No interrupt pending 1: Presentation time exceeded</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 2 | PTMF0 | B'0 | R/W | <p>Presentation Time Match Interrupt Flag 0</p> <p>This bit indicates that the value of the AVTP timer exceeds the value of the gPTP presentation time comparison register (GPTC). Only 0 can be written to the bit.</p> <p>[Conditions for Changing]</p> <ul style="list-style-type: none"> — The bit is set to 0 when the operating mode is not operation mode. — The bit is set to 1 when the AVTP timer value is greater than or equal to the value of the gPTP presentation time comparison register (GPTC). <p>0: The interrupt is not pending. 1: The interrupt is pending.</p> |
| 1 | PTOF | B'0 | R/W | <p>Presentation Time Overrun Flag</p> <p>This bit indicates that an external capture event (rising edge of avb_pt_capture[0] signal) has occurred before the last captured value was processed by CPU.</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger while GIS.PTCF is 1.</p> <p>0: No interrupt pending 1: Presentation time overrun</p> |
| 0 | PTCF | B'0 | R/W | <p>Presentation Time Capture Flag</p> <p>This bit indicates that the AVTP presentation time has been captured based on external capture event (rising edge of avb_pt_capture[0] signal).</p> <p>The CPU can only write 0 to this bit.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 1 when EthernetAVB-IF gets an external capture trigger.</p> <p>0: No interrupt pending 1: Presentation time captured</p> |

46.2.58 gPTP Captured Presentation Time register (GCPT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GCPT registers that captured presentation the gPTP timer value.

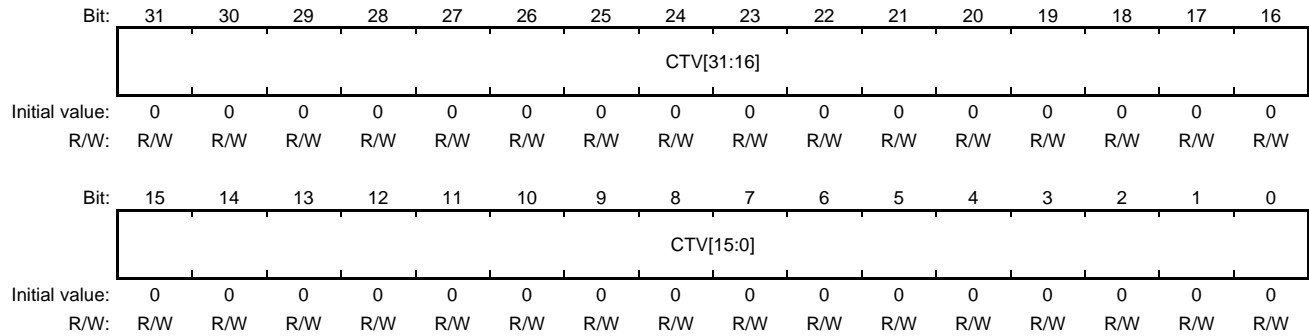


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 to 0 | CPTV[31:0] | H'0000_0000 | R | <p>Captured Presentation Time Value</p> <p>These bits represents the captured AVTP presentation time (with Max Transit Time added) due to HW trigger.</p> <p>The CPU should not read these bits when GIS.PTCF is 0.</p> <p>[Changing condition]</p> <p>These bits are updated with the current AVTP presentation time when an external capture trigger (avb_pt_capture[0]) occurs and GIS.PTCF is 0.</p> <p><i>Note for verification: These bits are set to 0 when leaving OPERATION mode and CCC.GAC is 0.</i></p> |

46.2.59 gPTP Timer Capture Register i (GCTi) (i = 0 to 2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GCTi registers form an 80-bit register that captures the gPTP timer value.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 0 | CTV[31:0] | H'0000_0000 | R/W | <p>gPTP Timer Capture Value</p> <p>These 80 bits consist of GCT0.CTV[31:0], GCT1.CTV[63:32 and GCT2.CTV[79:64], which together indicate captured timer values.</p> <p>When B'00 (value of the gPTP timer) or B'01 (adjusted gPTP timer value) is selected by the timer capture source select bits in the gPTP configuration control register, the corresponding 80-bit values are stored in these bits.</p> <p>When B'10 (AVTP presentation time) is selected by the timer capture source select bit, the corresponding 32-bit values are stored in these bits.</p> <p>In case of continuous update (GCCR.TCR is B'10) CPU should only use 32 bit read access to get consistent value.</p> <p>Actual writing of the timer value specified by the timer capture source select bits (GCCR.TCSS) proceeds when B'11 (timer capture request) is written to the timer control request bits in the gPTP configuration control register (GCCR.TCR).</p> <p>Do not read the value while the value of the timer control request bits (GCCR.TCR) is B'11, because this indicates that storage is still in progress.</p> <p><i>Note for verification: These bits are set to 0 when leaving OPERATION mode and CCC.GAC is 0.</i></p> |

46.2.60 gPTP Status Register (GSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GSR register indicates the state of the gPTP.

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----|-----|-----|------------|------|------|------|------------|------|------|------------|------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | AFU3 | AFU2 | AFU1 | AFU0 | AFFL3[3:0] | | | AFFL2[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | AFFL1[3:0] | | | | AFFL0[3:0] | | | | PCM7 | PCM6 | PCM5 | PCM4 | PCM3 | PCM2 | PCM1 | PCM0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 28 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 27 | AFU3 | B'0 | R | Avtp Fifo Update 3 This bit indicates an ongoing update of the AVTP-FIFO of compare unit 3. When this bit is 0, the AVTP-FIFO is able to process adding new values. [Changing condition] This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0. This bit is set to 0 when AVTP compare value is added to the AVTP-FIFO. This bit is set to 1 when CPU writes a new value to GPTF3.PTFV and GSR.AFFL3 is not 15. 0: No update of AVTP FIFO pending 1: Update of AVTP FIFO with value from GPTF3.PTFV is ongoing |
| 26 | AFU2 | B'0 | R | Avtp Fifo Update 2 This bit indicates an ongoing update of the AVTP-FIFO of compare unit 2. When this bit is 0, the AVTP-FIFO is able to process adding new values. [Changing condition] This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0. This bit is set to 0 when AVTP compare value is added to the AVTP-FIFO. This bit is set to 1 when CPU writes a new value to GPTF2.PTFV and GSR.AFFL2 is not 15. 0: No update of AVTP FIFO pending 1: Update of AVTP FIFO with value from GPTF2.PTFV is ongoing |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 25 | AFU1 | B'0 | R | <p>Avtp Fifo Update 1</p> <p>This bit indicates an ongoing update of the AVTP-FIFO of compare unit 1. When this bit is 0, the AVTP-FIFO is able to process adding new values.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 0 when AVTP compare value is added to the AVTP-FIFO.</p> <p>This bit is set to 1 when CPU writes a new value to GPTF1.PTFV and GSR.AFFL1 is not 15.</p> <p>0: No update of AVTP FIFO pending 1: Update of AVTP FIFO with value from GPTF1.PTFV is ongoing</p> |
| 24 | AFU0 | B'0 | R | <p>Avtp Fifo Update 0</p> <p>This bit indicates an ongoing update of the AVTP-FIFO of compare unit 0. When this bit is 0, the AVTP-FIFO is able to process adding new values.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 0 when AVTP compare value is added to the AVTP-FIFO.</p> <p>This bit is set to 1 when CPU writes a new value to GPTF0.PTFV and GSR.AFFL0 is not 15.</p> <p>0: No update of AVTP FIFO pending 1: Update of AVTP FIFO with value from GPTF0.PTFV is ongoing</p> |
| 23 to 20 | AFFL3[3:0] | H'0 | R | <p>FIFO fill level of AVTP compare unit 3</p> <p>These bits indicate the number of AVTP timestamps pending in AVTP-FIFO of compare unit 3. The timestamps are provided by CPU and taken by compare unit when no comparison on AVTP comparator 3 is active.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This value is incremented when a new value is added to AVTP-FIFO (GSR.AFU3 changes to 0).</p> <p>This value is decremented when a timestamp value from FIFO has been loaded to the AVTP comparator.</p> |
| 19 to 16 | AFFL2[3:0] | H'0 | R | <p>FIFO fill level of AVTP compare unit 2</p> <p>These bits indicate the number of AVTP timestamps pending in AVTP-FIFO of compare unit 2. The timestamps are provided by CPU and taken by compare unit when no comparison on AVTP comparator 2 is active.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This value is incremented when a new value is added to AVTP-FIFO (GSR.AFU2 changes to 0).</p> <p>This value is decremented when a timestamp value from FIFO has been loaded to the AVTP comparator.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 15 to 12 | AFFL1[3:0] | H'0 | R | <p>FIFO fill level of AVTP compare unit 1</p> <p>These bits indicate the number of AVTP timestamps pending in AVTP-FIFO of compare unit 1. The timestamps are provided by CPU and taken by compare unit when no comparison on AVTP comparator 1 is active.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This value is incremented when a new value is added to AVTP-FIFO (GSR.AFU1 changes to 0).</p> <p>This value is decremented when a timestamp value from FIFO has been loaded to the AVTP comparator.</p> |
| 11 to 8 | AFFL0[3:0] | H'0 | R | <p>FIFO fill level of AVTP compare unit 0</p> <p>These bits indicate the number of AVTP timestamps pending in AVTP-FIFO of compare unit 0. The timestamps are provided by CPU and taken by compare unit when no comparison on AVTP comparator 0 is active.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This value is incremented when a new value is added to AVTP-FIFO (GSR.AFU0 changes to 0).</p> <p>This value is decremented when a timestamp value from FIFO has been loaded to the AVTP comparator.</p> |
| 7 | PCM7 | B'0 | R | <p>Periodic Comparison Mode 7</p> <p>This bit indicates if single shot or periodic comparison mode is active on AVTP comparator unit 7.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 0 when the last match has happened after CPU has requested entering single shot mode.</p> <p>This bit is set to 1 when loading a period value for the AVTP comparator.</p> <p>0: Single shot comparison 1: Periodic comparison</p> |
| 6 | PCM6 | B'0 | R | <p>Periodic Comparison Mode 6</p> <p>This bit indicates if single shot or periodic comparison mode is active on AVTP comparator unit 6.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 0 when the last match has happened after CPU has requested entering single shot mode.</p> <p>This bit is set to 1 when loading a period value for the AVTP comparator.</p> <p>0: Single shot comparison 1: Periodic comparison</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 5 | PCM5 | B'0 | R | <p>Periodic Comparison Mode 5</p> <p>This bit indicates if single shot or periodic comparison mode is active on AVTP comparator unit 5.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 0 when the last match has happened after CPU has requested entering single shot mode.</p> <p>This bit is set to 1 when loading a period value for the AVTP comparator.</p> <p>0: Single shot comparison 1: Periodic comparison</p> |
| 4 | PCM4 | B'0 | R | <p>Periodic Comparison Mode 4</p> <p>This bit indicates if single shot or periodic comparison mode is active on AVTP comparator unit 4.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 0 when the last match has happened after CPU has requested entering single shot mode.</p> <p>This bit is set to 1 when loading a period value for the AVTP comparator.</p> <p>0: Single shot comparison 1: Periodic comparison</p> |
| 3 | PCM3 | B'0 | R | <p>Periodic Comparison Mode 3</p> <p>This bit indicates if single shot or periodic comparison mode is active on AVTP comparator unit 3.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 0 when the last match has happened after CPU has requested entering single shot mode.</p> <p>This bit is set to 1 when loading a period value for the AVTP comparator.</p> <p>0: Single shot comparison 1: Periodic comparison</p> |
| 2 | PCM2 | B'0 | R | <p>Periodic Comparison Mode 2</p> <p>This bit indicates if single shot or periodic comparison mode is active on AVTP comparator unit 2.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 0 when the last match has happened after CPU has requested entering single shot mode.</p> <p>This bit is set to 1 when loading a period value for the AVTP comparator.</p> <p>0: Single shot comparison 1: Periodic comparison</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 1 | PCM1 | B'0 | R | <p>Periodic Comparison Mode 1</p> <p>This bit indicates if single shot or periodic comparison mode is active on AVTP comparator unit 1.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 0 when the last match has happened after CPU has requested entering single shot mode.</p> <p>This bit is set to 1 when loading a period value for the AVTP comparator.</p> <p>0: Single shot comparison 1: Periodic comparison</p> |
| 0 | PCM0 | B'0 | R | <p>Periodic Comparison Mode 0</p> <p>This bit indicates if single shot or periodic comparison mode is active on AVTP comparator unit 0.</p> <p>[Changing condition]</p> <p>This bit is set to 0 when leaving OPERATION mode and CCC.GAC is 0.</p> <p>This bit is set to 0 when the last match has happened after CPU has requested entering single shot mode.</p> <p>This bit is set to 1 when loading a period value for the AVTP comparator.</p> <p>0: Single shot comparison 1: Periodic comparison</p> |

46.2.61 gPTP Interrupt Enable register (GIE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GIE register is used to control the gPTP Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | ATCS 15 | ATCS 14 | ATCS 13 | ATCS 12 | ATCS 11 | ATCS 10 | ATCS9 | ATCS8 | ATCS7 | ATCS 6 | ATCS 5 | ATCS 4 | ATCS3 | ATCS 2 | ATCS 1 | ATCS 0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | PTMS 7 | PTMS 6 | PTMS 5 | PTMS 4 | PTMS 3 | PTMS 2 | PTMS 1 | PTMS 0 | PTOS | PTCS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 31 | ATCS15 | B'0 | R/W | Avtp Time Captured interrupt Set 15 This bit supports interrupt enable. It controls set of GIC.ATCE15. This bit is always read as 0. 0: No change of GIC.ATCE15 1: Set GIC.ATCE15 to 1 |
| 30 | ATCS14 | B'0 | R/W | Avtp Time Captured interrupt Set 14 This bit supports interrupt enable. It controls set of GIC.ATCE14. This bit is always read as 0. 0: No change of GIC.ATCE14 1: Set GIC.ATCE14 to 1 |
| 29 | ATCS13 | B'0 | R/W | Avtp Time Captured interrupt Set 13 This bit supports interrupt enable. It controls set of GIC.ATCE13. This bit is always read as 0. 0: No change of GIC.ATCE13 1: Set GIC.ATCE13 to 1 |
| 28 | ATCS12 | B'0 | R/W | Avtp Time Captured interrupt Set 12 This bit supports interrupt enable. It controls set of GIC.ATCE12. This bit is always read as 0. 0: No change of GIC.ATCE12 1: Set GIC.ATCE12 to 1 |
| 27 | ATCS11 | B'0 | R/W | Avtp Time Captured interrupt Set 11 This bit supports interrupt enable. It controls set of GIC.ATCE11. This bit is always read as . 0: No change of GIC.ATCE11 1: Set GIC.ATCE11 to 1 |
| 26 | ATCS10 | B'0 | R/W | Avtp Time Captured interrupt Set 10 This bit supports interrupt enable. It controls set of GIC.ATCE10. This bit is always read as 0. 0: No change of GIC.ATCE10 1: Set GIC.ATCE10 to 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 25 | ATCS9 | B'0 | R/W | Avtp Time Captured interrupt Set 9 This bit supports interrupt enable. It controls set of GIC.ATCE9. This bit is always read as 0. 0: No change of GIC.ATCE9 1: Set GIC.ATCE9 to 1 |
| 24 | ATCS8 | B'0 | R/W | Avtp Time Captured interrupt Set 8 This bit supports interrupt enable. It controls set of GIC.ATCE8. This bit is always read as 0. 0: No change of GIC.ATCE8 1: Set GIC.ATCE8 to 1 |
| 23 | ATCS7 | B'0 | R/W | Avtp Time Captured interrupt Set 7 This bit supports interrupt enable. It controls set of GIC.ATCE7. This bit is always read as 0. 0: No change of GIC.ATCE7 1: Set GIC.ATCE7 to 1 |
| 22 | ATCS6 | B'0 | R/W | Avtp Time Captured interrupt Set 6 This bit supports interrupt enable. It controls set of GIC.ATCE6. This bit is always read as 0. 0: No change of GIC.ATCE6 1: Set GIC.ATCE6 to 1 |
| 21 | ATCS5 | B'0 | R/W | Avtp Time Captured interrupt Set 5 This bit supports interrupt enable. It controls set of GIC.ATCE5. This bit is always read as 0. 0: No change of GIC.ATCE5 1: Set GIC.ATCE5 to 1 |
| 20 | ATCS4 | B'0 | R/W | Avtp Time Captured interrupt Set 4 This bit supports interrupt enable. It controls set of GIC.ATCE4. This bit is always read as 0. 0: No change of GIC.ATCE4 1: Set GIC.ATCE4 to 1 |
| 19 | ATCS3 | B'0 | R/W | Avtp Time Captured interrupt Set 3 This bit supports interrupt enable. It controls set of GIC.ATCE3. This bit is always read as 0. 0: No change of GIC.ATCE3 1: Set GIC.ATCE3 to 1 |
| 18 | ATCS2 | B'0 | R/W | Avtp Time Captured interrupt Set 2 This bit supports interrupt enable. It controls set of GIC.ATCE2. This bit is always read as 0. 0: No change of GIC.ATCE2 1: Set GIC.ATCE2 to 1 |
| 17 | ATCS1 | B'0 | R/W | Avtp Time Captured interrupt Set 1 This bit supports interrupt enable. It controls set of GIC.ATCE1. This bit is always read as 0. 0: No change of GIC.ATCE1 1: Set GIC.ATCE1 to 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 16 | ATCS0 | B'0 | R/W | Avtp Time Captured interrupt Set 0 This bit supports interrupt enable. It controls set of GIC.ATCE0. This bit is always read as 0. 0: No change of GIC.ATCE0 1: Set GIC.ATCE0 to 1 |
| 15 to 10 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 9 | PTMS7 | B'0 | R/W | Presentation Time Matched interrupt Set 7 This bit supports interrupt enable. It controls set of GIC.PTME7. This bit is always read as 0. 0: No change of GIC.PTME7 1: Set GIC.PTME7 to 1 |
| 8 | PTMS6 | B'0 | R/W | Presentation Time Matched interrupt Set 6 This bit supports interrupt enable. It controls set of GIC.PTME6. This bit is always read as 0. 0: No change of GIC.PTME6 1: Set GIC.PTME6 to 1 |
| 7 | PTMS5 | B'0 | R/W | Presentation Time Matched interrupt Set 5 This bit supports interrupt enable. It controls set of GIC.PTME5. This bit is always read as 0. 0: No change of GIC.PTME5 1: Set GIC.PTME5 to 1 |
| 6 | PTMS4 | B'0 | R/W | Presentation Time Matched interrupt Set 4 This bit supports interrupt enable. It controls set of GIC.PTME4. This bit is always read as 0. 0: No change of GIC.PTME4 1: Set GIC.PTME4 to 1 |
| 5 | PTMS3 | B'0 | R/W | Presentation Time Matched interrupt Set 3 This bit supports interrupt enable. It controls set of GIC.PTME3. This bit is always read as 0. 0: No change of GIC.PTME3 1: Set GIC.PTME3 to 1 |
| 4 | PTMS2 | B'0 | R/W | Presentation Time Matched interrupt Set 2 This bit supports interrupt enable. It controls set of GIC.PTME2. This bit is always read as 0. 0: No change of GIC.PTME2 1: Set GIC.PTME2 to 1 |
| 3 | PTMS1 | B'0 | R/W | Presentation Time Matched interrupt Set 1 This bit supports interrupt enable. It controls set of GIC.PTME1. This bit is always read as 0. 0: No change of GIC.PTME1 1: Set GIC.PTME1 to 1 |
| 2 | PTMS0 | B'0 | R/W | Presentation Time Matched interrupt Set 0 This bit supports interrupt enable. It controls set of GIC.PTME0. This bit is always read as 0. 0: No change of GIC.PTME0 1: Set GIC.PTME0 to 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|------------|-----------------|----------------------|------------|---|
| 1 | PTOS | B'0 | R/W | Presentation Time Overrun interrupt Set This bit supports interrupt enable. It controls set of GIC.PTOE. This bit is always read as 0. 0: No change of GIC.PTOE 1: Set GIC.PTOE to 1 |
| 0 | PTCS | B'0 | R/W | Presentation Time Captured interrupt Set This bit supports interrupt enable. It controls set of GIC.PTCE. This bit is always read as 0. 0: No change of GIC.PTCE 1: Set GIC.PTCE to 1 |

46.2.62 gPTP Interrupt Disable register (GID)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GID register is used to control the gPTP Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | ATCD 15 | ATCD 14 | ATCD 13 | ATCD 12 | ATCD 11 | ATCD 10 | ATCD9 | ATCD8 | ATCD7 | ATCD 6 | ATCD 5 | ATCD 4 | ATCD3 | ATCD 2 | ATCD 1 | ATCD 0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | PTMD 7 | PTMD 6 | PTMD 5 | PTMD 4 | PTMD 3 | PTMD 2 | PTMD 1 | PTMD 0 | PTOD | PTCD |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 31 | ATCD15 | B'0 | R/W | Avtp Time Captured interrupt Disable 15 This bit supports interrupt enable. It controls set of GIC.ATCE15. This bit is always read as 0. 0: No change of GIC.ATCE15 1: Set GIC.ATCE15 to 0 |
| 30 | ATCD14 | B'0 | R/W | Avtp Time Captured interrupt Disable 14 This bit supports interrupt enable. It controls set of GIC.ATCE14. This bit is always read as 0. 0: No change of GIC.ATCE14 1: Set GIC.ATCE14 to 0 |
| 29 | ATCD13 | B'0 | R/W | Avtp Time Captured interrupt Disable 13 This bit supports interrupt enable. It controls set of GIC.ATCE13. This bit is always read as 0. 0: No change of GIC.ATCE13 1: Set GIC.ATCE13 to 0 |
| 28 | ATCD12 | B'0 | R/W | Avtp Time Captured interrupt Disable 12 This bit supports interrupt enable. It controls set of GIC.ATCE12. This bit is always read as 0. 0: No change of GIC.ATCE12 1: Set GIC.ATCE12 to 0 |
| 27 | ATCD11 | B'0 | R/W | Avtp Time Captured interrupt Disable 11 This bit supports interrupt enable. It controls set of GIC.ATCE11. This bit is always read as 0. 0: No change of GIC.ATCE11 1: Set GIC.ATCE11 to 0 |
| 26 | ATCD10 | B'0 | R/W | Avtp Time Captured interrupt Disable 10 This bit supports interrupt enable. It controls set of GIC.ATCE10. This bit is always read as 0. 0: No change of GIC.ATCE10 1: Set GIC.ATCE10 to 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 25 | ATCD9 | B'0 | R/W | Avtp Time Captured interrupt Disable 9 This bit supports interrupt enable. It controls set of GIC.ATCE9. This bit is always read as 0. 0: No change of GIC.ATCE9 1: Set GIC.ATCE9 to 0 |
| 24 | ATCD8 | B'0 | R/W | Avtp Time Captured interrupt Disable 8 This bit supports interrupt enable. It controls set of GIC.ATCE8. This bit is always read as 0. 0: No change of GIC.ATCE8 1: Set GIC.ATCE8 to 0 |
| 23 | ATCD7 | B'0 | R/W | Avtp Time Captured interrupt Disable 7 This bit supports interrupt enable. It controls set of GIC.ATCE7. This bit is always read as 0. 0: No change of GIC.ATCE7 1: Set GIC.ATCE7 to 0 |
| 22 | ATCD6 | B'0 | R/W | Avtp Time Captured interrupt Disable 6 This bit supports interrupt enable. It controls set of GIC.ATCE6. This bit is always read as 0. 0: No change of GIC.ATCE6 1: Set GIC.ATCE6 to 0 |
| 21 | ATCD5 | B'0 | R/W | Avtp Time Captured interrupt Disable 5 This bit supports interrupt enable. It controls set of GIC.ATCE5. This bit is always read as 0. 0: No change of GIC.ATCE5 1: Set GIC.ATCE5 to 0 |
| 20 | ATCD4 | B'0 | R/W | Avtp Time Captured interrupt Disable 4 This bit supports interrupt enable. It controls set of GIC.ATCE4. This bit is always read as 0. 0: No change of GIC.ATCE4 1: Set GIC.ATCE4 to 0 |
| 19 | ATCD3 | B'0 | R/W | Avtp Time Captured interrupt Disable 3 This bit supports interrupt enable. It controls set of GIC.ATCE3. This bit is always read as 0. 0: No change of GIC.ATCE3 1: Set GIC.ATCE3 to 0 |
| 18 | ATCD2 | B'0 | R/W | Avtp Time Captured interrupt Disable 2 This bit supports interrupt enable. It controls set of GIC.ATCE2. This bit is always read as 0. 0: No change of GIC.ATCE2 1: Set GIC.ATCE2 to 0 |
| 17 | ATCD1 | B'0 | R/W | Avtp Time Captured interrupt Disable 1 This bit supports interrupt enable. It controls set of GIC.ATCE1. This bit is always read as 0. 0: No change of GIC.ATCE1 1: Set GIC.ATCE1 to 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 16 | ATCD0 | B'0 | R/W | Avtp Time Captured interrupt Disable 0 This bit supports interrupt enable. It controls set of GIC.ATCE0. This bit is always read as 0. 0: No change of GIC.ATCE0 1: Set GIC.ATCE0 to 0 |
| 15 to 10 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 9 | PTMD7 | B'0 | R/W | Presentation Time Matched interrupt Disable 7 This bit supports interrupt enable. It controls set of GIC.PTME7. This bit is always read as 0. 0: No change of GIC.PTME7 1: Set GIC.PTME7 to 0 |
| 8 | PTMD6 | B'0 | R/W | Presentation Time Matched interrupt Disable 6 This bit supports interrupt enable. It controls set of GIC.PTME6. This bit is always read as 0. 0: No change of GIC.PTME6 1: Set GIC.PTME6 to 0 |
| 7 | PTMD5 | B'0 | R/W | Presentation Time Matched interrupt Disable 5 This bit supports interrupt enable. It controls set of GIC.PTME5. This bit is always read as 0. 0: No change of GIC.PTME5 1: Set GIC.PTME5 to 0 |
| 6 | PTMD4 | B'0 | R/W | Presentation Time Matched interrupt Disable 4 This bit supports interrupt enable. It controls set of GIC.PTME4. This bit is always read as 0. 0: No change of GIC.PTME4 1: Set GIC.PTME4 to 0 |
| 5 | PTMD3 | B'0 | R/W | Presentation Time Matched interrupt Disable 3 This bit supports interrupt enable. It controls set of GIC.PTME3. This bit is always read as 0. 0: No change of GIC.PTME3 1: Set GIC.PTME3 to 0 |
| 4 | PTMD2 | B'0 | R/W | Presentation Time Matched interrupt Disable 2 This bit supports interrupt enable. It controls set of GIC.PTME2. This bit is always read as 0. 0: No change of GIC.PTME2 1: Set GIC.PTME2 to 0 |
| 3 | PTMD1 | B'0 | R/W | Presentation Time Matched interrupt Disable 1 This bit supports interrupt enable. It controls set of GIC.PTME1. This bit is always read as 0. 0: No change of GIC.PTME1 1: Set GIC.PTME1 to 0 |
| 2 | PTMD0 | B'0 | R/W | Presentation Time Matched interrupt Disable 0 This bit supports interrupt enable. It controls set of GIC.PTME0. This bit is always read as 0. 0: No change of GIC.PTME0 1: Set GIC.PTME0 to 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | PTOD | B'0 | R/W | Presentation Time Overrun interrupt Disable This bit supports interrupt enable. It controls set of GIC.PTOE. This bit is always read as 0. 0: No change of GIC.PTOE 1: Set GIC.PTOE to 0 |
| 0 | PTCD | B'0 | R/W | Presentation Time Captured interrupt Disable This bit supports interrupt enable. It controls set of GIC.PTCE. This bit is always read as 0. 0: No change of GIC.PTCE 1: Set GIC.PTCE to 0 |

46.2.63 gPTP Interrupt Line selection register (GIL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GIL register is used to control the gPTP Interrupt line.

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | ATCL 15 | ATCL 14 | ATCL 13 | ATCL 12 | ATCL 11 | ATCL 10 | ATCL9 | ATCL8 | ATCL7 | ATCL6 | ATCL5 | ATCL4 | ATCL3 | ATCL2 | ATCL1 | ATCL0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | PTML 7 | PTML 6 | PTML 5 | PTML 4 | PTML 3 | PTML 2 | PTML 1 | PTML 0 | PTOL | PTCL |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 31 | ATCL15 | B'0 | R/W | Avtp Time Captured interrupt Line select 15 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 30 | ATCL14 | B'0 | R/W | Avtp Time Captured interrupt Line select 14 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 29 | ATCL13 | B'0 | R/W | Avtp Time Captured interrupt Line select 13 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 28 | ATCL12 | B'0 | R/W | Avtp Time Captured interrupt Line select 12 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 27 | ATCL11 | B'0 | R/W | Avtp Time Captured interrupt Line select 11 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 26 | ATCL10 | B'0 | R/W | Avtp Time Captured interrupt Line select 10 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 25 | ATCL9 | B'0 | R/W | Avtp Time Captured interrupt Line select 9 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 24 | ATCL8 | B'0 | R/W | Avtp Time Captured interrupt Line select 8 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 23 | ATCL7 | B'0 | R/W | Avtp Time Captured interrupt Line select 7 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 22 | ATCL6 | B'0 | R/W | Avtp Time Captured interrupt Line select 6 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 21 | ATCL5 | B'0 | R/W | Avtp Time Captured interrupt Line select 5 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 20 | ATCL4 | B'0 | R/W | Avtp Time Captured interrupt Line select 4 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 19 | ATCL3 | B'0 | R/W | Avtp Time Captured interrupt Line select 3 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 18 | ATCL2 | B'0 | R/W | Avtp Time Captured interrupt Line select 2 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 17 | ATCL1 | B'0 | R/W | Avtp Time Captured interrupt Line select 1 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 16 | ATCL0 | B'0 | R/W | Avtp Time Captured interrupt Line select 0 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 15 to 10 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 9 | PTML7 | B'0 | R/W | Presentation Time Matched interrupt Line select 7 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 8 | PTML6 | B'0 | R/W | Presentation Time Matched interrupt Line select 6 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 7 | PTML5 | B'0 | R/W | Presentation Time Matched interrupt Line select 5 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 6 | PTML4 | B'0 | R/W | Presentation Time Matched interrupt Line select 4 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 5 | PTML3 | B'0 | R/W | Presentation Time Matched interrupt Line select 3 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 4 | PTML2 | B'0 | R/W | Presentation Time Matched interrupt Line select 2 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 3 | PTML1 | B'0 | R/W | Presentation Time Matched interrupt Line select 1 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 2 | PTML0 | B'0 | R/W | Presentation Time Matched interrupt Line select 0 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | PTOL | B'0 | R/W | Presentation Time Overrun interrupt Line select This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 0 | PTCL | B'0 | R/W | Presentation Time Captured interrupt Line select This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |

46.2.64 gPTP Avtp Capture Prescaler register (GACP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GACP register is used to control the gPTP Capture Prescaler.

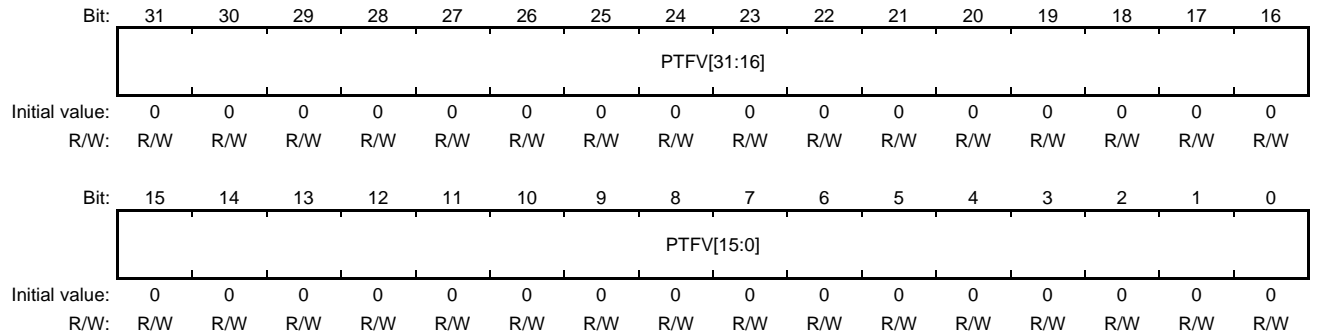
| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----------|-----|-----|-----|-----------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | ACPN[3:0] | | | | ACPV[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 12 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 11 to 8 | ACPN[3:0] | H'0 | R/W | Prescaler unit number to be configured Avtp Capture Prescaler unit Number These bits define the number <i>i</i> of AVTP capture unit where the prescaler value should be updated. These bits are always read as 0. The CPU can only write to these bits when CSR.OPS is OPERATION. The CPU can only write to these bits when CSR.OPS is CONFIG and CCC.GAC is 1. |
| 7 to 0 | ACPV[7:0] | H'00 | R/W | Prescaler configuration of AVTP capture unit <i>i</i> Avtp Capture Prescaler Value These bits define the prescaler configuration used by AVTP capture unit <i>i</i> . If the value is 0, on each rising edge of avb_pt_capture[<i>i</i> +1] the current AVTP timer value is captured. These bits are always read as 0. The CPU can only write to these bits when CSR.OPS is OPERATION. The CPU can only write to these bits when CSR.OPS is CONFIG and CCC.GAC is 1. |

46.2.65 gPTP Presentation Time FIFO register i (GPTFi) (i = 0 to 3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GPTFi register is used to control the gPTP FIFO.

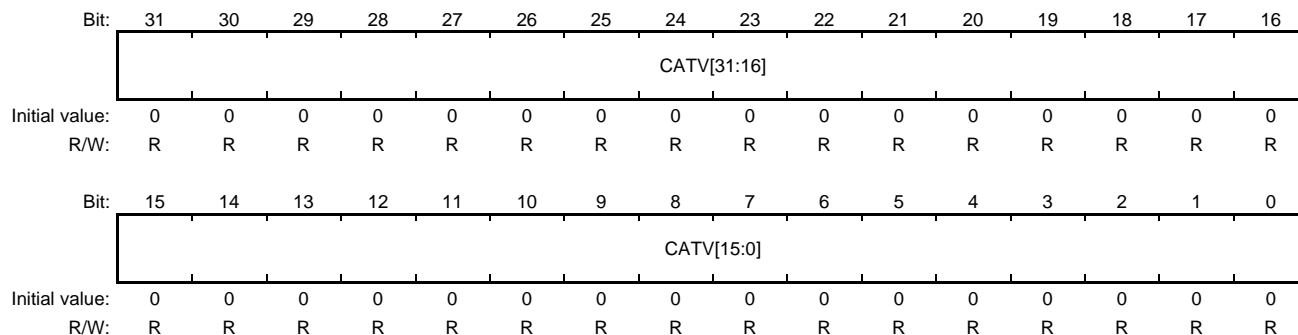


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 0 | PTFV[31:0] | H'0000_0000 | R/W | <p>Presentation Time Fifo Value</p> <p>These bits define a value to be compared with the AVTP timer value (without Max Transit Time added). When writing to this register the value is added to AVTP-FIFO of comparator unit i.</p> <p>These bits are always read as 0.</p> <p>The CPU cannot write to these bits if CSR.OPS is CONFIG and CCC.GAC is 0.</p> <p>The CPU should not write to these bits when GSR.PCMi is 1.</p> <p>The CPU should not write to these bits when GSR.AFUi is 1.</p> <p>The CPU should not write to these bits when GSR.AFFLi is 15.</p> |

46.2.66 gPTP Captured Avtp Time register i (GCATi) (i = 0 to 15)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GCATi register indicates the value of the gPTP captured time.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 0 | CATV[31:0] | H'0000_0000 | R | <p>Captured AVTP time value</p> <p>These bits represents the captured AVTP time (without Max Transit Time added) due to HW trigger.</p> <p>The CPU should not read these bits when GIS.ATCFi is 0.</p> <p>[Changing condition]</p> <p>These bits are updated with the current AVTP time when an external capture trigger (avb_pt_capture[i+1]) occurs and GIS.ATCFi is 0.</p> <p><i>Note for verification: These bits are set to 0 when leaving OPERATION mode and CCC.GAC is 0.</i></p> |

46.2.67 Descriptor Interrupt Line selection register (DIL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The DIL register is used to control the Descriptor Interrupt line.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DPL15 | DPL14 | DPL13 | DPL12 | DPL11 | DPL10 | DPL9 | DPL8 | DPL7 | DPL6 | DPL5 | DPL4 | DPL3 | DPL2 | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 | DPL15 | B'0 | R/W | Descriptor Processed interrupt Line select 15 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 14 | DPL14 | B'0 | R/W | Descriptor Processed interrupt Line select 14 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 13 | DPL13 | B'0 | R/W | Descriptor Processed interrupt Line select 13 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 12 | DPL12 | B'0 | R/W | Descriptor Processed interrupt Line select 12 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 11 | DPL11 | B'0 | R/W | Descriptor Processed interrupt Line select 11 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 10 | DPL10 | B'0 | R/W | Descriptor Processed interrupt Line select 10 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 9 | DPL9 | B'0 | R/W | Descriptor Processed interrupt Line select 9 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 8 | DPL8 | B'0 | R/W | Descriptor Processed interrupt Line select 8 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 7 | DPL7 | B'0 | R/W | Descriptor Processed interrupt Line select 7 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 6 | DPL6 | B'0 | R/W | Descriptor Processed interrupt Line select 6 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 5 | DPL5 | B'0 | R/W | Descriptor Processed interrupt Line select 5 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 4 | DPL4 | B'0 | R/W | Descriptor Processed interrupt Line select 4 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 3 | DPL3 | B'0 | R/W | Descriptor Processed interrupt Line select 3 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 2 | DPL2 | B'0 | R/W | Descriptor Processed interrupt Line select 2 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 1 to 0 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |

46.2.68 Error Interrupt Line selection register (EIL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The EIL register is used to control the Error Interrupt line.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|------|------|------|-------|-------|-------|-------|-----|-----|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | TBFL | MFFL | TFFL | CULL1 | CULL0 | CLLL1 | CLLL0 | SEL | QEL | MTEL | MREL |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 11 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 10 | TBFL | B'0 | R/W | Tx-Buffer Full interrupt Line select This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 9 | MFFL | B'0 | R/W | E-MAC status FIFO Full interrupt Line select This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 8 | TFFL | B'0 | R/W | Timestamp FIFO Full interrupt Line select This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 7 | CULL1 | B'0 | R/W | CBS Upper Limit reached interrupt Line select 1 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 6 | CULL0 | B'0 | R/W | CBS Upper Limit reached interrupt Line select 0 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 5 | CLLL1 | B'0 | R/W | CBS Lower Limit reached interrupt Line select 1 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 4 | CLLL0 | B'0 | R/W | CBS Lower Limit reached interrupt Line select 0 This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 3 | SEL | B'0 | R/W | Separation Error interrupt Line select This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 2 | QEL | B'0 | R/W | Queue Error interrupt Line select This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 1 | MTEL | B'0 | R/W | E-MAC Transmission Error interrupt Line select This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 0 | MREL | B'0 | R/W | E-MAC Reception Error interrupt Line select This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |

46.2.69 Transmission Interrupt Line selection register (TIL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The TIL register is used to control the Transmission Interrupt line.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|------|------|------|------|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | MFWL | MFUL | TFWL | TFUL | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 12 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 11 | MFWL | B'0 | R/W | E-MAC status FIFO Warning interrupt Line select This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 10 | MFUL | B'0 | R/W | E-MAC status FIFO Updated interrupt Line select This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 9 | TFWL | B'0 | R/W | Timestamp FIFO Warning interrupt Line select This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 8 | TFUL | B'0 | R/W | Timestamp FIFO Updated interrupt Line select This bit selects the interrupt line used for this notification. It is recommended to disable interrupt before change the line. 0: Interrupt line A used for this notification 1: Interrupt line B used for this notification |
| 7 to 0 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |

46.2.70 Descriptor Interrupt Enable register (DIE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The DIE register is used to control the Descriptor Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DPS15 | DPS14 | DPS13 | DPS12 | DPS11 | DPS10 | DPS9 | DPS8 | DPS7 | DPS6 | DPS5 | DPS4 | DPS3 | DPS2 | DPS1 | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 | DPS15 | B'0 | R/W | Descriptor Processed interrupt Set 15 This bit supports interrupt enable. It controls set of DIC.DPE15. This bit is always read as 0. 0: No change of DIC.DPE15 1: Set DIC.DPE15 to 1 |
| 14 | DPS14 | B'0 | R/W | Descriptor Processed interrupt Set 14 This bit supports interrupt enable. It controls set of DIC.DPE14. This bit is always read as 0. 0: No change of DIC.DPE14 1: Set DIC.DPE14 to 1 |
| 13 | DPS13 | B'0 | R/W | Descriptor Processed interrupt Set 13 This bit supports interrupt enable. It controls set of DIC.DPE13. This bit is always read as 0. 0: No change of DIC.DPE13 1: Set DIC.DPE13 to 1 |
| 12 | DPS12 | B'0 | R/W | Descriptor Processed interrupt Set 12 This bit supports interrupt enable. It controls set of DIC.DPE12. This bit is always read as 0. 0: No change of DIC.DPE12 1: Set DIC.DPE12 to 1 |
| 11 | DPS11 | B'0 | R/W | Descriptor Processed interrupt Set 11 This bit supports interrupt enable. It controls set of DIC.DPE11. This bit is always read as 0. 0: No change of DIC.DPE11 1: Set DIC.DPE11 to 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 10 | DPS10 | B'0 | R/W | Descriptor Processed interrupt Set 10 This bit supports interrupt enable. It controls set of DIC.DPE10. This bit is always read as 0. 0: No change of DIC.DPE10 1: Set DIC.DPE10 to 1 |
| 9 | DPS9 | B'0 | R/W | Descriptor Processed interrupt Set 9 This bit supports interrupt enable. It controls set of DIC.DPE9. This bit is always read as 0. 0: No change of DIC.DPE9 1: Set DIC.DPE9 to 1 |
| 8 | DPS8 | B'0 | R/W | Descriptor Processed interrupt Set 8 This bit supports interrupt enable. It controls set of DIC.DPE8. This bit is always read as 0. 0: No change of DIC.DPE8 1: Set DIC.DPE8 to 1 |
| 7 | DPS7 | B'0 | R/W | Descriptor Processed interrupt Set 7 This bit supports interrupt enable. It controls set of DIC.DPE7. This bit is always read as 0. 0: No change of DIC.DPE7 1: Set DIC.DPE7 to 1 |
| 6 | DPS6 | B'0 | R/W | Descriptor Processed interrupt Set 6 This bit supports interrupt enable. It controls set of DIC.DPE6. This bit is always read as 0. 0: No change of DIC.DPE6 1: Set DIC.DPE6 to 1 |
| 5 | DPS5 | B'0 | R/W | Descriptor Processed interrupt Set 5 This bit supports interrupt enable. It controls set of DIC.DPE5. This bit is always read as 0. 0: No change of DIC.DPE5 1: Set DIC.DPE5 to 1 |
| 4 | DPS4 | B'0 | R/W | Descriptor Processed interrupt Set 4 This bit supports interrupt enable. It controls set of DIC.DPE4. This bit is always read as 0. 0: No change of DIC.DPE4 1: Set DIC.DPE4 to 1 |
| 3 | DPS3 | B'0 | R/W | Descriptor Processed interrupt Set 3 This bit supports interrupt enable. It controls set of DIC.DPE3. This bit is always read as 0. 0: No change of DIC.DPE3 1: Set DIC.DPE3 to 1 |
| 2 | DPS2 | B'0 | R/W | Descriptor Processed interrupt Set 2 This bit supports interrupt enable. It controls set of DIC.DPE2. This bit is always read as 0. 0: No change of DIC.DPE2 1: Set DIC.DPEi to 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | DPS1 | B'0 | R/W | Descriptor Processed interrupt Set 1 This bit supports interrupt enable. It controls set of DIC.DPE1. This bit is always read as 0. 0: No change of DIC.DPE1 1: Set DIC.DPE1 to 1 |
| 0 | — | B'0 | R/W | Reserved These bits are read as 0. The write value should be 0. |

46.2.71 Descriptor Interrupt Disable register (DID)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The DID register is used to control the Descriptor Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DPD15 | DPD14 | DPD13 | DPD12 | DPD11 | DPD10 | DPD9 | DPD8 | DPD7 | DPD6 | DPD5 | DPD4 | DPD3 | DPD2 | DPD1 | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 | DPD15 | B'0 | R/W | Descriptor Processed interrupt Disable 15 This bit supports interrupt enable. It controls set of DIC.DPE15. This bit is always read as 0. 0: No change of DIC.DPE15 1: Set DIC.DPE15 to 0 |
| 14 | DPD14 | B'0 | R/W | Descriptor Processed interrupt Disable 14 This bit supports interrupt enable. It controls set of DIC.DPE14. This bit is always read as 0. 0: No change of DIC.DPE14 1: Set DIC.DPE14 to 0 |
| 13 | DPD13 | B'0 | R/W | Descriptor Processed interrupt Disable 13 This bit supports interrupt enable. It controls set of DIC.DPE13. This bit is always read as 0. 0: No change of DIC.DPE13 1: Set DIC.DPE13 to 0 |
| 12 | DPD12 | B'0 | R/W | Descriptor Processed interrupt Disable 12 This bit supports interrupt enable. It controls set of DIC.DPE12. This bit is always read as 0. 0: No change of DIC.DPE12 1: Set DIC.DPE12 to 0 |
| 11 | DPD11 | B'0 | R/W | Descriptor Processed interrupt Disable 11 This bit supports interrupt enable. It controls set of DIC.DPE11. This bit is always read as 0. 0: No change of DIC.DPE11 1: Set DIC.DPE11 to 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 10 | DPD10 | B'0 | R/W | Descriptor Processed interrupt Disable 10 This bit supports interrupt enable. It controls set of DIC.DPE10. This bit is always read as 0. 0: No change of DIC.DPE10 1: Set DIC.DPE10 to 0 |
| 9 | DPD9 | B'0 | R/W | Descriptor Processed interrupt Disable 9 This bit supports interrupt enable. It controls set of DIC.DPE9. This bit is always read as 0. 0: No change of DIC.DPE9 1: Set DIC.DPE9 to 0 |
| 8 | DPD8 | B'0 | R/W | Descriptor Processed interrupt Disable 8 This bit supports interrupt enable. It controls set of DIC.DPE8. This bit is always read as 0. 0: No change of DIC.DPE8 1: Set DIC.DPE8 to 0 |
| 7 | DPD7 | B'0 | R/W | Descriptor Processed interrupt Disable 7 This bit supports interrupt enable. It controls set of DIC.DPE7. This bit is always read as 0. 0: No change of DIC.DPE7 1: Set DIC.DPE7 to 0 |
| 6 | DPD6 | B'0 | R/W | Descriptor Processed interrupt Disable 6 This bit supports interrupt enable. It controls set of DIC.DPE6. This bit is always read as 0. 0: No change of DIC.DPE6 1: Set DIC.DPE6 to 0 |
| 5 | DPD5 | B'0 | R/W | Descriptor Processed interrupt Disable 5 This bit supports interrupt enable. It controls set of DIC.DPE5. This bit is always read as 0. 0: No change of DIC.DPE5 1: Set DIC.DPE5 to 0 |
| 4 | DPD4 | B'0 | R/W | Descriptor Processed interrupt Disable 4 This bit supports interrupt enable. It controls set of DIC.DPE4. This bit is always read as 0. 0: No change of DIC.DPE4 1: Set DIC.DPE4 to 0 |
| 3 | DPD3 | B'0 | R/W | Descriptor Processed interrupt Disable 3 This bit supports interrupt enable. It controls set of DIC.DPE3. This bit is always read as 0. 0: No change of DIC.DPE3 1: Set DIC.DPE3 to 0 |
| 2 | DPD2 | B'0 | R/W | Descriptor Processed interrupt Disable 2 This bit supports interrupt enable. It controls set of DIC.DPE2. This bit is always read as 0. 0: No change of DIC.DPE2 1: Set DIC.DPE2 to 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | DPD1 | B'0 | R/W | Descriptor Processed interrupt Disable 1 This bit supports interrupt enable. It controls set of DIC.DPE1. This bit is always read as 0. 0: No change of DIC.DPE1 1: Set DIC.DPE1 to 0 |
| 0 | — | B'0 | R/W | Reserved These bits are read as 0. The write value should be 0. |

46.2.72 Error Interrupt Enable register (EIE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The EIE register is used to control the Error Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|------|------|------|-------|-------|-------|-------|-----|-----|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | TBFS | MFFS | TFFS | CULS1 | CULS0 | CLLS1 | CLLS0 | SES | QES | MTES | MRES |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 11 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 10 | TBFS | B'0 | R/W | Tx-Buffer Full interrupt Set This bit supports interrupt enable. It controls set of EIC.TBFE. This bit is always read as 0. 0: No change of EIC.TBFE 1: Set EIC.TBFE to 1 |
| 9 | MFFS | B'0 | R/W | E-MAC status FIFO Full interrupt Set This bit supports interrupt enable. It controls set of EIC.MFFE. This bit is always read as 0. 0: No change of EIC.MFFE 1: Set EIC.MFFE to 1 |
| 8 | TFFS | B'0 | R/W | Timestamp FIFO Full interrupt Set This bit supports interrupt enable. It controls set of EIC.TFFE. This bit is always read as 0. 0: No change of EIC.TFFE 1: Set EIC.TFFE to 1 |
| 7 | CULS1 | B'0 | R/W | CBS Upper Limit reached interrupt Set 1 This bit supports interrupt enable. It controls set of EIC.CULE1. This bit is always read as 0. 0: No change of EIC.CULE1 1: Set EIC.CULE1 to 1 |
| 6 | CULS0 | B'0 | R/W | CBS Upper Limit reached interrupt Set 0 This bit supports interrupt enable. It controls set of EIC.CULE0. This bit is always read as 0. 0: No change of EIC.CULE0 1: Set EIC.CULE0 to 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 5 | CLLS1 | B'0 | R/W | <p>CBS Lower Limit reached interrupt Set 1</p> <p>This bit supports interrupt enable. It controls set of EIC.CLLE1.</p> <p>This bit is always read as 0.</p> <p>0: No change of EIC.CLLE1</p> <p>1: Set EIC.CLLE1 to 1</p> |
| 4 | CLLS0 | B'0 | R/W | <p>CBS Lower Limit reached interrupt Set 0</p> <p>This bit supports interrupt enable. It controls set of EIC.CLLE0.</p> <p>This bit is always read as 0.</p> <p>0: No change of EIC.CLLE0</p> <p>1: Set EIC.CLLE0 to 1</p> |
| 3 | SES | B'0 | R/W | <p>Separation Error interrupt Set</p> <p>This bit supports interrupt enable. It controls set of EIC.SEE.</p> <p>This bit is always read as 0.</p> <p>0: No change of EIC.SEE</p> <p>1: Set EIC.QEE to 1</p> |
| 2 | QES | B'0 | R/W | <p>Queue Error interrupt Set</p> <p>This bit supports interrupt enable. It controls set of EIC.QEE.</p> <p>This bit is always read as 0.</p> <p>0: No change of EIC.QEE</p> <p>1: Set EIC.QEE to 1</p> |
| 1 | MTES | B'0 | R/W | <p>E-MAC Transmission Error interrupt Set</p> <p>This bit supports interrupt enable. It controls set of EIC.MTEE.</p> <p>This bit is always read as 0.</p> <p>0: No change of EIC.MTEE</p> <p>1: Set EIC.MTEE to 1</p> |
| 0 | MRES | B'0 | R/W | <p>E-MAC Reception Error interrupt Set</p> <p>This bit supports interrupt enable. It controls set of EIC.MREE.</p> <p>This bit is always read as 0.</p> <p>0: No change of EIC.MREE</p> <p>1: Set EIC.MREE to 1</p> |

46.2.73 Error Interrupt Disable register (EID)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The EID register is used to control the Error Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|------|------|------|-------|-------|-------|-------|-----|-----|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | TBFD | MFFD | TFFD | CULD1 | CULD0 | CLLD1 | CLLD0 | SED | QED | MTED | MRED |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 11 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 10 | TBFD | B'0 | R/W | Tx-Buffer Full interrupt Disable This bit supports interrupt enable. It controls set of EIC.TBFE. This bit is always read as 0. 0: No change of EIC.TBFE 1: Set EIC.TBFE to 0 |
| 9 | MFFD | B'0 | R/W | E-MAC status FIFO Full interrupt Disable This bit supports interrupt enable. It controls set of EIC.MFFE. This bit is always read as 0. 0: No change of EIC.MFFE 1: Set EIC.MFFE to 0 |
| 8 | TFFD | B'0 | R/W | Timestamp FIFO Full interrupt Disable This bit supports interrupt enable. It controls set of EIC.TFFE. This bit is always read as 0. 0: No change of EIC.TFFE 1: Set EIC.TFFE to 0 |
| 7 | CULD1 | B'0 | R/W | CBS Upper Limit reached interrupt Disable 1 This bit supports interrupt enable. It controls set of EIC.CULE1. This bit is always read as 0. 0: No change of EIC.CULE1 1: Set EIC.CULE1 to 0 |
| 6 | CULD0 | B'0 | R/W | CBS Upper Limit reached interrupt Disable 0 This bit supports interrupt enable. It controls set of EIC.CULE0. This bit is always read as 0. 0: No change of EIC.CULE0 1: Set EIC.CULE0 to 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 5 | CLLD1 | B'0 | R/W | <p>CBS Lower Limit reached interrupt Disable 1</p> <p>This bit supports interrupt enable. It controls set of EIC.CLLE1. This bit is always read as 0. 0: No change of EIC.CLLE1 1: Set EIC.CLLE1 to 0</p> |
| 4 | CLLD0 | B'0 | R/W | <p>CBS Lower Limit reached interrupt Disable 0</p> <p>This bit supports interrupt enable. It controls set of EIC.CLLE0. This bit is always read as 0. 0: No change of EIC.CLLE0 1: Set EIC.CLLE0 to 0</p> |
| 3 | SED | B'0 | R/W | <p>Separation Error interrupt Disable</p> <p>This bit supports interrupt enable. It controls set of EIC.SEE. This bit is always read as 0. 0: No change of EIC.SEE 1: Set EIC.SEE to 0</p> |
| 2 | QED | B'0 | R/W | <p>Queue Error interrupt Disable</p> <p>This bit supports interrupt enable. It controls set of EIC.QEE. This bit is always read as 0. 0: No change of EIC.QEE 1: Set EIC.QEE to 0</p> |
| 1 | MTED | B'0 | R/W | <p>E-MAC Transmission Error interrupt Disable</p> <p>This bit supports interrupt enable. It controls set of EIC.MTEE. This bit is always read as 0. 0: No change of EIC.MTEE 1: Set EIC.MTEE to 0</p> |
| 0 | MRED | B'0 | R/W | <p>E-MAC Reception Error interrupt Disable</p> <p>This bit supports interrupt enable. It controls set of EIC.MREE. This bit is always read as 0. 0: No change of EIC.MREE 1: Set EIC.MREE to 0</p> |

46.2.74 Reception Interrupt Enable register 0 (RIE0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RIE0 register is used to control the Reception Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | FRS17 | FRS16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | FRS15 | FRS14 | FRS13 | FRS12 | FRS11 | FRS10 | FRS9 | FRS8 | FRS7 | FRS6 | FRS5 | FRS4 | FRS3 | FRS2 | FRS1 | FRS0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 | FRS17 | B'0 | R/W | Frame Received interrupt Set 17 This bit supports interrupt enable. It controls set of RIC0.FRE17. This bit is always read as 0. 0: No change of RIC0.FRE17 1: Set RIC0.FRE17 to 1 |
| 16 | FRS16 | B'0 | R/W | Frame Received interrupt Set 16 This bit supports interrupt enable. It controls set of RIC0.FRE16. This bit is always read as 0. 0: No change of RIC0.FRE16 1: Set RIC0.FRE16 to 1 |
| 15 | FRS15 | B'0 | R/W | Frame Received interrupt Set 15 This bit supports interrupt enable. It controls set of RIC0.FRE15. This bit is always read as 0. 0: No change of RIC0.FRE15 1: Set RIC0.FRE15 to 1 |
| 14 | FRS14 | B'0 | R/W | Frame Received interrupt Set 14 This bit supports interrupt enable. It controls set of RIC0.FRE14. This bit is always read as 0. 0: No change of RIC0.FRE14 1: Set RIC0.FRE14 to 1 |
| 13 | FRS13 | B'0 | R/W | Frame Received interrupt Set 13 This bit supports interrupt enable. It controls set of RIC0.FRE13. This bit is always read as 0. 0: No change of RIC0.FRE13 1: Set RIC0.FRE13 to 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 12 | FRS12 | B'0 | R/W | <p>Frame Received interrupt Set 12</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE12. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE12 1: Set RIC0.FRE12 to 1</p> |
| 11 | FRS11 | B'0 | R/W | <p>Frame Received interrupt Set 11</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE11. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE11 1: Set RIC0.FRE11 to 1</p> |
| 10 | FRS10 | B'0 | R/W | <p>Frame Received interrupt Set 10</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE10. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE10 1: Set RIC0.FRE10 to 1</p> |
| 9 | FRS9 | B'0 | R/W | <p>Frame Received interrupt Set 9</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE9. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE9 1: Set RIC0.FRE9 to 1</p> |
| 8 | FRS8 | B'0 | R/W | <p>Frame Received interrupt Set 8</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE8. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE8 1: Set RIC0.FRE8 to 1</p> |
| 7 | FRS7 | B'0 | R/W | <p>Frame Received interrupt Set 7</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE7. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE7 1: Set RIC0.FRE7 to 1</p> |
| 6 | FRS6 | B'0 | R/W | <p>Frame Received interrupt Set 6</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE6. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE6 1: Set RIC0.FRE6 to 1</p> |
| 5 | FRS5 | B'0 | R/W | <p>Frame Received interrupt Set 5</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE5. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE5 1: Set RIC0.FRE5 to 1</p> |
| 4 | FRS4 | B'0 | R/W | <p>Frame Received interrupt Set 4</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE4. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE4 1: Set RIC0.FRE4 to 1</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | FRS3 | B'0 | R/W | <p>Frame Received interrupt Set 3</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE3.</p> <p>This bit is always read as 0.</p> <p>0: No change of RIC0.FRE3</p> <p>1: Set RIC0.FRE3 to 1</p> |
| 2 | FRS2 | B'0 | R/W | <p>Frame Received interrupt Set 2</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE2.</p> <p>This bit is always read as 0.</p> <p>0: No change of RIC0.FRE2</p> <p>1: Set RIC0.FRE2 to 1</p> |
| 1 | FRS1 | B'0 | R/W | <p>Frame Received interrupt Set 1</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE1.</p> <p>This bit is always read as 0.</p> <p>0: No change of RIC0.FRE1</p> <p>1: Set RIC0.FRE1 to 1</p> |
| 0 | FRS0 | B'0 | R/W | <p>Frame Received interrupt Set 0</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE0.</p> <p>This bit is always read as 0.</p> <p>0: No change of RIC0.FRE0</p> <p>1: Set RIC0.FRE0 to 1</p> |

46.2.75 Reception Interrupt Disable register 0 (RID0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RID0 register is used to control the Reception Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | FRD17 | FRD16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | FRD15 | FRD14 | FRD13 | FRD12 | FRD11 | FRD10 | FRD9 | FRD8 | FRD7 | FRD6 | FRD5 | FRD4 | FRD3 | FRD2 | FRD1 | FRD0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 | FRD17 | B'0 | R/W | Frame Received interrupt Disable 17 This bit supports interrupt enable. It controls set of RIC0.FRE17. This bit is always read as 0. 0: No change of RIC0.FRE17 1: Set RIC0.FRE17 to 0 |
| 16 | FRD16 | B'0 | R/W | Frame Received interrupt Disable 16 This bit supports interrupt enable. It controls set of RIC0.FRE16. This bit is always read as 0. 0: No change of RIC0.FRE16 1: Set RIC0.FRE16 to 0 |
| 15 | FRD15 | B'0 | R/W | Frame Received interrupt Disable 15 This bit supports interrupt enable. It controls set of RIC0.FRE15. This bit is always read as 0. 0: No change of RIC0.FRE15 1: Set RIC0.FRE15 to 0 |
| 14 | FRD14 | B'0 | R/W | Frame Received interrupt Disable 14 This bit supports interrupt enable. It controls set of RIC0.FRE14. This bit is always read as 0. 0: No change of RIC0.FRE14 1: Set RIC0.FRE14 to 0 |
| 13 | FRD13 | B'0 | R/W | Frame Received interrupt Disable 13 This bit supports interrupt enable. It controls set of RIC0.FRE13. This bit is always read as 0. 0: No change of RIC0.FRE13 1: Set RIC0.FRE13 to 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 12 | FRD12 | B'0 | R/W | <p>Frame Received interrupt Disable 12</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE12. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE12 1: Set RIC0.FRE12 to 0</p> |
| 11 | FRD11 | B'0 | R/W | <p>Frame Received interrupt Disable 11</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE11. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE11 1: Set RIC0.FRE11 to 0</p> |
| 10 | FRD10 | B'0 | R/W | <p>Frame Received interrupt Disable 10</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE10. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE10 1: Set RIC0.FRE10 to 0</p> |
| 9 | FRD9 | B'0 | R/W | <p>Frame Received interrupt Disable 9</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE9. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE9 1: Set RIC0.FRE9 to 0</p> |
| 8 | FRD8 | B'0 | R/W | <p>Frame Received interrupt Disable 8</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE8. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE8 1: Set RIC0.FRE8 to 0</p> |
| 7 | FRD7 | B'0 | R/W | <p>Frame Received interrupt Disable 7</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE7. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE7 1: Set RIC0.FRE7 to 0</p> |
| 6 | FRD6 | B'0 | R/W | <p>Frame Received interrupt Disable 6</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE6. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE6 1: Set RIC0.FRE6 to 0</p> |
| 5 | FRD5 | B'0 | R/W | <p>Frame Received interrupt Disable 5</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE5. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE5 1: Set RIC0.FRE5 to 0</p> |
| 4 | FRD4 | B'0 | R/W | <p>Frame Received interrupt Disable 4</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE4. This bit is always read as 0.</p> <p>0: No change of RIC0.FRE4 1: Set RIC0.FRE4 to 0</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | FRD3 | B'0 | R/W | <p>Frame Received interrupt Disable 3</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE3.</p> <p>This bit is always read as 0.</p> <p>0: No change of RIC0.FRE3</p> <p>1: Set RIC0.FRE3 to 0</p> |
| 2 | FRD2 | B'0 | R/W | <p>Frame Received interrupt Disable 2</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE2.</p> <p>This bit is always read as 0.</p> <p>0: No change of RIC0.FRE2</p> <p>1: Set RIC0.FRE2 to 0</p> |
| 1 | FRD1 | B'0 | R/W | <p>Frame Received interrupt Disable 1</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE1.</p> <p>This bit is always read as 0.</p> <p>0: No change of RIC0.FRE1</p> <p>1: Set RIC0.FRE1 to 0</p> |
| 0 | FRD0 | B'0 | R/W | <p>Frame Received interrupt Disable 0</p> <p>This bit supports interrupt enable. It controls set of RIC0.FRE0.</p> <p>This bit is always read as 0.</p> <p>0: No change of RIC0.FRE0</p> <p>1: Set RIC0.FRE0 to 0</p> |

46.2.76 Reception Interrupt Enable register 1 (RIE1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RIE1 register is used to control the Reception Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|--------|--------|--------|--------|--------|--------|------|------|------|------|------|------|------|------|--------|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RFWS | — | — | — | — | — | — | — | — | — | — | — | — | — | RWS 17 | RWS 16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RWS 15 | RWS 14 | RWS 13 | RWS 12 | RWS 11 | RWS 10 | RWS9 | RWS8 | RWS7 | RWS6 | RWS5 | RWS4 | RWS3 | RWS2 | RWS1 | RWS0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | RFWS | B'0 | R/W | Rx-FIFO Warning interrupt Set This bit supports interrupt enable. It controls set of RIC1.RFWE. This bit is always read as 0. 0: No change of RIC1.RFWE 1: Set RIC1.RFWE to 1 |
| 30 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 | RWS17 | B'0 | R/W | Reception Warning interrupt Set 17 This bit supports interrupt enable. It controls set of RIC1.RWE17. This bit is always read as 0. 0: No change of RIC1.RWE17 1: Set RIC1.RWE17 to 1 |
| 16 | RWS16 | B'0 | R/W | Reception Warning interrupt Set 16 This bit supports interrupt enable. It controls set of RIC1.RWE16. This bit is always read as 0. 0: No change of RIC1.RWE16 1: Set RIC1.RWE16 to 1 |
| 15 | RWS15 | B'0 | R/W | Reception Warning interrupt Set 15 This bit supports interrupt enable. It controls set of RIC1.RWE15. This bit is always read as 0. 0: No change of RIC1.RWE15 1: Set RIC1.RWE15 to 1 |
| 14 | RWS14 | B'0 | R/W | Reception Warning interrupt Set 14 This bit supports interrupt enable. It controls set of RIC1.RWE14. This bit is always read as 0. 0: No change of RIC1.RWE14 1: Set RIC1.RWE14 to 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 13 | RWS13 | B'0 | R/W | <p>Reception Warning interrupt Set 13</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE13. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE13 1: Set RIC1.RWE13 to 1</p> |
| 12 | RWS12 | B'0 | R/W | <p>Reception Warning interrupt Set 12</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE12. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE12 1: Set RIC1.RWE12 to 1</p> |
| 11 | RWS11 | B'0 | R/W | <p>Reception Warning interrupt Set 11</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE11. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE11 1: Set RIC1.RWE11 to 1</p> |
| 10 | RWS10 | B'0 | R/W | <p>Reception Warning interrupt Set 10</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE10. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE10 1: Set RIC1.RWE10 to 1</p> |
| 9 | RWS9 | B'0 | R/W | <p>Reception Warning interrupt Set 9</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE9. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE9 1: Set RIC1.RWE9 to 1</p> |
| 8 | RWS8 | B'0 | R/W | <p>Reception Warning interrupt Set 8</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE8. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE8 1: Set RIC1.RWE8 to 1</p> |
| 7 | RWS7 | B'0 | R/W | <p>Reception Warning interrupt Set 7</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE7. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE7 1: Set RIC1.RWE7 to 1</p> |
| 6 | RWS6 | B'0 | R/W | <p>Reception Warning interrupt Set 6</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE6. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE6 1: Set RIC1.RWE6 to 1</p> |
| 5 | RWS5 | B'0 | R/W | <p>Reception Warning interrupt Set 5</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE5. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE5 1: Set RIC1.RWE5 to 1</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 4 | RWS4 | B'0 | R/W | Reception Warning interrupt Set 4 This bit supports interrupt enable. It controls set of RIC1.RWE4. This bit is always read as 0. 0: No change of RIC1.RWE4 1: Set RIC1.RWE4 to 1 |
| 3 | RWS3 | B'0 | R/W | Reception Warning interrupt Set 3 This bit supports interrupt enable. It controls set of RIC1.RWE3. This bit is always read as 0. 0: No change of RIC1.RWE3 1: Set RIC1.RWE3 to 1 |
| 2 | RWS2 | B'0 | R/W | Reception Warning interrupt Set 2 This bit supports interrupt enable. It controls set of RIC1.RWE2. This bit is always read as 0. 0: No change of RIC1.RWE2 1: Set RIC1.RWE2 to 1 |
| 1 | RWS1 | B'0 | R/W | Reception Warning interrupt Set 1 This bit supports interrupt enable. It controls set of RIC1.RWE1. This bit is always read as 0. 0: No change of RIC1.RWE1 1: Set RIC1.RWE1 to 1 |
| 0 | RWS0 | B'0 | R/W | Reception Warning interrupt Set 0 This bit supports interrupt enable. It controls set of RIC1.RWE0. This bit is always read as 0. 0: No change of RIC1.RWE0 1: Set RIC1.RWE0 to 1 |

46.2.77 Reception Interrupt Disable register 1 (RID1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RID1 register is used to control the Reception Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|--------|--------|--------|--------|--------|--------|------|------|------|------|------|------|------|------|--------|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RFWD | — | — | — | — | — | — | — | — | — | — | — | — | — | RWD 17 | RWD 16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RWD 15 | RWD 14 | RWD 13 | RWD 12 | RWD 11 | RWD 10 | RWD9 | RWD8 | RWD7 | RWD6 | RWD5 | RWD4 | RWD3 | RWD2 | RWD1 | RWD0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | RFWD | B'0 | R/W | Rx-FIFO Warning interrupt Disable This bit supports interrupt enable. It controls set of RIC1.RFWE. This bit is always read as 0. 0: No change of RIC1.RFWE 1: Set RIC1.RFWE to 0 |
| 30 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 | RWD17 | B'0 | R/W | Reception Warning interrupt Disable 17 This bit supports interrupt enable. It controls set of RIC1.RWE17. This bit is always read as 0. 0: No change of RIC1.RWE17 1: Set RIC1.RWE17 to 0 |
| 16 | RWD16 | B'0 | R/W | Reception Warning interrupt Disable 16 This bit supports interrupt enable. It controls set of RIC1.RWE16. This bit is always read as 0. 0: No change of RIC1.RWE16 1: Set RIC1.RWE16 to 0 |
| 15 | RWD15 | B'0 | R/W | Reception Warning interrupt Disable 15 This bit supports interrupt enable. It controls set of RIC1.RWE15. This bit is always read as 0. 0: No change of RIC1.RWE15 1: Set RIC1.RWE15 to 0 |
| 14 | RWD14 | B'0 | R/W | Reception Warning interrupt Disable 14 This bit supports interrupt enable. It controls set of RIC1.RWE14. This bit is always read as 0. 0: No change of RIC1.RWE14 1: Set RIC1.RWE14 to 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 13 | RWD13 | B'0 | R/W | <p>Reception Warning interrupt Disable 13</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE13. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE13 1: Set RIC1.RWE13 to 0</p> |
| 12 | RWD12 | B'0 | R/W | <p>Reception Warning interrupt Disable 12</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE12. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE12 1: Set RIC1.RWE12 to 0</p> |
| 11 | RWD11 | B'0 | R/W | <p>Reception Warning interrupt Disable 11</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE11. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE11 1: Set RIC1.RWE11 to 0</p> |
| 10 | RWD10 | B'0 | R/W | <p>Reception Warning interrupt Disable 10</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE10. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE10 1: Set RIC1.RWE10 to 0</p> |
| 9 | RWD9 | B'0 | R/W | <p>Reception Warning interrupt Disable 9</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE9. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE9 1: Set RIC1.RWE9 to 0</p> |
| 8 | RWD8 | B'0 | R/W | <p>Reception Warning interrupt Disable 8</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE8. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE8 1: Set RIC1.RWE8 to 0</p> |
| 7 | RWD7 | B'0 | R/W | <p>Reception Warning interrupt Disable 7</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE7. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE7 1: Set RIC1.RWE7 to 0</p> |
| 6 | RWD6 | B'0 | R/W | <p>Reception Warning interrupt Disable 6</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE6. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE6 1: Set RIC1.RWE6 to 0</p> |
| 5 | RWD5 | B'0 | R/W | <p>Reception Warning interrupt Disable 5</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE5. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE5 1: Set RIC1.RWE5 to 0</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 4 | RWD4 | B'0 | R/W | <p>Reception Warning interrupt Disable 4</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE4. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE4 1: Set RIC1.RWE4 to 0</p> |
| 3 | RWD3 | B'0 | R/W | <p>Reception Warning interrupt Disable 3</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE3. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE3 1: Set RIC1.RWE3 to 0</p> |
| 2 | RWD2 | B'0 | R/W | <p>Reception Warning interrupt Disable 2</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE2. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE2 1: Set RIC1.RWE2 to 0</p> |
| 1 | RWD1 | B'0 | R/W | <p>Reception Warning interrupt Disable 1</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE1. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE1 1: Set RIC1.RWE1 to 0</p> |
| 0 | RWD0 | B'0 | R/W | <p>Reception Warning interrupt Disable 0</p> <p>This bit supports interrupt enable. It controls set of RIC1.RWE0. This bit is always read as 0.</p> <p>0: No change of RIC1.RWE0 1: Set RIC1.RWE0 to 0</p> |

46.2.78 Reception Interrupt Enable register 2 (RIE2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RIE2 register is used to control the Reception Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RFFS | — | — | — | — | — | — | — | — | — | — | — | — | — | QFS17 | QFS16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | QFS15 | QFS14 | QFS13 | QFS12 | QFS11 | QFS10 | QFS9 | QFS8 | QFS7 | QFS6 | QFS5 | QFS4 | QFS3 | QFS2 | QFS1 | QFS0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | RFFS | B'0 | R/W | Rx-FIFO Full interrupt Set This bit supports interrupt enable. It controls set of RIC2.RFFE. This bit is always read as 0. 0: No change of RIC2.RFFE 1: Set RIC2.RFFE to 1 |
| 30 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 | QFS17 | B'0 | R/W | Queue Full interrupt Set 17 This bit supports interrupt enable. It controls set of RIC2.QFE17. This bit is always read as 0. 0: No change of RIC2.QFE17 1: Set RIC2.QFE17 to 1 |
| 16 | QFS16 | B'0 | R/W | Queue Full interrupt Set 16 This bit supports interrupt enable. It controls set of RIC2.QFE16. This bit is always read as 0. 0: No change of RIC2.QFE16 1: Set RIC2.QFE16 to 1 |
| 15 | QFS15 | B'0 | R/W | Queue Full interrupt Set 15 This bit supports interrupt enable. It controls set of RIC2.QFE15. This bit is always read as 0. 0: No change of RIC2.QFE15 1: Set RIC2.QFE15 to 1 |
| 14 | QFS14 | B'0 | R/W | Queue Full interrupt Set 14 This bit supports interrupt enable. It controls set of RIC2.QFE14. This bit is always read as 0. 0: No change of RIC2.QFE14 1: Set RIC2.QFE14 to 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 13 | QFS13 | B'0 | R/W | Queue Full interrupt Set 13 This bit supports interrupt enable. It controls set of RIC2.QFE13. This bit is always read as 0. 0: No change of RIC2.QFE13 1: Set RIC2.QFE13 to 1 |
| 12 | QFS12 | B'0 | R/W | Queue Full interrupt Set 12 This bit supports interrupt enable. It controls set of RIC2.QFE12. This bit is always read as 0. 0: No change of RIC2.QFE12 1: Set RIC2.QFE12 to 1 |
| 11 | QFS11 | B'0 | R/W | Queue Full interrupt Set 11 This bit supports interrupt enable. It controls set of RIC2.QFE11. This bit is always read as 0. 0: No change of RIC2.QFE11 1: Set RIC2.QFE11 to 1 |
| 10 | QFS10 | B'0 | R/W | Queue Full interrupt Set 10 This bit supports interrupt enable. It controls set of RIC2.QFE10. This bit is always read as 0. 0: No change of RIC2.QFE10 1: Set RIC2.QFE10 to 1 |
| 9 | QFS9 | B'0 | R/W | Queue Full interrupt Set 9 This bit supports interrupt enable. It controls set of RIC2.QFE9. This bit is always read as 0. 0: No change of RIC2.QFE9 1: Set RIC2.QFE9 to 1 |
| 8 | QFS8 | B'0 | R/W | Queue Full interrupt Set 8 This bit supports interrupt enable. It controls set of RIC2.QFE8. This bit is always read as 0. 0: No change of RIC2.QFE8 1: Set RIC2.QFE8 to 1 |
| 7 | QFS7 | B'0 | R/W | Queue Full interrupt Set 7 This bit supports interrupt enable. It controls set of RIC2.QFE7. This bit is always read as 0. 0: No change of RIC2.QFE7 1: Set RIC2.QFE7 to 1 |
| 6 | QFS6 | B'0 | R/W | Queue Full interrupt Set 6 This bit supports interrupt enable. It controls set of RIC2.QFE6. This bit is always read as 0. 0: No change of RIC2.QFE6 1: Set RIC2.QFE6 to 1 |
| 5 | QFS5 | B'0 | R/W | Queue Full interrupt Set 5 This bit supports interrupt enable. It controls set of RIC2.QFE5. This bit is always read as 0. 0: No change of RIC2.QFE5 1: Set RIC2.QFE5 to 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 4 | QFS4 | B'0 | R/W | Queue Full interrupt Set 4 This bit supports interrupt enable. It controls set of RIC2.QFE4. This bit is always read as 0. 0: No change of RIC2.QFE4 1: Set RIC2.QFE4 to 1 |
| 3 | QFS3 | B'0 | R/W | Queue Full interrupt Set 3 This bit supports interrupt enable. It controls set of RIC2.QFE3. This bit is always read as 0. 0: No change of RIC2.QFE3 1: Set RIC2.QFE3 to 1 |
| 2 | QFS2 | B'0 | R/W | Queue Full interrupt Set 2 This bit supports interrupt enable. It controls set of RIC2.QFE2. This bit is always read as 0. 0: No change of RIC2.QFE2 1: Set RIC2.QFE2 to 1 |
| 1 | QFS1 | B'0 | R/W | Queue Full interrupt Set 1 This bit supports interrupt enable. It controls set of RIC2.QFE1. This bit is always read as 0. 0: No change of RIC2.QFE1 1: Set RIC2.QFE1 to 1 |
| 0 | QFS0 | B'0 | R/W | Queue Full interrupt Set 0 This bit supports interrupt enable. It controls set of RIC2.QFE0. This bit is always read as 0. 0: No change of RIC2.QFE0 1: Set RIC2.QFE0 to 1 |

46.2.79 Reception Interrupt Disable register 2 (RID2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RID2 register is used to control the Reception Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|--------|--------|--------|--------|--------|--------|------|------|------|------|------|------|------|------|--------|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RFFD | — | — | — | — | — | — | — | — | — | — | — | — | — | QFD 17 | QFD 16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | QFD 15 | QFD 14 | QFD 13 | QFD 12 | QFD 11 | QFD 10 | QFD9 | QFD8 | QFD7 | QFD6 | QFD5 | QFD4 | QFD3 | QFD2 | QFD1 | QFD0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | RFFD | B'0 | R/W | Rx-FIFO Full interrupt Disable This bit supports interrupt enable. It controls set of RIC2.RFFE. This bit is always read as 0. 0: No change of RIC2.RFFE 1: Set RIC2.RFFE to 0 |
| 30 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 | QFD17 | B'0 | R/W | Queue Full interrupt Disable 17 This bit supports interrupt enable. It controls set of RIC2.QFE17. This bit is always read as 0. 0: No change of RIC2.QFE17 1: Set RIC2.QFE17 to 0 |
| 16 | QFD16 | B'0 | R/W | Queue Full interrupt Disable 16 This bit supports interrupt enable. It controls set of RIC2.QFE16. This bit is always read as 0. 0: No change of RIC2.QFE16 1: Set RIC2.QFE16 to 0 |
| 15 | QFD15 | B'0 | R/W | Queue Full interrupt Disable 15 This bit supports interrupt enable. It controls set of RIC2.QFE15. This bit is always read as 0. 0: No change of RIC2.QFE15 1: Set RIC2.QFE15 to 0 |
| 14 | QFD14 | B'0 | R/W | Queue Full interrupt Disable 14 This bit supports interrupt enable. It controls set of RIC2.QFE14. This bit is always read as 0. 0: No change of RIC2.QFE14 1: Set RIC2.QFE14 to 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 13 | QFD13 | B'0 | R/W | Queue Full interrupt Disable 13 This bit supports interrupt enable. It controls set of RIC2.QFE13. This bit is always read as 0. 0: No change of RIC2.QFE13 1: Set RIC2.QFE13 to 0 |
| 12 | QFD12 | B'0 | R/W | Queue Full interrupt Disable 12 This bit supports interrupt enable. It controls set of RIC2.QFE12. This bit is always read as 0. 0: No change of RIC2.QFE12 1: Set RIC2.QFE12 to 0 |
| 11 | QFD11 | B'0 | R/W | Queue Full interrupt Disable 11 This bit supports interrupt enable. It controls set of RIC2.QFE11. This bit is always read as 0. 0: No change of RIC2.QFE11 1: Set RIC2.QFE11 to 0 |
| 10 | QFD10 | B'0 | R/W | Queue Full interrupt Disable 10 This bit supports interrupt enable. It controls set of RIC2.QFE10. This bit is always read as 0. 0: No change of RIC2.QFE10 1: Set RIC2.QFE10 to 0 |
| 9 | QFD9 | B'0 | R/W | Queue Full interrupt Disable 9 This bit supports interrupt enable. It controls set of RIC2.QFE9. This bit is always read as 0. 0: No change of RIC2.QFE9 1: Set RIC2.QFE9 to 0 |
| 8 | QFD8 | B'0 | R/W | Queue Full interrupt Disable 8 This bit supports interrupt enable. It controls set of RIC2.QFE8. This bit is always read as 0. 0: No change of RIC2.QFE8 1: Set RIC2.QFE8 to 0 |
| 7 | QFD7 | B'0 | R/W | Queue Full interrupt Disable 7 This bit supports interrupt enable. It controls set of RIC2.QFE7. This bit is always read as 0. 0: No change of RIC2.QFE7 1: Set RIC2.QFE7 to 0 |
| 6 | QFD6 | B'0 | R/W | Queue Full interrupt Disable 6 This bit supports interrupt enable. It controls set of RIC2.QFE6. This bit is always read as 0. 0: No change of RIC2.QFE6 1: Set RIC2.QFE6 to 0 |
| 5 | QFD5 | B'0 | R/W | Queue Full interrupt Disable 5 This bit supports interrupt enable. It controls set of RIC2.QFE5. This bit is always read as 0. 0: No change of RIC2.QFE5 1: Set RIC2.QFE5 to 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 4 | QFD4 | B'0 | R/W | Queue Full interrupt Disable 4 This bit supports interrupt enable. It controls set of RIC2.QFE4. This bit is always read as 0. 0: No change of RIC2.QFE4 1: Set RIC2.QFE4 to 0 |
| 3 | QFD3 | B'0 | R/W | Queue Full interrupt Disable 3 This bit supports interrupt enable. It controls set of RIC2.QFE3. This bit is always read as 0. 0: No change of RIC2.QFE3 1: Set RIC2.QFE3 to 0 |
| 2 | QFD2 | B'0 | R/W | Queue Full interrupt Disable 2 This bit supports interrupt enable. It controls set of RIC2.QFE2. This bit is always read as 0. 0: No change of RIC2.QFE2 1: Set RIC2.QFE2 to 0 |
| 1 | QFD1 | B'0 | R/W | Queue Full interrupt Disable 1 This bit supports interrupt enable. It controls set of RIC2.QFE1. This bit is always read as 0. 0: No change of RIC2.QFE1 1: Set RIC2.QFE1 to 0 |
| 0 | QFD0 | B'0 | R/W | Queue Full interrupt Disable 0 This bit supports interrupt enable. It controls set of RIC2.QFE0. This bit is always read as 0. 0: No change of RIC2.QFE0 1: Set RIC2.QFE0 to 0 |

46.2.80 Transmission Interrupt Enable register (TIE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The TIE register is used to control the Transmission Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|------|------|------|------|-----|-----|-----|-----|-------|-----------|-----------|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | TDPS3 | TDPS 2 | TDPS 1 | TDPS 0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | MFWS | MFUS | TFWS | TFUS | — | — | — | — | FTS3 | FTS2 | FTS1 | FTS0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 20 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 19 | TDPS3 | B'0 | R/W | Transmit Descriptor Processed interrupt Set 3 This bit supports interrupt enable. It controls set of TIC.TDPE3. This bit is always read as 0. 0: No change of TIC.TDPE3 1: Set TIC.TDPE3 to 1 |
| 18 | TDPS2 | B'0 | R/W | Transmit Descriptor Processed interrupt Set 2 This bit supports interrupt enable. It controls set of TIC.TDPE2. This bit is always read as 0. 0: No change of TIC.TDPE2 1: Set TIC.TDPE2 to 1 |
| 17 | TDPS1 | B'0 | R/W | Transmit Descriptor Processed interrupt Set 1 This bit supports interrupt enable. It controls set of TIC.TDPE1. This bit is always read as 0. 0: No change of TIC.TDPE1 1: Set TIC.TDPE1 to 1 |
| 16 | TDPS0 | B'0 | R/W | Transmit Descriptor Processed interrupt Set 0 This bit supports interrupt enable. It controls set of TIC.TDPE0. This bit is always read as 0. 0: No change of TIC.TDPE0 1: Set TIC.TDPE0 to 1 |
| 15 to 12 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 11 | MFWS | B'0 | R/W | E-MAC status FIFO Warning interrupt Set This bit supports interrupt enable. It controls set of TIC.MFWE. This bit is always read as 0. 0: No change of TIC.MFWE 1: Set TIC.MFWE to 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 10 | MFUS | B'0 | R/W | E-MAC status FIFO Updated interrupt Set This bit supports interrupt enable. It controls set of TIC.MFUE. This bit is always read as 0. 0: No change of TIC.MFUE 1: Set TIC.MFUE to 1 |
| 9 | TFWS | B'0 | R/W | Timestamp FIFO Warning interrupt Set This bit supports interrupt enable. It controls set of TIC.TFWE. This bit is always read as 0. 0: No change of TIC.TFWE 1: Set TIC.TFWE to 1 |
| 8 | TFUS | B'0 | R/W | Timestamp FIFO Updated interrupt Set This bit supports interrupt enable. It controls set of TIC.TFUE. This bit is always read as 0. 0: No change of TIC.TFUE 1: Set TIC.TFUE to 1 |
| 7 to 4 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 3 | FTS3 | B'0 | R/W | Frame Transmitted interrupt Set 3 This bit supports interrupt enable. It controls set of TIC.FTE3. This bit is always read as 0. 0: No change of TIC.FTE3 1: Set TIC.FTE3 to 1 |
| 2 | FTS2 | B'0 | R/W | Frame Transmitted interrupt Set 2 This bit supports interrupt enable. It controls set of TIC.FTE2. This bit is always read as 0. 0: No change of TIC.FTE2 1: Set TIC.FTE2 to 1 |
| 1 | FTS1 | B'0 | R/W | Frame Transmitted interrupt Set 1 This bit supports interrupt enable. It controls set of TIC.FTE1. This bit is always read as 0. 0: No change of TIC.FTE1 1: Set TIC.FTE1 to 1 |
| 0 | FTS0 | B'0 | R/W | Frame Transmitted interrupt Set 0 This bit supports interrupt enable. It controls set of TIC.FTE0. This bit is always read as 0. 0: No change of TIC.FTE0 1: Set TIC.FTE0 to 1 |

46.2.81 Transmission Interrupt Disable register (TID)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The TID register is used to control the Transmission Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|------|------|------|------|-----|-----|-----|-----|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | TDPD3 | TDPD2 | TDPD1 | TDPD0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | MFWD | MFUD | TFWD | TFUD | — | — | — | — | FTD3 | FTD2 | FTD1 | FTD0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 20 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 19 | TDPD3 | All 0 | R/W | Transmit Descriptor Processed interrupt Disable 3 This bit supports interrupt enable. It controls set of TIC.TDPE3. This bit is always read as 0. 0: No change of TIC.TDPE3 1: Set TIC.TDPE3 to 0 |
| 18 | TDPD2 | All 0 | R/W | Transmit Descriptor Processed interrupt Disable 2 This bit supports interrupt enable. It controls set of TIC.TDPE2. This bit is always read as 0. 0: No change of TIC.TDPE2 1: Set TIC.TDPE2 to 0 |
| 17 | TDPD1 | All 0 | R/W | Transmit Descriptor Processed interrupt Disable 1 This bit supports interrupt enable. It controls set of TIC.TDPE1. This bit is always read as 0. 0: No change of TIC.TDPE1 1: Set TIC.TDPE1 to 0 |
| 16 | TDPD0 | All 0 | R/W | Transmit Descriptor Processed interrupt Disable 0 This bit supports interrupt enable. It controls set of TIC.TDPE0. This bit is always read as 0. 0: No change of TIC.TDPE0 1: Set TIC.TDPE0 to 0 |
| 15 to 12 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 11 | MFWD | B'0 | R/W | E-MAC status FIFO Warning interrupt Disable This bit supports interrupt enable. It controls set of TIC.MFWE. This bit is always read as 0. 0: No change of TIC.MFWE 1: Set TIC.MFWE to 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 10 | MFUD | B'0 | R/W | E-MAC status FIFO Updated interrupt Disable This bit supports interrupt enable. It controls set of TIC.MFUE. This bit is always read as 0. 0: No change of TIC.MFUE 1: Set TIC.MFUE to 0 |
| 9 | TFWD | B'0 | R/W | Timestamp FIFO Warning interrupt Disable This bit supports interrupt enable. It controls set of TIC.TFWE. This bit is always read as 0. 0: No change of TIC.TFWE 1: Set TIC.TFWE to 0 |
| 8 | TFUD | B'0 | R/W | Timestamp FIFO Updated interrupt Disable This bit supports interrupt enable. It controls set of TIC.TFUE. This bit is always read as 0. 0: No change of TIC.TFUE 1: Set TIC.TFUE to 0 |
| 7 to 4 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 3 | FTD3 | B'0 | R/W | Frame Transmitted interrupt Disable 3 This bit supports interrupt enable. It controls set of TIC.FTE3. This bit is always read as 0. 0: No change of TIC.FTE3 1: Set TIC.FTE3 to 0 |
| 2 | FTD2 | B'0 | R/W | Frame Transmitted interrupt Disable 2 This bit supports interrupt enable. It controls set of TIC.FTE2. This bit is always read as 0. 0: No change of TIC.FTE2 1: Set TIC.FTE2 to 0 |
| 1 | FTD1 | B'0 | R/W | Frame Transmitted interrupt Disable 1 This bit supports interrupt enable. It controls set of TIC.FTE1. This bit is always read as 0. 0: No change of TIC.FTE1 1: Set TIC.FTE1 to 0 |
| 0 | FTD0 | B'0 | R/W | Frame Transmitted interrupt Disable 0 This bit supports interrupt enable. It controls set of TIC.FTE0. This bit is always read as 0. 0: No change of TIC.FTE0 1: Set TIC.FTE0 to 0 |

46.2.82 Reception Interrupt Enable register 3 (RIE3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RIE3 register is used to control the Reception Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | RDPS 17 | RDPS 16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RDPS 15 | RDPS 14 | RDPS 13 | RDPS 12 | RDPS 11 | RDPS 10 | RDPS 9 | RDPS 8 | RDPS 7 | RDPS 6 | RDPS 5 | RDPS 4 | RDPS 3 | RDPS 2 | RDPS 1 | RDPS 0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 | RDPS17 | B'0 | R/W | Receive Descriptor Processed interrupt Set 17 This bit supports interrupt enable. It controls set of RIC3.RDPE17. This bit is always read as 0. 0: No change of RIC3.RDPE17 1: Set RIC3.RDPE17 to 1 |
| 16 | RDPS16 | B'0 | R/W | Receive Descriptor Processed interrupt Set 16 This bit supports interrupt enable. It controls set of RIC3.RDPE16. This bit is always read as 0. 0: No change of RIC3.RDPE16 1: Set RIC3.RDPE16 to 1 |
| 15 | RDPS15 | B'0 | R/W | Receive Descriptor Processed interrupt Set 15 This bit supports interrupt enable. It controls set of RIC3.RDPE15. This bit is always read as 0. 0: No change of RIC3.RDPE15 1: Set RIC3.RDPE15 to 1 |
| 14 | RDPS14 | B'0 | R/W | Receive Descriptor Processed interrupt Set 14 This bit supports interrupt enable. It controls set of RIC3.RDPE14. This bit is always read as 0. 0: No change of RIC3.RDPE14 1: Set RIC3.RDPE14 to 1 |
| 13 | RDPS13 | B'0 | R/W | Receive Descriptor Processed interrupt Set 13 This bit supports interrupt enable. It controls set of RIC3.RDPE13. This bit is always read as 0. 0: No change of RIC3.RDPE13 1: Set RIC3.RDPE13 to 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 12 | RDPS12 | B'0 | R/W | Receive Descriptor Processed interrupt Set 12 This bit supports interrupt enable. It controls set of RIC3.RDPE12. This bit is always read as 0. 0: No change of RIC3.RDPE12 1: Set RIC3.RDPE12 to 1 |
| 11 | RDPS11 | B'0 | R/W | Receive Descriptor Processed interrupt Set 11 This bit supports interrupt enable. It controls set of RIC3.RDPE11. This bit is always read as 0. 0: No change of RIC3.RDPE11 1: Set RIC3.RDPE11 to 1 |
| 10 | RDPS10 | B'0 | R/W | Receive Descriptor Processed interrupt Set 10 This bit supports interrupt enable. It controls set of RIC3.RDPE10. This bit is always read as 0. 0: No change of RIC3.RDPE10 1: Set RIC3.RDPE10 to 1 |
| 9 | RDPS9 | B'0 | R/W | Receive Descriptor Processed interrupt Set 9 This bit supports interrupt enable. It controls set of RIC3.RDPE9. This bit is always read as 0. 0: No change of RIC3.RDPE9 1: Set RIC3.RDPE9 to 1 |
| 8 | RDPS8 | B'0 | R/W | Receive Descriptor Processed interrupt Set 8 This bit supports interrupt enable. It controls set of RIC3.RDPE8. This bit is always read as 0. 0: No change of RIC3.RDPE8 1: Set RIC3.RDPE8 to 1 |
| 7 | RDPS7 | B'0 | R/W | Receive Descriptor Processed interrupt Set 7 This bit supports interrupt enable. It controls set of RIC3.RDPE7. This bit is always read as 0. 0: No change of RIC3.RDPE7 1: Set RIC3.RDPE7 to 1 |
| 6 | RDPS6 | B'0 | R/W | Receive Descriptor Processed interrupt Set 6 This bit supports interrupt enable. It controls set of RIC3.RDPE6. This bit is always read as 0. 0: No change of RIC3.RDPE6 1: Set RIC3.RDPE6 to 1 |
| 5 | RDPS5 | B'0 | R/W | Receive Descriptor Processed interrupt Set 5 This bit supports interrupt enable. It controls set of RIC3.RDPE5. This bit is always read as 0. 0: No change of RIC3.RDPE5 1: Set RIC3.RDPE5 to 1 |
| 4 | RDPS4 | B'0 | R/W | Receive Descriptor Processed interrupt Set 4 This bit supports interrupt enable. It controls set of RIC3.RDPE4. This bit is always read as 0. 0: No change of RIC3.RDPE4 1: Set RIC3.RDPE4 to 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | RDPS3 | B'0 | R/W | Receive Descriptor Processed interrupt Set 3 This bit supports interrupt enable. It controls set of RIC3.RDPE3. This bit is always read as 0. 0: No change of RIC3.RDPE3 1: Set RIC3.RDPE3 to 1 |
| 2 | RDPS2 | B'0 | R/W | Receive Descriptor Processed interrupt Set 2 This bit supports interrupt enable. It controls set of RIC3.RDPE2. This bit is always read as 0. 0: No change of RIC3.RDPE2 1: Set RIC3.RDPE2 to 1 |
| 1 | RDPS1 | B'0 | R/W | Receive Descriptor Processed interrupt Set 1 This bit supports interrupt enable. It controls set of RIC3.RDPE1. This bit is always read as 0. 0: No change of RIC3.RDPE1 1: Set RIC3.RDPE1 to 1 |
| 0 | RDPS0 | B'0 | R/W | Receive Descriptor Processed interrupt Set 0 This bit supports interrupt enable. It controls set of RIC3.RDPE0. This bit is always read as 0. 0: No change of RIC3.RDPE0 1: Set RIC3.RDPE0 to 1 |

46.2.83 Reception Interrupt Disable register 3 (RID3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RID3 register is used to control the Reception Interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | RDPD 17 | RDPD 16 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RDPD 15 | RDPD 14 | RDPD 13 | RDPD 12 | RDPD 11 | RDPD 10 | RDPD 9 | RDPD 8 | RDPD 7 | RDPD 6 | RDPD 5 | RDPD 4 | RDPD 3 | RDPD 2 | RDPD 1 | RDPD 0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 | RDPD17 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 17 This bit supports interrupt enable. It controls set of RIC3.RDPE17. This bit is always read as 0. 0: No change of RIC3.RDPE17 1: Set RIC3.RDPE17 to 0 |
| 16 | RDPD16 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 16 This bit supports interrupt enable. It controls set of RIC3.RDPE16. This bit is always read as 0. 0: No change of RIC3.RDPE16 1: Set RIC3.RDPE16 to 0 |
| 15 | RDPD15 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 15 This bit supports interrupt enable. It controls set of RIC3.RDPE15. This bit is always read as 0. 0: No change of RIC3.RDPE15 1: Set RIC3.RDPE15 to 0 |
| 14 | RDPD14 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 14 This bit supports interrupt enable. It controls set of RIC3.RDPE14. This bit is always read as 0. 0: No change of RIC3.RDPE14 1: Set RIC3.RDPE14 to 0 |
| 13 | RDPD13 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 13 This bit supports interrupt enable. It controls set of RIC3.RDPE13. This bit is always read as 0. 0: No change of RIC3.RDPE13 1: Set RIC3.RDPE13 to 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 12 | RDPD12 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 12 This bit supports interrupt enable. It controls set of RIC3.RDPE12. This bit is always read as 0. 0: No change of RIC3.RDPE12 1: Set RIC3.RDPE12 to 0 |
| 11 | RDPD11 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 11 This bit supports interrupt enable. It controls set of RIC3.RDPE11. This bit is always read as 0. 0: No change of RIC3.RDPE11 1: Set RIC3.RDPE11 to 0 |
| 10 | RDPD10 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 10 This bit supports interrupt enable. It controls set of RIC3.RDPE10. This bit is always read as 0. 0: No change of RIC3.RDPE10 1: Set RIC3.RDPE10 to 0 |
| 9 | RDPD9 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 9 This bit supports interrupt enable. It controls set of RIC3.RDPE9. This bit is always read as 0. 0: No change of RIC3.RDPE9 1: Set RIC3.RDPE9 to 0 |
| 8 | RDPD8 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 8 This bit supports interrupt enable. It controls set of RIC3.RDPE8. This bit is always read as 0. 0: No change of RIC3.RDPE8 1: Set RIC3.RDPE8 to 0 |
| 7 | RDPD7 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 7 This bit supports interrupt enable. It controls set of RIC3.RDPE7. This bit is always read as 0. 0: No change of RIC3.RDPE7 1: Set RIC3.RDPE7 to 0 |
| 6 | RDPD6 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 6 This bit supports interrupt enable. It controls set of RIC3.RDPE6. This bit is always read as 0. 0: No change of RIC3.RDPE6 1: Set RIC3.RDPE6 to 0 |
| 5 | RDPD5 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 5 This bit supports interrupt enable. It controls set of RIC3.RDPE5. This bit is always read as 0. 0: No change of RIC3.RDPE5 1: Set RIC3.RDPE5 to 0 |
| 4 | RDPD4 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 4 This bit supports interrupt enable. It controls set of RIC3.RDPE4. This bit is always read as 0. 0: No change of RIC3.RDPE4 1: Set RIC3.RDPE4 to 0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | RDPD3 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 3 This bit supports interrupt enable. It controls set of RIC3.RDPE3. This bit is always read as 0. 0: No change of RIC3.RDPE3 1: Set RIC3.RDPE3 to 0 |
| 2 | RDPD2 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 2 This bit supports interrupt enable. It controls set of RIC3.RDPE2. This bit is always read as 0. 0: No change of RIC3.RDPE2 1: Set RIC3.RDPE2 to 0 |
| 1 | RDPD1 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 1 This bit supports interrupt enable. It controls set of RIC3.RDPE1. This bit is always read as 0. 0: No change of RIC3.RDPE1 1: Set RIC3.RDPE1 to 0 |
| 0 | RDPD0 | B'0 | R/W | Receive Descriptor Processed interrupt Disable 0 This bit supports interrupt enable. It controls set of RIC3.RDPE0. This bit is always read as 0. 0: No change of RIC3.RDPE0 1: Set RIC3.RDPE0 to 0 |

46.2.84 E-MAC Mode Register (ECMR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

ECMR is used to specify the operating mode of the E-MAC. The settings in this register are normally made in the initialization process following a reset.

The operating mode settings must not be changed while transmission or reception is enabled (i.e. while the RE or TE bit in this register is 1).

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----------|------|-----|------|-----|------|------|------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | TRCC M | — | — | RCSC | — | DPAD | RZPF | TZPF | PFR | RXF | TXF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | MPDE | — | — | RE | TE | — | — | — | DM | PRM |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 27 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 26 | TRCCM | B'0 | R/W | Counter Clear Mode This bit sets the method for clearing the counter register. Refer to the descriptions of the counter registers. 0: Writing to a counter register leads to the register being cleared to 0. 1: Reading from a counter register leads to the register being cleared to 0. |
| 25, 24 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 23 | RCSC | B'0 | R/W | Checksum Calculation Setting this bit to 1 enables automatic calculation of checksums for data in received frames. Only the data field of an Ethernet frame without a VLAN field is in the scope of checksum calculation. Specifically, the checksum is calculated from the data field, which follows the length/type field and is followed by the CRC field. Calculation only involves 16-bit addition; it does not involve bit inversion. 0: Checksums are not automatically calculated. 1: Checksums are automatically calculated. |
| 22 | — | B'0 | R/W | Reserved This bit is read as 0. The write value should be 0. |

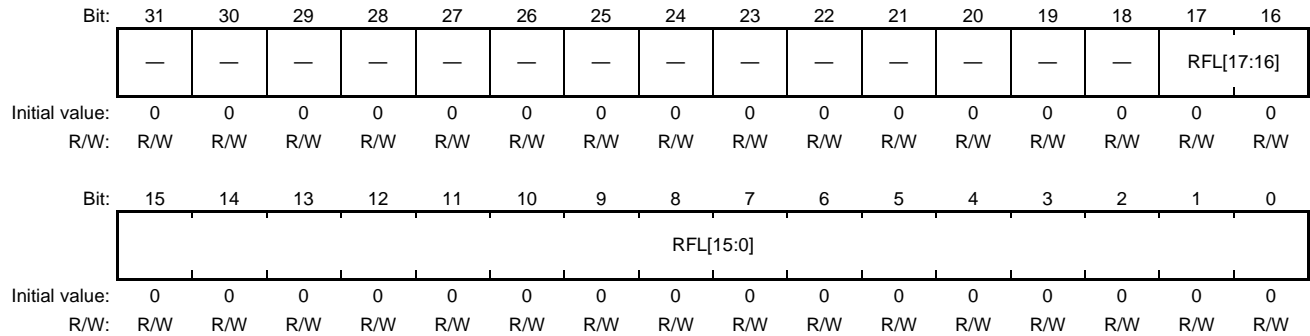
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 21 | DPAD | B'0 | R/W | <p>Data Padding</p> <p>This bit specifies padding or non-padding of data when less than 60 bytes are to be transmitted.</p> <p>When this bit is set to 1, data are transmitted without padding; when it is set to 0, data are padded to make up 60-byte units for transmission.</p> <p>0: Padding to make up 60 bytes is inserted in data for transmission when fewer than 60 bytes are to be transmitted.</p> <p>1: Padding is not inserted in data for transmission when fewer than 60 bytes are to be transmitted and the data are transmitted without being changed.</p> |
| 20 | RZPF | B'0 | R/W | <p>PAUSE Frame Reception with Time = 0</p> <p>When the RZPF bit is set to 0, received PAUSE frames with the Timer value 0 are discarded.</p> <p>When the RZPF bit is set to 1, release from the transmission wait state follows reception of a PAUSE frame with the Timer value 0.</p> <p>0: Reception of PAUSE frames with the TIME parameter value 0 is disabled.</p> <p>1: Reception of PAUSE frames with the TIME parameter value 0 is enabled.</p> |
| 19 | TZPF | B'0 | R/W | <p>Transmit Zero PAUSE Frame</p> <p>When setting CCC.FCE=1, and Rx FIFO become Warning Level.</p> <p>0: Not transmit the PAUSE frame of TIME parameter value 0.</p> <p>1: Transmit the PAUSE frame of TIME parameter value 0.</p> |
| 18 | PFR | B'0 | R/W | <p>PAUSE Frame Receive Mode</p> <p>This bit specifies whether PAUSE frames are transferred to the AVB-DMAC.</p> <p>0: PAUSE frames are not transferred to the AVB-DMAC.</p> <p>1: PAUSE frames are transferred to the AVB-DMAC.</p> |
| 17 | RXF | B'0 | R/W | <p>Operating Mode for Flow Control in Reception</p> <p>When the RXF bit is set to 1 and a PAUSE frame is received, a next frame to be transmitted is not transmitted until the time indicated by the Timer value in the PAUSE frame has elapsed. However, the transmission of a current frame is continued. The number of received PAUSE frames is also counted. For details, see section 46.2.95 PAUSE Frame Receive Counter (PFRCCR).</p> <p>Setting this bit to 0 disables PAUSE frame detection.</p> <p>0: Detection of PAUSE frames is disabled.</p> <p>1: Flow control for the receiving port is enabled.</p> |
| 16 | TXF | B'0 | R/W | <p>Transmit Flow control mode</p> <p>0: Flow control for the transmitting port is disabled.</p> <p>1: Flow control for the transmitting port is enabled.</p> |
| 15 to 10 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 9 | MPDE | B'0 | R/W | <p>Magic Packet™ Detection Enable</p> <p>The MPDE bit enables or disables Magic Packet™ detection by hardware to allow activation via the Ethernet connection.</p> <p>0: Magic Packet™ detection is not enabled.</p> <p>1: Magic Packet™ detection is enabled.</p> |
| 8, 7 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 6 | RE | B'0 | R/W | <p>Reception Enable</p> <p>If this bit is switched from reception being enabled (RE = 1) to reception being disabled (RE = 0) while a frame is being received, reception will continue until reception of the frame is completed.</p> <p>0: Reception is disabled. 1: Reception is enabled.</p> |
| 5 | TE | B'0 | R/W | <p>Transmission Enable</p> <p>If this bit is switched from transmission being enabled (TE = 1) to transmission being disabled (TE = 0) while a frame is being transmitted, transmission will continue until transmission of that frame is completed.</p> <p>0: Transmission is disabled. 1: Transmission is enabled.</p> |
| 4 to 2 | — | All 0 | R/W | <p>Reserved</p> <p>These bits are read as 0. The write value should be 0.</p> |
| 1 | DM | B'0 | R/W | <p>Duplex Mode</p> <p>This bit should always be set to 1. The value after reset is 0.</p> <p>0: Value after reset 1: Full-duplex operation</p> <p>This bit should always be set to 1.</p> |
| 0 | PRM | B'0 | R/W | <p>Promiscuous Mode</p> <p>Setting the PRM bit enables all Ethernet frames to be received. All Ethernet frames mean all receivable frames, irrespective of differences of destination address (exclude PAUSE frame).</p> <p>0: Normal operation 1: Promiscuous mode operation</p> |

46.2.85 Receive Frame Length Register (RFLR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RFLR register specifies the maximum length (in bytes) of frames that can be received by this LSI. Settings in this register must not be changed while reception is enabled (while the RE bit in the E-MAC mode register (EMCR) is 1).



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 18 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 17 to 0 | RFL[17:0] | H'0_0000 | R/W | Receive Frame Length Frame data described here refers to all fields from the destination address up to the CRC data. Frame contents from the destination address up to the data are actually transferred to memory. CRC data are not included in the transfer. When more data than the specified number of bytes are received, the portion of data that exceeds the specified value and more 8 byte length is discarded. CAUTION: The prepared descriptor data size is just the specified value (RFLR.RFL). Therefore descriptor data size must be more than RFLR.RFL + 8 if you will receive such the long frame. H'0_0000 to H'0_05EE: 1,518 bytes H'0_05EF: 1,519 bytes H'0_05F0: 1,520 bytes : : H'0_07FF: 2,047 bytes H'0_0800: 2,048 bytes : : H'0_1000: 4,096 bytes : : H'1_0000: 65,535 bytes : : H'2_0000 to H'3_FFFF: 131,072 bytes |

46.2.86 E-MAC Status Register (ECSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The ECSR register indicates the state of the E-MAC. The CPU can be notified of the state. For bits associated with interrupts, the interrupt can be enabled or disabled by the corresponding bit in the E-MAC Interrupt Permission Register (ECSIPR) described in section 46.2.88.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|-------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | PFRI | PHYI | LCHNG | MPD | ICD |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 5 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 4 | PFRI | B'0 | R/W | PAUSE Frame Retry Interrupt This bit indicate that reached to the number of PAUSE frame retransmission for register PFTTLR setting. Procedure for 0-clear to this bit is follows. 0: Read the PFTTCR register to clear the PFTTCR register. 1: Write to this bit 1. |
| 3 | PHYI | B'0 | R/W | PHY interrupt terminal state bit This bit indicates the state of input to the PHY interrupt pin (AVB_PHY_INT) from the PHY-LSI. 0: PHY interrupt terminal (AVB_PHY_INT) is not asserted. 1: PHY interrupt terminal (AVB_PHY_INT) is asserted. |
| 2 | LCHNG | B'0 | R/W | Link signal change bit This bit indicates a transition of the link status signal (AVB_LINK) input from the PHY-LSI from high to low or low to high. However, signal changes may also be detected at times when the link status signal (AVB_LINK) function is selected. To check the current link state, refer to the link status pin state bit in the PHY status register (PSR.LMON). 0: The change of Link status signal (AVB_LINK) is not detected. 1: The change of Link status signal (AVB_LINK) is detected. |
| 1 | MPD | B'0 | R/W | Magic Packet™ Detection This bit indicates that a Magic Packet™ has been detected on the line. Writing 1 to this bit clears it to 0. 0: A Magic Packet™ has not been detected. 1: A Magic Packet™ has been detected. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | ICD | B'0 | R/W | <p>Illegal Carrier Detection</p> <p>This bit indicates that the PHY-LSI has detected an illegal carrier on the line. If a change in the signal input from the PHY-LSI occurs in a period shorter than the software recognition period, the correct information may not be obtained. Refer to the timing specification for the PHY-LSI used.</p> <p>Writing 1 to this bit clears it to 0.</p> <p>0: PHY-LSI has not detected an illegal carrier on the line.</p> <p>1: PHY-LSI has detected an illegal carrier on the line.</p> |

46.2.87 E-MAC Interrupt Permission Register (ECSIPR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The ECSIPR register enables or disables the states indicated by the ECSR register as interrupt sources. Each effective bit disables or enables interrupts corresponding to the bits in ECSR.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-------|-------|--------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | PFRIM | PHYIM | LINKIM | MPDIP | ICDIP |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 5 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 4 | PFRIM | B'0 | R/W | PAUSE Frame Retry Interrupt Mask Setting this bit to 1 selects interrupt generation on setting of the PAUSE Frame Retry Interrupt bit (ECSR.PFRI) in the E-MAC status register to 1. 0: Interrupts on setting of the PFRI bit is disabled. 1: Interrupts on setting of the PFRI bit is enabled. |
| 3 | PHYIM | B'0 | R/W | PHY Interrupt Mask Setting this bit to 1 selects interrupt generation on setting of the PHY interrupt terminal state bit (ECSR.PHYI) in the E-MAC status register to 1. 0: Interrupts on setting of the PHYI bit is disabled. 1: Interrupts on setting of the PHYI bit is enabled. |
| 2 | LINKIM | B'0 | R/W | LINK Interrupt Mask Setting this bit to 1 selects interrupt generation on setting of the Link signal change bit (ECSR.LCHNG) in the E-MAC status register to 1. 0: Interrupts on setting of the LCHNG bit is disabled. 1: Interrupts on setting of the LCHNG bit is enabled. |
| 1 | MPDIP | B'0 | R/W | Magic Packet™ Detect Interrupt Enable Setting this bit to 1 selects interrupt generation on setting of the Magic Packet™ detection bit (ECSR.MPD) in the E-MAC status register to 1. 0: Interrupts on setting of the MPD bit is disabled. 1: Interrupts on setting of the MPD bit is enabled. |
| 0 | ICDIP | B'0 | R/W | Illegal Carrier Detect Interrupt Enable Setting this bit to 1 selects interrupt generation on setting of the illegal carrier detection bit (ECSR.ICD) in the E-MAC status register to 1. 0: Interrupts on setting of the ICD bit is disabled. 1: Interrupt on setting of the ICD bit is enabled. |

46.2.88 PHY Interface Register (PIR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The PIR register provides a means of access to the PHY-LSI internal registers via the MII.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | MDI | MDO | MMD | MDC |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 4 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 3 | MDI | — | R/W | MII Management Data-In This bit indicates the level of the AVB_MDIO pin. |
| 2 | MDO | B'0 | R/W | MII Management Data-Out This bit holds data for output from the AVB_MDIO pin. The AVB_MDIO pin outputs data when the MMD bit is set to 1 (to specify writing as the direction). Data are not output while the MMD bit is set to 0 (to specify reading as the direction). |
| 1 | MMD | B'0 | R/W | MII Management Mode This bit specifies the direction for data through MDIO (reading or writing). 0: Read direction is specified. 1: Write direction is specified. |
| 0 | MDC | B'0 | R/W | MII Management Data Clock Values set in this bit are output on the AVB_MDC pin to supply the MII with the management data clock. For the method of access to the MII registers, see section 46.3.13 Connection to PHY-LSI. |

46.2.89 PHY Status Register (PSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The PSR register is a read-only register that can read interface signals from the PHY-LSI.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | LMON |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are read as 0. |
| 0 | LMON | B'0 | R | Link Status Pin State The Link state can be read by connecting the Link signal output from the PHY-LSI to the link status pin (AVB_LNK) pin. For the active sense, refer to the specifications of the PHY-LSI to be connected. 0: The link status signal (AVB_LNK) is at the low level. 1: The link status signal (AVB_LNK) is at the high level. |

46.2.90 PHY_INT Polarity Register (PIPR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The PIPR register is used to set the active sense of the AVB_PHY-INT pin.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | PHYIP |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

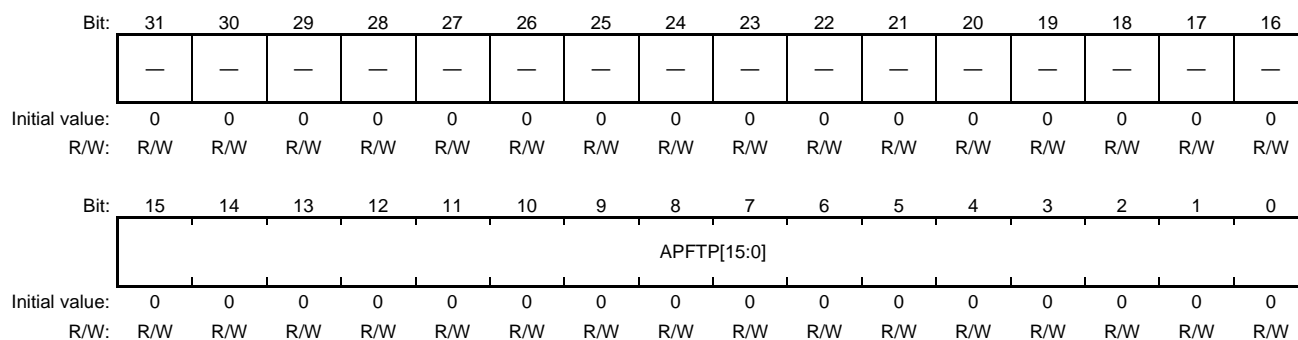
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 0 | PHYIP | B'0 | R/W | PHY Interrupt Input Pin Polarity This bit sets the active sense of the PHY interrupt pin (AVB_PHY-INT). For the active sense, refer to the specifications of the PHY-LSI to be connected. 0: PHY interrupt pin (AVB_PHY-INT) is active low (the low level triggers the interrupt state) 1: PHY interrupt pin (AVB_PHY-INT) is active high (the high level triggers the interrupt state) |

46.2.91 Automatic PAUSE frame register (APR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The APR register is used to set the value for the TIME parameter of auto generated PAUSE frames.

When ECMR.TXF = b'1, the value shouldn't set all 0 in this register.

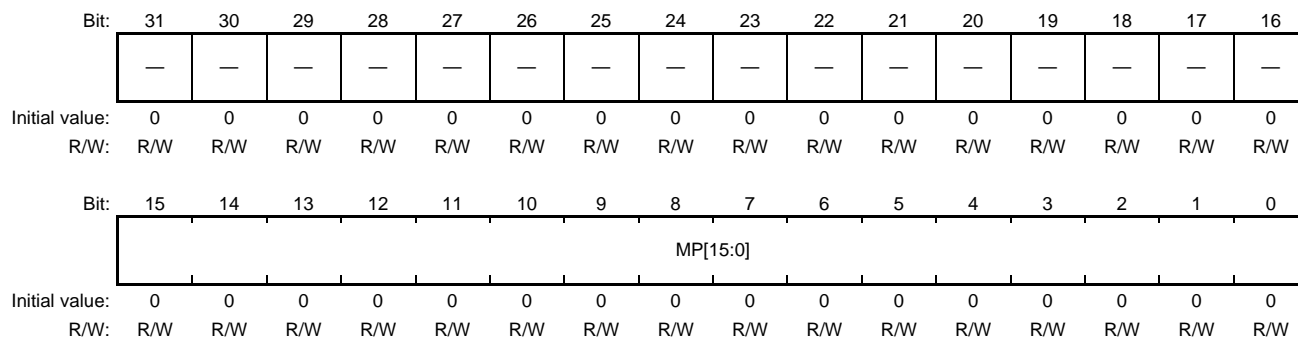


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 to 0 | APFTP[15:0] | H'0000 | R/W | Auto Pause Frame Time Parameter When a PAUSE frame is auto transmitted, the value set in this register is used as its TIME parameter. H'0000: — H'0001: 1 × 512 bit time H'0002: 2 × 512 bit time : : H'FFFF: 65535 × 512 bit time A bit time changes relative to the transfer speed as follows. 1000 Mbps: 1 bit time = 1 ns 100 Mbps: 1 bit time = 10 ns |

46.2.92 Manual PAUSE Frame Register (MPR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The MPR register is used to set the value for the TIME parameter of manually generated PAUSE frames. When a PAUSE frame is manually transmitted, the value set in this register is used as its TIME parameter.

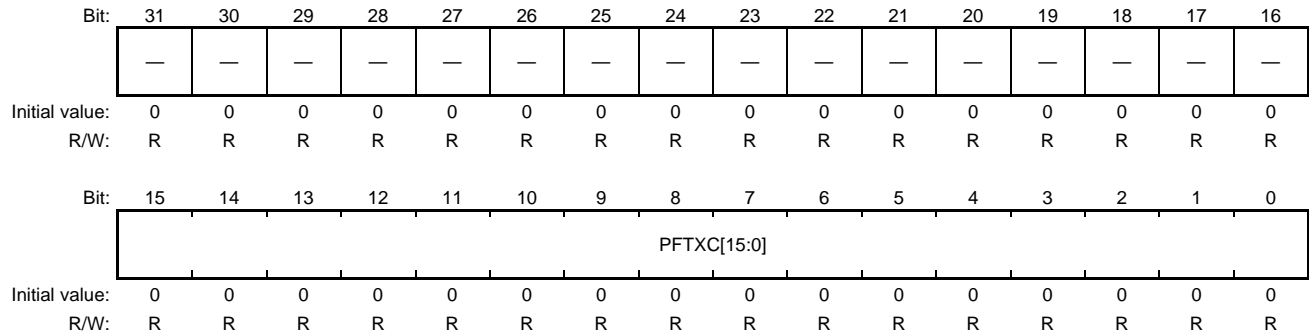


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 to 0 | MP[15:0] | H'0000 | R/W | Manual PAUSE These bits set the value of the TIME parameter in manually generated PAUSE frames. The unit for the setting is 512 bit time. These bits set the TIME parameter value of a manual PAUSE frame. H'0000: — H'0001: 1 × 512 bit time H'0002: 2 × 512 bit time : : H'FFFF: 65535 × 512 bit time A bit time changes relative to the transfer speed as follows. 1000 Mbps: 1 bit time = 1 ns 100 Mbps: 1 bit time = 10 ns |

46.2.93 PAUSE Frame Transmit Counter (PFTCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The PFTCR register is a counter that indicates the number of times PAUSE frames have been transmitted.

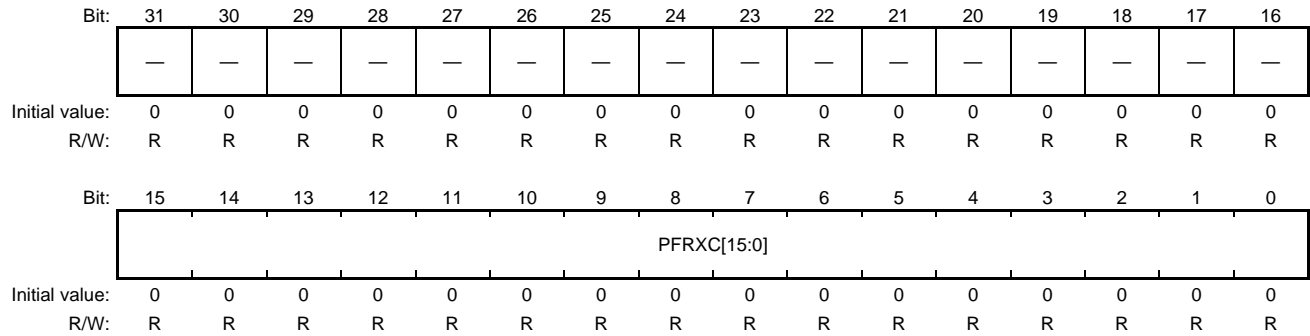


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are read as 0. The write value should be 0. |
| 15 to 0 | PFTXC[15:0] | H'0000 | R | PAUSE Frame Transmit Counter Counter for counting the number of transmitted PAUSE frames The bits are cleared to 0 when they are read. If counting up and clearing of the counter coincide, clearing the counter takes priority. |

46.2.94 PAUSE Frame Receive Counter (PFRCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RFRCCR register is a counter that indicates the number of times PAUSE frames have been received.



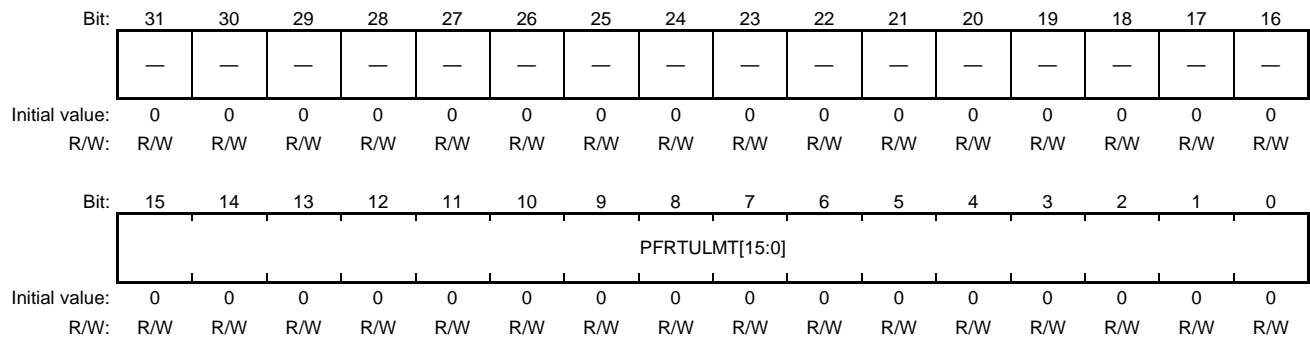
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are read as 0. The write value should be 0. |
| 15 to 0 | PFRXC[15:0] | H'0000 | R | PAUSE Frame Receive Counter These bits indicate the number of PAUSE frames that have been received when flow control in reception is enabled (the RXF bit in ECMR = 1). The bits are cleared to 0 when they are read. If counting up and clearing the counter coincide, clearing the counter takes priority. |

46.2.95 Automatic PAUSE frame retransmit count register (TPAUSER)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The TPAUSER register is used to set the value for upper limit number of auto generated PAUSE frames.

The setting in this register must not be changed while transmission is enabled.

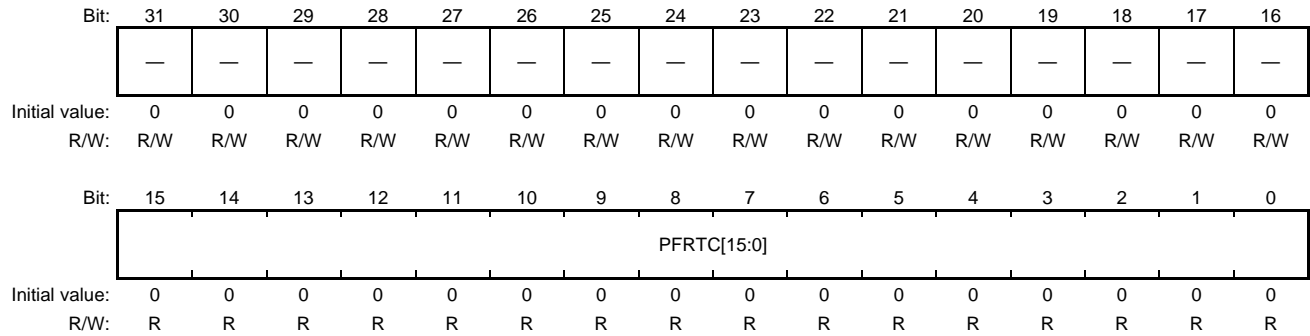


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 to 0 | PFRTULMT [15:0] | H'0000 | R/W | Pause Frame Retry Upper LiMiT These bit's set the value for upper limit time of auto generated PAUSE frames while Receive FIFO caution. H'0000: unlimited H'0001: 1 time H'0002: 2 times : : H'FFFF: 65535 times |

46.2.96 PAUSE frame transmit times counter (PFTTCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The PFTTCR register is a counter that indicates the number of times auto PAUSE frames have been transmit.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 to 0 | PFRTC[15:0] | H'0000 | R | Pause Frame Retry Counter These bits indicate the number of auto PAUSE frames that have been transmit while Receive FIFO caution. The bits are cleared to 0 when they are read and Receive FIFO caution start. |

46.2.97 E-MAC Mode Register 2 (GECMR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The GECMR register specifies the operating mode for the E-MAC.

The setting in the GECMR register must not be changed while transmission or reception is enabled.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SPEED |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

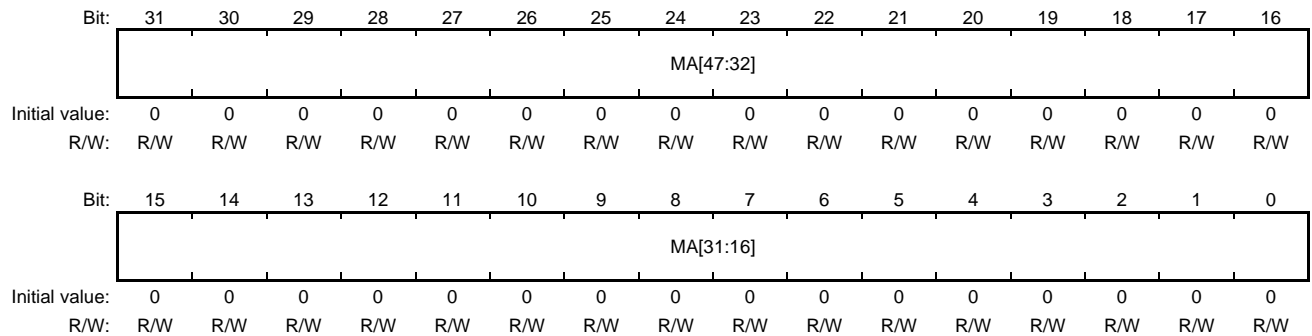
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 0 | SPEED | B'0 | R/W | Transfer Speed Setting This bit sets the transfer rate. 0: Transfer is at 100 Mbps. 1: Transfer is at 1000 Mbps. |

46.2.98 E-MAC Address High Register (MAHR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The MAHR register specifies the 32 higher-order bits of the 48-bit E-MAC address. The settings in this register are normally made in the initialization process after a reset.

The settings in this register must not be changed while transmission or reception is enabled.



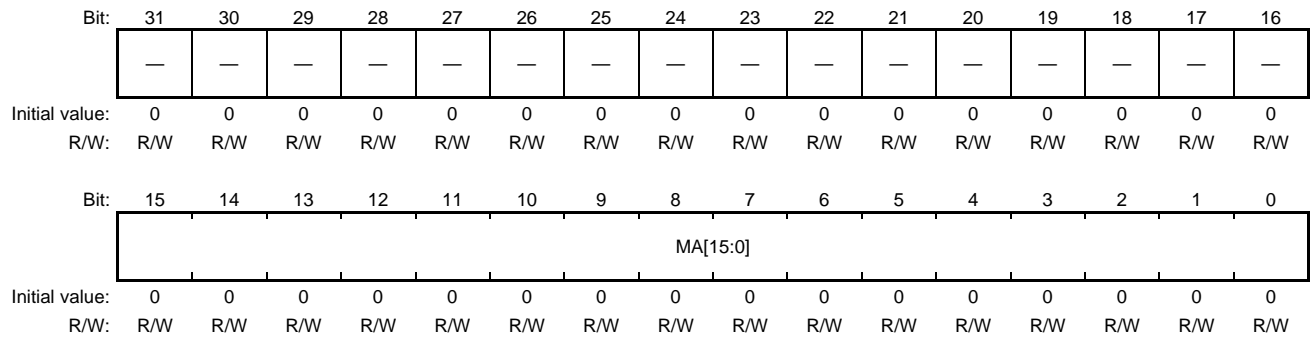
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | MA[47:16] | H'0000_0000 | R/W | E-MAC Address Bits 47 to 16 These bits are used to set the 32 higher-order bits of the E-MAC address. For example, if the E-MAC address is 01-23-45-67-89-AB (hexadecimal), set H'0123_4567 in the MAHR register. |

46.2.99 E-MAC Address Low Register (MALR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The MALR register specifies the 16 lower-order bits of the 48-bit E-MAC address. The settings in this register are normally made in the initialization process after a reset.

The settings in this register must not be changed while transmission or reception is enabled.

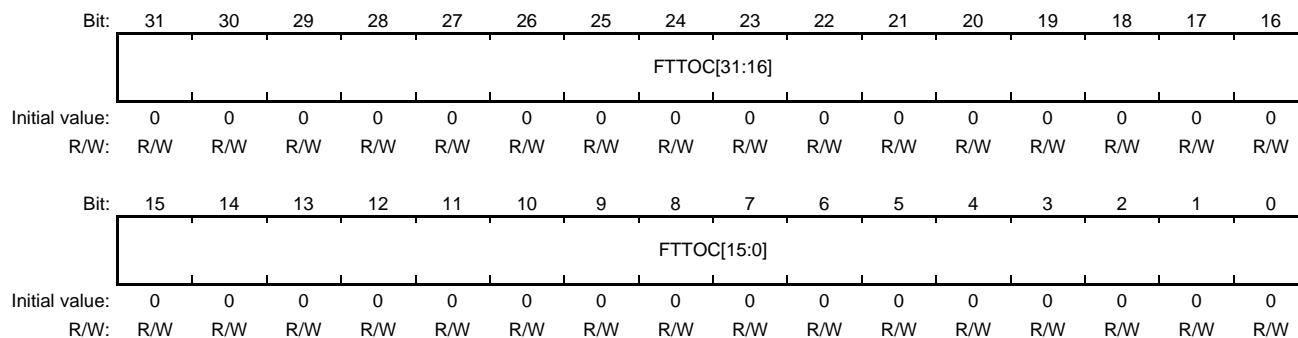


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 to 0 | MA[15:0] | H'0000 | R/W | E-MAC Address Bits 15 to 0 These bits are used to set the 16 lower-order bits of the E-MAC address. For example, if the E-MAC address is 01-23-45-67-89-AB (hexadecimal), set H'89AB in the MALR register. |

46.2.100 Transmit retry over counter register (TROCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The TROCR register is a counter that indicates the number of times frames with time-out were transmit. Counting up stops when the value in this register reaches H'FFFF_FFFF.

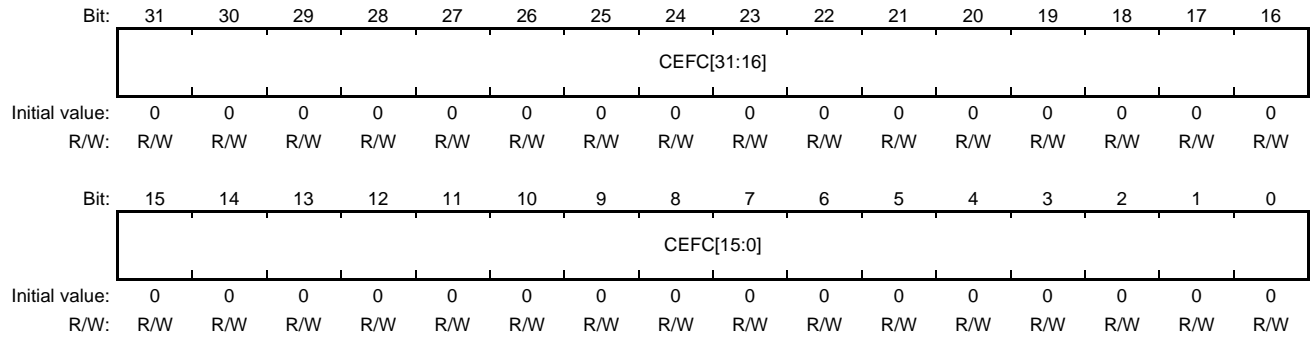


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|---|
| 31 to 16 | FTTOC[31:16] | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 to 0 | FTTOC [15:0] | H'0000 | R/W | Frame transmit time-out counter These bits indicate the number of transmit frames having time-out. The bits are cleared to 0 when they are read while the counter clear mode bit (TRCCM) in the E-MAC mode register is set to 1. When TRCCM = 0, they are cleared to 0 by the writing of any value to this register. |

46.2.101 CRC Error Frame Receive Counter Register (CEFCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CEFCR register is a counter that indicates the number of times frames with CRC errors were received. Counting up stops when the value in this register reaches H'FFFF_FFFF.

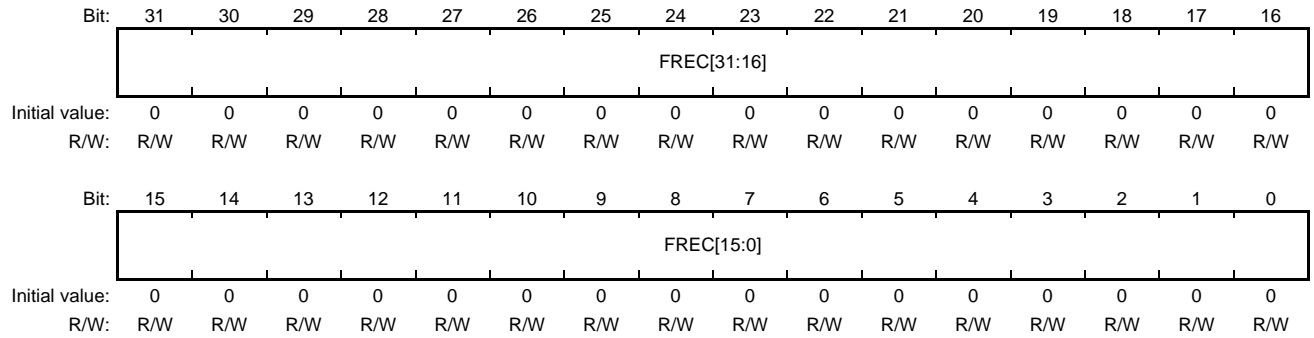


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 16 | CEFC[31:16] | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 to 0 | CEFC[15:0] | H'0000 | R/W | CRC Error Frame Counter These bits indicate the number of received frames having CRC errors. The bits are cleared to 0 when they are read while the counter clear mode bit (TRCCM) in the E-MAC mode register is set to 1. When TRCCM = 0, they are cleared to 0 by the writing of any value to this register. |

46.2.102 Frame Receive Error Counter Register (FRECR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The FRECR register is a counter that indicates the number of frames for which receive errors were generated by input on the RX_ER from the PHY-LSI. Counting up stops when the value in this register reaches H'FFFF_FFFF.

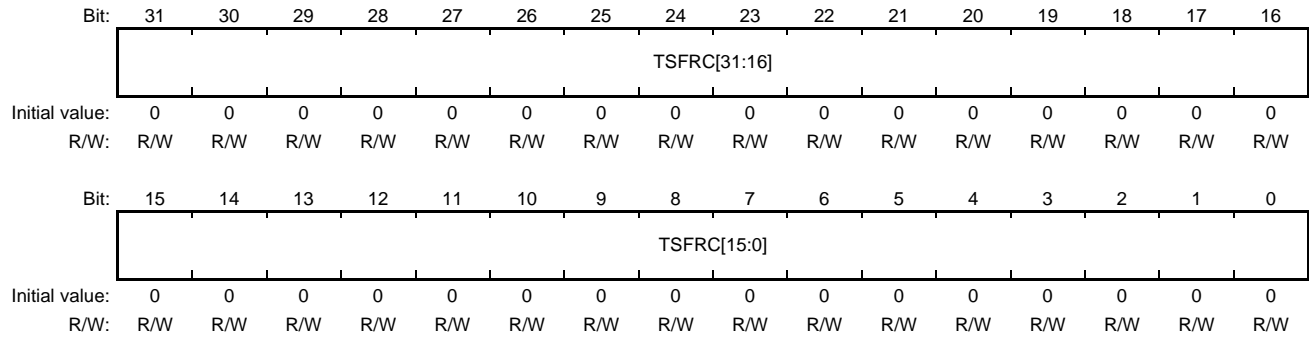


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 16 | FREC[31:16] | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 to 0 | FREC[15:0] | H'0000 | R/W | Frame Receive Error Counter These bits indicate the number of errors during frame reception. The bits are cleared to 0 when they are read while the counter clear mode bit (TRCCM) in the E-MAC mode register is set to 1. When TRCCM = 0, they are cleared to 0 by the writing of any value to this register. |

46.2.103 Too-Short Frame Receive Counter Register (TSFRCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The TSFRCR register is a counter that indicates the number of received frames that were fewer than 64 bytes in length. Counting stops when the value in this register reaches H'FFFF_FFFF.

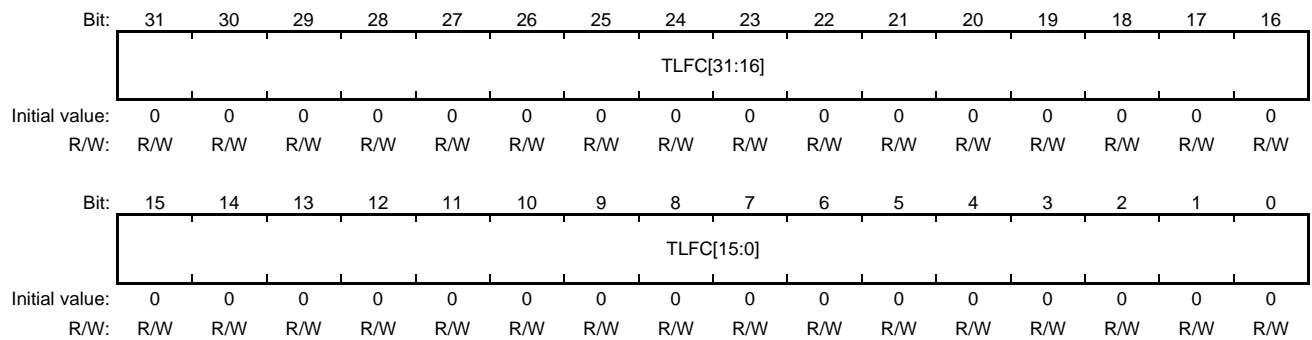


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|---|
| 31 to 16 | TSFRC[31:16] | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 to 0 | TSFRC[15:0] | H'0000 | R/W | Too-Short Frame Receive Counter These bits indicate the number of received frames that were fewer than 64 bytes in length. The bits are cleared to 0 when they are read while the counter clear mode bit (TRCCM) in the E-MAC mode register is set to 1. When TRCCM = 0, they are cleared to 0 by the writing of any value to this register. |

46.2.104 Too-Long Frame Receive Counter Register (TLFRCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The TLFRCR register is a counter that indicates the number of received frames that were longer than the value specified in the receive frame length register (RFLR). Counting up stops when the value in the TLFRCR register reaches H'FFFF_FFFF.

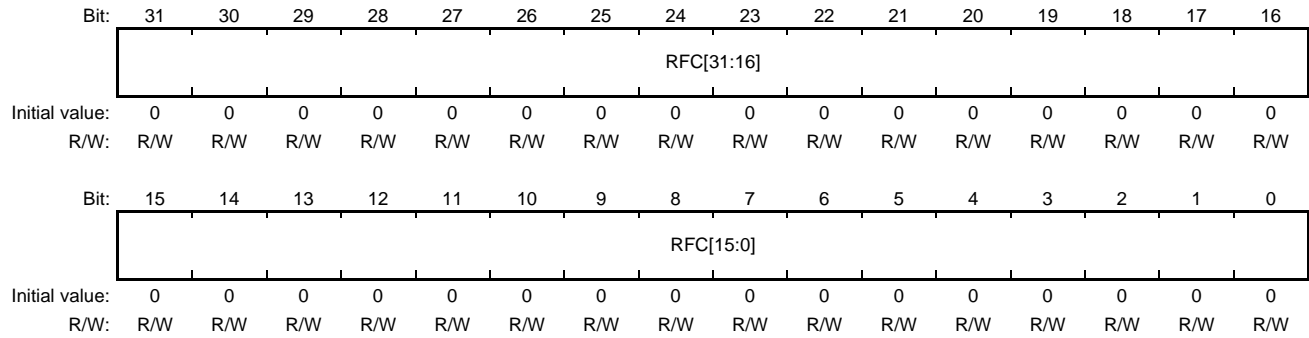


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 16 | TLFC[31:16] | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 to 0 | TLFC[15:0] | H'0000 | R/W | Too-Long Frame Receive Counter These bits indicate the number of received frames that were longer than the value in RFLR. The bits are cleared to 0 when they are read while the counter clear mode bit (TRCCM) in the E-MAC mode register is set to 1. When TRCCM = 0, they are cleared to 0 by the writing of any value to this register. |

46.2.105 Residual-Bit Frame Receive Counter Register (RFCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The RFCR register is a counter that indicates the number of received frames containing “residual bits” (trailing bits not making up an 8-bit unit). Counting up stops when the value in this register reaches H'FFFF_FFFF.

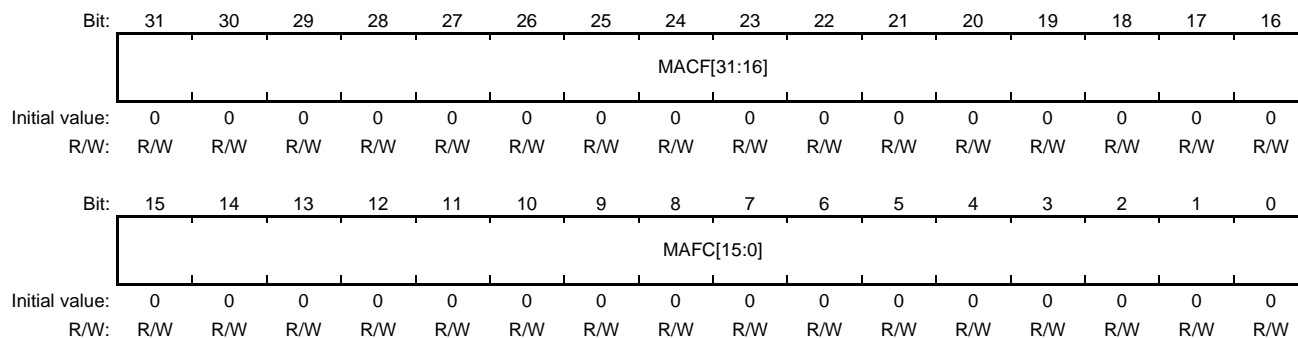


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 16 | RFC[31:16] | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 to 0 | RFC[15:0] | H'0000 | R/W | Residual-Bit Frame Receive Counter These bits indicate the number of received frames containing residual bits. The bits are cleared to 0 when they are read while the counter clear mode bit (TRCCM) in the E-MAC mode register is set to 1. When TRCCM = 0, they are cleared to 0 by the writing of any value to this register. |

46.2.106 Multicast Address Frame Receive Counter Register (MAFCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The MAFCR register is a counter that indicates the number of received frames for which a multicast address was specified. Counting up stops when the value in this register reaches H'FFFF_FFFF.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 16 | MACF[31:16] | All 0 | R/W | Reserved These bits are read as 0. The write value should be 0. |
| 15 to 0 | MAFC[15:0] | H'0000 | R/W | Multicast Address Frame Counter These bits indicate the number of multicast frames that have been received. The bits are cleared to 0 when they are read while the counter clear mode bit (TRCCM) in the E-MAC mode register is set to 1. When TRCCM = 0, they are cleared to 0 by the writing of any value to this register. |

46.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The EthernetAVB-IF consists of the following functional units:

- DMA transfer controller (AVB-DMAC): Handles DMA transfer between the data storage areas for reception and transmission in the URAM and the reception and transmission FIFO buffers
- MAC controller (E-MAC): Handles transfer between the reception and transmission FIFO buffers and the RGMII

Using its direct memory access (DMA) function, the AVB-DMAC handles DMA transfer of frame data between the destinations for storing Ethernet frame data for transmission and reception in the URAM and the FIFO buffers for reception and transmission. Data cannot be directly read from or written to the FIFO buffers.

To handle DMA transfer, the AVB-DMAC requires information that includes the addresses for storage of data for transmission and received data. These data are referred to as descriptors. The AVB-DMAC reads data for transmission from the storage area for data to be transmitted according to the information in descriptors and writes received data to the storage area for received data accompanied by information in descriptors. The descriptors are placed in the URAM. Arranging multiple descriptors in descriptor lists allows the continuous reception or transmission of multiple Ethernet frames.

The E-MAC supports a RGMII, which provides an interface format for the externally connected PHY-LSI. The E-MAC constructs Ethernet frames from data written to the transmission FIFO and transmits these frames to the RGMII. It also performs CRC checking of Ethernet frames received from the RGMII and writes the frames to the reception FIFO.

46.3.1 AVB-DMAC Operating Modes

Figure 46.2 illustrates the operating modes of the AVB-DMAC.

Transitions of AVB-DMAC operating mode are under the control of the items listed below.

- CPU operating mode (hardware reset and power-down mode)
- Configuration of the operating mode configuration bits (CCC.OPC) in the AVB-DMAC mode register

The current operating mode can be confirmed by reading the operating mode status bits in the AVB-DMAC status register (CSR.OPS).

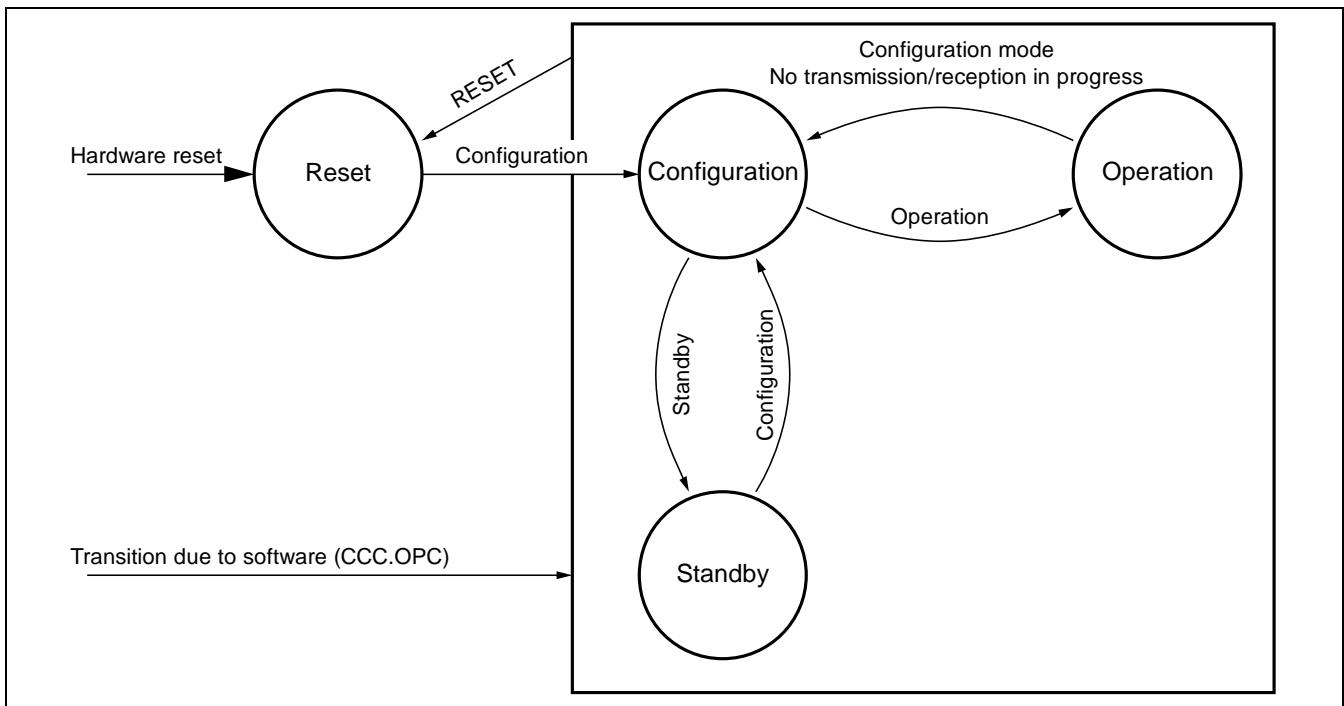


Figure 46.2 Operating Mode of AVB-DMAC

(1) Operating Modes

(a) Reset mode

After a hardware reset, the AVB-DMAC enters reset mode.

In reset mode, only the AVB-DMAC operating mode control function is controllable and other functions are all stopped. This mode is designed for reduced power when the Ethernet function is not necessary.

(b) Configuration mode

In configuration mode, various settings for the AVB-DMAC can be made.

The operation of most functions is stopped and all status registers are initialized to their reset values. The E-MAC functions in this mode.

By CCC.GAC it is possible to enable gPTP support already in CONFIG mode.

(c) Operation mode

In operation mode, all functions of the AVB-DMAC can operate. Ethernet communications can only proceed in this mode.

(d) Standby mode

In standby mode, the E-MAC can only be used to control the operating mode. Other functions cannot be used. When CCC.GAC is 1 CPU should not enter STANDBY mode.

(2) How to Set the Operating Mode

Set the operating mode configuration bits in the AVB-DMAC mode register (CCC.OPC) to select the operating mode. Furthermore, the current operating mode can be checked by reading the operating mode status bits in the AVB-DMAC status register (CSR.OPS).

Transitions other than from operation mode to configuration mode are made after the value is written to the operating mode configuration bits (CCC.OPC) (Figure 46.3).

For transitions from operation mode to configuration mode, follow the procedure in Figure 46.4 because any transmission and reception in progress will be executed before the transition to configuration mode.

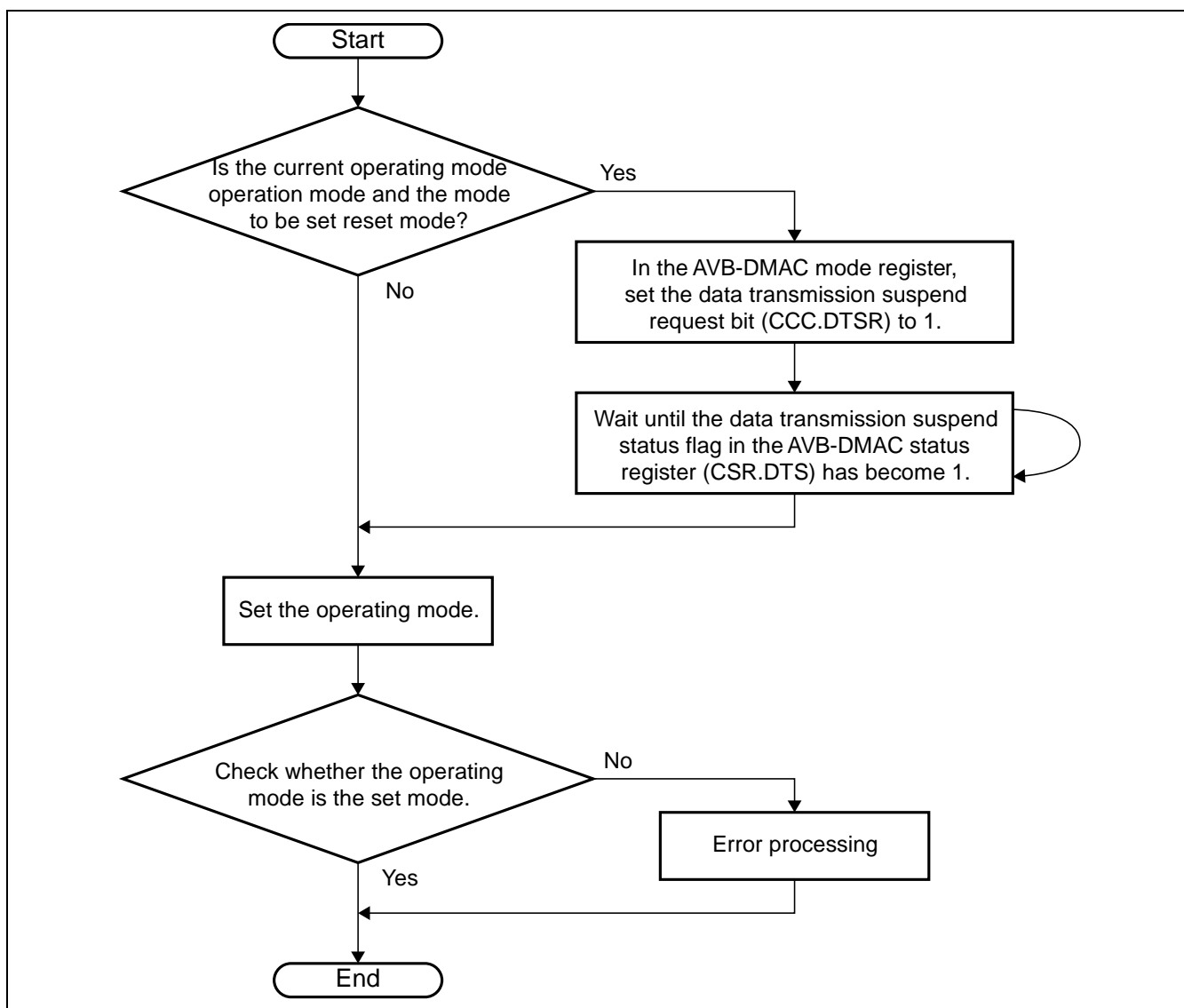
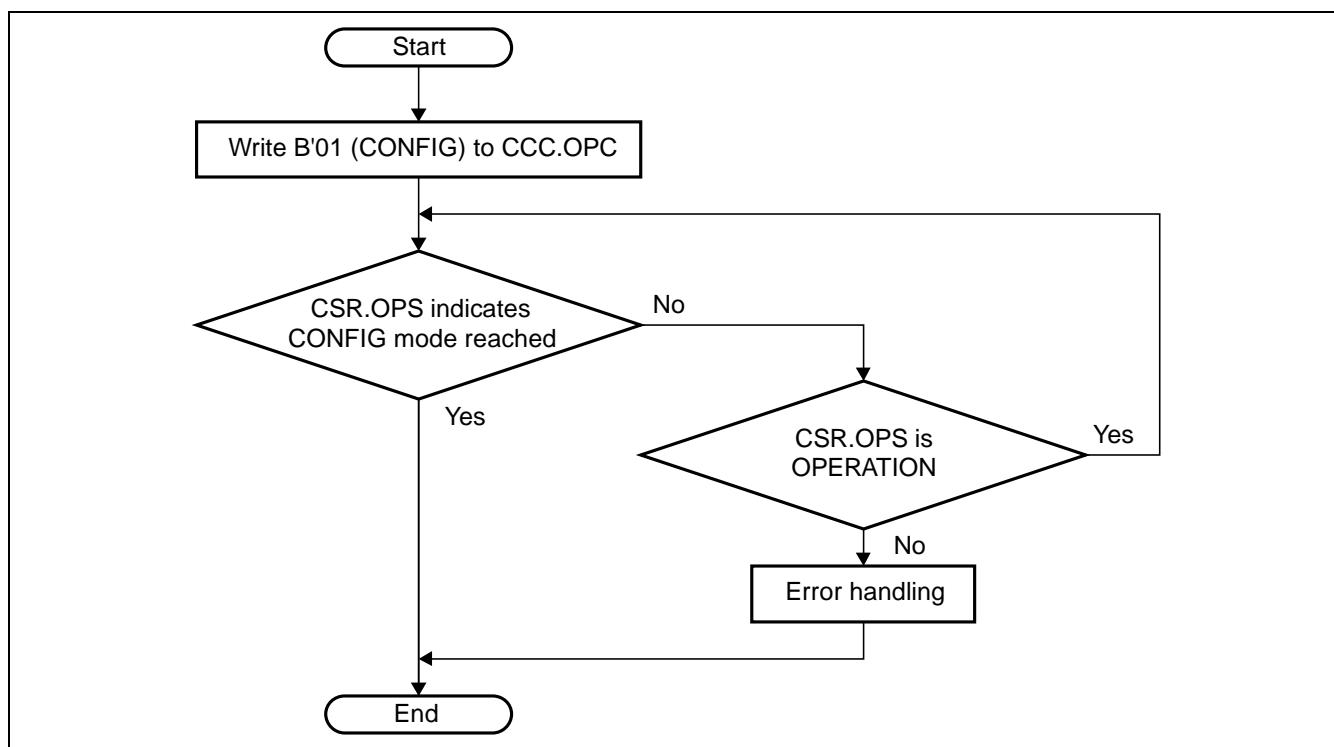


Figure 46.3 Flow for Transitions of Operating Mode (Other than from Operation Mode to Configuration Mode)



**Figure 46.4 Flow for Transitions of Operating Mode
(from Operation Mode to Configuration Mode)**

In the transition from operation mode to configuration mode, the AVB-DMAC executes the following operations before the transition is completed. Read the operating mode status bits in the AVB-DMAC status register (CSR.OPS) to check that the transition to configuration mode has been completed.

- If the transfer of a frame between the reception FIFO and URAM is in progress, this is completed (other received frames remaining in the FIFO and any frames that are subsequently received by the E-MAC are discarded).
- If the transfer of a frame is in progress between the transmission FIFO and URAM, this is completed (frames for transmission remaining in the URAM will not be transmitted).
- All frames for transmission in the transmission FIFO are transferred to the E-MAC.

The CPU should not disable URAM transfers by CCC.DTSR nor prevent transmission by MAC when requesting transition from OPERATION to CONFIG, else EthernetAVB-IF cannot perform requested mode change.

Notes: When the operating mode shifts to configuration mode, all status registers are cleared.

When need status register information, the reception and transmission path can be individually disabled and can check the status before leaving operating mode with follow.

We recommend following the procedure below in the case of this transition.

1. Disable reception.
2. Since reception actually stopping after being disabled requires time, wait for an interval equivalent to that for reception of a maximum length packet.
3. Stop the software task that is generating data for transmission.
4. Wait until the receive process status bit (CSR.RPO) and the transmit process status bits (CSR.TPO0 to 3) in the AVB-DMAC status register are set to 0.
5. Capture all of the required status information.
6. Set the operating mode configuration bits in the AVB-DMAC mode register (CCC.OPC) to initiate the transition to configuration mode.

(3) Leave RESET mode with additional configuration

When writing B'01 (CONFIG) to CCC.OPC while EthernetAVB-IF is in RESET mode, it is possible to configure availability of gPTP support within CONFIG mode

To enable gPTP support in CONFIG mode use a 32 bit write access when setting CCC.OPC to B'01. This write should also set CCC.GAC to 1 and CCC.CSEL to intended configuration.

(4) Operating Mode Transitions Due to Hardware

The following hardware factors can also initiate transitions of the AVB-DMAC operating mode.

(a) Hardware reset

Resetting of the LSI chip leads to resetting of the entire EthernetAVB-IF module. The operating mode shifts to reset mode.

(b) Transition during power-off by module standby

This transition is triggered by module standby of CPG during the power-off sequence.

The AVB-DMAC completes the bus master access in progress, and then shifts to reset mode. At this time, the operating mode configuration bits in the AVB-DMAC mode register (CCC.OPC) are set to B'00.

46.3.2 Common Control for Transmission and Reception

(1) Initialization Procedure

Figure 46.5 shows the overall initialization procedure in outline.

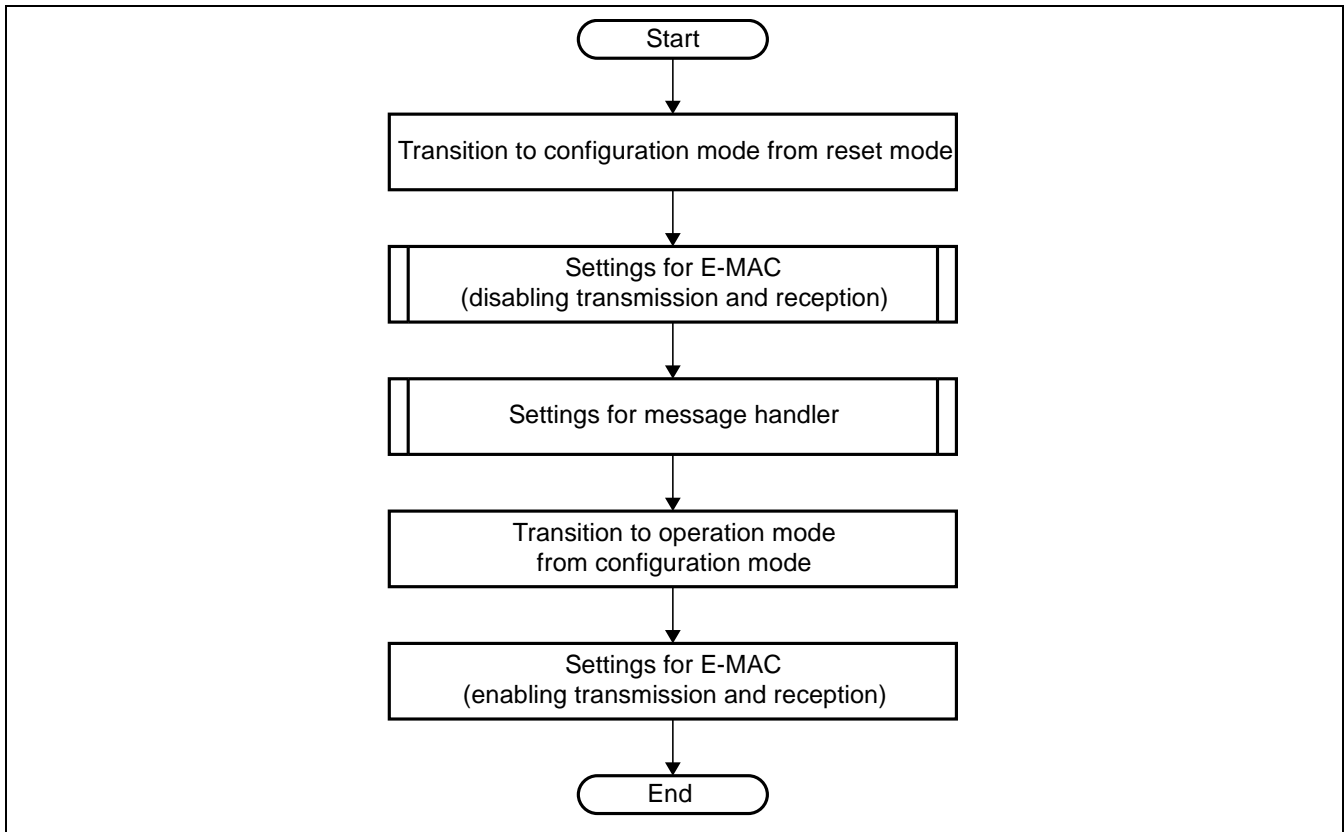


Figure 46.5 Outline of the Initialization Procedure

Note: Before starting configuration, all required clocks should be configured and enabled.

(a) Initializing the Receiver Section

Before starting reception, follow the procedure below.

Keep the operating mode to configuration mode, and do not enable reception until the settings for the AVB-DMAC are completed.

- Set the operating mode to configuration mode.
- Set AVB filtering for network control frames and AVB stream frames to suit the specifications of the product the chip will be used in.
- Create a descriptor chain for each queue to be used.
- Set the base address for table address in the descriptor base address table register (DBAT).
- Specify the maximum frame length with the receive frame length upper limit register (RFLR).
- Specify whether padding is to be used with the receive padding configuration register (RPC).
- Set the unread frame counter for each queue with unread frame counter registers (UFCVs) 0 to 4.

(b) Initializing the Transmitter Section

Figure 46.6 illustrates initialization of the transmitter section.

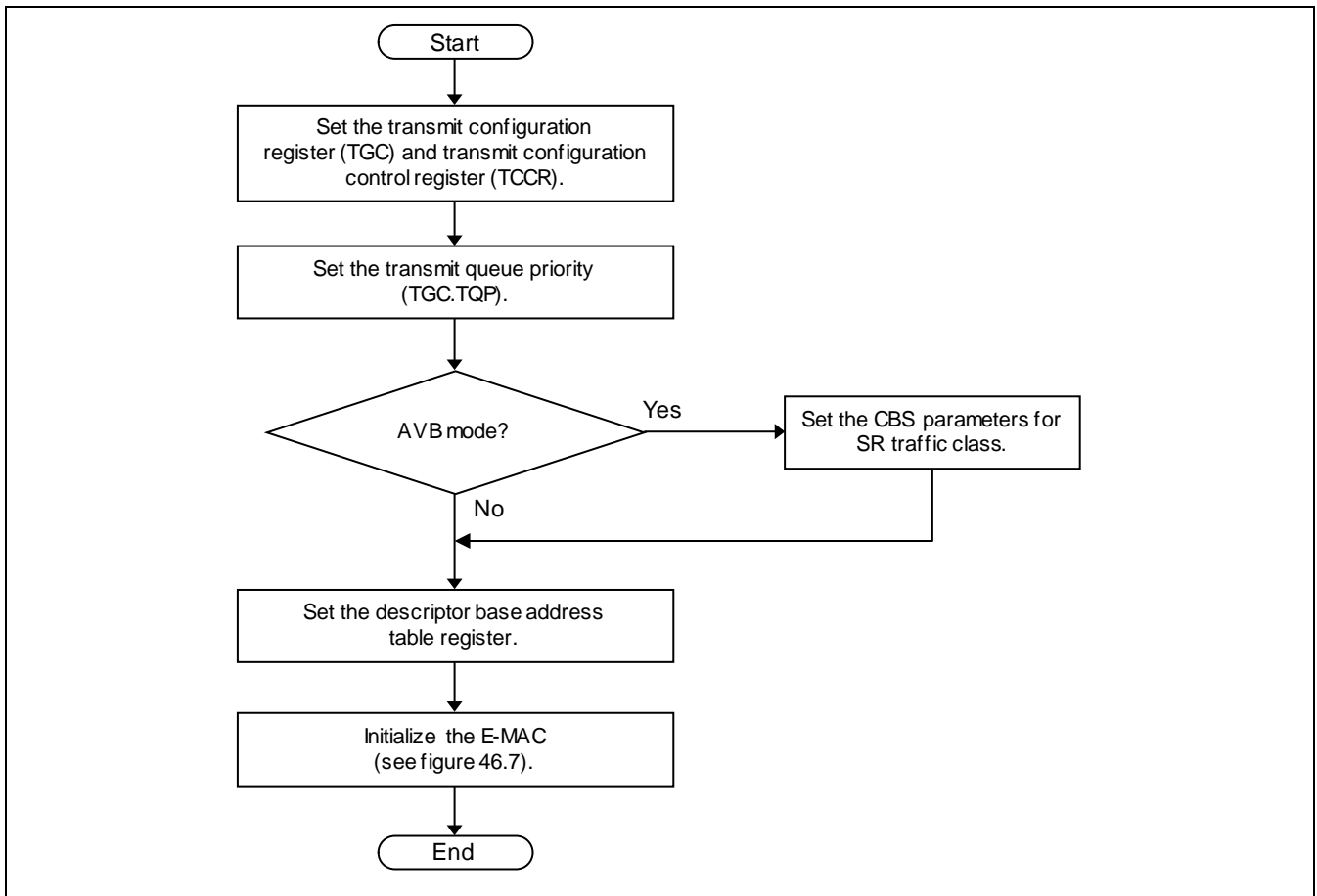


Figure 46.6 Procedure for Initializing the Transmitter Section

(c) Initializing the E-MAC Section

Figure 46.7 illustrates initialization of the E-MAC section.

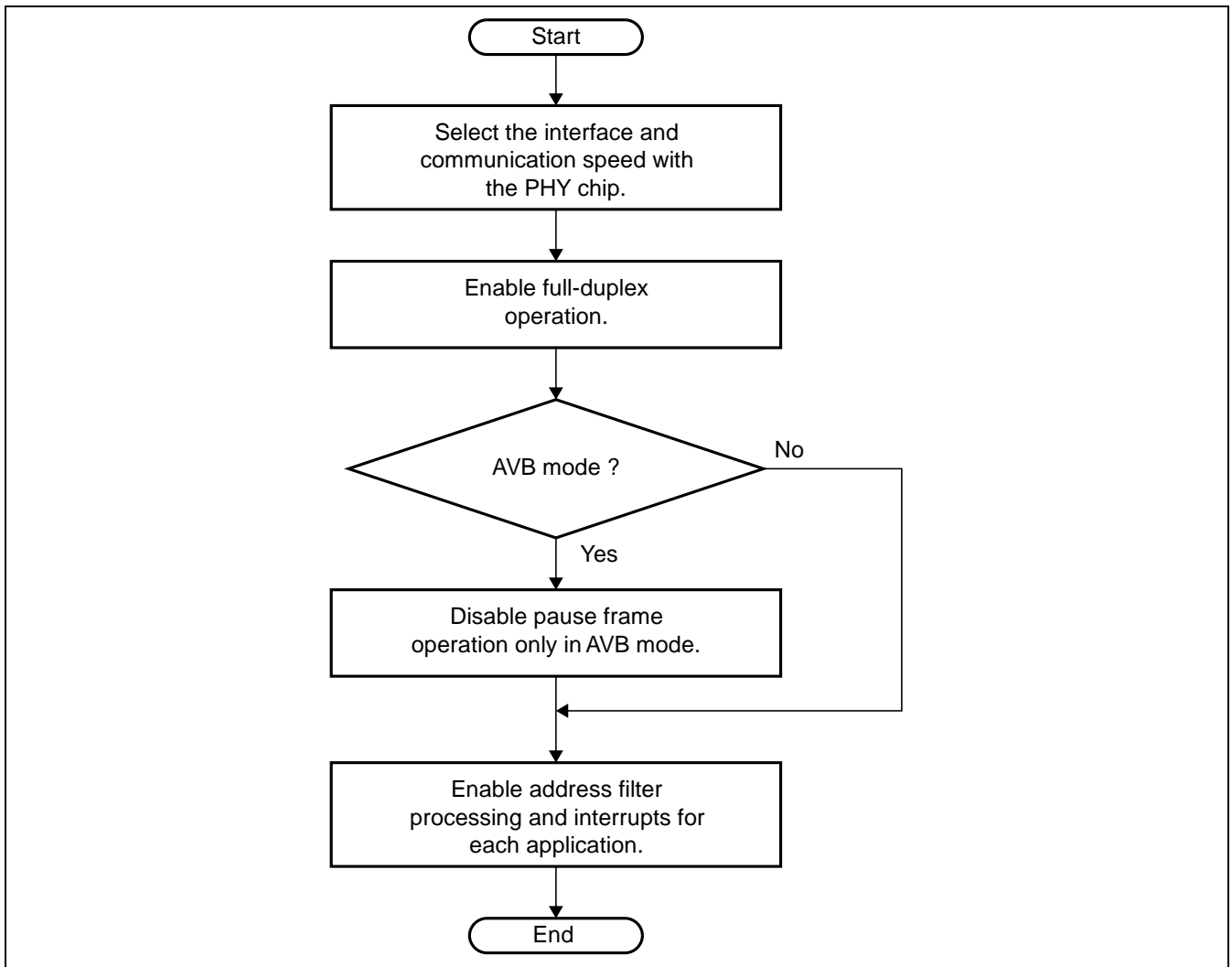


Figure 46.7 Procedure for Initializing the E-MAC Section

(d) Initialization of the Application Unit

Figure 46.8 illustrates initialization of the application unit.

For a description of how to set up the descriptors and the CBS traffic shaping parameters, see section 46.3.3 Descriptors, and section 46.3.6 CBS (Credit-Based Shaping).

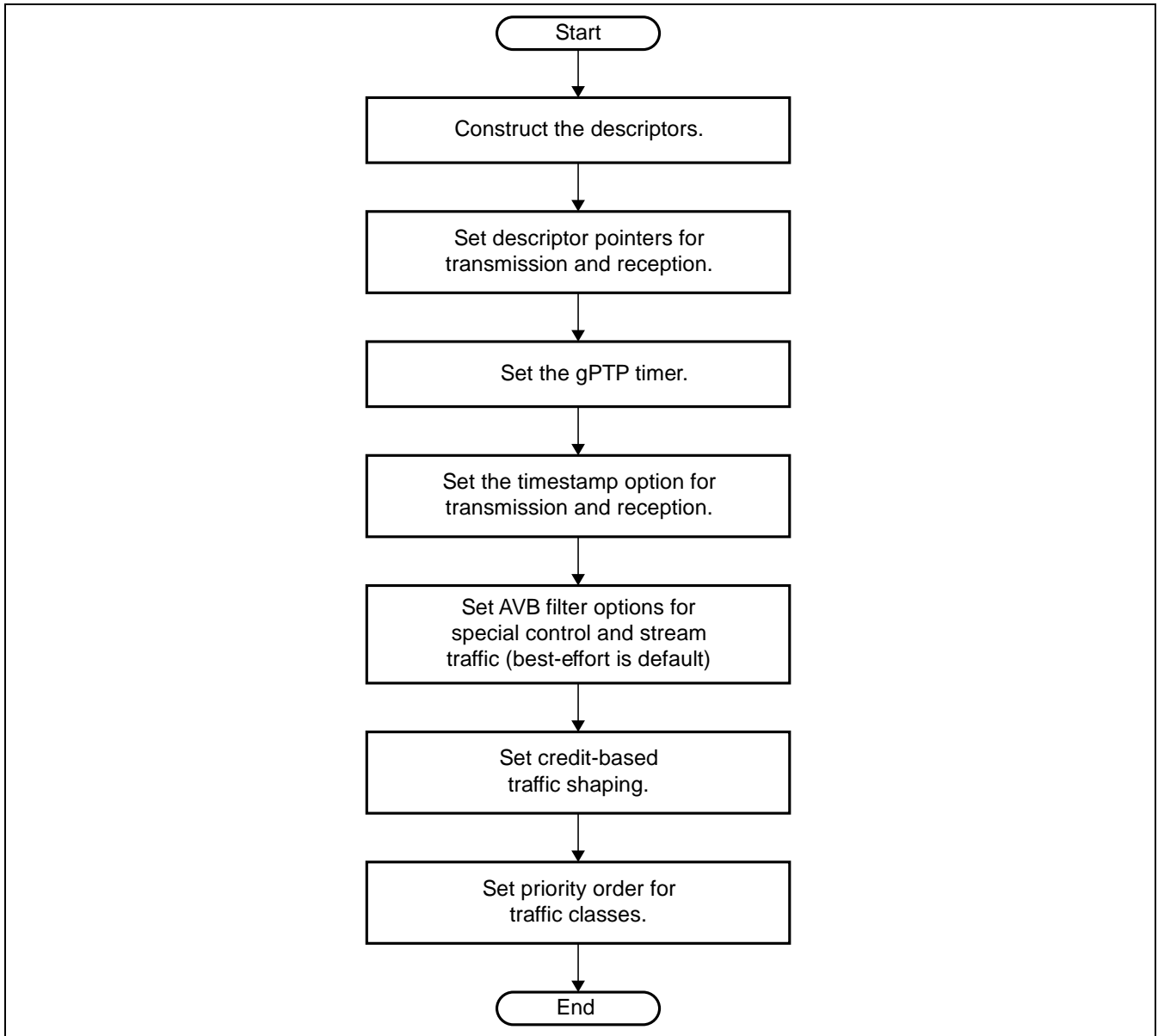


Figure 46.8 Initializing the Sections for Use by the Application

(e) Relationship between Transmission Queue Numbers and Traffic Classes

In fetching, the relationships between the transmission queues and traffic classes are fixed, so the priority specified by the transmit queue priority bits in the transmit configuration register (TGC.TQP) has no effect.

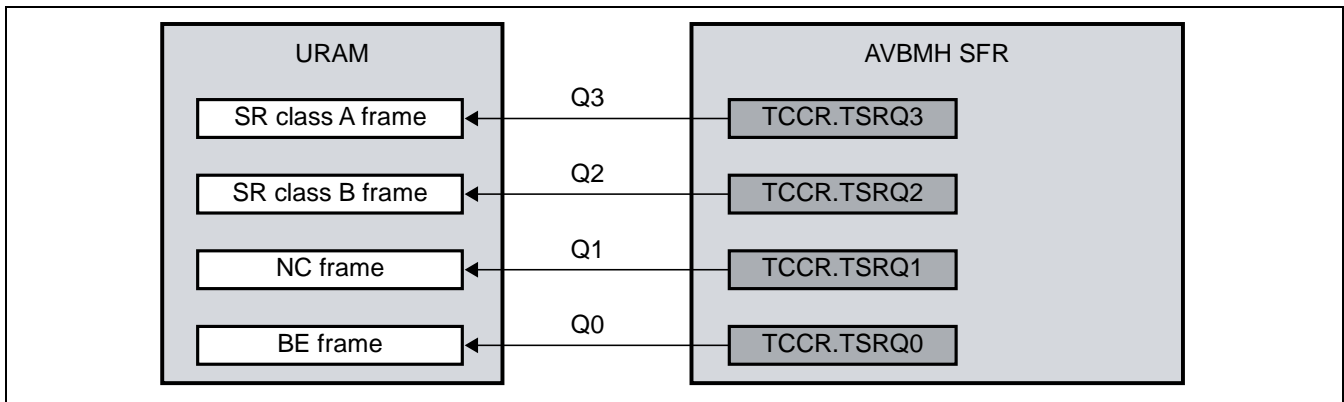


Figure 46.9 Class Associations of Queues for the Scheduler

In fetching, the credit values for stream classes A and B are not taken into account. Behavior depends on the setting of the transfer FIFO size configuration bits in the transfer configuration registers (TGC.TBDt) and on the frame size that can be fetched to the transmission FIFO.

When the transmit queue priority bits in the transmit configuration register (TGC.TQP) are B'00 or B'01, the priority order is Q3 → Q2 → Q1 → Q0.

When the transmit queue priority bits in the transmit configuration register (TGC.TQP) are B'11, the priority order is Q1 → Q3 → Q2 → Q0.

(2) Checking Integrity

The AVB-DMAC is capable of detecting and identifying errors produced in the processing of Ethernet frames and in the transfer of frame data for transmission and reception.

(a) Concept of Integrity Checking in Reception

The aim of integrity checking in reception is preventing the storage of error frames in the URAM. If an error frame is stored in the URAM, software can get information to identify the frame as an error frame.

Note: If a special descriptor chain is to be used for separation of received headers from the associated data, an error that breaks the sequence may lead to storage space for synchronization running out. In such cases, software interaction or re-synchronization via the EOS descriptor is required.

(b) Concept of Integrity Checking in Transmission

The purpose of integrity checking in transmission is to prevent the transmission of broken frames.

Since transmission of a frame by the E-MAC can neither be stopped nor disabled once it has started, this check involves intensive monitoring for problems that can arise during fetching.

(c) Items for Monitoring in Both Reception and Transmission

Errors in access to the URAM for reading of descriptors

The same descriptor may be processed again because the current descriptor address (CDARq.CDA) was not changed. If this problem occurs in a divided frame, the sequence may be broken.

In reception

- The received frame will be lost.
- The same problem will occur for the next frame of data received for the same queue.

In transmission

- The transmit start request bit in the transmit configuration control register (TCCR.TSRQt) is set to 0.
- The frame will be lost from the transmission FIFO.

Errors in access to read descriptors from the URAM are detected from the response signal of the AXI-Bus. EthernetAVB-IF flags only one access error per descriptor read.

Illegal configuration of a descriptor by an application

The same descriptor may be processed again because the current descriptor address (CDARq.CDA) was not changed. If this problem occurs in a divided frame, the sequence may be broken.

In reception

- The received frame will be lost.
- The same problem will occur for the next frame of data received for the same queue.

In transmission

- The transmit start request bit in the transmit configuration control register (TCCR.TSRQt) is set to 0.
- The frame will be lost from the transmission FIFO.

Errors in access to the URAM for writing of descriptors

As in the case where no error occurs, the current descriptor address (CDARq.CDA) and the transmit start request bit in the transmit configuration control register (TCCR.TSRQt) are updated.

As DESC.DT was not updated, hardware and software synchronization may have been destroyed.

Errors in access to write descriptors to the URAM are detected from the response signal of the AXI-Bus.

(d) Items for Monitoring in Reception

Errors in access to the URAM for writing of data or time stamps

- As in the case where no error occurs, the current descriptor address (CDARq.CDA) is updated.
- DESCR.EI is set to indicate incorrect contents.
- This problem occurring in a divided frame may break the descriptor sequence, making the queue unusable.

Errors in access to write data or descriptors to the URAM are detected from the response signal of the AXI-Bus. EthernetAVB-IF flags each access error individually, even they are related to same frame.

Error of the Reception FIFO

- Received frames are all invalidated.
- All frames stored as received frames are discarded. At this time, the number of frames and queue information cannot be captured.

If the reception FIFO RAM faults, AVB-DMAC detects the fault as error of the reception FIFO.

Queue Synchronisation Fault of the Reception

EthernetAVB-IF will not continue splitting frame across EOS descriptor. A queue synchronisation fault is detected when read descriptor with DESCR.DT = EOS at position where FMID, or FEND should be written

- RIS2.QFFr is set 1
- CDARq.CDA is updated as in fault free operation.
- Neither CIARr.CIA nor LIARr.LIA is changed.
- Neither UFCVi.CVr nor RIS0.FRr is changed.
- Received frame is lost. There is an incomplete frame (no FEND) in descriptor chain.

Note: Reading descriptor may be one or more storage elements ahead in descriptor chain.

(e) Items for Monitoring in Transmission

Errors in Access for Reading Data from the URAM

- Data that have already been fetched are discarded from the transmission FIFO.
- When an error of this type occurs during processing of an FSINGLE or FEND descriptor:
As in the case where no error occurs, the current descriptor address (CDARq.CDA) and the transmit start bit in the transmit configuration control register (TCCR.TSRQt) are updated. Fetching resumes after the error frame.
- When an error of this type occurs during processing of an FSTART or FMID descriptor:
— The current descriptor address (CDARq.CDA) is not updated.
— The transmit start bit in the transmit configuration control register (TCCR.TSRQt) is set to 0.

Errors in access to read data from the URAM are detected from the response signal of the AXI-Bus.

Overflow of the Transmission FIFO

- As in the case where no error occurs, the current descriptor address (CDARq.CDA) and the transmit start bit in the transmit configuration control register (TCCR.TSRQt) are updated. Fetching resumes after the error frame.
- The frame will be discarded from the FIFO.

Frame size error during transmission

- As in the case where no error occurs, the current descriptor address (CDARq.CDA) and the transmit start bit in the transmit configuration control register (TCCR.TSRQt) are updated. Fetching resumes after the error frame.

A transmit frame size error is detected when the size setting in one or more (in the case of a divided frame) descriptors for frame transmission is 1966 or more bytes. Such frames are cut out and transmitted.

Damaged Data in the Transmission FIFO

- Fetching is not affected by damaged data.
- Since damaged data from the FIFO is only detected during frame transmission, an error frame may be transmitted.

If damaged data in the transmission FIFO is an error due to the transmission FIFO, this is detected by the AVB-DMAC.

(f) Items for Monitoring in gPTP**Access error while reading AVTP timestamps from Tx-Buffer RAM**

- AVTP timestamps are invalidated.
- gPTP unit ignores the issue. It uses the value as read for AVTP comparison.

An access error when reading timestamps from Tx-Buffer RAM is flagged to EthernetAVB-IF by external RAM protection logic.

46.3.3 Descriptors

(1) Data Representation in URAM

The AVB-DMAC transfers data for transmission and received data to and from the application software via the URAM.

The memory in the URAM for use by the AVB-DMAC is configured with control structures referred to as descriptors and associated areas to which the frame data are allocated. Dividing the memory into a control area and data area allows the flexible allocation of frame data to the URAM. This enables sharing of the areas to which frame data are allocated and the use of non-contiguous areas. Frame data can be copied without using the CPU. Arbitration that ensures hardware and software access to the memory area is also available without access to registers of the AVB-DMAC.

Figure 46.10 shows an example of the memory maps for descriptors and the descriptor data area in the URAM.

A descriptor consists of its type (DESCR.DT), which controls the descriptor functions, a descriptor pointer (DESCR.DPTR) indicating the start address for storage of the frame data in the descriptor area, and the data size field (DESCR.DS), indicating the amount of frame data. Post-processing interrupt generation can be set up for each descriptor. Enabling and disabling of the interrupt is controlled by the descriptor interrupt enable bits (DESCR.DIE).

The descriptor may also hold information related to content. This information does not affect general descriptor functions. It provides information other than the frame data proper, such as on the state of reception.

For details, see section 46.3.4(2) Setting Up Reception Descriptors, and section 46.3.5(2) Setting Up Transmission Descriptors.

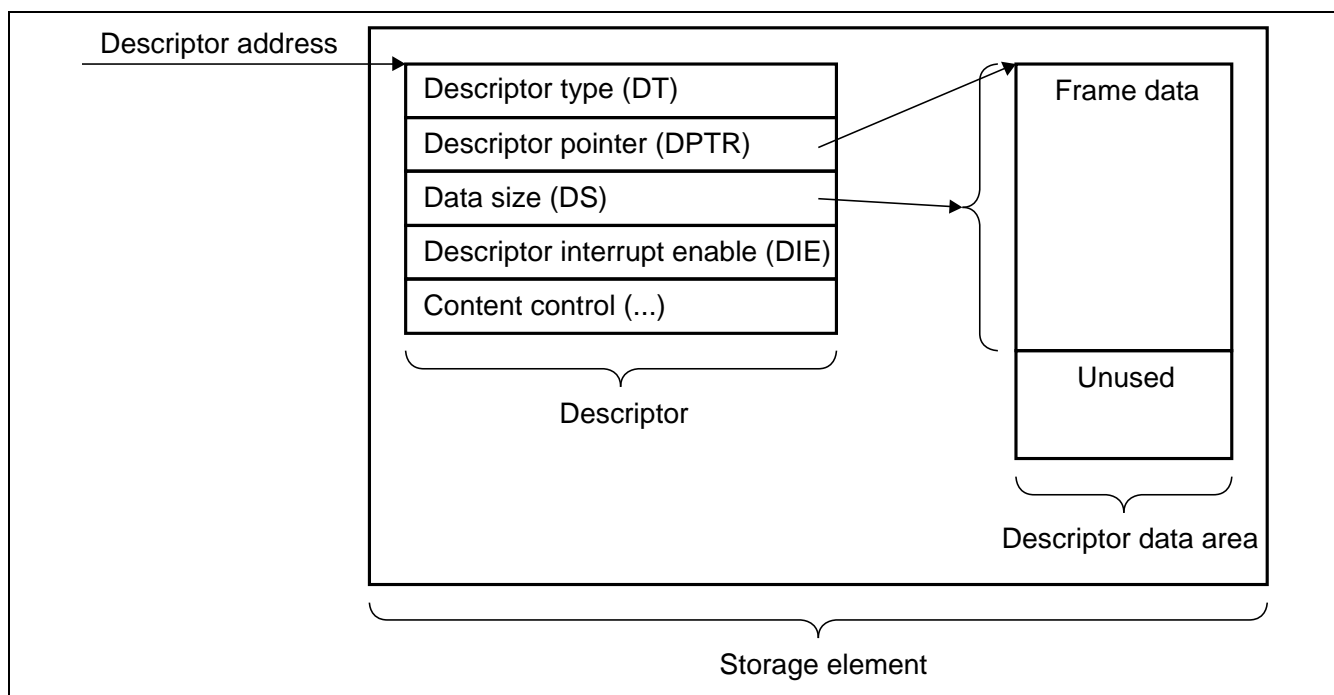


Figure 46.10 Outline of Storage Element Used for Receive and Transmit Queues

The descriptor must be aligned with a 32-bit boundary in the URAM.

Descriptors are generally configured of 64 bits, but are configured of 160 bits when reception and storage of gPTP time stamps is enabled.

The amount of data in the frame is defined by the data size bits (DESCR.DS). In reception, these bits indicate the upper limit on the size of frames to be received. If the data size is not aligned with a 32-bit boundary, the bytes to the next 32-bit boundary in the data area will be an unused area.

(2) Using Descriptor Chains in Queues

Transmission and reception descriptors in the URAM are grouped into queues. Each queue handles frames so that they are transmitted in order of priority and received separately. A queue is capable of controlling one or more frames. Accordingly, multiple descriptors can be assigned to one queue. A combination of multiple descriptors is referred to as a descriptor chain.

For a descriptor chain, the three general descriptor types listed below are defined. For details on these descriptor types, see section 46.3.3(6) Descriptor Type.

- Descriptors that define frame data
- Descriptors that control the descriptor chain itself (e.g. LINK, EOS).
- Descriptors that arbitrate access by hardware or software

Figure 46.11 shows the two basic topologies for descriptor chains. In the simplified examples in the figure, all descriptors allocated to the chain are stored in the array.

- For a linear descriptor chain, the last descriptor in the array is a control descriptor indicating the end of the descriptors (e.g. EEMPTY).
- For a cyclic descriptor chain, the last descriptor in the array is a control descriptor that returns to the first descriptor in the array (e.g. LINK).

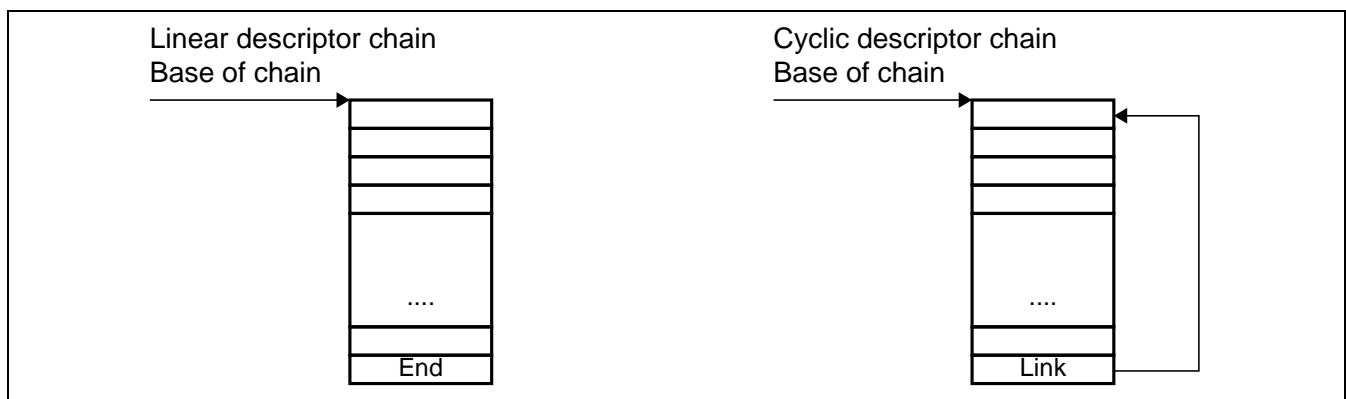


Figure 46.11 Outline of the Basic Descriptor Chains

The relationship between queues and descriptor chains is defined by the base addresses of chains. A queue is connected to one descriptor chain over one round of processing. There is also a method of switching to a different chain while in operation mode.

There are no restrictions on the number of link descriptors and their locations within the chain. The last descriptor of a designed chain determines the topology.

Which chain structure is to be used or which topology is suitable depends on the application. A description of how to design descriptor chains to suit various applications is given in section 46.3.4(2) Setting Up Reception Descriptors, and section 46.3.5(2) Setting Up Transmission Descriptors.

(3) Descriptor Base Address Table

The base address table in the URAM contains the address of the first descriptor of all chains to be handled by the respective queues.

Entries 0 to 3 are used to access transmission queues 0 to 3. Subsequent entries are used to access reception queues. Entry 4 thus corresponds to reception queue 0.

The configuration of entries in the base address table is the same as the configuration of link descriptors. We recommend using the descriptor type (DESCR.DT) LINKFIX. Processing of this link descriptor does not change it, so it does not require updating. The first descriptor of a chain performs hardware and software synchronization. If the application requires hardware and software synchronization for the base addresses, use the descriptor type (DESCR.DT) LINK.

The CPU is only capable of using LINKFIX and LINK as descriptor types (DESCR.DT) of descriptors in the base address table.

Set the location of the base address table in the URAM in the descriptor base address table register (DBAT).

Figure 46.12 shows an example of a base address table for controlling four transmission and three reception queues. The boxes to the right of the table represent descriptor chains with the desired topologies.

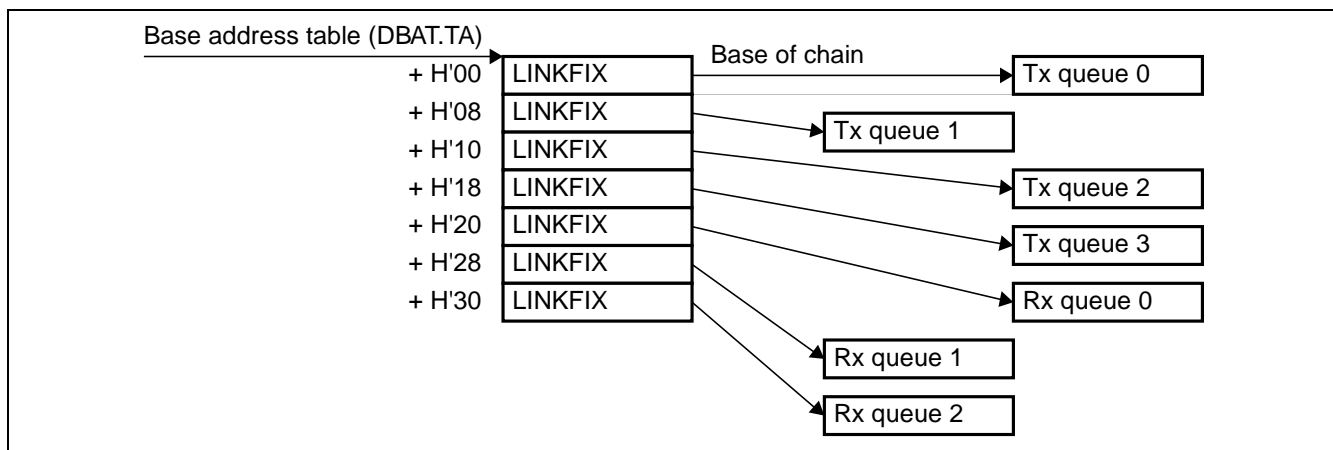


Figure 46.12 Example of a Base Address Table for Reception and Transmission Queues

Note: The size of the descriptors in the base address table is always eight bytes even if the queue itself includes extended descriptors.

(4) Descriptor Chain Processing

The descriptor that is currently processed or will be processed when the related queue gets active is the current descriptor. The current descriptor address for use by a queue q can be checked in the current descriptor address register q (CDARq).

Current descriptors are stored in registers or in descriptors as described below in the given situations.

- In the descriptor base address table registers for all q queues (DBAT) (DBAT.TA+8*q) when the operating mode shifts to operation mode.
- In the descriptor base address table register (DBAT) (DBAT.TA+8*q) when a base address load request is issued for a queue q by setting the corresponding bit (DLR.LBAq) in the descriptor base address load request register (DLR).
- In DESC.DPTR for a link descriptor (LINK, LINKFIX) to be processed.
- After a descriptor has been processed, the current descriptor for the same queue is incremented by the size of the descriptors being handled by the queue (8 bytes for normal descriptors and 20 bytes for extended descriptors). The AVB-DMAC updates the descriptor type and informs the CPU that the descriptor has been processed.

(5) Descriptor Interrupts

A descriptor is able to issue a descriptor interrupt on completion of its processing. The setting of the descriptor interrupt enable bits (DESCR.DIE) in each descriptor selects disabling or generation of the descriptor interrupt.

EthernetAVB-IF provides two kinds of descriptor interrupts:

- Queue specific descriptor interrupt (1 per receive/transmit queue)
- Universal descriptor interrupt (15 shared between all receive/transmit queues)

The descriptor interrupt is a common resource that is shared between reception and transmission queues. Software control of the descriptor interrupt provides a flexible method of application-specific flag processing.

Figure 46.13 illustrates the way in which the AVB-DMAC generates descriptor interrupts (or sets bits in the descriptor interrupt status register (DIS.DPFi)). Processing of a descriptor with the value i in the descriptor interrupt enable bits (DESCR.DIE) leads to the corresponding bit in the descriptor interrupt status register (DIS.DPFi) being set.

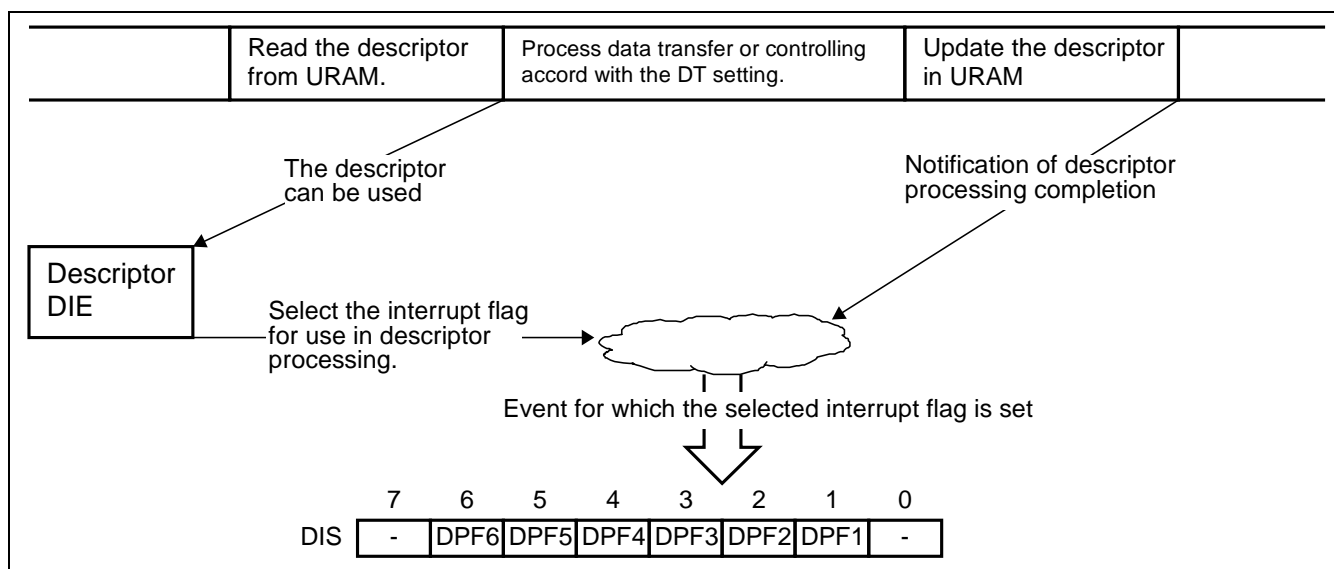


Figure 46.13 Method of Descriptor Interrupt Generation

(6) Descriptor Type

The descriptor types (indicated by the DESCR.DT bits) supported by the AVB-DMAC fall into the following three categories.

- Definitions of frame data
- Control of descriptor chains
- Hardware and software arbitration

Table 46.7 is a summary of the descriptor types available for the AVB-DMAC. Entries under “Name” are the names of the descriptor types and the values under “DT” are the corresponding values to be set in the descriptor type field (DESCR.DT). A given descriptor may be handled differently according to whether it is in a transmission or reception queue, so the transmission and reception columns list the scopes of control and processing of the descriptor types.

The abbreviations defined below are used in the transmission and reception columns.

Definition of SW:

- The descriptor is processed by software.
- Software has access to and may modify the descriptor and descriptor data area.
- This descriptor cannot be changed by hardware (AVB-DMAC).

Definition of HW:

- The descriptor is processed by hardware (AVB-DMAC).
- Software must modify neither the descriptor nor the descriptor data area.
- Hardware (AVB-DMAC) processes this descriptor and subsequently changes the descriptor type.

Invalid:

This descriptor type is not used in transfer in the given direction (transmission or reception).

Do not write this value to the descriptor type (DESCR.DT) field for transfer in the given direction.

Hardware does not process these descriptor types in the cases listed as invalid. The current descriptor address (CDARq.CDA) will not be changed when processing of a queue for the given direction arrives at a descriptor with this type setting.

Table 46.7 Summary of Descriptor Types

| Name | DT | Description | Reception | Transmission |
|-------------------|----|--|-----------|--------------|
| Frame data | | | | |
| FSTART | 5 | Frame Start The descriptor points to valid data for a frame. The frame starts with the given data and continues with that indicated by the next descriptor. | SW | HW |
| FMID | 4 | Frame Middle The descriptor points to valid data for a frame. The frame started with a previous descriptor and continues to the data indicated by the next descriptor. | SW | HW |
| FEND | 6 | Frame End The descriptor points to valid data for a frame. The frame continues from the previous descriptor and ends with the data indicated by in this descriptor. | SW | HW |
| FSINGLE | 7 | Frame Single The descriptor points to valid data for a complete frame. | SW | HW |
| Chain control | | | | |
| LINK | 8 | Link Defines the next descriptor in the chain. | HW | HW |
| LINKFIX | 9 | Fixed Link Same as LINK, but not changed by AVB-DMAC after processing. | SW | SW |
| EOS | 10 | End Of Set Control element to split a descriptor chain. The chain stops and waits for user interaction. | HW | HW |
| HW/SW arbitration | | | | |
| FEMPTY | 12 | Frame Empty A descriptor related to frame data but not containing valid data for a frame | HW | SW |
| FEMPTY_IS | 13 | Frame Empty Incremental Start A descriptor related to frame data but not containing valid data for a frame. DESCR.DPTR sets the base address of an "incremental data area" in the URAM. | HW | Invalid |
| FEMPTY_IC | 14 | Frame Empty Incremental Continue A descriptor related to frame data but not containing valid data for a frame. Data is stored to the incremental data area in the URAM. | HW | Invalid |
| FEMPTY_ND | 15 | Frame Empty No Data storage A descriptor related to frame data but not containing valid data for a frame. The descriptor is processed in the same way as FEMPTY but data are not stored in the URAM. | HW | Invalid |
| LEEMPTY | 2 | Link Empty A link descriptor for processing by the AVB-DMAC | SW | SW |
| EEMPTY | 3 | EOS Empty An EOS descriptor for processing by the AVB-DMAC | SW | SW |
| DT0 | 0 | Reserved | Invalid | Invalid |
| DT1 | 1 | Reserved | Invalid | Invalid |
| DT11 | 11 | Reserved | Invalid | Invalid |

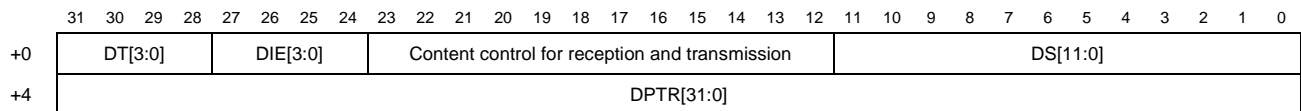
(7) Layout of General Descriptors in the URAM

The AVB-DMAC updates processed descriptors in the URAM. The field to be changed in the descriptor being updated depends upon whether the direction is transmission or reception and the queue mode. Other fields will not be changed. There are no restrictions on the values set in unused descriptor fields (indicated by “—” in the figure).

(a) Frame Data Descriptors

The allocation of bits in the frame data descriptors (FSTART, FMID, FEND, and FSINGLE) is shown below.

- Normal descriptor (usable in both reception and transmission)



- Extended descriptor (usable only in reception)

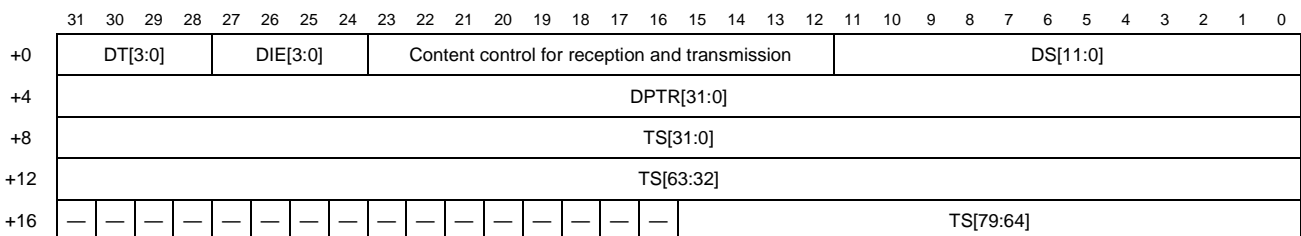


Table 46.8 Contents of Frame Data Descriptors (DESCR)

| Bit Name | Function |
|------------|---|
| DT[3:0] | Descriptor Type 5: FSTART 4: FMID 6: FEND 7: FSINGLE For details, see section 46.3.4 (2), Setting Up Reception Descriptors, and section 46.3.5 (2), Setting Up Transmission Descriptors. |
| DIE[3:0] | Descriptor Interrupt Enable B'0000: Descriptor interrupt is disabled. B'0001 to B'1111: The corresponding descriptor interrupt is generated (DIS.DPFI). |
| — | Content Control For details, see section 46.3.4 (2), Setting Up Reception Descriptors, and section 46.3.5 (2), Setting Up Transmission Descriptors. |
| DS[11:0] | Data Size Size of the data area/frame data for the descriptor (in bytes) |
| DPTR[31:0] | Descriptor Pointer Pointer to the data area for the descriptor |
| TS[79:0] | Time Stamp Time stamp of the received frame (only available in extended descriptors) |

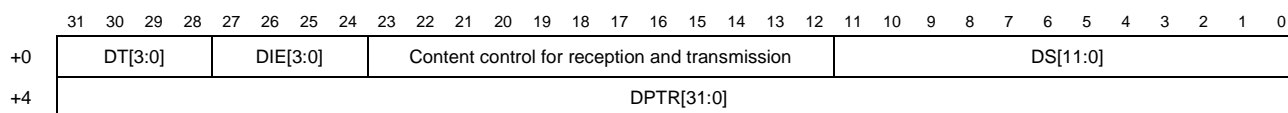
Note: Some bits in extended descriptors are reserved (The reserved bits (above DESCR.TS[79:64]) in an extended descriptor are set to H'0000 after the time stamp is stored).

(b) Hardware/Software Arbitration Descriptors (Only for Reception)

The allocation of bits in the descriptors for hardware/software arbitration (FEMPTY, FEMPTY_IS, FEMPTY_IC, and FEMPTY_ND) is shown below.

The allocation of bits in the arbitration descriptors for use in reception is the same as in frame data descriptors.

- Normal descriptor



- Extended descriptor (usable only in reception)

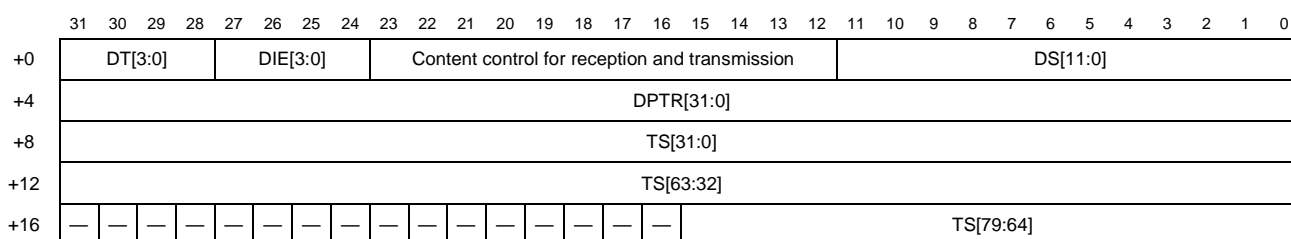


Table 46.9 Contents of Hardware/Software Arbitration Descriptors (DESCR)

| Bit Name | Function |
|------------|---|
| DT[3:0] | Descriptor Type 12: FEMPTY 13: FEMPTY_IS 14: FEMPTY_IC 15: FEMPTY_ND For details, see Table 46.7, Summary of Descriptor Types. |
| DIE[3:0] | Descriptor Interrupt Enable B'0000: Descriptor interrupt is disabled. B'0001 to B'1111: The corresponding descriptor interrupt is generated (DIS.DPFI). |
| — | Content Control For details, see section 46.3.4 (2), Setting Up Reception Descriptors, and section 46.3.5 (2), Setting Up Transmission Descriptors. |
| DS[11:0] | Data Size Size of the data area/frame data for the descriptor (in bytes) |
| DPTR[31:0] | Descriptor Pointer Pointer to the data area for the descriptor |
| TS[79:0] | Time Stamp Time stamp of the received frame (only available in extended descriptors) |

Note: When the descriptor is an extended descriptor, it has a 12-byte unused area.

In an FEMPTY descriptor, the descriptor type (DT), descriptor interrupt enable (DIE), data size (DS), and descriptor pointer (DPTR) fields are used.

In an FEMPTY_IS descriptor, the descriptor type (DT), descriptor interrupt enable (DIE), and descriptor pointer (DPTR) fields are used.

In an FEMPTY_IC descriptor, the descriptor type (DT) and descriptor interrupt enable (DIE) are used.

In an FEMPTY_ND descriptor, the descriptor type (DT), descriptor interrupt enable (DIE), and data size (DS) are used.

(c) Link Descriptors

The allocation of bits in the link descriptors (LINK and LINKFIX) is shown below.

- Normal descriptor

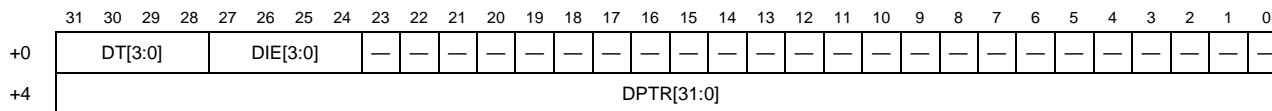


Table 46.10 Contents of Link Descriptors (DESCR)

| Bit Name | Function |
|------------|---|
| DT[3:0] | Descriptor Type 8: LINK 9: LINKFIX For details, see Table 46.7, Summary of Descriptor Types. |
| DIE[3:0] | Descriptor Interrupt Enable B'0000: Descriptor interrupt is disabled. B'0001 to B'1111: The corresponding descriptor interrupt is generated (DIS.DPFI). |
| DPTR[31:0] | Descriptor Pointer Pointer to the data area for the descriptor Register an address on a 32-bit boundary. |

Note: Register an address aligned with a 32-bit boundary as the descriptor pointer (DESCR.DPTR).

(d) Other Descriptors

The allocation of bits in the other descriptors (EOS, FEMPTY (only for transmission), LEMPTY, and EEMPTY) is shown below.

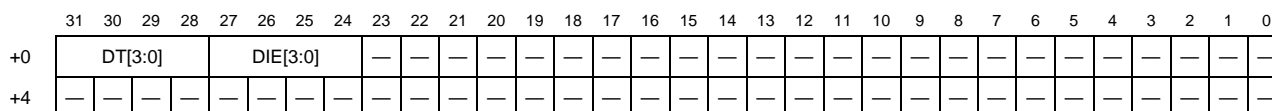


Table 46.11 Contents of Other Descriptors (DESCR)

| Bit Name | Function |
|----------|---|
| DT[3:0] | Descriptor Type 10: EOS 12: FEMPTY (only for transmission) 2: LEMPTY 3: EEMPTY For details, see Table 46.7, Summary of Descriptor Types. |
| DIE[3:0] | Descriptor Interrupt Enable B'0000: Descriptor interrupt is disabled. B'0001 to B'1111: The corresponding descriptor interrupt is generated (DIS.DPFI). |

(8) How to Use Frame Data Descriptors

The descriptor data area size field (DESCR.DS) can specify up to 2048 bytes of Ethernet frame data per data area. Settings higher than 2048 (bytes) cannot be made.

In general, Ethernet frames are not of uniform length. The AVB-DMAC is capable of dividing frame data into multiple descriptors in order to minimize the memory capacity for frame data. This function allows processing of frames that are longer than the limit for descriptor data areas. Division can also be applied to frames on the basis of their data structures.

To handle frames and descriptors, four descriptor types (DESCR.DT) as FSTART, FEND, FMID and FSINGLE are defined.

Figure 46.14 shows the mapping of frame data by frame data descriptors. The descriptor data areas are allocated to the URAM. For frames that require division into four or more data areas, additional FMID descriptors can be added as required.

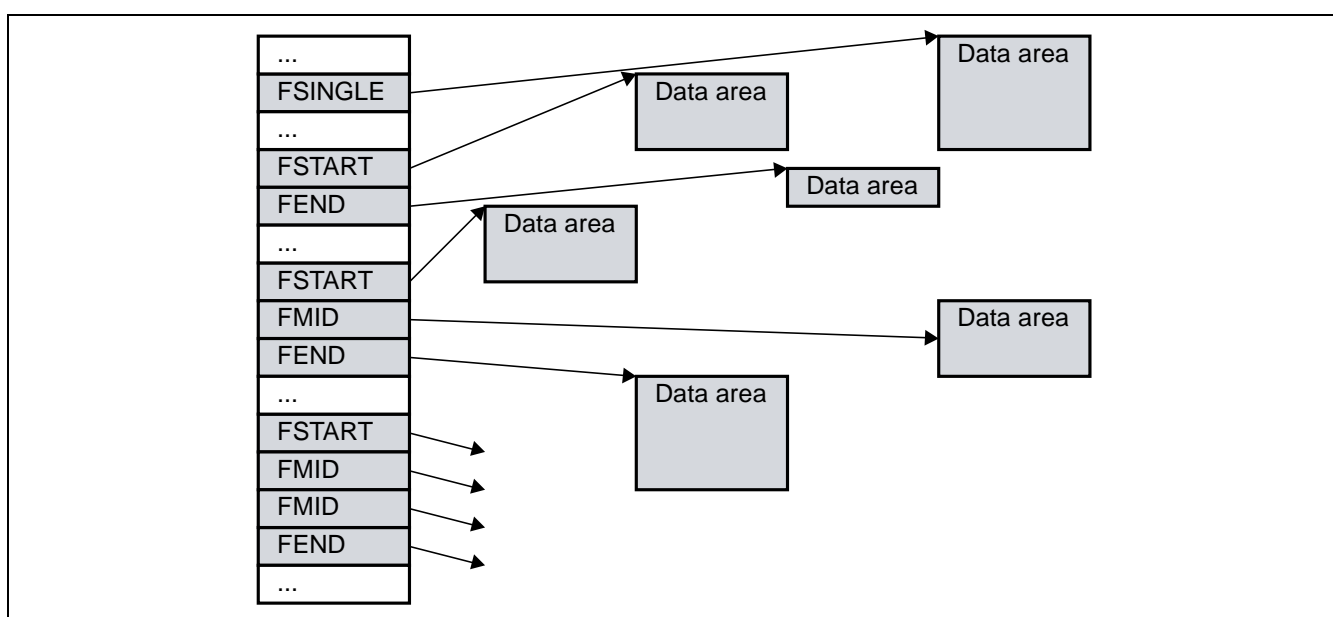


Figure 46.14 Mapping of Frame Data

For reception, set the descriptor data areas to the maximum size (i.e. give DESCR.DS its maximum value). The AVB-DMAC will store received frame data in the given area. If a received frame has more data than the maximum size, the AVB-DMAC will divide the data up.

For transmission, set the descriptor data area size field to the actual data size. The AVB-DMAC modifies the descriptor type (DESCR.DT) to FEMPTY after processing the relevant descriptor. The data size (DESCR.DS) and descriptor pointer (DESCR.PTR) fields retain their settings.

A descriptor data area including unused space produces an empty space between data areas. In reception, an “incremental data area” can be used to prevent empty spaces. For incremental data areas, see section 46.3.4(3)(b) Incremental Data Areas.

As well as reducing the memory capacity taken up by the descriptor area in the URAM, division into frames can be used to identify different sections of data (e.g. for separating a header and data).

(9) How to Use Chain Control Descriptors

Link Descriptors

The link descriptors can be used to set up cyclic descriptor chains (for details, see section 46.3.3(2) Using Descriptor Chains in Queues)

After a LINK descriptor is processed, its descriptor type (DESCR.DT) is changed to LEMPTY. The descriptor pointer (DESCR.PTR) retains its setting.

After processing of a LINKFIX descriptor, the descriptor type (DESCR.DT) is not updated. Software can change the descriptor type (DESCR.DT), descriptor interrupt enable (DESCR.DIE), and descriptor pointer (DESCR.DPTR). However, DESCR.DT should not be modified by software. Take care to check the current descriptor address register (CDARq.CDA) before changing the descriptor pointer (DESCR.DPTR).

EOS Descriptor

Use the EOS descriptor to divide a descriptor chain into various segments. The queue can continue even after an EOS descriptor.

In transmission, the response to an EOS descriptor is clearing of the transmit start request bit in the transmit configuration control register (TCCR.TSRQq) to 0.

In reception, the response is generation of a receive queue full interrupt (RIS2.QFFr), although if the frame currently being received is being divided for storage (received data such as those where some storage is in FMID- or FEND-type frames), the data are not completely stored.

(10) How to Use Hardware and Software Arbitration Descriptors

In hardware processing of descriptors, the empty descriptor types (FEMPTY, LEMPTY, and EEMPTY) are used to distinguish various descriptors. For software, they can be used to initiate checking for empty spaces, etc.

FEMPTY, FEMPTY_IS, FEMPTY_IC, and FEMPTY_ND

These descriptor types (DESCR.DT) are used for descriptors that do not contain effective data. Of these, only FEMPTY is used in transmission. The descriptor pointer of a FEMPTYxxx descriptor refers to a descriptor data area.

LEEMPTY

This descriptor type (DESCR.DT) is assigned to LINK descriptors after they have been processed. The descriptor pointer (DESCR.DPTR) of an LEMPTY descriptor still points to the linked descriptor.

EEMPTY

This descriptor type (DESCR.DT) is assigned to EOS descriptors after they have been processed. The descriptor pointer (DESCR.DPTR) of an EEMPTY descriptor is not used.

(11) Synchronization between Descriptor Access by Hardware and Software

The allocation of descriptor types (DESCR.DT) to the URAM can be used to set up the primary synchronization between hardware and software. By this, the number of CPU accesses to registers of EthernetAVB-IF can be minimized and performance can be increased.

Basic concepts of synchronization:

- Each descriptor type in the set is exclusively for processing by hardware or software, depending on the direction of transfer (see Table 46.7 Summary of Descriptor Types).
- Software must not change a descriptor assigned to hardware processing (the hardware does not change descriptors assigned to software processing).

In the case of software processing, the software must process the information in the descriptor and the corresponding frame data before changing the descriptor type. If a descriptor type for hardware is set in DESCR.DT, the software should not change any part of the descriptor or of the corresponding frame data.

(12) Tips for Optimizing Performance in Handling Descriptors

The following items are recommended as ways to ensure the optimal use of data structures in the URAM.

They are not requirements, but using a different approach may increase the load on the system bus within the LSI chip.

- Register descriptors with 64-bit alignment (this does not apply to extended descriptors).
- While in operation mode, use LINKFIX instead of LINK whenever a descriptor need not be changed. Hardware modifies the descriptor type (DESCR.DT) fields of LINK descriptors.
- Frame data is accessed in blocks up to 128 bytes.
- The number of 128 byte borders (addresses H'x_xx00 and H'x_xx80) and frame data inside should be minimized.
- Design the descriptor chains in ways that minimize parallelism of processing. This helps in dividing the chains into segments allocated to different cache pages, and in arranging the different segments exclusively for access by software or hardware.
- Minimize the number of divided frames. This can reduce the overhead of descriptor handling.

46.3.4 Control in Reception

The point of the AVB-DMAC is to transfer data between the E-MAC and URAM without intervention by the CPU.

AVB-DMAC needs descriptors that define the amounts of frame data to be stored and the locations. After the E-MAC receives a frame, it stores the received frame data and the conditions of reception as the E-MAC state. If the descriptor is extended, the time stamp is also stored. For a description of how to set up descriptors for use in reception, see section 46.3.4(2) Setting Up Reception Descriptors.

The AVB-DMAC filters received frames to separate them into various classifications (separation filtering). More specifically, this is done to separate received frames into the various reception queues and to set the priorities of different classes of received frames. For more on separation filtering, see section 46.3.4(1)(a) Separation Filtering.

Figure 46.15 shows the reception data bus and the selection of queues for use in reception.

Each frame received from the E-MAC is stored in the reception FIFO; in parallel with this, the frame is analyzed to identify its type and the target queue number. After the E-MAC completes reception, the target queue number is generated and stored in the reception FIFO. Appending of a reception flag depends on the storage of one frame among the reception queues in the URAM, and the unread frame counter (UFC) is also associated with frame storage.

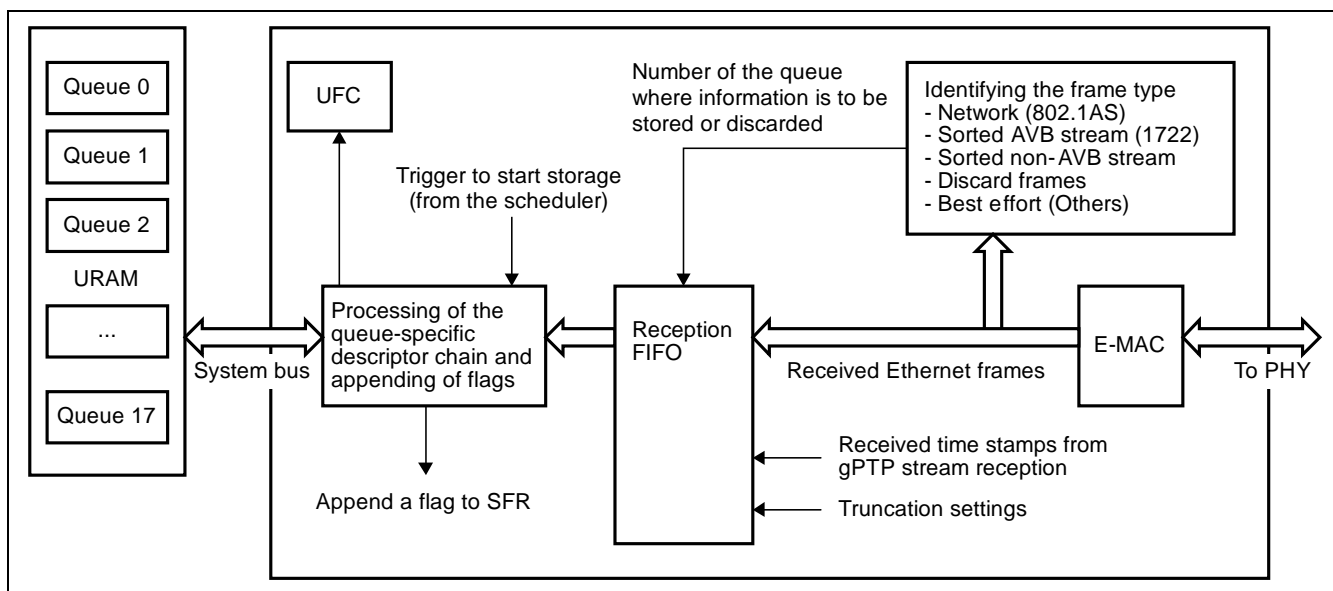


Figure 46.15 Mechanism of General Reception Queue Selection

(1) Reception Queues

The AVB-DMAC applies its separation filtering mechanism to select the reception queue for storing a received frame. The AVB-DMAC stores all received frames in the URAM.

There are two conditions for the AVB-DMAC to discard a received frame.

- Detection of an error during reception by the E-MAC
 - Whether error frames are discarded or stored in reception queue 0 (best effort) depends on the setting of the error frame enable bit in the receive configuration register (RCR.EFFS). If error frames are to be stored (RCR.EFFS = 1), they are always stored in queue 0 (best effort). In this case, characteristics specific to the queue (e.g. truncation) will vary. If the storage of time stamps for reception queue 0 (best effort) is enabled (the time stamp enable bit in the receive configuration register RCR.ETS0 = 1), time stamps are stored even for error frames.
- Frame fails the separation filter
 - It depends on RCR.ESF if such frame is discarded or stored in receive queue 0 (best effort).

The flowchart in Figure 46.16 shows how the AVB-DMAC selects the reception queue in accord with the frame type, including judgment by the separation filter. Selection of the queue starts when the E-MAC completes frame reception. The result is storage of the frame in the proper queue or the frame being discarded.

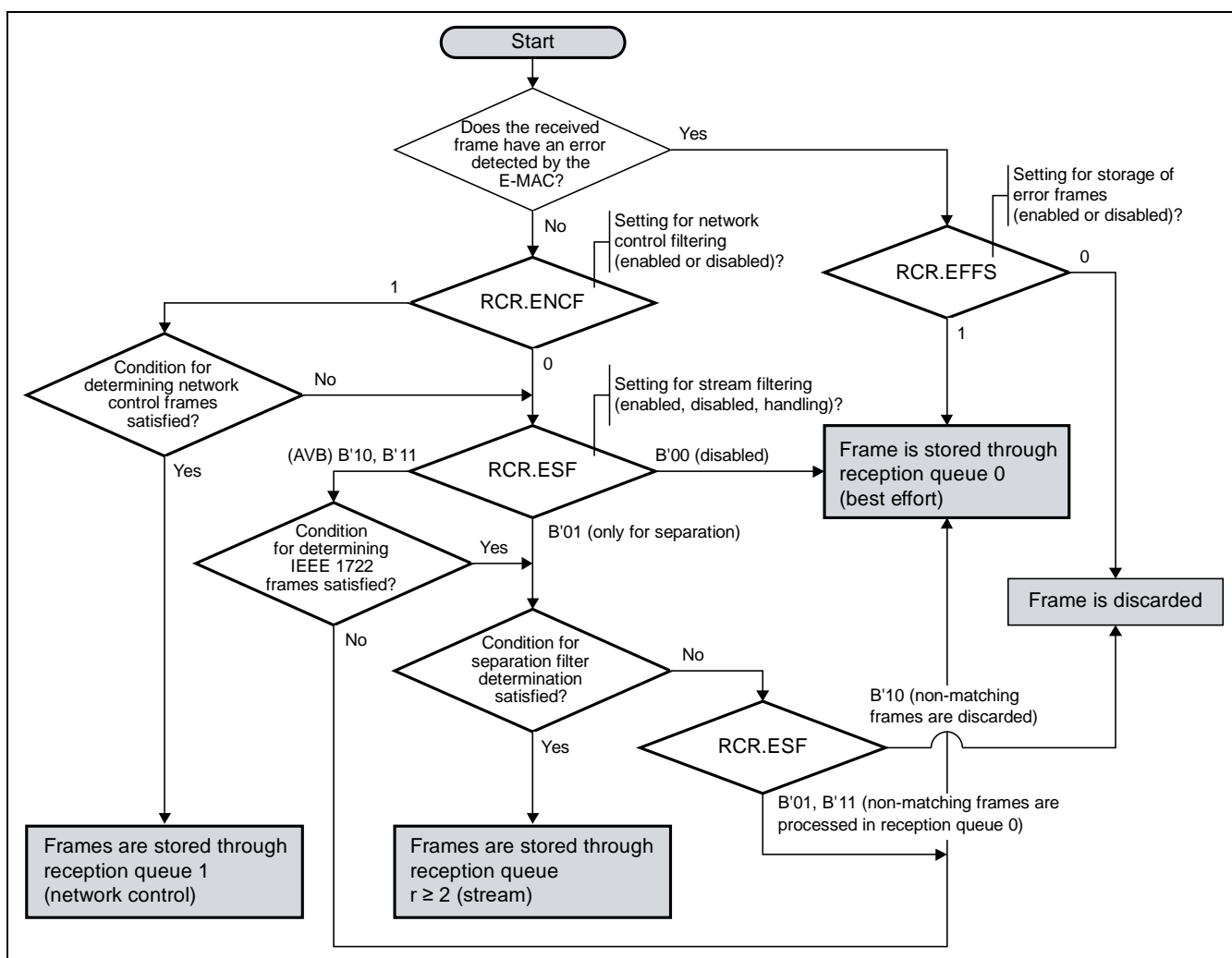


Figure 46.16 Mechanism of Reception Queue Selection

Notes on the meanings of entries in the flowchart

- “Condition for determining network control frames”
 The Ethernet destination address (DA) is 01:80:C2:00:00:0E.
 The Ethernet type (ET) is 88:F7.
- “Condition for determining IEEE 1722 frames”
 The Ethernet destination address (DA) is within the range from 91:E0:F0:00:00:00 to 91:E0:F0:00:FE:FF.
 The VLAN tagged TPID (tag protocol identifier) field (VL) is 81:00.
 The Ethernet type (ET) is 22:F0.
- “Condition for separation filter determination”
 See section 46.3.4(1)(a) Separation Filtering.

Figure 46.17 shows the allocation of bits related to the network and stream types in Ethernet frames. The preambles of Ethernet frames are not taken into account.

| Data bytes | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
|--------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Network type | DA1 | DA2 | DA3 | DA4 | DA5 | DA6 | SA1 | SA2 | SA3 | SA4 | SA5 | SA6 | ET1 | ET2 | ... | ... | ... | ... | ... | ... |
| Stream type | DA1 | DA2 | DA3 | DA4 | DA5 | DA6 | SA1 | SA2 | SA3 | SA4 | SA5 | SA6 | VL1 | VL2 | - | - | ET1 | ET2 | ... | ... |

Figure 46.17 Data Bytes of Ethernet Frames Used in Classification

(a) Separation Filtering

Separation filtering involves the checking of up to 64 bits (eight successive bytes) in received Ethernet frames. The setting for the first byte (i.e. the setting of the separation filter offset configuration register (SFO.FBP)), selects the part of frames to be used in separation filtering. There is also a common filter mask (set in the separation filter mask configuration register (SFMi.CFM)) that can be freely set to reduce the number of bytes used in separation filtering or to mask particular bits.

Examples

To use one byte in separation, set separation filter mask configuration register 0 (SFM0.CFM) to H'0000_00FF and separation filter mask configuration register 1 (SFM1.CFM) to H'0000_0000.

To use seven bytes in separation, set separation filter mask configuration register 0 (SFM0.CFM) to H'FFFF_FFFF and separation filter mask configuration register 1 (SFM1.CFM) to H'00FF_FFFF.

Note: If bits at some positions are set to 0 in the separation mask, in order to match with the pattern, the bits at the corresponding positions of the pattern must also be set to 0. Only those bits in which the separation filter pattern configuration register (SFPi.FPs) setting is equal to the separation filter mask configuration register (SFMi.CFM) are sorted by matching with received data.

Figure 46.18 shows separation filtering. The selected data from a received frame (Rx_Frame[63:0]) are masked by the common filter mask. As a result, the selected frame data can be obtained. This value is compared with all filter patterns. The separation filter circuit in the AVB-DMAC selects the filter pattern that matches the queue having the lowest index s or selects a flag to indicate that there is no matching separation pattern.

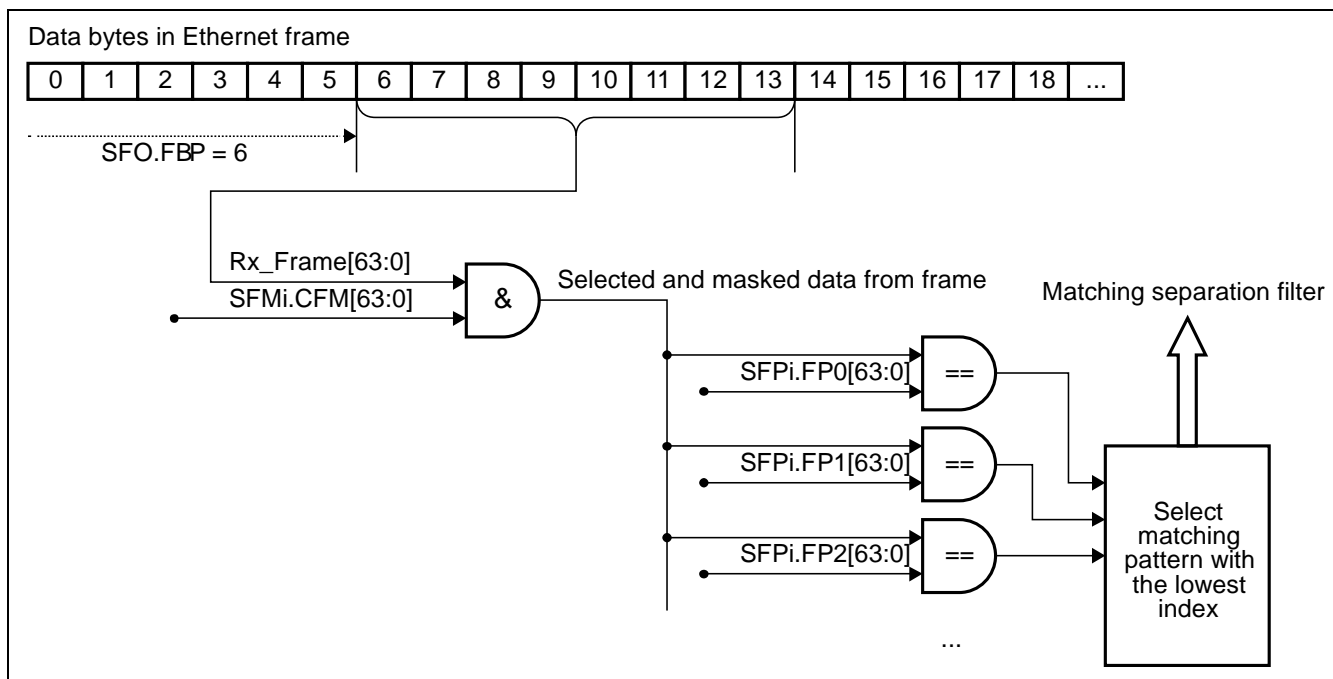


Figure 46.18 Separation Filtering

(b) Separating Streams

The AVB-DMAC applies separation filtering to sort frames received in streams. An AVB network has a concept of “Talker” and “Listener”. A Talker is an end station that generates one or more streams. A Listener is an end station that has the role of being a sink for at least one stream. The various A/V streams are identified by 8-byte stream IDs.

The number of end stations within an AVB network and their roles differ with the application.

The stream ID is a general pattern of the AVB network for identifying one stream. Figure 46.19 shows the bit allocation of bits in IEEE1722 Ethernet frames and stream ID fields.

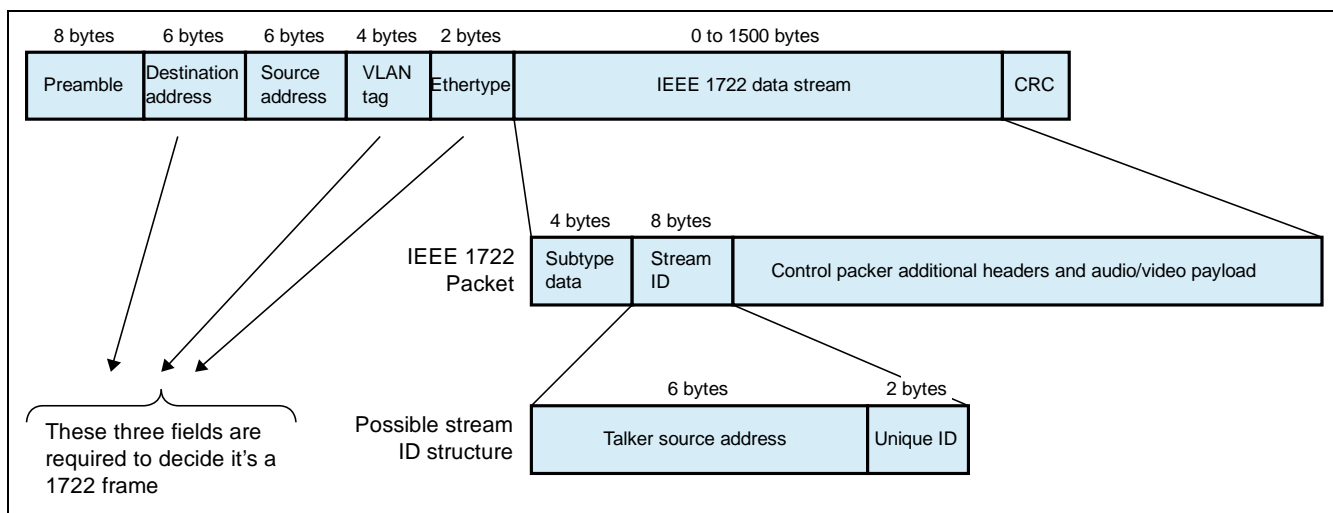


Figure 46.19 IEEE 1722 Frame Layout and Stream ID

The IEEE 1722 standard stipulates that the stream ID field starts from the 23rd byte (not counting the preamble). Accordingly, set the separation filter offset (SFO.FBP) to 22 in operations on IEEE 1722 streams. Set the separation filter mask (SFMi) and separation filter pattern (SFPi) in accord with the specification of the product in which the chip is being used.

Example: In the example of a stream ID shown in Figure 46.19, the current application divides the field into the talker source address and the unique stream ID. The unique ID is used to differentiate between multiple streams from the same talker. Based on this, there are two settings for separation filter masking:

- To divide various streams into individual queues, set SFM0.CFM to H'FFFF_FFFF and SFM1.CFM to H'FFFF_FFFF.
- To divide streams from various talkers into individual queues, set SFM0.CFM to H'FFFF_FFFF and SFM1.CFM to H'0000_FFFF. This excludes the unique ID from the filter condition.

(c) Reconfiguration of separation filter during OPERATION

In CONFIG mode the initial separation filter configuration is defined. To prevent inconsistent filter results during OPERATION direct parameter change is not possible; instead update mechanism by SFVi.LV and SFL.LC should be used.

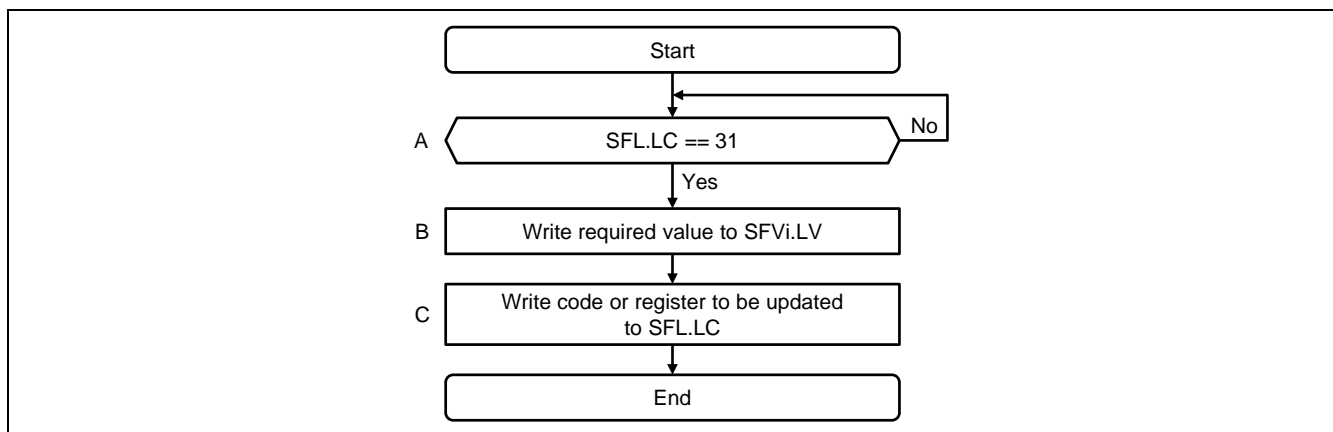


Figure 46.20 SW flow to update separation filter during OPERATION

When reconfigure separation filter, EthernetAVB-IF guarantees consistent classification for one received frame. If reconfiguration requires a sequence of parameter changes, CPU has to order reconfiguration based on application demand.

EthernetAVB-IF updates filter configurations when no frame reception is handled. In worse case the update is delayed by one frame reception. Exact duration depends on PHY speed and frame size. By reading SFL.LC CPU can observe when a previous update has been processed.

Note: Frame classification happens during frame reception and storage into internal Rx-FIFO. There may be frames with old classification pending in Rx-FIFO or under storage to URAM when filter reconfiguration is accepted.

(2) Setting Up Reception Descriptors

For reception, the descriptor mechanism is essentially as described in section 46.3.3 Descriptors.

This section describes memory operations that are especially required in handling reception queues.

(a) Reception Descriptor Type

The type of a descriptor is defined by the descriptor type (DESCR.DT) field.

Table 46.12 shows the descriptor types used in reception.

Table 46.12 Descriptor Types in Reception

| Descriptor Type (DESCR.DT) | Operation | Write-back |
|--|--|--------------------------------|
| Frame Start (FSTART) | Condition for data not being stored in a reception queue: The RIS2.QFFr bit indicates that queue r is full and the received frame is not stored. Descriptor processing proceeds again in response to further reception. | Not changed |
| Frame Middle (FMID) | Same as FSTART | Not changed |
| Frame End (FEND) | Same as FSTART | Not changed |
| Frame Single (FSINGLE) | Same as FSTART | Not changed |
| Link (LINK) | Processing proceeds to the descriptor specified by DESCR.DPTR. | LEMPY |
| Fixed Link (LINKFIX) | Same as LINK | Not changed |
| End Of Set (EOS) | A stop point defined by software has been reached. A descriptor of this type within a divided frame (writing of FMID or FEND) stops the frame being stored and the frame is lost. RIS2.QFFr indicates that the frame has been lost. If this happens at the start of a frame (writing of FSTART or FSINGLE), storing of frames starts from the next descriptor. In either case, processing shifts to the next descriptor in the chain. | EEMPTY |
| Frame Empty (FEMPTY) | The descriptor can be used to store received data. Up to DESCR.DS bytes are stored in the descriptor data area. For details, see section 46.3.4(3)(a) Storing Frame Data in the Descriptor Data Area. | FSTART, FMID, FEND, or FSINGLE |
| Frame Empty Incremental Start (FEMPTY_IS) | The descriptor can be used to store received data. All data for the frame are stored in the descriptor data area. DESCR.DPTR indicates the base address of the incremental data area. For details, see section 46.3.4(3)(b) Incremental Data Areas. | FEND or FSINGLE |
| Frame Empty Incremental Continue (FEMPTY_IC) | The descriptor can be used to store received data. The remaining bytes of frame data are stored in the descriptor data area. DESCR.DPTR is undefined, but is written back at the start position within the incremental data area after processing. For details, see section 46.3.4(3)(b) Incremental Data Areas. | FEND or FSINGLE |
| Frame Empty No Data storage (FEMPTY_ND) | The descriptor can be used to store received data. Up to DESCR.DS bytes are captured from the reception FIFO but not stored. After processing, DESCR.DS is written back as 0. For details, see section 46.3.4(3)(b) Incremental Data Areas. | FSTART, FMID, FEND or FSINGLE |
| Link Empty (LEMPY) | Same as FSTART | Not changed |
| EOS Empty (EEMPTY) | Same as FSTART | Not changed |

(b) Configuration of Reception Frame Data Descriptors

Figure 46.21 shows the configuration of descriptors for use with reception queues. The reception-specific fields are the same whether the descriptor is normal or extended. The reception-specific fields (DESCR.MSC, DESCR.PS, DESCR.EI, and DESCR.TI) are described in Table 46.13.

For the other fields and the descriptor types, see section 46.3.3(6) Descriptor Type.

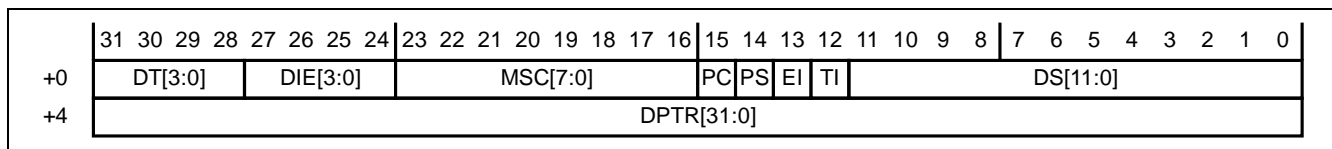


Figure 46.21 Configuration of Descriptor for a Received Frame

Table 46.13 Configuration of a Received Descriptor

| Bit Name | Function |
|----------|--|
| MSC | <p>E-MAC Status Code</p> <p>These bits indicate errors in reception detected by the E-MAC.</p> <p>In the case of a divided frame, these bits are set to the same value within all descriptors for the frame data. Details of the bits are as follows.</p> <p>MSC[7]: Received frame has a multicast address.</p> <p>MSC[6]: Carrier extend error</p> <p>MSC[5]: Carrier sense error</p> <p>MSC[4]: Received frame has alignment error.</p> <p>MSC[3]: Received frame is too long.</p> <p>MSC[2]: Received frame is too short</p> <p>MSC[1]: Error in frame reception (PHY detects an error)</p> <p>MSC[0]: Received frame has a CRC error.</p> |
| PS | <p>Padding Selection</p> <p>These bits specify whether frame data are to be padded when stored in the incremental data area.</p> <p>Insertion of padding data is in accord with the settings in the RPC register.</p> <p>0: Padding is not to be inserted.</p> <p>1: Padding data may be inserted. This depends on the RPC settings.</p> |
| EI | <p>Error Indication</p> <p>This bit indicates the detection of an error in frame data while a frame was being stored.</p> <p>The bit is set to 1 for a descriptor in which an error has been detected. If the descriptor is for a divided frame, storage of the frame is aborted.</p> <p>0: No error</p> <p>1: Error is detected</p> |
| TI | <p>Truncation Indication</p> <p>This bit indicates whether frame data received from the E-MAC have been truncated before being stored.</p> <p>These bits are set to the same value within all frame data descriptors for a divided frame.</p> <p>0: Data have not been truncated.</p> <p>1: Data have been truncated.</p> |

Note: The RCR.EFFS bit specifies whether or not frames with errors detected by the E-MAC are to be stored in the URAM. When the storing of error frames is disabled, error codes are not written to DESCR.MSC.

(3) Reception Processing

After initialization, the AVB-DMAC is able to select the proper reception queue and store received frames in the data area in the URAM as indicated by the descriptor. The AVB-DMAC continues to store received data in the URAM as long as space is available for descriptors and data areas.

Received frames are classified and stored in the reception FIFO in accord with the algorithm described in section 46.3.4(1)(a) Separation Filtering.

If the reception FIFO contains even one frame, storing is executed to the reception queue.

If there is even one empty data descriptor in a queue for which reception has started, the storage of frame data starts. Received frames for a queue that is already full (there is no empty frame descriptor or the UFC stop level has been reached) are discarded from the reception FIFO. This ensures that one queue being full does not prevent the storage of data in the other queues.

(a) Storing Frame Data in the Descriptor Data Area

Frame data for storage are assumed to be in either of the two patterns described below.

- The data for an entire frame will fit in the descriptor data area.
 - In this case, the descriptor type (DESCR.DT) is FSINGLE.
- Frame data to be stored in the descriptor data area arrive in divided form.
 - In this case, FSTART is written to the descriptor type (DESCR.DT) bits of the first descriptor of the frame data to arrive and FMID and FEND are written to the type bits of descriptors for subsequent data.

The descriptor type is updated by the AVB-DMAC in the last step of descriptor processing, so software can always access the descriptor assigned to DESCR.DT.

When normal synchronization mode is used, the CPU can write FEMPTYxxx directly to the descriptor type field after processing the stored element. Do not change the descriptor or any part of the descriptor data area after FEMPTYxxx is written to DESCR.DT.

Storing Frame Data for a Whole Single Frame

For a frame with an FSINGLE descriptor, all data for the frame are held at the position defined by DESCR.DPTR. DESCR.DS indicates the length of the received frame.

If DESCR.DS is bigger than the actual size of a received frame, the FSINGLE descriptor is stored in place of the FEMPTY or FEMPTY_ND descriptor after processing.

Also, the FSINGLE descriptor is stored in place of the FEMPTY_IS or FEMPTY_IC descriptor, which always hold the descriptor data area greater than all frame data in Rx-FIFO after processing.

Storing Frame Data as Divided Frames

Divided frames are handled in the same way as a single frame. A frame stored with divided descriptors must be recombined before use. DESCR.EI and DESCR.TS are only valid in the last descriptor of the sequence for a divided frame.

Note: If the data area size setting in DESCR.DS is not a multiple of four, the number of bytes set in DESCR.DS is fetched from the reception FIFO and the remaining bytes are used as the next storage area.

After a received frame is divided into different descriptors, each storage element is handled separately, and the descriptor type is assigned by software after processing. Accordingly, an error frame (FEMPTYxxx instead of FMID or FEND) may exist while a descriptor chain is being processed. In such a case, the CPU must postpone processing of the error frame to the next trigger point.

No Data are Stored

The application specification may lead to some types of received frames being unimportant (for example, when the application only requires stream data from the Ethernet frames). Storing frames in divided form makes separating out the unnecessary parts of Ethernet frames possible.

If part of a divided frame is not required, use the FEMPTY_ND descriptor for that part so that it is not stored in the URAM. Not storing the data negates the need for bandwidth on the data bus, improving the overall performance.

When an FEMPTY_ND descriptor is processed, DESCR.DS is set to 0. This brings the frame data section of the descriptor into agreement with the FEMPTY type. DESCR.DS = 0 is for the unique identification of the descriptor after writing.

(b) Incremental Data Areas

Secure space in the URAM for storing received data. Even when data are placed in the URAM area such that all descriptor data areas of a chain are contiguous, a received frame being shorter than the descriptor data area will lead to an empty space. Figure 46.22 shows an example of settings and the memory map.

Certain applications require that data areas be contiguous (e.g. when received data are to be processed other than by hardware as the A/V codec module). When the length of received frames differs (e.g. when payloads vary between having one or two A/V packages), the use of a static pointer in the descriptor produces empty spaces in the data area. This may necessitate direct additional processing to remove the empty spaces.

Accordingly, and to reduce the CPU load imposed by copying data, the AVB-DMAC supports an “incremental data area” function.

When incremental data areas are in use, all descriptors use a common data area for storage. One descriptor (FEMPTY_IS) defines the base address of the incremental data area and the next descriptor (FEMPTY_IC) within the descriptor chain holds received data. Figure 46.23 shows an example of settings and the memory map.

Use of an incremental data area does not reduce the memory space in the individual descriptor data areas.

The hardware and software synchronization strategy and performance are also not changed.

It is also possible to divide a frame up among various descriptors in a way that reflects its structure (e.g. one descriptor for the Ethernet header and one for the data payload).

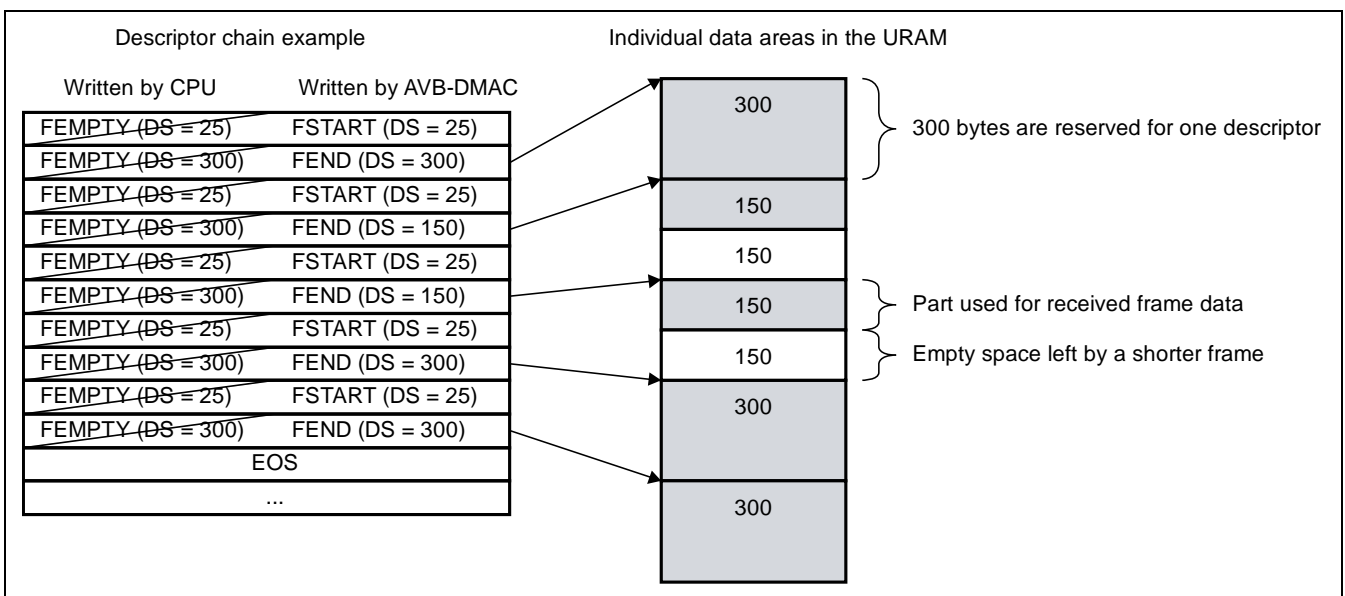


Figure 46.22 A Reception Queue Using Individual Descriptor Data Areas

Figure 46.22 and Figure 46.23 show how control of the data storage areas by a descriptor chain varies according to whether individual or incremental data areas are in use. The chains are configured for storing received frames consisting of a 25-byte header (which is treated as one descriptor; and is outside the scope) and a 150- or 300-byte payload (whether one or two 150-byte payload packages are transmitted with one Ethernet frame depends on the data source).

In Figure 46.22, the EOS descriptor is added as an example of a re-synchronization point. If the frame source transmits a frame containing more than 325 bytes, the frame will be divided among three descriptors, meaning that synchronization of the header and data sequences is lost. Despite this, however, the frame is not divided across the EOS descriptor, so recovery is automatic without software interaction. The EOS is not required with the incremental descriptors because all data being processed are always stored while an incremental data area is in use.

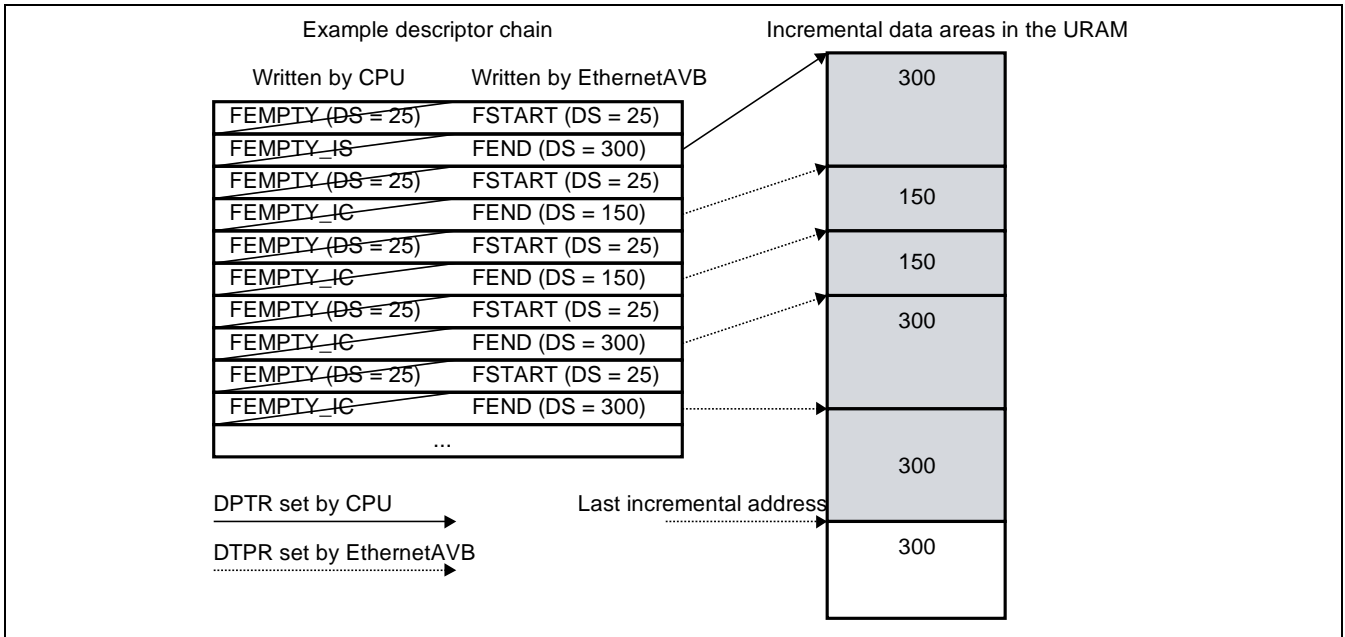


Figure 46.23 Reception Queue Using a Common Incremental Data Area

As Figure 46.23 shows, when data are stored in an incremental data area, the descriptor pointers in the FEMPTY_IC descriptors (DESCR.DPTR) are updated. Accordingly, the resulting FEND or FSINGLE descriptor is in the same format as after writing to an FEMPTY descriptor.

Software captures received data from an incremental data area, which has no empty storage areas between frame data. The only empty space is that at the end of the incremental data area. The sizes of incremental data areas and of blocks of data for storage in incremental data areas must be multiples of four bytes. When the amount of data for storage in an incremental data area is not a multiple of four bytes, from one to three bytes of empty space will be produced. DESCR.DS can be read to check for such empty spaces.

CPU cannot restrict directly the amount of received data stored by an incremental descriptor (FEMPTY_IS, FEMPTY_IC) as this is possible for other descriptors (FEMPTY, FEMPTY_ND) by DESCR.DS. Incremental descriptors store always all received data.

Setting Up an Incremental Data Area

A descriptor chain in the incremental data area having N descriptors (one FEMPTY_IS and N-1 FEMPTY_IC) means that the incremental storage area of N times of maximum frame size is needed.

As Figure 46.23 shows, DESCR.DPTR of an FEMPTY_IS descriptor indicates the base address of the incremental data area. The next FEMPTY_IS descriptor in the chain indicates the processing step where data must be stored in the incremental data area.

Processing an Incremental Data Area Based on Descriptors

Since data processing by the CPU is the same regardless of how the AVB-DMAC stores the data, data stored in an incremental data area do not require any special handling.

Padding

Use padding for received frame data that are not aligned correctly in the specified memory structure. Padding can be set individually for each descriptor. Accordingly, in the reception of divided frames, padding can be restricted to only those frames that require it (e.g. A/V payload data.)

Padding can also be used to optimize system performance in an incremental data area (e.g. to prevent inefficient access by aligning received data with 32-byte boundaries in the incremental data area), as well as to fulfill application-specific requirements for specified memory structures (e.g. formats required by other modules that will be processing the received data).

Padding can only be used in an incremental data area.

The value H'0000_0000 is always used in padding.

Padding is the addition of the number of words (from one to seven 32-bit words) set in the stored padding counter in the receive padding configuration register (RPC.PCNT). This padding is repeatedly inserted in accord with the value in the stored data counter (RPC.DCNT) (from one to 255 32-bit words). When the stored data counter (RPC.DCNT) reaches 0, however, padding is not repeated.

The first word of padding is always inserted at the position specified by DESCR.DPTR. When divided frames are in use, a padding word can be inserted at any byte position, and padding is handled on a 32-bit basis (e.g. an incremental data area where the first descriptor is for a 42-byte header data and the second descriptor holds padded payload data).

The next figure shows a general example of how padding is inserted and an example of setting up padding. A indicates frame data A received by the E-MAC, and B indicates the frame data stored in the descriptor data area (32-bit word units).

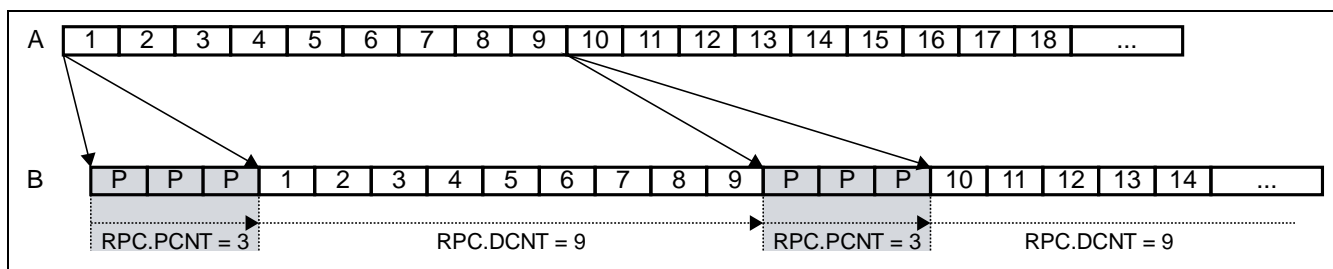


Figure 46.24 Example of a Padding Setting

Both padding and received frame data are counted in the descriptor size (DESCR.DS).

(c) Mode with Write-Back

Constructing a descriptor chain requires software (see Figure 46.25).

In the example in the figure, the variable SWdescr (software descriptor pointer) is a structure to identify a descriptor being processed. SWdescr must be initialized after operation mode is entered and a descriptor base address load request (DLR.LBAq) is executed (condition for starting the flow of software operations).

The frame_processing() function processes the stored data. The function can use SWdescr.DT to check whether processing of a frame is completed. How frame data are processed differs with the application, so create functions that handle processing in accord with the specification.

The processing section is common to all modes of reception. The number of frames processed in response to each trigger can be restricted. When multiple frames have to be processed in a batch, waiting for individual trigger boxes must be skipped for these frames.

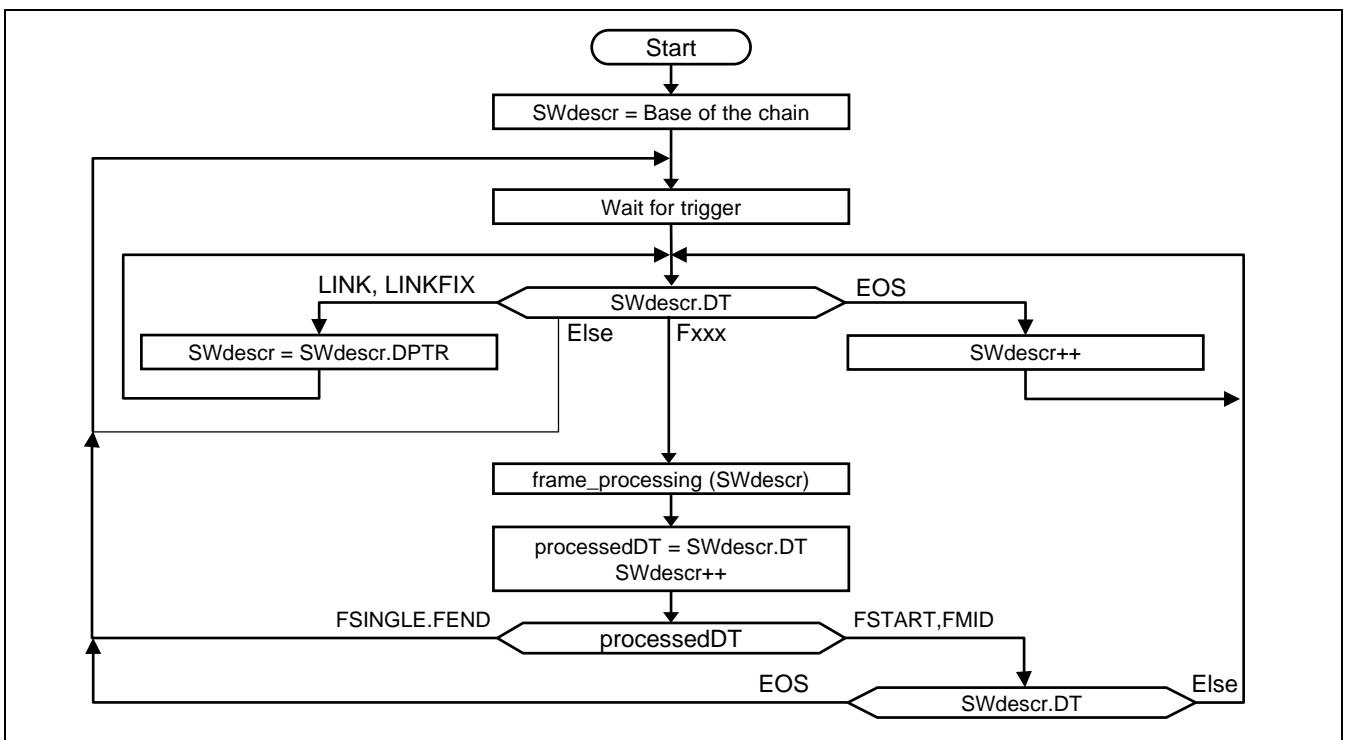


Figure 46.25 Flow of Reception Descriptor Processing (with Write-Back)

(d) Support for Reception Time Stamps

Capturing reception time stamps is essential for IEEE 802.1AS time synchronization. Other types of received frames may also require that a reception time stamp be appended; this depends on the application. The AVB-DMAC supports reception time stamps based on the gPTP timer by storing time stamps, which have been captured when Start Frame Delimiter (SFD) of a received frame is arrived, in the last frame data descriptor (FEND or FSINGLE). For the gPTP timer, see section 46.3.7(1) gPTP Timer.

When time stamps are to be stored, use extended descriptors for the entire reception queue. Furthermore, time stamps are always stored for reception queue 1 (network control). Time stamps for reception queue 0 (best effort) and reception queue r (r ≥ 2; for stream data) can be selected by the time stamp enable bits in the receive configuration register (RCR.ETS0 or RCR.ETS2).

(4) Unread Frame Counters

Each reception queue has an unread frame counter (UFCVi). Use the unread frame counter configuration bits in the receive queue configuration register (RQCi.UFCCr) to select from among the four warning and stop levels for each unread frame counter. The 0 setting disables the stop and warning functions. For how to set this up, see Figure 46.26.

Operations of the AVB-DMAC (hardware) and CPU (software) drive an unread frame counter (UFC) in the following ways.

- The hardware indicates that it has added a new frame to the descriptor chain for the queue (this increments the counter).
- Software indicates how many frames from the descriptor chain it has processed by writing to the corresponding bits of the unread frame counter decrement register for the queue (this decrements the register by the number written).

The unread frame counter is based on the number of frames stored in the URAM and is only incremented by one even when a received frame is divided into different descriptors. Failure in storing a descriptor chain requires care because this unread frame counter may fail in synchronization as described in section 46.3.4(4)(a) Unread Frame (UFC) Synchronization Failure.

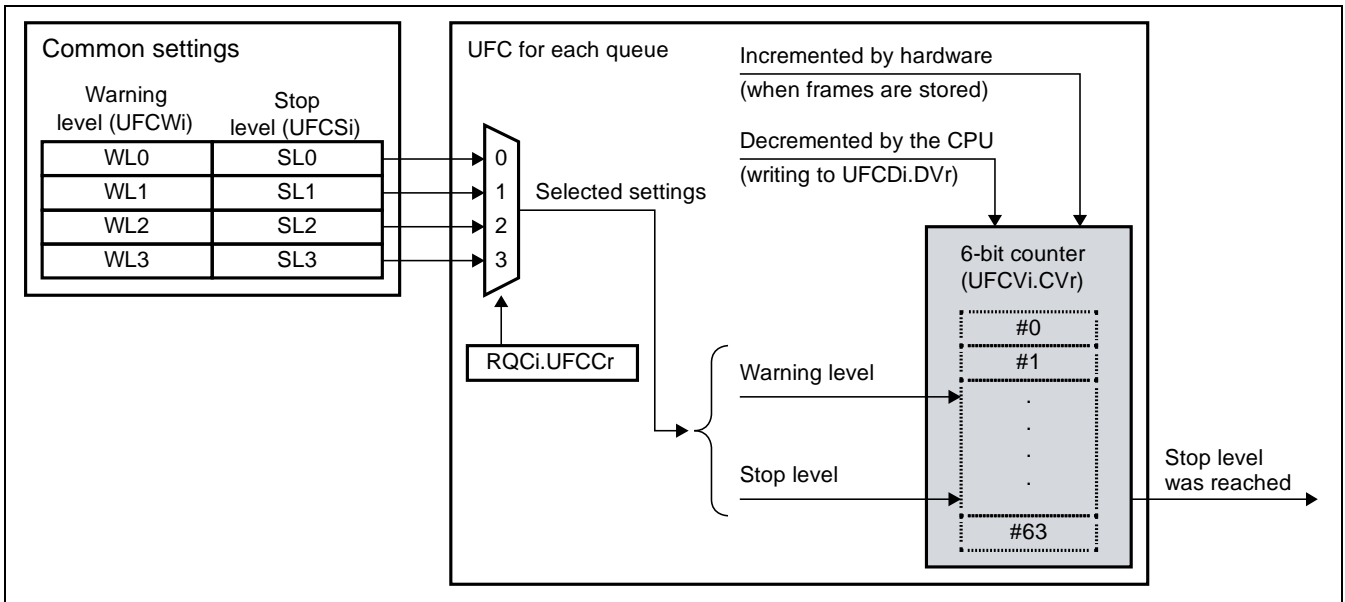


Figure 46.26 Overview of an Unread Frame Counter

Unless synchronization of hardware and software is lost, the current unread frame counter value (UFCVi.CVr) indicates the number of unread frames in the queue.

The indicator that the stop level has been reached prevents the storage of further received frames in the descriptor chain. Selecting 0 as the stop level disables this function. Otherwise, further received frames for the queue are discarded once its unread frame counter reaches the stop level. Activation of the unread frame counter stop function is indicating by setting of the receive queue full interrupt flag in the receive interrupt status register 2 (RIS2.QFFr).

Set the unread frame counter warning level configuration register (UFCW) and the unread frame counter stop level configuration register (UFCS) for each reception queue that will use the unread frame counter function while the current operating mode is configuration mode.

(a) Unread Frame (UFC) Synchronization Failure

The unread frame counters do not recognize failure to store a frame in the URAM. In other words, the AVB-DMAC increments the counter for a queue each time it captures a frame for that queue from the reception FIFO whether or not it succeeds in storing the frame normally in the descriptor chain.

In general, synchronization of hardware and software fails under the following conditions.

- An unread frame counter reaching its maximum value
When the value of a counter in an unread frame counter register i (UFCVi) ($i = 0$ to 4) reaches 63, synchronization for the corresponding queue can fail.
The CPU can only judge that a failure in synchronization has not occurred when the stop level is set to 63.
- A queue not having enough space for a descriptor
In this case, the corresponding receive queue full interrupt flag (RIS2.QFFr) in receive interrupt status register 2 (RIS2) is set.
If an unread frame counter reaches its stop level while synchronization remains normal, the receive queue full interrupt flag (RIS2.QFFr) in the receive interrupt status register 2 (RIS2) is set. Software must respond to this.
- A problem occurring during access to memory

The result of a failure in synchronization is the unread frame counter indicating that the corresponding descriptor chain may contain more available frames than it actually can. To retrieve the correct starting point for operations, use the descriptor base address load request (DLR.LBAq) for the given queue.

46.3.5 Transmission Control

Areas in the URAM for storing transmission descriptors must also be secured (for descriptors, see section 46.3.3 Descriptors).

The AVB-DMAC fetches data from the URAM in accord with the procedure the descriptor describes. The descriptor also retains tag information once the frame has been fetched for transmission. The tag information is used to maintain the relationships between status and time stamps for the software and the AVB-DMAC. The status and time stamp information for transmitted frames remains accessible after their transmission is completed.

(1) Transmission Modes

The AVB-DMAC has two modes of transmission.

- AVB transmission mode
This mode is selected by the priority level setting for the transmission queue in the transmit configuration register (setting of the TGC.TQP[1:0] bits) being B'01 or B'11.
- Non-AVB transmission mode
This mode is selected by the priority level setting for the transmission queue in the transmit configuration register (setting of the TGC.TQP[1:0] bits) being B'00.

(a) AVB Transmission Mode

AVB transmission supports the control of traffic through the output port to implement various traffic classes.

Support for Traffic Classes and Associated Priority

When transmission is in AVB transmission mode, streams of traffic are transmitted in accord with the part of the AVB specification called Forwarding and Queuing for Time Sensitive Streams (FQTSS; for details on this, see the IEEE 802.1Q standard).

In the AVB specification, at least one queue for a reserving stream under the Stream Reservation Protocol (SR stream) and at least one queue for a non-SR stream are present, and the queues for SR traffic is Highest priority queue.

The AVB-DMAC supports four traffic classes: SR class A, SR class B, network control (NC) traffic (gPTP frames), and best effort (BE) traffic. Allocating a specific queue to network control (NC) frames ensures the control of synchronization.

The AVB-DMAC realizes compliance with the AVB standards by handling queues with the following architecture (in terms of traffic classes).

- Four transmission queues (Q3, Q2, Q1, and Q0) are available.
- Q3 and Q2 are for SR streams (one each for class A and class B).
- Q1 is for low-bandwidth network control (NC) traffic (gPTP frames)
- Q0 is for other types of traffic (MSRPDU*¹, MVRPDU*², best effort (BE), etc.)

Notes: 1. MSRPDU: Multiple Stream Registration Protocol Data Unit

2. MVRPDU: Multiple VLAN Registration Protocol Data Unit

Fetching from queues proceeds in order of priority of the above traffic types. Three systems of priority are available through the setting of the transmit queue priority bits in the transmit configuration register (TGC.TQP[1:0]). In the default priority scheme, which is called AVB mode 1 (selected by TGC.TQP[1:0] = B'01), operation of the AVB-DMAC is fully in accord with the AVB specification. AVB mode 2 (transmit queue priority bits (TGC.TQP[1:0] = B'11) is an alternative priority scheme and varies from the AVB specification. Using this scheme thus requires more care. The other setting is for non-AVB-mode transmission.

Table 46.14 Default and Alternative Priority Orders in AVB Transmission Mode

| Priority Schemes (AVB Mode) | Priority Order of Queues |
|-----------------------------|---|
| Default | Q3 (SR class A) > Q2 (SR class B) > Q1 (NC) > Q0 (BE) |
| Alternative | Q1 (NC) > Q3 (SR class A) > Q2 (SR class B) > Q0 (BE) |

Transmission Selecting Algorithm and CBS

The algorithm the AVB-DMAC applies to select frames for transmission is in accord with the specifications under section 11.6.8, Transmission selection, of the IEEE 802.1Q standard. For AVB mode, the CBS (credit-based shaping) algorithm is applied to the class A and class B SR queues (Q3 and Q2). Use of the CBS enables correct handling of the priorities of transmission from the SR queues. For the CBS algorithm, see section 46.3.6 CBS (Credit-Based Shaping).

When the following conditions both hold, transmission from an SR queue (Q3 or Q2) proceeds at the specified time.

- The queue contains at least one frame ready for transmission.
- The queue has available credit.
- Unless an SR queue satisfies the above conditions, a higher priority queue is not present (not ready for transmission).

A non-SR queue (Q1 or Q0) is selected if the conditions below both hold.

- The queue contains at least one frame ready for transmission.
- As well as the above condition, a higher priority queue is not present (not ready for transmission).

Figure 46.27 and Figure 46.28 are flowcharts of selection for transmission in AVB mode 1 (default) and AVB mode 2 (alternative).

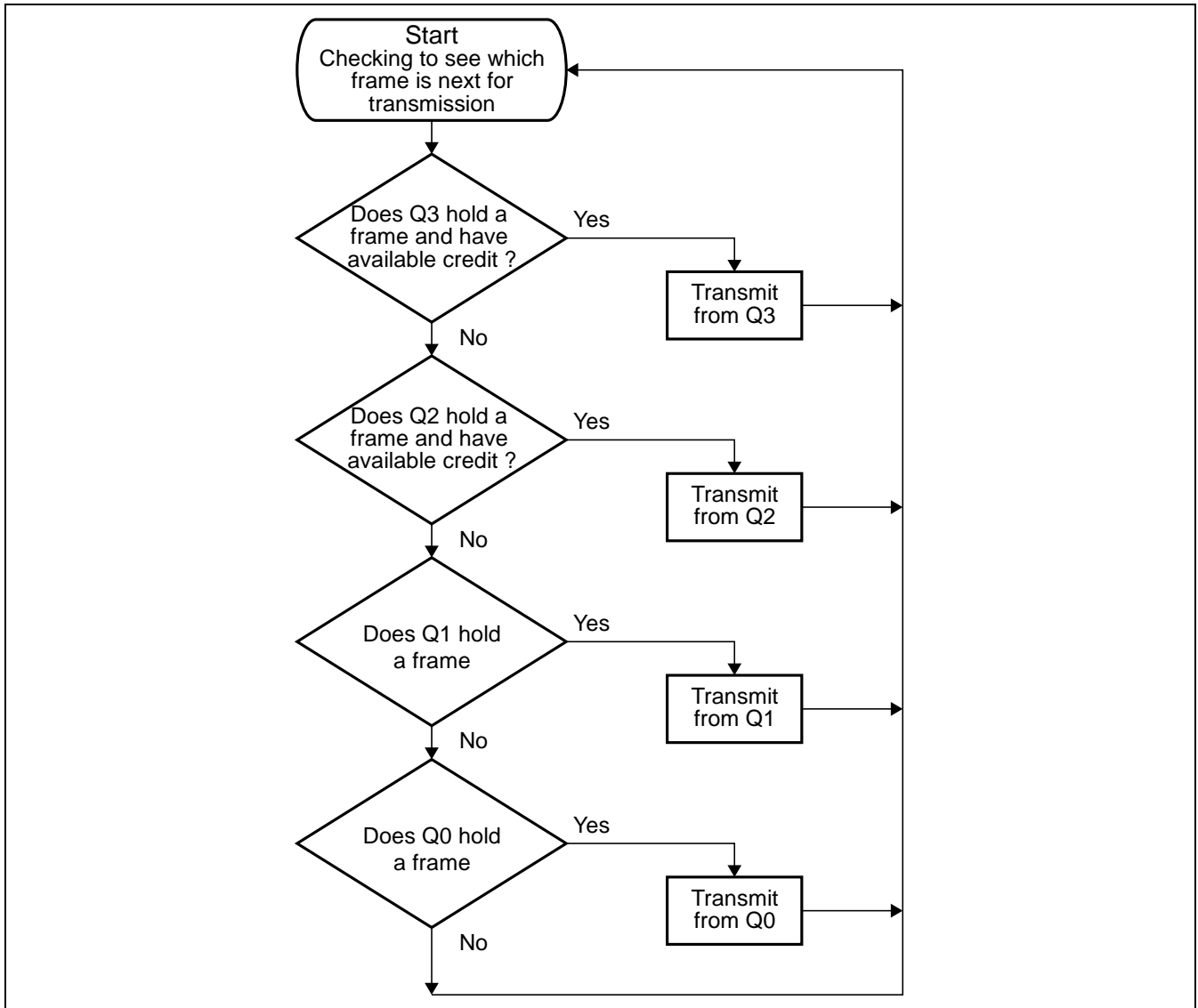


Figure 46.27 Flow of Selection for Transmission in AVB Mode 1 (Default)

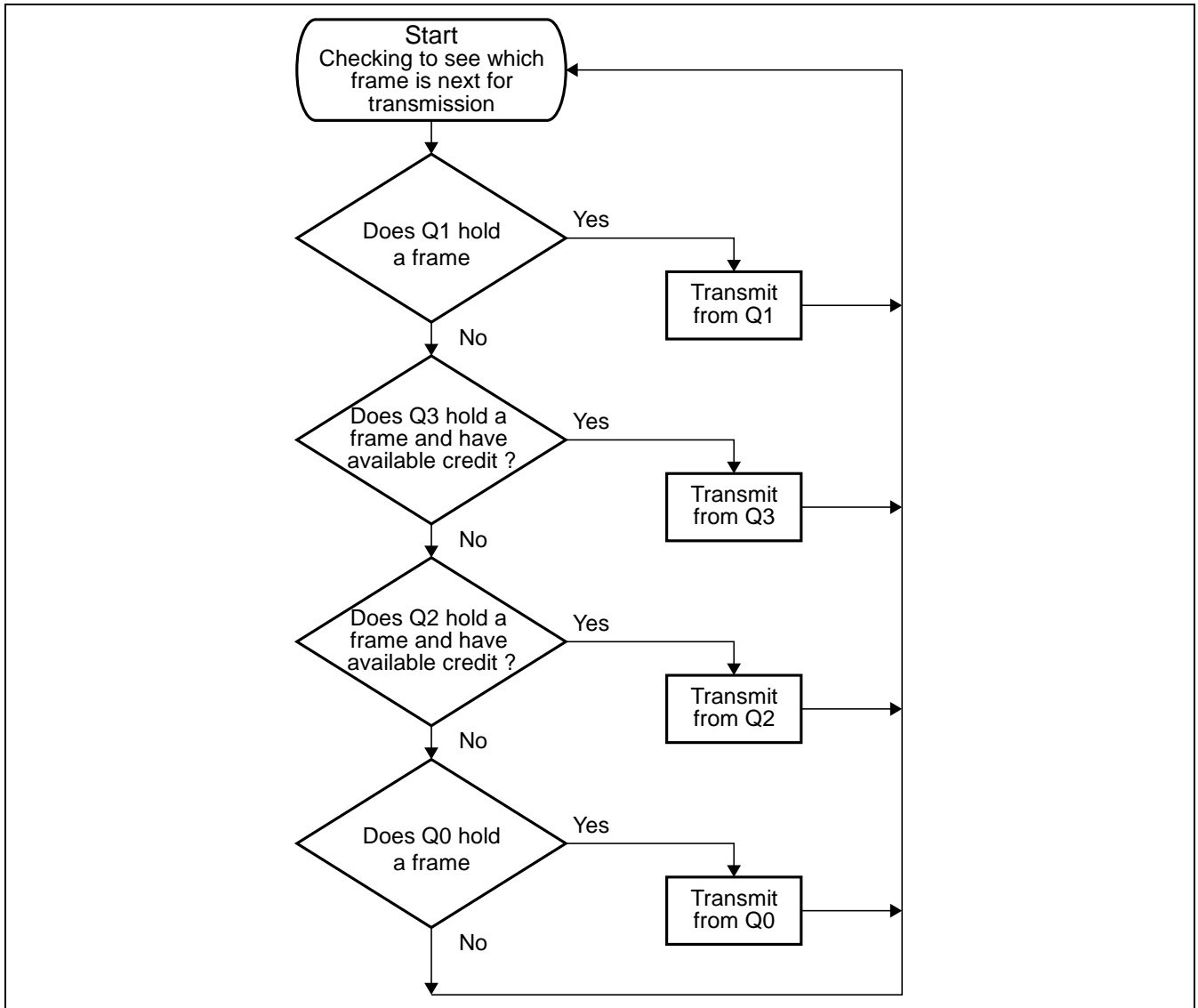


Figure 46.28 Flow of Selection for Transmission in AVB Mode 2 (Alternative)

(b) Non-AVB Transmission Mode

In non-AVB transmission mode, an absolute priority scheme is used. The SR class is not supported and the CBS algorithm is not used.

In non-AVB transmission mode (when the transmit queue priority bits in the transmit configuration register (TGC.TQP[1:0]) are B'00), data is fetched for transmission in a strict order of priority ($Q3 > Q2 > Q1 > Q0$).

Figure 46.29 shows the flow of selection in non-AVB transmission mode.

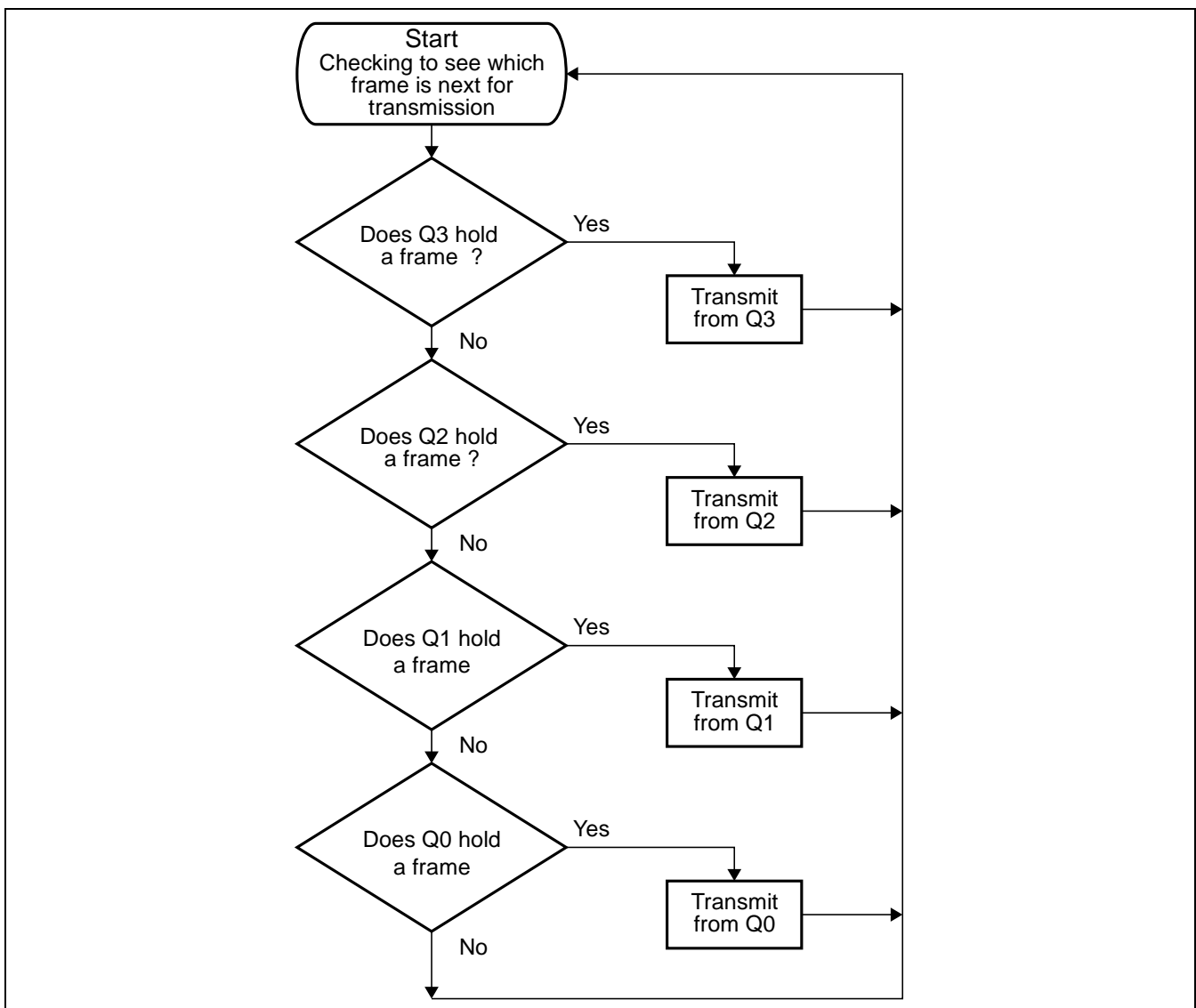


Figure 46.29 Flow of Selection for Transmission in Non-AVB Mode

(c) Setting the Size of the Transmission FIFO

The transmission FIFO is made up of 122 clusters. Each cluster can hold up to 128 bytes.

The size of the part of the transmission FIFO for use by each of the four transmission queues can be set by the corresponding transmit queue configuration q bits in the transmit control register (TGC.TBDq).

General Usage Examples:

Q0: Frames containing up to 1500 bytes $\rightarrow 1500/128 = 11.7 \rightarrow 12$ clusters

Q1: Frames containing up to 1024 bytes $\rightarrow 1024/128 = 8.0 \rightarrow 8$ clusters

Q3: Frames containing up to 1996 bytes $\rightarrow 1996/128 = 15.6 \rightarrow 16$ clusters

Q4: Frames containing up to 1996 bytes $\rightarrow 1996/128 = 15.6 \rightarrow 16$ clusters

When the depth of all transmission queues is 2, only the following number of clusters is required.

$$2 \times (12 + 8 + 16 + 16) + 16 = 2 \times 52 + 16 = 120$$

In the worst case (each of the transmission queues holds 1996 bytes), $2 \times 64 + 16 = 144$ clusters are required, so the maximum length of transmission frames must be less than 1996 bytes for some queues.

(2) Setting Up Transmission Descriptors

(a) Transmission Descriptor Type

The type of a descriptor is defined by the descriptor type (DESCR.DT) field.

Table 46.15 shows the descriptor types used in transmission.

Table 46.15 Descriptor Types in Transmission

| Descriptor Type (DESCR.DT) | Operation | Write-back |
|-----------------------------------|--|-------------------|
| Frame Start (FSTART) | The AVB-DMAC fetches the first of the data for the divided frame and proceeds to processing of the next descriptor. | FEMPTY |
| Frame Middle (FMID) | The AVB-DMAC fetches the second or subsequent data for the divided frame and proceeds to processing of the next descriptor. | FEMPTY |
| Frame End (FEND) | The AVB-DMAC fetches the last of the data for the divided frame. The frame of data that has been fetched to the transmission FIFO is ready for transmission by the E-MAC, and then the AVB-DMAC proceeds to processing of the next descriptor. | FEMPTY |
| Frame Single (FSINGLE) | The AVB-DMAC fetches the frame of data. The frame of data that has been fetched to the transmission FIFO is ready for transmission by the E-MAC, and then the AVB-DMAC proceeds to processing of the next descriptor. | FEMPTY |
| Link (LINK) | Processing proceeds to the descriptor specified by DESCR.DPTR. | LEEMPTY |
| Fixed Link (LINKFIX) | Same as LINK | Not changed |
| End Of Set (EOS) | This is a transmission stop point defined by software This leads to clearing of the transmit start request bit (TCCR.TSRQt), which stops transmission. When the TCCR.TSRQt is again set to 1 (a new transmission start request is issued), processing proceeds to the next descriptor. | EEMPTY |
| Frame Empty (FEMPTY) | No frame data are ready for transmission This leads to clearing of the transmit start request bit (TCCR.TSRQt), which stops transmission. When the TCCR.TSRQt is again set to 1 (a new transmission start request is issued), processing starts at this descriptor. | Not changed |
| Link Empty (LEEMPTY) | Same as FEMPTY | Not changed |
| EOS Empty (EEMPTY) | Same as FEMPTY | Not changed |

(b) Configuration of Transmission Frame Data Descriptors

Figure 46.30 shows the configuration of descriptors for use with transmission queues. The transmission-specific fields (DESCR.MSR, DESCR.TSR, and DESCR.TAG) are described in Table 46.16.

For the other fields and the descriptor types, see section 46.3.3(6) Descriptor Type.

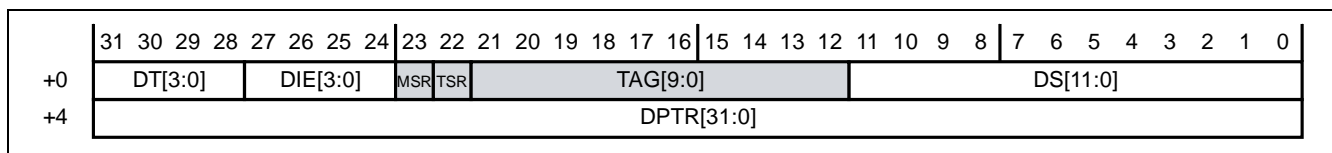


Figure 46.30 Configuration of Descriptor for a Transmitted Frame

Table 46.16 Configuration of a Transmission Descriptor

| Bit Name | Function |
|----------|---|
| MSR | <p>MAC Status storage Request</p> <p>This bit configures if E-MAC transmission status of the frame is stored in E-MAC status FIFO.</p> <p>0: No status information is stored</p> <p>1: Status information is to be kept for this frame in MAC status FIFO</p> <p>This bit is considered only if DESCR.DT = FEND or FSINGLE.</p> <p>Note: If there is error during transmission of frame by E-MAC then status is always stored.</p> |
| TSR | <p>Time Stamp Store Request</p> <p>This bit specifies whether the transmission time stamp is to be stored within the EthernetAVB-IF module.</p> <p>0: The time stamp status FIFO within the EthernetAVB-IF module does not retain a transmission time stamp.</p> <p>1: The time stamp status FIFO within the EthernetAVB-IF module retains a transmission time stamp.</p> <p>This bit is available while the current DESCR.DT is FEND or FSINGLE.</p> |
| TAG | <p>Frame Tag</p> <p>This TAG field is used to associate each frame data with a time stamp. Frame TAG is not required but is recommended.</p> <p>This bit is available while the current DESCR.DT is FEND or FSINGLE.</p> |

For the time stamp FIFO function, see section 46.3.5(4) Time Stamping in Transmission.

(3) Transmission

(a) Transmitting Frames

Setting the transmit start request bit in the transmit configuration control register (TCCR.TSRQt) starts the transfer of frames from the corresponding transmission queue.

The descriptor in the current descriptor address (CDARq.CDA) for the queue t (with q = t) is read first.

If this descriptor is a descriptor for frame transmission (FSINGLE, etc.), the AVB-DMAC fetches the frame data from the data area indicated by the descriptor, writes FEMPTY back to the descriptor type (DESCR.DT) bits to indicate completion of this processing, then proceeds to processing of the next descriptor.

If the descriptor is not for transmission, processing is as dictated by the given descriptor (for these descriptors, see the descriptions in section 46.3.3 Descriptors).

If a base address load request is issued for a descriptor chain while it is being processed (by setting 1 in the LBAq bit for transmission queue q that is currently being processed in the descriptor base address load request register, DLR), processing proceeds to the new descriptor chain. Changing the chain does not interrupt frame fetching, but note that frames that have not been fetched from the old chain remain where they are.

Figure 46.31 shows descriptor processing during transmission.

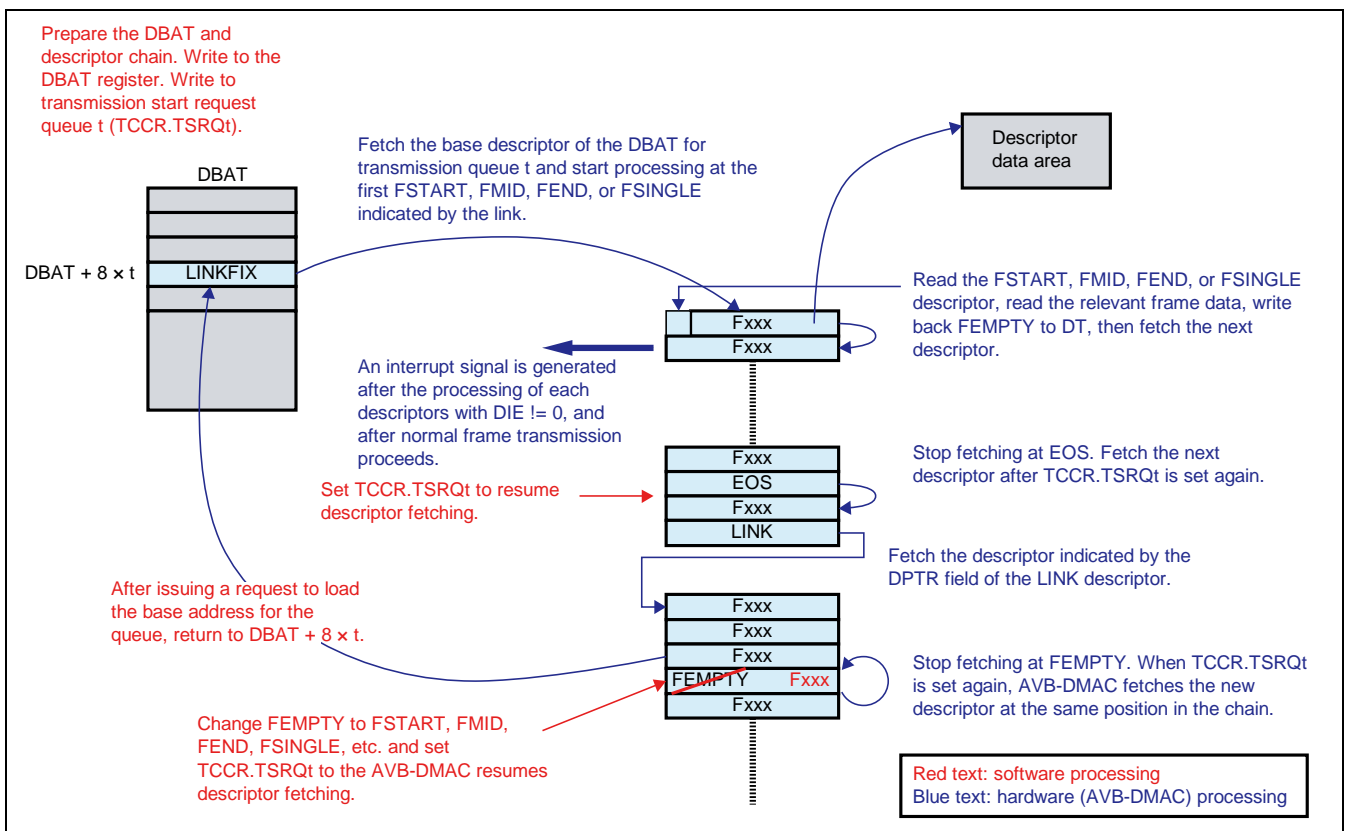


Figure 46.31 Descriptor Processing During Transmission

(b) Examples of Descriptor Usage

Immediate Frame Transmission

Immediate frame transmission is a pattern in which fetching by the AVB-DMAC starts whenever software adds data to a queue. FEMPTY descriptors are used as stop points to keep the hardware and software in synchronization.

Create descriptor chains that have FEMPTY descriptors at the stop points.

Figure 46.32 shows the flow for software implementing this pattern. SW should read back written descriptor before changing FEMPTY descriptor before writing to TCCR.TSRQt.

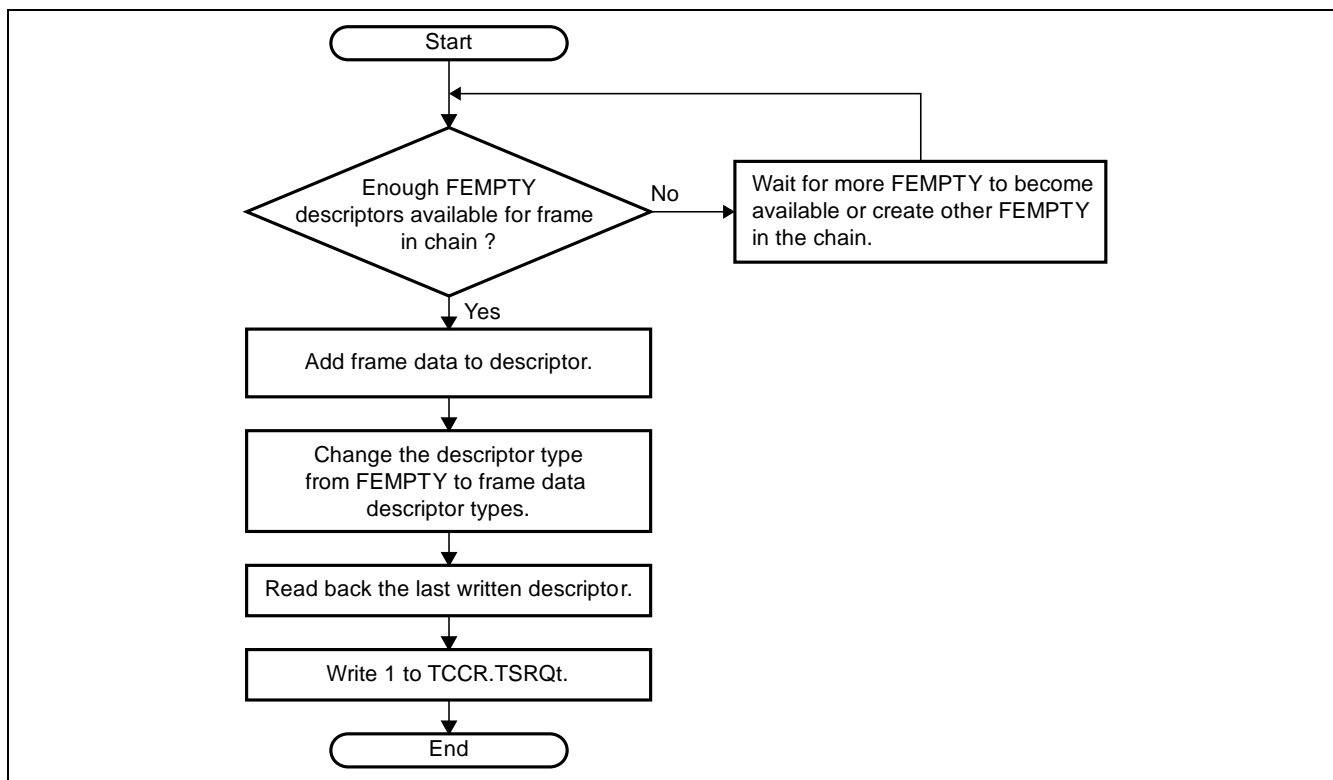


Figure 46.32 Software Flow for Immediate Frame Transmission

Figure 46.33 shows software and AVB-DMAC operations for immediate frame transmission.

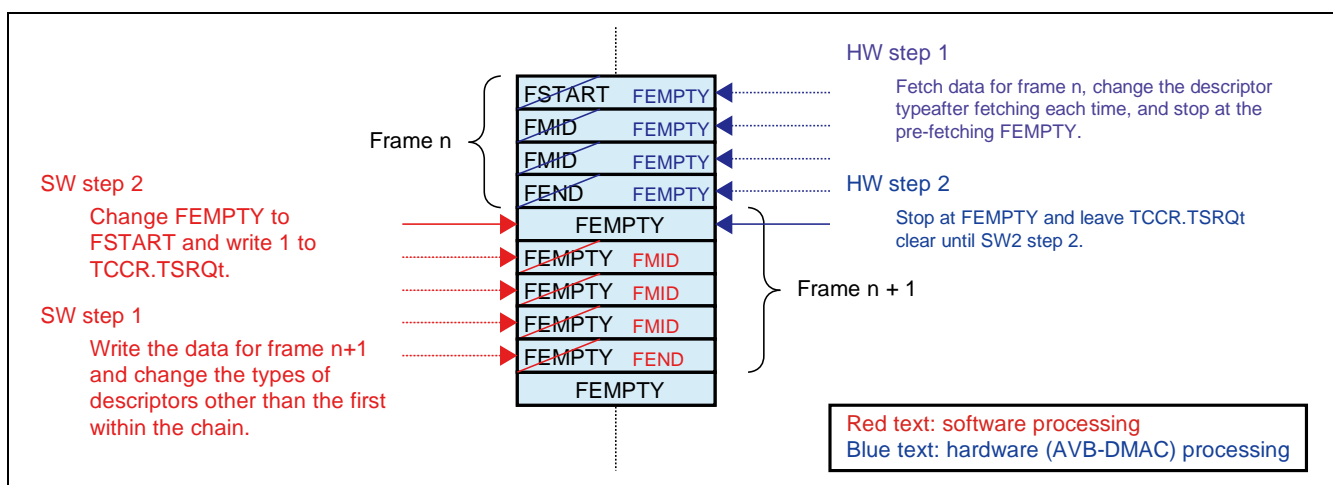


Figure 46.33 Software and AVB-DMAC Operations for Immediate Frame Transmission

Frame Set Transmission with Changing of the Active Descriptor Chain

This pattern is used when data are transmitted with a delay for software control to secure bandwidth or for other reasons, rather than immediately transmitted. EOS descriptors are used for the stop points. Start by creating a descriptor chain that has a FEMPTY descriptor as its stop point.

Figure 46.34 shows the software flow in this pattern.

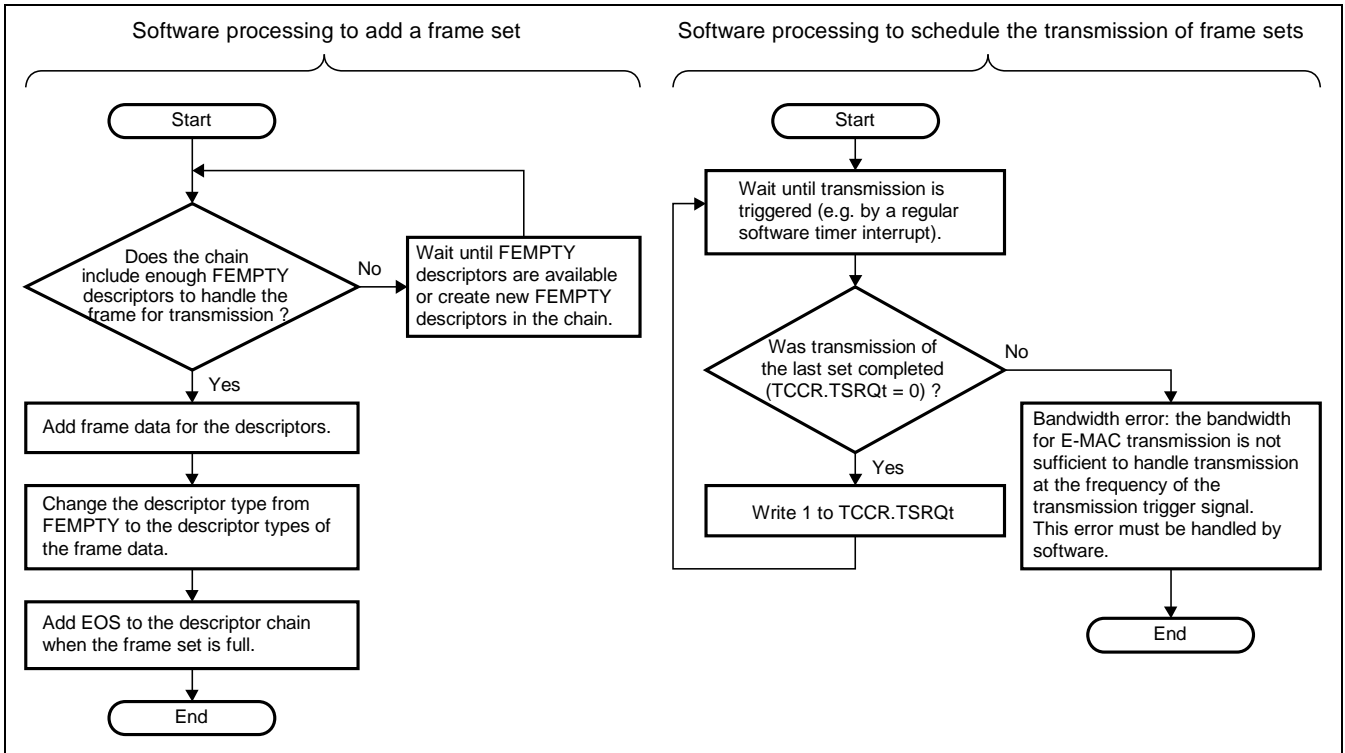


Figure 46.34 Software Flow for Frame Set Transmission with Changing of the Active Descriptor Chain

Figure 46.35 shows software and AVB-DMAC operations for frame set transmission.

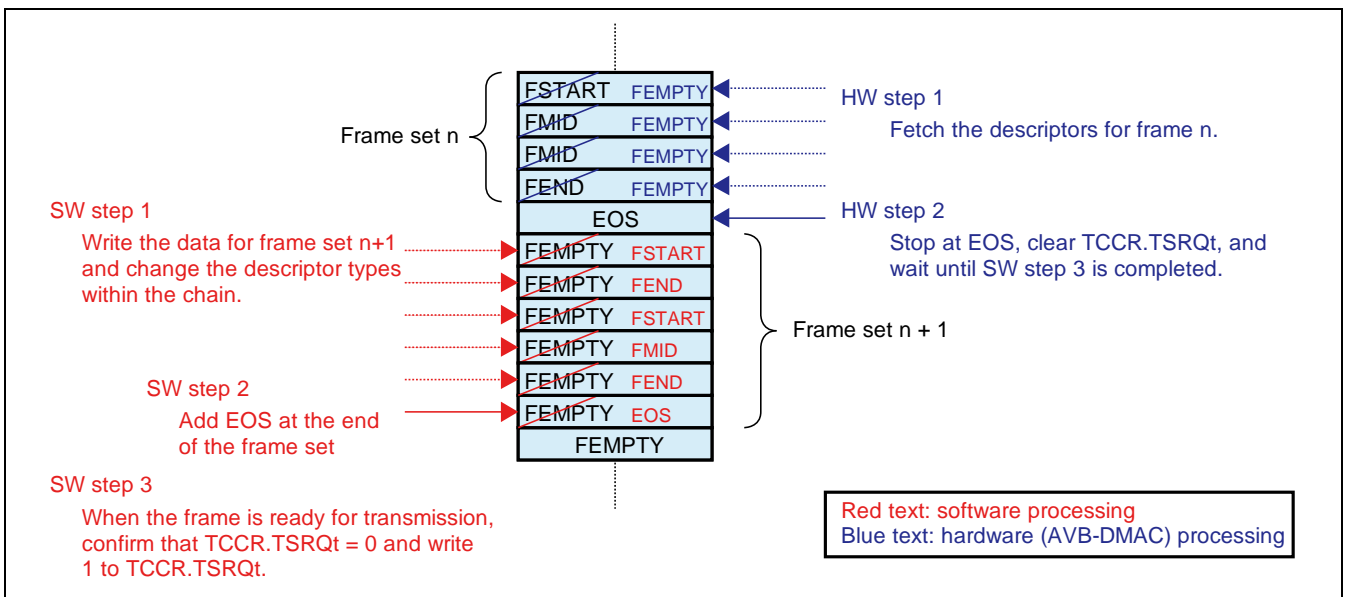


Figure 46.35 SW and AVB-DMAC Operations for Frame Set Transmission with Changing of the Active Descriptor Chain

Frame Set Transmission Using a Shadow Descriptor Chain

This pattern is used when data are transmitted with a delay for software control to secure bandwidth or for other reasons, rather than immediately transmitted. Two or more descriptor chains are used. The chains are classified as active or shadow chains. EOS descriptors are used for the stop points.

Create descriptor chains that have FEMPTY descriptors at the stop points.

Figure 46.36 shows the flow for software implementing this pattern.

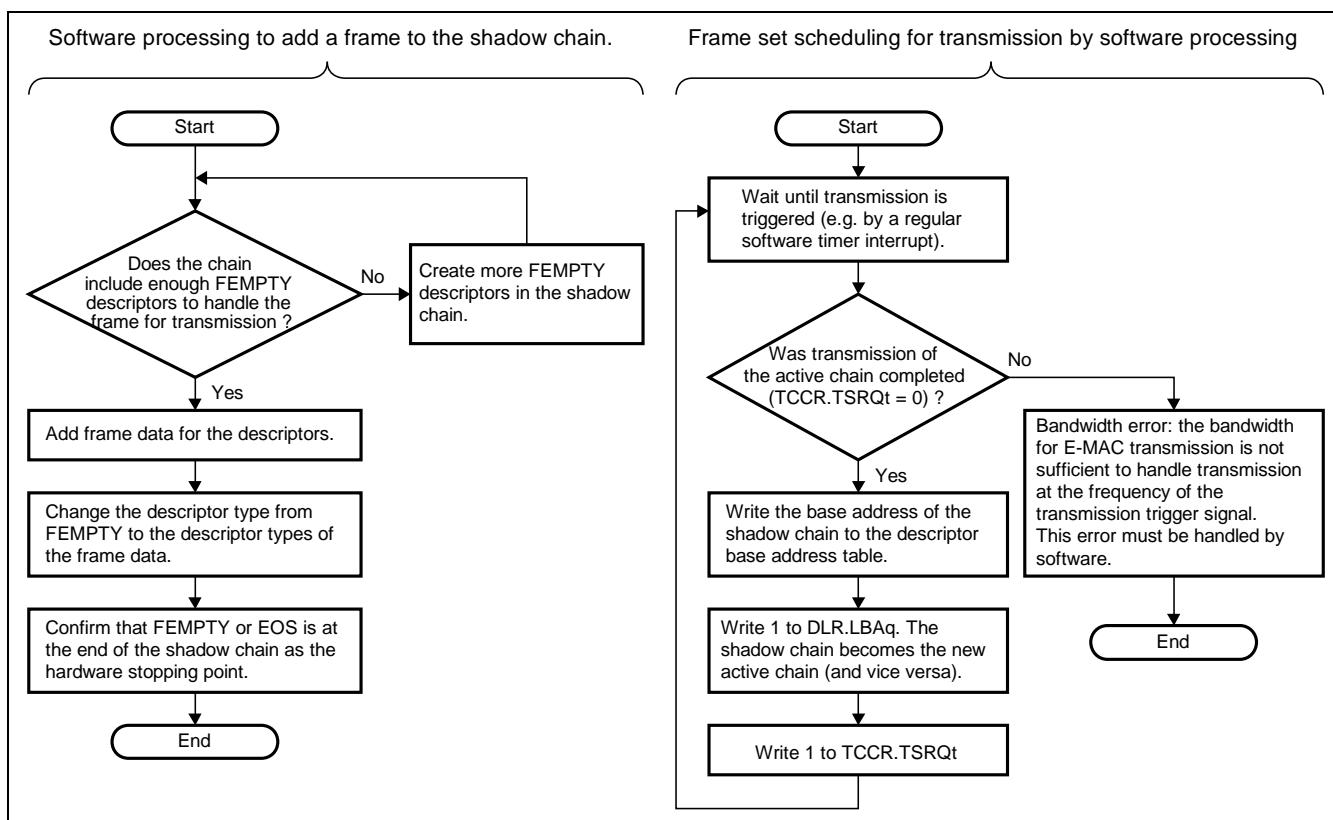


Figure 46.36 Software Flow for Frame Set Transmission Using the Shadow Descriptor Chain

Figure 46.37 shows software and AVB-DMAC operations for frame set transmission.

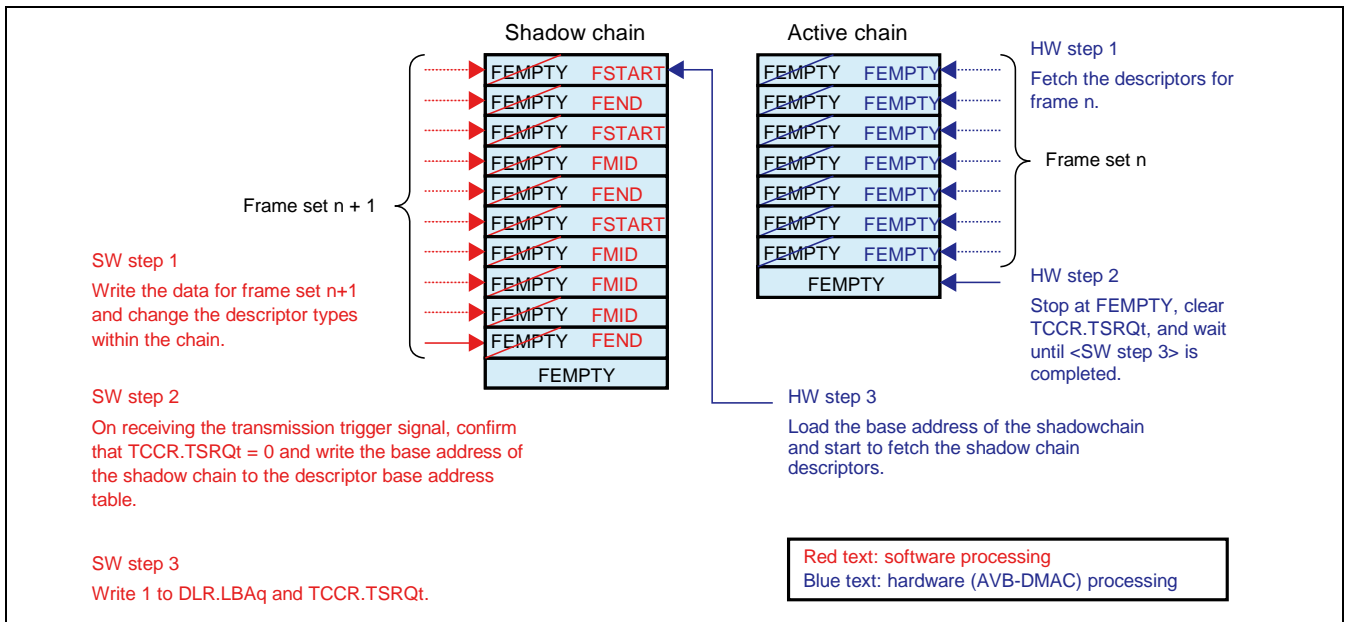


Figure 46.37 SW and AVB-DMAC Operations for Frame Set Transmission Using the Shadow Descriptor Chain

(4) Time Stamping in Transmission

Transmission time stamps are important in satisfying the requirements for timing and synchronization of the IEEE 802.1AS standard. Reference to this information can also be useful to other applications and in testing. The AVB-DMAC supports the storage of time stamps for transmitted frames. The time-stamp values are based on the gPTP timer and are captured at the same time as sending of the Start of Frame Delimiter (SFD) for transmitted frames.

When the time stamp storage request field (DESCR.TSR) is set to 1, selecting storage of a time stamp, the tag number defined in the tag field (DESCR.TAG) of the last descriptor (FEND or FSINGLE) for the frame being transmitted is stored with the time stamp. This eases identification and association. The time-stamp FIFO is accessible at any time.

Figure 46.38 shows the mechanism supporting transmission time stamping.

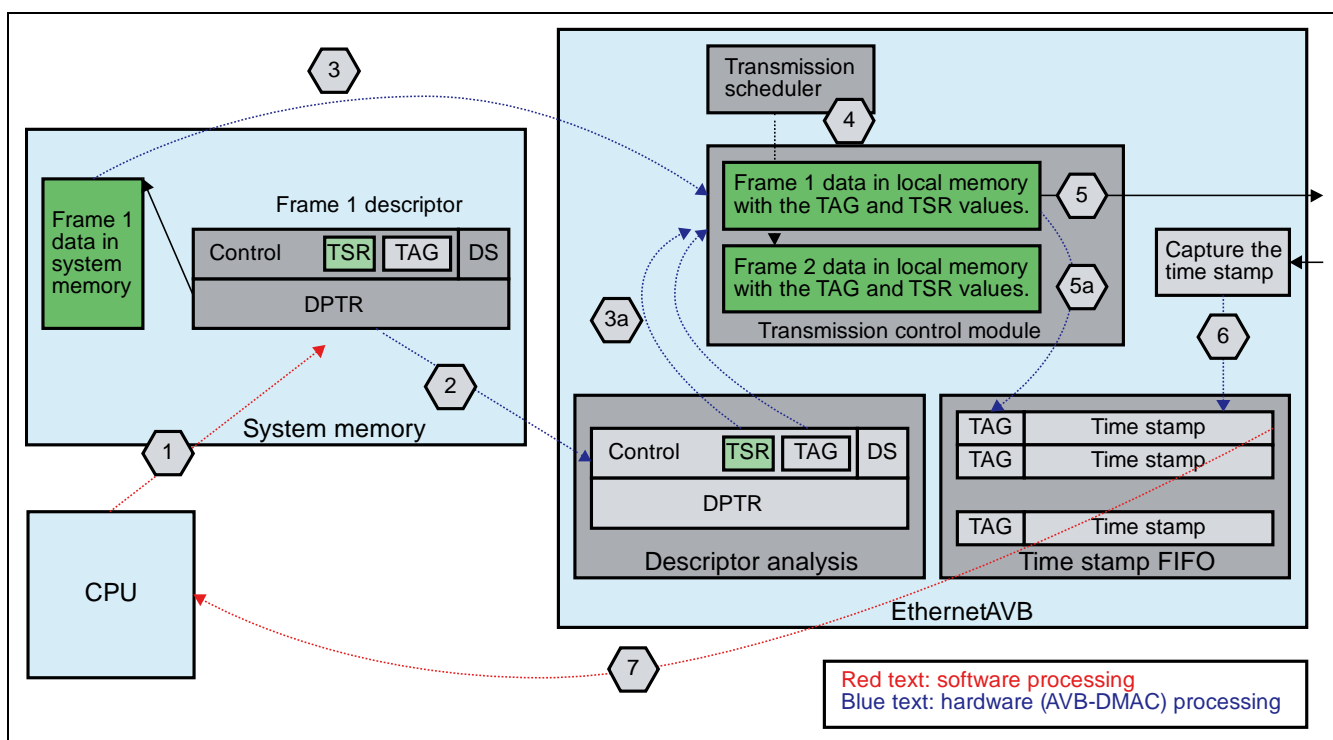


Figure 46.38 Mechanism to Support Transmission Time Stamps

The method of using this function is described below:

- Secure space in the URAM for the frame requiring time stamping.
Write the tag number of the frame to the frame tag field (DESCR.TAG) and set the time stamp storage request field (DESCR.TSR) to 1.
- The AVB-DMAC fetches and analyzes the descriptor. The time stamp storage request field (DESCR.TSR) is 1, so it recognizes that transmitting this frame also requires storage of the time stamp.
- The AVB-DMAC fetches the data for frame 1 and temporarily stores the frame in internal memory for scheduling.
3a: The frame tag field (DESCR.TAG) and time stamp storage request field (DESCR.TSR) are stored with the fetched data.)
- Under the control of priority settings according to credit-based shaping (CBS) or another scheme, the transmission scheduler decides it is time to transmit frame 1.
- Transmission of frame 1 starts.
5a: Frame 1's tag is stored in the time-stamp FIFO.
- The gPTP time stamp is captured at the start of frame delimiter (SFD) for transmission and stored with the tag in the time-stamp FIFO when the frame is completely transmitted. An interrupt is generated to indicate existence of new timestamp information. For this to happen, the descriptor interrupt control register (DIC) must be set beforehand.
- The entry can now be read from the time-stamp FIFO.

Use the time-stamp FIFO for the timing and synchronization of frames with IEEE 802.1AS compliance.

Time stamping can also be used with other frames, but take care not to allow the time-stamp FIFO to overflow. When the FIFO is full, further time stamps supplied to it are lost.

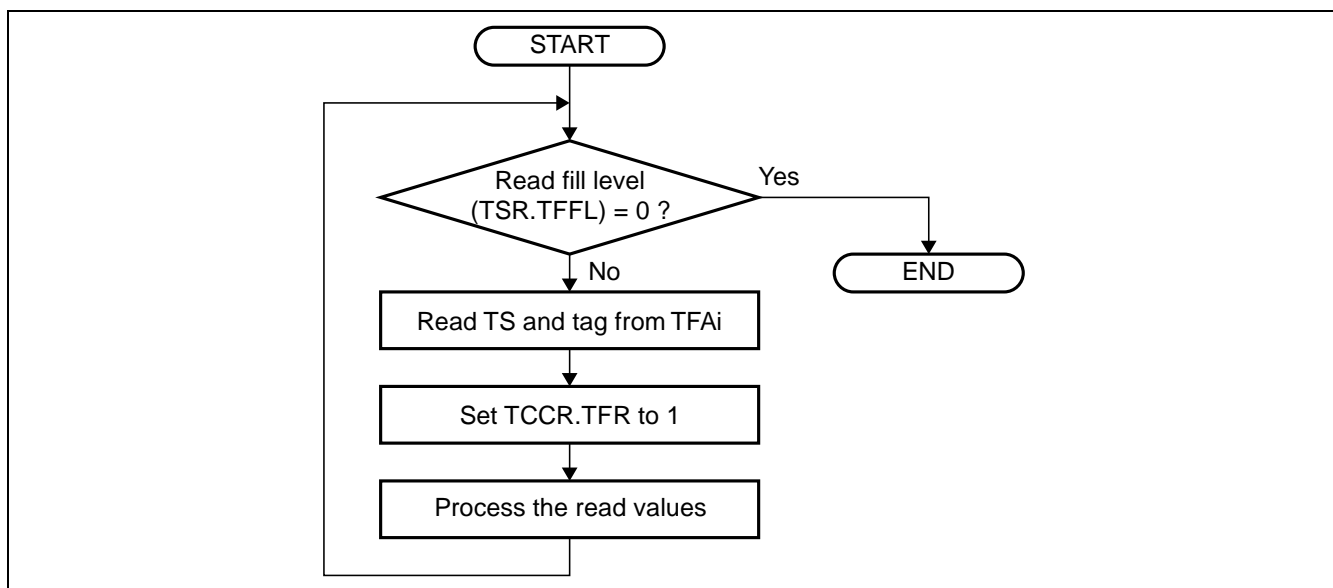


Figure 46.39 Flow of Transmission Time Stamping

(5) Ending Transmission

Figure 46.40 shows the procedure for ending transmission.

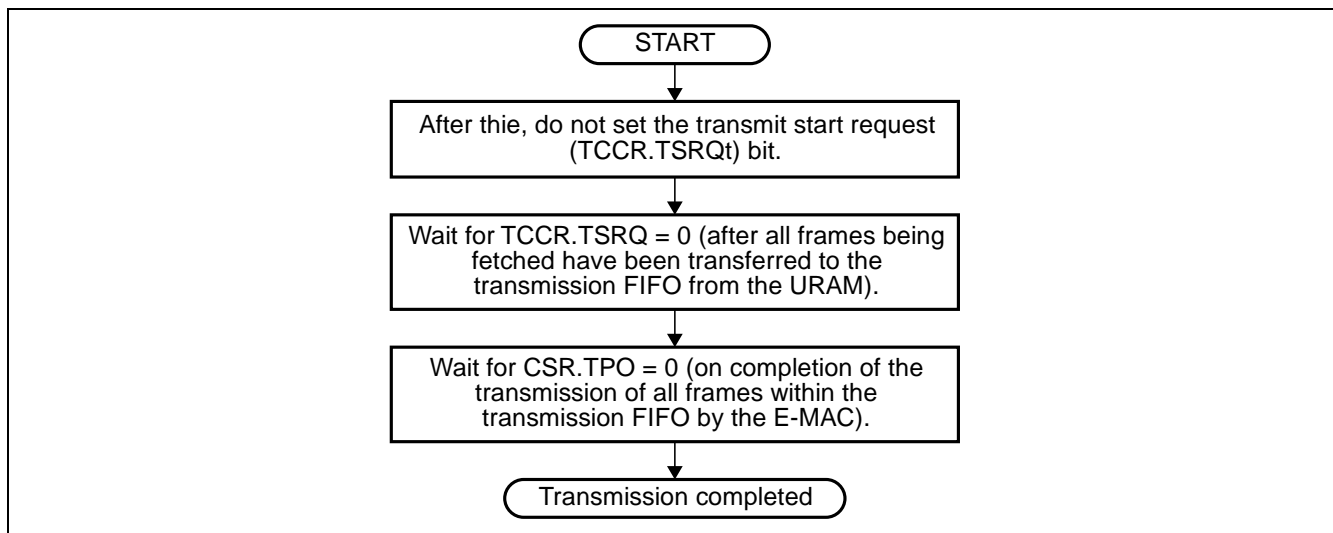


Figure 46.40 Procedures for Ending Transmission

46.3.6 CBS (Credit-Based Shaping)

In AVB transmission mode (i.e. when the transmit queue priority field in the transmit configuration register (TGC.TQP) is B'01 or B'11), transmission queues Q3 and Q2 are respectively assigned to class A and class B stream traffic and the CBS (Credit Based Shaping) algorithm is used to select the transmission queues in order to satisfy the Forwarding and Queuing for Time Sensitive Streams (FQTSS) specification (see section 11.6.8 or section 35 in IEEE 802.1Q).

The CBS algorithm is based on the concept of transmission credit for each queue. Credit can be thought of as the degree to which a queue has the “right” to transmit at a given time. Actually, in AVB transmission mode as specified in IEEE 802.1Q, queues that are subject to the CBS algorithm are able to transmit when the following conditions are met.

At least one frame is stored in the queue.

The credit for the queue is 0 or a positive value.

The credit for a transmission queue is incremented while one or more frames from the queue are present in the transmission FIFO but transmission of these frames is not proceeding. This state is indicated by the transmission process status bit for queue t in the AVB-DMAC status register (CSR.TPOt). The credit is decremented while transmission of a frame from the queue is in progress. This mechanism is used to control transmission so that the transmission of frames from the queues for each of the traffic classes does not exceed the specified maximum bandwidths.

IEEE 802.1Q defines the following parameters for queues under the control of the CBS algorithm.

portTransmitRate: Maximum transmission data rate of an external port. The E-MAC determines this parameter.

bandwidthFraction: Maximum fraction of portTransmitRate that can be used for a queue.

idleSlope: Rate of change of credit for a queue when transmission of frames from the queue is not proceeding so the credit value (in bits per second) is increasing. idleSlope is also equal to the maximum fraction of the total bandwidth (portTransmitRate) that is available to the given queue under a specified condition (frames from the queue can be placed in a continuous stream. See Annex L of IEEE 802.Q.

$$\text{idleSlope} = \text{bandwidthFraction} \times \text{portTransmitRate}$$

sendSlope: Rate of change of credit for a queue while transmission of a frame from the queue is in progress so the credit value is decreasing.

The value of sendSlope is defined as follows:

$$\text{sendSlope} = \text{idleSlope} - \text{portTransmitRate}$$

Furthermore, the values below are used to define individual traffic classes (or queues for the classes) under control of the algorithm. See Annex L of IEEE 802.Q.

maxFrameSize: Maximum size of frames (in bits) of the corresponding traffic class that can be transmitted from a port

maxInterferenceSize: Maximum burst size (in bits) by which delays for the corresponding traffic class can be allowed

hiCredit: Maximum credit value (positive number). Can be calculated by using the following equation: $\text{hiCredit} = \text{maxInterferenceSize} \times (\text{idleSlope} / \text{portTransmitRate})$

loCredit: Minimum credit value (negative number). Can be calculated by using the following equation:

$$\text{loCredit} = \text{maxFrameSize} \times (\text{sendSlope} / \text{portTransmitRate})$$

Figure 46.41 shows how the CBS algorithm works and the meaning of the above parameters.

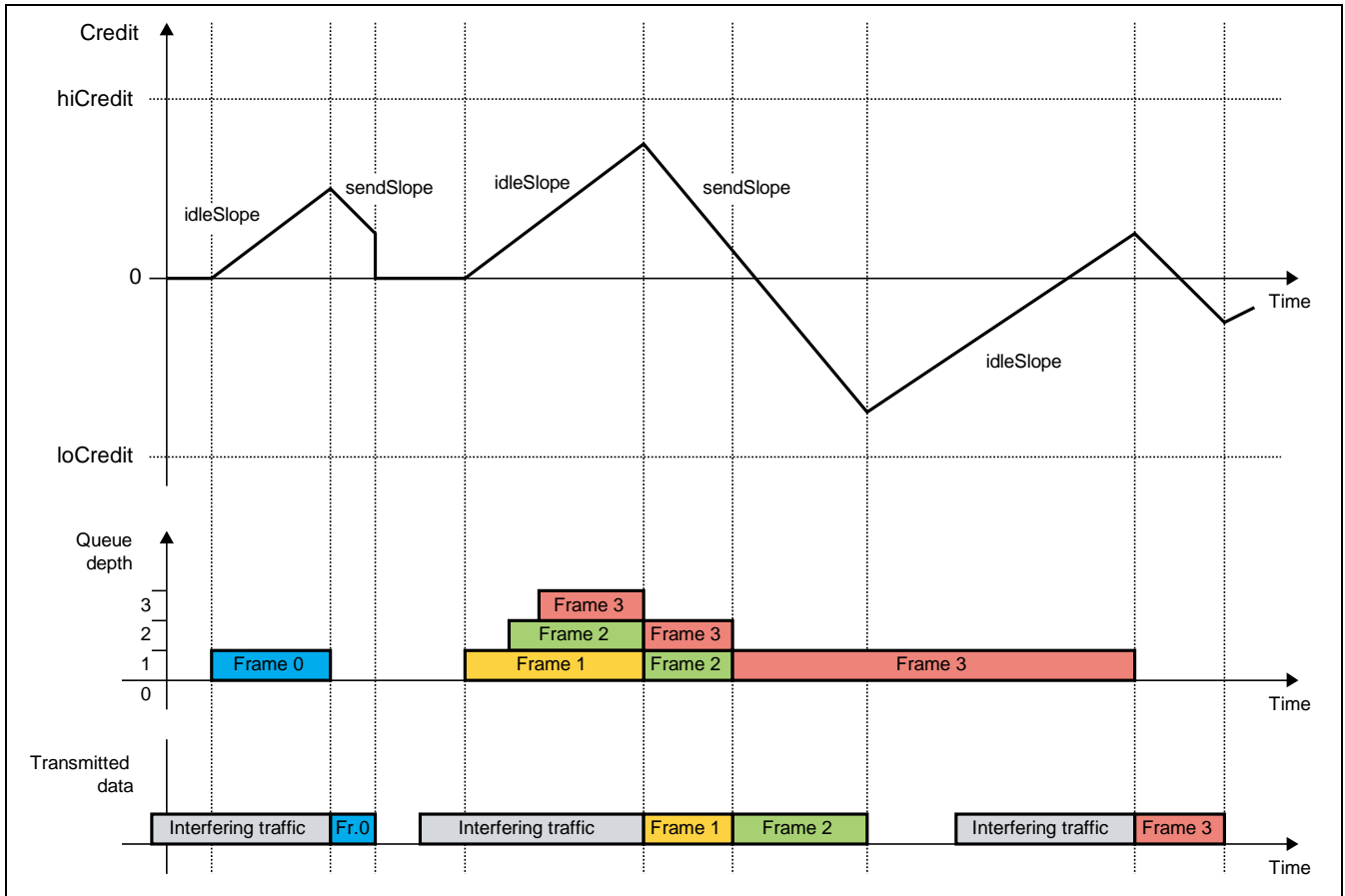


Figure 46.41 CBS (Credit-Based Shaping) Operation

Figure 46.42 shows the implementation of CBS in the AVB-DMAC.

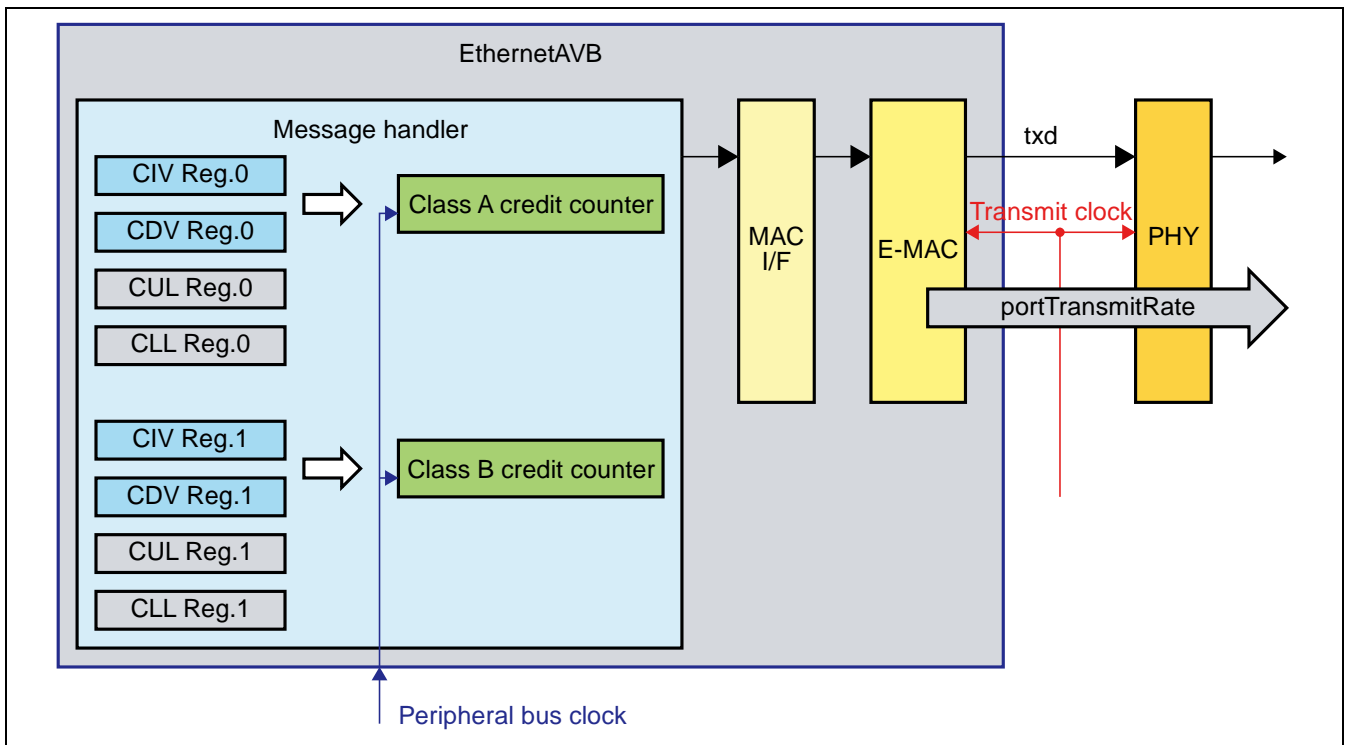


Figure 46.42 CBS (Credit-Based Shaping) Operation in the AVB-DMAC

The above implementation is based on “credit counters” for the respective traffic classes (SR class A and class B). The following parameters apply for these classes.

CBS increment value (CIV): Signed positive number

The credit is incremented by this amount every high-speed peripheral bus clock cycle while a frame from the queue is pending but transmission has not started (idleSlope).

CBS decrement value (CDV): Signed negative number

The credit is decremented by this amount every high-speed peripheral bus clock cycle while transmission of a frame from the queue is proceeding (sendSlope).

The CBS increment value (CIV) and CBS decrement value (CDV) are defined as follows.

$$\begin{aligned} \text{CIV} &= \text{idleSlope} \times \text{Mfactor} \\ \text{CDV} &= \text{sendSlope} \times \text{Mfactor} \end{aligned}$$

Mfactor is a multiplier factor to ensure accuracy for CIV and CDV. CIV and CDV are calculated by using the following equations.

$$\begin{aligned} \text{CIV} &= (\text{portTransmitRate}/\text{HP}\phi) \times \text{bwFraction} \times \text{Mfactor} \\ \text{CDV} &= (\text{portTransmitRate}/\text{HP}\phi) \times (\text{bwFraction} - 1) \times \text{Mfactor} \end{aligned}$$

HP ϕ is the frequency of the high-speed peripheral bus clock. The credit counters are driven by the high-speed peripheral bus clock, so calculating the slope parameters for CBS requires (1/HP ϕ).

Use software to prepare Mfactor for the CBS parameters. All queues for the same class must have the same Mfactor for all CBS parameters. Mfactor for a specified class c can be changed during operation, unless transmission is pending for that class (i.e. the transmit process status bit in the AVB-DMAC status register (CSR.TPOt) = 0). At that time, the credit counter values for class A and class B are 0. Note that the credit value will not match a new incrementation or decrementation parameter if Mfactor is changed while the credit counter value is non-zero. Mfactor is not present in the AVB-DMAC registers.

Set the CIV and CDV parameters in the CBS increment value registers (CIVRc) and the CBS decrement value registers (CDVRc). These are treated as dynamic settings since they should be updated when streams are registered and erased in accord with IEEE 802.1Qat.

The AVB-DMAC also has CBS upper limit registers (CULc) (the upper limit registers for classes A and B) and CBS lower limit registers (CLLc) (the lower limit registers for classes A and B). Set Mfactor to match the credit value and set the upper limit (hiCredit) and the lower limit (loCredit) for each class as defined above.

$$\begin{aligned} \text{CUL} &= \text{hiCredit} \times \text{Mfactor} = \text{maxInterferenceSize} \times \text{bwFraction} \times \text{Mfactor} \\ \text{CLL} &= \text{loCredit} \times \text{Mfactor} = \text{maxFrameSize} \times (\text{bwFraction} - 1) \times \text{Mfactor} \end{aligned}$$

Example:

Assume that portTransmitRate = 100 Mbps, HP ϕ = 133.33 MHz and bwFraction = 3%. Then idleSlope and sendSlope represented as one bit vs. cycles of the high-speed peripheral bus clock are as follows.

$$\begin{aligned} \text{idleSlope} &= (\text{portTransmitRate}/\text{HP}\phi) \times \text{bwFraction} = 100/133.33 \text{ (Mbps/MHz)} \times 3\% = 0.023 \text{ bit per high-speed peripheral bus clock cycle} \\ \text{sendSlope} &= \text{idleSlope} - (\text{portTransmitRate} / \text{HP}\phi) = -0.727 \text{ bits per high-speed peripheral bus clock cycle} \end{aligned}$$

Let Mfactor be 100, then CIV and CDV parameters are determined as follows.

$$\begin{aligned} \text{CIV} &= \text{idleSlope} \times \text{Mfactor} = 2.3 \\ \text{CDV} &= \text{sendSlope} \times \text{Mfactor} = -72.7 \end{aligned}$$

(1) Restrictions on CIV, CDV and Mfactor

The maximum value (the minimum value for negative numbers) up to which the credit counter will not overflow determines the maximum values of CIV and CDV that can be set in the CBS registers. This maximum credit value is equivalent to the worst case of the hiCredit value, and the maximum values for class A and class B are calculated as follows.

<Conditions>

- Class A maximum value (hiCredit_max_classA)

classA bwFraction \cong 100%

Maintaining the proper relations in the transmission priority order requires waiting for a period equivalent to the maximum frame size.

hiCredit_max_classA \cong maxInterferenceSize for class A = Interference due to one max. sized frame = header + max. size payload + CRC (2000 bytes) + preamble (8 bytes) + IFG (12 bytes) + processing_delay (\cong 80 bytes) \cong 2100 bytes

- Class B maximum value (hiCredit_max_classB)

classB bwFraction \cong 100%

Maintaining the proper relations in the transmission priority order requires waiting for a period equivalent to the maximum size of frames in the class A transmission queue and other transmission queues.

hiCredit_max_classB \cong maxInterferenceSize for class B = Interference due to two max-size frames = $2 \times$ hiCredit_max_classA \cong 4200 bytes

hiCredit_max_classA = 16800

hiCredit_max_classB = 33600

The maximum values that can be selected with Mfactor for the 32-bit signed counter without overflow are:

Mfactor_max_classA = $2^{31}-1 / \text{hiCredit_max_classA} \cong 127826$ and

Mfactor_max_classB = $2^{31}-1 / \text{hiCredit_max_classB} \cong 63913$.

A high degree of accuracy can be achieved even with a low bandwidth. In class B, bandwidthFraction = 0.05% and the bandwidth error < 0.1%.

The maximum value of CIV is calculated from the following equation.

$\text{CIV} = \text{idleSlope} \times \text{Mfactor} = (\text{portTransmitRate} / \text{HP}\phi) \times \text{bandwidthFraction} \times \text{Mfactor}$

When Mfactor is the maximum value and bandwidthFraction is the maximum value (up to 100%):

CIV_max_classA = $(\text{portTransmitRate} / \text{HP}\phi) \times \text{Mfactor_max_classA}$ and

CIV_max_classB = $(\text{portTransmitRate} / \text{HP}\phi) \times \text{Mfactor_max_classB}$.

The maximum values when portTransmitRate = 1000 Mbps and HP ϕ = 133.33 MHz are as follows:

CIV_max_classA \cong 958719

CIV_max_classB \cong 479359

These values in the table are the limits of CIV up to which the 32-bit credit counter will not overflow. The CIV parameters are implemented as 16 bits + a sign bit, so a further limit of CIV \leq 65535 applies to both class A and class B.

(2) Credit Incrementation during Inter-Frame Gaps (IFGs)

The inter-frame gap (IFG) after a frame is transmitted is not treated as part of frame transmission by the CBS credit counter. During an IFG, the credit is incremented for all SR queues that have pending frames or negative credit. Figure 46.43 illustrates credit operations during IFGs.

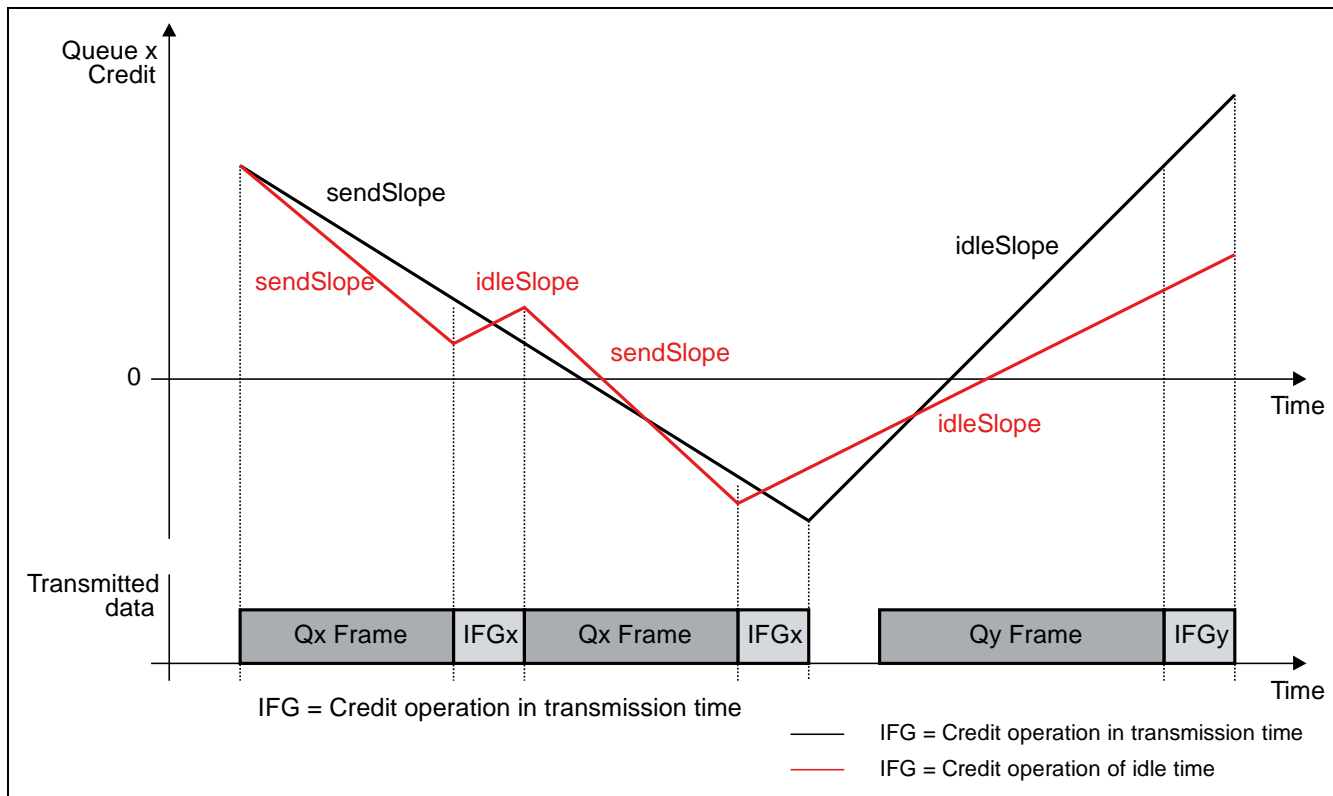


Figure 46.43 Credit Operations during IFGs

Accordingly, the IFG need not be included in calculation of the bandwidth requirements for the specified SR class when deciding the idleSlope, sendSlope, and CIV and CDV parameters. However, IFG must also be included in the calculation in order to confirm that the total bandwidth allocated to all SR classes does not exceed 100% of portTransmitRate. This is described in Section 36.2.2.8.4 of IEEE 802.1Q.

(3) Example

The case of a class A 48-kHz stereo audio stream among Ethernet frames is described as an example.

After every class A measurement interval (125 μ s), 80 octets consisting of two sets of six 32-bit samples plus a 32-octet header are stored as audio data within a frame. The IEEE 802.3 also imposes a 42-octet media-specific framing overhead (an 8-octet preamble, 14-octet IEEE 802.3 header, 4-octet IEEE 802.1Q priority/VID Tag, 4-octet CRC, and 12-octet IFG) are also added. Accordingly, the total frame size is $80 + 42 = 122$, and one such frame is transmitted after every class measurement interval.

This represents a total bandwidth of about 7.8 Mbits per second ($122 \text{ octets} \times 8 \text{ bits per octet} \times 8000 \text{ frames per second}$) for this class. If the E-MAC is assumed to run at 1 Gbps (portTransmitRate), this is equivalent to the allocation of about 0.78% of the total bandwidth to each class A queue. If other traffic classes are to share the total transmission bandwidth, checking that this 0.78% allocation does not lead to the total allocation of bandwidth being greater than 100% of portTransmitRate is required.

To obtain the CIV and CDV parameters for a given class, the IFG must not be taken into account in calculation of the frame size must not include the IFG. For this case, therefore, we obtain an 80-bit payload + 30-bit overhead = 110-octet measurement interval for the class \rightarrow the total bandwidth for the class = 7.04 Mbps = 0.704% of portTransmitRate.

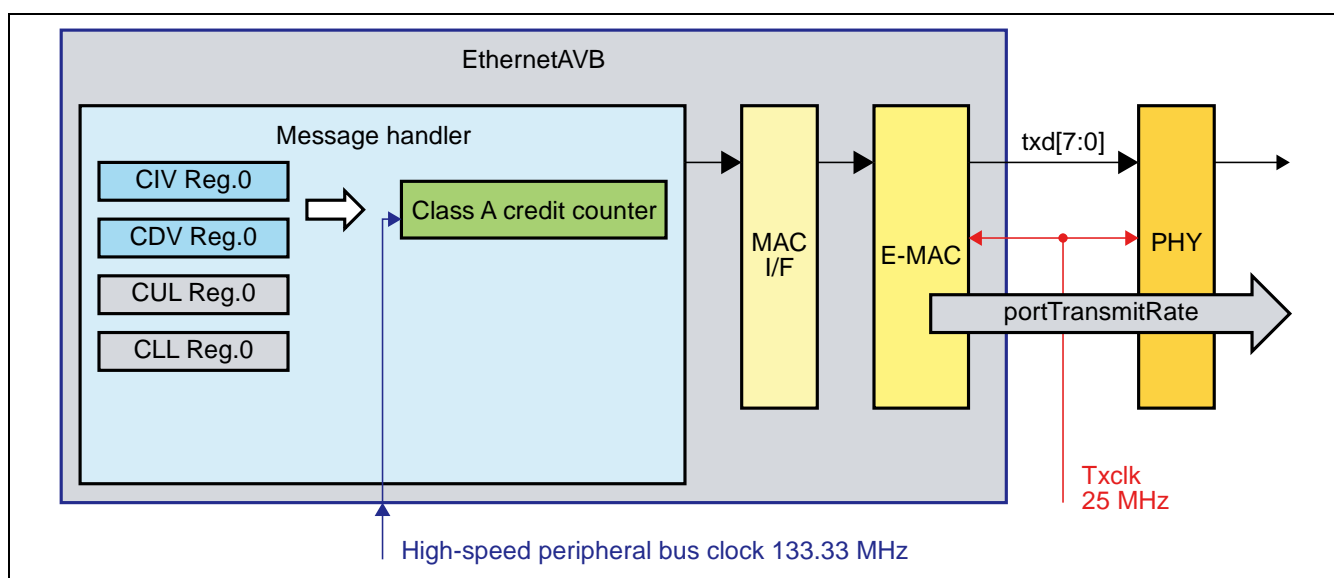


Figure 46.44 Example of CBS Settings

Given that:

- the E-MAC runs at 1000 Mbps, so portTransmitRate = 1000 Mbps and
- high-speed peripheral bus clock (operating clock for the credit counter) frequency = 133.33 MHz, securing a bandwidth of 7.04 Mbits/sec for class A requires configuring the CBS parameters as follows.

$$\text{bandwidthFraction} = 0.704\%$$

$$\text{idleSlope} = (\text{portTransmitRate} / \text{HP}\phi) \times \text{bandwidthFraction} \cong 0.0528 \text{ bits per high-speed peripheral bus clock cycle}$$

$$\text{sendSlope} = \text{idleSlope} - (\text{portTransmitRate} / \text{HP}\phi) \cong -7.44739 \text{ bits per high-speed peripheral bus clock cycle}$$

When Mfactor = 100000, the parameters are as follows.

- CIV = idleSlope \times Mfactor = 5280 bits per high-speed peripheral bus clock cycle
- CDV = sendSlope \times Mfactor = 744739 bits per high-speed peripheral bus clock cycle

These are the final values for setting in the CIVR1 and CDVR1 registers.

46.3.7 Time Synchronization

(1) gPTP Timer

An 84-bit timer is provided to support the gPTP function. Figure 46.45 shows the definitions of bits for the timer and in related registers.

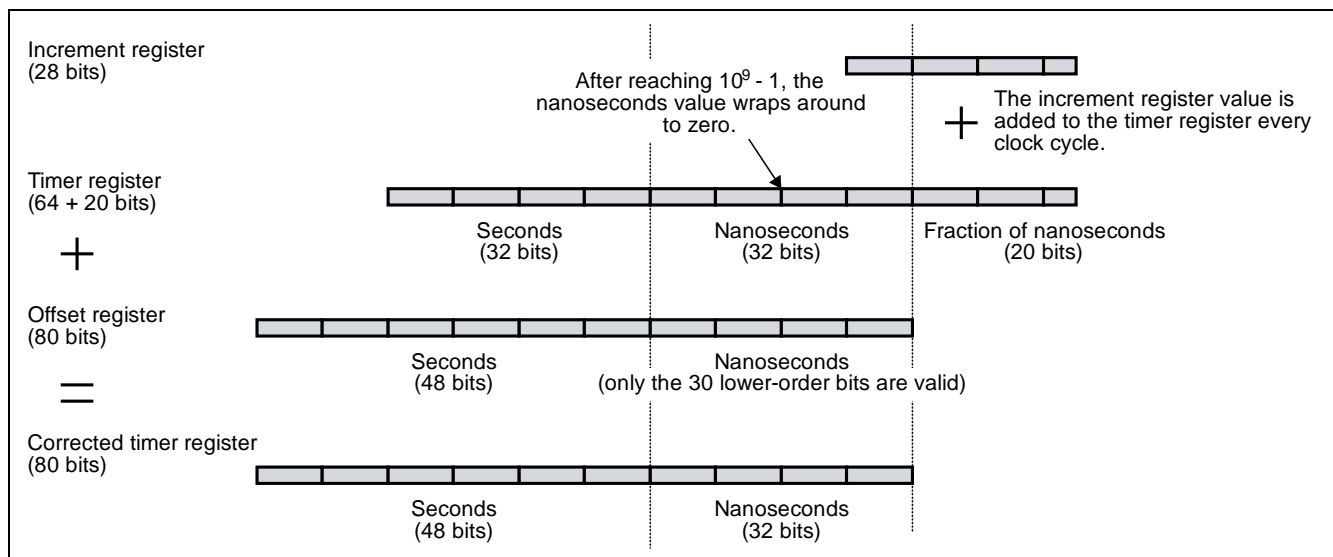


Figure 46.45 Definitions of gPTP Timer Bits and Related Bits

The higher-order 32 bits indicate seconds. For the next 32 bits, counting by one corresponds to the passage of 1 ns. The lower-order 20 bits are a fractional value (less than 1 ns). Software can only read the 32 higher-order bits, indicating seconds, and the subsequent 32-bits, indicating nanoseconds. The 20 lower-order 20 bits, representing less than 1 ns, are not readable. They are only used within the AVB-DMAC to maintain accuracy in time measurement.

The timer can be reset by setting the timer control request bits in the gPTP configuration control register (GCCR.TCR[1:0]) to B'01. These bits are set to B'00 on completion of normal resetting of the timer.

After the timer starts, the value in the gPTP timer increment register (GTI.TIV) is added to the value of the gPTP timer every clock cycle.

Before setting a value in the gPTP timer increment register (GTI.TIV), set the timer increment value setting request bit in the gPTP configuration control register (GCCR.LTI). If this bit is not set to 1, new values that are written will not be reflected in the register. This bit returns to 0 after the setting is completed.

An offset to the gPTP timer is also available. If this is required, set the value in the gPTP timer offset register (GTO.TOV). After setting a value in this register, set the timer offset value setting request bit in the gPTP configuration control register (GCCR.LTO). If this bit is not set to 1, new values that are written will not be reflected in the register. This bit returns to 0 after the setting is completed. When adding an offset, take care that it does not exceed 80 bits.

The value of the gPTP timer can be read from the gPTP timer capture register (GCTi.CTV). Set the timer capture source select bits in the gPTP configuration control register (TCCR.TCSS) to select the timer value for capture as the value of the gPTP timer, the corrected value of the gPTP timer (value with the offset added), or the AVTP presentation time. Setting the timer control request bits in the gPTP configuration control register (GCCR.TCR[1:0]) to B'11 initiates the capture. Once normal capture of the timer is complete, the value of the timer control request bits in the gPTP configuration control register (GCCR.TCR[1:0]) returns to B'00.

The timer for gPTP operates as a free-running timer but can be synchronized with the Grandmaster clock.

(2) Free-Running Operation

The IEEE 802.1 AS standard for timing and synchronization does not prescribe the physical adjustment of local clocks to the Grandmaster clock. To avoid negative effects from the correction procedure, we recommend the use of a free-running timer.

As a free-running timer, the timer counts the local time in seconds or nanoseconds. The gPTP timer increment register (GTI.TIV) is set to 1 ns (the setting value = H'0010_0000) and the gPTP timer offset register (GTOi.TOV) is set to 0. The ratio information captured at the time of the gPTP delay measurement and synchronization procedures is used to correct the frequency ratio to that of the Grandmaster clock. The Grandmaster clock can be calculated from the local clock by using the information collected during the gPTP measurement and synchronization procedures.

(3) Synchronization with the Grandmaster Clock

In situations requiring physical synchronization of the local clock with the Grandmaster clock, the fractional nanoseconds value (the 20 lower-order bits of the gPTP timer) is used to make the adjustment. Specifically, the increment value is finely adjusted to correct for deviations of the clock frequency from that of the Grandmaster clock.

Use the timer offset value (in the gPTP timer offset registers, GTOi.TOV) to correct for offsets from the theoretical value (at start-up, etc.). The sum of the timer value and the offset register is the “corrected timer” value.

Note that for the nanoseconds part of the Offset register GTOi.TOV[31:0] should be below 10^9 .

The following equation gives a method of calculating the increment (GTI.TIV) from the frequency of the gPTP clock and its deviation from that of the Grandmaster clock. Variable d is the deviation ($d = 10^{-6}$ for 1 ppm).

$$GTI.TIV = round \left(\frac{2^{20} GHz}{f_{GPTP}} \times (1 + d) \right)$$

After adjusting for the current deviation of clock frequency, reset the gPTP timer increment register (GTI.TIV).

After calculating the new offset value, reset the gPTP timer offset register (GTOi.TOV).

(4) Continuous AVTP capture

CPU can request a selected timer value by setting bit GCCR.TCR to B'11. GCCR.TCSS determines which of the three values:

- gPTP Timer value,
- Corrected Timer value or
- AVTP Timer value

is captured in GCTi.CTV. GCCR.TCR is set to B'00 after capturing the related timer value. Successful capturing operation can be confirmed by checking GCCR.TCR is B'00.

By setting GCCR.TCR to B'10 continuous AVTP timer value capture is activated. In this mode CPU has always access to current AVTP value by reading GCTi.CTV[31:0].

Flow to enter continuous AVTP capture mode:

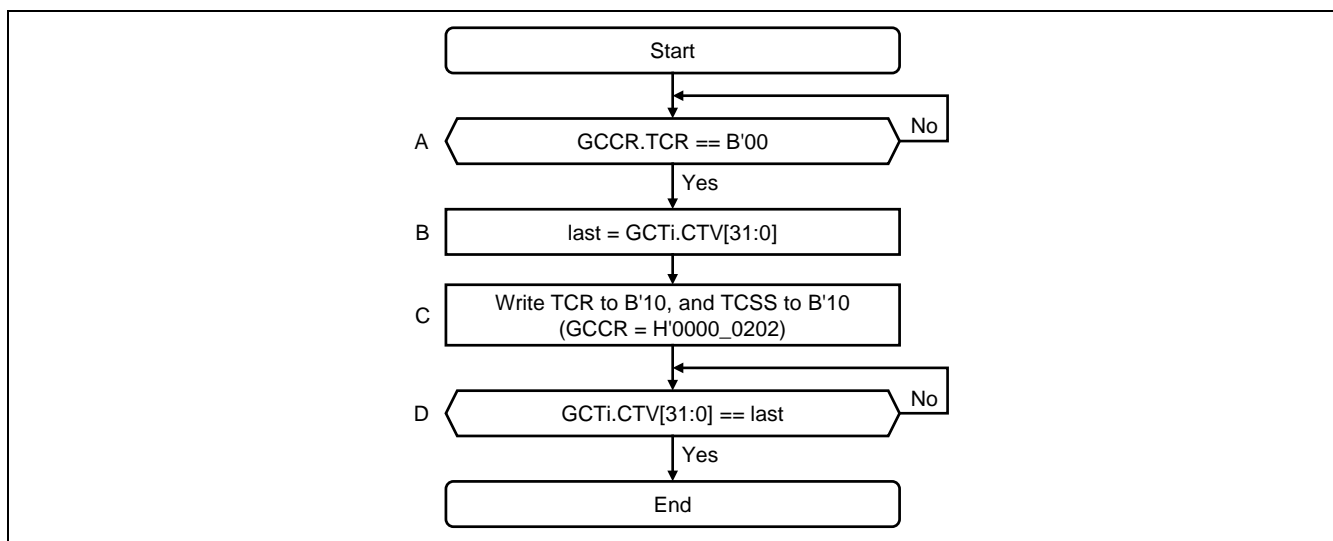


Figure 46.46 SW flow entering continuous AVTP capture mode

Flow to leave continuous AVTP capture mode:

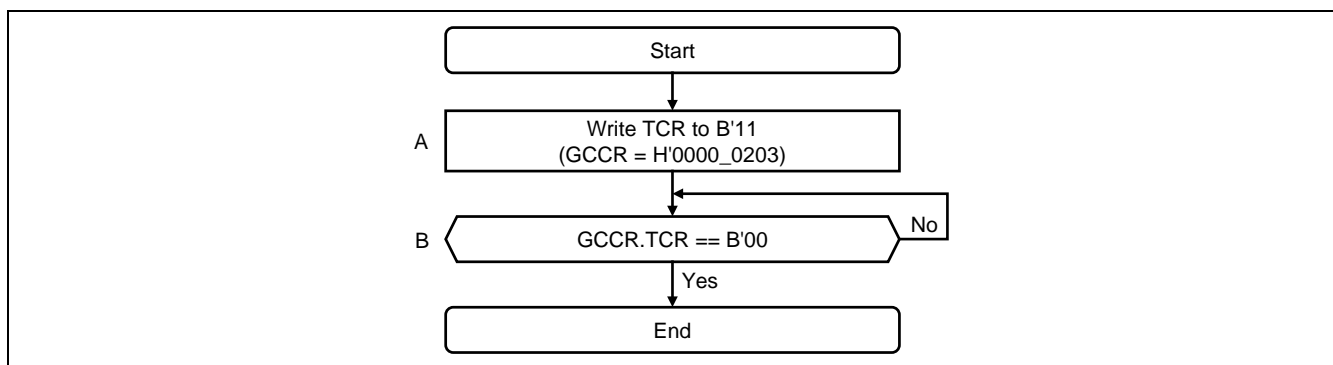


Figure 46.47 SW flow leaving continuous AVTP capture mode

(5) Support Provided by the gPTP Timer in Transmission and Reception

The timer value described above is used in the time-stamp values captured when start frame delimiters are detected in reception and generated in transmission at PHY interface.

Captured time stamp values for received frames are stored in the corresponding descriptors. Those for transmitted frames are stored with tag information in the time-stamp FIFO. The time stamp values are thus correlated with both transmitted and received frames.

Note that the use of corrected timer values can introduce an error due to the offset correction in the gPTP synchronization procedure.

Due to asynchronous interface between the SFD notification and the timer modules, errors must also be taken into account.

46.3.8 Support for IEEE 1722

For IEEE 1722, the following two functions are supported.

- Output and capture of values in the IEEE 1722 AVTP (Audio/Video Transport Protocol) presentation time format
- Comparison of IEEE 1722 AVTP presentation time stamps

The 32-bit AVTP time stamp field of IEEE 1722 frames holds the AVTP presentation time when the AVTP time-stamp enable bit in the frame is 1. The AVTP time stamp field is generated from the gPTP timer and is given as seconds (gPTP_seconds) and nanoseconds (gPTP_nanoseconds) according to the following equation.

$$\text{AVTP time stamp} = (\text{gPTP_seconds} \times 10^9 + \text{gPTP_nanoseconds}) \text{ modulo } 2^{32}$$

The AVTP presentation time can be read from the gPTP timer capture register (GCTi.CTV). Set the timer capture source select bits in the gPTP configuration control register (GCCR.TCSS) to select the timer value for capture as the AVTP presentation time. Setting the timer control request bits in the gPTP configuration control register (GCCR.TCR[1:0]) to B'11 initiates the capture. The value is obtained by adding the maximum transit time defined in the gPTP maximum transit time register (GMTT.MTTV) to the corrected timer value. The AVTP presentation time wraps around approximately every four seconds.

Note: The AVTP presentation time captured in GCTi.CTV is only valid when the corrected timer value is in synchronization with the Grandmaster clock. That is, the timer increment and timer offset values for the corrected timer value must be adjusted during the synchronization procedure so that the corrected gPTP clock is physically adjusted to match the time kept by the Grandmaster clock.

(1) AVTP capture(HW)

EthernetAVB-IF provides 17 units to capture AVTP based on external events.

Table 46.17 to Table 46.20 lists the assignment of the EthernetAVB-IF capture unit.

Table 46.17 Capture unit assignment for RZ/G2H

| Unit | signal | from | Function |
|--------------------|-------------------------|----------|---------------------------------|
| avb_pt_capture[0] | AVB_AVTP_CAPTURE | External | External input |
| avb_pt_capture[1] | AVB_AVTP_CAPTURE (copy) | External | External input (copy) |
| avb_pt_capture[2] | avb_adg0_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[3] | avb_adg1_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[4] | avb_adg2_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[5] | avb_adg3_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[6] | avb_adg4_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[7] | avb_adg5_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[8] | avb_adg6_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[9] | avb_adg7_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[10] | avb_adg8_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[11] | avb_adg9_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[12] | avb_adg10_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[13] | avb_adg11_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[14] | VSYNC | DU0 | Vsync signal from DU module |
| avb_pt_capture[15] | VSYNC | DU2 | Vsync signal from DU module |
| avb_pt_capture[16] | VSYNC | DU3 | Vsync signal from DU module |

Table 46.18 Capture unit assignment for RZ/G2M V1.3, RZ/G2M V3.0

| Unit | signal | from | Function |
|--------------------|-------------------------|----------|---------------------------------|
| avb_pt_capture[0] | AVB_AVTP_CAPTURE | External | External input |
| avb_pt_capture[1] | AVB_AVTP_CAPTURE (copy) | External | External input (copy) |
| avb_pt_capture[2] | avb_adg0_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[3] | avb_adg1_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[4] | avb_adg2_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[5] | avb_adg3_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[6] | avb_adg4_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[7] | avb_adg5_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[8] | avb_adg6_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[9] | avb_adg7_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[10] | avb_adg8_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[11] | avb_adg9_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[12] | avb_adg10_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[13] | avb_adg11_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[14] | VSYNC | DU0 | Vsync signal from DU module |
| avb_pt_capture[15] | VSYNC | DU1 | Vsync signal from DU module |
| avb_pt_capture[16] | VSYNC | DU2 | Vsync signal from DU module |

Table 46.19 Capture unit assignment for RZ/G2E

| Unit | signal | from | Function |
|--------------------|-------------------------|----------|---------------------------------|
| avb_pt_capture[0] | AVB_AVTP_CAPTURE | External | External input |
| avb_pt_capture[1] | AVB_AVTP_CAPTURE (copy) | External | External input (copy) |
| avb_pt_capture[2] | avb_adg0_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[3] | avb_adg1_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[4] | avb_adg2_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[5] | avb_adg3_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[6] | avb_adg4_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[7] | avb_adg5_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[8] | avb_adg6_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[9] | avb_adg7_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[10] | avb_adg8_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[11] | avb_adg9_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[12] | avb_adg10_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[13] | avb_adg11_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[14] | VSYNC | DU0 | Vsync signal from DU module |
| avb_pt_capture[15] | VSYNC | DU1 | Vsync signal from DU module |
| avb_pt_capture[16] | — | — | — |

Table 46.20 Capture unit assignment for RZ/G2N

| Unit | signal | from | Function |
|--------------------|-------------------------|----------|---------------------------------|
| avb_pt_capture[0] | AVB_AVTP_CAPTURE | External | External input |
| avb_pt_capture[1] | AVB_AVTP_CAPTURE (copy) | External | External input (copy) |
| avb_pt_capture[2] | avb_adg0_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[3] | avb_adg1_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[4] | avb_adg2_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[5] | avb_adg3_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[6] | avb_adg4_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[7] | avb_adg5_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[8] | avb_adg6_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[9] | avb_adg7_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[10] | avb_adg8_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[11] | avb_adg9_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[12] | avb_adg10_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[13] | avb_adg11_sync | ADG | Timing signal from AUDIO module |
| avb_pt_capture[14] | VSYNC | DU0 | Vsync signal from DU module |
| avb_pt_capture[15] | VSYNC | DU1 | Vsync signal from DU module |
| avb_pt_capture[16] | VSYNC | DU3 | Vsync signal from DU module |

To allow proper detection of the edges of avb_pt_capture[i] signal, a pulse equal to or longer than 3 peripheral bus clock cycles should be provided on this input to trigger AVTP capture. Also SW should limit frequency of external capturing signal; period should be at least (6 peripheral bus clock periods + 3 gPTP periods).

There are two variants of capture units in EthernetAVB-IF:

- First one captures AVTP presentation time (including max transit time) and have overflow flagging.
- Second one captures AVTP timer value (without max transit timing) and have prescaler.

The AVTP capture units provides always the first captured value until SW has acknowledged reading of this value.

(a) AVTP presentation time capture unit

EthernetAVB-IF provides one capture unit of this functionality.

When a rising edge is applied to avb_pt_capture[0], AVTP presentation time is captured to GCPT.CPTV and the interrupt flag GIS.PTCF is set to 1.

The captured value includes the configured max transit time value.

Capture of a new AVTP presentation time is suppressed until CPU acknowledges processing of the captured value by setting GIS.PTCF to 0. GIS.PTOF informs about one or more lost capture events.

(b) AVTP timer capture unit

EthernetAVB-IF provides sixteen capture unit ($i = 0$ to 15) of this functionality.

Each N-times when a rising edge is applied to avb_pt_capture[i+1], AVTP timer value is captured to GCATi.CATV and the interrupt flag GIS.ATCFi is set to 1. The prescaler factor N is configured by GACP register for each unit individually.

The captured value does not include the configured max transit time value.

Capture of a new AVTP timer value is suppressed until CPU acknowledges processing of the captured value by setting GIS.ATCFi to 0.

(c) Prescaler configuration

CPU should use this flow to configure prescaler value of capture unit i:

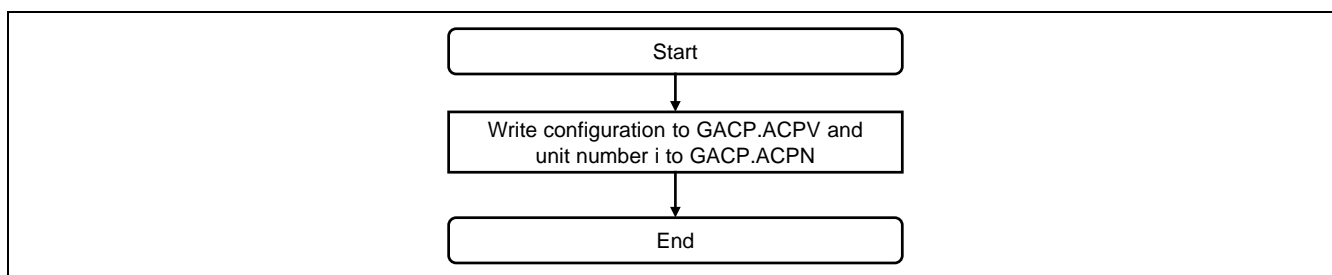


Figure 46.48 SW flow to program prescaler value of capture unit

This figure illustrates prescaler function of AVTP timer capture unit. The prescaler is free running, there is no visible relation which capture pulse triggers capturing of AVTP timer. When prescaler value is reconfigured, there may be one capturing based on old prescaler configuration.

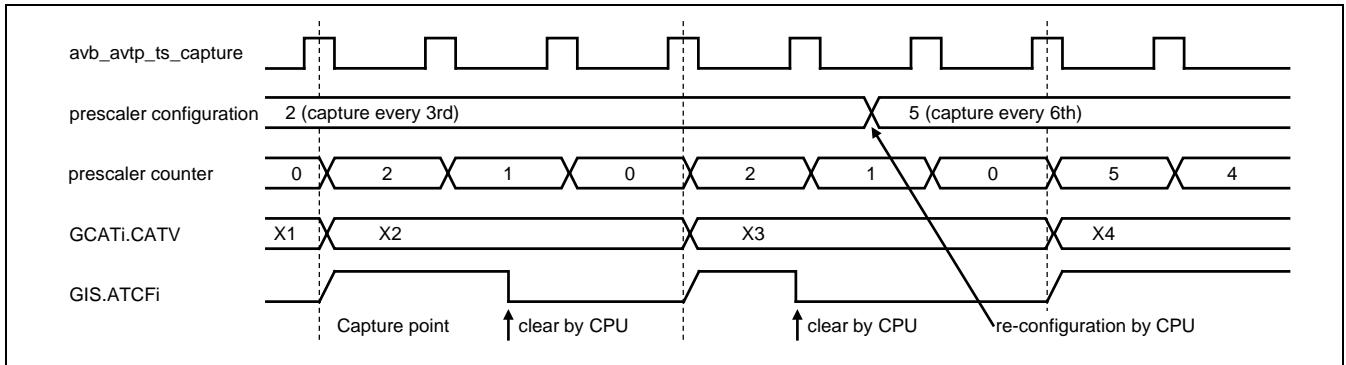


Figure 46.49 Example of capture timing when prescaler value is reconfigured

It is recommended not to change prescaler value again before there is at least one match related to previous prescaler value flagged by GIS.ATCFi.

(2) AVTP compare

EthernetAVB-IF provides 8 independent AVTP compare units. Each unit is able to handle single shot comparison as well as periodic comparison. GSR.PCMi informs about the current mode of unit i. To 4 units (0 to 3) the single shot compare value can be provided via AVTP-FIFO to relax timing requirements to application SW.

When the AVTP Presentation Time value has reached or exceeded the configured comparison value the interrupt flag GIS.PTMFi is set to 1.

And Comparison unit 0, a pulse of one or four High speed Peripheral clock cycle is applied to AVB_AVTP_MATCH.

The configured max transit time value is not taken in count when comparing.

SW should ensure that configured comparison value is well in future to take into account SW inaccuracy.

SW should ensure that comparison value is not too far in future (more than 2 seconds) with respect to current AVTP timer value to guarantee comparison functionality.

When writing to GCCR, SW can use a 32 bit access if GCCR.TCSS is written to B'11 and all other bits not required for comparison value programming are written to 0.

(a) Single shot compare

An AVTP compare unit used in single shot mode generates exactly one match after CPU programs the comparison value.

CPU should use this flow to program absolute comparison values in single shot mode:

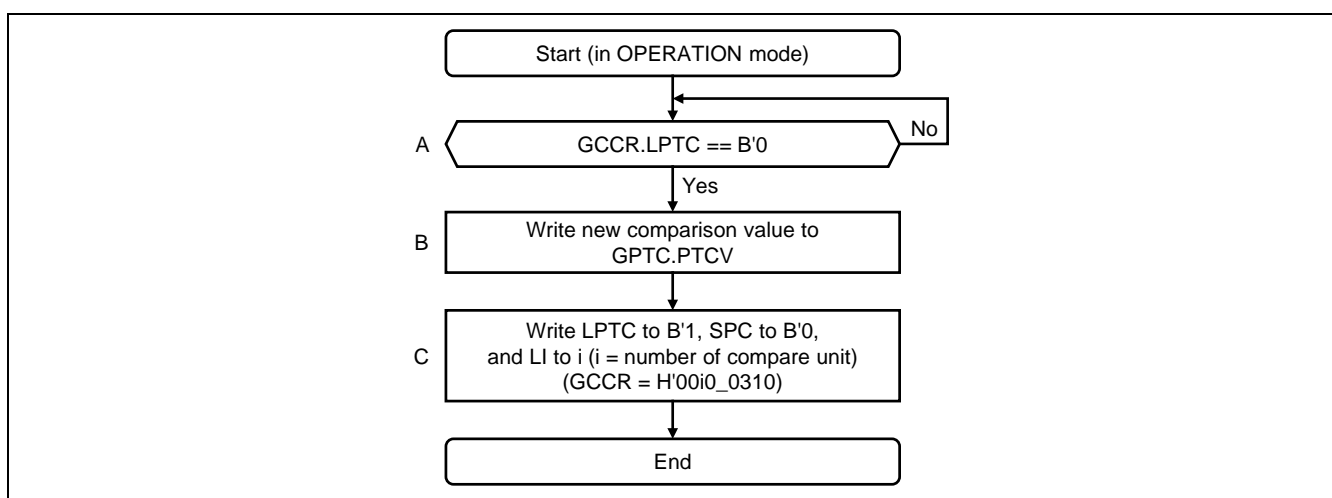


Figure 46.50 SW flow to program a single shot compare value

(b) Periodic compare

An AVTP compare unit used in periodic compare mode generates matches of configured period until CPU disables periodic mode. Optional it is possible to define the initial phase before repetition starts.

CPU should use this flow to start periodic compare mode or to change period value:

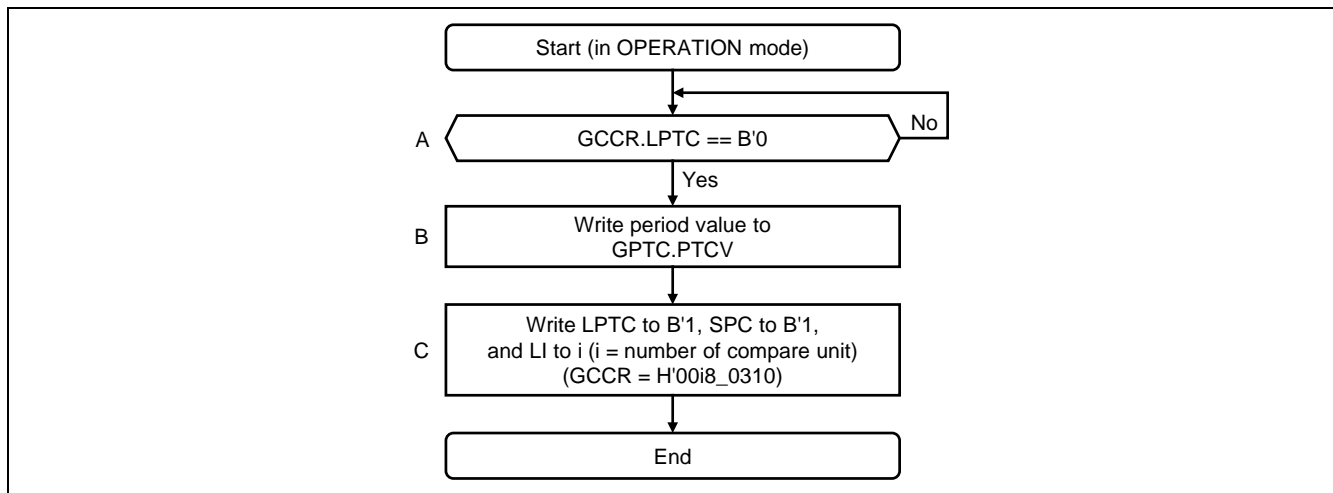


Figure 46.51 SW flow to program period value (without phase relation)

Some application may require phase relation between periodical match events generated by different units. In this case the next flow can be used to enter periodic compare mode with phase relation. A compare unit will only start a period when there is no pending comparison. If the absolute starting point of all units (box B in flow) is well in future the period of all units start at the same time (e.g. 48 kHz on unit 1 and 96 kHz on unit 2).

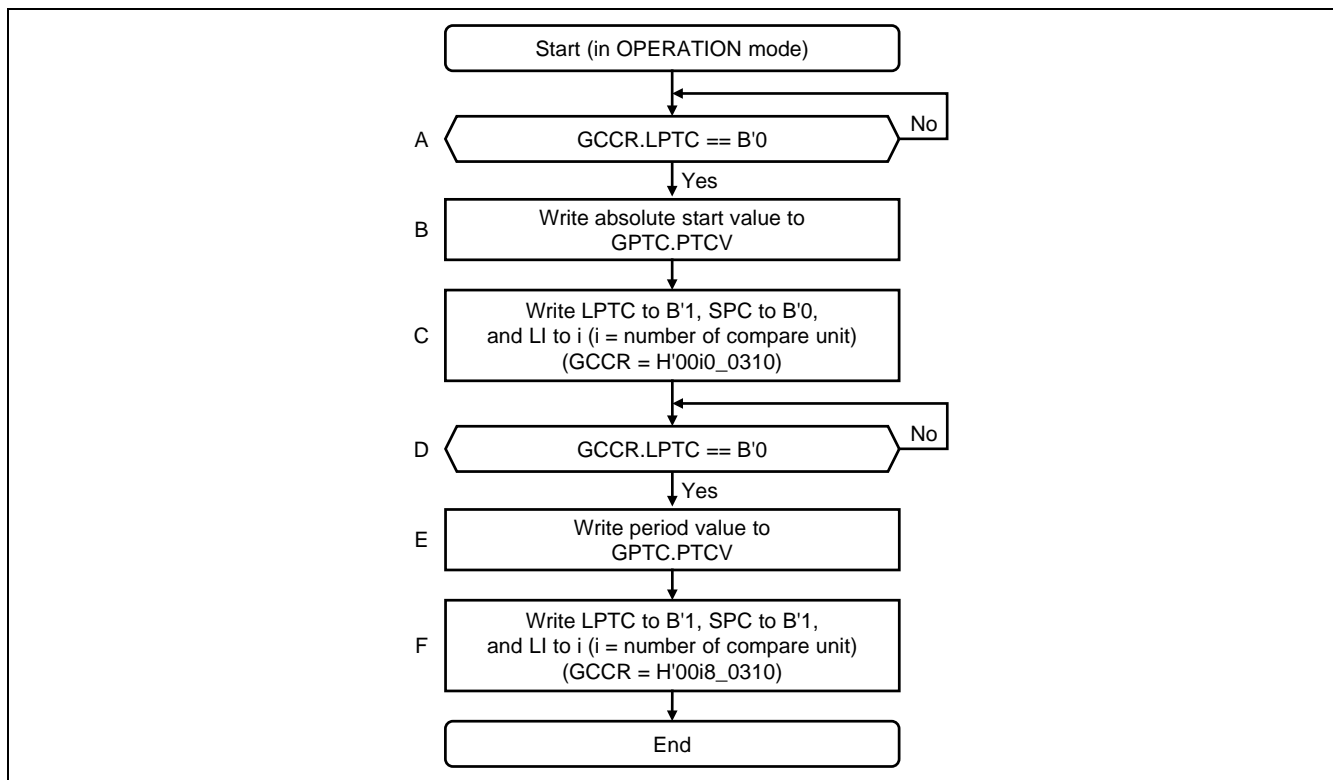


Figure 46.52 SW flow to program period value (with phase relation)

The periodic mode is left by entering single-shot mode. Do not write new comparison value in GPTC.PTCV for changing mode as shown in flow below, this value will not be used as new comparison value.

CPU should use this flow to leave periodic compare mode:

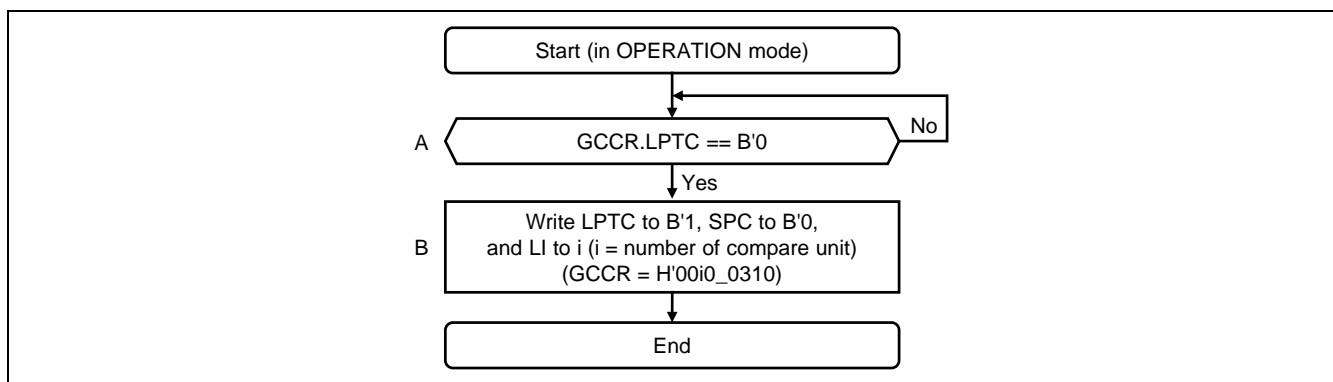


Figure 46.53 SW flow to end period mode

The last periodical comparison happens after Point B in SW flow. EthernetAVB-IF leaves periodic mode after last periodical comparison happens. This is flagged by GSR.PCMi. Comparison unit can only be reused after GCR.PCMi is 0.

(c) FIFO based single-shot comparison

Some comparison units provides the possibility to use timestamps from a FIFO. To use this possibility CPU writes AVTP timestamp to GPTFi instead of GPTC. By writing to GPTFi the value is added to FIFO, no load is required but CPU needs to check if AVTP-FIFO is able to accept new values.

FIFO usage may reduce CPU load because CPU can program a bunch of timestamps at same time.

When there is no comparison ongoing in AVTP compare unit, the next value from AVTP-FIFO is loaded and comparison starts. If the FIFO is empty no value is loaded and no further match event is generated.

When moving from single-shot comparison to FIFO based single-shot comparison, the 1st FIFO value is immediately loaded independent of an ongoing comparison.

By reading GSR.AFFLi CPU can observe number of pending timestamps.

CPU should use this flow to add timestamps to the AVTP-FIFO of compare unit. If FIFO depth is insufficient, FIFO size should be extended in SW (box E).

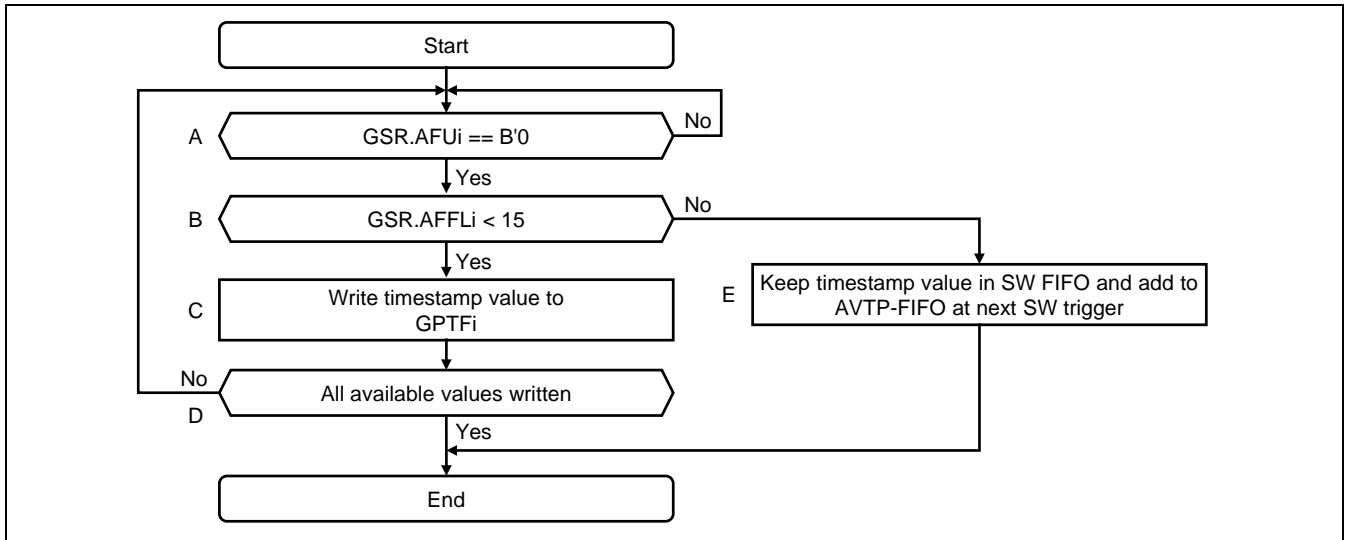


Figure 46.54 SW flow to add timestamp to AVTP-FIFO

It is recommended to combine box A and B into one access to limit accesses to GSR register.

(3) Pulse per second

EthernetAVB-IF provides by AVB_AVTP_PPS a pulse signal, which is generated at carryover of the nanosecond part of timer.

By GCCR.PGM it is selectable if the carryover of the gPTP timer (B'0) or the corrected gPTP timer (B'1) is used.

The pulse has a length of 32 gPTP clock cycles.

The CPU should not configure an offset correction value of 0 when using the corrected gPTP timer as source (GCCR.PGM is 1) for pulse per second function.

Note: A pulse may be lost or added when using the corrected gPTP timer as source and changing the offset value.

(a) Changing pulse per second source

When gPTP timer functionality is used in CONFIG mode (CCC.GAC is 1), following flow should be used to change pulse per second source by GCCR.PGM. When CCC.GAC is 0, CPU can change GCCR.PGM at any time.

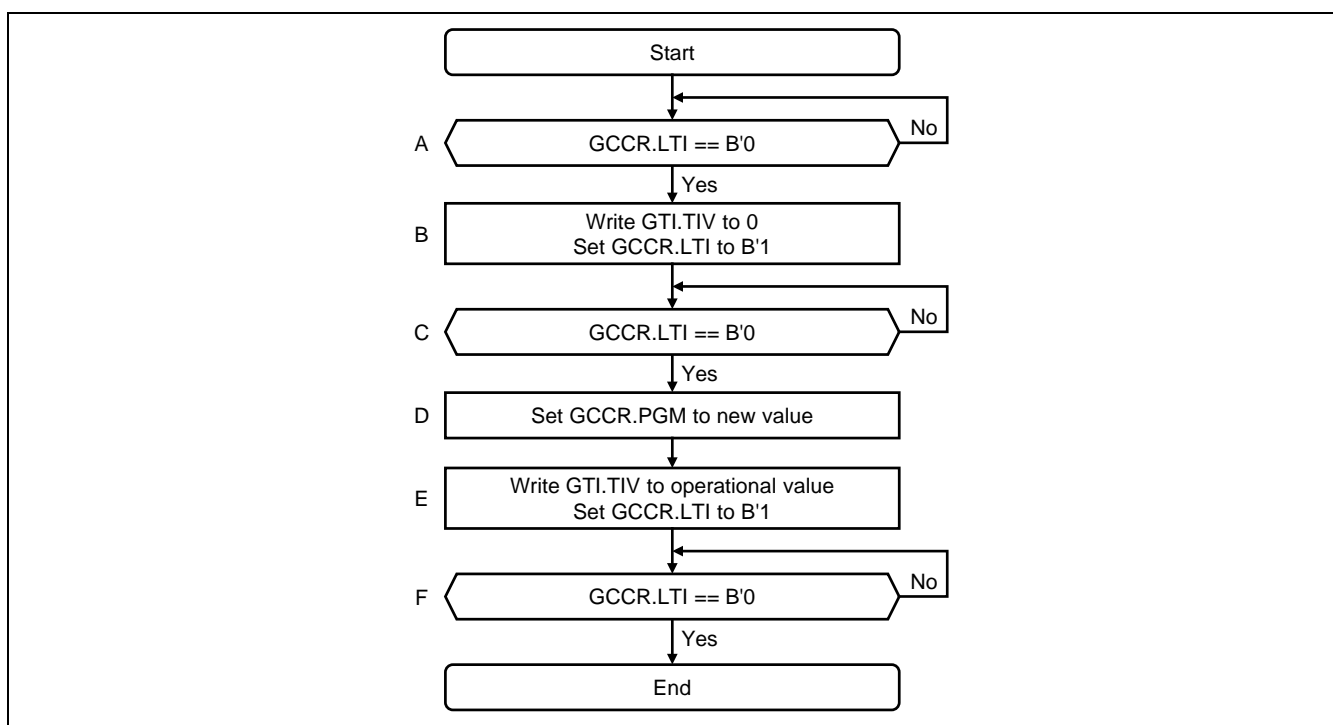


Figure 46.55 SW flow to change GCCR.PGM when CCC.GAC is 1

The increment value of 0 in box B intentionally violates SW restriction of GTI.TIV.

To limit side effects to other gPTP functions 8 bit access to GCCR should be used in box B and E.

EthernetAVB-IF does not guarantee that value read from GTI.TIV is same value as increment value used by timer. Due to this it is recommended, to use in box E. the increment value intended by SW.

(4) gPTP timer functionality in CONFIG mode

When CCC.GAC is 1, it is possible to use some gPTP functions in CONFIG mode. Because the frame handling mechanisms are not available in CONFIG mode, usage of gPTP support is restricted. Only basic timer functions of gPTP support can be used:

- gPTP timer is running based on selected clock source (CCC.CSEL),
- Offset and increment value can be configured and adjusted
- SW capture function (timer, corrected timer, AVTP)
- HW capture function (AVTP)
- SW compare function

Note: There is SW flow required to change pulse per second source when gPTP timer is used in CONFIG mode. See section 46.3.8(3)(a) for details.

When AVTP FIFO functionality is used in CONFIG mode, there is no error flagging by ESR.ET = B'1011 available. CPU can use error flagging by external RAM protection directly, there are no other read accesses to RAM by EthernetAVB-IF.

46.3.9 Flow Control

The EthernetAVB-IF does not support flow control in the specification. Flow control is only effective when transmitting and receiving only normal (non-AVB) packets.

The E-MAC supports flow control for full-duplex operation in compliance with the IEEE 802.3 standards. This flow control is applicable to both reception and transmission. In regard to the transmission of PAUSE frames, flow control operates in the following ways.

(1) PAUSE Frame Transmission

PAUSE frames can also be transmitted in response to software operations. Writing a timer value to the manual PAUSE frame register (MPR) starts the transmission of a PAUSE frame. This only causes the transmission of one PAUSE frame.

(2) PAUSE Frame Reception

After reception of a PAUSE frame, transmission of the next frame does not proceed until the time indicated by the Timer value elapses. However, transmission of a frame currently being transmitted continues. PAUSE frames are only received while the RXF bit in the E-MAC mode register (ECMR) is set to 1. The number of received PAUSE frames is counted.

(3) PAUSE Frames with the Timer Value 0

The setting of the 0-time PAUSE frame enable bit (ECMR.ZPF) enables or disables the reception of PAUSE frames with the TIME parameter value 0.

- When control of PAUSE frames with the TIME parameter value 0 is disabled

PAUSE frames with the TIME parameter value 0 are not transmitted. Received PAUSE frames with the TIME parameter value 0 are discarded.

46.3.10 Magic Packet Detection

The E-MAC has a Magic Packet detection function. This function provides a facility for host devices and other sources to start other peripheral devices connected to a LAN. A peripheral device that handles Magic Packets starts itself in response to receiving a Magic Packet.

When a Magic Packet is detected, data from broadcast packets that were previously being received are stored in the FIFO and the E-MAC is notified of the receiving status. To return to normal operation from the associated interrupt processing, the E-MAC and AVB-DMAC must be initialized by using the operating mode configuration bit in the AVB-DMAC mode register (CCC.OPC) to set the operating mode to reset mode.

Magic Packets are received regardless of the destination address. As a result, the AVB_MAGIC pin is only enabled in the case of a match with the destination address specified in the Magic Packet.

The procedure for using the Magic Packet detection function with this LSI chip is as follows.

1. Use the various interrupt enabling and masking registers to disable the output of interrupts from interrupt sources.
2. Set the Magic Packet detection enable bit in the E-MAC mode register (ECMR.MPDE).
3. Set the Magic Packet detection interrupt enable bit in the E-MAC interrupt enable register (ECSIPR.MPDIP) to enable the interrupt.
4. Place the CPU in sleep mode as required.
5. An interrupt is conveyed to the CPU on detection of a Magic Packet.
6. The AVB_MAGIC pin notifies connected devices of Magic Packet detection.

Note: The Magic Packet detection interrupt status can be read in the E-MAC status register (ECSR.MPD). The bit can be cleared by setting the E-MAC status register Magic Packet™ detection bit (ECSR.MPD) to 1.

But EthernetAVB-IF can't detect 2nd or later Magic Packet™ and the AVB_MAGIC pin can be negated though the bit is cleared.

The AVB_MAGIC pin can be negated and EthernetAVB-IF can detect MAGIC Packet™ again according to below way.

- | | |
|-----------------------------------|---|
| 1) Hardware reset | set reset pin in the RZ/G series to reset signal. |
| 2) Software reset | set mstp/srst register for EthernetAVB-IF to software reset mode. |
| 3) Software reset register in AVB | set AVB DMAC mode register Operating Mode Configuration bits (CCC.OPC) to reset mode. |

46.3.11 Interrupts

The EthernetAVB-IF is capable of generating 25 interrupt (RZ/G2M V1.3, RZ/G2M V3.0).

The EthernetAVB-IF is capable of generating 27 interrupt (without RZ/G2M V1.3, RZ/G2M V3.0).

For details, refer to the “Interrupt Controller” section in the manual.

The interrupts of EthernetAVB-IF are indicated by individual flags DIS, EIS, RISi, TIS, and GIS. Each interrupt is controllable by the corresponding interrupt enable bit. The status flags are independent from this enable bits.

Additionally grouped interrupt status available in ISS and EIS.QFS. The summary status allows CPU to locate the interrupt source with a minimum number of read accesses to EthernetAVB-IF.

EthernetAVB-IF interrupts are combined to 4 groups of interrupt lines for request an interrupt service by CPU. Interrupt requests in group 0 and 1 are only generated in OPERATION mode. Interrupt requests in group 2 are only generated in OPERATION mode and in CONFIG mode if gPTP support is enabled (CCC.GAC). Interrupt request on line 3 depends on E-MAC configuration and can be generated in CONFIG, OPERATION and STANDBY mode.

Interrupt lines are level output and each line is asserted when at least one of the interrupt status registers bits belonging to the group associated with that line is set to 1, and the corresponding enable bit in the related interrupt control register is also set to 1.

For ISS bits the combined flag/enable is used. The green dots in detailed interrupt figures in detailed interrupt line description illustrates the point where status information for ISS is taken.

Table 46.21 Interrupt assignment

| Interrupt Name | Cause of interrupt | Second Generation RZ/G Series Products | | | |
|-------------------|--------------------|---|----------------------------|--------|--------|
| | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Ethernet AVB.ch0 | Line0_Rx[0] | √ | √ | √ | √ |
| Ethernet AVB.ch1 | Line0_Rx[1] | √ | √ | √ | √ |
| Ethernet AVB.ch2 | Line0_Rx[2] | √ | √ | √ | √ |
| Ethernet AVB.ch3 | Line0_Rx[3] | √ | √ | √ | √ |
| Ethernet AVB.ch4 | Line0_Rx[4] | √ | √ | √ | √ |
| Ethernet AVB.ch5 | Line0_Rx[5] | √ | √ | √ | √ |
| Ethernet AVB.ch6 | Line0_Rx[6] | √ | √ | √ | √ |
| Ethernet AVB.ch7 | Line0_Rx[7] | √ | √ | √ | √ |
| Ethernet AVB.ch8 | Line0_Rx[8] | √ | √ | √ | √ |
| Ethernet AVB.ch9 | Line0_Rx[9] | √ | √ | √ | √ |
| Ethernet AVB.ch10 | Line0_Rx[10] | √ | √ | √ | √ |
| Ethernet AVB.ch11 | Line0_Rx[11] | √ | √ | √ | √ |
| Ethernet AVB.ch12 | Line0_Rx[12] | √ | √ | √ | √ |
| Ethernet AVB.ch13 | Line0_Rx[13] | √ | √ | √ | √ |
| Ethernet AVB.ch14 | Line0_Rx[14] | √ | √ | √ | √ |
| Ethernet AVB.ch15 | Line0_Rx[15] | √ | √ | √ | √ |
| Ethernet AVB.ch16 | Line0_Rx[16] | √ | √ | √ | √ |
| Ethernet AVB.ch17 | Line0_Rx[17] | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Interrupt Name | Cause of interrupt | Second Generation RZ/G Series Products | | | |
|-------------------|-------------------------------|--|----------------------------|--------|--------|
| | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Ethernet AVB.ch18 | Line0_Tx[0] | √ | √ | √ | √ |
| Ethernet AVB.ch19 | Line0_Tx[1] | √ | √ | √ | √ |
| Ethernet AVB.ch20 | Line0_Tx[2] | √ | √ | √ | √ |
| Ethernet AVB.ch21 | Line0_Tx[3] | √ | √ | √ | √ |
| Ethernet AVB.ch22 | Line0_DiA Line1_A Line2_A | √ | √ | √ | √ |
| Ethernet AVB.ch23 | Line0_DiB Line1_B Line2_B | √ | √ | √ | √ |
| Ethernet AVB.ch24 | Line 3 | √ | √ | √ | √ |

(1) Concurrent interrupt enable support

The interrupt enable bits can be directly modified by writing the interrupt control registers (**xxC.yyE**). If concurrent processes enable/disable interrupts located in same interrupt control register inter-process communication is required. To reduce SW overhead there are dedicated registers available controlling conflict free enable/disable of interrupts.

To enable an interrupt CPU should write 1 to the corresponding bit (**xxE.yyS**) in the enable register. Because written 0 have no effect, there is no side effect to concurrent processes.

To disable an interrupt CPU should write 1 to the corresponding bit (**xxD.yyD**) in the disable register. Because written 0 have no effect, there is no side effect to concurrent processes.

The current interrupt enable status is readable in **xxC.yyE**.

Grouping of interrupts for each interrupt line is as described in following sub-chapters.

(2) Group 0 (Data related interrupts)

This group consists on:

Frame received interrupts **RIS0.FRFr** (for each received frame stored in URAM)

Receive warning interrupts **RIS1.RWFr** (when Unread Frame Counter reaches warning level)

Descriptor processed interrupts **RIS3.RDPFr** (for reception) Frame transmitted interrupts **TIS.FTFt** (for completion of frame transmission by E-MAC)

Descriptor processed interrupts **TIS.TDPFr** (for transmission)

Descriptor processed interrupts **DIS.DPFi** (for reception and transmission)

By reading **ISS**, CPU get an overview of all data related interrupts (**ISS.FRM**, **ISS.RWM**, **ISS.DPMi**, **ISS.FTM**).

Interrupt group 0 consists on these 24 interrupt lines:

Line0_Rx[17:0]

Line0_Tx[3:0]

Line0_DiA (Share the interrupt line: Group0.Line0_DiA, Group1.Line1_A, Group2.Line2_A)

Line0_DiB (Share the interrupt line: Group0.Line0_DiB, Group1.Line1_B, Group2.Line2_B)

(a) Reception interrupts

There are 18 dedicated interrupt lines flagging frame reception. For the queue full flag it is configurable by CIE.RQFM if these notification are part of the queue specific reception interrupt lines or part of the error interrupt lines (group 1).

There is a common interrupt enable (CIE.CRIE) for all reception lines.

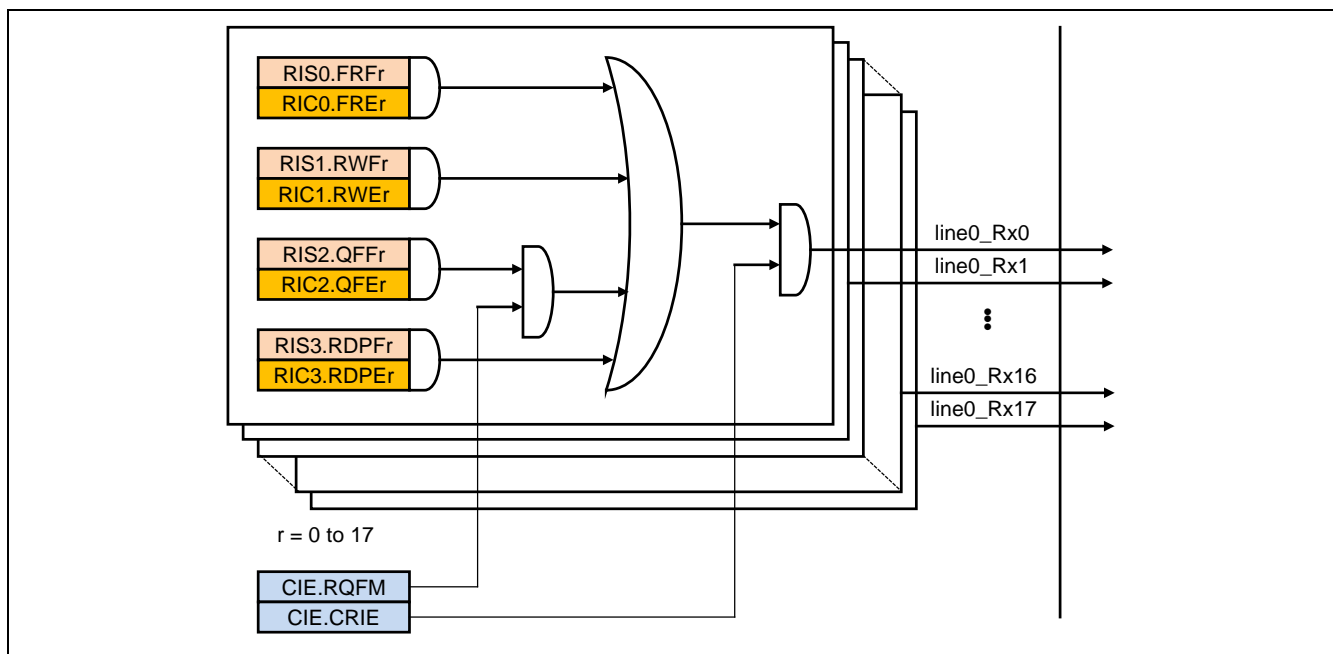


Figure 46.56 Interrupt group 0, reception interrupt lines

(b) Transmission interrupts

There are 4 dedicated interrupt lines flagging frame transmission.

There is a common interrupt enable (CIE.CTIE) for all reception lines.

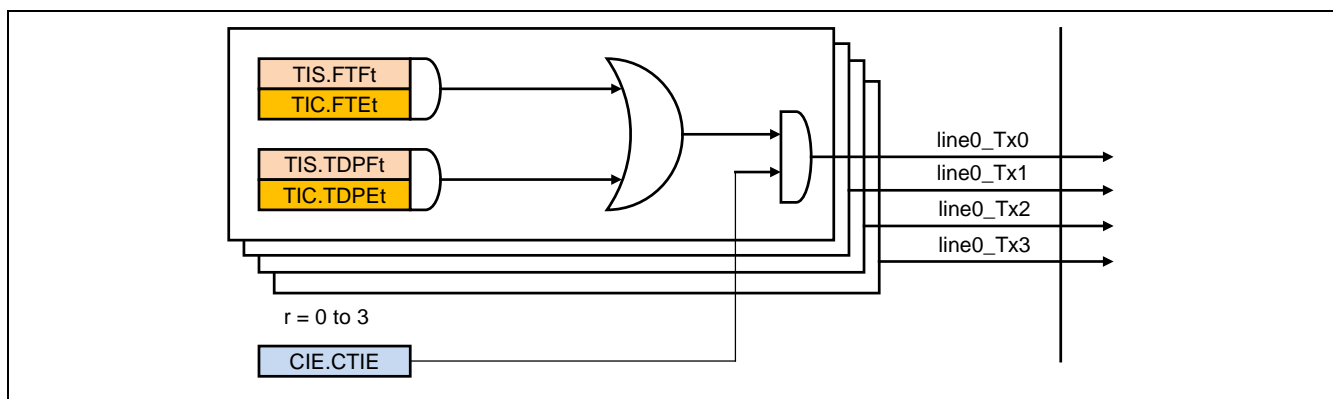


Figure 46.57 Interrupt group 0, transmission interrupt lines

(c) Descriptor and merged interrupts

There are 2 interrupt lines flagging descriptor interrupts. Line line0_DiA has also the capability to generate a merged data interrupt (controlled by CIE.CL0M).

Descriptor interrupt 1 is not available as common descriptor interrupt; this descriptor interrupt is split into queue specific descriptor interrupts.

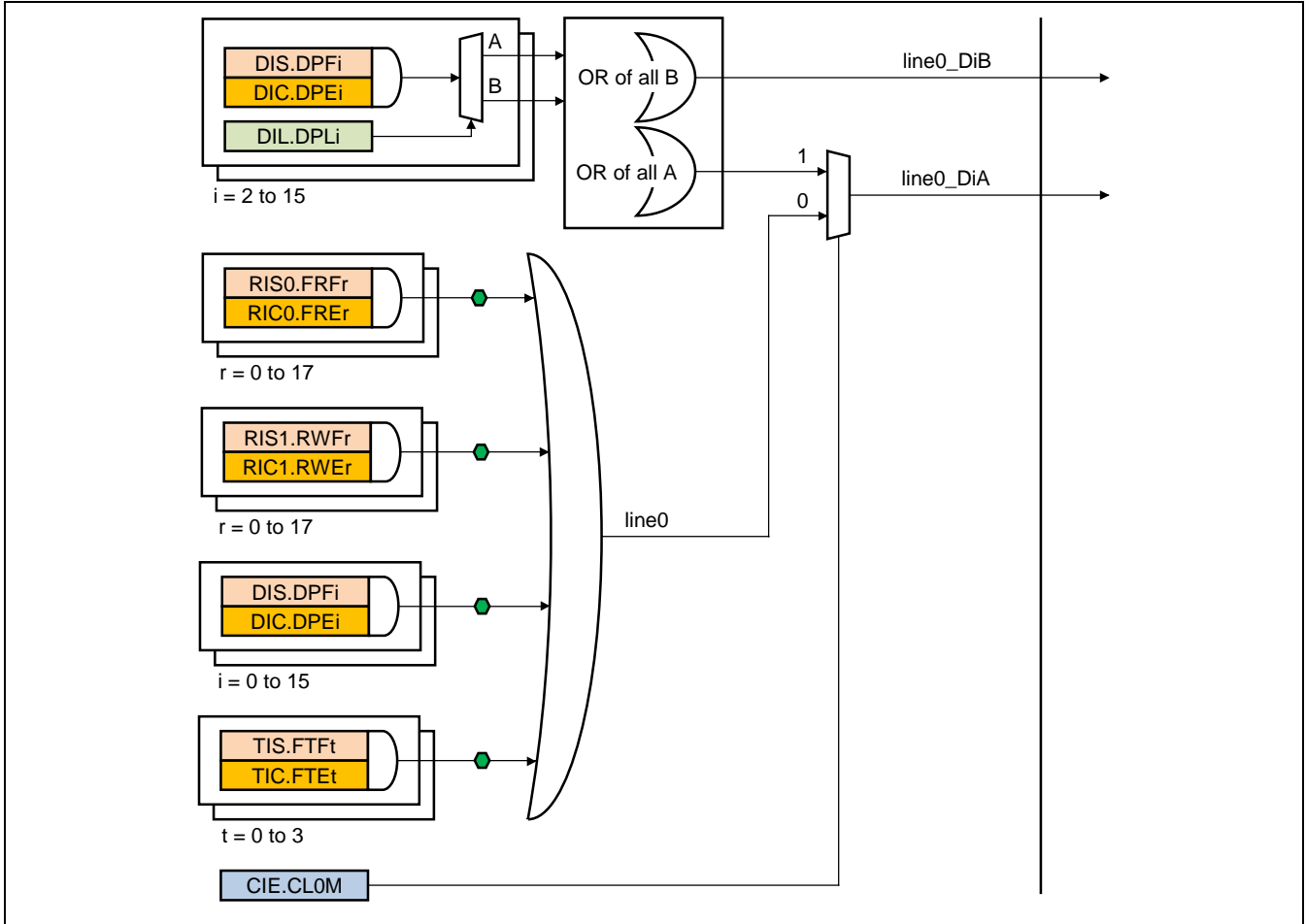


Figure 46.58 Interrupt group 0, common descriptor and merged interrupt lines

(3) Group 1 (Error related interrupts)

This group consists on:

- Tx-Buffer full interrupt EIS.TBFF (when delaying fetching of transmit frame)
- Interrupt for E-MAC Status FIFO overwrite (EIS.MFFF)
- Interrupt for Timestamp FIFO overrun (EIS.TFFF)
- Interrupts for CBS limits (EIS.CULF1, EIS.CULF0, EIS.CLLF1, EIS.CLLF0)
- Interrupt for separation filtering error (EIS.SEF)
- Interrupt for queue error (error while processing receive or transmit queue) (EIS.QEF)
- Interrupt for transmission error detected by E-MAC (EIS.MTEF, EIS.MREF)
- Rx-FIFO full interrupt RIS2.RFFF (when Rx-FIFO overflows)
- Receive queue full interrupts RIS2.QFFr (when no sufficient space is in receive queue)

By reading EIS, CPU get an overview of all error interrupts. Additionally ISS.EM informs about pending error related interrupt.

Interrupt group 1 consists on these 2 interrupt lines:

Line1_A (Share the interrupt line: Group0.Line0_DiA, Group1.Line1_A, Group2.Line2_A)

Line1_B (Share the interrupt line: Group0.Line0_DiB, Group1.Line1_B, Group2.Line2_B)

For the queue full flag it is configurable by CIE.RQFM if these notification are part of the queue specific reception interrupt lines or part of the error interrupt line line1_A.

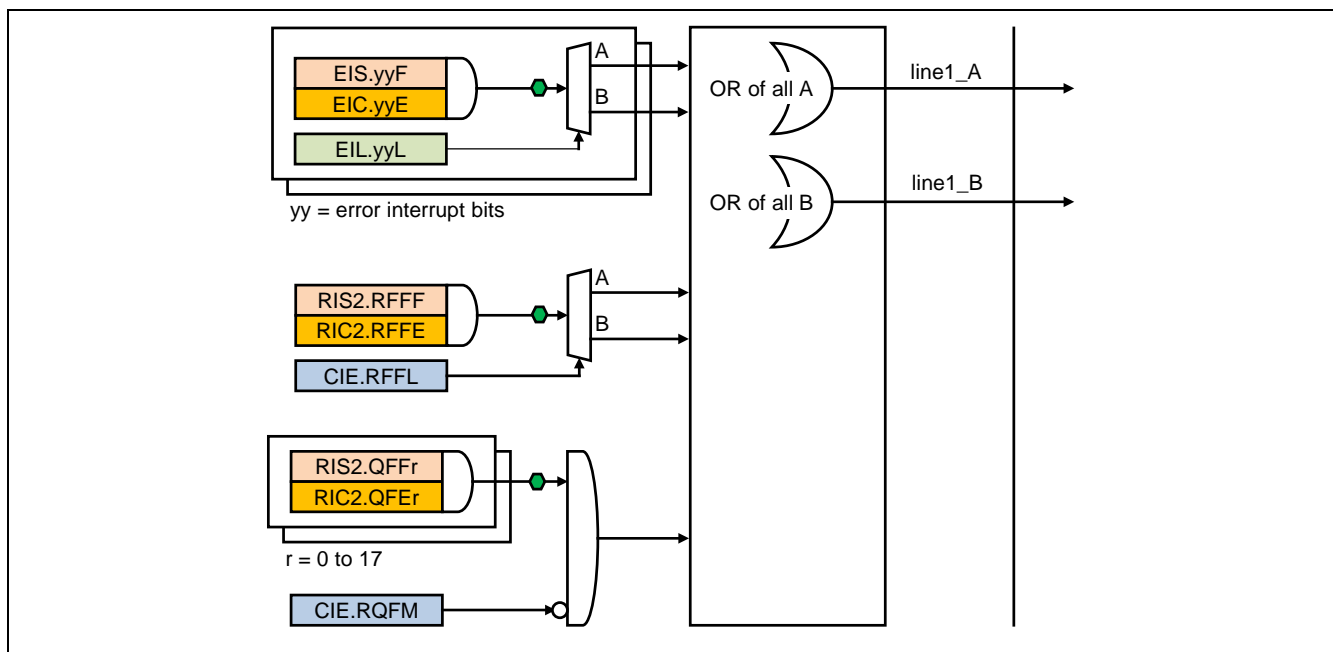


Figure 46.59 Interrupt group 1, error interrupt lines

The abbreviation yy in figure indicates error interrupts: TBF, MFF, TFF, CULFc, CLLFc, SE, QE, MTE, and MRE.

(4) Group 2 (Management related interrupts)

Reception

Rx-FIFO Warning interrupt RIS1.RFWF (Rx-FIFO has reached configured warning level)

Transmission

E-MAC status FIFO warning interrupt TIS.MFWF

E-MAC Status FIFO updated interrupt TIS.MFUF

Timestamp FIFO warning interrupt TIS.TFWF

Timestamp FIFO updated interrupt TIS.TFUF

gPTP

AVTP time captured interrupt GIS.ATCFi

AVTP presentation target matched interrupt GIS.PTMFi

AVTP presentation time overrun interrupt GIS.PTOF (when external trigger before old value is processed)

AVTP presentation time captured interrupt GIS.PTCF

By reading ISS, CPU get an overview of all management related interrupts (ISS.RFWM, ISS.MFWM, ISS.MFUM, ISS.TFWM, ISS.TFUM, ISS.CGIM).

Interrupt group 2 consists on these 2 interrupt lines:

Line2_A (Share the interrupt line: Group0.Line0_DiA, Group1.Line1_A, Group2.Line2_A)

Line2_B (Share the interrupt line: Group0.Line0_DiB, Group1.Line1_B, Group2.Line2_B)

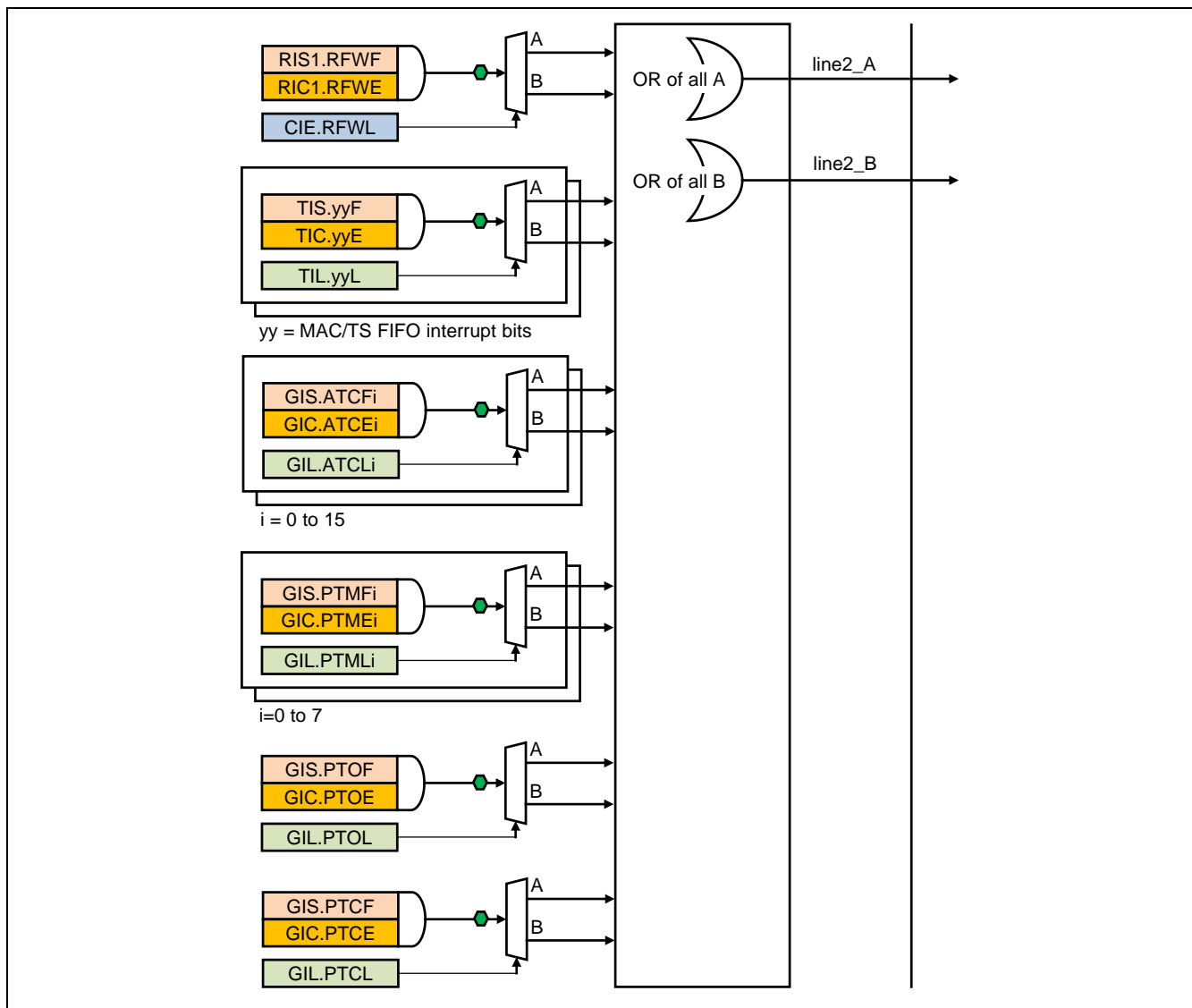


Figure 46.60 Interrupt group 2, management interrupt lines

The abbreviation yy in figure indicates error interrupts: MFW, MFU, TFW, and TFU.

(5) Line 3 (E-MAC interrupt)

The E-MAC interrupt is conveyed when the E-MAC interrupt source is generated.

The general error interrupt state can be checked by reading the E-MAC interrupt summary bit in the interrupt summary status register (ISS.MS).

46.3.12 Flows of Operations

(1) Flow of E-MAC Initialization

Figure 46.61 shows the flow of E-MAC initialization (for AVB mode and full-duplex operation).

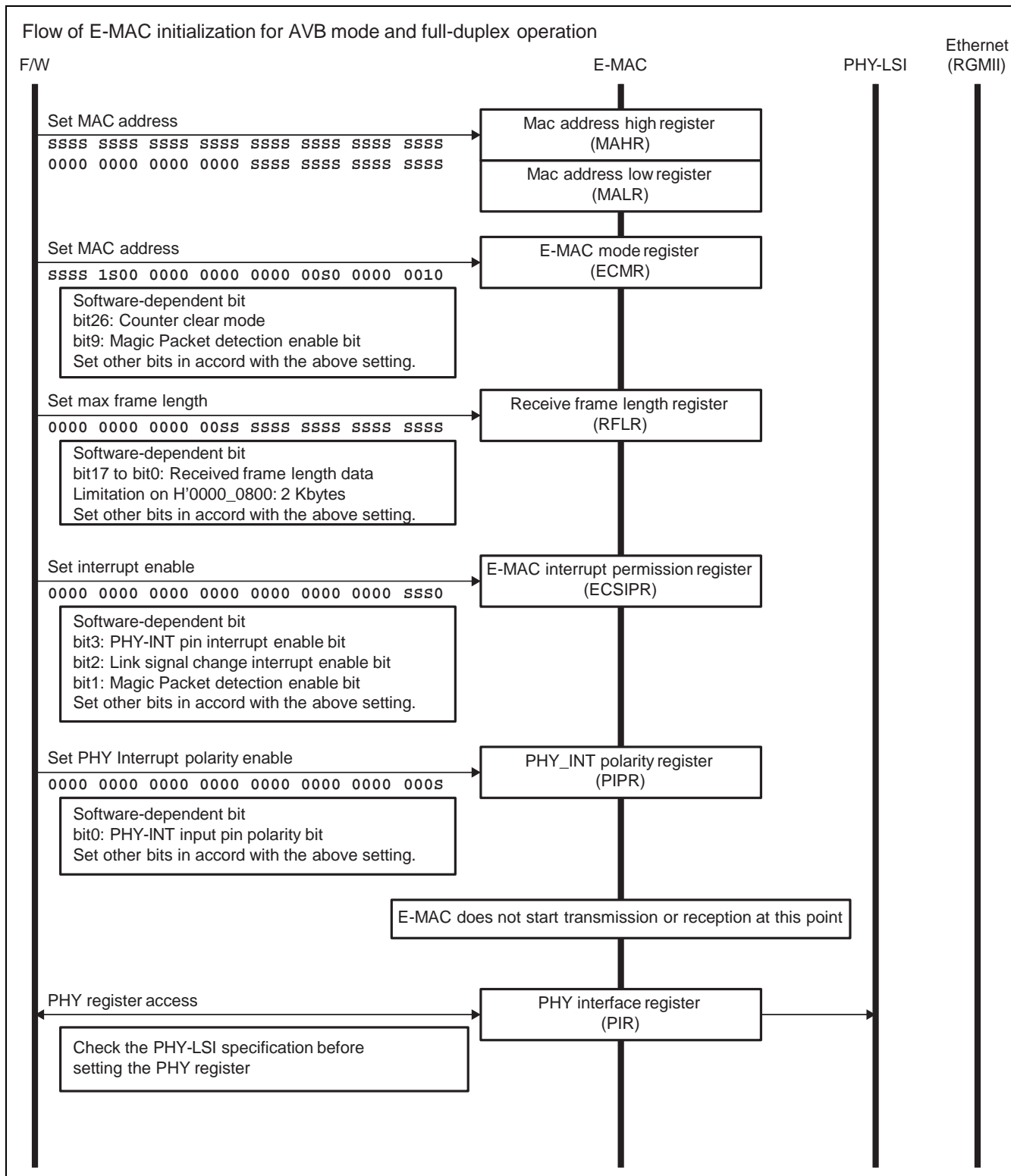


Figure 46.61 Flow of E-MAC Initialization (for AVB Mode and Full-Duplex Operation)

(2) Flow of AVB-DMAC Initialization

Figure 46.62 shows the flow of AVB-DMAC initialization (for AVB mode and full-duplex operation).

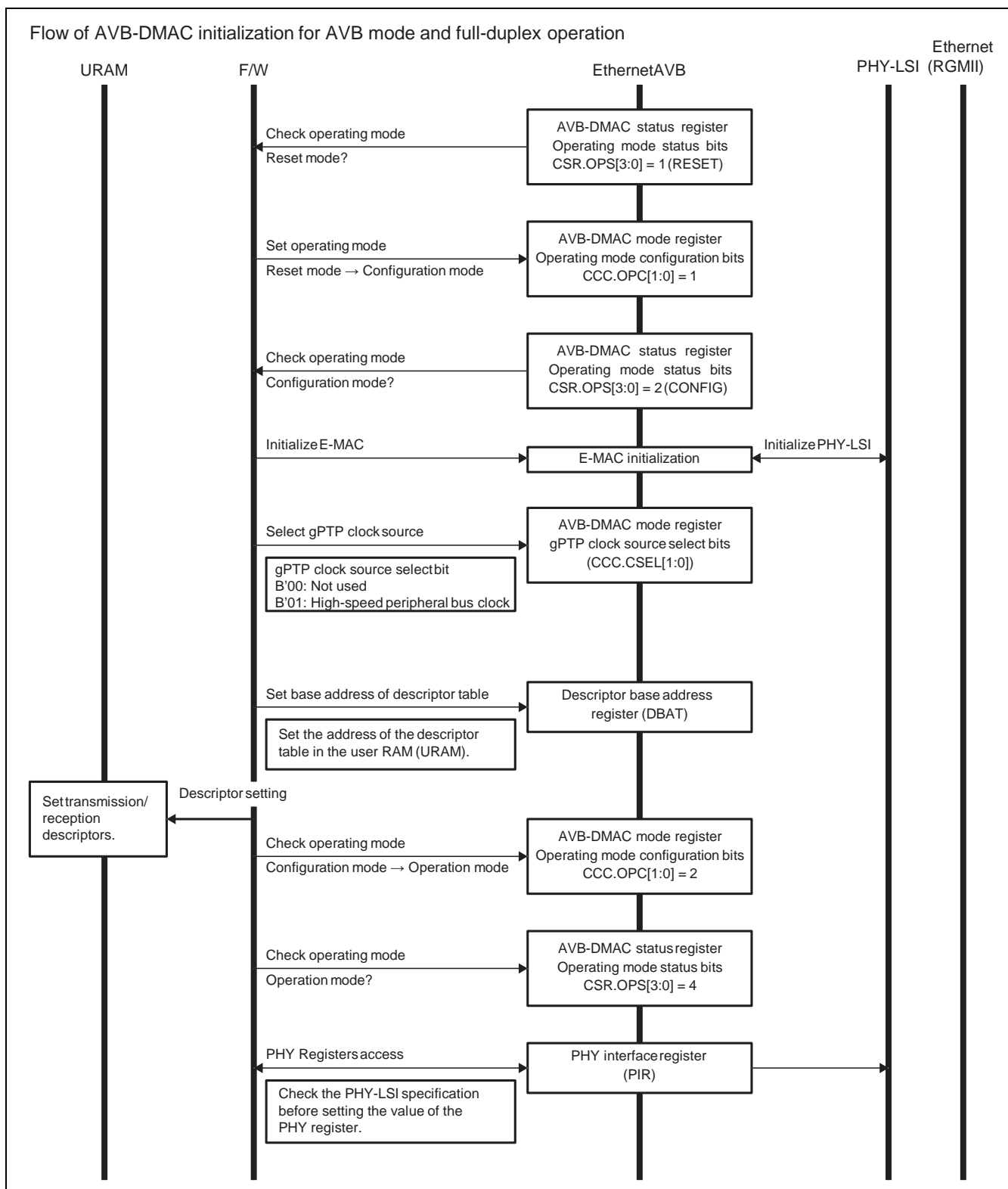


Figure 46.62 Flow of AVB-DMAC Initialization (for AVB Mode and Full-Duplex Operation)

(3) Flow for the AVB-DMAC in Reception

Figure 46.63 shows the flow for the AVB-DMAC in reception (in AVB mode and full-duplex operation).

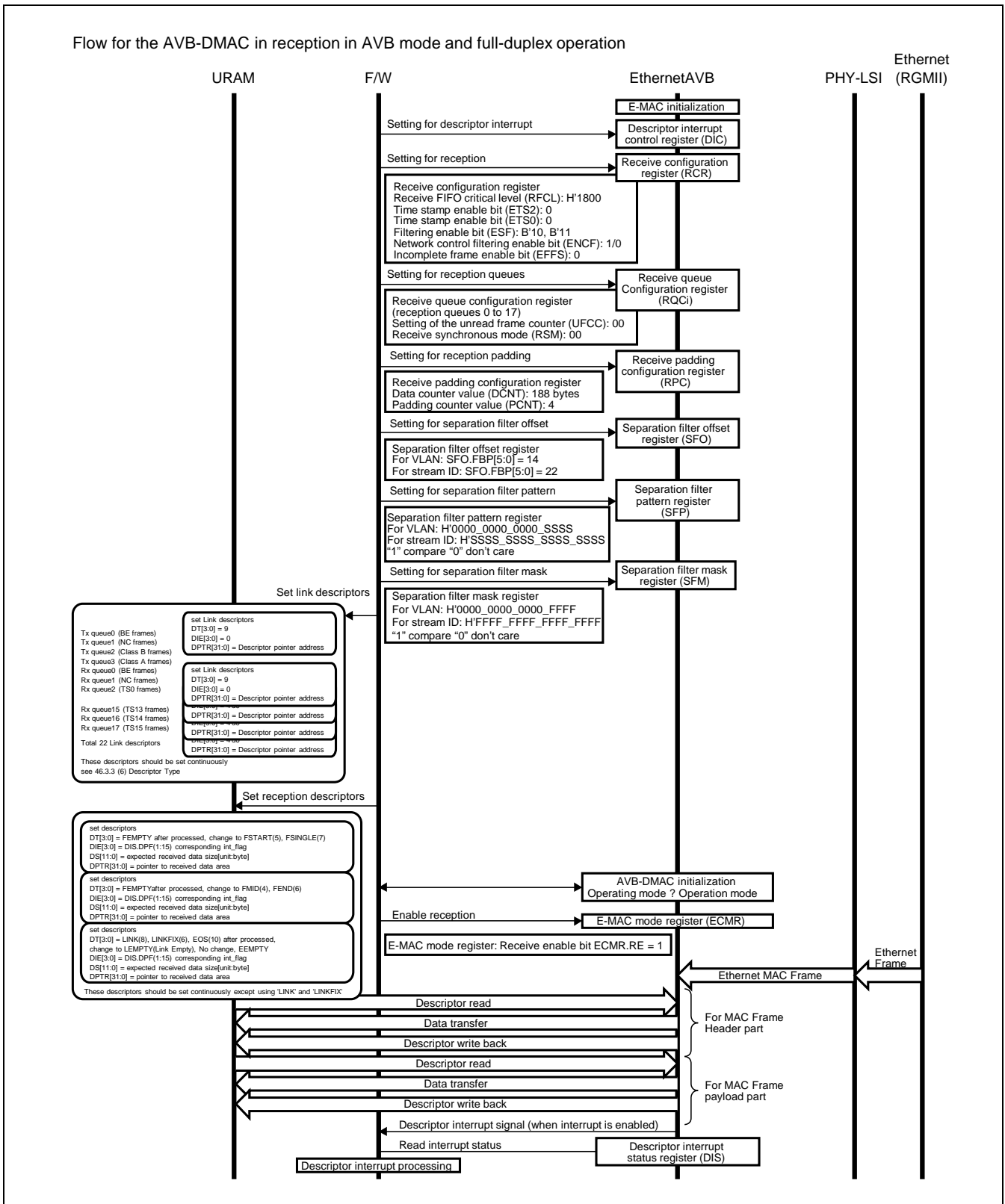


Figure 46.63 Flow for the AVB-DMAC in Reception (in AVB Mode and Full-Duplex Operation)

(4) Flow for the AVB-DMAC in Transmission

Figure 46.64 shows the flow for the AVB-DMAC in transmission (in AVB mode and full-duplex operation).

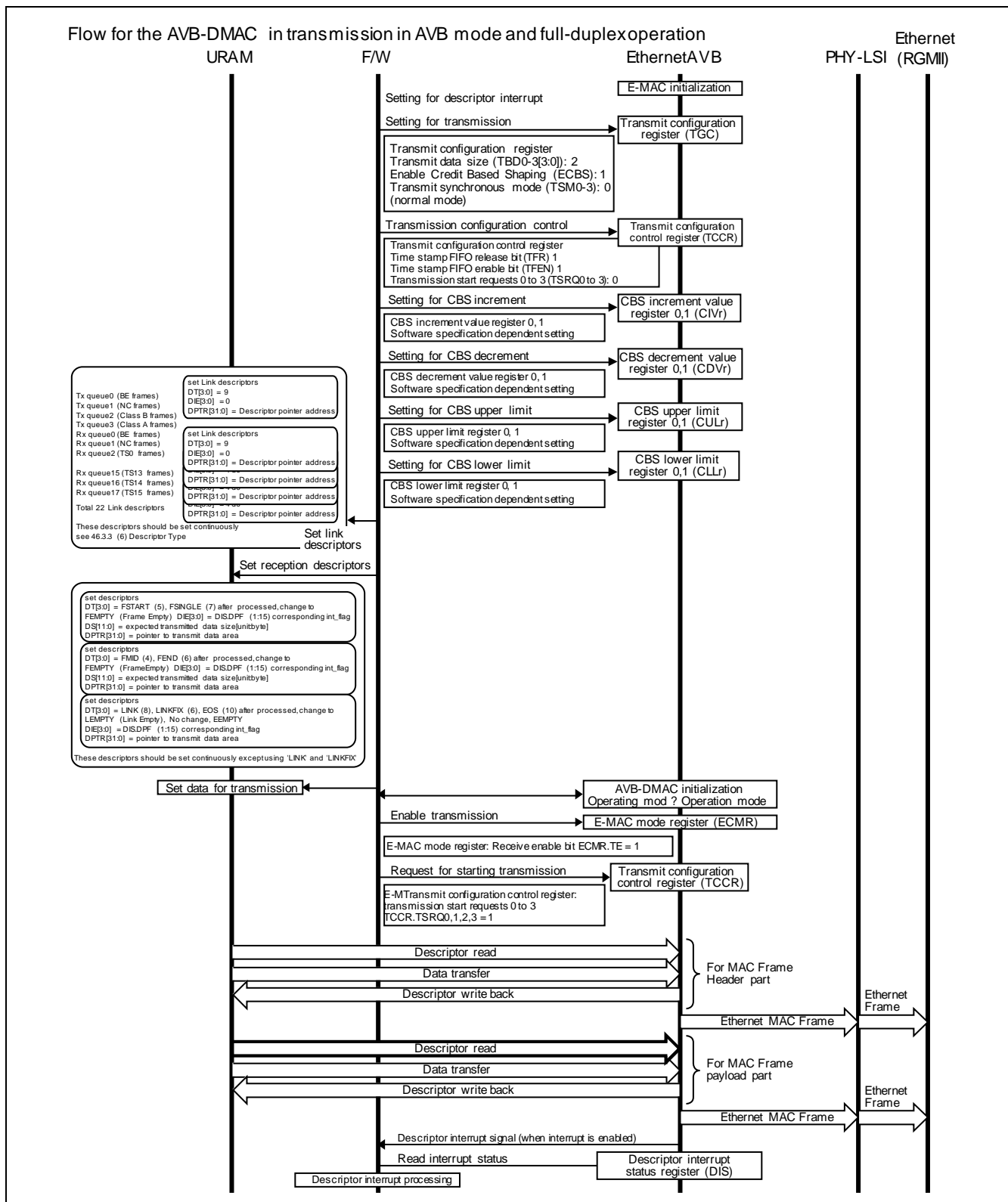


Figure 46.64 Flow for the AVB-DMAC in Transmission (in AVB Mode and Full-Duplex Operation)

(5) Flow for Stopping AVB-DMAC Operation in Reception

Figure 46.65 shows the flow for stopping AVB-DMAC operation in reception (normal, common to all modes).

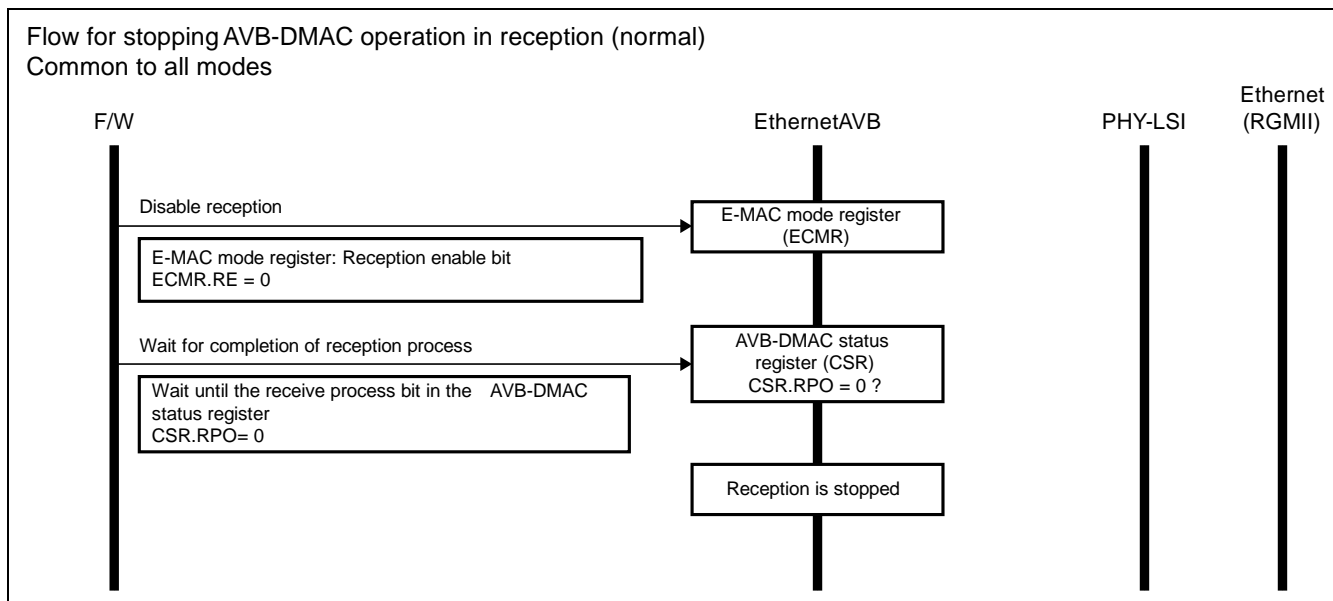


Figure 46.65 Flow for Stopping AVB-DMAC Operation in Reception (Normal, Common to All Modes)

(6) Flow for Stopping AVB-DMAC Operation in Transmission

Figure 46.66 shows the flow for stopping AVB-DMAC operation in transmission (normal, common to all modes).

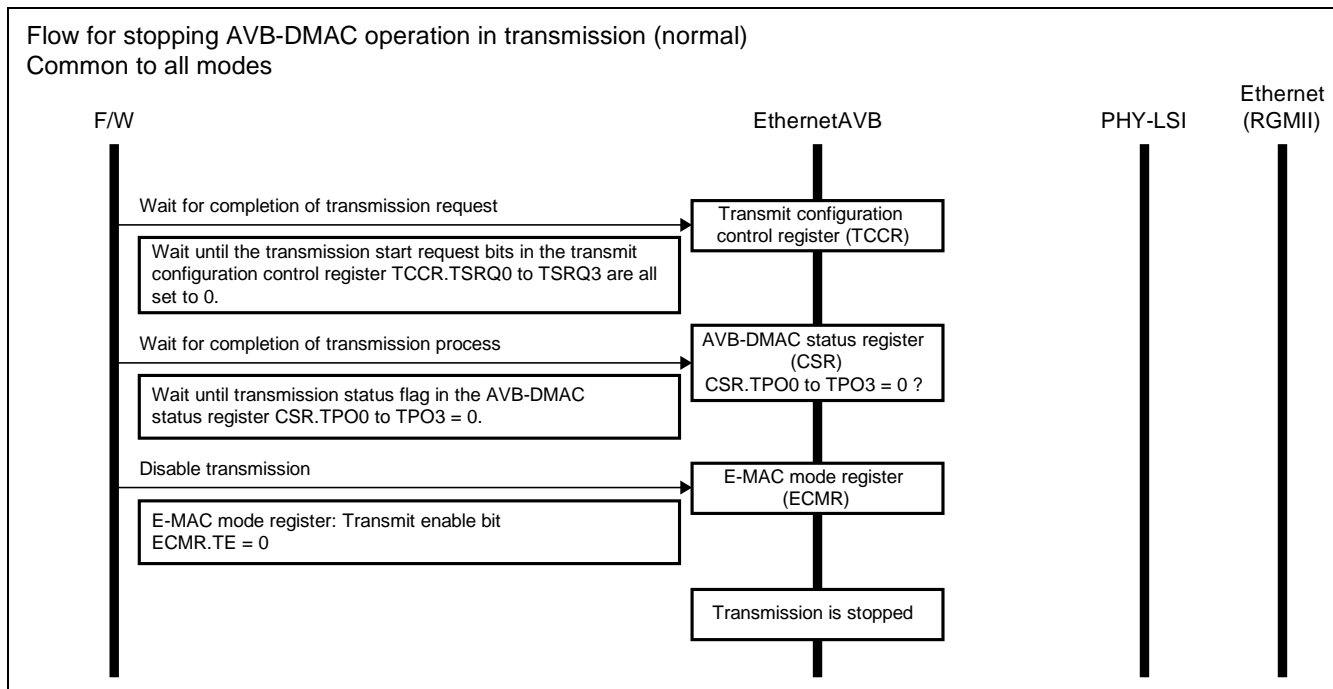


Figure 46.66 Flow for Stopping AVB-DMAC Operation in Transmission (Normal, Common to All Modes)

(7) Flow for Stopping and Resetting the AVB-DMAC

Figure 46.67 shows the flow for stopping and resetting the AVB-DMAC (normal, common to all modes).

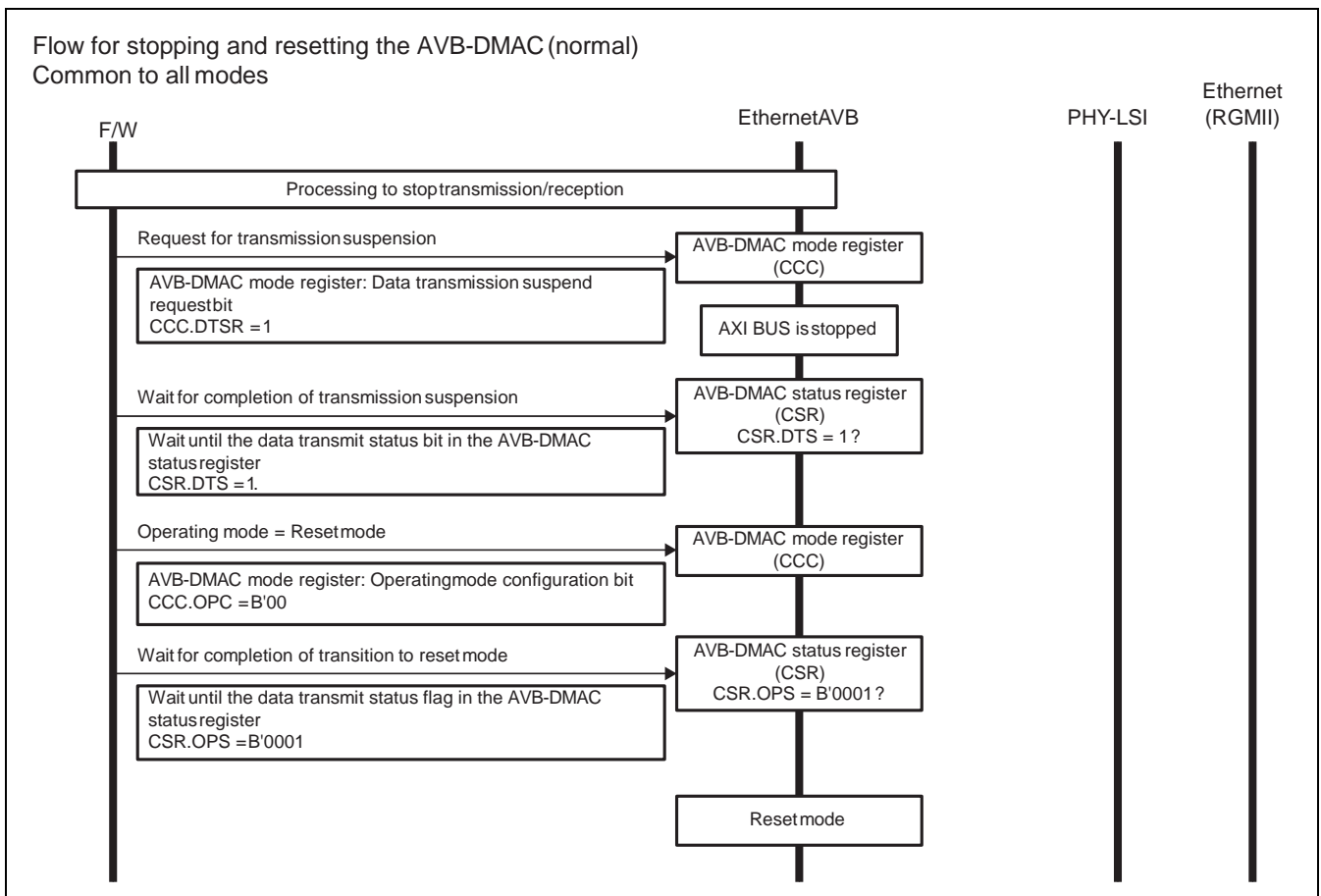


Figure 46.67 Flow for Stopping and Resetting the AVB-DMAC (Normal, Common to All Modes)

(8) Flow for Emergency Stopping the AVB-DMAC

Figure 46.68 shows the flow for emergency stopping the AVB-DMAC (normal, common to all modes).

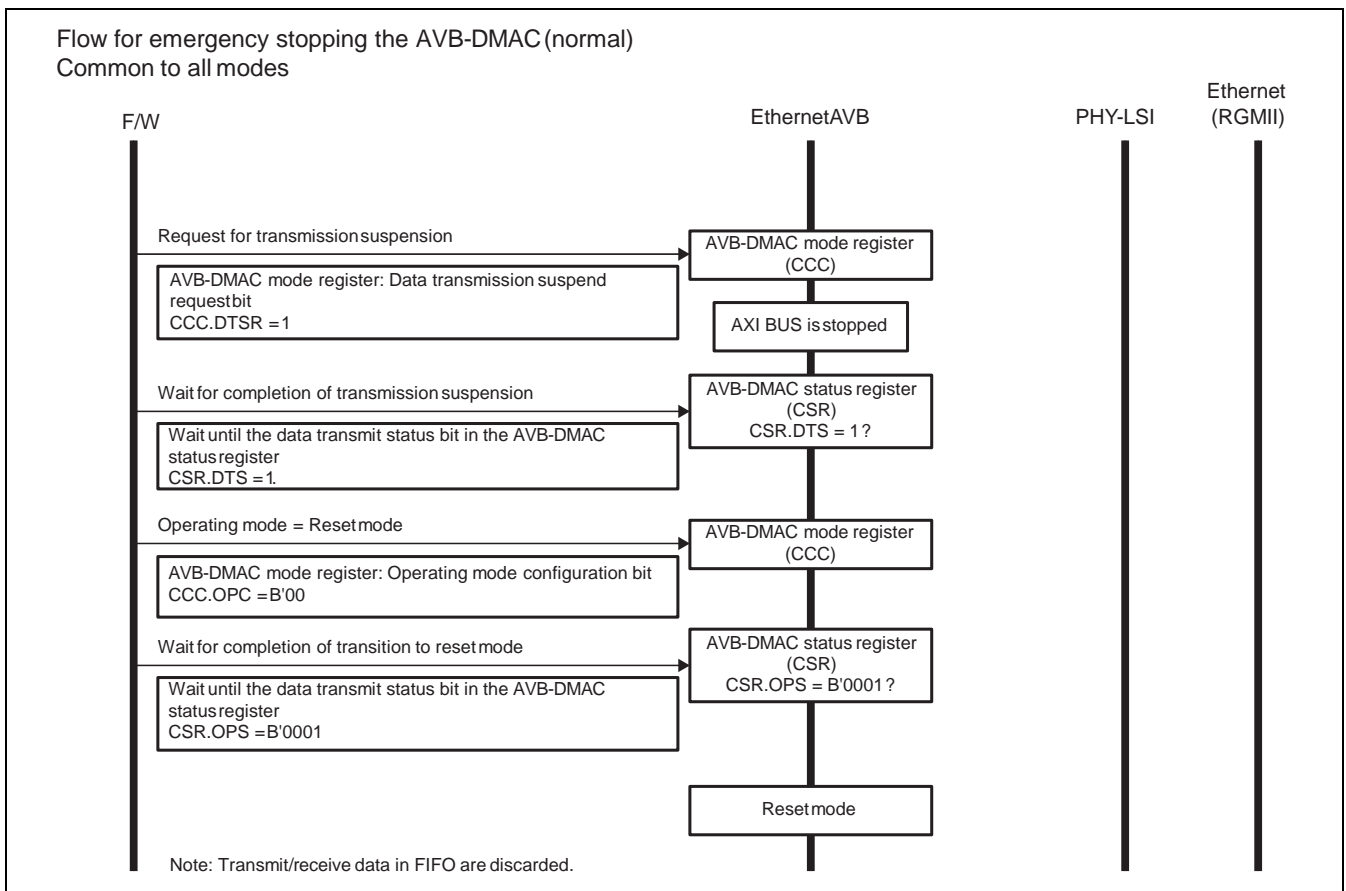


Figure 46.68 Flow for Emergency Stopping the AVB-DMAC (Normal, Common to All Modes)

(9) Flow of gPTP Initialization

Figure 46.69 shows the flow of gPTP initialization (normal, common to all modes).

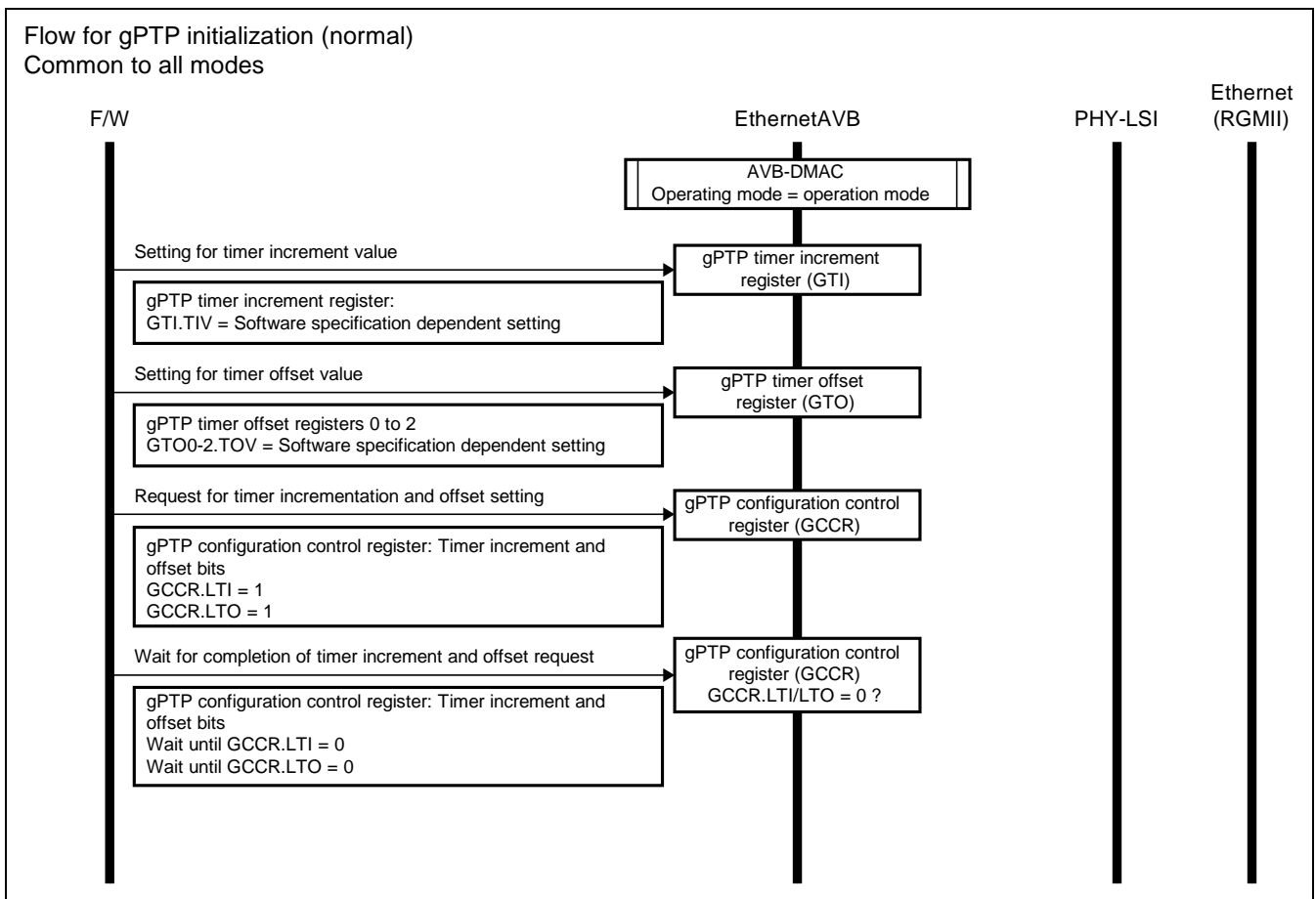


Figure 46.69 Flow of gPTP Initialization (Normal, Common to All Modes)

(10) Flow of gPTP Time Stamping in Transmission

Figure 46.70 shows the flow of gPTP time stamping in transmission (normal, common to all modes).

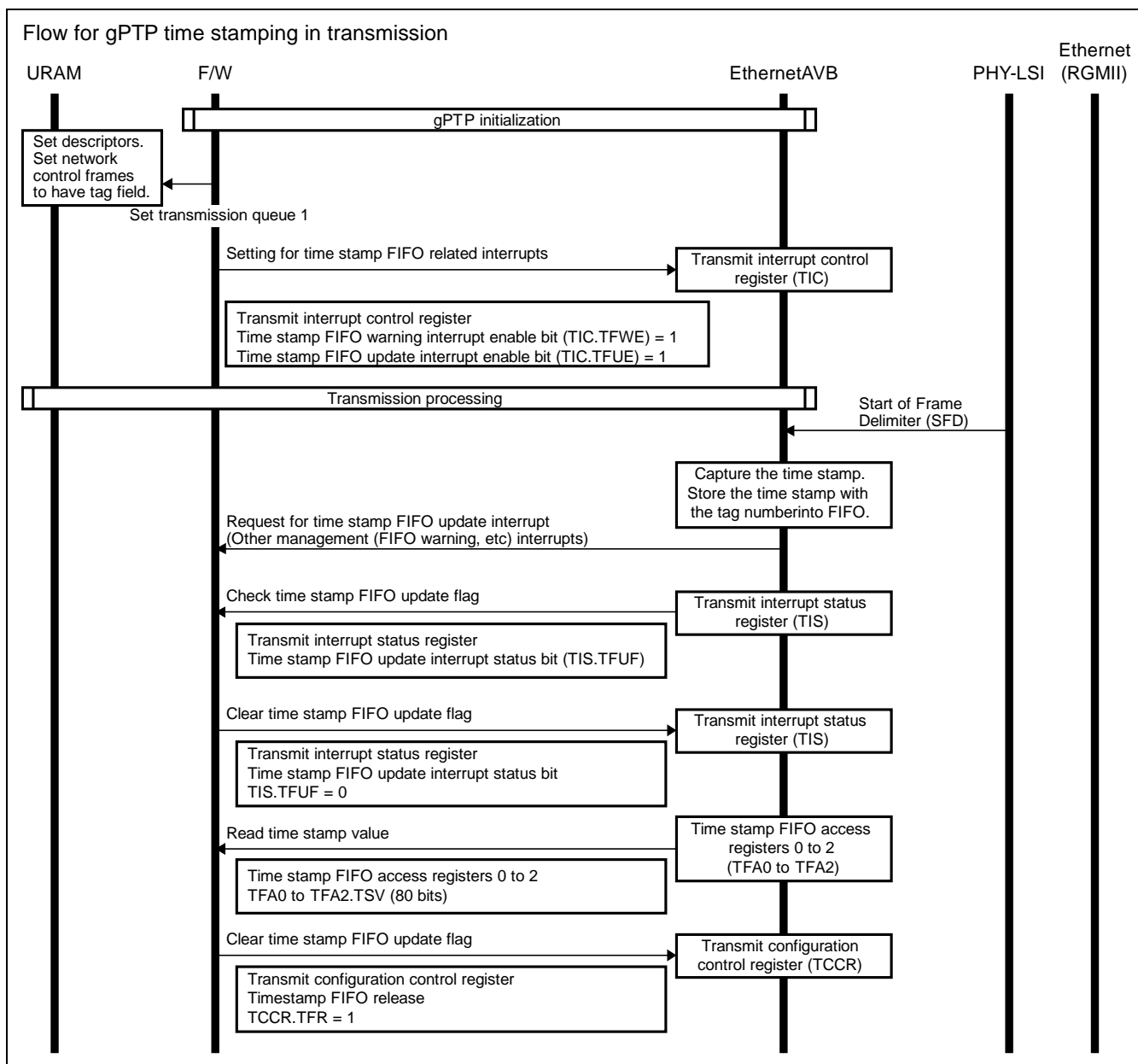


Figure 46.70 Flow of gPTP Time Stamping in Transmission (Normal, Common to All Modes)

(11) Flow of gPTP Time Stamping and Synchronization in Reception

Figure 46.71 shows the flow of gPTP time stamping and synchronization in reception (normal, common to all modes).

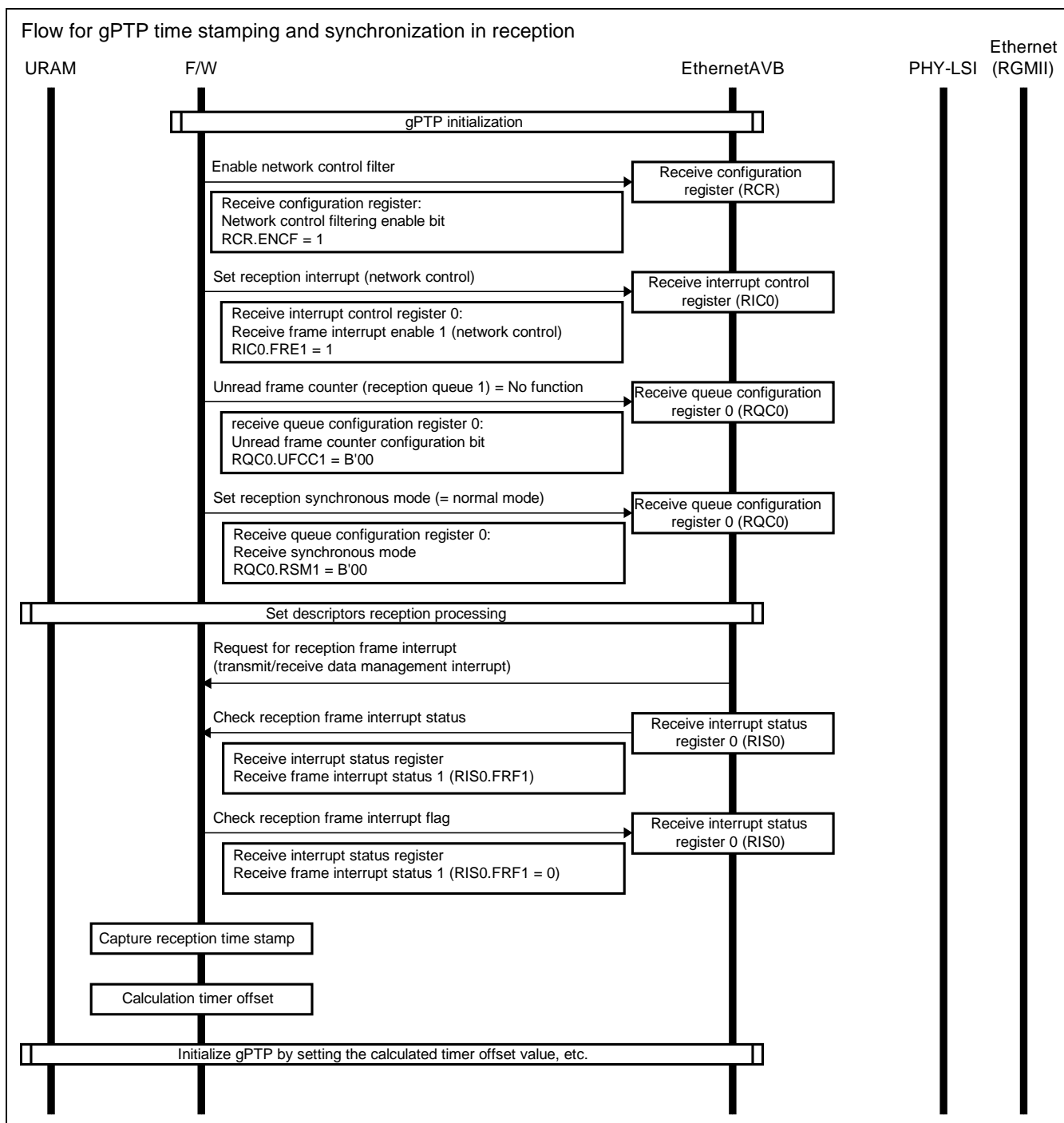


Figure 46.71 Flow of gPTP Time Stamping and Synchronization in Reception (Normal, Common to All Modes)

(12) Flow of Capturing gPTP Presentation Times

Figure 46.72 shows the flow of capturing gPTP presentation times (common to all modes).

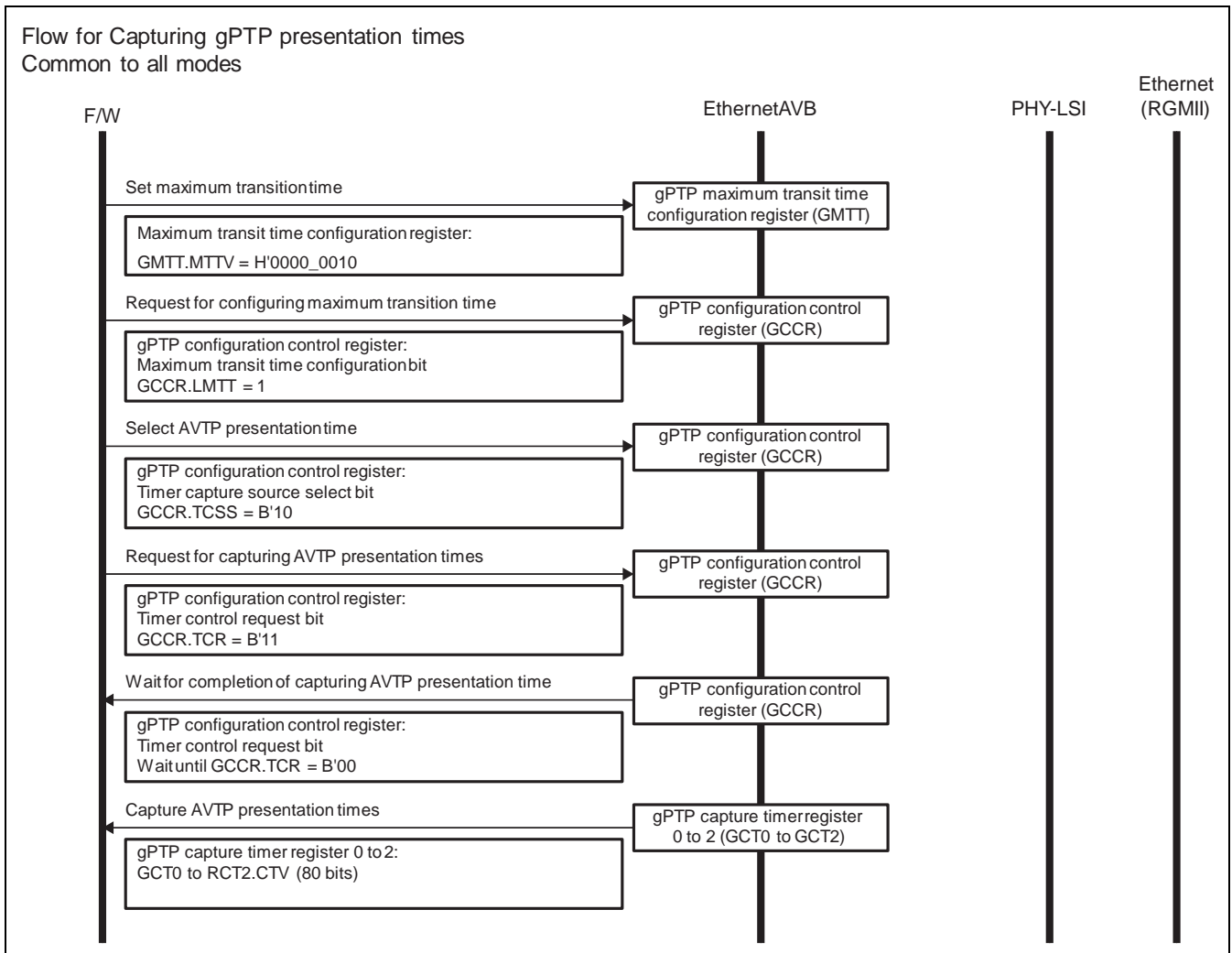


Figure 46.72 Flow of Capturing gPTP Presentation Times (Common to All Modes)

(13) Flow of AVTP Presentation Time Comparison

Figure 46.73 shows the flow of AVTP presentation time comparison (common to all modes).

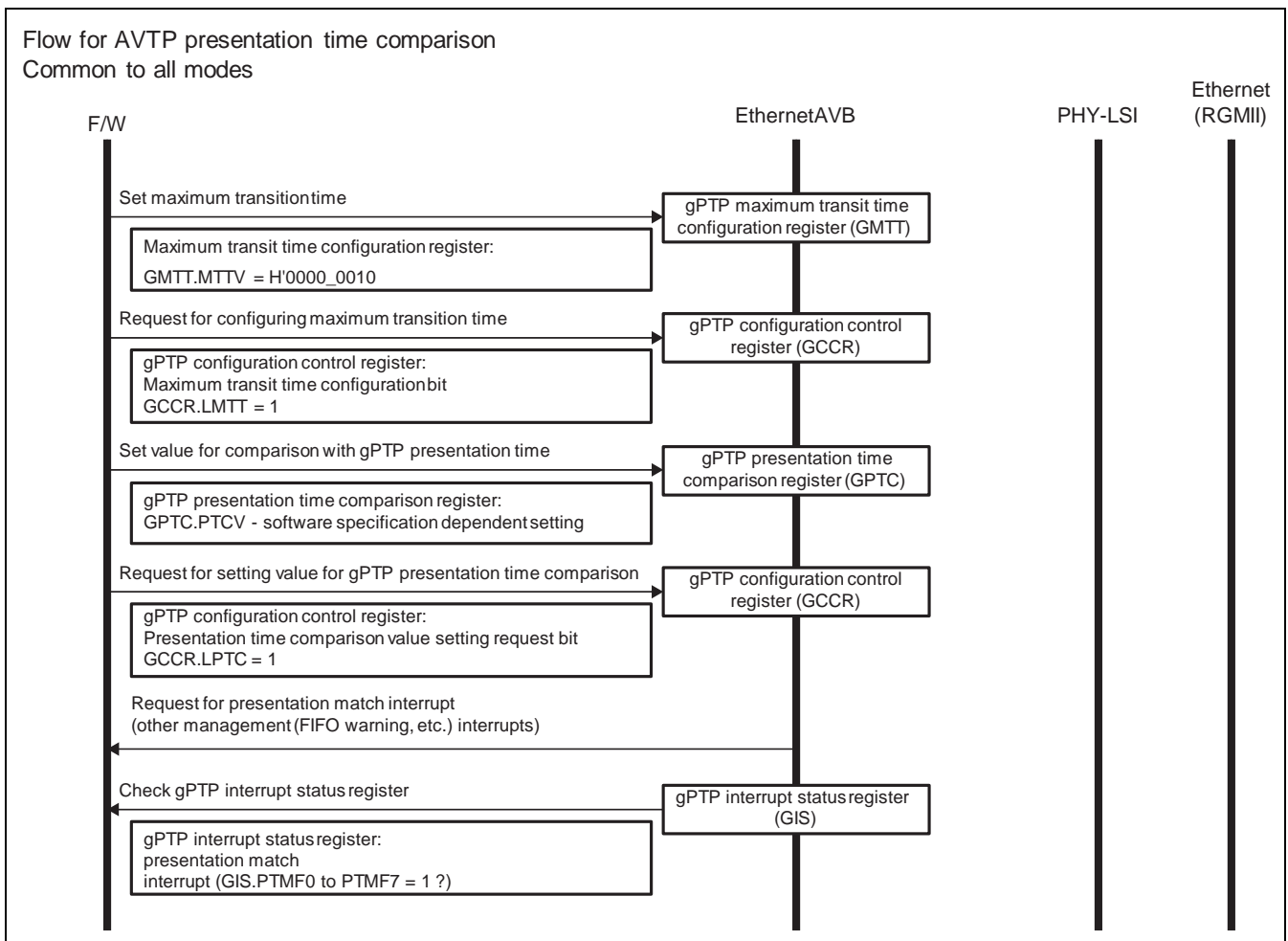


Figure 46.73 Flow of AVTP Presentation Time Comparison (Common to All Modes)

(14) Flow of Loopback Mode Operation

Figure 46.74 shows the flow of loopback mode operation.

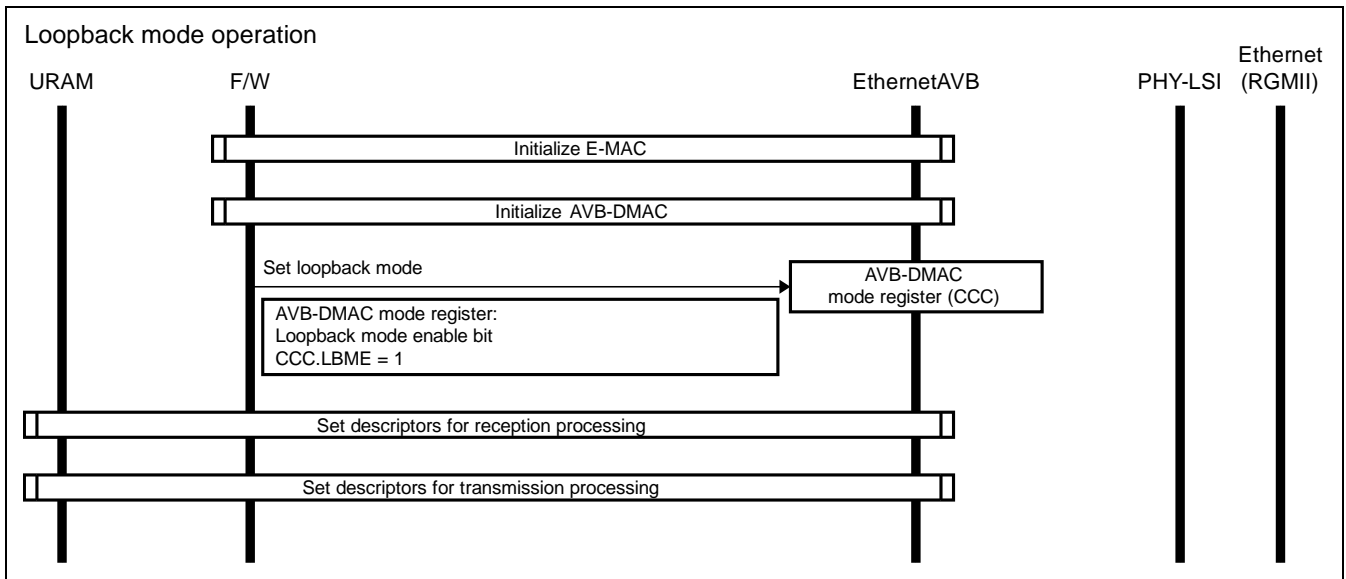


Figure 46.74 Flow of Loopback Mode Operation

46.3.13 Connection to PHY-LSI

(1) RGMII Frame Transmission/Reception Timing

Each RGMII frame transmission/reception timing is shown in Figure 46.75 to Figure 46.82.

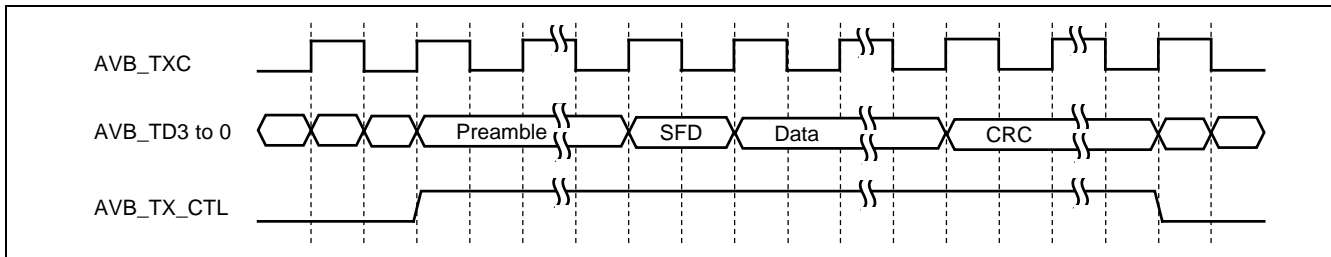


Figure 46.75 RGMII Frame Transmit Timing (Normal Transmission)

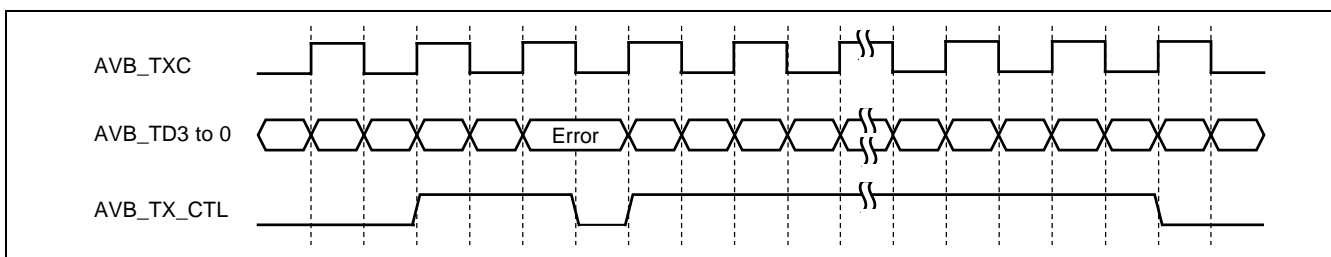


Figure 46.76 RGMII Frame Transmit Timing (Transmit Error)

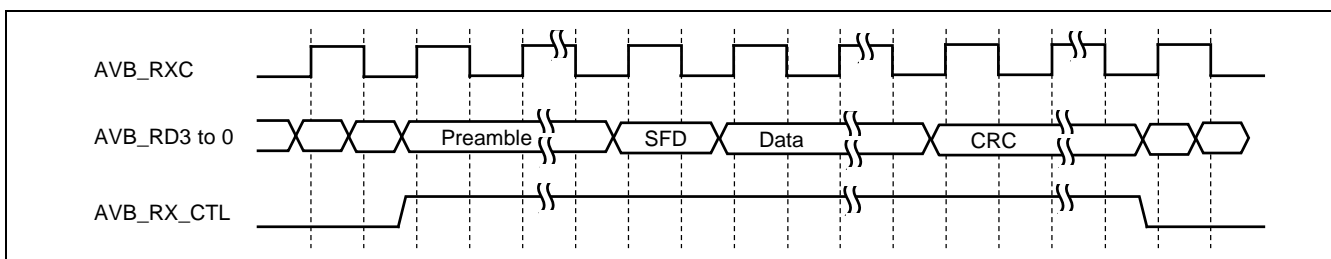


Figure 46.77 RGMII Frame Receive Timing (Normal Reception)

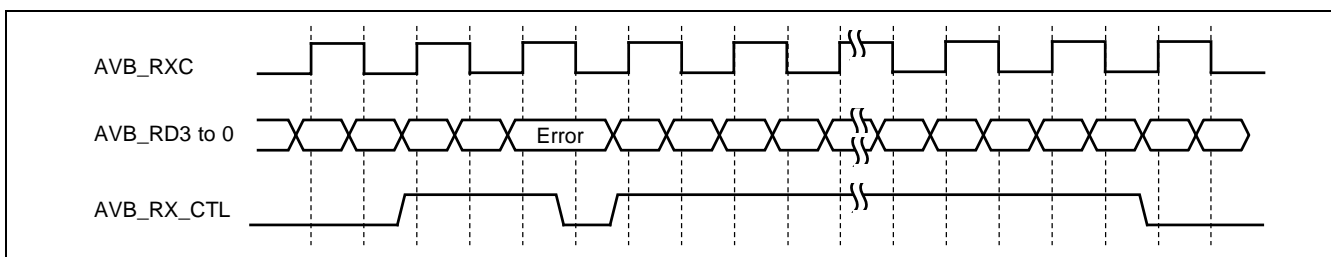


Figure 46.78 RGMII Frame Receive Timing (Reception Error)

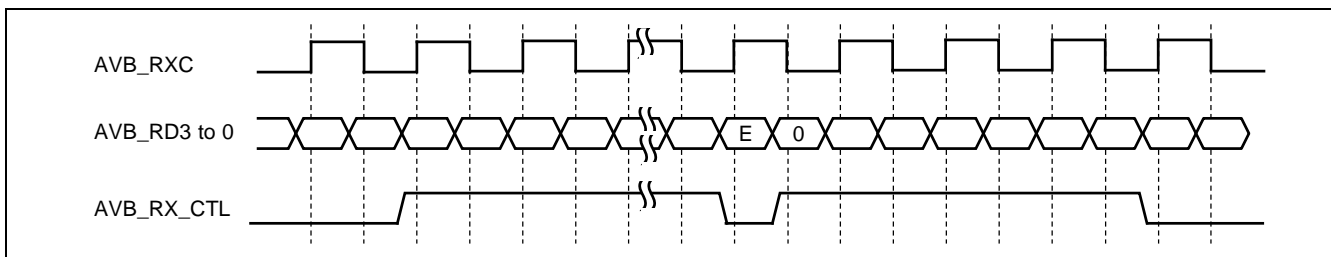


Figure 46.79 RGMII Frame Receive Timing (False Carrier indication)

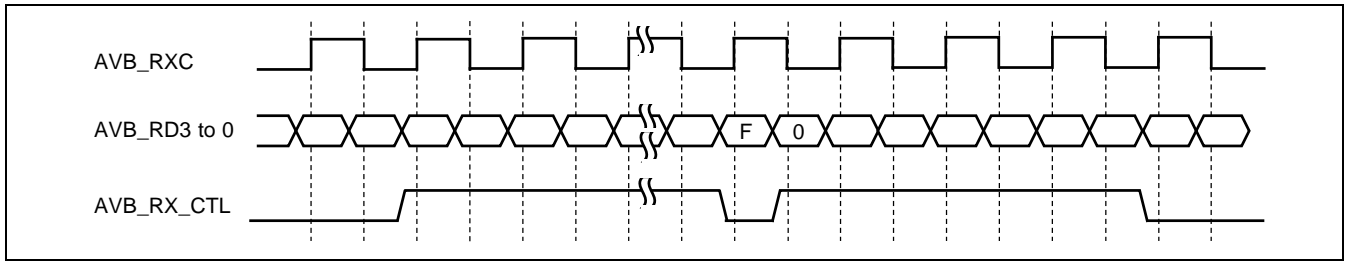


Figure 46.80 RGMII Frame Receive Timing (Carrier Extend)

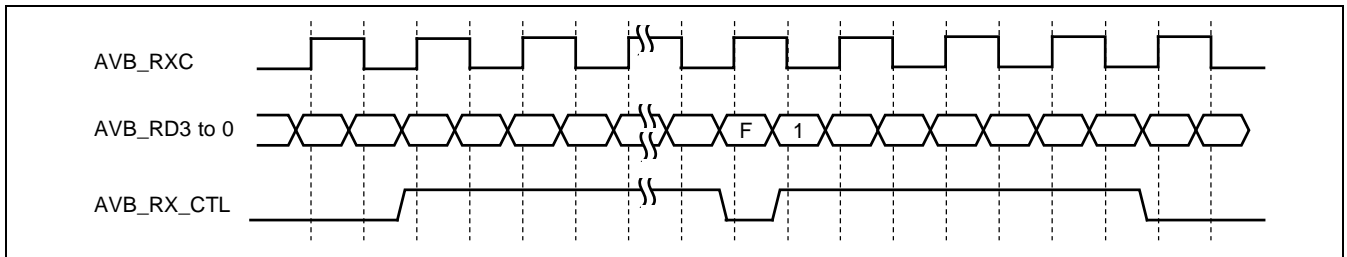


Figure 46.81 RGMII Frame Receive Timing (Carrier Extend Error)

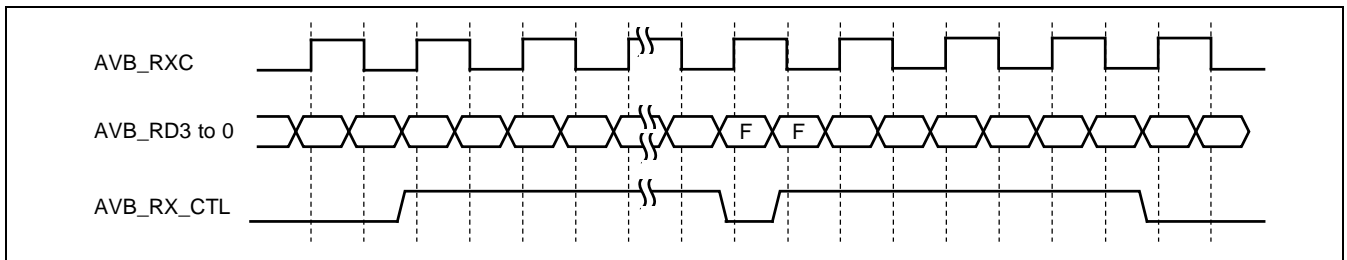


Figure 46.82 RGMII Frame Receive Timing (Carrier Sense)

(2) Accessing MII Management Registers

MII management registers in the PHY-LSI are accessed via PIR in this LSI. PIR is used as a serial interface conforming to the MII frame format specified in IEEE802.3u.

(a) MII Management Frame Format

Figure 46.83 shows the format of an MII management frame. To access an MII management register, a management frame is implemented by the program in accordance with the procedures shown in MII Management Register Access Procedure.

| Access Type | MII Management Frame | | | | | | | |
|----------------|----------------------|----|----|-------|-------|----|-------|------|
| Item | PRE | ST | OP | PHYAD | REGAD | TA | DATA | IDLE |
| Number of bits | 32 | 2 | 2 | 5 | 5 | 2 | 16 | |
| Read | 1..1 | 01 | 10 | 00001 | RRRRR | Z0 | D...D | |
| Write | 1..1 | 01 | 01 | 00001 | RRRRR | 10 | D...D | X |

[Legend]

PRE: 32 consecutive 1 s

ST: Write of B'01 indicating start of frame

OP: Write of code indicating access type

PHYAD: Write of B'0_0001 if the PHY-LSI address is 1 (sequential write starting with the MSB).
This bit changes depending on the PHY-LSI address.

REGAD: Write of B'000q if the register address is 1 (sequential write starting with the MSB).
This bit changes depending on the PHY-LSI register address.

TA: Time for switching data transmission source on MII interface
(a) Write: 10 written

(b) Read: Bus release (notation: Z0) performed

DATA: 16-bit data. Sequential write or read from MSB

(c) Write: 16-bit data write

(d) Read: 16-bit data read

IDLE: Wait time until next MII management format input

(a) Write: Independent bus release (notation: X) performed

(b) Read: Bus already released in TA: control unnecessary

Figure 46.83 MII Management Frame Format

(b) MII Management Register Access Procedure

The program accesses MII management registers via PIR. Access is implemented by a combination of 1-bit-unit data write, 1-bit-unit data read, bus release, and independent bus release. Figure 46.84 to Figure 46.87 show the MII management register timing. The timing will differ depending on the PHY-LSI type.

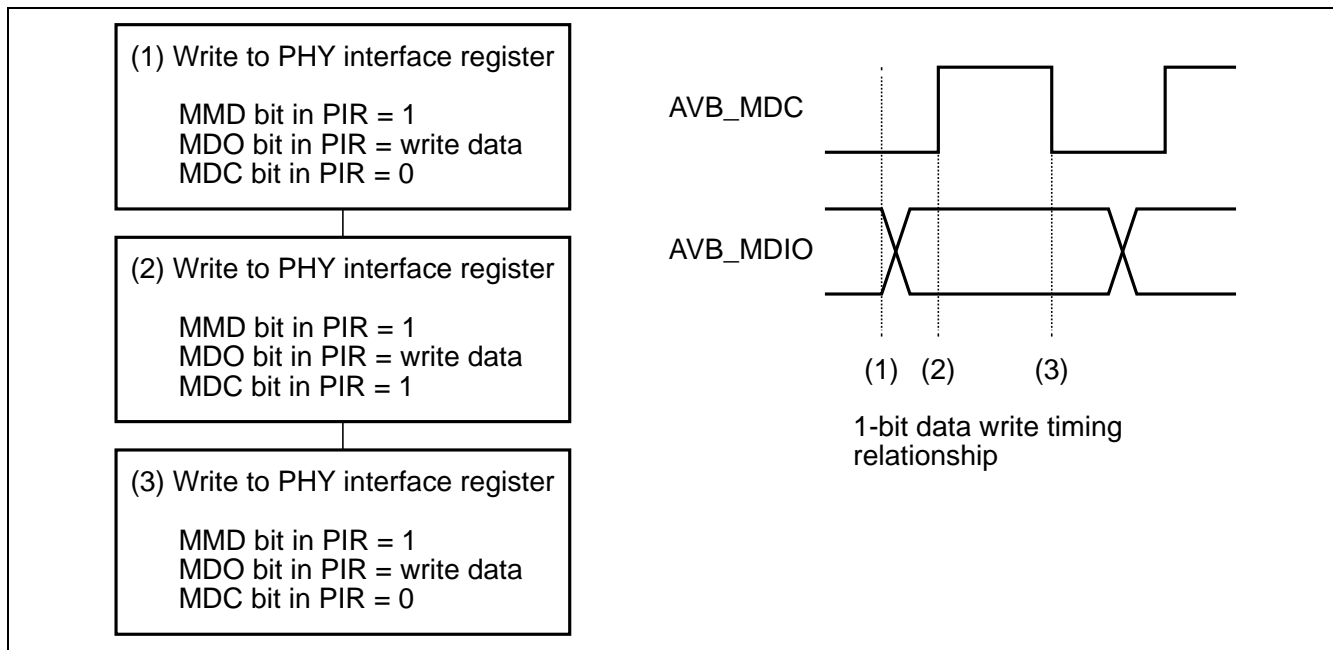


Figure 46.84 1-Bit Data Write Flowchart

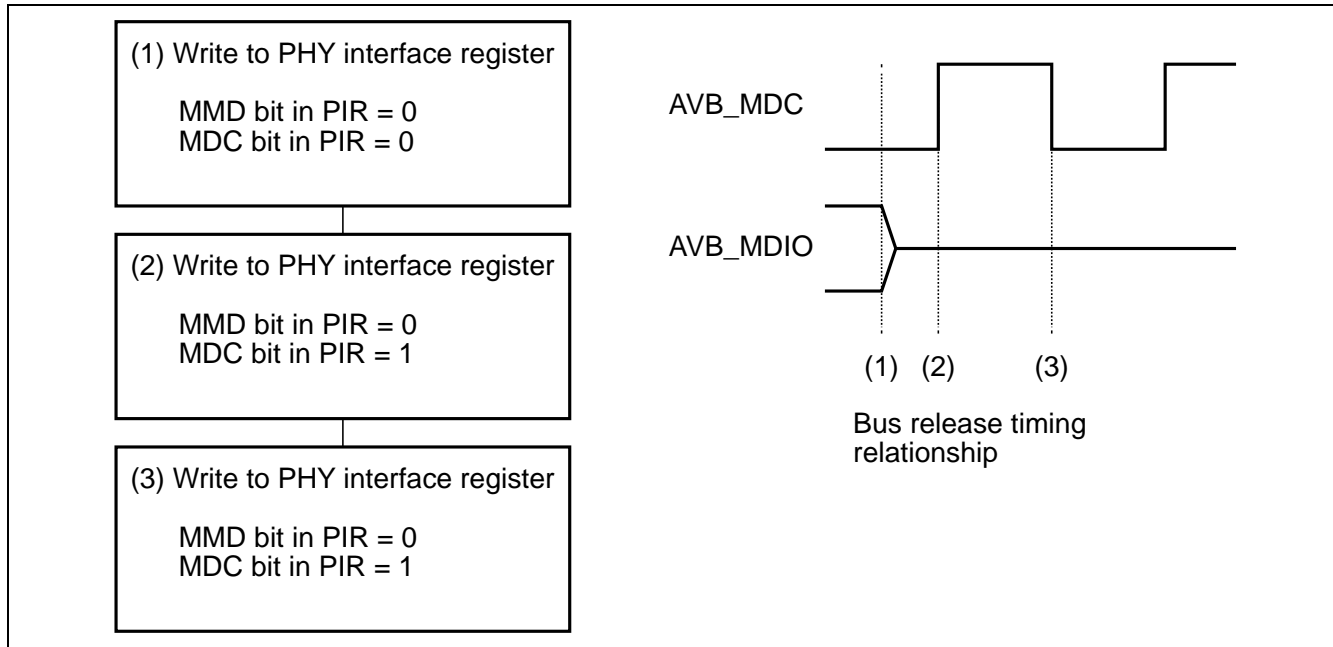


Figure 46.85 Bus Release Flowchart (TA in Read in Figure 46.83)

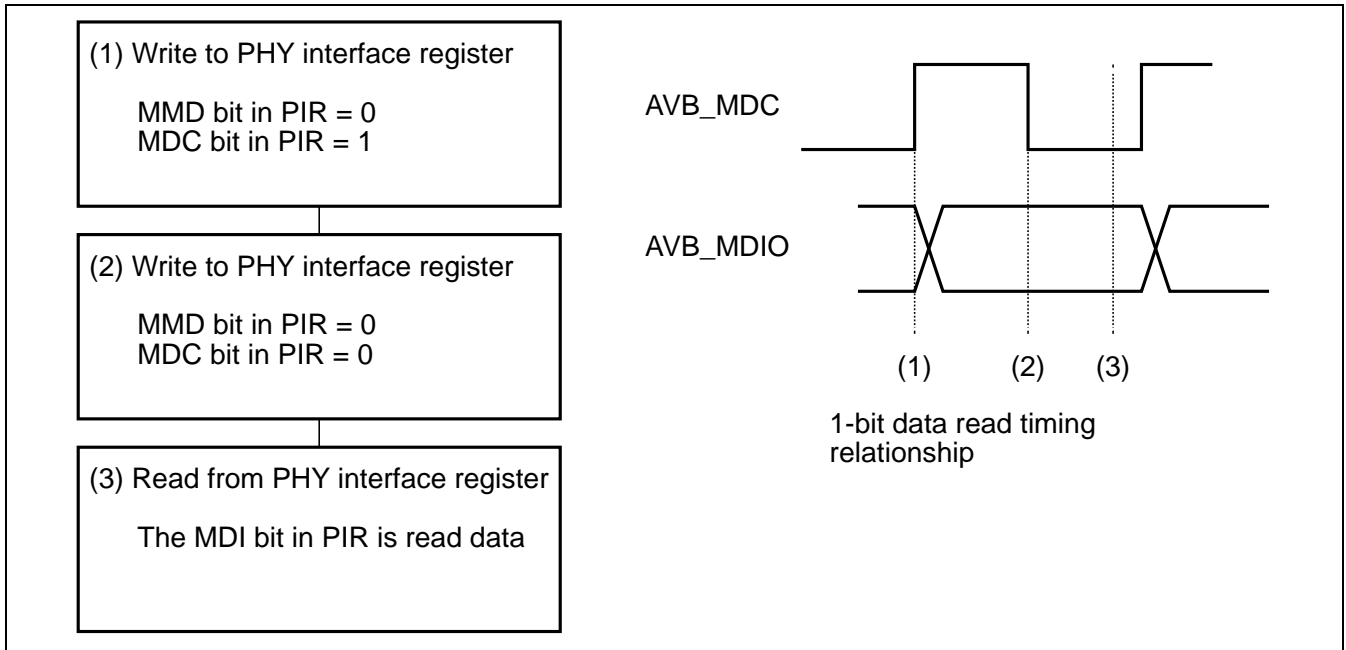


Figure 46.86 1-Bit Data Read Flowchart

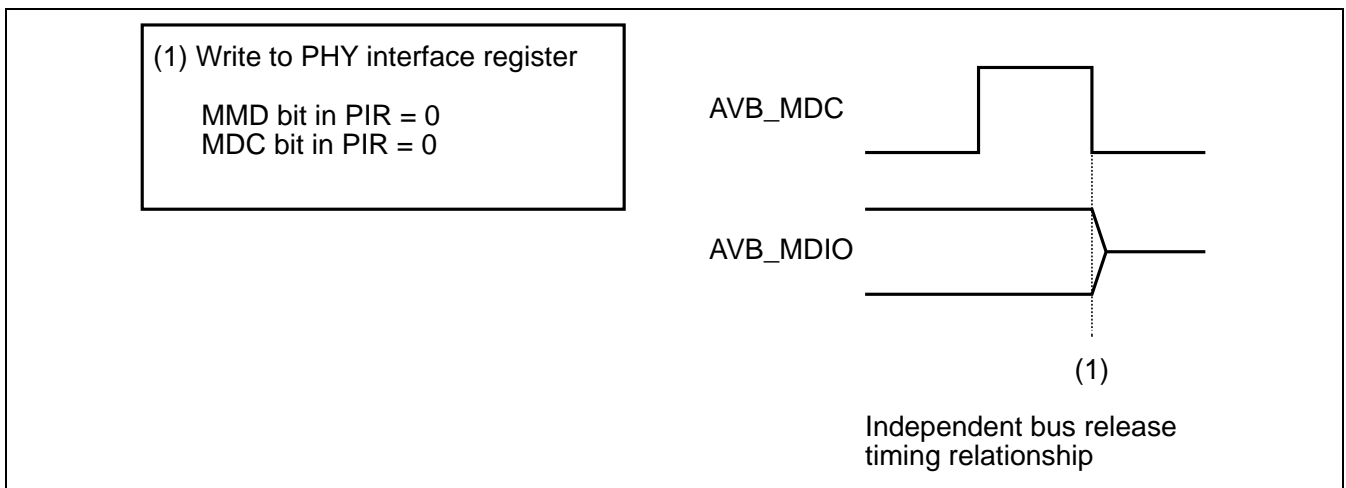


Figure 46.87 Independent Bus Release Flowchart (IDLE in Write in Figure 46.83)

46.3.14 Usage Notes

(a) Reception is Ignored when AVB_RX_CTL indicates an error from PHY is Flagged at 1st Data Byte

The issue is limited to reception of frames with physical faults. Flagging of occurrence in SFR is missing. When EthernetAVB-IF is configured to store faulty frames in reception queue 0 (BE) by RCR.EFFS there is no storage.

(b) Unexpected Data are Exported when AVB_RX_CTL indicates an error is Flagged at 8th Data Byte

The issue is limited to reception of frames with physical faults.

Flagging of occurrence in SFR is invalid. When EthernetAVB-IF is configured to store faulty frames in reception queue 0 (BE) by RCR.EFFS corrupted data is stored and MAC status code is wrong.

(c) Receive Frame Interrupt and Descriptor Interrupt may be Issued before Completion of Writing Data

When receive frame interrupt and descriptor interrupt are issued, whether writing data is completed by DESCR.DT in a descriptor can be confirmed.

And CDARq register has the descriptor address which EthernetAVB-IF is processing currently or is going to process at the next time.

The descriptors before one indicated by CDARq register are scope for receive frame interrupt and descriptor interrupt scope.

If there is a descriptor which wait for HW processing (ex. FEMPTY), it's delayed. Confirm the descriptor again after waiting for 1ms.

46.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

46.4.1 Checksum Calculation of Ethernet Frames

This LSI is capable of calculating the checksum data of the received frames. Only the data fields of the Ethernet frames are subject to checksum calculation. Specifically, a data field follows the length/type field and is followed by the CRC field. Figure 46.88 shows schematics indicating which parts of the Ethernet frames are calculated. Calculation involves 16-bit addition only; it does not involve bit inversion. Note that when the checksum data is valid, the CRC data (4 bytes) is not transferred as a receive frame, and the checksum data (sum data) is added automatically. Figure 46.89 shows schematics of Ethernet frames to which the checksum data has been added.

Note: Also for the frames with VLANtag inserted, the 15th byte from the top and the following bytes before the CRC field are subject to calculation.

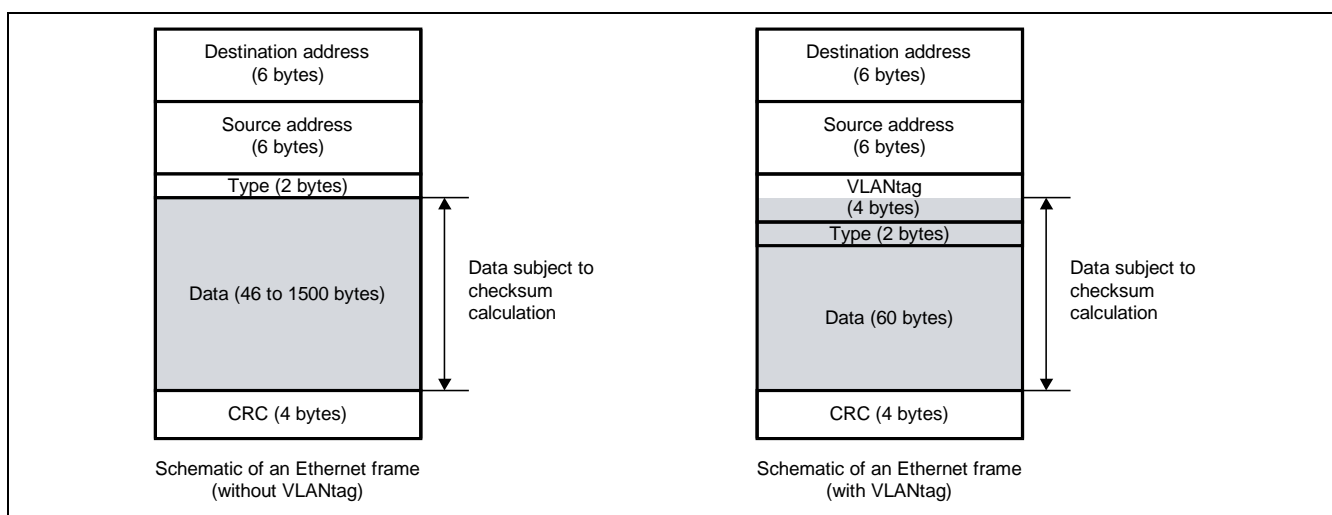


Figure 46.88 Data Subject to Checksum Calculation

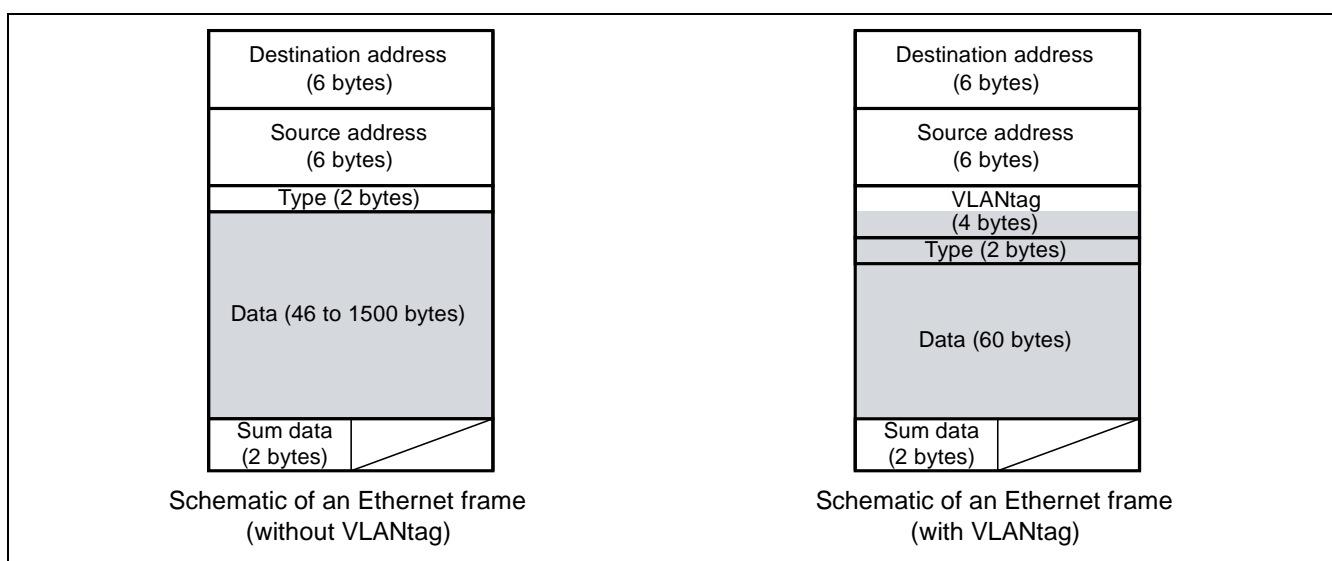


Figure 46.89 Data after Checksum Data Addition

46.4.2 Data Transfer Function Stops after Read Access Error

If EthernetAVB-IF gets an access error when reading from work RAM, it may happen that no further work RAM accesses are issued. So transfer of receive/transmit frames is halted. By using the data transfer suspend function SW can observe if data transfer function is available or not.

Perform one of given SW flows which are triggered by this start condition: when AXI read access error has been flagged by EIS.QEF is 1, and ESR.ET is B'0000 or B'0100; additionally when ESR.EIL is 1 (this is an indication that an work RAM read error may be lost).

There are two possible SW flows given. Figure 46.90 applies always the SW reset, independently if this is required or not. Figure 46.91 checks in addition if SW reset is required; checking CSR.DTS after wait gives an indication that issue has been occurred.

Note that the wait after write of CCC.DTSR is mandatory to prevent cases where issue has not been occurred.

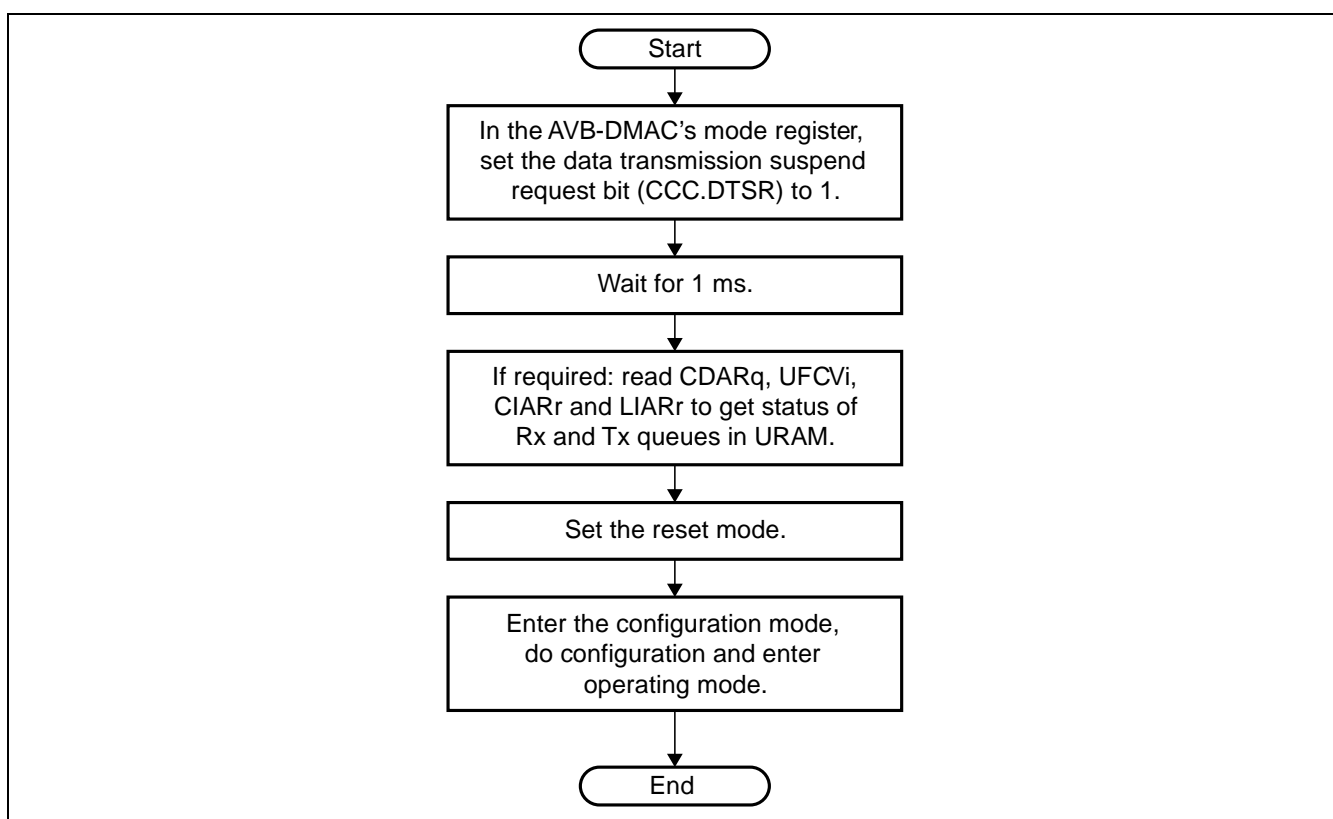


Figure 46.90 Data Transfer Function Stop Flow1 after Read Access Error

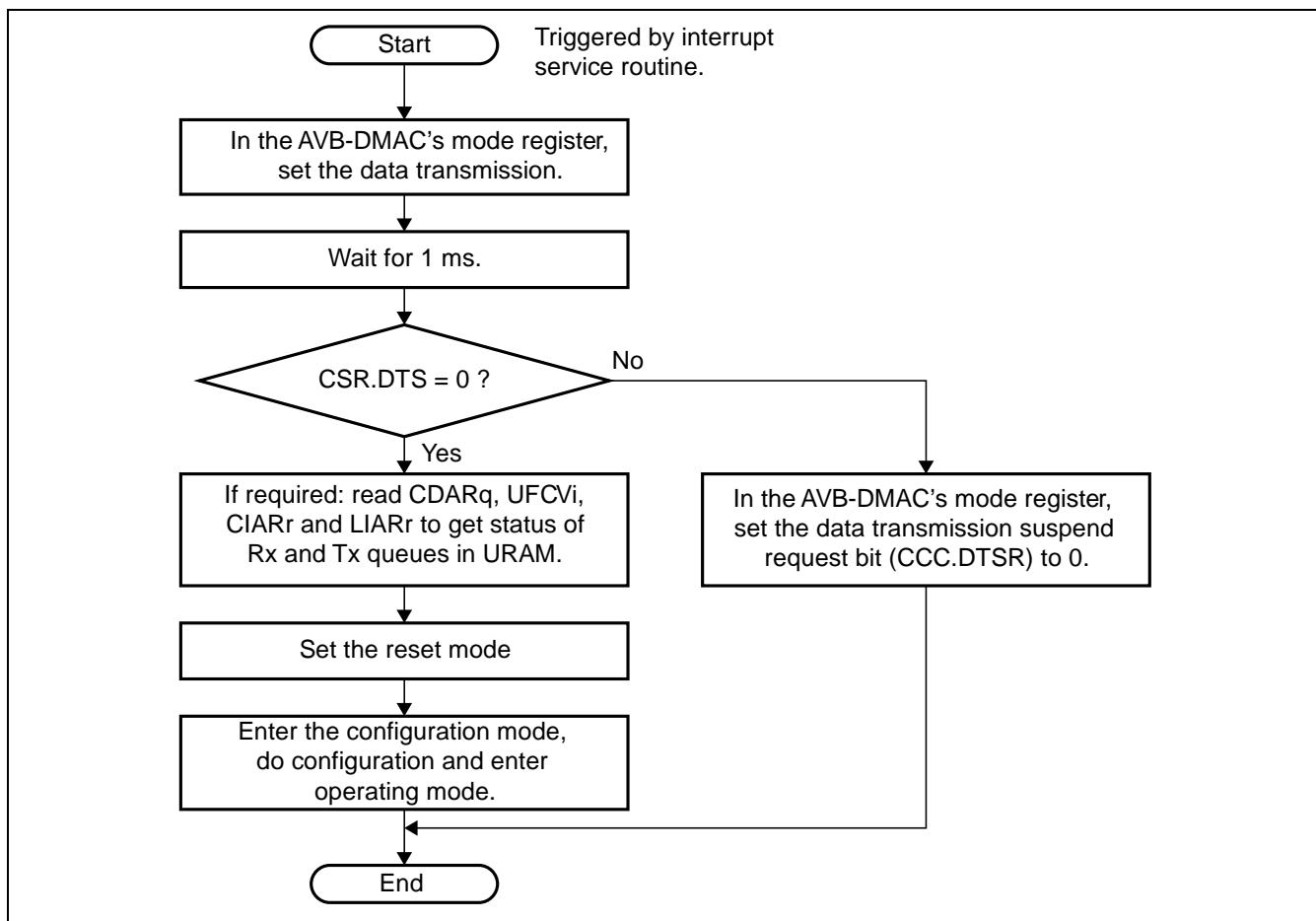


Figure 46.91 Data Transfer Function Stop Flow2 after Read Access Error

46.4.3 Power Down Request

If EthernetAVB-IF gets a power down request while writing to work RAM, it may happen that the request is not accepted even if the transfer is completed. Due to this the external power down module cannot complete the power down sequence.

A power down request is acceptable only when EthernetAVB-IF is in RESET state.

47. Controller Area Network Interface (CAN interface)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

47.1 Overview

47.1.1 Features

This MCU implements two channels (referred to as CAN0 and CAN1) of CAN (Controller Area Network) module that complies with the ISO11898-1 Specifications. The CAN module transmits and receives both formats of messages, namely the standard identifier (11 bits) (identifier hereafter referred to as ID) and extended ID (29 bits). Table 47.1 lists the CAN module overview and figure 47.1 shows the CAN module block diagram. Connect the CAN bus transceiver externally.

Table 47.1 CAN module overview

| Item | Overview |
|--------------------------------------|---|
| Protocol | <ul style="list-style-type: none"> ISO11898-1 compliant |
| Bit-rate | <ul style="list-style-type: none"> Up to 1 Mbps |
| Message box | <ul style="list-style-type: none"> 64 mailboxes: Two selectable mailbox mode <p>Normal mailbox mode: Of the 64 mailboxes, 32 can be configured for either transmission or reception (and the other 32 are reception-only).</p> <p>FIFO mailbox mode: 24 mailboxes configurable as transmission or reception (and the other 32 are reception-only). 4 stages FIFO for transmission and 4 stages FIFO for reception</p> |
| Reception | <ul style="list-style-type: none"> Data frame and remote frame can be received. Selectable receiving ID format (only standard ID, only extended ID or both ID) Programmable one-shot reception function Selectable overwrite mode (message overwritten) or overrun mode (message discarded) The reception complete interrupt can be enabled or disabled for each mailbox. |
| Acceptance filter | <ul style="list-style-type: none"> 8 acceptance masks (one mask every 4 mailboxes) 2 acceptance masks (one mask every 16 mailboxes) The mask can be enabled or disabled for each mailbox. |
| Transmission | <ul style="list-style-type: none"> Data frame and remote frame can be transmitted. Selectable transmitting ID format (only standard ID, only extended ID or both ID) Programmable one-shot transmission function (enable or disable) Selectable ID priority mode or mailbox number priority mode Transmission request can be aborted (The completion of abort can be confirmed with a flag.) The transmission complete interrupt can be enabled or disabled for each mailbox. |
| Mode transition for bus-off recovery | <ul style="list-style-type: none"> Mode transition for the recovery from the bus-off state can be selected: <ul style="list-style-type: none"> Automatic entry to CAN halt mode at bus-off entry Automatic entry to CAN halt mode at bus-off end Entry to CAN halt mode by a program Transition into error-active state by a program |

| Item | Overview |
|-------------------------|--|
| Error status monitoring | <ul style="list-style-type: none"> • CAN bus errors (stuff error, form error, ACK error, CRC error, bit error, and ACK delimiter error) can be monitored. • Transition to error states can be detected (error-warning, error-passive, bus-off entry, and bus-off recovery). • The error counters can be read. |
| Time stamp function | <ul style="list-style-type: none"> • Time stamp function using a 16-bit counter • The reference clock can be selected from either 1-, 2-, 4- or 8-bit time periods. |
| Interrupt sources | <ul style="list-style-type: none"> • 5 types: <ul style="list-style-type: none"> Reception complete Transmission complete Receive FIFO Transmit FIFO Error |
| CAN sleep mode | <ul style="list-style-type: none"> • Current consumption can be reduced by stopping the CAN clock. |
| Software support unit | <ul style="list-style-type: none"> • 3 software support units: <ul style="list-style-type: none"> Acceptance filter support Mailbox search support (receive mailbox search, transmit mailbox search and message lost search) Channel search support |
| CAN clock source (fCAN) | <ul style="list-style-type: none"> • Peripheral clocks (clkp1 or clkp2) or externally input clock is selectable. clkp1 = S3D4φ, clkp2 = CANFDφ |
| Test mode | <ul style="list-style-type: none"> • 3 test modes available for user evaluation: <ul style="list-style-type: none"> Listen-only mode Self-test mode 0 (external loop back) Self-test mode 1 (internal loop back) |

47.1.2 Block Diagram

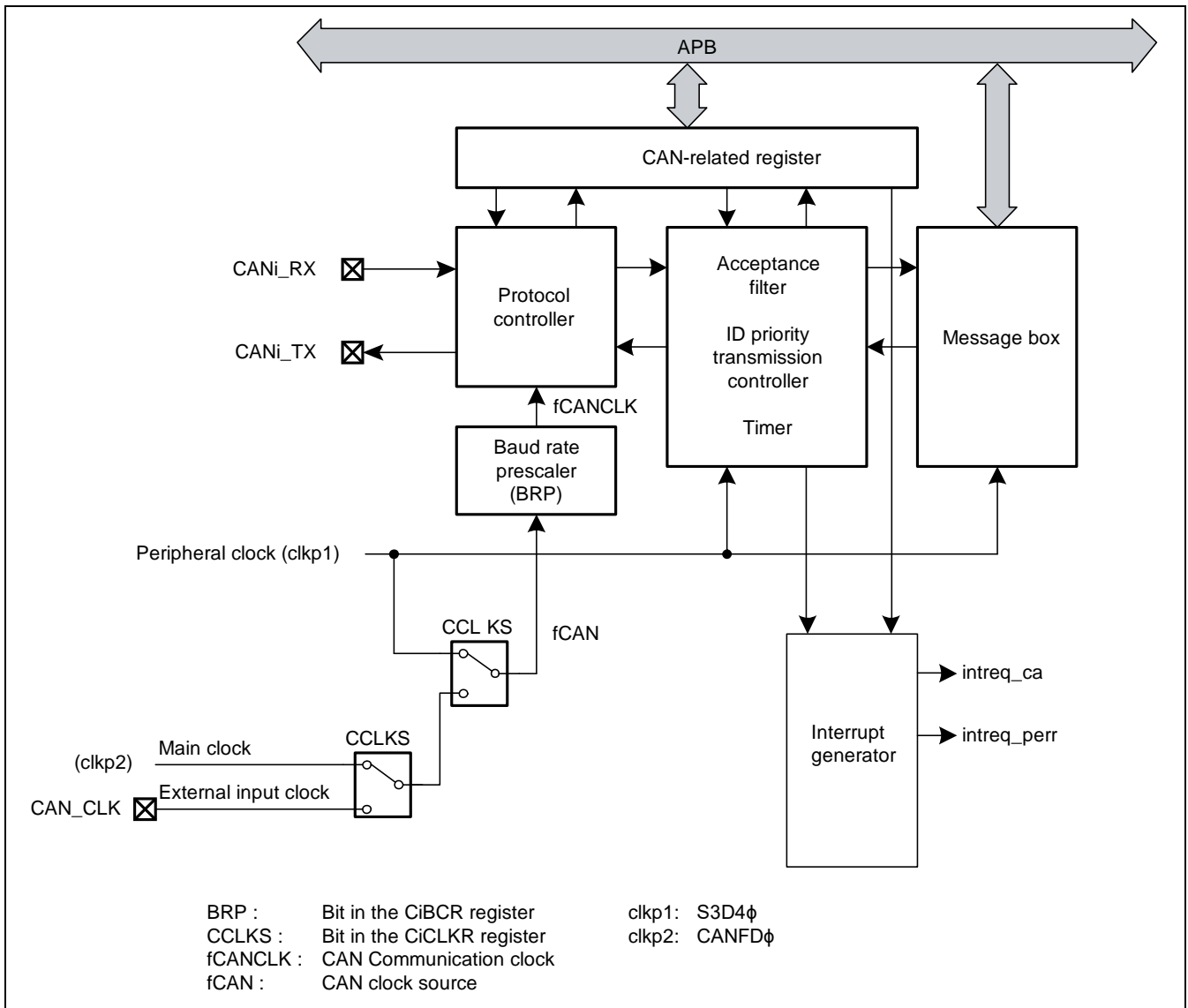


Figure 47.1 Block Diagram of CAN Module (i = 0, 1)

- CANi_RX/CANi_TX (i = 0 and 1):
CAN input/output pins
- Protocol controller:
Handles CAN protocol processing such as bus arbitration, bit timing at transmission and reception, stuffing, and error handling, etc.
- Message box:
Consists of 64 mailboxes which can be configured as either transmit or receive mailboxes. Each mailbox has an individual ID, data length code, a data field (8 bytes), and a time stamp.
- Acceptance filter:
Performs filtering of received messages. Registers CiMKR0 to CiMKR9 are used for the filtering process.
- Timer:
Used for the time stamp function. The timer value when storing a message into the mailbox is written as the time stamp value.
- Interrupt generator:
Generates the following five types of interrupts:
CANi reception complete interrupt
CANi transmission complete interrupt
CANi receive FIFO interrupt
CANi transmit FIFO interrupt
CANi error interrupt
- Support parity error detection as safety mechanism function

47.1.3 External Pins

Table 47.2 shows the CAN module pin.

Pin switching is required for pins which are multiplexed with other functions. For details, see section 6, Pin Function Controller (PFC).

Table 47.2 Pin Configuration

| Name | Abbreviation | I/O | Function | Second Generation RZ/G Series Products | | | |
|---------|--------------|--------|--|---|----------------------------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| CANi_RX | — | Input | Pins for receiving data | √ | √ | √ | √ |
| CANi_TX | — | Output | Pins for transmitting data | √ | √ | √ | √ |
| CAN_CLK | — | Input | Input pin used for external clock input. | √ | √ | √ | √ |

Legend: i = 0 and 1

47.1.4 Connected Module

Table 47.3 Connected Module

| Module name | Connected Module name | function |
|-------------|-----------------------|-------------------------------|
| CAN | APB | Register access of CPU |
| | CPG | Clock output |
| | PFC | Choice of an input-output pin |
| | Module Standby | Clock stop control |
| | Reset | Power on reset execution |
| | INTC-SYS, | Interrupt control |

47.1.5 Register Configuration

Table 47.4 Register Configurations

| Channel | Name | Abbreviation | R/W | Offset Address | Size | Second Generation RZ/G Series Products | | | |
|---------|--|---------------------|-----|----------------------------|-----------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 0 | CAN0 Control Register | C0CTLR | R/W | H'E6C3_0840 | 8, 16 | √ | √ | √ | √ |
| 0 | CAN0 Clock Select Register | C0CLKR | R/W | H'E6C3_0847 | 8 | √ | √ | √ | √ |
| 0 | CAN0 Bit Configuration Register | C0BCR | R/W | H'E6C3_0844 | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mask Register 0 | C0MKR0 | R/W | H'E6C3_0430 | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mask Register 1 | C0MKR1 | R/W | H'E6C3_0434 | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mask Register 2 | C0MKR2 | R/W | H'E6C3_0400 | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mask Register 3 | C0MKR3 | R/W | H'E6C3_0404 | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mask Register 4 | C0MKR4 | R/W | H'E6C3_0408 | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mask Register 5 | C0MKR5 | R/W | H'E6C3_040C | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mask Register 6 | C0MKR6 | R/W | H'E6C3_0410 | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mask Register 7 | C0MKR7 | R/W | H'E6C3_0414 | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mask Register 8 | C0MKR8 | R/W | H'E6C3_0418 | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mask Register 9 | C0MKR9 | R/W | H'E6C3_041C | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 FIFO Received ID Compare Register 0 | C0FIDCR0 | R/W | H'E6C3_0420 | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 FIFO Received ID Compare Register 1 | C0FIDCR1 | R/W | H'E6C3_0424 | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mask Invalid Register 0 | C0MKIVLR0 | R/W | H'E6C3_0438 | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mask Invalid Register 1 | C0MKIVLR1 | R/W | H'E6C3_0428 | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mailbox Register 0 to 63 | C0MB0 to C0MB63 | R/W | H'E6C3_0000 to H'E6C3_03FF | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mailbox Interrupt Enable Register 0 | C0MIER0 | R/W | H'E6C3_043C | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Mailbox Interrupt Enable Register 1 | C0MIER1 | R/W | H'E6C3_042C | 8, 16, 32 | √ | √ | √ | √ |
| 0 | CAN0 Message Control Register 0 to 63 | C0MCTL0 to C0MCTL63 | R/W | H'E6C3_0800 to H'E6C3_083F | 8 | √ | √ | √ | √ |

| Channel | Name | Abbreviation | R/W | Offset Address | Size | Second Generation RZ/G Series Products | | | |
|---------|---|--------------|-----|----------------|--------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 0 | CAN0 Receive FIFO Control Register | C0RFCR | R/W | H'E6C3_0848 | 8 | √ | √ | √ | √ |
| 0 | CAN0 Receive FIFO Pointer Control Register | C0RFPCR | R/W | H'E6C3_0849 | 8 | √ | √ | √ | √ |
| 0 | CAN0 Transmit FIFO Control Register | C0TFCR | R/W | H'E6C3_084A | 8 | √ | √ | √ | √ |
| 0 | CAN0 Transmit FIFO Pointer Control Register | C0TFPCR | R/W | H'E6C3_084B | 8 | √ | √ | √ | √ |
| 0 | CAN0 Status Register | C0STR | R | H'E6C3_0842 | 8, 16 | √ | √ | √ | √ |
| 0 | CAN0 Mailbox Search Mode Register | C0MSMR | R/W | H'E6C3_0853 | 8 | √ | √ | √ | √ |
| 0 | CAN0 Mailbox Search Status Register | C0MSSR | R | H'E6C3_0852 | 8 | √ | √ | √ | √ |
| 0 | CAN0 Channel Search Support Register | C0CSSR | R/W | H'E6C3_0851 | 8 | √ | √ | √ | √ |
| 0 | CAN0 Acceptance Filter Support Register | C0AFSR | R/W | H'E6C3_0856 | 8, 16 | √ | √ | √ | √ |
| 0 | CAN0 Error Interrupt Enable Register | C0EIER | R/W | H'E6C3_084C | 8 | √ | √ | √ | √ |
| 0 | CAN0 Error Interrupt Factor Judge Register | C0EIFR | R/W | H'E6C3_084D | 8 | √ | √ | √ | √ |
| 0 | CAN0 Receive Error Count Register | C0RECR | R | H'E6C3_084E | 8 | √ | √ | √ | √ |
| 0 | CAN0 Transmit Error Count Register | C0TECR | R | H'E6C3_084F | 8 | √ | √ | √ | √ |
| 0 | CAN0 Error Code Store Register | C0ECSR | R/W | H'E6C3_0850 | 8 | √ | √ | √ | √ |
| 0 | CAN0 Time Stamp Register | C0TSR | R | H'E6C3_0854 | 8, 16 | √ | √ | √ | √ |
| 0 | CAN0 Test Control Register | C0TCR | R/W | H'E6C3_0858 | 8 | √ | √ | √ | √ |
| 0 | CAN0 Interrupt Enable Register | C0IER | R/W | H'E6C3_0860 | 8 | √ | √ | √ | √ |
| 0 | CAN0 Interrupt Status Register | C0ISR | R/W | H'E6C3_0861 | 8 | √ | √ | √ | √ |
| 0 | CAN0 Mailbox Search Mask Register | C0MBSMR | R/W | H'E6C3_0863 | 8 | √ | √ | √ | √ |
| 0 | CAN0 Parity Error Control Register | C0PECR | R/W | H'E6C3_085C | 8 | √ | √ | √ | √ |
| 0 | CAN0 Parity Error Address Capture Register | C0PEACR | R | H'E6C3_085E | 16 | √ | √ | √ | √ |
| 1 | CAN1 Control Register | C1CTLR | R/W | H'E6C3_8840 | 8, 16 | √ | √ | √ | √ |
| 1 | CAN1 Clock Select Register | C1CLKR | R/W | H'E6C3_8847 | 8 | √ | √ | √ | √ |
| 1 | CAN1 Bit Configuration Register | C1BCR | R/W | H'E6C3_8844 | 8, 16, 32 | √ | √ | √ | √ |

| Channel | Name | Abbreviation | R/W | Offset Address | Size | Second Generation RZ/G Series Products | | | |
|---------|---|---------------------|-----|----------------------------|-----------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 1 | CAN1 Mask Register 0 | C1MKR0 | R/W | H'E6C3_8430 | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Mask Register 1 | C1MKR1 | R/W | H'E6C3_8434 | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Mask Register 2 | C1MKR2 | R/W | H'E6C3_8400 | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Mask Register 3 | C1MKR3 | R/W | H'E6C3_8404 | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Mask Register 4 | C1MKR4 | R/W | H'E6C3_8408 | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Mask Register 5 | C1MKR5 | R/W | H'E6C3_840C | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Mask Register 6 | C1MKR6 | R/W | H'E6C3_8410 | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Mask Register 7 | C1MKR7 | R/W | H'E6C3_8414 | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Mask Register 8 | C1MKR8 | R/W | H'E6C3_8418 | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Mask Register 9 | C1MKR9 | R/W | H'E6C3_841C | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 FIFO Received ID Compare Register 0 | C1FIDCR0 | R/W | H'E6C3_8420 | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 FIFO Received ID Compare Register 1 | C1FIDCR1 | R/W | H'E6C3_8424 | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Mask Invalid Register 0 | C1MKIVLR0 | R/W | H'E6C3_8438 | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Mask Invalid Register 1 | C1MKIVLR1 | R/W | H'E6C3_8428 | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Mailbox Register 0 to 63 | C1MB0 to C1MB63 | R/W | H'E6C3_8000 to H'E6C3_83FF | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Mailbox Interrupt Enable Register 0 | C1MIER0 | R/W | H'E6C3_843C | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Mailbox Interrupt Enable Register 1 | C1MIER1 | R/W | H'E6C3_842C | 8, 16, 32 | √ | √ | √ | √ |
| 1 | CAN1 Message Control Register 0 to 63 | C1MCTL0 to C1MCTL63 | R/W | H'E6C3_8800 to H'E6C3_883F | 8 | √ | √ | √ | √ |
| 1 | CAN1 Receive FIFO Control Register | C1RFCR | R/W | H'E6C3_8848 | 8 | √ | √ | √ | √ |
| 1 | CAN1 Receive FIFO Pointer Control Register | C1RFPCR | R/W | H'E6C3_8849 | 8 | √ | √ | √ | √ |
| 1 | CAN1 Transmit FIFO Control Register | C1TFCR | R/W | H'E6C3_884A | 8 | √ | √ | √ | √ |
| 1 | CAN1 Transmit FIFO Pointer Control Register | C1TFPCR | R/W | H'E6C3_884B | 8 | √ | √ | √ | √ |

| Channel | Name | Abbreviation | R/W | Offset Address | Size | Second Generation RZ/G Series Products | | | |
|---------|--|--------------|-----|----------------|-------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 1 | CAN1 Status Register | C1STR | R | H'E6C3_8842 | 8, 16 | √ | √ | √ | √ |
| 1 | CAN1 Mailbox Search Mode Register | C1MSMR | R/W | H'E6C3_8853 | 8 | √ | √ | √ | √ |
| 1 | CAN1 Mailbox Search Status Register | C1MSSR | R | H'E6C3_8852 | 8 | √ | √ | √ | √ |
| 1 | CAN1 Channel Search Support Register | C1CSSR | R/W | H'E6C3_8851 | 8 | √ | √ | √ | √ |
| 1 | CAN1 Acceptance Filter Support Register | C1AFSR | R/W | H'E6C3_8856 | 8, 16 | √ | √ | √ | √ |
| 1 | CAN1 Error Interrupt Enable Register | C1EIER | R/W | H'E6C3_884C | 8 | √ | √ | √ | √ |
| 1 | CAN1 Error Interrupt Factor Judge Register | C1EIFR | R/W | H'E6C3_884D | 8 | √ | √ | √ | √ |
| 1 | CAN1 Receive Error Count Register | C1RECR | R | H'E6C3_884E | 8 | √ | √ | √ | √ |
| 1 | CAN1 Transmit Error Count Register | C1TECR | R | H'E6C3_884F | 8 | √ | √ | √ | √ |
| 1 | CAN1 Error Code Store Register | C1ECSR | R/W | H'E6C3_8850 | 8 | √ | √ | √ | √ |
| 1 | CAN1 Time Stamp Register | C1TSR | R | H'E6C3_8854 | 8, 16 | √ | √ | √ | √ |
| 1 | CAN1 Test Control Register | C1TCR | R/W | H'E6C3_8858 | 8 | √ | √ | √ | √ |
| 1 | CAN1 Interrupt Enable Register | C1IER | R/W | H'E6C3_8860 | 8 | √ | √ | √ | √ |
| 1 | CAN1 Interrupt Status Register | C1ISR | R/W | H'E6C3_8861 | 8 | √ | √ | √ | √ |
| 1 | CAN1 Mailbox Search Mask Register | C1MBSMR | R/W | H'E6C3_8863 | 8 | √ | √ | √ | √ |
| 1 | CAN1 Parity Error Control Register | C1PECR | R/W | H'E6C3_885C | 8 | √ | √ | √ | √ |
| 1 | CAN1 Parity Error Address Capture Register | C1PEACR | R | H'E6C3_885E | 16 | √ | √ | √ | √ |

47.2 Register Description

Legend:

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R: Read-only. When writing to the register, read out the value of these bits and write it back without alteration.

W: Write-only. The read value is undefined.

47.2.1 CANi Control Register (CiCTLR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CAN0 Control Register (C0CTLR)

CAN1 Control Register (C1CTLR)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|------|-----|------|------|------|-----|------|-----|-----|------|-----|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | RBOC | BOM | SLPM | CANM | TSPS | | TSRC | TPM | MLM | IDFM | | MBM | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

<After Reset: H'0500>

| Bit | Symbol | Initial Value | R/W | Description |
|--------|--------|---------------|-----|--|
| 15, 14 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 13 | RBOC | B'0 | R/W | Forcible Return From Bus-OFF Bit*1 When the RBOC bit is set to "1" (forcible return from bus-off) in bus-off state, the CAN module forcibly returns from the bus-off state. This bit is automatically set to 0. The error state changes from bus-off to error-active. When the RBOC bit is set to "1", registers CiRECR and CiTECR are set to "H'00" and the BOST bit in the CiSTR register is set to "0" (the CAN module is not in bus-off state). The other registers remain unchanged. No bus-off recovery interrupt request is generated by this recovery from the bus-off state. Use the RBOC bit only when the BOM bit = "00" (normal mode). 0: Nothing occurred 1: Forcible return from bus-off*2 |

| Bit | Symbol | Initial Value | R/W | Description |
|--------|--------|---------------|-----|--|
| 12, 11 | BOM | All 0 | R/W | <p>Bus-Off Recovery Mode Bit*3</p> <p>The BOM bit is used to select bus-off recovery mode.</p> <p>When the BOM bit is "B'00", the recovery from bus-off is compliant with ISO11898-1, i.e. the CAN module re-enters CAN communication (error-active state) after detecting 11 consecutive recessive bits 128 times. A bus-off recovery interrupt request is generated when recovering from bus-off.</p> <p>When the BOM bit is "B'01", as soon as the CAN reaches the bus-off state, the CANM bit in the CiCTRL register is set to "B'10" (CAN halt mode) and the CAN enters CAN halt mode. No bus-off recovery interrupt request is generated when recovering from bus-off and registers CiTECR and CiRECR are set to "H'00".</p> <p>When the BOM bit is "B'10", the CANM bit is set to B'10 as soon as the CAN module reaches the bus-off state. The CAN module enters CAN halt mode after the recovery from the bus-off state, i.e. after detecting 11 consecutive recessive bits 128 times. A bus-off recovery interrupt request is generated when recovering from bus-off and registers CiTECR and CiRECR are set to "H'00".</p> <p>When the BOM bit is "B'11", the CAN module enters CAN halt mode by setting the CANM bit to "B'10" while the CAN module is still in bus-off state. No bus-off recovery interrupt request is generated when recovering from bus-off and registers CiTECR and CiRECR are set to "H'00". However, if the CAN module recovers from bus-off after detecting 11 consecutive recessive bits 128 times before the CANM bit is set to "B'10", a bus-off recovery interrupt request is generated.</p> <p>If the CPU requests an entry to CAN reset mode at the same time as the CAN module attempts to enter CAN halt mode (at bus-off entry when the BOM bit is "B'01", or at bus-off end when the BOM bit is "B'10"), then the CPU request to enter CAN reset mode has higher priority.</p> <p>B'00: Normal mode (ISO11898-1 compliant) B'01: Entry to CAN halt mode automatically at bus-off entry B'10: Entry to CAN halt mode automatically at bus-off end B'11: Entry to CAN halt mode (during bus-off recovery period) by a program request</p> |
| 10 | SLPM | B'1 | R/W | <p>CAN Sleep Mode Bit*4*5</p> <p>When the SLPM bit is set to "1", the CAN module enters CAN sleep mode.</p> <p>When the SLPM bit is set to "0", the CAN module exits CAN sleep mode. Refer to section 47.3, Operating Mode for detail.</p> <p>0: Other than CAN sleep mode 1: CAN sleep mode</p> |
| 9, 8 | CANM | B'01 | R/W | <p>CAN Operating Mode Select Bit*4</p> <p>The CANM bit selects one of the following modes for the CAN module: CAN operation mode, CAN reset mode or CAN halt mode. Refer to section 47.3, Operating Mode for detail. CAN sleep mode is set by the SLPM bit.</p> <p>When the CAN module enters CAN halt mode according to the setting of the BOM bit, the CANM bit is automatically set to "10".</p> <p>B'00: CAN operation mode B'01: CAN reset mode B'10: CAN halt mode B'11: CAN reset mode (forcible transition)</p> |

| Bit | Symbol | Initial Value | R/W | Description |
|------|--------|---------------|-----|--|
| 7, 6 | TSPS | All 0 | R/W | <p>Time Stamp Prescaler Select Bit*³</p> <p>The TSPS bit selects the prescaler for the time stamp. The reference clock for the time stamp can be selected to be either 1-, 2-, 4- or 8-bit time periods.</p> <p>B'00: Every bit time B'01: Every 2-bit time B'10: Every 4-bit time B'11: Every 8-bit time</p> |
| 5 | TSRC | B'0 | R/W | <p>Time Stamp Counter Reset Command Bit*⁶</p> <p>The TSRC bit is used to reset the time stamp counter. When the TSRC bit is set to "1", the CiTSR register is set to H'0000. It is automatically set to 0.</p> <p>0: Nothing occurred 1: Reset*²</p> |
| 4 | TPM | B'0 | R/W | <p>Transmission Priority Mode Select Bit*³</p> <p>The TPM bit specifies the priority of modes when transmitting messages. ID priority transmit mode or mailbox number transmit mode can be selected.</p> <p>All mailboxes are set for either ID priority transmission or mailbox number priority transmission.</p> <p>When the TPM bit is 0, ID priority transmit mode is selected and transmission priority complies with the CAN bus arbitration rule, as defined in the ISO 11898-1 Specifications. In ID priority transmit mode, mailboxes [0] to [63] (in normal mailbox mode), and mailboxes [0] to [55] (in FIFO mailbox mode), and the transmit FIFO are compared for the IDs of mailboxes configured for transmission. If two or more mailbox IDs are the same, the mailbox with the smaller number has higher priority.</p> <p>Only the next message to be transmitted from the transmit FIFO is included in the transmission arbitration. If a transmit FIFO message is being transmitted, the next pending message within the transmit FIFO is included in the transmission arbitration.</p> <p>When the TPM bit is 1, mailbox number transmit mode is selected and the transmit mailbox with the smallest mailbox number has the highest priority. In FIFO mailbox mode, the transmit FIFO has lower priority than normal mailboxes (mailboxes [0] to [55]).</p> <p>0: ID priority transmit mode 1: Mailbox number priority transmit mode</p> |
| 3 | MLM | B'0 | R/W | <p>Message Lost Mode Select Bit*³</p> <p>The MLM bit specifies the operation when a new message is captured in the unread mailbox.</p> <p>Overwrite mode or overrun mode can be selected. All mailboxes (including the receive FIFO) are set to either overwrite mode or overrun mode.</p> <p>When the MLM bit is "0", all mailboxes are set to overwrite mode and the new message is overwriting the old message.</p> <p>When this bit is "1", all mailboxes are set to overrun mode and the new message is discarded.</p> <p>0: Overwrite mode 1: Overrun mode</p> |

| Bit | Symbol | Initial Value | R/W | Description |
|------|--------|---------------|-----|--|
| 2, 1 | IDFM | All 0 | R/W | <p>ID Format Mode Select Bit*³</p> <p>The IDFM bit specifies the ID format.</p> <p>B'00: Standard ID mode All mailboxes (including FIFO mailboxes) handle only standard IDs.</p> <p>B'01: Extended ID mode All mailboxes (including FIFO mailboxes) handle only extended IDs.</p> <p>B'10: Mixed ID mode All mailboxes (including FIFO mailboxes) handle both standard IDs and extended IDs. Standard IDs or extended IDs are specified by using the IDE bit in the corresponding mailbox in normal mail box mode. In FIFO mailbox mode, the IDE bit in the corresponding mailbox is used for mailboxes [0] to [55], the IDE bit in registers CiFIDCR0 and CiFIDCR1 is used for the receive FIFO, and the IDE bit in mailbox [56] is used for the transmit FIFO.</p> <p>B'11: Do not use this combination</p> |
| 0 | MBM | B'0 | R/W | <p>CAN Mailbox Mode Select Bit*³</p> <p>When the MBM bit is "0" (normal mailbox mode), mailboxes [0] to [63] are configured as transmit or receive mailboxes. When the MBM bit is "1" (FIFO mailbox mode), mailboxes [0] to [55] are configured as transmit or receive mailboxes. Mailboxes [56] to [59] are configured as a transmit FIFO and mailboxes [60] to [63] as a receive FIFO.</p> <p>Transmit data is written into mailbox [56] (mailbox [56] is a window mailbox for the transmit FIFO).</p> <p>Receive data is read from mailbox [60] (mailbox [60] is a window mailbox for the receive FIFO).</p> <p>Table 47.5 lists the Mailbox Configuration.</p> <p>0: Normal mailbox mode 1: FIFO mailbox mode</p> |

- Notes:
1. Set the RBOC bit to "1" in bus-off state.
 2. Bits RBOC and TSRC are automatically set back to "0" after being set to "1". It should be read as "0".
 3. Write to bits BOM, MBM, IDFM, MLM,, TPM, and TSPS in CAN reset mode.
 4. When bits CANM and SLPM are changed, check the CiSTR register to ensure that the mode has been switched.
 5. Write to the SLPM bit in CAN reset mode or CAN halt mode. When rewriting the SLPM bit, set only this bit to "0" or "1".
 6. Set the TSRC bit to "1" in CAN operation mode.

Table 47.5 Mailbox Configuration

| Mailbox | MBM Bit = "0" (Normal mailbox mode) | MBM Bit = "1" (FIFO mailbox mode) |
|------------------------|-------------------------------------|-----------------------------------|
| Mailboxes [0] to [55] | Normal mailbox | Normal mailbox |
| Mailboxes [56] to [59] | | Transmit FIFO |
| Mailboxes [60] to [63] | | Receive FIFO |

Notes: Points 1 to 5 below should be considered when the MBM bit is set to "1".

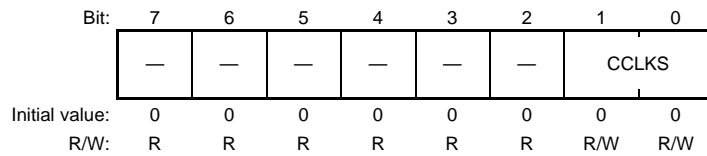
1. Transmit FIFO is controlled by the CiTFCR register.
The CiMCTLj register of mailboxes [56] to [59] are disabled.
Registers CiMCTL56 to CiMCTL59 cannot be used.
2. Receive FIFO is controlled by the CiRFCR register.
The CiMCTLj register of mailboxes [60] to [63] are disabled.
Registers CiMCTL60 to CiMCTL63 cannot be used.
3. Refer to the CiMIER1 register about the FIFO interrupts.
4. The corresponding bits in the CiMKIVLR register for mailboxes [56] to [63] are disabled. Set 0 to these bits.
5. Transmit/receive FIFOs can be used for both data frames and remote frames.

47.2.2 CANi Clock Select Register (CiCLKR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CAN0 Clock Select Register (C0CLKR)

CAN1 Clock Select Register (C1CLKR)



<After Reset: H'00>

| Bit | Symbol | Initial Value | R/W | Description |
|--------|--------|---------------|-----|--|
| 7 to 2 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 1, 0 | CCLKS | B'00 | R/W | CAN Clock Source Select Bits* The CCLKS bits select clock source from among the peripheral clock (clkp1 or clkp2) generated by the PLL frequency synthesizer, and an externally input clock. B'00: Peripheral clock (clkp1) B'01: Peripheral clock (clkp2) B'10: Illegal value B'11: Externally input clock |

Note: * Write to the CCLKS bit in CAN reset mode.

47.2.3 CANi Bit Configuration Register (CiBCR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

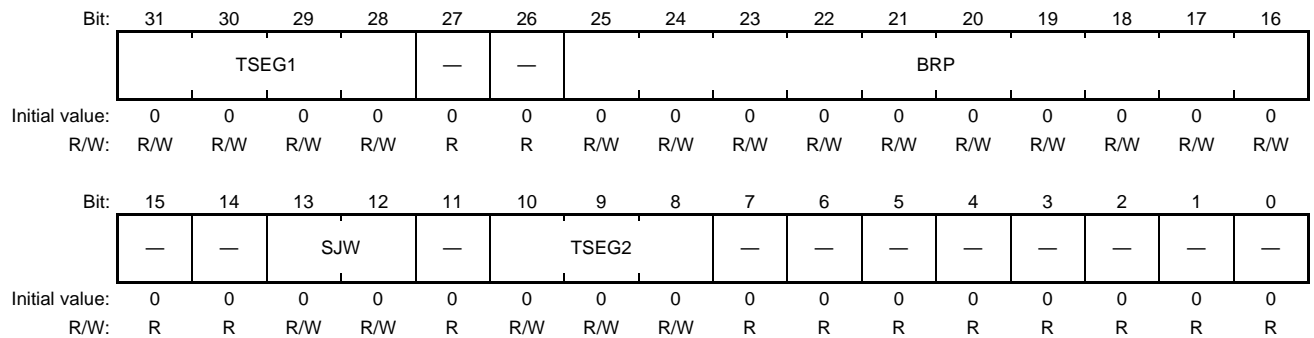
For the bit timing configuration rule, refer to section 47.4, CAN Communication Speed Configuration.

Set the CiBCR register before entering CAN halt mode from CAN reset mode or CAN operation mode from CAN reset mode. After the setting is made once, this register can be written to in CAN reset mode or CAN halt mode.

The CiBCR register consists of 24 bits. A 32-bit read/write access should be performed carefully not to rewrite the CiCLKR register.

CAN0 Bit Configuration Register (C0BCR)

CAN1 Bit Configuration Register (C1BCR)



<After Reset: H'0000_0000>

| Bit | Symbol | Initial Value | R/W | Description |
|----------|--------|---------------|-----|--|
| 31 to 28 | TSEG1 | All 0 | R/W | <p>Time Segment 1 Control Bits</p> <p>The TSEG1 bit is used to specify the total length of the propagation time segment (PROP_SEG) and phase buffer segment 1 (PHASE_SEG1) with the value of Tq.</p> <p>A value from 4 to 16 time quanta can be set.</p> <p>B'0000: Do not use this combination</p> <p>B'0001: Do not use this combination</p> <p>B'0010: Do not use this combination</p> <p>B'0011: 4 Tq</p> <p>B'0100: 5 Tq</p> <p>B'0101: 6 Tq</p> <p>B'0110: 7 Tq</p> <p>B'0111: 8 Tq</p> <p>B'1000: 9 Tq</p> <p>B'1001: 10 Tq</p> <p>B'1010: 11 Tq</p> <p>B'1011: 12 Tq</p> <p>B'1100: 13 Tq</p> <p>B'1101: 14 Tq</p> <p>B'1110: 15 Tq</p> <p>B'1111: 16 Tq</p> |

| Bit | Symbol | Initial Value | R/W | Description |
|----------|--------|---------------|-----|--|
| 27, 26 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 25 to 16 | BRP | All 0 | R/W | Prescaler Division Ratio Set Bits The BRP bit is used to set the peripheral bus clock periods contained in a Time Quantum. If the setting value is P (0 to 1023), the baud rate prescaler divides fCAN by P + 1. |
| 15, 14 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 13, 12 | SJW | All 0 | R/W | Resynchronization Jump Width Control Bits The SJW bit is used to specify the resynchronization jump width with the value of Tq. A value from 1 to 4 time quanta can be set. Set the value smaller than or equal to that of the TSEG2 bit. B'00: 1 Tq B'01: 2 Tq B'10: 3 Tq B'11: 4 Tq |
| 11 | — | B'0 | R | Reserved bit This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit. |
| 10 to 8 | TSEG2 | All 0 | R/W | Time Segment 2 Control Bits The TSEG2 bit is used to specify the length of phase buffer segment 2 (PHASE_SEG2) with the value of Tq. A value from 2 to 8 time quanta can be set. Set the value smaller than that of the TSEG1 bit. B'000: Do not use this combination B'001: 2 Tq B'010: 3 Tq B'011: 4 Tq B'100: 5 Tq B'101: 6 Tq B'110: 7 Tq B'111: 8 Tq |
| 7 to 0 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |

47.2.4 CANi Mask Register k (CiMKRk) (i = 0, 1; k = 0 to 9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

For masking the function in FIFO mailbox mode, refer to section 47.6, Acceptance Filtering and Masking Function.

Write to registers CiMKR0 to CiMKR9 in CAN reset mode or CAN halt mode.

CAN0 Mask Register 0 (C0MKR0)

CAN0 Mask Register 1 (C0MKR1)

CAN0 Mask Register 2 (C0MKR2)

:

CAN0 Mask Register 9 (C0MKR9)

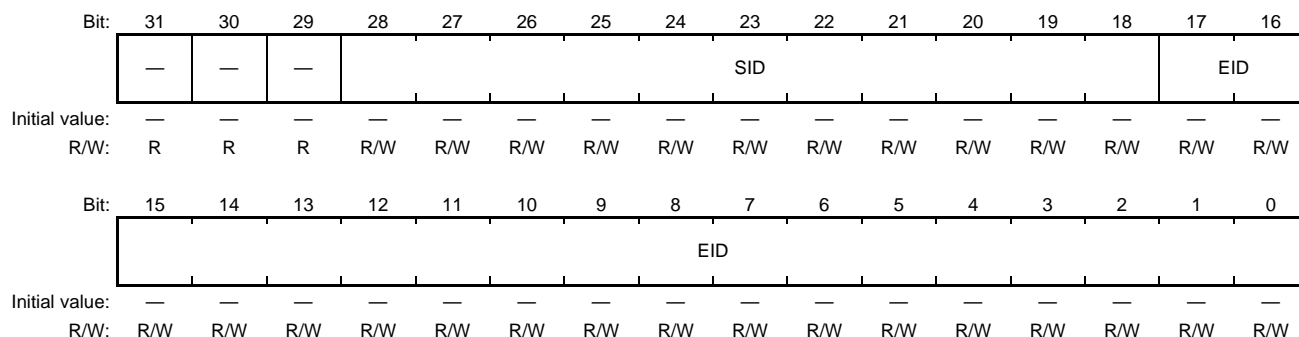
CAN1 Mask Register 0 (C1MKR0)

CAN1 Mask Register 1 (C1MKR1)

CAN1 Mask Register 2 (C1MKR2)

:

CAN1 Mask Register 9 (C1MKR9)



<After Reset: Undefined>

| Bit | Symbol | Initial Value | R/W | Description |
|----------|--------|---------------|-----|---|
| 31 to 29 | — | Undefined | R | Reserved bits The reset value is undefined. The write value should be "0". These bits are read as "0" after "0" is written to. |

| Bit | Symbol | Initial Value | R/W | Description |
|----------|--------|---------------|-----|--|
| 28 to 18 | SID | Undefined | R/W | <p>Standard ID Bits</p> <p>The SID bit is the filter mask bit corresponding to the CAN standard ID bit. The SID bit is used to receive both standard ID and extended ID messages.</p> <p>When the SID bit is set to "0", the corresponding SID bit is not compared for the received ID and the mailbox ID.</p> <p>When the SID bit is set to "1", corresponding SID bit compares received ID with mailbox ID.</p> <p>0: Corresponding SID bit is not compared 1: Corresponding SID bit is compared</p> |
| 17 to 0 | EID | Undefined | R/W | <p>Extended ID Bits</p> <p>The EID bit is the filter mask bit for CAN extended ID bit.</p> <p>This bit is used to receive extended ID messages.</p> <p>When the EID bit is set to "0", corresponding EID bit does not compare received ID with mailbox ID.</p> <p>When the EID bit is set to "1", corresponding EID bit compares received ID with mailbox ID.</p> <p>0: Corresponding EID bit is not compared 1: Corresponding EID bit is compared</p> |

47.2.5 CANi FIFO Received ID Compare Registers (CiFIDCR0 and CiFIDCR1) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Registers CiFIDCR0 and CiFIDCR1 are enabled when the MBM bit in the CiCTLR register is set to "1" (FIFO mailbox mode). Bits EID, SID, RTR, and IDE in registers CiMB60 to CiMB63 are disabled.

For the usage of these registers, refer to section 47.6, Acceptance Filtering and Masking Function.

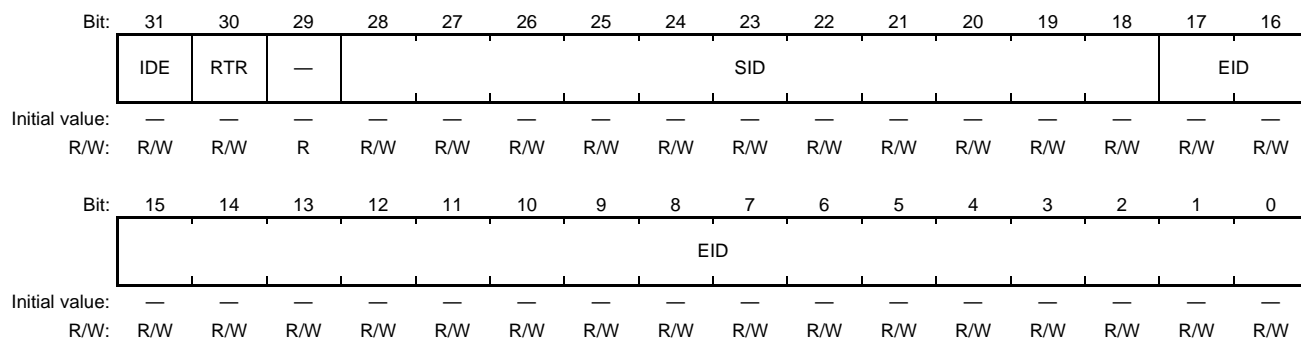
Write to registers CiFIDCR0 and CiFIDCR1 in CAN reset mode or CAN halt mode.

CAN0 FIFO Received ID Compare Register 0 (C0FIDCR0)

CAN0 FIFO Received ID Compare Register 1 (C0FIDCR1)

CAN1 FIFO Received ID Compare Register 0 (C1FIDCR0)

CAN1 FIFO Received ID Compare Register 1 (C1FIDCR1)



<After Reset: Undefined>

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|---|
| 31 | IDE | Undefined | R/W | <p>ID Extension Bit*</p> <p>The IDE bit sets the ID format to standard ID or extended ID. The IDE bit is enabled when the IDFM bit in the CiCTLR register is "10" (mixed ID mode). When the IDFM bit is "10", the IDE bit specifies the following operation.</p> <p>When both IDE bits in registers CiFIDCR0 and CiFIDCR1 are set to "0", only standard ID frames can be received.</p> <p>When both IDE bits in registers CiFIDCR0 and CiFIDCR1 are set to "1", only extended ID frames can be received.</p> <p>When the IDE bits in registers CiFIDCR0 and CiFIDCR1 are set to "0" or "1" individually, both standard ID and extended ID frames can be received.</p> <p>0: Standard ID 1: Extended ID</p> |

| Bit | Symbol | Initial Value | R/W | Description |
|----------|--------|---------------|-----|--|
| 30 | RTR | Undefined | R/W | <p>Remote Transmission Request Bit</p> <p>The RTR bit sets the specified frames format of data frame or remote frames. The RTR bit specifies the following operation.</p> <p>When both RTR bits in registers CiFIDCR0 and CiFIDCR1 are set to "0", only data frames can be received.</p> <p>When both RTR bits in registers CiFIDCR0 and CiFIDCR1 are set to "1", only remote frames can be received.</p> <p>When the RTR bits in registers CiFIDCR0 and CiFIDCR1 are set to "0" or "1" individually, both data frames and remote frames can be received.</p> <p>0: Data frame 1: Remote frame</p> |
| 29 | — | Undefined | R | <p>Reserved bit</p> <p>The reset value is undefined. The write value should be "0". This bit is read as "0" after "0" is written to.</p> |
| 28 to 18 | SID | Undefined | R/W | <p>Standard ID Bits</p> <p>The SID bit sets the standard ID of data frames and remote frames. The SID bit is used to receive both standard ID and extended ID messages.</p> <p>0: Corresponding SID bit is "0" 1: Corresponding SID bit is "1"</p> |
| 17 to 0 | EID | Undefined | R/W | <p>Extended ID Bits</p> <p>The EID bit sets the extended ID of data frames and remote frames. The EID bit is used to receive extended ID messages.</p> <p>0: Corresponding EID bit is "0" 1: Corresponding EID bit is "1"</p> |

Note: * When the IDFM bit is not "10", the IDE bit should be written with "0".

47.2.6 CANi Mask Invalid Registers (CiMKIVLR0 and CiMKIVLR1) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Each bit in registers CiMKIVLR0 and CiMKIVLR1 corresponds to a mailbox. The correspondence between the bits and mailboxes is as follows:

- Bit 0 in the CiMKIVLR0 register corresponds to mailbox 0 (MB0).
- Bit 31 in the CiMKIVLR0 register corresponds to mailbox 31 (MB31).
- Bit 0 in the CiMKIVLR1 register corresponds to mailbox 32 (MB32).
- Bit 31 in the CiMKIVLR1 register corresponds to mailbox 63 (MB63).

When each bit is "1", the acceptance mask for the mailbox corresponding to the bit number is disabled. In this case, a receiving message is stored into mailbox only its ID matches bits SID and EID in the CiMBj register.

Write to registers CiMKIVLR0 and CiMKIVLR1 either in CAN reset mode or CAN halt mode.

CAN0 Mask Invalid Register 0 (C0MKIVLR0)

CAN1 Mask Invalid Register 0 (C1MKIVLR0)

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MB31 | MB30 | MB29 | MB28 | MB27 | MB26 | MB25 | MB24 | MB23 | MB22 | MB21 | MB20 | MB19 | MB18 | MB17 | MB16 |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MB15 | MB14 | MB13 | MB12 | MB11 | MB10 | MB9 | MB8 | MB7 | MB6 | MB5 | MB4 | MB3 | MB2 | MB1 | MB0 |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

<After Reset: Undefined>

| Bit | Symbol | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 0 | MB31 to 0 | Undefined | R/W | Bit 31 corresponds to mailbox 31 (MB31), and bit 0 corresponds to mailbox 0 (MB0). 0: Mask valid 1: Mask invalid |

CAN0 Mask Invalid Register 1 (COMKIVLR1)

CAN1 Mask Invalid Register 1 (C1MKIVLR1)

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MB63 | MB62 | MB61 | MB60 | MB59 | MB58 | MB57 | MB56 | MB55 | MB54 | MB53 | MB52 | MB51 | MB50 | MB49 | MB48 |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MB47 | MB46 | MB45 | MB44 | MB43 | MB42 | MB41 | MB40 | MB39 | MB38 | MB37 | MB36 | MB35 | MB34 | MB33 | MB32 |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

<After Reset: Undefined>

| Bit | Symbol | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 0 | MB63 to 32 | Undefined | R/W | Bit 31 corresponds to mailbox 63 (MB63), and bit 0 corresponds to mailbox 32 (MB32). 0: Mask valid 1: Mask invalid |

47.2.7 CANi Mailbox Register j (CiMBj) (i = 0, 1; j = 0 to 63)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Table 47.6 lists the CANi mailbox memory mapping and Table 47.7 lists the CAN data frame construction.

The value after reset of CANi Mailbox is undefined.

Write to the CiMBj register only when the associated CiMCTLj register is "H'00" and the corresponding mailbox is not processing an abort request.

Refer to Table 47.6 for detailed addresses.

Table 47.6 CANi Mailbox Memory Mapping (i = 0, 1)

| Address | | Message Content |
|---------------------------|---------------------------|----------------------------|
| CAN0 | CAN1 | Memory Mapping |
| H'E6C3_0000 + 16 × j + 0 | H'E6C3_8000 + 16 × j + 0 | IDE, RTR, SID10 to SID6 |
| H'E6C3_0000 + 16 × j + 1 | H'E6C3_8000 + 16 × j + 1 | SID5 to SID0, EID17, EID16 |
| H'E6C3_0000 + 16 × j + 2 | H'E6C3_8000 + 16 × j + 2 | EID15 to EID8 |
| H'E6C3_0000 + 16 × j + 3 | H'E6C3_8000 + 16 × j + 3 | EID7 to EID0 |
| H'E6C3_0000 + 16 × j + 4 | H'E6C3_8000 + 16 × j + 4 | — |
| H'E6C3_0000 + 16 × j + 5 | H'E6C3_8000 + 16 × j + 5 | Data length code (DLC) |
| H'E6C3_0000 + 16 × j + 6 | H'E6C3_8000 + 16 × j + 6 | Data byte 0 |
| H'E6C3_0000 + 16 × j + 7 | H'E6C3_8000 + 16 × j + 7 | Data byte 1 |
| : | : | : |
| H'E6C3_0000 + 16 × j + 13 | H'E6C3_8000 + 16 × j + 13 | Data byte 7 |
| H'E6C3_0000 + 16 × j + 14 | H'E6C3_8000 + 16 × j + 14 | Time stamp upper byte |
| H'E6C3_0000 + 16 × j + 15 | H'E6C3_8000 + 16 × j + 15 | Time stamp lower byte |

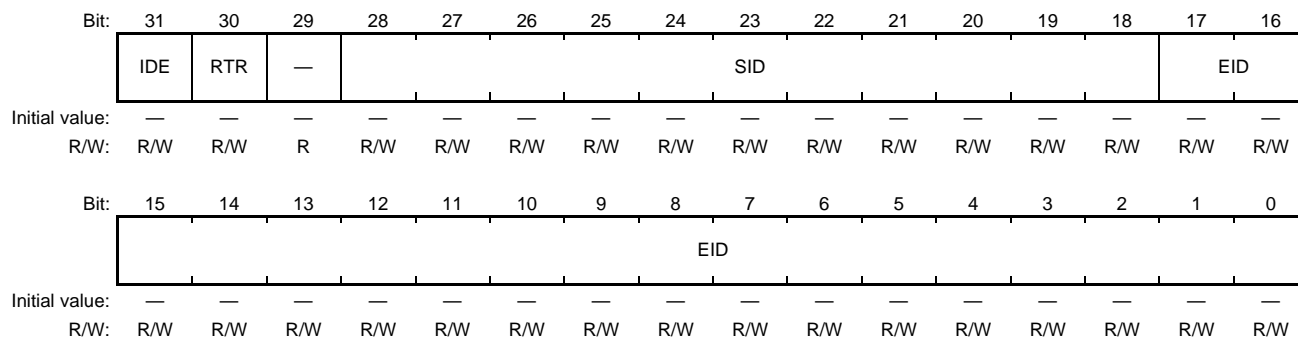
Table 47.7 CAN Data Frame Construction

| | | | | | | | | | |
|------------------|-----------------|-------------------|------------------|-----------------|-----------------|-------|-------|-----|-------|
| SID10 to SID6 | SID5 to SID0 | EID17 to EID16 | EID15 to EID8 | EID7 to EID0 | DLC3 to DLC0 | DATA0 | DATA1 | ... | DATA7 |
|------------------|-----------------|-------------------|------------------|-----------------|-----------------|-------|-------|-----|-------|

The previous value of each mailbox is retained unless a new message is received.

CAN0 Mailbox Register 0 to 63 (COMB0 to 63)

CAN1 Mailbox Register 0 to 63 (C1MB0 to 63)



<After Reset: Undefined>

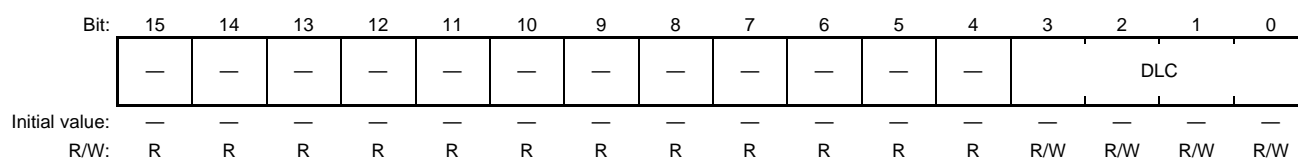
| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|--|
| 31 | IDE | Undefined | R/W | <p>ID Extension Bit*1</p> <p>The IDE bit sets the ID format of standard IDs or extended IDs.</p> <p>The IDE bit is enabled when the IDFM bit in the CiCTLR register is "10" (mixed ID mode).</p> <p>When the IDFM bit is "10", the IDE bit specifies the following operation.</p> <p>Receive mailbox receives only ID format specified by the IDE bit.</p> <p>Transmit mailbox transmits with ID format specified by the IDE bit.</p> <p>Receive FIFO mailbox receives messages with the standard ID, extended ID, or both IDs specified by the IDE bit in registers CiFIDCR0 and CiFIDCR1.</p> <p>Transmit FIFO mailbox transmits messages with the standard ID or extended ID specified by the IDE bit in the relevant transmitting message.</p> <p>0: Standard ID 1: Extended ID</p> |
| 30 | RTR | Undefined | R/W | <p>Remote Frame Request Bit</p> <p>The RTR bit sets the frame format of data frames or remote frames.</p> <p>This bit specifies the following operation:</p> <p>Receive mailbox receives only frames with the format specified by the RTR bit.</p> <p>Transmit mailbox transmits according to the frame format specified by the RTR bit.</p> <p>Receive FIFO mailbox receives the data frame, remote frame, or both frames specified by the RTR bit in registers CiFIDCR0 and CiFIDCR1.</p> <p>Transmit FIFO mailbox transmits the data frame or remote frame specified by the RTR bit in the relevant transmitting message.</p> <p>0: Data frame 1: Remote frame</p> |
| 29 | — | Undefined | R | <p>Reserved bit</p> <p>The reset value is undefined. The write value should be "0". This bit is read as "0" after "0" is written to.</p> |

| Bit | Symbol | Initial Value | R/W | Description |
|----------|--------|---------------|-----|---|
| 28 to 18 | SID | Undefined | R/W | Standard ID Bits The SID bit sets the standard ID of data frames and remote frames. The SID bit is used to transmit or receive both standard ID and extended ID messages. 0: Corresponding SID bit is "0" 1: Corresponding SID bit is "1" |
| 17 to 0 | EID | Undefined | R/W | Extended ID Bits*2 The EID bit sets the extended ID of data frames and remote frames. The EID bit is used to transmit or receive extended ID messages. 0: Corresponding EID bit is "0" 1: Corresponding EID bit is "1" |

Notes: 1. When the IDFM bit is not "10", it should be written with "0".
2. If the mailbox has received a standard ID message, the EID bit in the mailbox is undefined.

CAN0 Mailbox Register 0 to 63 (COMB0 to 63)

CAN1 Mailbox Register 0 to 63 (C1MB0 to 63)



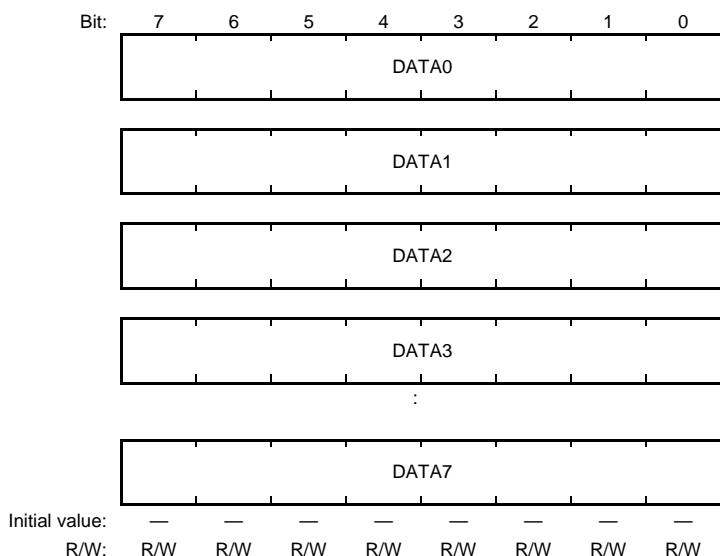
<After Reset: Undefined>

| Bit | Symbol | Initial Value | R/W | Description |
|---------|--------|---------------|-----|--|
| 15 to 4 | — | Undefined | R | Reserved bits The reset value is undefined. The write value should be "0". These bits are read as "0" after "0" is written to. |
| 3 to 0 | DLC | Undefined | R/W | Data Length Code Bits* The DLC is used to set the number of data bytes to be transmitted in a data frame. When data is requested using a remote frame, the number of data bytes to be requested is set. When a data frame is received, the number of received data bytes is stored. When a remote frame is received, the number of requested data bytes is stored. B'0000: Data length = 0 byte B'0001: Data length = 1 byte B'0010: Data length = 2 bytes B'0011: Data length = 3 bytes B'0100: Data length = 4 bytes B'0101: Data length = 5 bytes B'0110: Data length = 6 bytes B'0111: Data length = 7 bytes B'1xxx: Data length = 8 bytes Legend: x represents any value. |

Note: * If the mailbox has received a message with n bytes less than 8 bytes, the values of DATAn to DATA7 in the mailbox are undefined.

CAN0 Mailbox Register 0 to 63 (C0MB0 to 63)

CAN1 Mailbox Register 0 to 63 (C1MB0 to 63)



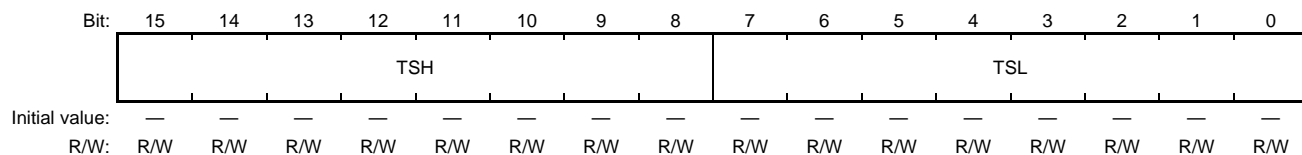
<After Reset: Undefined>

| Bit | Symbol | Initial Value | R/W | Description |
|--------|------------|---------------|-----|--|
| 7 to 0 | DATA0 to 7 | Undefined | R/W | Data Bytes 0 to 7*1*2 DATA0 to DATA7 store the transmitted or received CAN message data. Transmission or reception starts from DATA0. The bit order on the CAN bus is MSB first, and transmission or reception starts from bit 7. |

- Notes:
1. If the mailbox has received a message with n bytes less than 8 bytes, the values of DATAn to DATA7 in the mailbox are undefined.
 2. If the mailbox has received a remote frame, the previous values of DATA0 to DATA7 in the mailbox are retained.

CAN0 Mailbox Register 0 to 63 (C0MB0 to 63)

CAN1 Mailbox Register 0 to 63 (C1MB0 to 63)



<After Reset: Undefined>

| Bit | Symbol | Initial Value | R/W | Description |
|---------|--------|---------------|-----|---|
| 15 to 8 | TSH | Undefined | R/W | Time Stamp Higher Byte TSL and TSH store the counter value of the time stamp when received messages are stored in the mailbox. |
| 7 to 0 | TSL | Undefined | R/W | Time Stamp Lower Byte TSL and TSH store the counter value of the time stamp when received messages are stored in the mailbox. |

47.2.8 CANi Mailbox Interrupt Enable Registers (CiMIER0 and CiMIER1) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Interrupts can enable individually for each mailbox.

In normal mailbox mode (all bits) and in FIFO mailbox mode (bits 23 to 0 in the CiMIER1 register and all bits in the CiMIER0 register), each bit corresponds to the mailbox with the related number. These bits enable or disable transmission/reception complete interrupts for the corresponding mailboxes.

- Bit 0 in the CiMIER0 register corresponds to mailbox 0 (MB0).
- Bit 31 in the CiMIER0 register corresponds to mailbox 31 (MB31).
- Bit 0 in the CiMIER1 register corresponds to mailbox 32 (MB32).
- Bit 31 in the CiMIER1 register corresponds to mailbox 63 (MB63).

In FIFO mailbox mode, bits 29, 28, 25, and 24 of the CiMIER1 register specify whether transmit/receive FIFO interrupts are enabled/disabled and timing when interrupt requests are generated.

Write to registers CiMIER0 and CiMIER1 only when the associated CiMCTLj register (i = 0, 1) (j = 0 to 63) is "H'00" and the corresponding mailbox is not processing a transmission or reception abort request. In FIFO mailbox mode, change the bits in the CiMIER1 register for the associated FIFO only when:

- The TFE bit in the CiTFCR register is 0 and the TFEST bit is 1.
- The RFE bit in the CiRFCR register is 0 and the RFEST bit is 1.

CAN0 Mailbox Interrupt Enable Register 1 (COMIER1)

CAN1 Mailbox Interrupt Enable Register 1 (C1MIER1)

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MB63 | MB62 | MB61 | MB60 | MB59 | MB58 | MB57 | MB56 | MB55 | MB54 | MB53 | MB52 | MB51 | MB50 | MB49 | MB48 |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MB47 | MB46 | MB45 | MB44 | MB43 | MB42 | MB41 | MB40 | MB39 | MB38 | MB37 | MB36 | MB35 | MB34 | MB33 | MB32 |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

- Normal mailbox mode

<After Reset: Undefined>

| Bit | Symbol | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|--|
| 31 to 0 | MB63 to MB32 | Undefined | R/W | Interrupt Enable Bits Bit 31 corresponds to mailbox 63 (MB63), and bit 0 corresponds to mailbox 32 (MB32). 0: Interrupt disabled 1: Interrupt enabled |

- FIFO mailbox mode (CiMIER1 only)

<After Reset: Undefined>

| Bit | Symbol | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 31, 30 | MB63 MB62 | Undefined | R | Reserved bits The reset value is undefined. The write value should be "0". These bits are read as "0" after "0" is written to. |
| 29 | MB61 | Undefined | R/W | Receive FIFO Interrupt Generation Timing Control Bit* Receive FIFO interrupt request is generated 0: Every time reception is completed 1: When receive FIFO becomes buffer warning by completion of reception |
| 28 | MB60 | Undefined | R/W | Receive FIFO Interrupt Enable Bit 0: Interrupt disabled 1: Interrupt enabled |
| 27, 26 | MB59 MB58 | Undefined | R | Reserved bits The reset value is undefined. The write value should be "0". These bits are read as "0" after "0" is written to. |
| 25 | MB57 | Undefined | R/W | Transmit FIFO Interrupt Generation Timing Control Bit Transmit FIFO interrupt request is generated 0: Every time transmission is completed 1: When transmit FIFO becomes empty due to completion of transmission |
| 24 | MB56 | Undefined | R/W | Transmit FIFO Interrupt Enable Bit 0: Interrupt disabled 1: Interrupt enabled |
| 23 to 0 | MB55 to MB32 | Undefined | R/W | Interrupt Enable Bits Bit 23 corresponds to mailbox 55 (MB55), and bit 0 corresponds to mailbox 32 (MB32). 0: Interrupt disabled 1: Interrupt enabled |

Note: * No interrupt request is generated when the receive FIFO becomes buffer warning from full. "Buffer warning" indicates a state in which the third unread message is stored in the receive FIFO.

CAN0 Mailbox Interrupt Enable Register 0 (COMIER0)

CAN1 Mailbox Interrupt Enable Register 0 (C1MIER0)

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MB31 | MB30 | MB29 | MB28 | MB27 | MB26 | MB25 | MB24 | MB23 | MB22 | MB21 | MB20 | MB19 | MB18 | MB17 | MB16 |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MB15 | MB14 | MB13 | MB12 | MB11 | MB10 | MB9 | MB8 | MB7 | MB6 | MB5 | MB4 | MB3 | MB2 | MB1 | MB0 |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

<After Reset: Undefined>

| Bit | Symbol | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|--|
| 31 to 0 | MB31 to MB0 | Undefined | R/W | Interrupt Enable Bits Bit 31 corresponds to mailbox 31 (MB31), and bit 0 corresponds to mailbox 0 (MB0). 0: Interrupt disabled 1: Interrupt enabled |

47.2.9 CANi Message Control Register j (CiMCTLj) (i = 0, 1; j = 0 to 63)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Write to the CiMCTLj register in CAN operation mode or CAN halt mode.

Do not use registers CiMCTL56 to CiMCTL63 in FIFO mailbox mode.

CAN0 Message Control Register 0 to 63 (C0MCTL0 to C0MCTL63)

CAN1 Message Control Register 0 to 63 (C1MCTL0 to C1MCTL63)

Registers CiMCTL32 to CiMCTL63

- Transmit mailbox setting enabled (When the TRMREQ bit is "1" and the RECREQ bit is "0")

| | | | | | | | | |
|----------------|------------|------------|---|-------------|---|------------|---------------|--------------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TRM REQ | REC REQ | — | ONE SHOT | — | TRM ABT | TRMAC TIVE | SENTD ATA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R/W | R | R/W | R | R/W |

- Receive mailbox setting enabled (When the TRMREQ bit is "1" and the RECREQ bit is "0")

| | | | | | | | | |
|----------------|------------|------------|---|-------------|---|-------------|---------------|-------------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TRM REQ | REC REQ | — | ONE SHOT | — | MSG LOST | INVAL DATA | NEWD ATA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R/W | R | R/W | R | R/W |

Registers CiMCTL0 to CiMCTL31

| | | | | | | | | |
|----------------|---|------------|---|---|---|-------------|---------------|-------------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | REC REQ | — | — | — | MSG LOST | INVAL DATA | NEWD ATA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R | R | R | R/W | R | R/W |

<After Reset: H'00>

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|--|
| 7 | TRMREQ | B'0 | R/W | <p>Transmit Mailbox Request Bit*2*4</p> <p>The TRMREQ bit selects transmit modes shown in Table 47.12.</p> <p>When TRMREQ bit is set to "1", the corresponding mailbox is configured for transmission of a data frame or a remote frame.</p> <p>When TRMREQ bit is set to "0", the corresponding mailbox is not configured for transmission of a data frame or a remote frame.</p> <p>If the TRMREQ bit is changed from "1" to "0" to cancel the corresponding transmission request, either the TRMABT or SENTDATA bit is set to "1".</p> <p>When setting the TRMREQ bit to "1", do not set the RECREQ bit to "1". To change the configuration of a mailbox from reception to transmission, first abort the reception and then set bits NEWDATA and MSGLOST to "0" before changing to transmission.</p> <p>0: Not configured for transmission 1: Configured for transmission</p> |
| — | — | B'0 | R | <p>Reserved bit (Registers CiMCTL0 to CiMCTL31)</p> <p>This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.</p> |
| 6 | RECREQ | B'0 | R/W | <p>Receive Mailbox Request Bit*2*4*5</p> <p>The RECREQ bit selects receive modes shown in Table 47.12.</p> <p>When the RECREQ bit is set to "1", the corresponding mailbox is configured for reception of a data frame or a remote frame.</p> <p>When the RECREQ bit is set to "0", the corresponding mailbox is not configured for reception of a data frame or a remote frame.</p> <p>Due to hardware protection the RECREQ bit cannot be set to "0" by writing "0" by a program during the following period.</p> <p>Hardware protection is started From the acceptance filter procedure. (the beginning of CRC field) Hardware protection is released</p> <p>For the mailbox that is specified to receive the incoming message, after the received data is stored into the mailbox or a CAN bus error occurs. (i.e. a maximum period of hardware protection is from the beginning of CRC field to the end of the 7th bit of EOF.)</p> <p>For the other mailboxes, after the acceptance filter procedure.</p> <p>If no mailbox is specified to receive the message, after the acceptance filter procedure.</p> <p>When setting the RECREQ bit to "1", do not set "1" to the TRMREQ bit.</p> <p>To change the configuration of a mailbox from transmission to reception, first abort the transmission and then set bits SENTDATA and TRMABT to "0" before changing to reception.</p> <p>0: Not configured for reception 1: Configured for reception</p> |
| 5 | — | B'0 | R | <p>Reserved bit</p> <p>This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.</p> |

| Bit | Symbol | Initial Value | R/W | Description |
|-----|---------|---------------|-----|---|
| 4 | ONESHOT | B'0 | R/W | <p>One-shot Enable Bit*³</p> <p>The ONESHOT bit can be used in the following two ways, receive mode and transmit mode:</p> <p>One-Shot Receive Mode</p> <p>When the ONESHOT bit is set to "1" in receive mode (RECREQ bit = "1" and TRMREQ bit = "0"), the mailbox receives a message only one time. The mailbox does not behave as a receive mailbox after having received a message one time. The behavior of bits NEWDATA and INVALIDDATA is the same as in normal reception mode. In one-shot receive mode, the MSGLOST bit is not set to "1".</p> <p>To set the ONESHOT bit to "0", first write "0" to the RECREQ bit and ensure that it has been set to "0".</p> <p>One-Shot Transmit Mode</p> <p>When the ONESHOT bit is set to "1" in transmit mode (RECREQ bit = "0" and TRMREQ bit = "1"), the CAN module transmits a message only one time.</p> <p>The CAN module does not transmit the message again if a CAN bus error or CAN bus arbitration lost occurs. When transmission is completed, the SENTDATA bit is set to "1". If transmission is not completed due to a CAN bus error or CAN bus arbitration lost, the TRMABT bit is set to "1".</p> <p>Set the ONESHOT bit to "0" after the SENTDATA or TRMABT bit is set to "1".</p> <p>0: One-shot reception or one-shot transmission disabled 1: One-shot reception or one-shot transmission enabled</p> |
| — | — | B'0 | R | <p>Reserved bit (Registers CiMCTL0 to CiMCTL31)</p> <p>This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.</p> |
| 3 | — | B'0 | R | <p>Reserved bit</p> <p>This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.</p> |

| Bit | Symbol | Initial Value | R/W | Description |
|-----|-------------|---------------|-----|---|
| 2 | TRMABT | B'0 | R/W | <p>Transmission Abort Complete Flag (Transmit mailbox setting enabled)*1*2</p> <p>The TRMABT bit is set to "1" in the following cases:</p> <p>Following a transmission abort request, when the transmission abort is completed before starting transmission.</p> <p>Following a transmission abort request, when the CAN module detects CAN bus arbitration lost or a CAN bus error.</p> <p>In one-shot transmission mode (RECREQ bit = "0", TRMREQ bit = "1", and ONESHOT bit = "1"), when the CAN module detects CAN bus arbitration lost or a CAN bus error.</p> <p>The TRMABT bit is not set to "1" when data transmission is completed. In this case, the SENTDATA bit is set to "1".</p> <p>The TRMABT bit is set to "0" by writing "0" by a program.</p> <p>0: Transmission has started, transmission abort failed because transmission is completed, or transmission abort is not requested</p> <p>1: Transmission abort is completed</p> |
| | MSGLOST | B'0 | R/W | <p>Message Lost Flag (Receive mailbox setting enabled)*1*2</p> <p>The MSGLOST bit is set to "1" when the mailbox is overwritten or overrun by a new received message while the NEWDATA bit is "1".</p> <p>The MSGLOST bit is set to "1" at the end of the 6th bit of EOF.</p> <p>The MSGLOST bit is set to "0" by writing "0" by a program.</p> <p>In both overwrite and overrun modes, the MSGLOST bit is not set to "0" by writing "0" by a program during the 5 peripheral clock (clkp1) cycles following the 6th bit of EOF.</p> <p>0: Message is not overwritten or overrun</p> <p>1: Message is overwritten or overrun</p> |
| 1 | TRMACTIVE | B'0 | R | <p>Transmission-in-Progress Status Flag (Transmit mailbox setting enabled)</p> <p>The TRMACTIVE bit is set to "1" when the corresponding mailbox of the CAN module begins transmitting a message.</p> <p>The TRMACTIVE is set to "0" when the CAN module has lost CAN bus arbitration, a CAN bus error occurs, or data transmission is completed.</p> <p>0: Transmission is pending or transmission is not requested</p> <p>1: From acceptance of transmission request to completion of transmission, or error/arbitration lost</p> |
| | INVALIDDATA | B'0 | R | <p>Reception-in-Progress Status Flag (Receive mailbox setting enabled)</p> <p>After the completion of a message reception, the INVALIDDATA bit is set to "1" while the received message is being updated into the corresponding mailbox.</p> <p>The INVALIDDATA bit is set to "0" immediately after the message has been stored. If the mailbox is read while the INVALIDDATA bit is "1", the data is undefined.</p> <p>0: Message valid</p> <p>1: Message being updated</p> |

| Bit | Symbol | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | SENTDATA | B'0 | R/W | <p>Transmission Complete Flag (Transmit mailbox setting enabled)*1*2</p> <p>The SENTDATA bit is set to "1" when data transmission from the corresponding mailbox is completed.</p> <p>The SENTDATA bit is set to "0" by writing "0" by a program.</p> <p>To set the SENTDATA bit to "0", first set the TRMREQ bit to "0".</p> <p>Bits SENTDATA and TRMREQ cannot be set to "0" simultaneously.</p> <p>To transmit a new message from the corresponding mailbox, set the SENTDATA bit to "0".</p> <p>0: Transmission is not completed (pending)</p> <p>1: Transmission is completed (success)</p> |
| | NEWDATA | B'0 | R/W | <p>Reception Complete Flag (Receive mailbox setting enabled)*1*2</p> <p>The NEWDATA bit is set to "1" when a new message is being stored or has been stored to the mailbox. The timing for setting this bit to "1" is simultaneous with the INVALIDDATA bit.</p> <p>The NEWDATA bit is set to "0" by writing "0" by a program.</p> <p>The NEWDATA bit is not set to "0" by writing "0" by a program while the related INVALIDDATA bit is "1".</p> <p>0: No data has been received or "0" is written to the NEWDATA bit</p> <p>1: A new message is being stored or has been stored to the mailbox</p> |

- Notes:
1. Write "0" only. Writing "1" has no effect.
 2. When writing "0" to bits NEWDATA, SENTDATA, MSGLOST, TRMABT, RECREQ, and TRMREQ by a program, use the MOV instruction to ensure that only the specified bit is set to "0" and the other bits are set to "1".
 3. To enter one-shot receive mode, write "1" to the ONESHOT bit at the same time as setting the RECREQ bit to "1".
To exit one-shot receive mode, write "0" to the ONESHOT bit after writing "0" to the RECREQ bit and confirming it has been set to "0".
To enter one-shot transmit mode, write "1" to the ONESHOT bit at the same time as setting the TRMREQ bit to "1".
To exit one-shot transmit mode, write "0" to the ONESHOT bit after the message has been transmitted or aborted.
 4. Do not set both the RECREQ and TRMREQ bits to "1".
 5. When setting the RECREQ bit to "0", set bits MSGLOST, NEWDATA, RECREQ to "0" simultaneously.

47.2.10 CANi Receive FIFO Control Register (CiRFCR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Write to the CiRFCR registers in CAN operation mode or CAN halt mode.

CAN0 Receive FIFO Control Register (C0RFCR)

CAN1 Receive FIFO Control Register (C1RFCR)

| | | | | | | | | |
|----------------|-------|-------|-------|-------|-------|---|---|-----|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RFEST | RFWST | RFFST | RFMLF | RFUST | | | RFE |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W |

<After Reset: H'80>

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|--|
| 7 | RFEST | B'1 | R | <p>Receive FIFO Empty Status Flag</p> <p>The RFEST bit is set to "1" (no unread message in receive FIFO) when the number of unread messages in the receive FIFO is "0". The RFEST bit is set to "1" when the RFE bit is set to "0". The RFEST bit is set to "0" (unread message in receive FIFO) when the number of unread messages in the receive FIFO is one or more.</p> <p>0: Unread message in receive FIFO 1: No unread message in receive FIFO</p> |
| 6 | RFWST | B'0 | R | <p>Receive FIFO Buffer Warning Status Flag</p> <p>The RFWST bit is set to "1" (receive FIFO is buffer warning) when the number of unread messages in the receive FIFO is 3. The RFWST bit is "0" (receive FIFO is not buffer warning) when the number of unread messages in the receive FIFO is less than 3 or equal to 4. The RFWST bit is set to "0" when the RFE bit is "0".</p> <p>0: Receive FIFO is not buffer warning 1: Receive FIFO is buffer warning (3 unread messages)</p> |
| 5 | RFFST | B'0 | R | <p>Receive FIFO Full Status Flag</p> <p>The RFFST bit is set to "1" (receive FIFO is full) when the number of unread messages in the receive FIFO is 4. The RFFST bit is "0" (receive FIFO is not full) when the number of unread messages in the receive FIFO is less than 4. The RFFST bit is set to "0" when the RFE bit is "0".</p> <p>0: Receive FIFO is not full 1: Receive FIFO full (4 unread messages)</p> |

| Bit | Symbol | Initial Value | R/W | Description |
|--------|--------|---------------|-----|--|
| 4 | RFMLF | B'0 | R | <p>Receive FIFO Message Lost Flag</p> <p>The RFMLF bit is set to "1" (receive FIFO message lost has occurred) when the receive FIFO receives a new message and the receive FIFO is full. The timing for setting this bit to "1" is at the end of the 6th bit of EOF.</p> <p>The RFMLF bit is set to "0" by writing "0" by a program (writing "1" has no effect). In both overwrite and overrun modes, the RFMLF bit cannot be set to "0" (receive FIFO message lost has not occurred) by writing "0" by a program due to hardware protection during the five cycles of peripheral clock (clkp1) following the 6th bit of EOF, if the receive FIFO is full and determined to receive the message.</p> <p>0: No receive FIFO message lost has occurred 1: Receive FIFO message lost has occurred</p> |
| 3 to 1 | RFUST | All 0 | R | <p>Receive FIFO Unread Message Number Status Flag</p> <p>The RFUST bit indicates the number of unread messages in the receive FIFO.</p> <p>The value of the RFUST bit is initialized to "000" when the RFE bit is set to "0".</p> <p>B'000: No unread message B'001: 1 unread message B'010: 2 unread messages B'011: 3 unread messages B'100: 4 unread messages B'101: Reserved B'110: Reserved B'111: Reserved</p> |
| 0 | RFE | B'0 | R/W | <p>Receive FIFO Enable Bit</p> <p>When the RFE bit is set to "1", the receive FIFO is enabled.</p> <p>When this bit is set to "0", the receive FIFO is disabled for reception and becomes empty (RFEST bit = "1").</p> <p>Do not set this bit to "1" in normal mailbox mode (MBM bit in the CiCTLR register = "0").</p> <p>Due to hardware protection, the RFE bit is not set to "0" by writing "0" by a program during the following period:</p> <p>The hardware protection is started</p> <p>From the acceptance filter procedure (the beginning of CRC field)</p> <p>The hardware protection is released</p> <p>If the receive FIFO is specified to receive the incoming message, after the received data is stored into the receive FIFO or a CAN bus error occurs. (i.e. maximum period of hardware protection is from the beginning of CRC field to the end of 7th bit of EOF.)</p> <p>If the receive FIFO is not specified to receive the message, after the acceptance filter procedure.</p> <p>0: Receive FIFO disabled 1: Receive FIFO enabled</p> |

Figure 47.2 shows the receive FIFO mailbox operation.

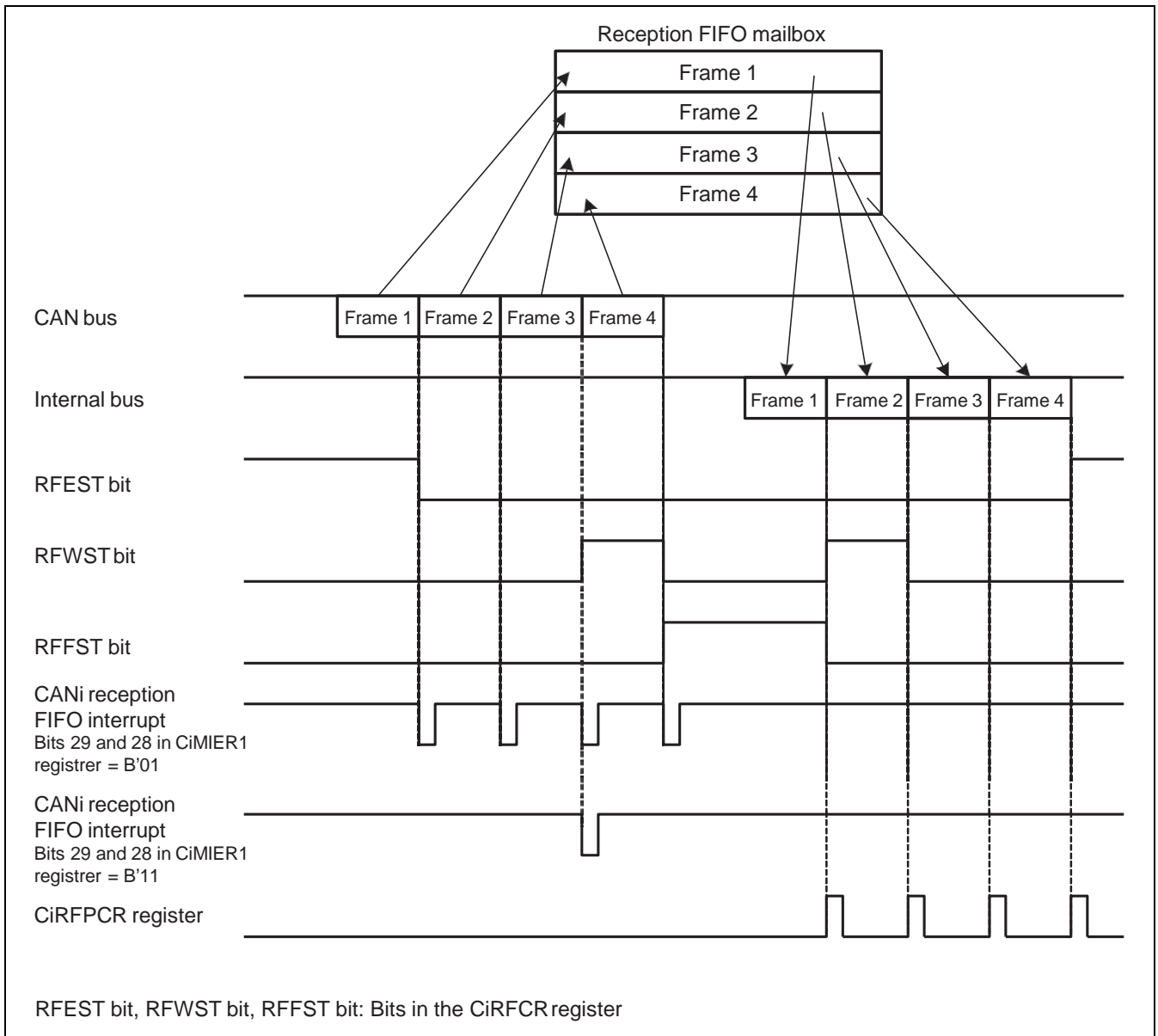


Figure 47.2 Receive FIFO Mailbox Operation
 (Bits 29 and 28 in CiMIER1 Register = "01" and "11") (i = 0, 1)

47.2.11 CANi Receive FIFO Pointer Control Register (CiRFPCR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

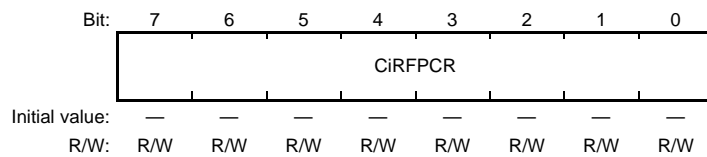
When the receive FIFO is not empty, write "H'FF" to the CiRFPCR register by a program to increment the CPU-side pointer for the receive FIFO to the next mailbox location.

Do not write to the CiRFPCR register when the RFE bit in the CiRFCR register is "0" (receive FIFO disabled).

Both the CAN-side pointer and the CPU-side pointer are incremented when a new message is received and the RFFST bit is "1" (receive FIFO is full) in overwrite mode. When the RFMLF bit is "1" in this condition, the CPU-side pointer cannot be incremented by writing to the CiRFPCR register by a program.

CAN0 Receive FIFO Pointer Control Register (C0RFPCR)

CAN1 Receive FIFO Pointer Control Register (C1RFPCR)



<After Reset: Undefined>

| Bit | Symbol | Initial Value | R/W | Description |
|--------|---------|---------------|-----|--|
| 7 to 0 | CiRFPCR | Undefined | R/W | The CPU-side pointer for the receive FIFO is incremented by writing "H'FF" |

47.2.12 CANi Transmit FIFO Control Register (CiTFCR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Write to the CiTFCR register in CAN operation mode or CAN halt mode.

CAN0 Transmit FIFO Control Register (C0TFCR)

CAN1 Transmit FIFO Control Register (C1TFCR)

| | | | | | | | | |
|----------------|-------|-------|---|---|-------|---|-----|-----|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TFEST | TFFST | — | — | TFUST | | TFE | |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W |

<After Reset: H'80>

| Bit | Symbol | Initial Value | R/W | Description |
|--------|--------|---------------|-----|---|
| 7 | TFEST | B'1 | R | <p>Transmit FIFO Empty Status Bit</p> <p>The TFEST bit is set to "1" (no message in transmit FIFO) when the number of unsent messages in the transmit FIFO is "0". The TFEST bit is set to "1" when transmission from the transmit FIFO has been aborted.</p> <p>The TFEST bit is set to "0" (message in transmit FIFO) when the number of unsent messages in the transmit FIFO is not "0".</p> <p>0: Unsent message in transmit FIFO 1: No unsent message in transmit FIFO</p> |
| 6 | TFFST | B'0 | R | <p>Transmit FIFO Full Status Bit</p> <p>The TFFST bit is set to "1" (transmit FIFO is full) when the number of unsent messages in the transmit FIFO is 4. The TFFST bit is set to "0" (transmit FIFO is not full) when the number of unsent messages in the transmit FIFO is less than 4. The TFFST bit is set to "0" when transmission from the transmit FIFO has been aborted.</p> <p>0: Transmit FIFO is not full 1: Transmit FIFO is full (4 unsent messages)</p> |
| 5, 4 | — | All 0 | R | <p>Reserved bits</p> <p>These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits.</p> |
| 3 to 1 | TFUST | B'000 | R | <p>Transmit FIFO Unsent Message Number Status Bits</p> <p>The TFUST bit indicates the number of unsent messages in the transmit FIFO.</p> <p>After the TFE bit is set to "0", the value of the TFUST bit is initialized to "B'000" when transmission abort or transmission is completed.</p> <p>B'000: No unsent message B'001: 1 unsent message B'010: 2 unsent messages B'011: 3 unsent messages B'100: 4 unsent messages B'101: Reserved B'110: Reserved B'111: Reserved</p> |

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|--|
| 0 | TFE | B'0 | R/W | <p>Transmit FIFO Enable Bit</p> <p>When the TFE bit is set to "1", the transmit FIFO is enabled.</p> <p>When the TFE bit is set to "0", the transmit FIFO becomes empty (TFEST bit = "1") and then unsent messages from the transmit FIFO are lost as described below:</p> <p>If a message from the transmit FIFO is not scheduled for the next transmission or during transmission.</p> <p>Following the completion of transmission, a CAN bus error, CAN bus arbitration lost, or entry to CAN halt mode if a message from the transmit FIFO is scheduled for the next transmission or already during transmission.</p> <p>Before setting the TFE bit to set to "1" again, ensure that the TFEST bit has been set to "1".</p> <p>After setting the TFE bit to "1", write transmit data into the CiMB56 register.</p> <p>Do not set the TFE bit to "1" in normal mailbox mode (MBM bit in the CiCTRL register = "0").</p> <p>0: Transmit FIFO disabled 1: Transmit FIFO enabled</p> |

Figure 47.3 shows the transmit FIFO mailbox operation.

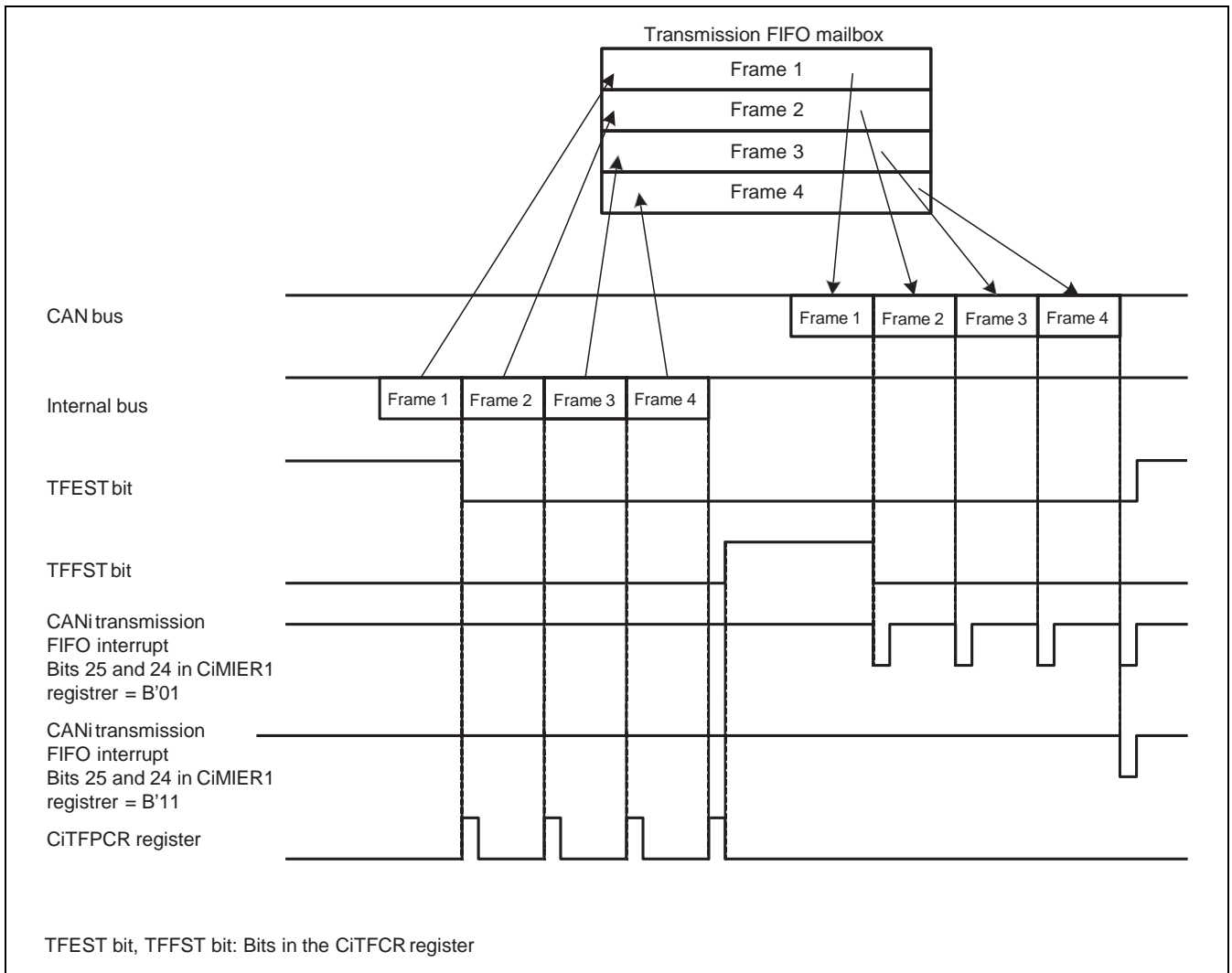


Figure 47.3 Transmit FIFO Mailbox Operation
(Bits 25 and 24 in CiMIER1 Register = "01" and "11") (i = 0, 1)

47.2.13 CANi Transmit FIFO Pointer Control Register (CiTFPCR) (i = 0, 1)

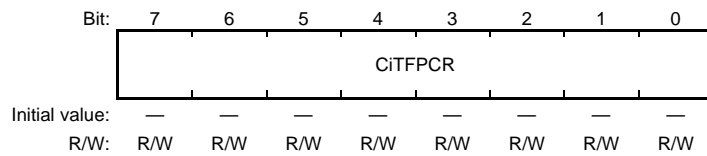
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

When the transmit FIFO is not full, write "H'FF" to the CiTFPCR register by a program to increment the CPU-side pointer for the transmit FIFO to the next mailbox location.

Do not write to the CiTFPCR register when the TFE bit in the CiTFPCR register is "0" (transmit FIFO disabled).

CAN0 Transmit FIFO Pointer Control Register (C0TFPCR)

CAN1 Transmit FIFO Pointer Control Register (C1TFPCR)



<After Reset: Undefined>

| Bit | Symbol | Initial Value | R/W | Description |
|--------|---------|---------------|-----|---|
| 7 to 0 | CiTFPCR | Undefined | R/W | The CPU-side pointer for the transmit FIFO is incremented by writing "H'FF" |

47.2.14 CANi Status Register (CiSTR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CAN0 Status Register (C0STR)

CAN1 Status Register (C1STR)

| | | | | | | | | | | | | | | | | |
|----------------|----|-------|-------|------|------|-------|-------|-------|-----|-------|-------|-------|------|------|------|------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | RECST | TRMST | BOST | EPST | SLPST | HLTST | RSTST | EST | TABST | FMLST | NMLST | TFST | RFST | SDST | NDST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

<After Reset: H'0500>

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|---|
| 15 | — | B'0 | R | Reserved bit This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit. |
| 14 | RECST | B'0 | R | Receive Status Flag (receiver) The RECST bit is set to "1" when the CAN module performs as a receiver node. The RECST bit is set to "0" when the CAN module performs as a transmitter node or is in bus-idle state. 0: Bus idle or transmission in progress 1: Reception in progress |
| 13 | TRMST | B'0 | R | Transmit Status Flag (transmitter) The TRMST bit is set to "1" when the CAN module performs as a transmitter node or is in the bus-off state. The TRMST bit is set to "0" when the CAN module performs as a receiver node or is in bus-idle state. 0: Bus idle or reception in progress 1: Transmission in progress or in bus-off state |
| 12 | BOST | B'0 | R | Bus-Off Status Flag The BOST bit is set to "1" when the value of the CiTECR register exceeds 255 and the CAN module is in the bus-off state ($TEC \geq 256$). The BOST bit is set to "0" when the CAN module is not in the bus-off state. 0: Not in bus-off state 1: In bus-off state |
| 11 | EPST | B'0 | R | Error-Passive Status Flag The EPST bit is set to "1" when the value of the CiTECR or CiRECR register exceeds 127 and the CAN module is in error-passive state ($128 \leq TEC < 256$ or $128 \leq REC < 256$). The EPST bit is set to "0" when the CAN module is not in the error-passive state. TEC indicates the value of the transmit error counter (CiTECR register) and REC indicates the value of the receive error counter (CiRECR register). 0: Not in error-passive state 1: In error-passive state |

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|---|
| 10 | SLPST | B'1 | R | <p>CAN Sleep Status Flag</p> <p>The SLPST bit is set to "1" when the CAN module is in CAN sleep mode. The SLPST bit is set to "0" when the CAN module is not in CAN sleep mode.</p> <p>0: Not in CAN sleep mode 1: In CAN sleep mode</p> |
| 9 | HLTST | B'0 | R | <p>CAN Halt Status Flag</p> <p>The HLTST bit is set to "1" when the CAN module is in CAN halt mode. The HLTST bit is set to "0" when the CAN module is not in CAN halt mode. Even when the state is changed from CAN halt mode to CAN sleep mode, the HLTST bit remains "1".</p> <p>0: Not in CAN halt mode 1: In CAN halt mode</p> |
| 8 | RSTST | B'1 | R | <p>CAN Reset Status Flag</p> <p>The RSTST bit is set to "1" when the CAN module is in CAN reset mode. The RSTST bit is "0" when the CAN module is not in CAN reset mode. Even when the state is changed from CAN reset mode to CAN sleep mode, the RSTST bit remains "1".</p> <p>0: Not in CAN reset mode 1: In CAN reset mode</p> |
| 7 | EST | B'0 | R | <p>Error Status Flag</p> <p>The EST bit is "1" when at least one error is detected by the CiEIFR register regardless of the value of the CiEIER register. The EST bit is set to "0" when no error is detected by the CiEIFR register.</p> <p>0: No error occurred 1: Error occurred</p> |
| 6 | TABST | B'0 | R | <p>Transmission Abort Status Flag</p> <p>The TABST bit is set to "1" when at least one TRMABT bit in the CiMCTLj register is "1" regardless of the value of the CiMIER register. The TABST bit is set to "0" when all TRMABT bits are "0".</p> <p>0: No mailbox with TRMABT bit = "1" 1: Mailbox(es) with TRMABT bit = "1"</p> |
| 5 | FMLST | B'0 | R | <p>FIFO Mailbox Message Lost Status Flag</p> <p>The FMLST bit is set to "1" when the RFMLF bit in the CiRFCR register is "1" regardless of the value of the CiMIER register. The FMLST bit is set to "0" when the RFMLF bit is "0".</p> <p>0: RFMLF bit = "0" 1: RFMLF bit = "1"</p> |
| 4 | NMLST | B'0 | R | <p>Normal Mailbox Message Lost Status Flag</p> <p>The NMLST bit is set to "1" when at least one MSGLOST bit in the CiMCTLj register is "1" regardless of the value of the CiMIER register. The NMLST bit is set to "0" when all MSGLOST bit is "0".</p> <p>0: No mailbox with MSGLOST bit = "1" 1: Mailbox(es) with MSGLOST bit = "1"</p> |
| 3 | TFST | B'0 | R | <p>Transmit FIFO Status Flag</p> <p>The TFST bit is set to "1" when the transmit FIFO is not full. The TFST bit is set to "0" when the transmit FIFO is full. The TFST bit is set to "0" when normal mailbox mode is selected.</p> <p>0: Transmit FIFO is full 1: Transmit FIFO is not full</p> |

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|--|
| 2 | RFST | B'0 | R | <p>Receive FIFO Status Flag</p> <p>The RFST bit is set to "1" when the receive FIFO is not empty. The RFST bit is set to "0" when the receive FIFO is empty.</p> <p>The RFST bit is set to "0" when normal mailbox mode is selected.</p> <p>0: No message in receive FIFO 1: Message in receive FIFO</p> |
| 1 | SDST | B'0 | R | <p>SENTDATA Status Flag</p> <p>The SDST bit is set to "1" when at least one SENTDATA bit in the CiMCTLj register is "1" regardless of the value of the CiMIER register. The SDST bit is set to "0" when all SENTDATA bits are "0".</p> <p>0: No mailbox with SENTDATA bit = "1" 1: Mailbox(es) with SENTDATA bit = "1"</p> |
| 0 | NDST | B'0 | R | <p>NEWDATA Status Flag</p> <p>The NDST bit is set to "1" when at least one NEWDATA bit in the CiMCTLj register is "1" regardless of the value of the CiMIER register. The NDST bit is set to "0" when all NEWDATA bits are "0".</p> <p>0: No mailbox with NEWDATA bit = "1" 1: Mailbox(es) with NEWDATA bit = "1"</p> |

47.2.15 CANi Mailbox Search Mode Register (CiMSMR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Write to the CiMSMR register in CAN operation mode or CAN halt mode.

CAN0 Mailbox Search Mode Register (COMSMR)

CAN1 Mailbox Search Mode Register (C1MSMR)

| | | | | | | | | |
|----------------|---|---|---|---|---|---|------|-----|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | MBSM | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W |

<After Reset: H'00>

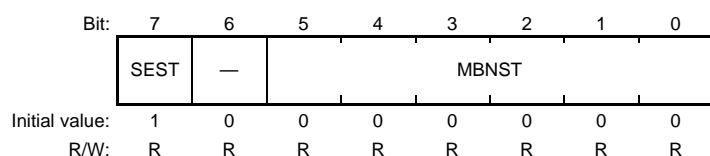
| Bit | Symbol | Initial Value | R/W | Description |
|--------|--------|---------------|-----|---|
| 7 to 2 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 1, 0 | MBSM | All 0 | R/W | Mailbox Search Mode Select Bits The MBSM bit selects the search mode for the mailbox search function. When the MBSM bit is "00", receive mailbox search mode is selected. In this mode, the search targets are the NEWDATA bit in the CiMCTLj register (j = 0 to 63) for the normal mailbox and the RFEST bit in the CiRFCR register. When the MBSM bit is "01", transmit mailbox search mode is selected. In this mode, targets the SENTDATA bit in the CiMCTLj register. When the MBSM bit is "10", message lost search mode is selected. In this mode, targets the MSGLOST bit in the CiMCTLj register for the normal mailbox and the RFMLF bit in the CiRFCR register. When the MBSM bit is "11", channel search mode is selected. In this mode, the search target is the CiCSSR register. Refer to section 47.2.17, CANi Channel Search Support Register (CiCSSR) (i = 0, 1). B'00: Receive mailbox search mode B'01: Transmit mailbox search mode B'10: Message lost search mode B'11: Channel search mode |

47.2.16 CANi Mailbox Search Status Register (CiMSSR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CAN0 Mailbox Search Status Register (C0MSSR)

CAN1 Mailbox Search Status Register (C1MSSR)



<After Reset: H'80>

| Bit | Symbol | Initial Value | R/W | Description |
|--------|--------|---------------|-----|--|
| 7 | SEST | B'1 | R | <p>Search Result Status Bit</p> <p>The SEST bit is set to "1" when no corresponding mailbox is found after searching all mailboxes. For example, in transmit mailbox search mode, the SEST bit is set to "1" when no SENTDATA bit for mailboxes is "1". The SEST bit is set to "0" when at least one SENTDATA bit is "1". When the SEST bit is "1", the value of the MBNST bits is undefined.</p> <p>0: Search result found 1: No search result</p> |
| 6 | — | B'0 | R | <p>Reserved bit</p> <p>This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to this bit.</p> |
| 5 to 0 | MBNST | All 0 | R | <p>Search Result Mailbox Number Status Bits</p> <p>The MBNST bit outputs the smallest mailbox number that is searched in each mode of the CiMSMR register. In receive mailbox, transmit mailbox, and message lost search modes, the value of the mailbox i.e., the search result to be output, is updated as described below:</p> <p>When the NEWDATA, SENTDATA or MSGLOST bit for the output mailbox is set to "0".</p> <p>When the NEWDATA, SENTDATA or MSGLOST bit for a higher-priority mailbox is set to "1".</p> <p>In receive mailbox search and message lost search modes, the receive FIFO (mailbox [60]) is output when the receive FIFO is not empty and there are no unread received messages or no lost messages in any of the normal mailboxes (mailboxes [0] to [55]). In transmit mailbox search mode, the transmit FIFO (mailbox [56]) is not output. Table 47.8 lists the behavior of MBNST bit in FIFO mailbox mode.</p> <p>In channel search mode, the MBNST bit outputs the corresponding channel number. After the CiMSSR register is read by a program, the next target channel number is output.</p> |

Table 47.8 Operation of MBNST Bit in FIFO Mailbox Mode

| MBSM Bit | Mailbox [56] (Transmit FIFO) | Mailbox [60] (Receive FIFO) |
|-----------------|-------------------------------------|---|
| "00" | Mailbox [56] is not output. | Mailbox [60] is output when no NEWDATA bit for the normal mailbox is set to "1" and the receive FIFO is not empty. |
| "01" | | Mailbox [60] is not output. |
| "10" | | Mailbox [60] is output when no MSGLOST bit for the normal mailbox is set to "1" and the RFMLF bit is set to "1" (receive FIFO message lost has occurred) in the receive FIFO. |
| "11" | | Mailbox [60] is not output. |

47.2.17 CANi Channel Search Support Register (CiCSSR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

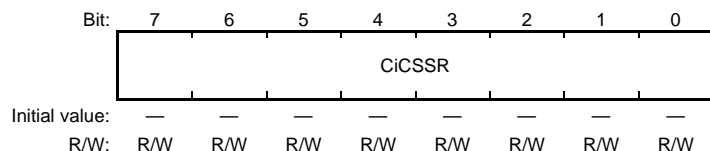
The bits in the CiCSSR register, which are set to "1", are encoded by an 8/3 encoder (the lower bit position, the higher priority) and output to the MBNST bits in the CiMSSR register.

The CiMSSR register outputs the updated value whenever the CiMSSR register is read by a program.

Write to the CiCSSR register only when the MBSM bit in the CiMSMR register is "11" (channel search mode). Write to this register in CAN operation mode or CAN halt mode.

CAN0 Channel Search Support Register (C0CSSR)

CAN1 Channel Search Support Register (C1CSSR)



<After Reset: Undefined>

| Bit | Symbol | Initial Value | R/W | Description |
|--------|--------|---------------|-----|--|
| 7 to 0 | CiCSSR | Undefined | R/W | When the value for the channel search is input, the channel number is output to the CiMSSR register. |

Figure 47.4 shows the write and read of registers CiCSSR and CiMSSR.

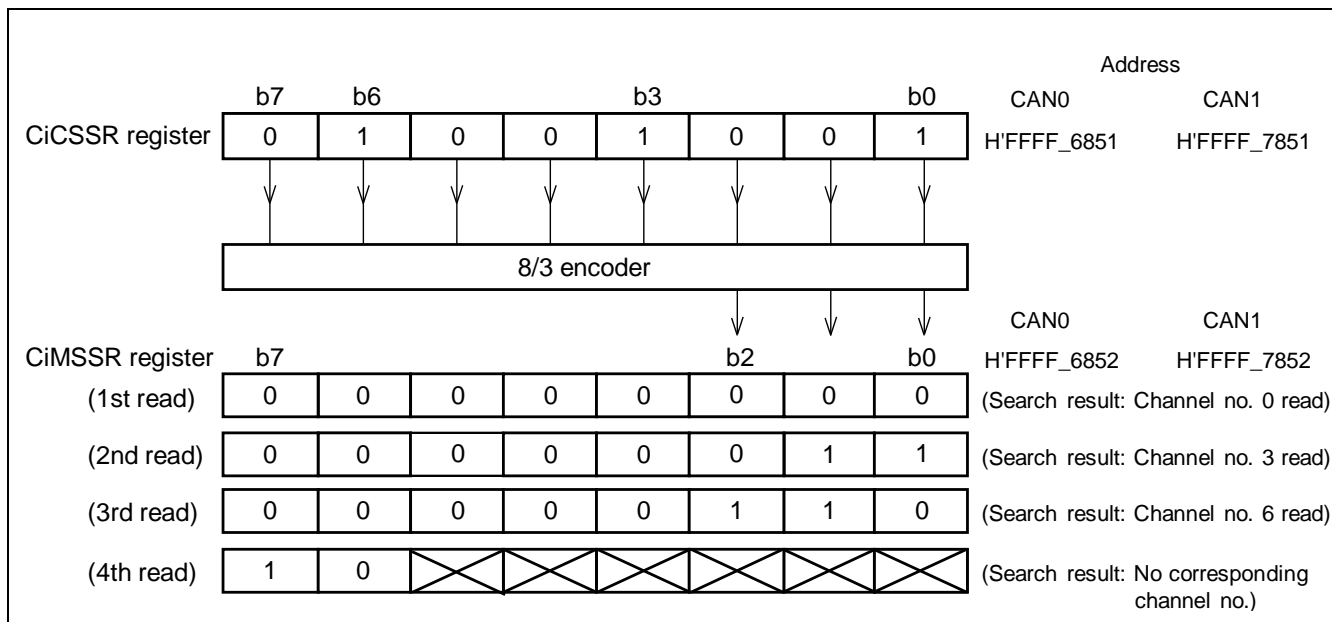


Figure 47.4 Write and Read of Registers CiCSSR and CiMSSR (i = 0, 1)

The value of the CiCSSR register is also updated whenever the CiMSSR register is read. When the CiCSSR register is read, the value before the 8/3 encoder conversion is read.

47.2.18 CANi Acceptance Filter Support Register (CiAFSR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The acceptance filter support unit (ASU) can be used for data table (8 bits × 256) search. In the data table, all standard IDs created by the user are set to be valid/invalid in bit units. When the CAFSR register is written with the 16-bit unit data including the SID bit in the CiMBj register (j = 0 to 63), in which a received ID is stored, a decoded row (byte offset) position and column (bit) position for data table search can be read. The ASU can be used for standard (11-bit) IDs only.

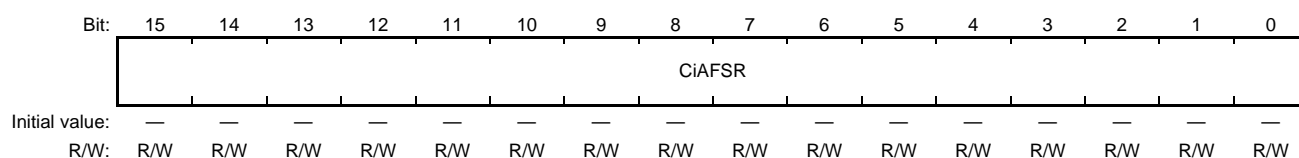
The ASU is enabled in the following cases:

- When the ID to receive cannot be masked by the acceptance filter.
(Example) IDs to receive: H'078, H'087, H'111
- When there are too many IDs to receive and software filtering time is expected to be shortened.

Write to the CiAFSR register in CAN operation mode or CAN halt mode.

CAN0 Acceptance Filter Support Register (COAFSR)

CAN1 Acceptance Filter Support Register (C1AFSR)



<After Reset: Undefined>

| Bit | Symbol | Initial Value | R/W | Description |
|---------|--------|---------------|-----|--|
| 15 to 0 | CiAFSR | Undefined | R/W | After the standard ID of a received message is written, the value converted for data table search can be read. |

Figure 47.5 shows the write and read of CiAFSR register.

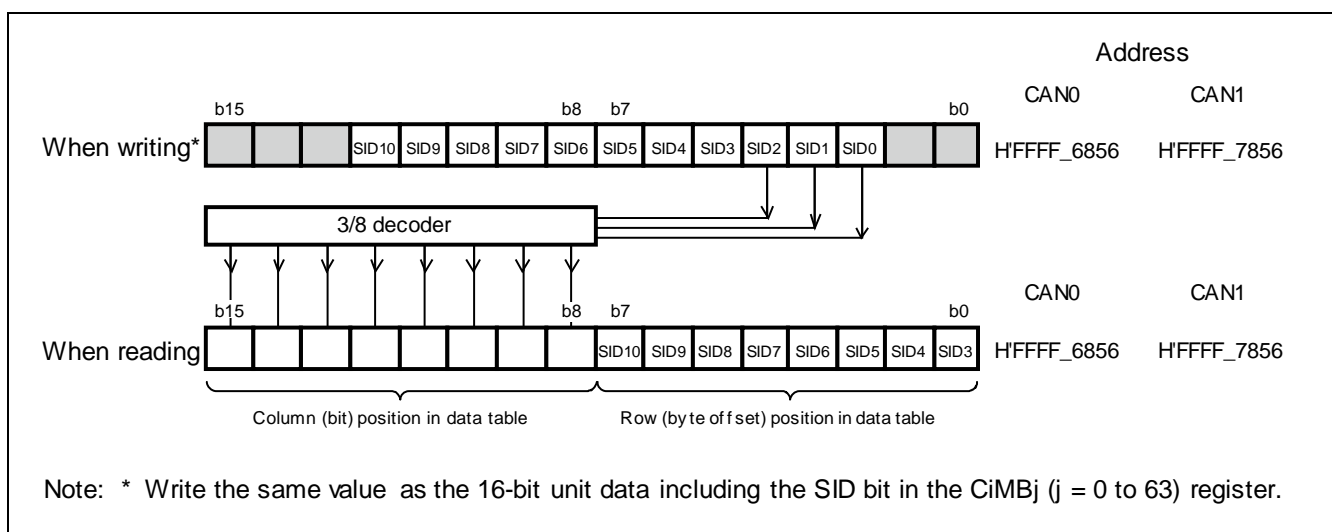


Figure 47.5 Write and Read of CiAFSR Register (i = 0, 1)

47.2.19 CANi Error Interrupt Enable Register (CiEIER) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CiEIER register is used to set the error interrupt enabled/disabled individually for each error interrupt source in the CiEIFR register.

Write to the CiEIER register in CAN reset mode.

CAN0 Error Interrupt Enable Register (C0EIER)

CAN1 Error Interrupt Enable Register (C1EIER)

| | | | | | | | | |
|----------------|------|------|------|-------|-------|------|------|------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | BLIE | OLIE | ORIE | BORIE | BOEIE | EPIE | EWIE | BEIE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

<After Reset: H'00>

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|--|
| 7 | BLIE | B'0 | R/W | <p>Bus Lock Interrupt Enable Bit</p> <p>When the BLIE bit is "0", no error interrupt request is generated even if the BLIF bit in the CiEIFR register is set to "1".</p> <p>When the BLIE bit is "1", an error interrupt request is generated if the BLIF bit is set to "1".</p> <p>0: Bus lock interrupt disabled</p> <p>1: Bus lock interrupt enabled</p> |
| 6 | OLIE | B'0 | R/W | <p>Overload Frame Transmit Interrupt Enable Bit</p> <p>When the OLIE bit is "0", no error interrupt request is generated even if the OLIF bit in the CiEIFR register is set to "1".</p> <p>When the OLIE bit is "1", an error interrupt request is generated if the OLIF bit is set to "1".</p> <p>0: Overload frame transmit interrupt disabled</p> <p>1: Overload frame transmit interrupt enabled</p> |
| 5 | ORIE | B'0 | R/W | <p>Receive Overrun Interrupt Enable Bit</p> <p>When the ORIE bit is "0", an error interrupt request is not generated even if the ORIF bit in the CiEIFR register is set to "1".</p> <p>When the ORIE bit is "1", an error interrupt request is generated if the ORIF bit is set to "1".</p> <p>0: Receive overrun interrupt disabled</p> <p>1: Receive overrun interrupt enabled</p> |
| 4 | BORIE | B'0 | R/W | <p>Bus-Off Recovery Interrupt Enable Bit</p> <p>When the BORIE bit is "0", an error interrupt request is not generated even if the BORIF bit in the CiEIFR register is set to "1". When the BORIE bit is set to "1", an error interrupt request is generated if the BORIF bit is set to "1".</p> <p>0: Bus-off recovery interrupt disabled</p> <p>1: Bus-off recovery interrupt enabled</p> |

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|---|
| 3 | BOEIE | B'0 | R/W | <p>Bus-Off Entry Interrupt Enable Bit</p> <p>When the BOEIE bit is "0", no error interrupt request is generated even if the BOEIF bit in the CiEIFR register is set to "1".</p> <p>When the BOEIE bit is "1", an error interrupt request is generated if the BOEIF bit is set to "1".</p> <p>0: Bus-off entry interrupt disabled 1: Bus-off entry interrupt enabled</p> |
| 2 | EPIE | B'0 | R/W | <p>Error-Passive Interrupt Enable Bit</p> <p>When the EPIE bit is "0", no error interrupt request is generated even if the EPIF bit in the CiEIFR register is set to "1".</p> <p>When the EPIE bit is "1", an error interrupt request is generated if the EPIF bit is set to "1".</p> <p>0: Error-passive interrupt disabled 1: Error-passive interrupt enabled</p> |
| 1 | EWIE | B'0 | R/W | <p>Error-Warning Interrupt Enable Bit</p> <p>When the EWIE bit is "0", no error interrupt request is generated even if the EWIF bit in the CiEIFR register is set to "1".</p> <p>When the EWIE bit is "1", an error interrupt request is generated if the EWIF bit is set to "1".</p> <p>0: Error-warning interrupt disabled 1: Error-warning interrupt enabled</p> |
| 0 | BEIE | B'0 | R/W | <p>Bus Error Interrupt Enable Bit</p> <p>When the BEIE bit is "0", no error interrupt request is generated even if the BEIF bit in the CiEIFR register is set to "1".</p> <p>When the BEIE bit is "1", an error interrupt request is generated if the BEIF bit is set to "1".</p> <p>0: Bus error interrupt disabled 1: Bus error interrupt enabled</p> |

47.2.20 CANi Error Interrupt Factor Judge Register (CiEIFR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

If an event corresponding to each bit occurs, the corresponding bit in the CiEIFR register is set to "1" regardless of the setting of the CiEIER register.

To set each bit to "0", write "0" by a program. If the set timing occurs simultaneously with the clear timing by the program, the bit becomes "1".

When writing "0" to a single bit by a program, use the MOV instruction to ensure that only the specified bit is set to "0" and the other bits are set to "1". Writing "1" has no effect to these bit values.

CAN0 Error Interrupt Factor Judge Register (C0EIFR)

CAN1 Error Interrupt Factor Judge Register (C1EIFR)

| | | | | | | | | |
|----------------|------|------|------|-------|-------|------|------|------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | BLIF | OLIF | ORIF | BORIF | BOEIF | EPIF | EWIF | BEIF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

<After Reset: H'00>

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|--|
| 7 | BLIF | B'0 | R/W | Bus Lock Detect Flag* The BLIF bit becomes 1 if 32 consecutive dominant bits are detected on the CAN bus while the CAN module is in CAN operation mode. After the BLIF bit becomes 1, bus lock can be detected again after either of the following conditions is satisfied: <ul style="list-style-type: none"> • After this bit is set to 0 from 1, recessive bits are detected (bus lock is resolved). • After this bit is set to 0 from 1, the CAN module enters CAN reset mode and then enters CAN operation mode again (internal reset). 0: No bus lock detected 1: Bus lock detected |
| 6 | OLIF | B'0 | R/W | Overload Frame Transmission Detect Flag* The OLIF bit is set to "1" if the transmitting condition of an overload frame is detected when the CAN module performs transmission or reception. 0: No overload frame transmission detected 1: Overload frame transmission detected |

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|--|
| 5 | ORIF | B'0 | R/W | <p>Receive Overrun Detect Flag*</p> <p>The ORIF bit is set to "1" when a receive overrun occurs.</p> <p>This bit is not to set to "1" in overwrite mode. In overwrite mode, a reception complete interrupt request is generated if an overwrite condition occurs and the ORIF bit is not set to "1".</p> <p>In normal mailbox mode, if an overrun occurs in any of mailboxes [0] to [63] in overrun mode, this bit is set to "1".</p> <p>In FIFO mailbox mode, if an overrun occurs in any of mailboxes [0] to [55] or the receive FIFO in overrun mode, this bit is set to "1".</p> <p>0: No receive overrun detected 1: Receive overrun detected</p> |
| 4 | BORIF | B'0 | R/W | <p>Bus-Off Recovery Detect Flag*</p> <p>The BORIF bit is set to "1" when the CAN module recovers from the bus-off state normally by detecting 11 consecutive bits 128 times in the following conditions:</p> <p>When the BOM bit in the CiCTRL register is "00". When the BOM bit is "10". When the BOM bit is "11".</p> <p>The BORIF bit is not set to "1" if the CAN module recovers from the bus-off state in the following conditions:</p> <p>When the CANM bit in the CiCTRL register is set to "01" or "11" (CAN reset mode). When the RBOC bit in the CiCTRL register is set to "1" (forcible return from bus-off). When the BOM bit is "01". When the BOM bit is "11" and the CANM bit is set to "10" (CAN halt mode) before normal recovery occurs.</p> <p>0: No bus-off recovery detected 1: Bus-off recovery detected</p> |
| 3 | BOEIF | B'0 | R/W | <p>Bus-Off Entry Detect Flag*</p> <p>The BOEIF bit is set to "1" when the CAN error state becomes bus-off (the TEC value exceeds 255).</p> <p>The BOEIF bit is also set to "1" when the BOM bit in the CiCTRL register is "01" (entry to CAN halt mode automatically at bus-off entry) and the CAN module becomes the bus-off state.</p> <p>0: No bus-off entry detected 1: Bus-off entry detected</p> |
| 2 | EPIF | B'0 | R/W | <p>Error Passive Detect Flag*</p> <p>The EPIF bit is set to "1" when the CAN error state becomes error-passive (the REC or TEC value exceeds 127).</p> <p>The EPIF bit is set to "1" only when the REC or TEC initially exceeds 127. Thus, if "0" is written to the EPIF bit by a program while the REC or TEC remains greater than 127, the EPIF bit is not set to "1" until the REC and TEC goes below 127 and then exceeds 127 again.</p> <p>0: No error passive detected 1: Error passive detected</p> |

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|---|
| 1 | EWIF | B'0 | R/W | <p>Error Warning Detect Flag*</p> <p>The EWIF bit is set to "1" when the value of the receive error counter (REC) or transmit error counter (TEC) exceeds 95.</p> <p>The EWIF bit is set to "1" only when the REC or TEC initially exceeds 95. Thus, if "0" is written to the EWIF bit by a program while the REC or TEC remains greater than 95, the EWIF bit is not set to "1" until the REC and TEC goes below 95 and then exceeds 95 again.</p> <p>0: No error warning detected 1: Error warning detected</p> |
| 0 | BEIF | B'0 | R/W | <p>Bus Error Detect Flag*</p> <p>The BEIF bit is set to "1" when a bus error is detected.</p> <p>0: No bus error detected 1: Bus error detected</p> |

Note: * Only "0" may be written to this bit. (Writing "1" has no effect.) When writing "0" to specific bits in software, use the MOV instruction and write "0" to each bit to be cleared to "0" and "1" to all other bits.

Table 47.9 lists the behavior of bits BOEIF and BORIF according to BOM bit setting value.

Table 47.9 Behavior of Bits BOEIF and BORIF according to BOM Bit Setting Value

| BOM Bit | BOEIF Bit | BORIF Bit |
|----------------|---|--|
| B'00 | Set to "1" on entry to the bus-off state. | Set to "1" on exit from the bus-off state. |
| B'01 | | Do not set to "1". |
| B'10 | | Set to "1" on exit from the bus-off state. |
| B'11 | | Set to "1" if normal bus-off recovery occurs before the CANM bit is set to "10" (CAN halt mode). |

47.2.21 CANi Receive Error Count Register (CiRECR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CiRECR register indicates the value of the receive error counter.

For the increment/decrement conditions of the receive error counter, refer to the CAN Specifications (ISO11898-1).

The value in bus-off state is undefined.

CAN0 Receive Error Count Register (C0RECR)

CAN1 Receive Error Count Register (C1RECR)



<After Reset: H'00>

| Bit | Symbol | Initial Value | R/W | Description |
|--------|--------|---------------|-----|--|
| 7 to 0 | CiRECR | All 0 | R | Receive Error Count Function The CiRECR register increments or decrements the counter value according to error status of the CAN module during reception. |

47.2.22 CANi Transmit Error Count Register (CiTECR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

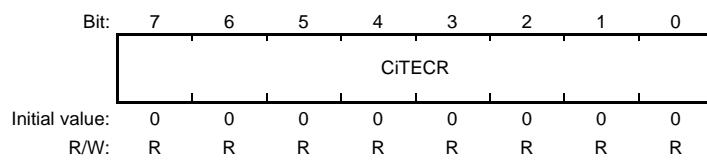
The CiTECR register indicates the value of the transmit error counter.

For the increment/decrement conditions of the transmit error counter, refer to the CAN Specifications (ISO11898-1).

The value in bus-off state is undefined.

CAN0 Transmit Error Count Register (C0TECR)

CAN1 Transmit Error Count Register (C1TECR)



<After Reset: H'00>

| Bit | Symbol | Initial Value | R/W | Description |
|--------|--------|---------------|-----|--|
| 7 to 0 | CiTECR | All 0 | R | Transmit Error Count Function The CiTECR register increments or decrements the counter value according to error status of the CAN module during transmission. |

47.2.23 CANi Error Code Store Register (CiECSR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CiECSR register can be used to monitor whether an error has occurred on the CAN bus. Refer to the CAN Specifications (ISO11898-1) to check the generation conditions of each error.

To set each bit except the EDPM bit to "0", write "0" by a program. If the timing at which each bit is set to "1" and the timing at which "0" is written by a program are the same, the relevant bit is set to "1".

CAN0 Error Code Store Register (C0ECSR)

CAN1 Error Code Store Register (C1ECSR)

| | | | | | | | | |
|----------------|------|------|------|------|-----|-----|-----|-----|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | EDPM | ADEF | BE0F | BE1F | CEF | AEF | FEF | SEF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

<After Reset: H'00>

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|--|
| 7 | EDPM | B'0 | R/W | <p>Error Display Mode Select Bit*1*2</p> <p>The EDPM bit selects the output mode of the CiECSR register.</p> <p>When the EDPM bit is set to "0", the CiECSR register outputs the first error code.</p> <p>When the EDPM bit is set to "1", the CiECSR register outputs the accumulated error code.</p> <p>0: Output of first detected error code 1: Output of accumulated error code</p> |
| 6 | ADEF | B'0 | R/W | <p>ACK Delimiter Error Flag*3*4</p> <p>The ADEF bit is set to "1" when a form error is detected with the ACK delimiter during transmission.</p> <p>0: No ACK delimiter error detected 1: ACK delimiter error detected</p> |
| 5 | BE0F | B'0 | R/W | <p>Bit Error (dominant) Flag*3*4</p> <p>The BE0F bit is set to "1" when a dominant bit error is detected.</p> <p>0: No bit error (dominant) detected 1: Bit error (dominant) detected</p> |
| 4 | BE1F | B'0 | R/W | <p>Bit Error (recessive) Flag*3*4</p> <p>The BE1F bit is set to "1" when a recessive bit error is detected.</p> <p>0: No bit error (recessive) detected 1: Bit error (recessive) detected</p> |
| 3 | CEF | B'0 | R/W | <p>CRC Error Flag*3*4</p> <p>The CEF bit is set to "1" when a CRC error is detected.</p> <p>0: No CRC error detected 1: CRC error detected</p> |

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|--|
| 2 | AEF | B'0 | R/W | ACK Error Flag*3*4 The AEF bit is set to "1" when an ACK error is detected. 0: No ACK error detected 1: ACK error detected |
| 1 | FEF | B'0 | R/W | Form Error Flag*3*4 The FEF bit is set to "1" when a form error is detected. 0: No form error detected 1: Form error detected |
| 0 | SEF | B'0 | R/W | Stuff Error Flag*3*4 The SEF bit is set to "1" when a stuff error is detected. 0: No stuff error detected 1: Stuff error detected |

- Notes:
1. Write to the EDPM bit in CAN reset mode or CAN halt mode.
 2. If more than one error condition is detected simultaneously, all related bits are set to "1".
 3. Writing "1" has no effect to these bit values.
 4. When writing "0" to bits SEF, FEF, AEF, CEF, BE1F, BE0F, and ADEF by a program, use the MOV instruction to ensure that only the specified bit is set to "0" and the other bits are set to "1".

47.2.24 CANi Time Stamp Register (CiTSR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

When the CiTSR register is read, the value of the time stamp counter (16-bit free-running counter) at that moment is read.

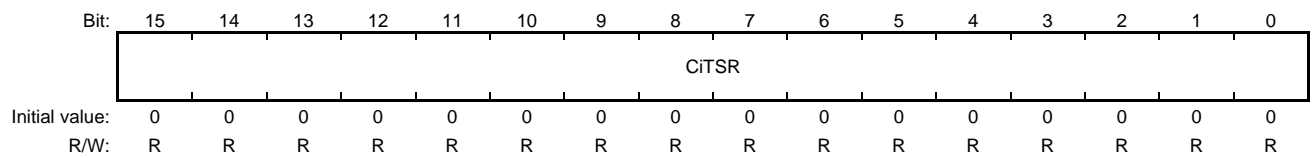
The value of the time stamp counter reference clock is a multiple of 1 bit time, as configured by the TSPS bit in the CiCTRL register.

The time stamp counter stops in CAN sleep mode and CAN halt mode, and is initialized in CAN reset mode.

The time stamp counter value is stored to TSL and TSH in the CiMBj register when a received message is stored in a receive mailbox.

CAN0 Time Stamp Register (C0TSR)

CAN1 Time Stamp Register (C1TSR)



<After Reset: H'0000>

| Bit | Symbol | Initial Value | R/W | Description |
|---------|--------|---------------|-----|--|
| 15 to 0 | CiTSR | All 0 | R | Free-running counter value for the time stamp function |

Note: Read the CiTSR register in 16-bit units.

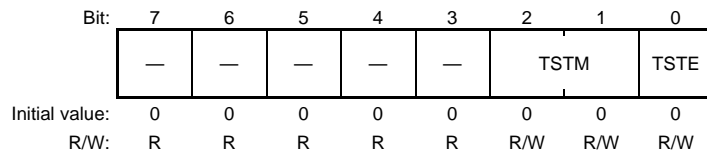
47.2.25 CANi Test Control Register (CiTCR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Write to the CiTCR register in CAN halt mode only.

CAN0 Test Control Register (C0TCR)

CAN1 Test Control Register (C1TCR)



<After Reset: H'00>

| Bit | Symbol | Initial Value | R/W | Description |
|--------|--------|---------------|-----|--|
| 7 to 3 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 2, 1 | TSTM | All 0 | R/W | CAN Test Mode Select Bits The TSTM bit selects the CAN test mode. For details on the CAN test modes, see section 47.2.25 (1), Listen-Only Mode, section 47.2.25 (2), Self-Test Mode 0 (External Loop Back), and section 47.2.25 (3), Self-Test Mode 1 (Internal Loop Back). B'00: Other than CAN test mode B'01: Listen-only mode B'10: Self-test mode 0 (external loop back) B'11: Self-test mode 1 (internal loop back) |
| 0 | TSTE | B'0 | R/W | CAN Test Mode Enable Bit When the TSTE bit is set to "0", CAN test mode is disabled. When the TSTE bit is set to "1", CAN test mode is enabled. 0: CAN test mode disabled 1: CAN test mode enabled |

(1) Listen-Only Mode

The ISO 11898-1 recommends an optional bus monitoring mode. In listen-only mode, the CAN node is able to receive valid data frames and valid remote frames. It sends only recessive bits on the CAN bus, and the protocol controller is not required to send the ACK bit, overload flag, or active error flag.

Listen-only mode can be used for baud rate detection.

Do not request transmission from any mailboxes in this mode.

Figure 47.6 shows the connection when listen-only mode is selected.

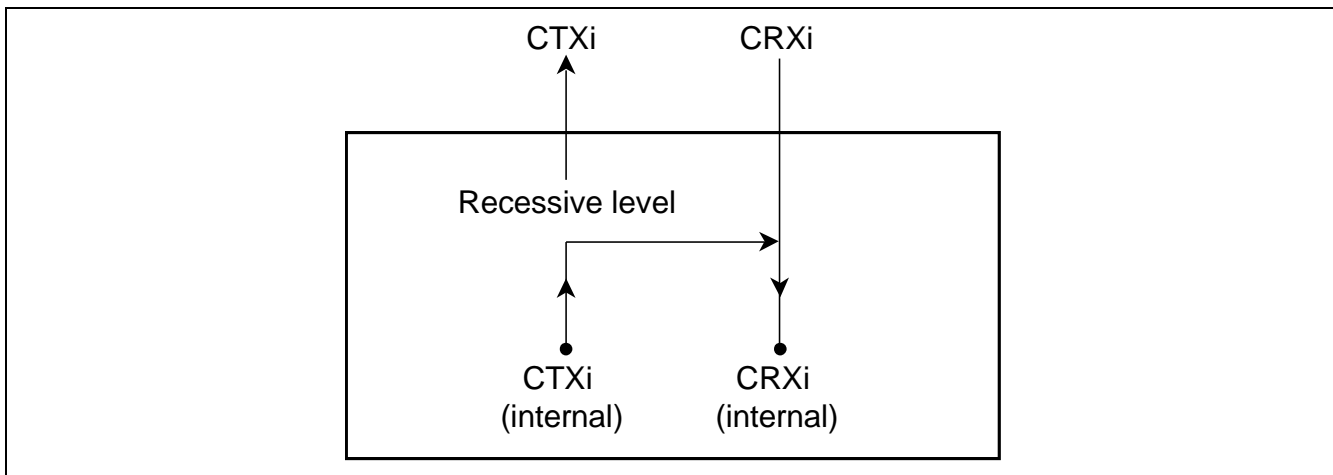


Figure 47.6 Connection when Listen-Only Mode is Selected

(2) Self-Test Mode 0 (External Loop Back)

Self-test mode 0 is provided for CAN transceiver tests.

In this mode, the protocol controller treats its own transmitted messages as messages received via the CAN transceiver and stores them into the receive mailbox. To be independent from external stimulation, the protocol controller generates the ACK bit.

Connect the CANi_TX/CANi_RX pins to the transceiver.

Figure 47.7 shows the connection when self-test mode 0 is selected.

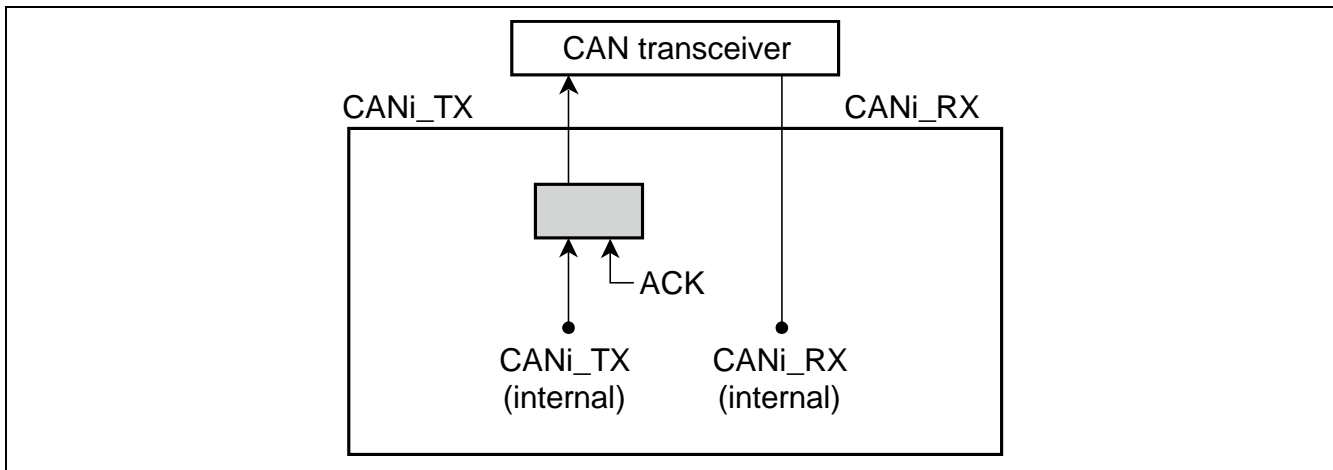


Figure 47.7 Connection when Self-Test Mode 0 is Selected

(3) Self-Test Mode 1 (Internal Loop Back)

Self-test mode 1 is provided for self-test functions.

In this mode, the protocol controller treats its transmitted messages as received messages and stores them into the receive mailbox. To be independent from external stimulation, the protocol controller generates the ACK bit.

In self-test mode 1, the protocol controller performs an internal feedback from the internal CANi_TX pin to the internal CANi_RX pin. The input value of the external CANi_RX pin is ignored. The external CANi_TX pin outputs only recessive bits. The CANi_TX/CANi_RX pins do not need to be connected to the CAN bus or any external device.

Figure 47.8 shows the connection when self-test mode 1 is selected.

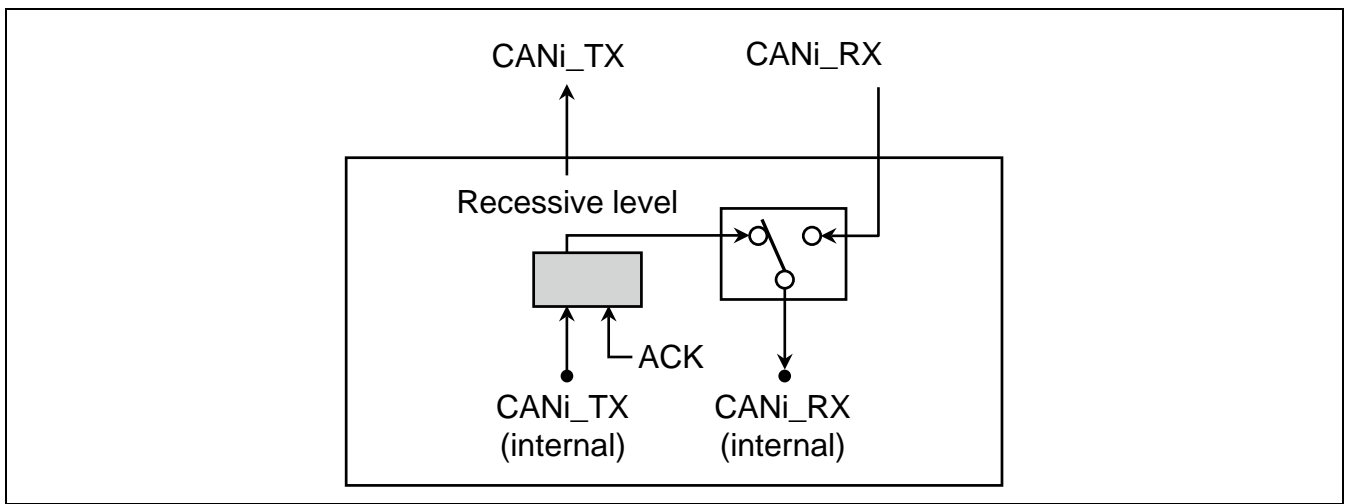


Figure 47.8 Connection when Self-Test Mode 1 is Selected

47.2.26 CANi Interrupt Status Register (CiISR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CiISR register shows interrupt sources before masking by the CiIER register.

CAN0 Interrupt Status Register (C0ISR)

CAN1 Interrupt Status Register (C1ISR)

| | | | | | | | | |
|----------------|---|---|------|------|------|-----------|-------|------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | ERSF | RXFF | TXFF | RX M0F | RXM1F | TXMF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R | R | R |

<After Reset: H'00>

| Bit | Symbol | Initial Value | R/W | Description |
|------|--------|---------------|-----|---|
| 7, 6 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 5 | ERSF | B'0 | R | Error (ERS) Interrupt Status Bit* ¹ The ERSF bit shows the error interrupt source status. 0: ERS interrupt source not detected 1: ERS interrupt source detected |
| 4 | RXFF | B'0 | R/W | Reception FIFO (RXF) Interrupt Status Bit* ² The RXFF bit shows the FIFO receive interrupt source status. 0: RXF interrupt source not detected 1: RXF interrupt source detected The RXFF is cleared to "0" by writing "0" to it by software. (writing "1" has no effect) |
| 3 | TXFF | B'0 | R/W | Transmission FIFO (TXF) Interrupt Status Bit* ³ The TXFF bit shows the FIFO transmit interrupt source status. 0: TXF interrupt source not detected 1: TXF interrupt source detected The TXFF is cleared to "0" by writing "0" to it by software. (writing "1" has no effect) |
| 2 | RXM0F | B'0 | R | Mailbox 0 Successful Reception (RXM0) Interrupt Status Bit* ⁴ The RXM0F bit shows the successful reception interrupt source status for mailbox 0. 0: RXM0 interrupt source not detected 1: RXM0 interrupt source detected |
| 1 | RXM1F | B'0 | R | Mailbox 1 to 63 Successful Reception (RXM1) Interrupt Status Bit* ⁵ The RXM1F bit shows the successful reception interrupt source status for mailboxes 1 to 63. 0: RXM1 interrupt source not detected 1: RXM1 interrupt source detected |

| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|---|
| 0 | TXMF | B'0 | R | Mailbox 32 to 63 Successful Transmission (TXM) Interrupt Status Bit* ⁶ The TXMF bit shows the successful transmission interrupt source status for mailboxes 32 to 63. 0: TXM interrupt source not detected 1: TXM interrupt source detected |

- Notes:
1. The ERSF bit is set to "1" when a bit in one of the CiEIFR[j] registers is set to "1" due to a communication error while the corresponding bit in the CiEIER[j] register is set to "1" (j = 0 to 7).
 2. The RXFF bit is set to "1" when bit 6 or 5 in the CiRFCR register is set to "1" because of a reception FIFO full or warning condition due to the setting of CiMIER[61].
 3. The TXFF bit is set to "1" when the transmission FIFO message count reaches the specified value due to the setting of CiMIER[57].
 4. After the NEWDATA bit in CiMCTL0 is set to "1" at the end of a receive operation, the RXM0F bit is set to "1" if storing of the receive data is complete (corresponding INVALIDDATA bit value changed from "1" to "0") and the CiMIER0[0] bit has been set to "1".
 5. After the NEWDATA bit in CiMCTLj is set to "1" at the end of a receive operation, the RXM1F bit is set to "1" if storing of the receive data is complete (corresponding INVALIDDATA bit value changed from "1" to "0") and the bit in CiMIER0 or CiMIER1 corresponding to mailbox j has been set to "1" (j = 1 to 63).
 6. The TXMF bit is set to "1" when the bit in the CiMIER1 register corresponding to mailbox j is set to "1" while the value of the SENTDATA bit in the CiMCTLj register is "1" following a successful reception (j = 32 to 63).

47.2.27 CANi Interrupt Enable Register (CiIER) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The CiIER register can be used by an application to cause some interrupts to be ignored while processing by an interrupt service routine is taking place. Each bit affects the individual interrupt source corresponding to it.

CAN0 Interrupt Enable Register (C0IER)

CAN1 Interrupt Enable Register (C1IER)

| | | | | | | | | |
|----------------|---|---|-------|-------|-------|--------|--------|-------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | ERSIE | RXFIE | TXFIE | RXM0IE | RXM1IE | TXMIE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

<After Reset: H'00>

| Bit | Symbol | Initial Value | R/W | Description |
|------|--------|---------------|-----|--|
| 7, 6 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 5 | ERSIE | B'0 | R/W | Error (ERS) Interrupt Enable Bit The ERSIE bit enables or disables the ERS interrupt controller. 0: ERS interrupt disabled 1: ERS interrupt enabled |
| 4 | RXFIE | B'0 | R/W | Reception FIFO (RXF) Interrupt Enable Bit The RXFIE bit enables or disables the RXF interrupt controller. 0: RXF interrupt disabled 1: RXF interrupt enabled |
| 3 | TXFIE | B'0 | R/W | Transmission FIFO (TXF) Interrupt Enable Bit The TXFIE bit enables or disables the TXF interrupt controller. 0: TXF interrupt disabled 1: TXF interrupt enabled |
| 2 | RXM0IE | B'0 | R/W | Mailbox 0 Successful Reception (RXM0) Interrupt Enable Bit The RXM0IE bit enables or disables the RXM0 interrupt controller. 0: RXM0 interrupt disabled 1: RXM0 interrupt enabled |
| 1 | RXM1IE | B'0 | R/W | Mailbox 1 to 63 Successful Reception (RXM1) Interrupt Enable Bit The RXM1IE bit enables or disables the RXM1 interrupt controller. 0: RXM1 interrupt disabled 1: RXM1 interrupt enabled |
| 0 | TXMIE | B'0 | R/W | Mailbox 32 to 63 Successful Transmission (TXM) Interrupt Enable Bit The TXMIE bit enables or disables the TXM interrupt controller. 0: TXM interrupt disabled 1: TXM interrupt enabled |

47.2.28 CANi Mailbox Search Mask Register (CiMBSMR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Write to the CiMBSMR register in CAN halt mode only.

CAN0 Mailbox Search Mask Register (COMBSMR)

CAN1 Mailbox Search Mask Register (C1MBSMR)

| | | | | | | | | |
|----------------|---|---|---|---|---|---|---|-----------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | MBO SM |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W |

<After Reset: H'00>

| Bit | Symbol | Initial Value | R/W | Description |
|--------|--------|---------------|-----|---|
| 7 to 1 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 0 | MB0SM | B'0 | R/W | Mailbox 0 Search Mask Bit* When the MB0SM bit is set to "1", message box 0 is excluded from the search target for the CANi mailbox search status register. |

Note: * The MB0SM bit is enabled in the search modes except channel search mode. In the RXM1 interrupt handling, this bit is usable to exclude message box 0 from the search target for the CiMSSR register in receive mailbox search mode.

47.2.29 CANi Parity Error Control Register (CiPECR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register controls the function of the parity mode and displays a parity error flag

| | | | | | | | | |
|----------------|-----|---|---|---|---|-----|-----|-----|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PF | — | — | — | — | PIE | PME | PM |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R/W | R/W | R/W |

<After Reset: H'00>

| Bit | Symbol | Initial Value | R/W | Description |
|--------|--------|---------------|-----|---|
| 7 | PF | B'0 | R/W | Parity error flag *2 This bit is set to "1" when a parity error is detected. The bit has to be cleared by the program. As long as this bit is set no further parity error can be detected and the address value in the Parity Error Address Capture Register keeps the address for the first captured parity error. 0: Parity error is not detected 1: Parity error is detected |
| 6 to 3 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 2 | PIE | B'0 | R/W | Parity error interrupt *1 This bit controls the parity error interrupt enable. When this bit is configured to "0", the parity error interrupt is disabled. When this bit is set to "1", the parity error interrupt is enabled. When a parity error interrupt is generated, the parity error flag (CANi Parity Error Control Register, bit 7) should be checked to determine the interrupt source. 0: Parity error interrupt is disabled 1: Parity error interrupt is enabled |
| 1 | PME | B'0 | R/W | Parity enable bit This bit controls the Dual Port RAM parity functionality. When this bit is set to "0", parity checkers are disabled keeping the parity generators still active to store the correct parity bit for each byte when writing to the Dual Port RAM. When this bit is set to "1", parity checkers and parity generators are both active. 0: Parity checker off 1: Parity checker on |

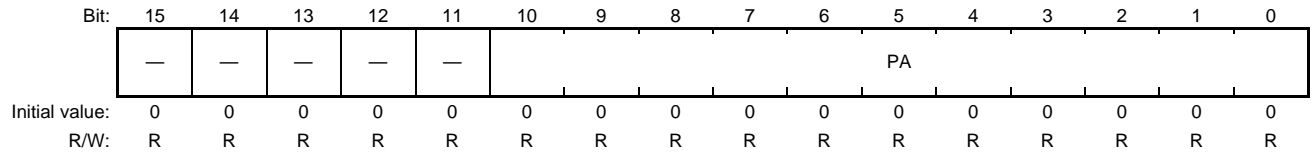
| Bit | Symbol | Initial Value | R/W | Description |
|-----|--------|---------------|-----|--|
| 0 | PM | B'0 | R/W | <p>Parity mode</p> <p>This bit controls the action of the CAN Module following the detection of a parity error.</p> <p>When this bit is programmed to “1” and a parity error is detected while in Operation Mode the CAN module forces the cancellation of a possible pending communication (either reception or transmission) and then enters Halt mode. When the pending communication is cancelled, the Reception Error Counter and Transmission Error Counter are initialized. If a parity error is detected while the CAN module is in its Reset or Halt mode no state transition will take place.</p> <p>In the case where CPU requests transition to Reset Mode at the same time as a parity error occurs then the CPU request to enter Reset Mode will have the highest priority.</p> <p>When this bit is set to “0”, CAN module continues in Operation Mode and the pending communication is not canceled.</p> <p>Regardless of the setting of ParMode (PM) when a parity error occurs and ParModeEn (PME) bit is set to ‘1’ (parity function enabled):</p> <ul style="list-style-type: none"> • A parity error flag is set in bit 7 of the CAN Parity Error Control Register • A parity error interrupt is generated (providing the ParIntEn (PIE) bit is asserted) • The address where a parity error occurs is captured in the CANi Parity Error Capture Address Register. <p>When parity error is detected</p> <p>0: Continue operation mode</p> <p>1: Go to Halt mode</p> |

- Notes:
1. Do not write to this bit in Operation Mode
 2. Write “0” only (writing “1” has no effect)
 3. The software sequence required to set these bits is described in section 47.9, CPU is prevented from writing this bit in operation mode by H/W write protection.

47.2.30 CANi Parity Error Address Capture Register (CiPEACR) (i = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register displays the address where the parity error occurred.



<After Reset: H'0000>

| Bit | Symbol | Initial Value | R/W | Description |
|----------|--------|---------------|-----|---|
| 15 to 11 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |

| | | | | |
|---------|----|-------|---|---|
| 10 to 0 | PA | All 0 | R | Parity error captured address bits This register displays the address offset from a starting address of Mailbox 0 where parity error is detected. When a parity error is detected, the address in which the error occurred is captured in this register. Only the longword address is shown. The address is shown as below. |
|---------|----|-------|---|---|

| ParAddress value | Location |
|------------------|--------------------|
| H'000 - H'00F | Mailbox 0* |
| H'010 - H'01F | Mailbox 1 |
| H'020 - H'02F | Mailbox 2 |
| ... | ... |
| H'3F0 - H'3FF | Mailbox 63 |
| H'400 - H'41F | CiMKR 2 to 9 |
| H'420 - H'427 | FIFO ID 0 to 1 |
| H'428 - H'42B | Mask Invalid 1 |
| H'42C - H'42F | Interrupt enable 1 |
| H'430 - H'433 | CiMKR 0 |
| H'434 - H'437 | CiMKR 1 |
| H'438 - H'43B | Mask Invalid 0 |
| H'43C - H'43F | Interrupt enable 0 |

Note: * H'000 is the first byte of the mailbox 0.

Only the address related to the first parity error is reported. The program has to clear the ParFlag status to enable the capture of other locations leading to a parity error.

In case parity error is flagged at the same time from DPRAM CAN side parity checker and the DPRAM CPU side parity checker, the CPU side access address will be displayed in the parity error address capture register.

47.3 Operating Mode

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The CAN module has the following four operating modes.

- CAN reset mode
- CAN halt mode
- CAN operation mode
- CAN sleep mode

Figure 47.9 shows the transition between CAN operating modes.

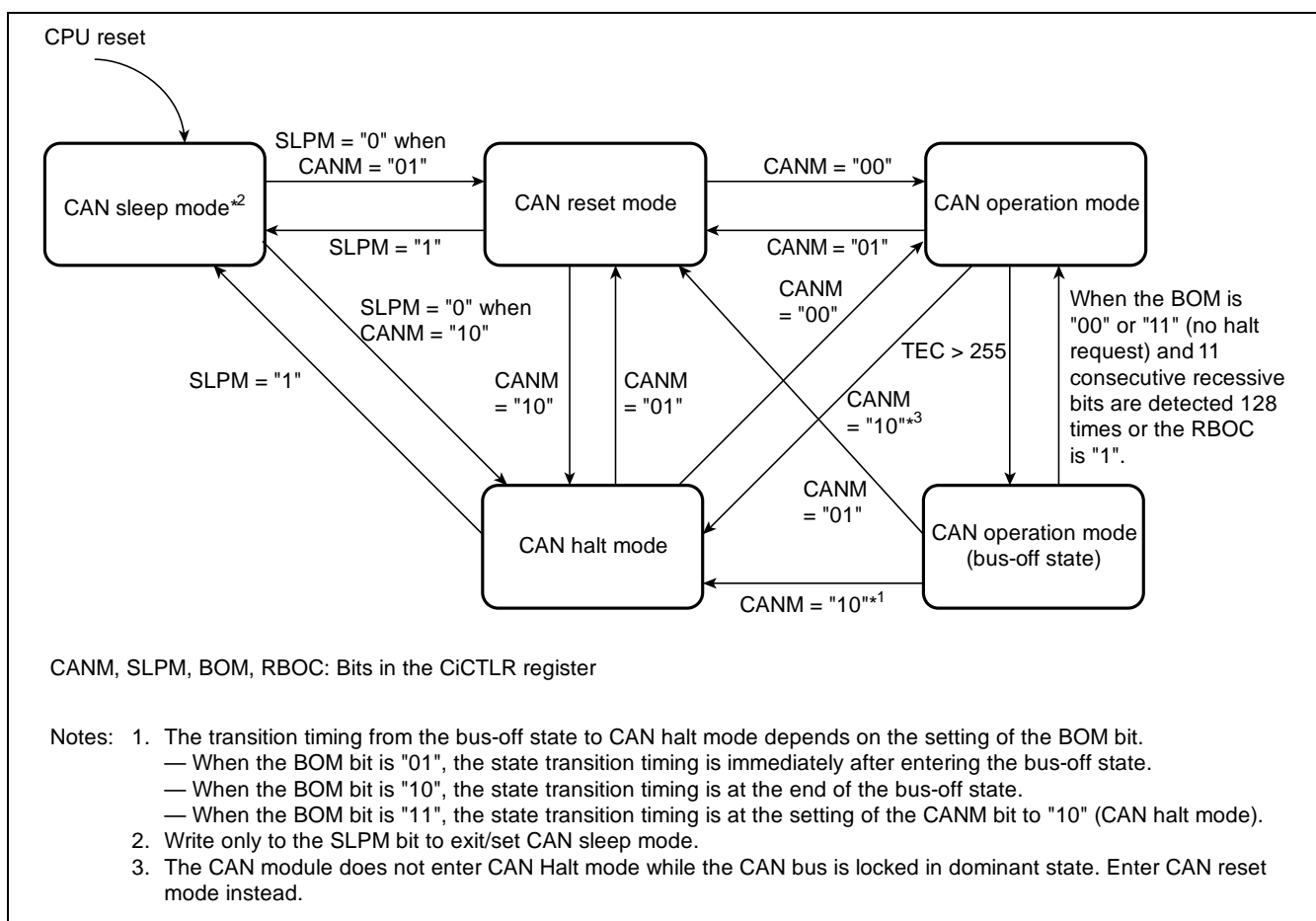


Figure 47.9 Transition between CAN Operating Modes (i = 0, 1)

47.3.1 CAN Reset Mode

CAN reset mode is provided for CAN communication configuration.

When the CANM bit in the CiCTLR register is set to "01" or "11", the CAN module enters CAN reset mode. Then the RSTST bit in the CiSTR register is set to "1". Do not change the CANM bit until the RSTST bit is set to "1". Configure the CiBCR register before exiting CAN reset mode to any other modes.

The following registers are initialized to their reset values after entering CAN reset mode and their initialized values are retained during CAN reset mode:

- CiMCTLj register
- CiSTR register (except bits SLPST and TFST)
- CiEIFR register
- CiRECR register
- CiTECR register
- CiTSR register
- CiMSSR register
- CiMSMR register
- CiRFCR register
- CiTFCR register
- CiTCR register
- CiECSR register (except EDPM bit)
- CiISR register
- CiMBSMR register

The following registers retain their values after entering CAN reset mode.

- CiCLKR register
- CiCTLR register
- CiSTR register (bits SLPST and TFST)
- Registers CiMIER0 and CiMIER1
- CiEIER register
- CiBCR register
- CiCSSR register
- CiECSR register (EDPM bit only)
- CiMBj register
- Registers CiMKR0 to CiMKR9
- Registers CiFIDCR0 and CiFIDCR1
- Registers CiMKIVLR0 and CiMKIVLR1
- CiAFSR register
- CiRFPCR register
- CiTFPCR register
- CiIER register

47.3.2 CAN Halt Mode

CAN halt mode is used for mailbox configuration and test mode setting.

When the CANM bit in the CiCTRL register is set to "10", CAN halt mode is selected. Then the HLTST bit in the CiSTR register is set to "1". Do not change the CANM bit until the HLTST bit is set to "1".

Refer to Table 47.10, Operation in CAN Reset Mode and CAN Halt Mode regarding the state transition conditions when transmitting or receiving.

All registers except bits RSTST, HLTST, and SLPST in the CiSTR register remain unchanged when the CAN module enters CAN halt mode.

Do not change registers CiCLKR, CiCTRL (except bits CANM and SLPM), and CiEIER in CAN halt mode. The CiBCR register can be changed in CAN halt mode only when listen-only mode is selected to use for automatic bit rate detection.

Table 47.10 Operation in CAN Reset Mode and CAN Halt Mode

| Mode | Receiver | Transmitter | Bus-Off |
|----------------|---|---|--|
| CAN reset mode | CAN module enters CAN reset mode without waiting for the end of message reception | CAN module enters CAN reset mode after waiting for the end of message transmission (1, 4) | CAN module enters CAN reset mode without waiting for the end of bus-off recovery |
| CAN halt mode | CAN module enters CAN halt mode after waiting for the end of message reception (2, 3) | CAN module enters CAN halt mode after waiting for the end of message transmission (1, 2, 4) | <ul style="list-style-type: none"> When the BOM bit is "B'00" A halt request from a program will be acknowledged only after bus-off recovery When the BOM bit is "B'01" CAN module automatically enters CAN halt mode without waiting for the end of bus-off recovery (regardless of a halt request from a program) When the BOM bit is "B'10" CAN module automatically enters CAN halt mode after waiting for the end of bus-off recovery (regardless of a halt request from a program) When the BOM bit is "B'11" CAN module enters CAN halt mode (without waiting for the end of bus-off recovery) if a halt is requested by a program during bus-off |

BOM bit: Bit in the CiCTRL register (i = 0, 1)

- Notes:
- If several messages are requested to be transmitted, mode transition occurs after the completion of the first message transmission. When CAN reset mode is being requested during suspend transmission, mode transition occurs when the bus is idle, the next transmission ends, or the CAN module becomes a receiver.
 - If the CAN bus is locked in dominant state, the program can detect this state by monitoring the BLIF bit in the CiEIFR register. The CAN module does not enter CAN Halt mode while the CAN bus is locked in dominant state. Enter CAN reset mode instead.
 - If a CAN bus error occurs during reception after CAN halt mode is requested, the CAN module enters CAN halt mode. However, the CAN module does not enter CAN Halt mode when the CAN bus is locked in dominant state.
 - If a CAN bus error or arbitration lost occurs during transmission after CAN reset mode or CAN halt mode is requested, the CAN module enters the requested operating mode. However, the CAN module does not enter CAN Halt mode when the CAN bus is locked in dominant state.

47.3.3 CAN Sleep Mode

CAN sleep mode is used for reducing current consumption by stopping the clock supply to the CAN module. After MCU hardware reset or software reset, the CAN module starts from CAN sleep mode.

When the SLPM bit in the CiCTLR register is set to "1", the CAN module enters CAN sleep mode. Then the SLPST bit in the CiSTR register is set to "1". Do not change the value of the SLPM bit until the SLPST bit is set to "1". The other registers remain unchanged when the MCU enters CAN sleep mode.

Write to the SLPM bit in CAN reset mode and CAN halt mode. Do not change any registers (except the SLPM bit) during CAN sleep mode. Read operation is still allowed. When the SLPM bit is set to "0", the CAN module is released from CAN sleep mode. When the CAN module exits CAN sleep mode, the other registers remain unchanged.

47.3.4 CAN Operation Mode (Excluding Bus-Off State)

CAN operation mode is used for CAN communication.

When the CANM bit in the CiCTLR register is set to "B'00", the CAN module enters CAN operation mode.

Then bits RSTST and HLTST in the CiSTR register are set to "0". Do not change the value of the CANM bit until these bits are set to "0".

If 11 consecutive recessive bits are detected after entering CAN operation mode, the CAN module is in the following states:

- The CAN module becomes an active node on the network that enables transmission and reception of CAN messages.
- Error monitoring of the CAN bus, such as receive and transmit error counters, is performed.

During CAN operation mode, the CAN module may be in one of the following three sub-modes, depending on the status of the CAN bus:

- Idle mode: Transmission or reception is not being performed.
- Receive mode: A CAN message sent by another node is being received.
- Transmit mode: A CAN message is being transmitted. The CAN module may receive its own message simultaneously when self-test mode 0 (TSTM bit in the CiTCR register = "B'10") or self-test mode 1 (TSTM bit = "B'11") is selected.

Figure 47.10 shows the sub mode in CAN operation mode.

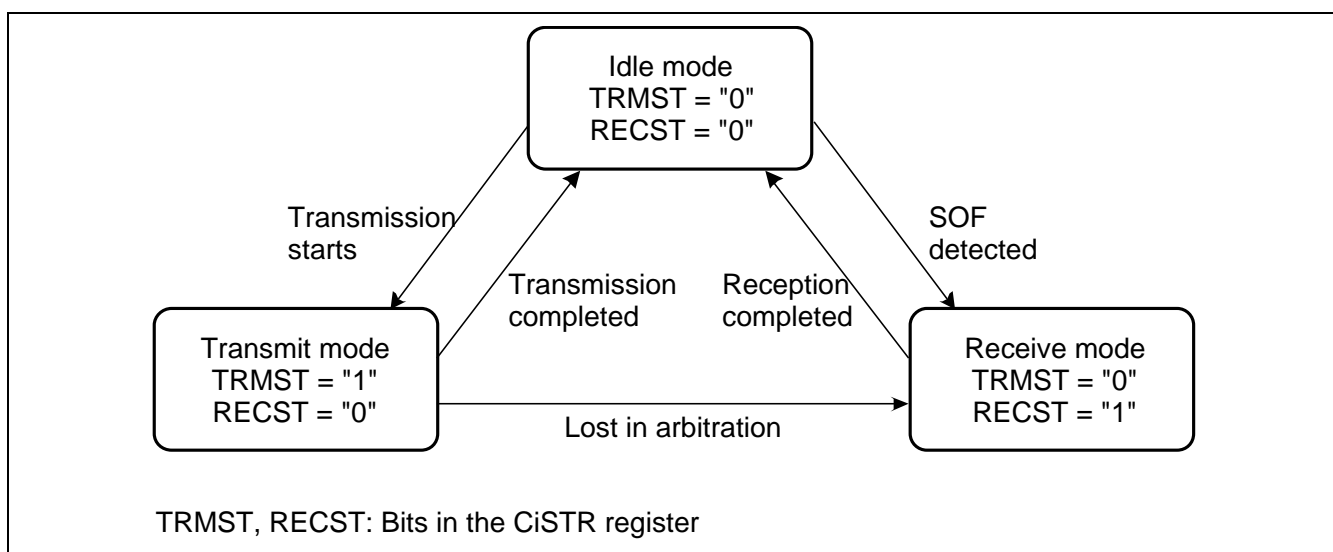


Figure 47.10 Sub Mode in CAN Operation Mode (i = 0, 1)

47.3.5 CAN Operation Mode (Bus-Off State)

The CAN module enters the bus-off state according to the increment/decrement rules for the transmit/error counters in the CAN Specifications.

The following cases apply when recovering from the bus-off state. When the CAN module is in bus-off state, the values of the associated registers, except registers CiSTR, CiEIFR, CiRECR, CiTECR and CiTSR, remain unchanged.

1. When the BOM bit in the CiCTLR register is "B'00" (normal mode)

The CAN module enters the error-active state after it has completed the recovery from the bus-off state and CAN communication is enabled instantly. The BORIF bit in the CiEIFR register is set to "1" (bus-off recovery detected) at this time.

2. When the RBOC bit in the CiCTLR register is set to "1" (forcible return from bus-off)

The CAN module enters the error-active state when it is in bus-off state and the RBOC bit is set to "1". CAN communication is enabled again after 11 consecutive recessive bits are detected. The BORIF bit is not set to "1" at this time.

3. When the BOM bit is "B'01" (entry to CAN halt mode automatically at bus-off entry)

The CAN module enters CAN halt mode when it reaches the bus-off state. The BORIF bit is not set to "1" at this time.

4. When the BOM bit is "B'10" (entry to CAN halt mode automatically at bus-off end)

The CAN module enters CAN halt mode when it has completed the recovery from bus-off. The BORIF bit is set to "1" at this time.

5. When the BOM bit is "B'11" (entry to CAN halt mode by a program) and the CANM bit in the CiCTLR register is set to "B'10" (CAN halt mode) during the bus-off state

The CAN module enters CAN halt mode when it is in bus-off state and the CANM bit is set to "B'10" (CAN halt mode). The BORIF bit is not set to "1" at this time.

If the CANM bit is not set to "B'10" during bus-off, the same behavior as (1) applies.

47.4 CAN Communication Speed Configuration

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The following description explains about the CAN communication speed configuration.

47.4.1 CAN Clock Configuration

This MCU has a CAN clock selector. The CAN clock can be configured by setting the CCLKS bit in the CiCLKR register and the BRP bit in the CiBCR register.

Figure 47.11 shows the block diagram of CAN clock generator.

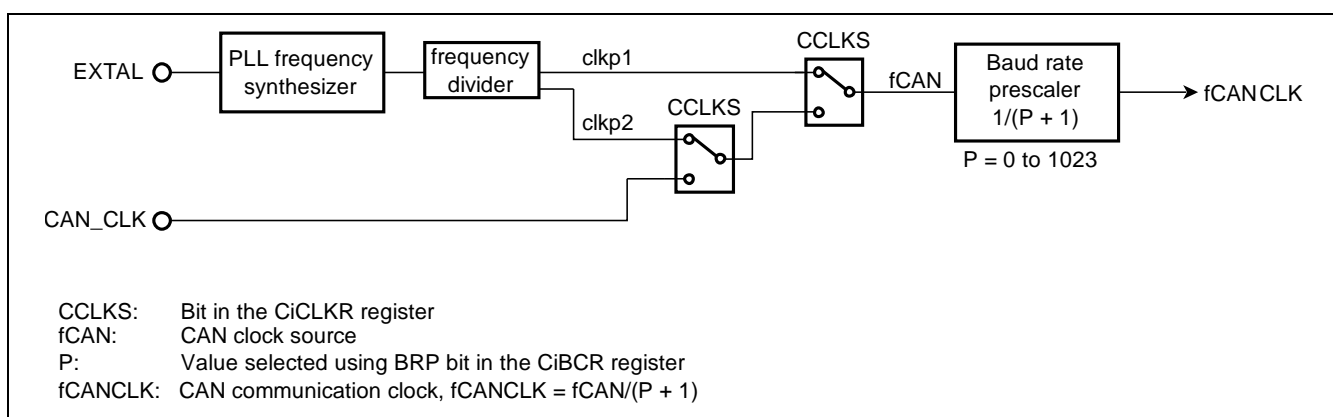


Figure 47.11 Block Diagram of CAN Clock Generator (i = 0, 1)

The CAN operating clock can be selected from the internal clock (clkp1, clkp2) from the CPG (PLL), and the external clock input to external pin (CAN_CLK).

In the initial state, clkp1 is selected. If an external clock input is to be selected, set the CiCLKR register (section 47.2.2) before accessing the CAN module. The range of frequencies that may be input as an external clock is 8 to 50 MHz.

47.4.2 Bit Timing Configuration

The bit time is a single bit time for transmitting/receiving a message and consists of the following three segments.

Figure 47.12 shows the bit timing.

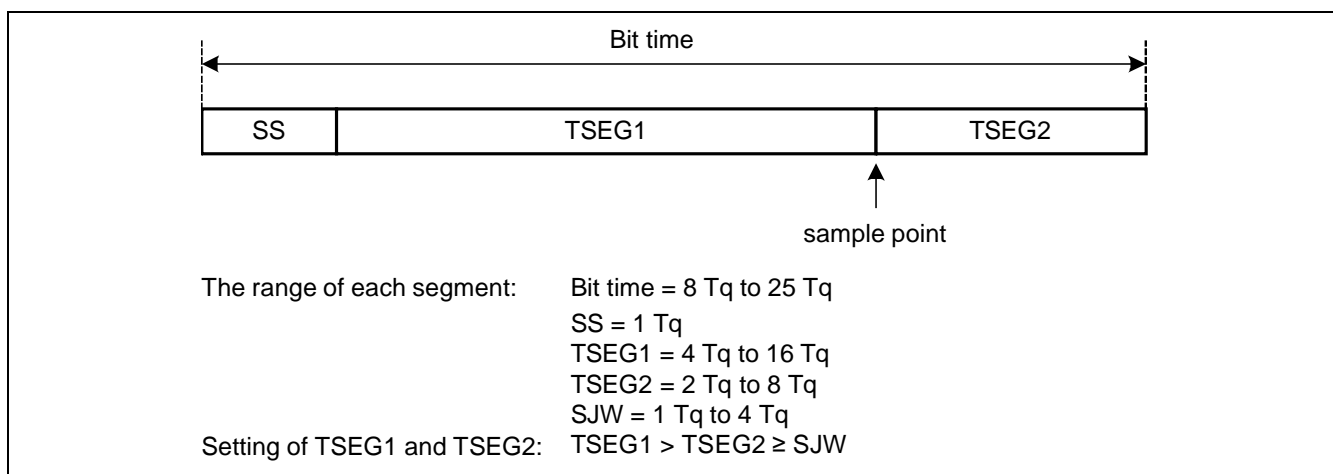


Figure 47.12 Bit Timing

47.4.3 Bit-rate

The bit rate depends on the division value of the fCAN (CAN clock source), the division value of the baud rate prescaler, and the number of Tq of one bit time.

$$\text{Bit rate [bps]} = \frac{f_{CAN}}{\text{Baud rate prescaler division value}^* \times \text{number of Tq of one bit time}} = \frac{f_{CANCLK}}{\text{Number of Tq of one bit time}}$$

Note: * Division value of the baud rate prescaler = P + 1 (P: 0 to 1023)

P: Setting value of the BRP bit in the CiBCR register (i = 0, 1)

Table 47.11 lists bit rate examples.

Table 47.11 Example of Bit-rate

| fCAN | 40 MHz | | 32 MHz | | 20 MHz | | 16 MHz | |
|-----------|-----------|-------|-----------|-------|-----------|-------|-----------|-------|
| | No. of Tq | P + 1 | No. of Tq | P + 1 | No. of Tq | P + 1 | No. of Tq | P + 1 |
| 1 Mbps | 10Tq | 4 | 8Tq | 4 | 10Tq | 2 | 8Tq | 2 |
| | 20Tq | 2 | 16Tq | 2 | 20Tq | 1 | 16Tq | 1 |
| 500 kbps | 10Tq | 8 | 8Tq | 8 | 10Tq | 4 | 8Tq | 4 |
| | 20Tq | 4 | 16Tq | 4 | 20Tq | 2 | 16Tq | 2 |
| 250 kbps | 10Tq | 16 | 8Tq | 16 | 10Tq | 8 | 8Tq | 8 |
| | 20Tq | 8 | 16Tq | 8 | 20Tq | 4 | 16Tq | 4 |
| 83.3 kbps | 8Tq | 60 | 8Tq | 48 | 8Tq | 30 | 8Tq | 24 |
| | 10Tq | 48 | 16Tq | 24 | 10Tq | 24 | 16Tq | 12 |
| | 16Tq | 30 | | | 16Tq | 15 | | |
| | 20Tq | 24 | | | 20Tq | 12 | | |
| 33.3 kbps | 8Tq | 150 | 8Tq | 120 | 8Tq | 75 | 8Tq | 60 |
| | 10Tq | 120 | 10Tq | 96 | 10Tq | 60 | 10Tq | 48 |
| | 20Tq | 60 | 16Tq | 60 | 20Tq | 30 | 16Tq | 30 |
| | | | 20Tq | 48 | | | 20Tq | 24 |

47.5 Mailbox and Mask Register Structure

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Figure 47.13 shows the structure of the CiMBj register.

There are 64 mailboxes with the same structure.

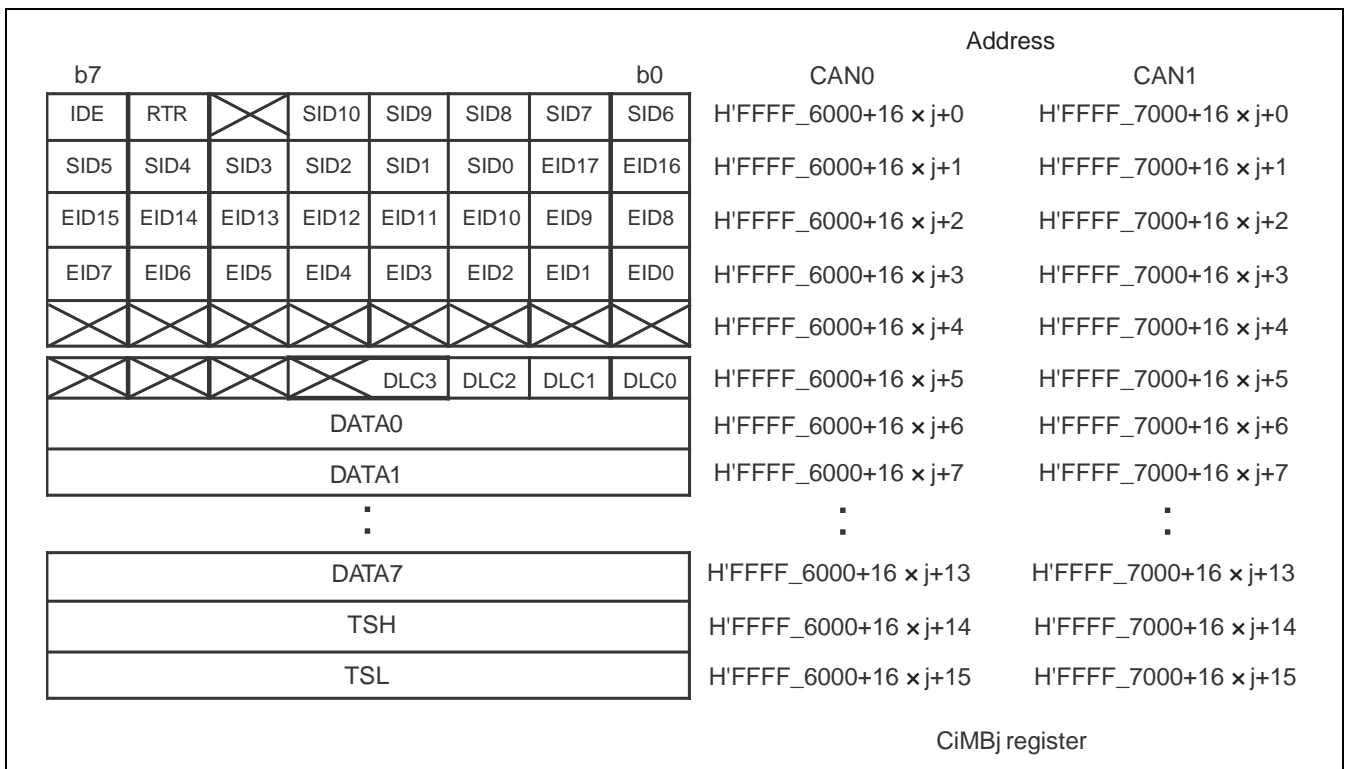


Figure 47.13 Structure of CiMBj Register (i = 0, 1; j = 0 to 63)

Figure 47.14 shows the structure of registers CiMKR0, CiMKR1, and CiMKR2 to CiMKR9.

There are 10 mask registers with the same structure.

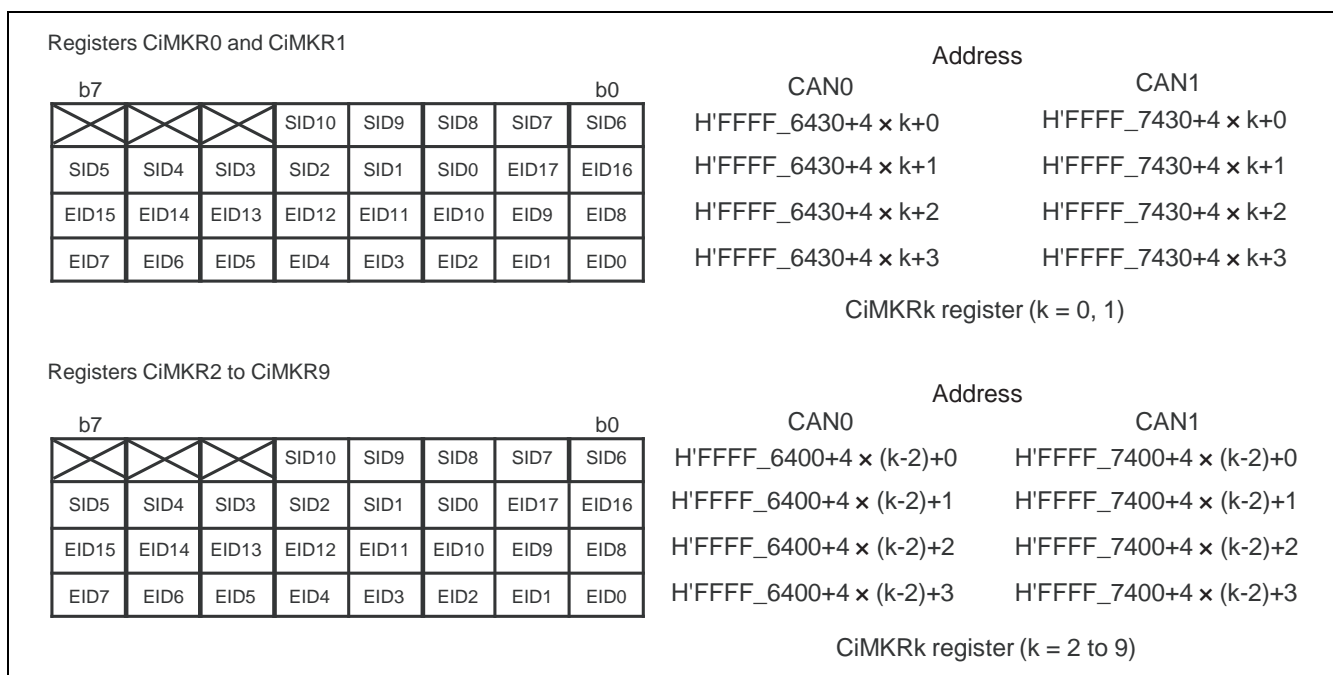


Figure 47.14 Structure of CiMKRk Register (i = 0, 1; k = 0 to 9)

Figure 47.15 shows the structure of the CiFIDCRn register.

There are 2 FIFO received ID compare registers with the same structure.

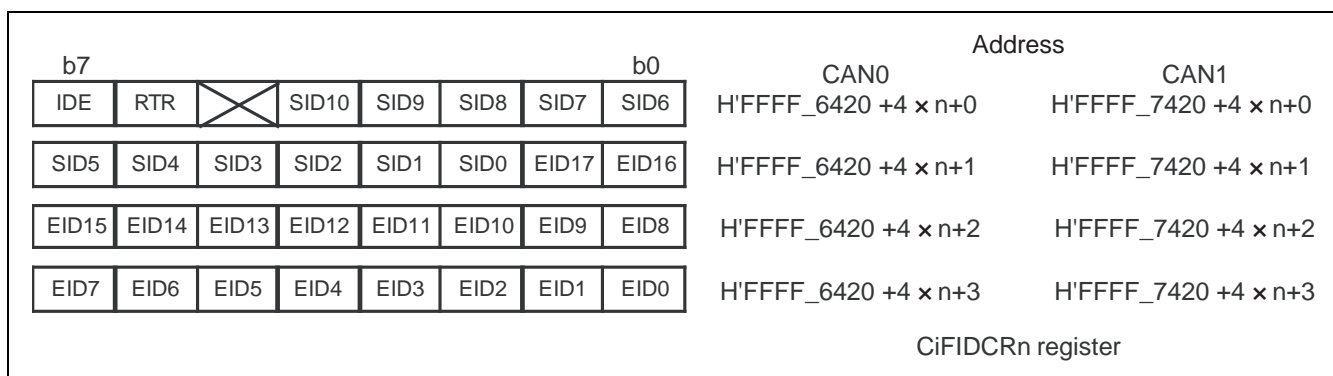


Figure 47.15 Structure of CiFIDCRn register (i = 0, 1; n = 0, 1)

47.6 Acceptance Filtering and Masking Function

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Acceptance filtering allows the user to receive messages with a specified range of multiple IDs for mailboxes. Registers CiMKR0 to CiMKR9 can perform masking of the standard ID and the extended ID of 29 bits.

- The CiMKR0 register corresponds to mailboxes [0] to [15].
- The CiMKR1 register corresponds to mailboxes [16] to [31].
- The CiMKR2 register corresponds to mailboxes [32] to [35].
- The CiMKR3 register corresponds to mailboxes [36] to [39].
- The CiMKR4 register corresponds to mailboxes [40] to [43].
- The CiMKR5 register corresponds to mailboxes [44] to [47].
- The CiMKR6 register corresponds to mailboxes [48] to [51].
- The CiMKR7 register corresponds to mailboxes [52] to [55].
- The CiMKR8 register corresponds to mailboxes [56] to [59] in normal mailbox mode and the receive FIFO mailboxes [60] to [63] in FIFO mailbox mode.
- The CiMKR9 register corresponds to mailboxes [60] to [63] in normal mailbox mode and the receive FIFO mailboxes [60] to [63] in FIFO mailbox mode.

Registers CiMKIVLR0 and CiMKIVLR1 disable acceptance filtering individually for each mailbox.

The IDE bit in the CiMBj register is enabled when the ID bit in the CiCTRL register is "B'10" (mixed ID mode).

The RTR bit in the CiMBj register selects a data frame or a remote frame.

In FIFO mailbox mode, normal mailboxes (mailboxes [0] to [55]) use the single corresponding register among registers CiMKR0 to CiMKR7 for acceptance filtering. Receive FIFO mailboxes (mailboxes [60] to [63]) use two registers CiMKR8 and CiMKR9 for the acceptance filtering.

Also, the receive FIFO uses two registers CiFIDCR0 and CiFIDCR1 for ID comparison. Bits EID, SID, RTR, and IDE in registers CiMB60 to CiMB63 for the receive FIFO are disabled. As acceptance filtering depends on the result of two ID-mask sets, two ranges of IDs can be received into the receive FIFO.

Registers CiMKIVLR0 and CiMKIVLR1 are disabled for the receive FIFO.

If both setting of standard ID and extended ID are set in the IDE bits in registers CiFIDCR0 and CiFIDCR1 individually, both ID formats are received.

If both setting of data frame and remote frame are set in the RTR bits in registers CiFIDCR0 and CiFIDCR1 individually, both data and remote frames are received.

When combination with two ranges of IDs is not necessary, set the same mask value and the same ID into both of the FIFO ID/mask register sets.

Figure 47.16 shows the correspondence of mask registers to mailboxes. Figure 47.17 shows the acceptance filtering.

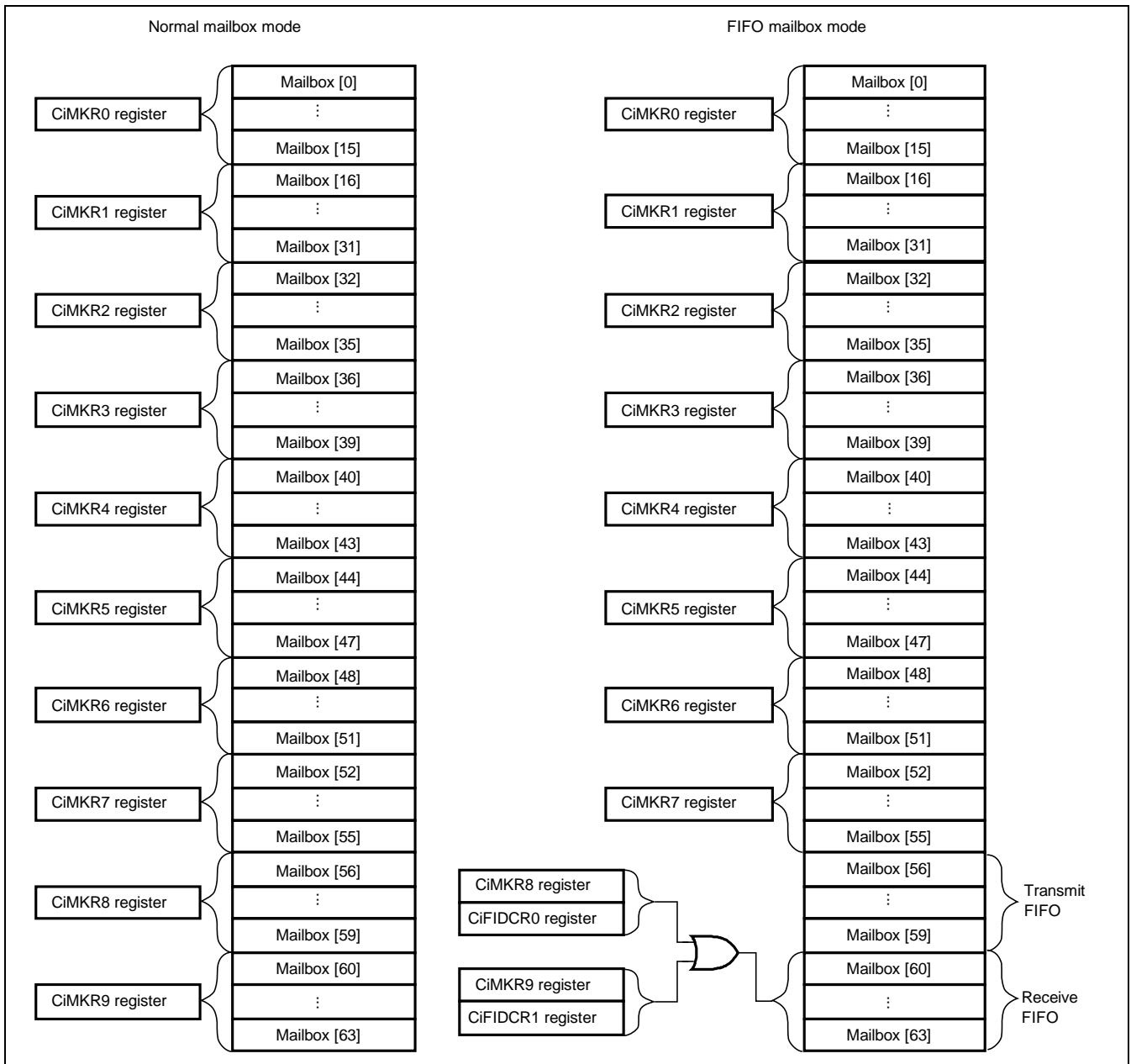


Figure 47.16 Correspondence of Mask Registers to Mailboxes (i = 0, 1)

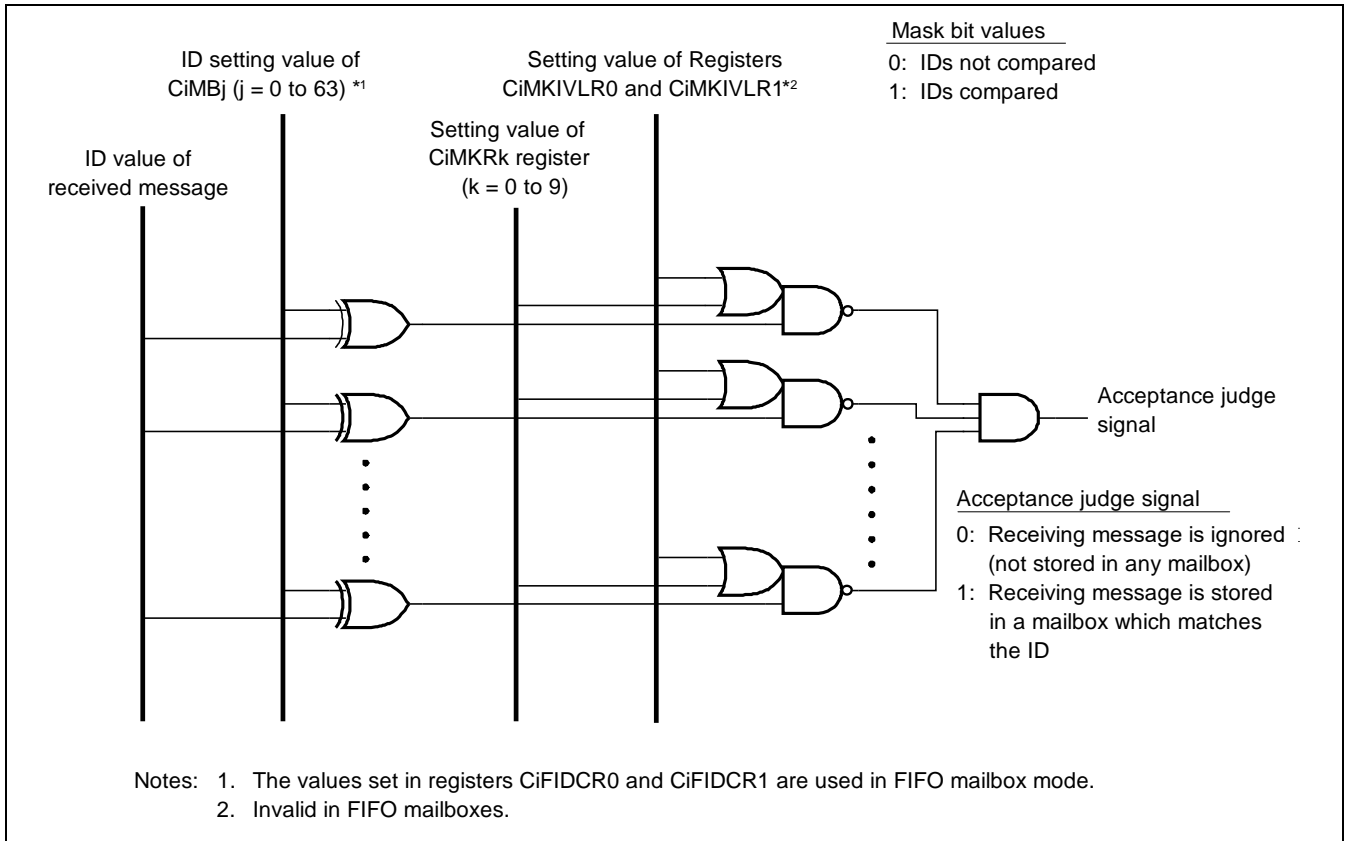


Figure 47.17 Acceptance Filtering (i = 0, 1)

47.7 Reception and Transmission

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 47.12 lists the CAN communication mode configuration.

Table 47.12 Configuration for CAN Reception Mode and Transmission Mode

| TRMREQ | RECREQ | ONESHOT | Communication Mode of Mailbox |
|--------|--------|---------|---|
| 0 | 0 | 0 | Mailbox disabled or transmission being aborted |
| 0 | 0 | 1 | Configurable only when transmission or reception from a mailbox (programmed in one-shot mode) is aborted. |
| 0 | 1 | 0 | Configured as a receive mailbox for a data frame or a remote frame. |
| 0 | 1 | 1 | Configured as a one-shot receive mailbox for a data frame or a remote frame. |
| 1 | 0 | 0 | Configured as a transmit mailbox for a data frame or a remote frame. |
| 1 | 0 | 1 | Configured as a one-shot transmit mailbox for a data frame or a remote frame. |
| 1 | 1 | 0 | Do not set. |
| 1 | 1 | 1 | Do not set. |

Note: TRMREQ, RECREQ, ONESHOT: Bits in CiMCTLj register (i = 0, 1; j = 32 to 63)

When a mailbox is configured as a receive mailbox or a one-shot receive mailbox, note the following:

1. Before a mailbox is configured as a receive mailbox or a one-shot receive mailbox, set the CiMCTLj register to "H'00".
2. A received message is stored into the first mailbox that matches the condition according to the result of receive mode configuration and acceptance filtering. Upon deciding a mailbox which stores the received message, the mailbox with the smaller number has higher priority.
3. In CAN operation mode, when a CAN module transmits a message whose ID matches with the ID/mask set of a mailbox configured to receive messages, the CAN module never receives the transmitted data. In self-test mode, however, the CAN module may receive its transmitted data. In this case, the CAN module sends an ACK.

When configuring a mailbox as a transmit mailbox or a one-shot transmit mailbox, note the following:

1. Before a mailbox is configured as a transmit mailbox or one-shot transmit mailbox, ensure that the CiMCTLj register is "H'00" and that there is no pending abort process.

47.7.1 Reception

Figure 47.18 shows an operation example of data frame reception in overwrite mode.

This example shows the operation of overwriting the first message when the CAN module receives two consecutive CAN messages that matches the receiving conditions of the CiMCTLj register.

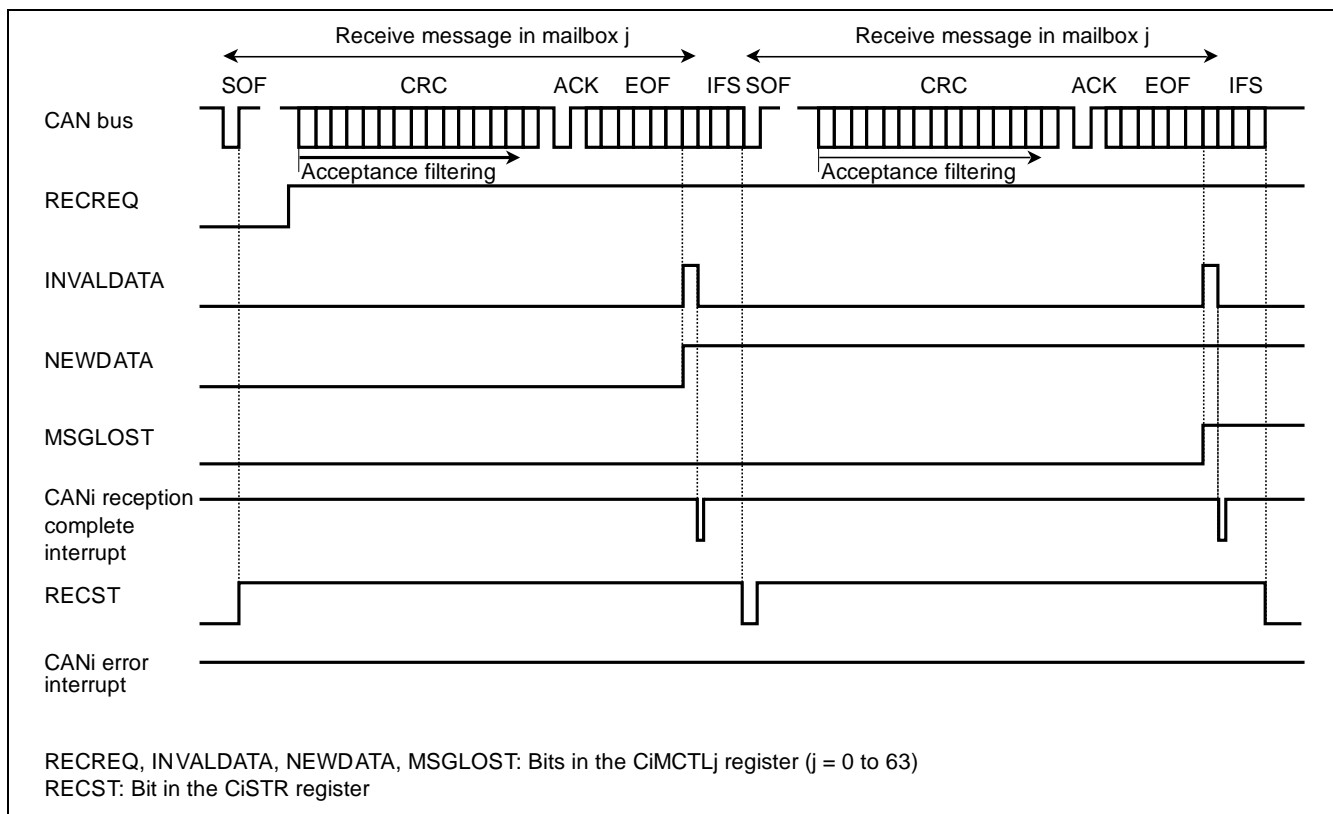


Figure 47.18 Operation Example of Data Frame Reception in Overwrite Mode (i = 0, 1)

1. When a SOF is detected on the CAN bus, the RECST bit in the CiSTR register is set to "1" (reception in progress) if the CAN module has no message ready to start transmission.
2. The acceptance filter procedure starts at the beginning of the CRC field to select the receive mailbox.
3. After a message has been received, the NEWDATA bit in the CiMCTLj register for the receive mailbox is set to "1" (new data being updated/stored in the mailbox). The INVALIDDATA bit in the CiMCTLj register is set to "1" (message is being updated) at the same time, and then the INVALIDDATA bit is set to "0" (message valid) again after the complete message is transferred to the mailbox.
4. When the interrupt enable bit in the CiMIER register for the receive mailbox is "1" (interrupt enabled), the CANi reception complete interrupt request is generated. This interrupt is generated when the INVALIDDATA bit is set to "0".
5. After reading the message from the mailbox, the NEWDATA bit needs to be set to "0" by a program.
6. In overwrite mode, if the next CAN message has been received into a mailbox whose NEWDATA bit is still set to "1", the MSGLOST bit in the CiMCTLj register is set to "1" (message has been overwritten). The new received message is transferred to the mailbox. The CANi reception complete interrupt request is generated the same as in 4.

Figure 47.19 shows the operation example of data frame reception in overrun mode. This example shows the operation of overrunning the second message when the CAN module receives two consecutive CAN messages that matches the receiving conditions of the CiMCTLj register.

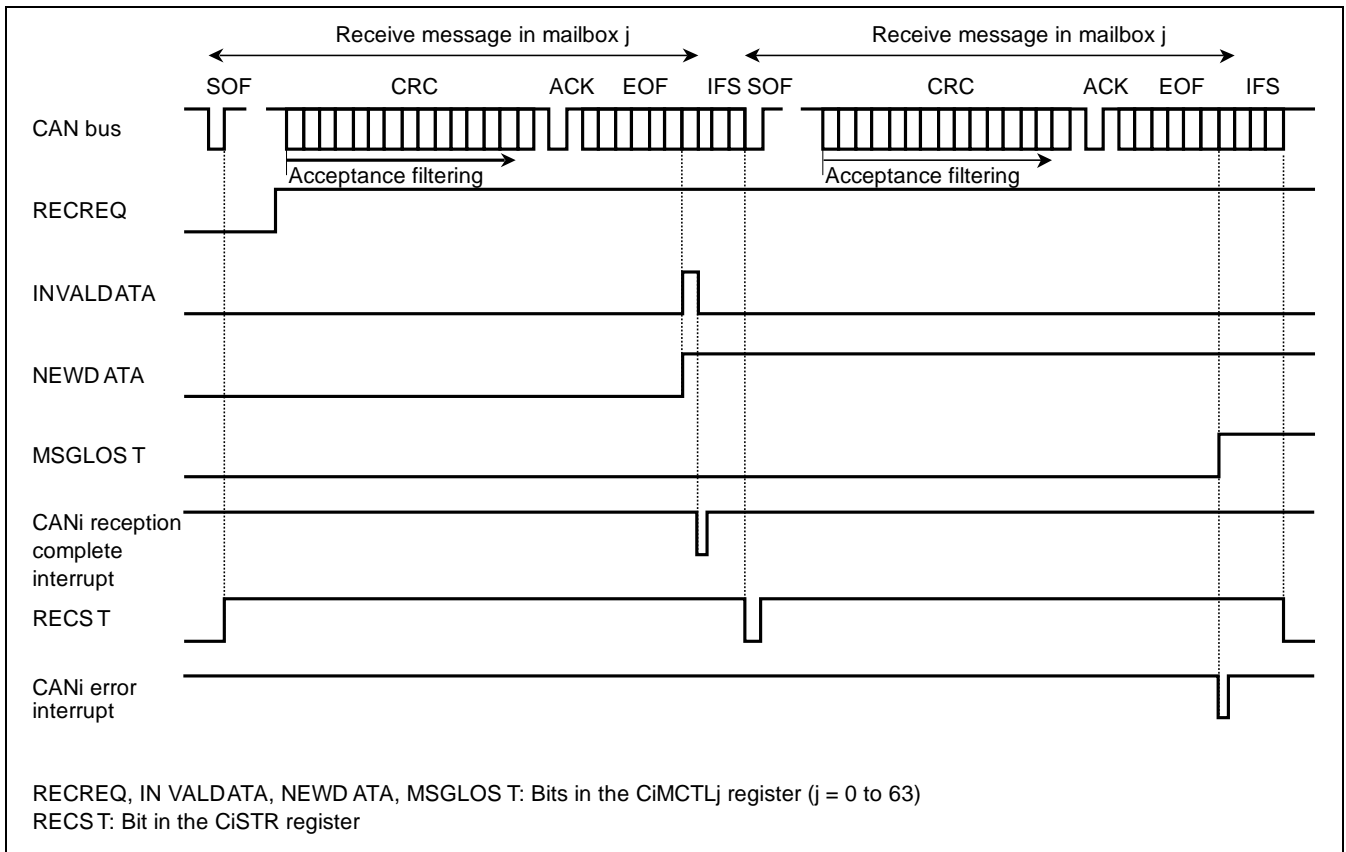


Figure 47.19 Operation Example of Data Frame Reception in Overrun Mode (i = 0, 1)

1. to 5. are the same as overwrite mode.
6. In overrun mode, if the next message has been received before the NEWDATA bit is set to "0", the MSGLOST bit in the CiMCTLj register is set to "1" (message has been overrun). The new received message is discarded and a CANi error interrupt request is generated if the corresponding interrupt enable bit in the CiEIER register is set to "1" (interrupt enabled).

47.7.2 Transmission

Figure 47.20 shows an operation example of data frame transmission.

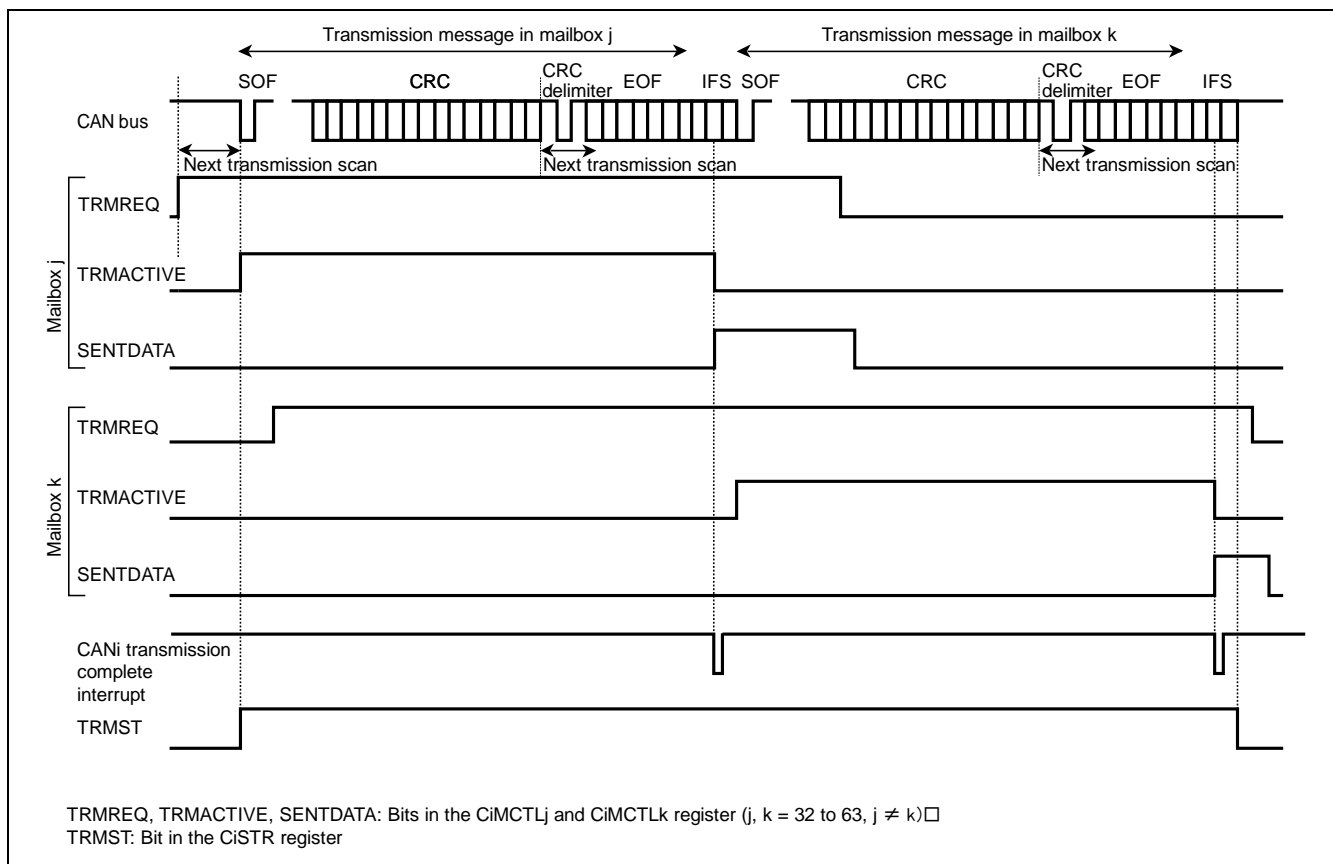


Figure 47.20 Operation Example of Data Frame Transmission (i = 0, 1)

1. When a TRMREQ bit in the CiMCTLj register is set to 1 (transmit mailbox) in bus-idle state, the mailbox scan procedure starts to decide the highest-priority mailbox for transmission. Once the transmit mailbox is decided, the TRMACTIVE bit in the CiMCTLj register is set to "1" (from when a transmission request is received until transmission is completed, or an error/arbitration lost has occurred), the TRMST bit in the CiSTR register is set to "1" (transmission in progress), and the CAN module starts transmission.*
2. If other TRMREQ bits are set, the transmission scan procedure starts with the CRC delimiter for the next transmission.
3. If transmission is completed without losing arbitration, the SENDTDATA bit in the CiMCTLj register is set to "1" (transmission completed) and the TRMACTIVE bit is set to "0" (transmission is pending, or no transmission request). If the interrupt enable bit in the CiMIER register is "1" (interrupt enabled), the CANi transmission complete interrupt request is generated.
4. When requesting the next transmission from the same mailbox, set bits SENDTDATA and TRMREQ to "0", then set the TRMREQ bit to "1" after checking that bits SENDTDATA and TRMREQ have been set to "0".

Note: * If arbitration is lost after the CAN module starts transmission, the TRMACTIVE bit is set to "0". The transmission scan procedure is performed again to search for the highest-priority transmit mailbox from the beginning of the CRC delimiter. If an error occurs either during transmission or following the arbitration lost the transmission scan procedure is performed again from the start of the error delimiter to search for the highest-priority transmit mailbox.

47.8 CAN Interrupt

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The CAN module provides the following CAN interrupts for each channel. Table 47.13 lists CAN interrupts.

- CANi reception complete interrupt (mailbox 0) [RXM0i]
- CANi reception complete interrupt (mailbox 1 to 63) [RXM1i]
- CANi transmission complete interrupt (mailbox 32 to 63) [TXMi]
- CANi reception FIFO interrupt [RXFi]
- CANi transmission FIFO interrupt [TXFi]
- CANi error interrupt [ERSi]

There are eight types of interrupt sources for the CANi error interrupts. These sources can be determined by checking the CiEIFR register.

- Bus error
- Error-warning
- Error-passive
- Bus-off entry
- Bus-off recovery
- Receive overrun
- Overload frame transmission
- Bus lock

Table 47.13 CAN Interrupts

| Module | Interrupt Symbol | Interrupt Source | Source Flag |
|--------|------------------|--|--|
| CANi | ERSi | Bus lock detected | CiEIFR.BLIF |
| | | Overload frame transmission detected | CiEIFR.OLIF |
| | | Overrun detected | CiEIFR.ORIF |
| | | Bus-off recovery detected | CiEIFR.BORIF |
| | | Bus-off entry detected | CiEIFR.BOEIF |
| | | Error-passive detected | CiEIFR.EPIF |
| | | Error-warning detected | CiEIFR.EWIF |
| | | Bus error detected | CiEIFR.BEIF |
| RXFi | RXFi | Receive FIFO message received (CiMIER1[29] = 0) | CiISR.RXFF |
| | | Receive FIFO warning (CiMIER1[29] = 1) | |
| TXFi | TXFi | Transmit FIFO message transmission completed (CiMIER1[25] = 0) | CiISR.TXFF |
| | | FIFO last message transmission completed (CiMIER1[25] = 1) | |
| RXM0i | RXM0i | Mailbox 0 message received | CiMCTL0.NEWDATA |
| RXM1i | RXM1i | Mailbox 1 to 63 message received | CiMCTL1.NEWDATA to CiMCTL63.NEWDATA |
| TXMi | TXMi | Mailbox 32 to 63 message transmission completed | CiMCTL32.SENTDATA to CiMCTL63.SENTDATA |

Legend: i = 0, 1

(1) CANi reception complete interrupt (mailbox 0) [RXM0i]

After the CiMCTL0.NEWDATA bit is set by the completion of reception, if received data has been stored (the corresponding INVALIDDATA bit changes from "1" to "0"), the CiISR.RXM0F bit is set to "1" when CiMIER0[0] has been set to "1". When the mailbox 0 reception complete (RXM0) interrupt has been enabled, the RXM0 interrupt is requested to the interrupt controller.

To clear the RXM0 interrupt, clear the CiMCTL0.NEWDATA bit in the RXM0 interrupt handling routine. To change the CiIER.RXM0IE bit to disabled after having set the bit, make the change while no RXM0 interrupt is generated or during the RXM0 interrupt handling routine. This also applies to CiMIER0[0].

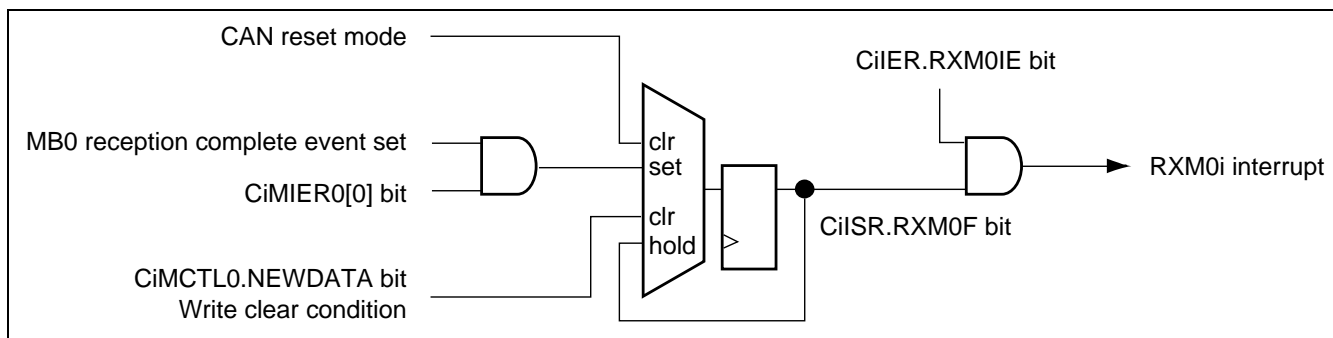


Figure 47.21 Block Diagram of CANi Reception Complete Interrupt (Mailbox 0) [RXM0i]

(2) CANi reception complete interrupt (mailbox 1 to 63) [RXM1i]

After the CiMCTLj.NEWDATA bit is set by the completion of reception, if received data has been stored (the corresponding INVALIDDATA bit changes from "1" to "0"), the CiISR.RXM1F bit is set to "1" when the CiMIER0 or CiMIER1 register bit corresponding to mailbox j has been set to "1". When the mailbox 1 to 63 reception complete (RXM1) interrupt has been enabled, the RXM1 interrupt is requested to the interrupt controller.

To clear the RXM1 interrupt, clear the CiMCTLj.NEWDATA bit in the RXM1 interrupt handling routine. To change the CiIER.RXM1IE bit to disabled after having set the bit, make the change while no RXM1 interrupt is generated or during the RXM1 interrupt handling routine. This also applies to CiMIER0[j] (j = 1 to 31) or CiMIER1[j-32] (j = 32 to 63).

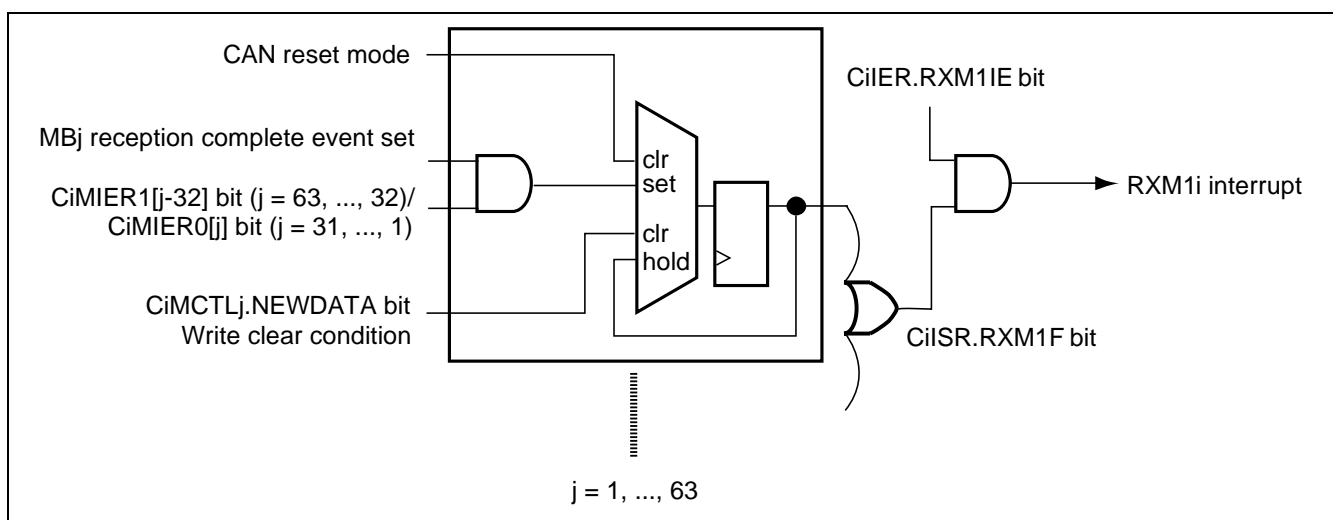


Figure 47.22 Block Diagram of CANi Reception Complete Interrupt (Mailbox 1 to 63) [RXM1i]

(3) CANi transmission complete interrupt (mailbox 32 to 63) [TXMi]

If the CiMCTLj.SENTDATA bit is set by the completion of transmission, the CiISR.TXMF bit is set to "1" when the CiMIER1 register bit corresponding to mailbox j has been set to "1". When the mailbox 32 to 63 transmission complete (TXM) interrupt has been enabled, the TXM interrupt is requested to the interrupt controller.

To clear the TXM interrupt, clear the CiMCTLj.SENTDATA bit in the TXM interrupt handling routine. To change the CiIER.TXMIE bit to disabled after having set the bit, make the change while no TXM interrupt is generated or during the TXM interrupt handling routine. This also applies to CiMIER1[j-32] (j = 32 to 63).

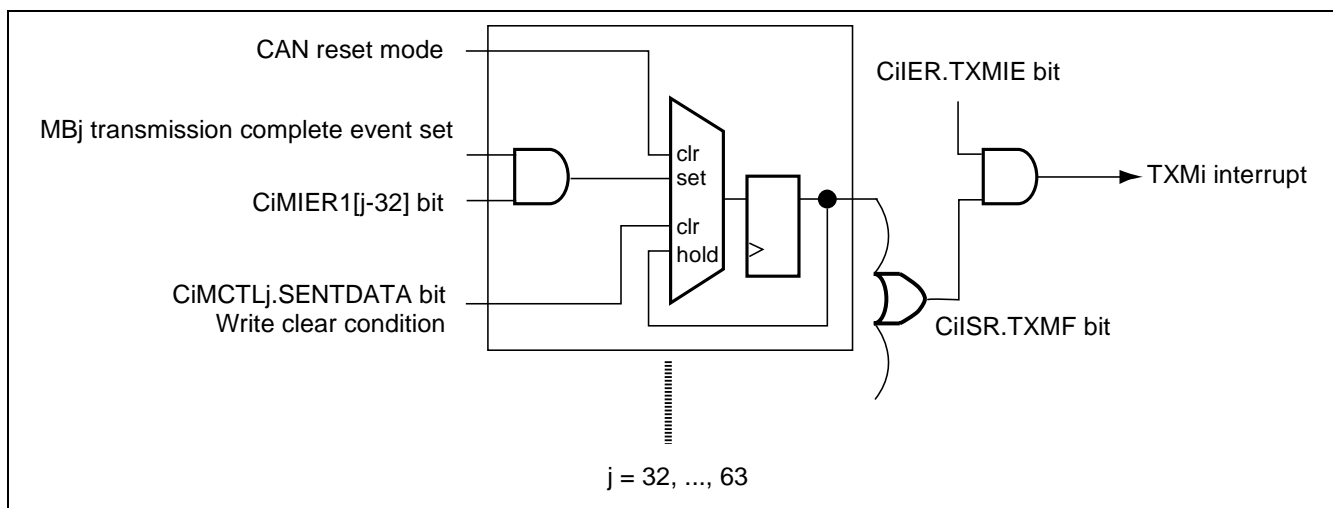


Figure 47.23 Block Diagram of CANi Transmission Complete Interrupt (Mailbox 32 to 63) [TXMi]

(4) CANi receive FIFO interrupt [RXFi]

If CiRFCR[6:5] are set by the reception of a receive FIFO message or by a warning with the settings of CiMIER1[29:28], the CiISR.RXFF bit is set to "1". When the receive FIFO (RXF) interrupt has been enabled with the CiIER.RXFIE bit, the RXF interrupt is requested to the interrupt controller.

To clear the RXF interrupt, clear the CiISR.RXFF bit in the RXF interrupt handling routine. To change the CiIER.RXFIE bit to disabled after having set the bit, make the change while no RXF interrupt is generated or during the RXF interrupt handling routine. This also applies to CiMIER1[28].

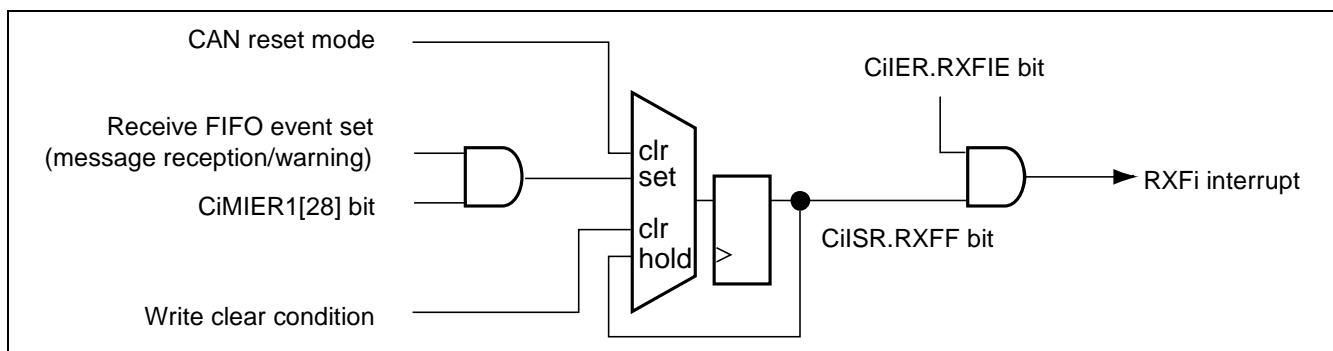


Figure 47.24 Block Diagram of CANi Receive FIFO Interrupt [RXFi]

(5) CANi transmit FIFO interrupt [TXFi]

When the transmission of a transmit FIFO message is counted for the specified number of times with the settings of CiMIER1 [25:24], the CiISR.TXFF bit is set to "1". When the transmit FIFO (TXF) interrupt has been enabled with the CiIER.TXFIE bit, the TXF interrupt is requested to the interrupt controller.

To clear the TXF interrupt, clear the CiISR.TXFF bit in the TXF interrupt handling routine. To change the CiIER.TXFIE bit to disabled after having set the bit, make the change while no TXF interrupt is generated or during the TXF interrupt handling routine. This also applies to CiMIER1[24].

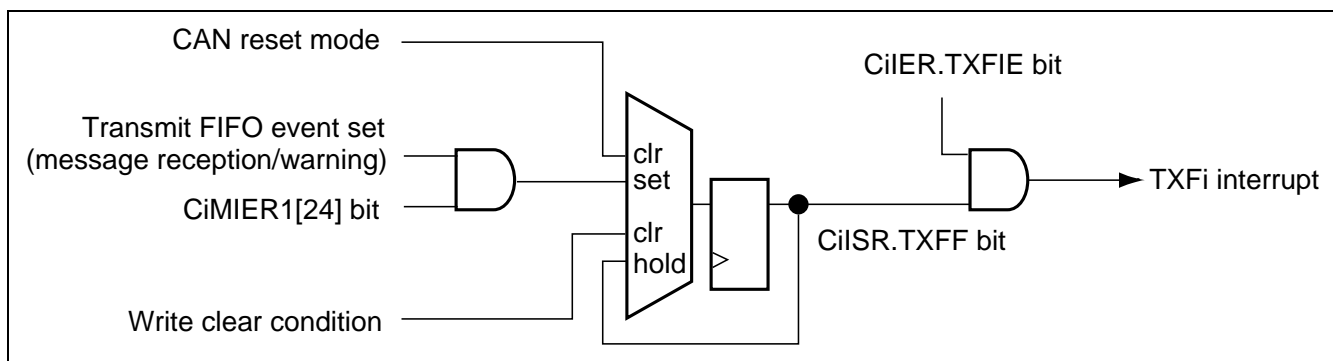


Figure 47.25 Block Diagram of CANi Transmit FIFO Interrupt [TXFi]

(6) CANi error interrupt [ERSi]

If CiEIFR[j] is set by a communication error, the CiISR.ERSF bit is set to "1" when the corresponding CiEIER[j] has been set to "1". When the error (ERS) interrupt has been enabled with the CiIER.ERSIE bit, the ERS interrupt is requested to the interrupt controller.

To clear the ERS interrupt, clear each CiEIFR[j] register bit in the ERS interrupt handling routine. To change the CiIER.ERSIE bit to disabled after having set the bit, make the change while no ERS interrupt is generated or during the ERS interrupt handling routine. This also applies to CiEIER[j] (j = 7 to 0).

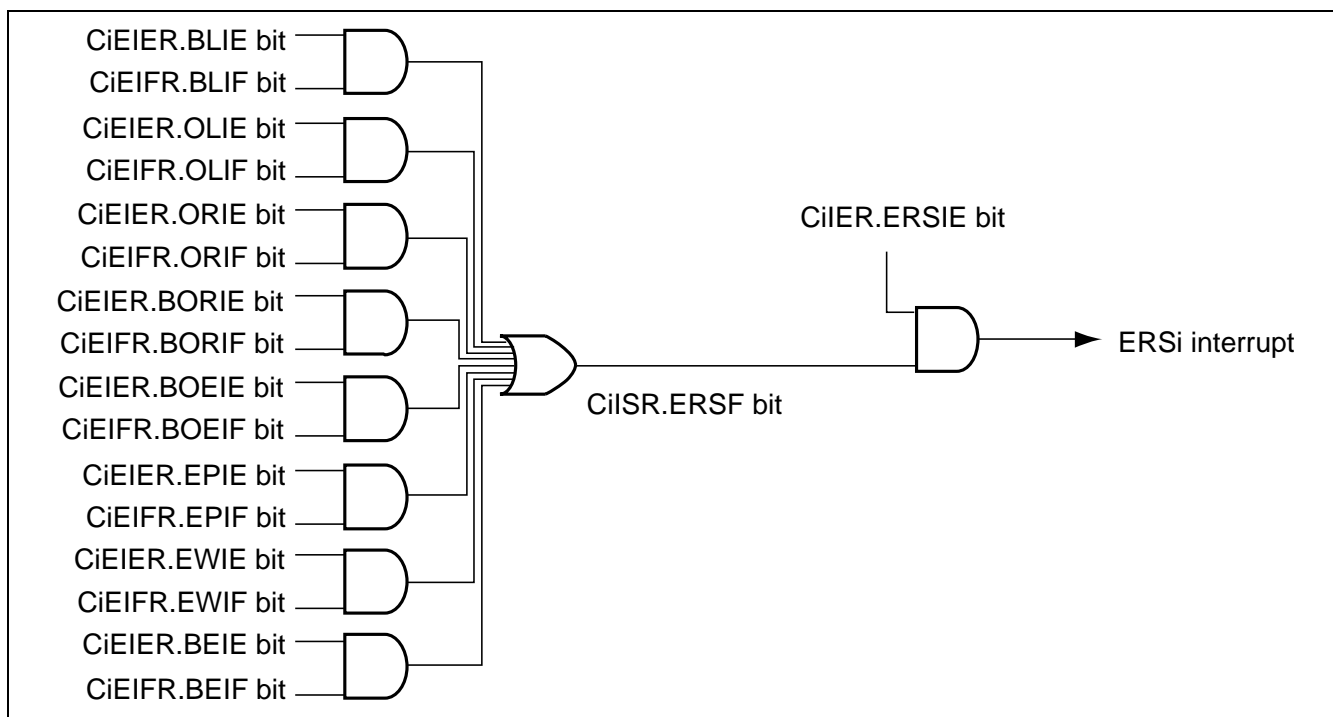


Figure 47.26 Block Diagram of CANi Error Interrupt [ERSi]

47.9 CAN module can support Parity Check

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

47.9.1 Overview

The Parity bit generator for an Even Parity check is a simple XOR of each bit. Exclusive or (XOR) is represented by "^" as follows.

$$\text{parity}[0] = \text{DataBus}[7:0]$$

$$\text{parity}[1] = \text{DataBus}[15:8]$$

$$\text{parity}[2] = \text{DataBus}[23:16]$$

$$\text{parity}[3] = \text{DataBus}[31:24]$$

The Parity error detection for an Even Parity check is a simple XOR of each bit including the Parity bit. In case the result is equal to the Data Byte no error is generated. The detection logic is only working during read access for the selected byte.

$$\text{parity_error}[0] = \{ \text{DataBus}[7:0] \} \wedge \text{parity}[0]$$

$$\text{parity_error}[1] = \{ \text{DataBus}[15:8] \} \wedge \text{parity}[1]$$

$$\text{parity_error}[2] = \{ \text{DataBus}[23:16] \} \wedge \text{parity}[2]$$

$$\text{parity_error}[3] = \{ \text{DataBus}[31:24] \} \wedge \text{parity}[3]$$

Each memory location in the DPRAM consists of 36 bits. 32 bits are used for storage of user data. For each byte of a longword memory location a parity bit is allocated. The value of this parity bit is calculated by the parity generation logic for each byte and both data byte and parity bit are stored in memory during the same write access. The alignment of data bytes and corresponding parity bits in each memory location is shown in the following diagram:

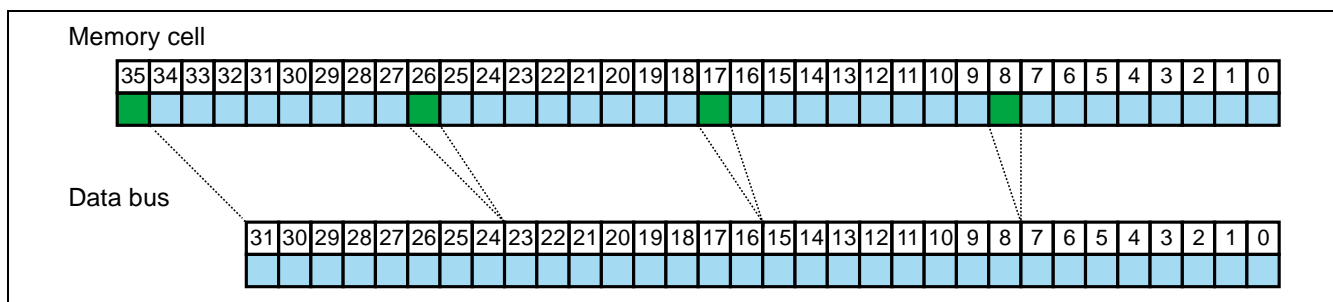


Figure 47.27 Parity bit alignment

47.9.2 Recommend Usage

In order to use this function, refer to 47.2.29 (Parity Error Control Register (CiPECR) (i = 0, 1)) to know how to enable this function.

47.9.3 Failure Control

When parity error detected, an interrupt request is issued to Interrupt handle is obtained.

48. CAN-FD

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

This section contains a generic description of the CANFD interface (RS-CANFD).

The first part of this section describes all specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of RS-CANFD.

48.1 Overview

48.1.1 Functional Overview

Table 48.1 shows the RS-CANFD module specifications. Figure 48.1 shows the RS-CANFD module block diagram.

Table 48.1 RS-CANFD Module Specifications

| Item | Specification |
|---------------------|--|
| Number of channels | |
| Protocol | ISO11898-1 compliant Use of CAN FD frames can be selected by switching the interface modes. |
| Communication speed | <p>Classical CAN mode:</p> <p>Maximum 1 Mbps</p> $\text{Communication speed (CANm bit time clock)} = \frac{1}{\text{CANm bit time}}$ $\text{CANm bit time} = \text{CANmTq} \times \text{Tq count per bit}$ $\text{CANmTq} = \frac{(\text{BRP}[9:0] \text{ bits in the RSCFDnCmCFG register} + 1)}{f_{\text{CAN}}}$ <p>fCAN: Frequency of CAN clock (selected by the DCS bit in the RSCFDnGCFG register)</p> <p>CAN FD mode:</p> <p>Normal bit rate: 1 Mbps (max.), Data bit rate: 8 Mbps (max.)</p> $\text{Communication speed (CANm normal bit time clock)} = \frac{1}{\text{CANm normal bit time}}$ $\text{Communication speed (CANm data bit time clock)} = \frac{1}{\text{CANm data bit time}}$ $\text{CANm normal bit time} = \text{CANmTq(N)} \times \text{Tq count per normal bit}$ $\text{CANm data bit time} = \text{CANmTq(D)} \times \text{Tq count per normal bit}$ $\text{CANmTq(N)} = \frac{(\text{NBRP}[9:0] \text{ bits in RSCFDnCFDCmNCFG register} + 1)}{f_{\text{CAN}}}$ $\text{CANmTq(D)} = \frac{(\text{DBRP}[7:0] \text{ bits in RSCFDnCFDCmDCFGregister} + 1)}{f_{\text{CAN}}}$ <p>fCAN: Frequency of CAN clock (selected by the DCS bit in the RSCFDnCFDGCFCFG register)</p> <p>m = 0 or 1</p> <p>Tq: Time quantum</p> |

| Item | Specification |
|-----------------------------------|--|
| Buffer | <ul style="list-style-type: none"> Individual buffers: 32 buffers (16 buffers × 2 channels) Transmit buffer: 16 buffers per channel Transmit FIFO: 3 FIFO per channel Transmit queue: Single queue per channel (shared with the transmit buffer; up to 16 buffers allocated) Shared buffers: 128 buffers for all channels Receive buffer: 32 buffers Receive FIFO: 8 FIFO (up to 128 buffers allocated to each) Transmit/receive FIFO buffer: 3 FIFO buffers per channel (up to 128 buffers allocated to each) ECC included |
| Reception function | <ul style="list-style-type: none"> Receives data frames and remote frames. Selects ID format (standard ID, extended ID, or both IDs) to be received. Sets interrupt enable/disable for each FIFO. Mirror function (reception of messages transmitted from the own CAN node) Timestamp function (to record message reception time as a 16-bit timer value) |
| Reception filter function | <ul style="list-style-type: none"> Selects receive messages according to 1 receive rule. Sets the number of receive rules (0 to 128) for each channel. Acceptance filter processing: Sets ID and mask for each receive rule. DLC filter processing: Enables DLC filter check for each acceptance rule. |
| Receive message transfer function | <ul style="list-style-type: none"> Routing function Transfers receive messages to arbitrary destinations (can be transferred to up to 8 buffers) Transfer destination: Receive buffer, receive FIFO buffer, and/or transmit/receive FIFO buffer Label addition function Stores label information together with a message in a receive buffer and FIFO buffer. |
| Transmission function | <ul style="list-style-type: none"> Transmits data frames and remote frames. Selects ID format (standard ID, extended ID, or both IDs) to be transmitted. Sets interrupt enable/disable for each transmit buffer and transmit/receive FIFO buffer. Selects ID priority transmission or transmit buffer number priority transmission. Transmit request can be aborted (possible to confirm with a flag) One-shot transmission function |
| Interval transmission function | Transmits messages at configurable intervals (transmit mode or gateway mode of transmit/receive FIFO buffers) |
| Transmit queue function | Transmits all stored messages according to the ID priority. |
| Transmit history function | Stores the history information of transmission-completed messages Adds the timestamp to the history information (records the 16-bit timer value for the message transmission time). |
| Gateway function | Transmits a received message automatically. |
| Bus off recovery mode selection | Selects the method for returning from bus off state. <ul style="list-style-type: none"> ISO11898-1 compliant Automatic entry to channel halt mode at bus-off entry Automatic entry to channel halt mode at bus-off end Transition to channel standby mode by program request Transition to the error-active state by program request (forcible return from the bus off state) |

| Item | Specification |
|-------------------------|--|
| Error status monitoring | <ul style="list-style-type: none"> Monitors CAN protocol errors (stuff error, form error, ACK error, CRC error, bit error, ACK delimiter error, and bus dominant lock). Detects error status transitions (error warning, error passive, bus off entry, and bus off recovery) Reads the error counter. Monitors DLC errors. |
| Interrupt source | <ul style="list-style-type: none"> Global Interrupts (2 sources) <ul style="list-style-type: none"> Receive FIFO interrupt Global error interrupt Channel interrupts (3 sources/channel) <ul style="list-style-type: none"> CANm transmit interrupt (m = 0 or 1) CANm transmit complete interrupt CANm transmit abort interrupt CANm transmit/receive FIFO transmit complete interrupt (in transmit mode, gateway mode) CANm transmit history interrupt CANm transmit queue interrupt <p>CANm transmit/receive FIFO receive complete interrupt (in receive mode, gateway mode)</p> <p>CANm error interrupt</p> |
| CAN stop mode | Reduces power consumption by stopping clock supply to the RS-CANFD module. |
| CAN clock source | <p>Selects the clk_c or the clk_{xincan}.</p> <p>As for the range of operating frequency, see section 48.16.1.3, Setting Communication Speed.</p> |
| Test function | <p>Test function for user evaluation</p> <ul style="list-style-type: none"> Listen-only mode Self-test mode 0 (external loopback) Self-test mode 1 (internal loopback) Limited operation mode RAM test (read/write test) Inter-channel communication test [CRC error test is available] |

48.1.2 Interface Mode

RS-CANFD has the following two interface modes.

- Classical CAN mode: Only classical CAN frames are handled.
- CAN FD mode: both the classical CAN frames and CAN FD frames are handled.

Two modes use different register maps with the same base address, and the register maps are switched by switching the modes.

Interface modes can be switched using the RCMC bit in the RSCFDnCFDGRMCFG register.

48.2 Block Diagram

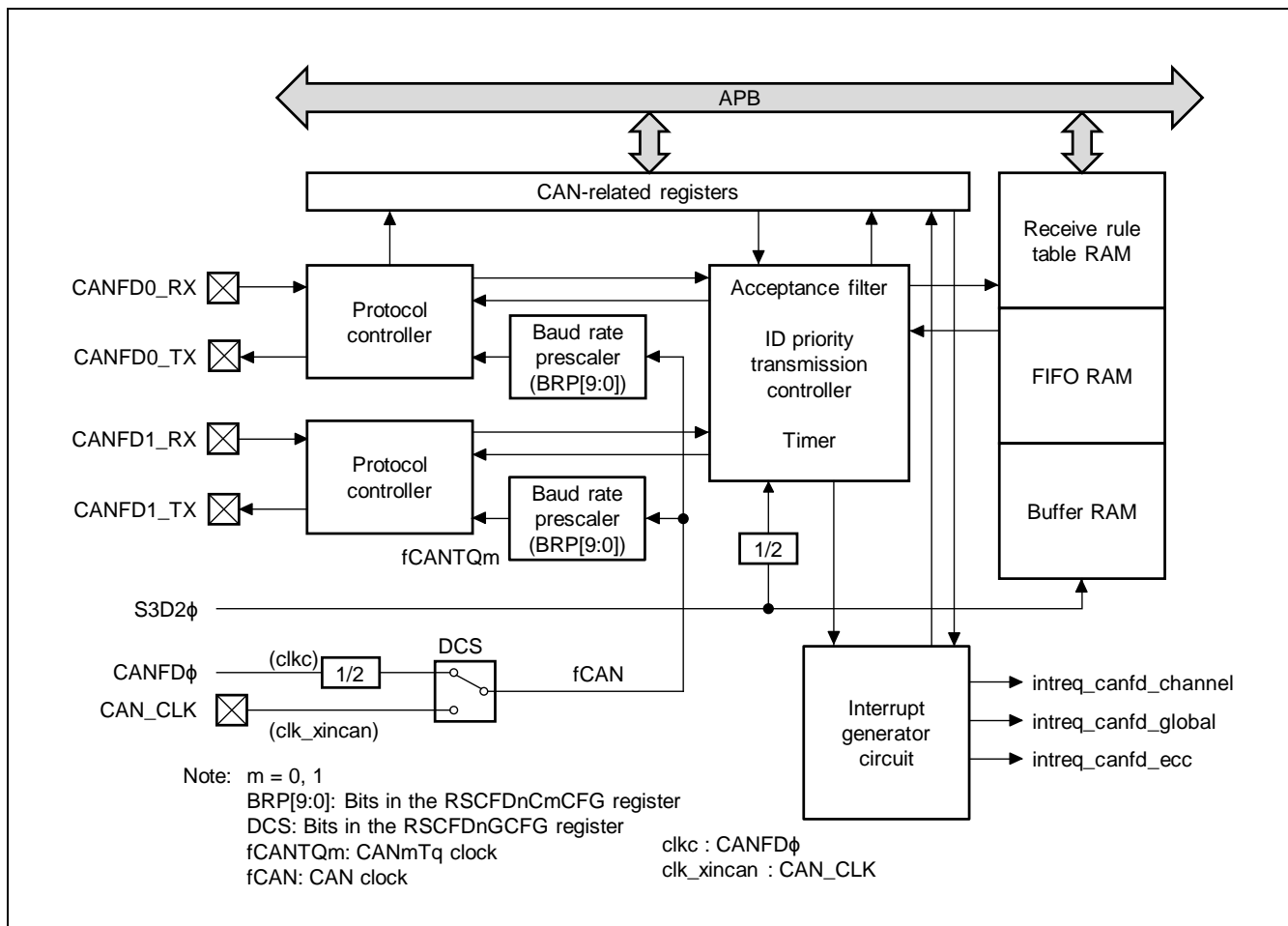


Figure 48.1 RS-CANFD Module Block Diagram (in classical CAN mode)

In CAN FD mode, the specification of the clock to be input to the baud rate prescaler and protocol controller is different from that of the clock in classical CAN mode. Refer to section 48.16.1.3, Setting Communication Speed.

CANFDφ must be set as 80 (MHz).

48.3 External Pins

Table 48.2 shows the CAN module pin.

Pin switching is required for pins which are multiplexed with other Function. For details, see section 8, Pin Function Controller (PFC).

Table 48.2 Pin Configuration

| Name | Abbreviation | I/O | Function | Second Generation RZ/G Series Products | | | |
|-----------|--------------|--------|--|---|----------------------------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| CANFDi_RX | — | Input | Pins for receiving data | √ | √ | √ | √ |
| CANFDi_TX | — | Output | Pins for transmitting data | √ | √ | √ | √ |
| CAN_CLK | — | Input | Input pin used for external clock input. | √ | √ | √ | √ |

Legend: i = 0 and 1

48.4 Connected module

Table 48.3 Connected module

| Module name | Connected module name | function of the related module |
|-------------|-----------------------|--------------------------------|
| CANFD | AXI | Register access of CPU |
| | CPG | Clock output |
| | PFC | Selection of external pins |
| | Module Standby | Clock stop control |
| | Software Reset | Soft reset execution |
| | INTC | Interruption |
| | SYS-DMAC | DMA transmission |

48.5 Features of RS-CANFD

RS-CANFD has two interface modes (classical CAN mode and CAN FD mode), and uses different registers in each mode. The different register names RSCFDnXXX and RSCFDnCFDXXX are used in the different interface modes (XXX can be any character). In this document, the registers common to the two modes are indicated as RSCFDn(CFD)XXX.

Table 48.4 Index

| Index | Meaning |
|-------|--|
| n | Throughout this section, the individual RS-CANFD units are generically indicated by the index “n” (n = 0); for example, RSCFDn(CFD)GCTR is the global control register of the RSCFDn unit. |
| m | Throughout this section, the individual channels of RS-CANFD units are generically indicated by the index “m” (m = 0, 1); for example, RSCFDn(CFD)CmSTS is the channel m status register. |
| j | The individual registers associated with receive rule table are generically indicated by the index “j” (j = 0 to 15); for example, RSCFDn(CFD)GAFLIDj is the receive rule ID register. |
| k | The individual transmit/receive FIFO buffers are generically indicated by the index “k” (k = 0 to [channel m × 3 + 2]); for example, RSCFDn(CFD)CFCK is the transmit/receive FIFO buffer configuration/control register. |
| x | The individual receive FIFO buffers in the RS-CANFD units are identified by the index “x” (x = 0 to 7); for example, RSCFDn(CFD)RFSTSx is the receive FIFO buffer status register. |
| d | The individual transmit/receive FIFO buffers are generically indicated by the index “d” (d = 0 to 1 in classical CAN mode, and d = 0 to 15 in CAN FD mode); for example, RSCFDn(CFD)CFDFd_k is the transmit/receive FIFO buffer data field register. |
| q | The individual receive buffers are generically indicated by the index “q” (q = 0 to [channel m × 16 + 15]); for example, RSCFDn(CFD)RMIDq is the receive buffer ID register. |
| p | The individual transmit buffers are generically indicated by the index “p” (p = 0 to [channel m × 16 + 15]); for example, RSCFDn(CFD)TMCp is the transmit buffer control register. |
| b | The individual transmit and receive buffers are generically indicated by the index “b” (b = 0 to 1 in classical CAN mode, and b = 0 to 4 in CAN FD mode); for example, RSCFDn(CFD)RMDfb_q is the receive buffer data field register. |
| r | The individual RAM tests for CAN are generically indicated by the index “r” (r = 0 to 63); for example, RSCFDn(CFD)RPGACCr is the RAM test page access register. |
| y | The registers not covered above are indicated by the letter “y” (y = 0); for example, RSCFDn(CFD)RMNDy is a receive buffer new data register. |

Note: The Function and descriptions of registers in this section are for the RS-CANFDs that has 2 channels (m = 0, 1). When referring to information with indices, regard the index values as the ones corresponding to your target product. Also, note that, if the value of an index exceeds the range described in this section due to your target product, write the value after reset when writing to bits outside the index range.

48.5.1 Register Base Address

RSCFDn base addresses are listed in the following table.

RSCFDn register addresses are given as offsets from the base addresses in general.

Table 48.5 Register Base Address

| Base Address Name | Base Address |
|-------------------|--------------|
| <RSCFD0_base> | H'E66C_0000 |

48.6 Registers Configuration (classical CAN mode)

48.6.1 List of Registers

Table 48.6 lists the RS-CANFD registers used in classical CAN mode.

For details about <RSCFDn_base>, see section 48.5.1, Register Base Address.

Table 48.6 Registers

| Channel | Name | Abbreviation | R/W | Offset Address | Size | Second Generation RZ/G Series Products | | | |
|-------------------------|--|-----------------|-----|------------------------|------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Index of Table 48.4. | Global interface mode select register | RSCFDnCFDGRMCFG | R/W | H'04FC | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Channel m configuration register | RSCFDnCmCFG | R/W | H'0000 + (H'10 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Channel m control register | RSCFDnCmCTR | R/W | H'0004 + (H'10 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Channel m status register | RSCFDnCmSTS | R | H'0008 + (H'10 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Channel m error flag register | RSCFDnCmERFL | R/W | H'000C + (H'10 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global configuration register | RSCFDnGCFG | R/W | H'0084 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global control register | RSCFDnGCTR | R/W | H'0088 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global status register | RSCFDnGSTS | R | H'008C | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global error flag register | RSCFDnGERFL | R/W | H'0090 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global timestamp counter register | RSCFDnGTSC | R | H'0094 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global TX interrupt status register 0 | RSCFDnGTINTSTS0 | R | H'0460 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global TX interrupt status register 1 | RSCFDnGTINTSTS1 | R | H'0464 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive rule entry control register | RSCFDnGAFLECTR | R/W | H'0098 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive rule configuration register 0 | RSCFDnGAFLCFG0 | R/W | H'009C | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive rule configuration register 1 | RSCFDnGAFLCFG1 | R | H'00A0 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive rule ID register j | RSCFDnGAFLIDj | R/W | H'0500 + (H'10 × j) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive rule mask register j | RSCFDnGAFLMj | R/W | H'0504 + (H'10 × j) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive rule pointer 0 register j | RSCFDnGAFLP0_j | R/W | H'0508 + (H'10 × j) | 32 | √ | √ | √ | √ |

| | | | | | | Second Generation RZ/G Series Products | | | |
|----------------------|---|----------------|-----|---------------------|------|---|----------------------------|--------|--------|
| Channel | Name | Abbreviation | R/W | Offset Address | Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Index of Table 48.4. | Receive rule pointer 1 register j | RSCFDnGAFLP1_j | R/W | H'050C + (H'10 × j) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive buffer number register | RSCFDnRMNB | R/W | H'00A4 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive buffer new data register y | RSCFDnRMNDy | R/W | H'00A8 + (H'04 × y) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive buffer ID register q | RSCFDnRMIDq | R | H'0600 + (H'10 × q) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive buffer pointer register q | RSCFDnRMPTRq | R | H'0604 + (H'10 × q) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive buffer data field 0 register q | RSCFDnRMDf0_q | R | H'0608 + (H'10 × q) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive buffer data field 1 register q | RSCFDnRMDf1_q | R | H'060C + (H'10 × q) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer configuration and control register x | RSCFDnRFCCx | R/W | H'00B8 + (H'04 × x) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer status register x | RSCFDnRFSTx | R/W | H'00D8 + (H'04 × x) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer pointer control register x | RSCFDnRFPCTRx | R/W | H'00F8 + (H'04 × x) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer access ID register x | RSCFDnRFIDx | R | H'0E00 + (H'10 × x) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer access pointer register x | RSCFDnRFPTRx | R | H'0E04 + (H'10 × x) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer access data field 0 register x | RSCFDnRFDF0_x | R | H'0E08 + (H'10 × x) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer access data field 1 register x | RSCFDnRFDF1_x | R | H'0E0C + (H'10 × x) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer configuration and control register k | RSCFDnCFCCk | R/W | H'0118 + (H'04 × k) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer status register k | RSCFDnCFSTk | R/W | H'0178 + (H'04 × k) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer pointer control register k | RSCFDnCFPCTRk | R/W | H'01D8 + (H'04 × k) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer access ID register k | RSCFDnCFIDk | R/W | H'0E80 + (H'10 × k) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer access pointer register k | RSCFDnCFPTRk | R/W | H'0E84 + (H'10 × k) | 32 | √ | √ | √ | √ |

| Channel | Name | Abbreviation | R/W | Offset Address | Size | Second Generation RZ/G Series Products | | | |
|----------------------|--|-----------------|-----|---------------------|------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Index of Table 48.4. | Transmit/receive FIFO buffer access data field 0 register k | RSCFDnCFDF0_k | R/W | H'0E88 + (H'10 × k) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer access data field 1 register k | RSCFDnCFDF1_k | R/W | H'0E8C + (H'10 × k) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | FIFO empty status register | RSCFDnFESTS | R | H'0238 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | FIFO full status register | RSCFDnFFSTS | R | H'023C | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | FIFO Msg lost status register | RSCFDnFMSTS | R | H'0240 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer interrupt flag status register | RSCFDnRFISTS | R | H'0244 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer receive interrupt flag status register | RSCFDnCFRISTS | R | H'0248 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer transmit interrupt flag status register | RSCFDnCTISTS | R | H'024C | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer control register p | RSCFDnTMCp | R/W | H'0250 + (H'01 × p) | 8 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer status register p | RSCFDnTMSTSp | R/W | H'02D0 + (H'01 × p) | 8 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer ID register p | RSCFDnTMIDp | R/W | H'1000 + (H'10 × p) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer pointer register p | RSCFDnTMPTRp | R/W | H'1004 + (H'10 × p) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer data field 0 register p | RSCFDnTMDf0_p | R/W | H'1008 + (H'10 × p) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer data field 1 register p | RSCFDnTMDf1_p | R/W | H'100C + (H'10 × p) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer interrupt enable configuration register y | RSCFDnTMIECy | R/W | H'0390 + (H'04 × y) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer transmit request status register y | RSCFDnTMTRSTSy | R | H'0350 + (H'04 × y) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer transmit abort request status register y | RSCFDnTMTARSTSy | R | H'0360 + (H'04 × y) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer transmit complete status register y | RSCFDnTMTcSTSy | R | H'0370 + (H'04 × y) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer transmit abort status register y | RSCFDnTMTASTSy | R | H'0380 + (H'04 × y) | 32 | √ | √ | √ | √ |

| | | | | | | Second Generation RZ/G Series Products | | | |
|----------------------|---|----------------|-----|---------------------|------|---|----------------------------|--------|--------|
| Channel | Name | Abbreviation | R/W | Offset Address | Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Index of Table 48.4. | Transmit queue configuration and control register m | RSCFDnTXQCCm | R/W | H'03A0 + (H'04 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit queue status register m | RSCFDnTXQSTSm | R/W | H'03C0 + (H'04 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit queue pointer control register m | RSCFDnTXQPCTrm | R/W | H'03E0 + (H'04 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit history configuration and control register m | RSCFDnTHLCCm | R/W | H'0400 + (H'04 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit history status register m | RSCFDnTHLSTSm | R/W | H'0420 + (H'04 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit history pointer control register m | RSCFDnTHLPCTrm | R/W | H'0440 + (H'04 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit history access register m | RSCFDnTHLACCm | R | H'1800 + (H'04 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global test configuration register | RSCFDnGTSTCFG | R/W | H'0468 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global test control register | RSCFDnGTSTCTR | R/W | H'046C | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global lock key register | RSCFDnGLOCKK | R/W | H'047C | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | RAM test page access register r | RSCFDnRPGACCr | R/W | H'1900 + (H'04 × r) | 32 | √ | √ | √ | √ |

Table 48.7 Transmit Buffer p Allocated to Each Channel

| | CANm |
|-------------------|-----------------------------|
| Transmit buffer p | Transmit buffer 16 × m + 0 |
| | Transmit buffer 16 × m + 1 |
| | Transmit buffer 16 × m + 2 |
| | Transmit buffer 16 × m + 3 |
| | Transmit buffer 16 × m + 4 |
| | Transmit buffer 16 × m + 5 |
| | Transmit buffer 16 × m + 6 |
| | Transmit buffer 16 × m + 7 |
| | Transmit buffer 16 × m + 8 |
| | Transmit buffer 16 × m + 9 |
| | Transmit buffer 16 × m + 10 |
| | Transmit buffer 16 × m + 11 |
| | Transmit buffer 16 × m + 12 |
| | Transmit buffer 16 × m + 13 |
| | Transmit buffer 16 × m + 14 |
| | Transmit buffer 16 × m + 15 |

Table 48.8 Transmit/Receive FIFO Buffer k Allocated to Each Channel

| | CANm |
|--------------------------------|--|
| Transmit/receive FIFO buffer k | Transmit/receive FIFO buffer 3 × m + 0 |
| | Transmit/receive FIFO buffer 3 × m + 1 |
| | Transmit/receive FIFO buffer 3 × m + 2 |

Table 48.9 Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer by the Setting of Bits CFTML[3:0]

| Setting of Bits CFTML[3:0] | Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer |
|-----------------------------------|---|
| B'0000 | Transmit buffer 16 × m + 0 |
| B'0001 | Transmit buffer 16 × m + 1 |
| B'0010 | Transmit buffer 16 × m + 2 |
| B'0011 | Transmit buffer 16 × m + 3 |
| B'0100 | Transmit buffer 16 × m + 4 |
| B'0101 | Transmit buffer 16 × m + 5 |
| B'0110 | Transmit buffer 16 × m + 6 |
| B'0111 | Transmit buffer 16 × m + 7 |
| B'1000 | Transmit buffer 16 × m + 8 |
| B'1001 | Transmit buffer 16 × m + 9 |
| B'1010 | Transmit buffer 16 × m + 10 |
| B'1011 | Transmit buffer 16 × m + 11 |
| B'1100 | Transmit buffer 16 × m + 12 |
| B'1101 | Transmit buffer 16 × m + 13 |
| B'1110 | Transmit buffer 16 × m + 14 |
| B'1111 | Transmit buffer 16 × m + 15 |

Table 48.103 Transmit Buffer p Allocated to the Transmit Queue of Each Channel

| Setting of Bits TXQDC[3:0] | Transmit Buffer p Allocated to the Transmit Queue |
|-----------------------------------|--|
| B'0000 | Setting prohibited |
| B'0001 | Setting prohibited |
| B'0010 | Transmit buffer $16 \times m + 15$ to $16 \times m + 13$ |
| B'0011 | Transmit buffer $16 \times m + 15$ to $16 \times m + 12$ |
| B'0100 | Transmit buffer $16 \times m + 15$ to $16 \times m + 11$ |
| B'0101 | Transmit buffer $16 \times m + 15$ to $16 \times m + 10$ |
| B'0110 | Transmit buffer $16 \times m + 15$ to $16 \times m + 9$ |
| B'0111 | Transmit buffer $16 \times m + 15$ to $16 \times m + 8$ |
| B'1000 | Transmit buffer $16 \times m + 15$ to $16 \times m + 7$ |
| B'1001 | Transmit buffer $16 \times m + 15$ to $16 \times m + 6$ |
| B'1010 | Transmit buffer $16 \times m + 15$ to $16 \times m + 5$ |
| B'1011 | Transmit buffer $16 \times m + 15$ to $16 \times m + 4$ |
| B'1100 | Transmit buffer $16 \times m + 15$ to $16 \times m + 3$ |
| B'1101 | Transmit buffer $16 \times m + 15$ to $16 \times m + 2$ |
| B'1110 | Transmit buffer $16 \times m + 15$ to $16 \times m + 1$ |
| B'1111 | Transmit buffer $16 \times m + 15$ to $16 \times m + 0$ |

48.7 Register Description (classical CAN mode)

48.7.1 Details of Interface Mode Related Registers

48.7.1.1 RSCFDnCFDGRMCFG - Global Interface Mode Select Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGRMCFG register can be read/written in 32-bit units

RSCFDnCFDGRMCFG, RSCFDnCFDGRMCFGH registers can be read/written in 16-bit units

RSCFDnCFDGRMCFG, RSCFDnCFDGRMCFGH, RSCFDnCFDGRMCFGH, RSCFDnCFDGRMCFGH registers can be read/written in 8-bit units

Address: RSCFDnCFDGRMCFG: <RSCFDn_base> + H'04FC

RSCFDnCFDGRMCFG: <RSCFDn_base> + H'04FC,

RSCFDnCFDGRMCFGH: <RSCFDn_base> + H'04FE

RSCFDnCFDGRMCFG: <RSCFDn_base> + H'04FC,

RSCFDnCFDGRMCFGH: <RSCFDn_base> + H'04FD,

RSCFDnCFDGRMCFGH: <RSCFDn_base> + H'04FE,

RSCFDnCFDGRMCFGH: <RSCFDn_base> + H'04FF

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | RCMC |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 0 | RCMC | B'0 | R/W | Interface Mode Select Setting this bit to 0 places the CANFD into classical CAN mode. To place the RS-CANFD module from CAN FD mode to classical CAN mode, modify the RSCFDnCFDGRMCFG register after setting the post-reset values to all the registers and bits that are solely assigned to the register map used in CAN FD mode. 0: Classical CAN mode 1: CAN FD mode |

Modify the RSCFDnCFDGRMCFG register only in global reset mode. Also set the register before setting any other RS-CANFD register.

48.7.2 Details of Channel Related Registers

48.7.2.1 RSCFDnCmCFG — Channel Configuration Register (m = 0, 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCmCFG register can be read/written in 32-bit units

RSCFDnCmCFGH, RSCFDnCmCFGGL registers can be read/written in 16-bit units

RSCFDnCmCFGLL, RSCFDnCmCFGHL, RSCFDnCmCFGHL, RSCFDnCmCFGHH registers can be read/written in 8-bit units

Address: RSCFDnCmCFG: <RSCFDn_base> + H'0000 + (H'10 × m)

RSCFDnCmCFGH: <RSCFDn_base> + H'0000 + (H'10 × m),

RSCFDnCmCFGGL: <RSCFDn_base> + H'0002 + (H'10 × m)

RSCFDnCmCFGLL: <RSCFDn_base> + H'0000 + (H'10 × m),

RSCFDnCmCFGHL: <RSCFDn_base> + H'0001 + (H'10 × m),

RSCFDnCmCFGHL: <RSCFDn_base> + H'0002 + (H'10 × m),

RSCFDnCmCFGHH: <RSCFDn_base> + H'0003 + (H'10 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----------|-----|-----|------------|-----|-----|------------|-----|-----|-----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | SJW[1:0] | | — | TSEG2[2:0] | | | TSEG1[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | — | BRP[9:0] | | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 26 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 25, 24 | SJW[1:0] | B'00 | R/W | Resynchronization Jump Width Control These bits are used to specify a Tq value for the resynchronization jump width. Allowed values are 1 Tq to 4 Tq, inclusive. Set a value less than or equal to the value of the TSEG2 bits. B'00: 1 Tq B'01: 2 Tq B'10: 3 Tq B'11: 4 Tq |
| 23 | Reserved | B'0 | R | This bit is read as the value after reset. The write value should be the value after reset. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 22 to 20 | TSEG2[2:0] | B'000 | R/W | <p>Time Segment 2 Control</p> <p>These bits are used to specify a Tq value for the length of phase segment 2 (PHASE_SEG2).</p> <p>Allowed values are 2 Tq to 8 Tq, inclusive.</p> <p>Set a value smaller than the value of the TSEG1 bits.</p> <p>B'000: Setting prohibited</p> <p>B'001: 2 Tq</p> <p>B'010: 3 Tq</p> <p>B'011: 4 Tq</p> <p>B'100: 5 Tq</p> <p>B'101: 6 Tq</p> <p>B'110: 7 Tq</p> <p>B'111: 8 Tq</p> |
| 19 to 16 | TSEG1[3:0] | B'0000 | R/W | <p>Time Segment 1 Control</p> <p>These bits are used to specify a Tq value for the total length of the propagation segment (PROP_SEG) and phase segment 1 (PHASE_SEG1).</p> <p>Allowed values are 4 Tq to 16 Tq, inclusive.</p> <p>B'0000: Setting prohibited</p> <p>B'0001: Setting prohibited</p> <p>B'0010: Setting prohibited</p> <p>B'0011: 4 Tq</p> <p>B'0100: 5 Tq</p> <p>B'0101: 6 Tq</p> <p>B'0110: 7 Tq</p> <p>B'0111: 8 Tq</p> <p>B'1000: 9 Tq</p> <p>B'1001: 10 Tq</p> <p>B'1010: 11 Tq</p> <p>B'1011: 12 Tq</p> <p>B'1100: 13 Tq</p> <p>B'1101: 14 Tq</p> <p>B'1110: 15 Tq</p> <p>B'1111: 16 Tq</p> |
| 15 to 10 | Reserved | All 0 | R | <p>These bits are read as the value after reset.</p> <p>The write value should be the value after reset.</p> |
| 9 to 0 | BRP[9:0] | H'000 | R/W | <p>Prescaler Division Ratio Set</p> <p>The CANmTq clock (fCANTQm) is calculated by dividing the CAN clock (fCAN) by the baud rate prescaler, ((BRP[9:0]) + 1). One clock cycle of the CANmTq clock is 1 Time Quantum (Tq).</p> <p>When these bits are set to P (0 to 1023), the baud rate prescaler divides fCAN by P + 1.</p> |

Modify the RSCFDnCmCFG register in channel reset mode or channel halt mode. Set this register in channel reset mode before shifting to channel communication mode or channel wait mode. For a description of the bit timing parameters and settings, see section 48.16.1, Initial Settings.

48.7.2.2 RSCFDnCmCTR — Channel Control Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCmCTR register can be read/written in 32-bit units

RSCFDnCmCTRL, RSCFDnCmCTRLH registers can be read/written in 16-bit units

RSCFDnCmCTRL, RSCFDnCmCTRLH, RSCFDnCmCTRLH, RSCFDnCmCTRLH registers can be read/written in 8-bit units

Address: RSCFDnCmCTR: <RSCFDn_base> + H'0004 + (H'10 × m)

RSCFDnCmCTRL: <RSCFDn_base> + H'0004 + (H'10 × m),

RSCFDnCmCTRLH: <RSCFDn_base> + H'0006 + (H'10 × m)

RSCFDnCmCTRL: <RSCFDn_base> + H'0004 + (H'10 × m),

RSCFDnCmCTRLH: <RSCFDn_base> + H'0005 + (H'10 × m),

RSCFDnCmCTRLH: <RSCFDn_base> + H'0006 + (H'10 × m),

RSCFDnCmCTRLH: <RSCFDn_base> + H'0007 + (H'10 × m)

Value after reset: H'0000_0005

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|-------|-------|-----------|------|------|----------|-----|-----|----|------|-------|------------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | CRCT | — | — | — | CTMS[1:0] | CTME | ERRD | BOM[1:0] | — | — | — | — | — | — | TAIE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | ALIE | BLIE | OLIE | BORIE | BOEIE | EPIE | EWIE | BEIE | — | — | — | — | RTBO | CSLPR | CHMDC[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 31 | Reserved | B'0 | R | This bit is read as the value after reset. The write value should be the value after reset. |
| 30 | CRCT | B'0 | R/W | <p>CRC Error Test Enable</p> <p>This bit is used to test the CRC generator circuit in the RS-CANFD module. Setting this bit to 1 allows the first bit in the ID field to be reversed when a message is received. Due to the bit reversal, the CRC calculation result does not agree with the correct CRC value of the received frame. A CRC error is thus detected (the CERR bit in the RSCFDnCmERFL register is 1). Note the following when using this function:</p> <ul style="list-style-type: none"> This function is available when the CTME bit in the RSCFDnCmCTR register is 1 (communication test mode enabled). Communicating with another CAN node is impossible. Use the function in the inter-channel communication test (the CmICBCE bit in the RSCFDnGTSTCFG register is 1). Bit reversal in the ID field may violate the bit stuffing rule. In this case, a stuff error is detected instead of a CRC error. <p>Modify this bit only in channel standby mode. In channel standby mode, this bit is 0.</p> <p>0: The first bit in the receive ID field is not reversed. 1: The first bit in the receive ID field is reversed.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 29 to 27 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 26, 25 | CTMS[1:0] | B'00 | R/W | Communication Test Mode Select These bits are used to select a communication test mode. Modify these bits only in channel halt mode. These bits are set to 0 in channel reset mode. B'00: Standard test mode B'01: Listen-only mode B'10: Self-test mode 0 (external loopback mode) B'11: Self-test mode 1 (internal loopback mode) |
| 24 | CTME | B'0 | R/W | Communication Test Mode Enable Setting this bit to 1 enables communication test mode. Modify these bits in channel halt mode. This bit is set to 0 in channel reset mode. 0: Communication test mode is disabled. 1: Communication test mode is enabled. |
| 23 | ERRD | B'0 | R/W | Error Display Mode Select This bit is used to control the display mode of bits 14 to 8 in the RSCFDnCmERFL register. When this bit is clear to 0, only the flags of the first error event are set to 1. If two or more errors occur in the first error event, all the flags of the detected errors are set to 1. When this bit is set to 1, all the flags of errors that have occurred are set to 1 regardless of the error occurrence order. Modify this bit only in channel reset mode or channel halt mode. 0: Error flags are displayed only for the first error information after bits 14 to 8 in RSCFDnCmERFL are all cleared. 1: Error flags for all error information are displayed. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 22, 21 | BOM[1:0] | B'00 | R/W | <p>Bus Off Recovery Mode Select</p> <p>These bits are used to select the bus off recovery mode of the RS-CANFD module.</p> <p>When the BOM[1:0] bits are set to B'00, return from the bus off state to the error active state is compliant with the CAN specifications. That is, the RS-CANFD module reenters the CAN communication (error active state) after 11 consecutive recessive bits are detected 128 times. A bus off recovery interrupt request is generated at the time of return from the bus off state. Even if the CHMDC[1:0] bits are set to B'10 (channel halt mode) before recessive bits are detected 128 times, the RS-CANFD module does not transition to channel halt mode until recessive bits are detected 128 times.</p> <p>When the module reaches the bus off state when the BOM[1:0] bits are set to B'01, the CHMDC[1:0] bits in the RSCFDnCTR register ($m = 0$ or 1) are set to B'10 and the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated and the TEC[7:0] and REC[7:0] bits in the RSCFDnCMSTS register are cleared to H'00.</p> <p>When the RS-CANFD module reaches the bus off state when the BOM[1:0] bits are set to B'10, the CHMDC[1:0] bits are set to B'10 and the RS-CANFD module transitions to channel halt mode after return from the bus off state (11 consecutive recessive bits are detected 128 times). A bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to H'00.</p> <p>When the BOM[1:0] bits are set to B'11 and the CHMDC[1:0] bits are set to B'10 while the RS-CANFD module is in the bus off state, the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to H'00. However, if 11 consecutive recessive bits are detected 128 times and the RS-CANFD module has recovered to the error active state from the bus off state before the CHMDC[1:0] bits are set to B'10, a bus off recovery interrupt request is generated.</p> <p>If a program writes to the CHMDC[1:0] bit at the same time as the RS-CANFD module transition to channel halt mode (at bus off entry when the BOM[1:0] bits are B'01 or at bus off end when the BOM[1:0] bits are B'10), the program's writing takes precedence. Modify the BOM[1:0] bits only in channel reset mode.</p> <p>B'00: ISO11898-1 compliant B'01: Entry to channel halt mode automatically at bus-off entry B'10: Entry to channel halt mode automatically at bus-off end B'11: Entry to channel halt mode (in bus-off state) by program request</p> |
| 20 to 17 | Reserved | All 0 | R | <p>These bits are read as the value after reset.</p> <p>The write value should be the value after reset.</p> |
| 16 | TAIE | B'0 | R/W | <p>Transmit Abort Interrupt Enable</p> <p>When transmit abort of the transmit buffer is completed with the TAIE bit set to 1, an interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Transmit abort interrupt is disabled. 1: Transmit abort interrupt is enabled.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 15 | ALIE | B'0 | R/W | <p>Arbitration Lost Interrupt Enable</p> <p>When the ALF flag in the RSCFDnCmERFL register is set to 1 with the ALIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Arbitration lost interrupt is disabled. 1: Arbitration lost interrupt is enabled.</p> |
| 14 | BLIE | B'0 | R/W | <p>Bus Lock Interrupt Enable</p> <p>When the BLF flag in the RSCFDnCmERFL register is set to 1 with the BLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Bus lock interrupt is disabled. 1: Bus lock interrupt is enabled.</p> |
| 13 | OLIE | B'0 | R/W | <p>Overload Frame Transmit Interrupt Enable</p> <p>When the OVLF flag in the RSCFDnCmERFL register is set to 1 with the OLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Overload frame transmit interrupt is disabled. 1: Overload frame transmit interrupt is enabled.</p> |
| 12 | BORIE | B'0 | R/W | <p>Bus Off Recovery Interrupt Enable</p> <p>When the BORF flag in the RSCFDnCmERFL register is set to 1 with the BORIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Bus off recovery interrupt is disabled. 1: Bus off recovery interrupt is enabled.</p> |
| 11 | BOEIE | B'0 | R/W | <p>Bus Off Entry Interrupt Enable</p> <p>When the BOEF flag in the RSCFDnCmERFL register is set to 1 with the BOEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Bus off entry interrupt is disabled. 1: Bus off entry interrupt is enabled.</p> |
| 10 | EPIE | B'0 | R/W | <p>Error Passive Interrupt Enable</p> <p>When the EPF flag in the RSCFDnCmERFL register is set to 1 with the EPIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Error passive interrupt is disabled. 1: Error passive interrupt is enabled.</p> |
| 9 | EWIE | B'0 | R/W | <p>Error Warning Interrupt Enable</p> <p>When the EWF flag in the RSCFDnCmERFL register is set to 1 with the EWIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Error warning interrupt is disabled. 1: Error warning interrupt is enabled.</p> |
| 8 | BEIE | B'0 | R/W | <p>Bus Error Interrupt Enable</p> <p>When the BEF flag in the RSCFDnCmERFL register is set to 1 with the BEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Bus error interrupt is disabled. 1: Bus error interrupt is enabled.</p> |
| 7 to 4 | Reserved | All 0 | R | <p>These bits are read as the value after reset.</p> <p>The write value should be the value after reset.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|------------|---------------|-----|--|
| 3 | RTBO | B'0 | R/W | <p>Forcible Return from Bus-off</p> <p>Setting this bit to 1 in the bus off state forcibly returns the state from the bus off state to the error active state. This bit is automatically cleared to 0. Setting this bit to 1 clears the TEC[7:0] and REC[7:0] bits in the RSCFDnCmSTS register to H'00 and also clears the BOSTS flag in the RSCFDnCmSTS register to 0 (not in bus off state). The other registers remain unchanged. No bus off recovery interrupt request is generated upon return from the bus off state in this case. Use this bit only when the BOM[1:0] bits in the RSCFDnCmCTR are B'00 (ISO11898-1 compliant).</p> <p>A delay of up to 1 CAN bit time occurs after the RTBO bit is set to 1 until the RS-CANFD module transitions to the error active state. Set this bit to 1 in channel communication mode.</p> |
| 2 | CSLPR | B'1 | R/W | <p>Channel Stop Mode</p> <p>Setting this bit to 1 places the channel into channel stop mode. Clearing this bit to 0 makes the channel exit channel stop mode. This bit should not be modified in channel communication mode or channel wait mode.</p> <p>0: Other than channel stop mode 1: Channel stop mode</p> |
| 1, 0 | CHMDC[1:0] | B'01 | R/W | <p>Mode Select</p> <p>These bits are used to select a channel mode (channel communication mode, channel reset mode, or channel halt mode). For details, see section 48.11.2, Channel Modes. Setting the CSLPR bit to 1 in channel reset mode allows transition to channel stop mode. Do not set the CHMDC[1:0] bits to B'11.</p> <p>When the RS-CANFD module has automatically transitioned to channel halt mode based on the setting of the BOM[1:0] bits, the CHMDC[1:0] bits automatically become B'10.</p> <p>B'00: Channel communication mode B'01: Channel reset mode B'10: Channel halt mode B'11: Setting prohibited</p> |

48.7.2.3 RSCFDnCmSTS — Channel Status Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCmSTS register can be read only in 32-bit units

RSCFDnCmSTSL, RSCFDnCmSTSH registers can be read only in 16-bit units

RSCFDnCmSTSLL, RSCFDnCmSTSLH, RSCFDnCmSTSHL, RSCFDnCmSTSHH registers can be read only in 8-bit units

Address: RSCFDnCmSTS: <RSCFDn_base> + H'0008 + (H'10 × m)

RSCFDnCmSTSL: <RSCFDn_base> + H'0008 + (H'10 × m),

RSCFDnCmSTSH: <RSCFDn_base> + H'000A + (H'10 × m)

RSCFDnCmSTSLL: <RSCFDn_base> + H'0008 + (H'10 × m),

RSCFDnCmSTSLH: <RSCFDn_base> + H'0009 + (H'10 × m),

RSCFDnCmSTSHL: <RSCFDn_base> + H'000A + (H'10 × m),

RSCFDnCmSTSHH: <RSCFDn_base> + H'000B + (H'10 × m)

Value after reset: H'0000_0005

| | | | | | | | | | | | | | | | | |
|----------------|----------|----|----|----|----|----|----|----|----------|--------|--------|-------|-------|--------|--------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TEC[7:0] | | | | | | | | REC[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | COMSTS | RECSTS | TRMSTS | BOSTS | EPSTS | CSLPTS | CHLPTS | CRSTS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 24 | TEC[7:0] | H'00 | R | The transmit error counter (TEC) can be read. These bits contain the transmit error counter value. For transmit error counter increment/decrement conditions, see the CAN specification (ISO11898-1). These bits are cleared to 0 in channel reset mode. |
| 23 to 16 | REC[7:0] | H'00 | R | The receive error counter (REC) can be read. These bits contain the receive error counter value. For receive error counter increment/decrement conditions, see the CAN specifications (ISO11898-1). These bits are cleared to 0 in channel reset mode. |
| 15 to 8 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 7 | COMSTS | B'0 | R | Communication Status Flag This bit indicates that communication is ready. This flag becomes 1 when the RS-CANFD module has detected 11 consecutive recessive bits after it has transitioned from channel reset mode or channel halt mode to channel communication mode. This flag is cleared to 0 in channel reset mode or channel halt mode. 0: Communication is not ready. 1: Communication is ready. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 6 | RECSTS | B'0 | R | <p>Receive Status Flag</p> <p>This flag is set to 1 when reception has started, and is cleared to 0 when the bus has become idle or transmission has started.</p> <p>0: Bus idle, in transmission or bus off state 1: In reception</p> |
| 5 | TRMSTS | B'0 | R | <p>Transmit Status Flag</p> <p>This flag is set to 1 when transmission has started, and is cleared to 0 when the bus has become idle or reception has started. This flag remains 1 in the bus off state.</p> <p>0: Bus idle or in reception 1: In transmission or bus off state</p> |
| 4 | BOSTS | B'0 | R | <p>Bus Off Status Flag</p> <p>This flag is set to 1 when the bus off state ($TEC[7:0] > 255$) is entered. It is cleared to 0 when the RS-CANFD module has exited the bus off state.</p> <p>0: Not in bus off state 1: In bus off state</p> |
| 3 | EPSTS | B'0 | R | <p>Error Passive Status Flag</p> <p>This flag is set to 1 when the RS-CANFD module has entered the error passive state ($(128 \leq TEC[7:0] \leq 255)$ or $(128 \leq REC[7:0])$), It is cleared to 0 when the RS-CANFD module has exited the error passive state or has entered channel reset mode.</p> <p>0: Not in error passive state 1: In error passive state</p> |
| 2 | CSLPSTS | B'1 | R | <p>Channel Stop Status Flag</p> <p>This flag is set to 1 when the RS-CANFD module has transitioned to channel stop mode, and is cleared to 0 when the RS-CANFD module has returned from channel stop mode.</p> <p>0: Not in channel stop mode 1: In channel stop mode</p> |
| 1 | CHLTSTS | B'0 | R | <p>Channel Halt Status Flag</p> <p>This flag is set to 1 when the RS-CANFD module has transitioned to channel halt mode, and is cleared to 0 when the RS-CANFD module has returned from channel halt mode.</p> <p>0: Not in channel halt mode 1: In channel halt mode</p> |
| 0 | CRSTSTS | B'1 | R | <p>Channel Reset Status Flag</p> <p>This flag is set to 1 when the RS-CANFD module has transitioned to channel reset mode, and is cleared to 0 when the RS-CANFD module has transitioned to channel communication mode or channel halt mode. This flag remains 1 when the RS-CANFD module transitions from channel reset mode to channel stop mode.</p> <p>0: Not in channel reset mode 1: In channel reset mode</p> |

48.7.2.4 RSCFDnCmERFL — Channel Error Flag Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCmERFL register can be read/written in 32-bit units
 RSCFDnCmERFLL, RSCFDnCmERFLH registers can be read/written in 16-bit units
 RSCFDnCmERFLLL, RSCFDnCmERFLLH, RSCFDnCmERFLHL, RSCFDnCmERFLHH registers can be read/written in 8-bit units

Address: RSCFDnCmERFL: <RSCFDn_base> + H'000C + (H'10 × m)
 RSCFDnCmERFLL: <RSCFDn_base> + H'000C + (H'10 × m),
 RSCFDnCmERFLH: <RSCFDn_base> + H'000E + (H'10 × m)
 RSCFDnCmERFLLL: <RSCFDn_base> + H'000C + (H'10 × m),
 RSCFDnCmERFLLH: <RSCFDn_base> + H'000D + (H'10 × m),
 RSCFDnCmERFLHL: <RSCFDn_base> + H'000E + (H'10 × m),
 RSCFDnCmERFLHH: <RSCFDn_base> + H'000F + (H'10 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|--------------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CRCREG[14:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | ADERR | B0ERR | B1ERR | CERR | AERR | FERR | SERR | ALF | BLF | OVLf | BORF | BOEF | EPF | EWf | BEF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* |

Note: * The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|------|--|
| 31 | Reserved | B'0 | R | When read, the value after reset is returned. When writing to this bit, write the value after reset. |
| 30 to 16 | CRCREG [14:0] | H'0000 | R | CRC Calculation Data When the CTME bit in the RSCFDnCmCTR register is set to 1 (communication test mode is enabled), the CRC value calculated based on the transmit or receive message can be read. When the CTME bit is set to 0 (communication test mode is disabled), these bits are always read as 0. |
| 15 | Reserved | B'0 | R | When read, the value after reset is returned. When writing to this bit, write the value after reset. |
| 14 | ADERR | B'0 | R/W* | ACK Delimiter Error Flag This flag is set to 1 when a form error has been detected in the ACK delimiter during transmission. 0: No ACK delimiter error is detected. 1: ACK delimiter error is detected. |
| 13 | B0ERR | B'0 | R/W* | Dominant Bit Error Flag This flag is set to 1 when a recessive bit has been detected though a dominant bit was transmitted. 0: No dominant bit error is detected. 1: Dominant bit error is detected. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 12 | B1ERR | B'0 | R/W* | <p>Recessive Bit Error Flag</p> <p>This flag is set to 1 when a dominant bit has been detected though a recessive bit was transmitted.</p> <p>0: No recessive bit error is detected.</p> <p>1: Recessive bit error is detected.</p> |
| 11 | CERR | B'0 | R/W* | <p>CRC Error Flag</p> <p>This flag is set to 1 when a CRC error has been detected.</p> <p>0: No CRC error is detected.</p> <p>1: CRC error is detected.</p> |
| 10 | AERR | B'0 | R/W* | <p>ACK Error Flag</p> <p>This flag is set to 1 when an ACK error has been detected.</p> <p>0: No ACK error is detected.</p> <p>1: ACK error is detected.</p> |
| 9 | FERR | B'0 | R/W* | <p>Form Error Flag</p> <p>This flag is set to 1 when a form error has been detected.</p> <p>0: No form error is detected.</p> <p>1: Form error is detected.</p> |
| 8 | SERR | B'0 | R/W* | <p>Stuff Error Flag</p> <p>This flag is set to 1 when a stuff error has been detected.</p> <p>0: No stuff error is detected.</p> <p>1: Stuff error is detected.</p> |
| 7 | ALF | B'0 | R/W* | <p>Arbitration-lost Flag</p> <p>This flag is set to 1 when an arbitration-lost has been detected.</p> <p>0: No arbitration-lost is detected.</p> <p>1: Arbitration-lost is detected.</p> |
| 6 | BLF | B'0 | R/W* | <p>Bus Lock Flag</p> <p>This flag is set to 1 when 32 consecutive dominant bits have been detected on the CAN bus in channel communication mode. After that, detection of a dominant lock is restarted when either of the following conditions is met.</p> <ul style="list-style-type: none"> • A recessive bit is detected after the BLF bit has been cleared from 1 to 0. • The RS-CANFD module transitions to channel reset mode and returns to channel communication mode after the BLF bit has been cleared from 1 to 0. <p>0: No channel bus lock is detected.</p> <p>1: Channel bus is lock detected.</p> |
| 5 | OVLFL | B'0 | R/W* | <p>Overload Flag</p> <p>This flag is set to 1 when the overload frame transmit condition has been detected when performing reception or transmission.</p> <p>0: No overload is detected.</p> <p>1: Overload is detected.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|---|
| 4 | BORF | B'0 | R/W* | <p>Bus Off Recovery Flag</p> <p>This flag is set to 1 when 11 consecutive recessive bits have been detected 128 times and the RS-CANFD module returns from the bus off state. However, this flag is not set to 1 if the RS-CANFD module returns from the bus off state in any of the following ways before 11 consecutive recessive bits are detected 128 times.</p> <ul style="list-style-type: none"> The CHMDC[1:0] bits in the RSCFDnCmCTR register are set to B'01 (channel reset mode). The RTBO bit in the RSCFDnCmCTR register is set to 1 (forcible return from the bus off state is made). The BOM[1:0] bits in the RSCFDnCmCTR register are set to B'01 (transition to channel halt mode at bus off entry). The CHMDC[1:0] bits in the RSCFDnCmCTR register are set to B'10 (channel halt mode) before 11 consecutive recessive bits are detected 128 times with the BOM[1:0] bits set to B'11 (transition to channel halt mode upon a request from the program during bus off). <p>0: No bus off recovery is detected. 1: Bus off recovery is detected.</p> |
| 3 | BOEF | B'0 | R/W* | <p>Bus Off Entry Flag</p> <p>This flag is set to 1 when the bus off state is reached (TEC[7:0] value > 255). This flag is also set to 1 if the bus off state is reached when the BOM[1:0] bits in the RSCFDnCmCTR register (m = 0 or 1) set to B'01 (transition to channel halt mode at bus off entry).</p> <p>0: No bus off entry is detected. 1: Bus off entry is detected.</p> |
| 2 | EPF | B'0 | R/W* | <p>Error Passive Flag</p> <p>This flag becomes 1 when the error passive state is reached (REC[7:0] or TEC[7:0] value > 127). This flag becomes 1 only when the REC[7:0] or TEC[7:0] value first exceeds 127. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 127, this bit is not set to 1 until both REC[7:0] and TEC[7:0] values become 127 or less and then the REC[7:0] or TEC[7:0] value exceeds 127 again.</p> <p>0: No error passive is detected. 1: Error passive is detected.</p> |
| 1 | EWF | B'0 | R/W* | <p>Error Warning Flag</p> <p>This flag is set to 1 only when the REC[7:0] or TEC[7:0] value first exceeds 95. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 95, this bit is not set to 1 until both REC[7:0] and TEC[7:0] values become 95 or less and then the REC[7:0] or TEC[7:0] value exceeds 95 again.</p> <p>0: No error warning is detected. 1: Error warning is detected.</p> |
| 0 | BEF | B'0 | R/W* | <p>Bus Error Flag</p> <p>This flag is set to 1 when any one of the ADERR, B0ERR, B1ERR, CERR, AERR, FERR, and SERR flags in the RSCFDnCmERFL register is set to 1.</p> <p>0: No channel bus error is detected. 1: Channel bus error is detected.</p> |

Note: To clear the flag of this register to 0, use a store instruction to write "0" to the given flag and "1" to the other flags.

See the CAN specification (ISO11898-1) for a description of error occurrence conditions. To clear each flag of this register, the program must write a 0 to the corresponding bit. These flags cannot be set to 1 by the program. If any of these flags is set to 0 at the same time that the program writes 0 to the flag, the flag is still set to 1. The channel reset mode transition clears all of these flags to 0.

If the ERRD bit in the RSCFDnCMCTR register is set to 0 (ie, only the flags for the first error event are displayed) and an error related to bits 14 to 8 of RSCFDnCMERFL is detected, the flag bits are only set by the error event if bits 14 to 8 were all 0 at the when time the error occurred.

48.7.3 Details of Global Related Registers

48.7.3.1 RSCFDnGCFG - Global Configuration Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnGCFG register can be read/written in 32-bit units

RSCFDnGCFGL, RSCFDnGCFGH registers can be read/written in 16-bit units

RSCFDnGCFGLL, RSCFDnGCFGLH, RSCFDnGCFGHL, RSCFDnGCFGHH registers can be read/written in 8-bit units

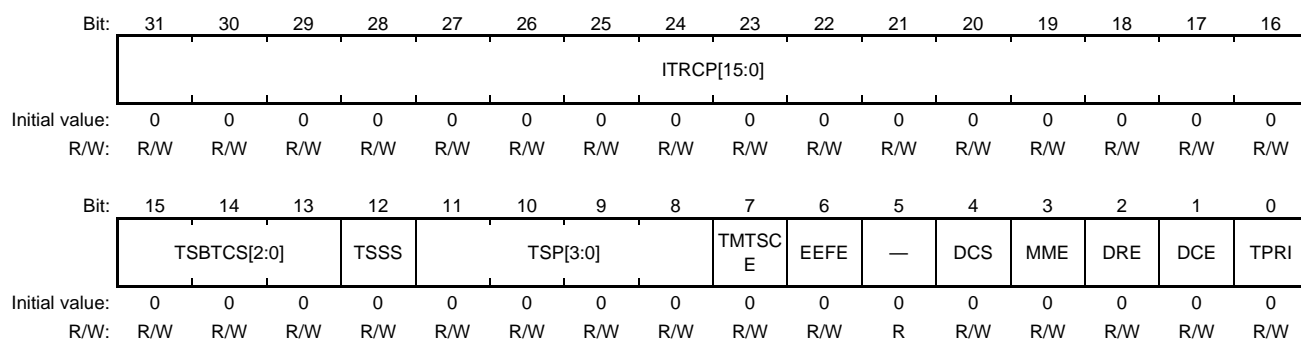
Address: RSCFDnGCFG: <RSCFDn_base> + H'0084

RSCFDnGCFGL: <RSCFDn_base> + H'0084, RSCFDnGCFGH: <RSCFDn_base> + H'0086

RSCFDnGCFGLL: <RSCFDn_base> + H'0084, RSCFDnGCFGLH: <RSCFDn_base> + H'0085,

RSCFDnGCFGHL: <RSCFDn_base> + H'0086, RSCFDnGCFGHH: <RSCFDn_base> + H'0087

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 16 | ITRCP[15:0] | H'0000 | R/W | Interval Timer Prescaler Set These bits are used to set a clock source division value of the interval timer for FIFO buffers. See section 48.13.3.1, Interval Transmission Function. When these bits are set to M, the pclk is divided by M. Setting H'0000 is prohibited when the interval timer is in use. |
| 15 to 13 | TSBTCS[2:0] | B'000 | R/W | Timestamp Clock Source Select When the TSSS bit is 1, these bits are used to select the channel of the bit time clock that will be the clock source of the timestamp counter. B'000: Channel 0 bit time clock B'001: Channel 1 bit time clock B'010: Setting prohibited B'011: Setting prohibited B'100: Setting prohibited B'101: Setting prohibited B'110: Setting prohibited B'111: Setting prohibited |
| 12 | TSSS | 0 | R/W | Timestamp Source Select This bit is used to select a clock source of the timestamp counter. 0: pclk/2*1 1: Bit time clock |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 11 to 8 | TSP[3:0] | B'0000 | R/W | <p>Timestamp Clock Source Division</p> <p>A clock obtained by dividing the clock source selected with the TSBTCS[2:0] bits and TSSS bit according to the TSP[3:0] bits is used as the timestamp counter count source.</p> <p>B'0000: Not divided B'0001: Divided by 2 B'0010: Divided by 4 B'0011: Divided by 8 B'0100: Divided by 16 B'0101: Divided by 32 B'0110: Divided by 64 B'0111: Divided by 128 B'1000: Divided by 256 B'1001: Divided by 512 B'1010: Divided by 1024 B'1011: Divided by 2048 B'1100: Divided by 4096 B'1101: Divided by 8192 B'1110: Divided by 16384 B'1111: Divided by 32768</p> |
| 7 | TMTSCE | B'0 | R/W | <p>Transmit Timestamp Enable</p> <p>Setting this bit to 1 can store the timestamp of the transmission-completed message into the transmit history buffer. The timestamp is stored in the TMTS[15:0] bits in the RSCFDnTHLACCm register.</p> <p>0: Transmit timestamp is disabled. 1: Transmit timestamp is enabled.</p> |
| 6 | EEFE | B'0 | R/W | <p>ECC Error Flag Enable</p> <p>Setting this bit to 1 sets the EEfm bit in the RSCFDnGERFL register to 1 if an ECC2 bit error is detected as a result of a transmit priority selection. Here, the message in which the ECC2 bit error is detected is not transmitted.</p> <p>0: ECC error flag is disabled. 1: ECC error flag is enabled.</p> |
| 5 | Reserved | B'0 | R | <p>When read, the value after reset is returned.</p> <p>When writing to this bit, write the value after reset.</p> |
| 4 | DCS | B'0 | R/W | <p>CAN Clock Source Select*2</p> <p>When this bit is set to 0, clk is used as the clock source of the CAN clock (fCAN).</p> <p>When this bit is set to 1, clk_xincan is used as the clock source of the CAN clock (fCAN).</p> <p>For the CAN clock frequency settings, see section 48.16.1.3, Setting Communication Speed.</p> <p>0: clk 1: clk_xincan</p> |
| 3 | MME | B'0 | R/W | <p>Mirror Function Enable</p> <p>Setting this bit to 1 makes the mirror function available.</p> <p>0: Mirror function is disabled. 1: Mirror function is enabled.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | DRE | B'0 | R/W | <p>DLC Replacement Enable</p> <p>When the DRE bit is set to 1, the DLC value of the receive rule is stored in the buffer instead of the DLC value of the received message after the DLC value has passed through the DLC filter. In this case, a value of H'00 is stored in each data byte beyond the DLC value of the receive rule.</p> <p>The DLC replacement function is only available when the DCE bit is set to 1 (DLC check is enabled).</p> <p>0: DLC replacement is disabled. 1: DLC replacement is enabled.</p> |
| 1 | DCE | B'0 | R/W | <p>DLC Check Enable</p> <p>Setting this bit to 1 makes the DLC check function available. When disabling the DLC check function, set the GAFLDLC[3:0] bits in the RSCFDnGAFLPO_j register to B'0000 before clearing the DCE bit in the RSCFDnGCFG register to 0.</p> <p>0: DLC check is disabled. 1: DLC check is enabled.</p> |
| 0 | TPRI | B'0 | R/W | <p>Transmit Priority Select</p> <p>This bit is used to set the transmit priority.</p> <p>When this bit is set to 0, ID priority is selected and the transmit priority complies with the CAN bus arbitration rule (ISO11898-1 specifications). When this bit is set to 1, transmit buffer number priority is selected and the lowest transmit buffer number of those has the highest priority.</p> <p>While the transmit queue is in use, this bit should be set to 0.</p> <p>0: ID priority 1: Transmit buffer number priority</p> |

- Notes:
1. When specifying pclk/2 as the timestamp counter count source, set bits TSBTCS[2:0] to B'000.
 2. For the CAN clock frequency settings, see section 48.16.1.3, Setting Communication Speed.

Modify the RSCFDnGCFG register only in global reset mode.

48.7.3.2 RSCFDnGCTR — Global Control Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnGCTR register can be read/written in 32-bit units

RSCFDnGCTRL, RSCFDnGCTRH registers can be read/written in 16-bit units

RSCFDnGCTRL, RSCFDnGCTRLH, RSCFDnGCTRHL, RSCFDnGCTRHH registers can be read/written in 8-bit units

Address: RSCFDnGCTR: <RSCFDn_base> + H'0088

RSCFDnGCTRL: <RSCFDn_base> + H'0088, RSCFDnGCTRH: <RSCFDn_base> + H'008A

RSCFDnGCTRL: <RSCFDn_base> + H'0088, RSCFDnGCTRLH: <RSCFDn_base> + H'0089,

RSCFDnGCTRHL: <RSCFDn_base> + H'008A, RSCFDnGCTRHH: <RSCFDn_base> + H'008B

Value after reset: H'0000_0005

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|--------|------|------|----|----|----|----|----|-------|-----------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | TSRST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | THLEIE | MEIE | DEIE | — | — | — | — | — | GSLPR | GMDC[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 17 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 16 | TSRST | B'0 | R/W | Timestamp Counter Reset This bit is used to reset the timestamp counter. When this bit is set to 1, the RSCFDnGTSC register is cleared to H'0000. This bit is always read as 0. |
| 15 to 11 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 10 | THLEIE | B'0 | R/W | Transmit History Buffer Overflow Interrupt Enable When the THLEIE bit is set to 1 and the THLES flag in the RSCFDnGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode. 0: Transmit history buffer overflow interrupt is disabled. 1: Transmit history buffer overflow interrupt is enabled. |
| 9 | MEIE | B'0 | R/W | FIFO Message Lost Interrupt Enable When the MEIE bit is set to 1 and the MES flag in the RSCFDnGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode. 0: FIFO message lost interrupt is disabled. 1: FIFO message lost interrupt is enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|--|
| 8 | DEIE | B'0 | R/W | <p>DLC Error Interrupt Enable</p> <p>When the DEIE bit is set to 1 and the DEF flag in the RSCFDnGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.</p> <p>0: DLC error interrupt is disabled. 1: DLC error interrupt is enabled.</p> |
| 7 to 3 | Reserved | All 0 | R | <p>When read, the value after reset is returned.</p> <p>When writing to these bits, write the value after reset.</p> |
| 2 | GSLPR | B'1 | R/W | <p>Global Stop Mode</p> <p>Setting this bit to 1 places the RS-CANFD module into global stop mode.</p> <p>Clearing this bit to 0 makes the RS-CANFD module leave from global stop mode.</p> <p>This bit should not be modified in global operating mode or global test mode.</p> <p>0: Other than global stop mode 1: Global stop mode</p> |
| 1, 0 | GMDC[1:0] | B'01 | R/W | <p>Global Mode Select</p> <p>These bits are used to select the mode of entire RS-CANFD module (global operating mode, global reset mode, or global test mode). For details, see section 48.11.1, Global Modes. Setting the GSLPR bit to 1 when in global reset mode places the RS-CANFD module into global stop mode.</p> <p>B'00: Global operating mode B'01: Global reset mode B'10: Global test mode B'11: Setting prohibited</p> |

48.7.3.3 RSCFDnGSTS — Global Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnGSTS register can be read only in 32-bit units

RSCFDnGSTSL, RSCFDnGSTSH registers can be read only in 16-bit units

RSCFDnGSTSLL, RSCFDnGSTSLH, RSCFDnGSTSHL, RSCFDnGSTSHH registers can be read only in 8-bit units

Address: RSCFDnGSTS: <RSCFDn_base> + H'008C

RSCFDnGSTSL: <RSCFDn_base> + H'008C, RSCFDnGSTSH: <RSCFDn_base> + H'008E

RSCFDnGSTSLL: <RSCFDn_base> + H'008C, RSCFDnGSTSLH: <RSCFDn_base> + H'008D,

RSCFDnGSTSHL: <RSCFDn_base> + H'008E, RSCFDnGSTSHH: <RSCFDn_base> + H'008F

Value after reset: H'0000_000D

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|--------------|-------------|------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | GRAMI NIT | GSLPS TS | GHLT TS | GRSTS TS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 4 | Reserved | All 0 | R | When read, the value after reset is returned. |
| 3 | GRAMINIT | B'1 | R | <p>CAN RAM Initialization Status Flag</p> <p>This flag indicates the initialization status of the CAN RAM.</p> <p>This flag is set to 1 after the MCU has been reset, and is cleared to 0 when CAN RAM initialization is completed.</p> <p>0: CAN RAM initialization is completed.</p> <p>1: CAN RAM initialization is ongoing.</p> |
| 2 | GSLPSTS | B'1 | R | <p>Global Stop Status Flag</p> <p>This flag is set to 1 when the RS-CANFD module has transitioned to global stop mode, and is cleared to 0 when the RS-CANFD module has returned from global stop mode.</p> <p>0: Not in global stop mode</p> <p>1: In global stop mode</p> |
| 1 | GHLTSTS | B'0 | R | <p>Global Test Status Flag</p> <p>This flag is set to 1 when the RS-CANFD module has transitioned to global test mode, and is cleared to 0 when the RS-CANFD module has exited global test mode.</p> <p>0: Not in global test mode</p> <p>1: In global test mode</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | GRSTSTS | B'1 | R | Global Reset Status Flag This flag is set to 1 when the RS-CANFD module has transitioned to global reset mode, and is cleared to 0 when the RS-CANFD module has exited global reset mode. This flag remains 1 even when the RS-CANFD module has transitioned from global reset mode to global stop mode. 0: Not in global reset mode 1: In global reset mode |

48.7.3.4 RSCFDnGERFL — Global Error Flag Register

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

Access: RSCFDnGERFL register can be read/written in 32-bit units

RSCFDnGERFLL, RSCFDnGERFLH registers can be read/written in 16-bit units

RSCFDnGERFLLL, RSCFDnGERFLLH, RSCFDnGERFLHL, RSCFDnGERFLHH registers can be read/written in 8-bit units

Address: RSCFDnGERFL: <RSCFDn_base> + H'0090

RSCFDnGERFLL: <RSCFDn_base> + H'0090, RSCFDnGERFLH: <RSCFDn_base> + H'0092

RSCFDnGERFLLL: <RSCFDn_base> + H'0090, RSCFDnGERFLLH: <RSCFDn_base> + H'0091,

RSCFDnGERFLHL: <RSCFDn_base> + H'0092, RSCFDnGERFLHH: <RSCFDn_base> + H'0093

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | EEF1 | EEF0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W* | R/W* |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | THLES | MES | DEF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W* |

Note: * The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 18 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 17 | EEF1 | B'0 | R/W | ECC Error Flag for Channel 1 The EEF1 flag is set to 1 disabling message transmission if an ECC2 bit error is detected in transmit priority selection of channel 1 when the EEF0 bit in the RSCFDnGCFG register is 1. The program must write 0 to the corresponding flag to be cleared. The program can clear this flag by writing 0 to this bit. 0: No 2-bit ECC error has occurred in transmit priority selection. 1: A 2-bit ECC error has occurred in transmit priority selection. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 16 | EEF0 | B'0 | R/W | <p>ECC Error Flag for channel 0</p> <p>The EEF0 flag is set to 1 disabling message transmission if an ECC2 bit error is detected in transmit priority selection of channel 0 when the EEFE bit in the RSCFDnGCFG register is 1. The program must write 0 to the corresponding flag to be cleared. The program can clear this flag by writing 0 to this bit.</p> <p>0: No 2-bit ECC error has occurred in transmit priority selection. 1: A 2-bit ECC error has occurred in transmit priority selection.</p> |
| 15 to 3 | Reserved | All 0 | R | <p>When read, the value after reset is returned.</p> <p>When writing to these bits, write the value after reset.</p> |
| 2 | THLES | B'0 | R | <p>Transmit History Buffer Overflow Status Flag</p> <p>The THLES flag is set to 1 when any one of the THLELT flags in the RSCFDnTHLSTSm register (m = 0, 1) is set to 1.</p> <p>This flag is cleared to 0 when the THLELT flags of all channels are set to 0.</p> <p>0: No transmit history buffer overflow has occurred. 1: A transmit history buffer overflow has occurred.</p> |
| 1 | MES | B'0 | R | <p>FIFO Message Lost Status Flag</p> <p>The MES flag is set to 1 when any one of the RFMLT flags in the RSCFDnRFSTSk register (x = 0 to 7) or the CFMLT flags in the RSCFDnCFSTSk register (k = 0 to 5) is set to 1.</p> <p>This flag is cleared to 0 when all RFMLT flags and CFMLT flags are set to 0.</p> <p>0: No FIFO message lost error has occurred. 1: A FIFO message lost error has occurred.</p> |
| 0 | DEF | B'0 | R/W | <p>DLC Error Flag</p> <p>The DEF flag is set to 1 when an error has been detected during the DLC check. The program can clear this flag by writing 0 to this bit.</p> <p>0: No DLC error has occurred. 1: A DLC error has occurred.</p> |

Note: To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1".

All flags in the RSCFDnGERFL register are cleared to 0 in global reset mode.

48.7.3.5 RSCFDnGTSC — Global Timestamp Counter Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnGTSC register can be read only in 32-bit units.

RSCFDnGTSCSL, RSCFDnGTSCSCH registers can be read only in 16-bit units.

Address: RSCFDnGTSC: <RSCFDn_base> + H'0094

RSCFDnGTSCSL: <RSCFDn_base> + H'0094, RSCFDnGTSCSCH: <RSCFDn_base> + H'0096

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TS[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 15 to 0 | TS[15:0] | H'0000 | R | <p>Timestamp Value</p> <p>The timestamp counter value can be read.</p> <p>Counter Value: H'0000 to H'FFFF</p> <p>When the TS[15:0] bits are read, the read value shows the timestamp counter (16-bit free-running counter) value at that time. When the SOF is detected, the TS[15:0] value is captured and later stored in the receive buffer or the FIFO buffer. When the TMTSCE bit in the RSCFDnGCFG register is 1, the TS[15:0] value is stored in the transmit history buffer. The timestamp counter is initialized in global reset mode.</p> <p>The timestamp counter starts and stops counting differently, depending on the count source.</p> <ul style="list-style-type: none"> When the TSSS bit in the RSCFDnGCFG register is 0 (pclk): <ul style="list-style-type: none"> The timestamp counter starts counting when the RS-CANFD module has transitioned to global operating mode. This counter stops counting when the RS-CANFD module has transitioned to global stop mode or global test mode. When the TSSS bit is 1 (CANm bit time clock): <ul style="list-style-type: none"> The timestamp counter starts counting when the corresponding channel has transitioned to channel communication mode. This counter stops counting when the corresponding channel has transitioned to channel reset mode or channel halt mode. |

48.7.3.6 RSCFDnGTINTSTS0 — Global TX Interrupt Status Register 0

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnGTINTSTS0 register can be read only in 32-bit units

RSCFDnGTINTSTS0L, RSCFDnGTINTSTS0H registers can be read only in 16-bit units

RSCFDnGTINTSTS0LL, RSCFDnGTINTSTS0LH, RSCFDnGTINTSTS0HL, RSCFDnGTINTSTS0HH registers can be read only in 8-bit units

Address: RSCFDnGTINTSTS0: <RSCFDn_base> + H'0460

RSCFDnGTINTSTS0L: <RSCFDn_base> + H'0460, RSCFDnGTINTSTS0H: <RSCFDn_base> + H'0462

RSCFDnGTINTSTS0LL: <RSCFDn_base> + H'0460, RSCFDnGTINTSTS0LH: <RSCFDn_base> + H'0461,

RSCFDnGTINTSTS0HL: <RSCFDn_base> + H'0462, RSCFDnGTINTSTS0HH: <RSCFDn_base> + H'0463

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-------|--------|-------|-------|-------|----|----|----|-------|--------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | THIF1 | CFTIF1 | TQIF1 | TAIF1 | TSIF1 | — | — | — | THIF0 | CFTIF0 | TQIF0 | TAIF0 | TSIF0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R* | R* | R* | R* | R* | R | R | R | R* | R* | R* | R* | R* |

Note: * This bit is automatically cleared in the global reset or channel reset mode.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 13 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 12 | THIF1 | B'0 | R | Channel 1 Transmit History Interrupt Status Flag When the THLIE bit in the RSCFDnTHLCCm register is set to 1 (transmit history interrupt enabled) and the THLIF bit in the RSCFDnTHLSTSm register is set to 1 (transmit history interrupt request), this bit is set to 1. When the THLIF bit in the RSCFDnTHLSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the THLIE bit is cleared to 0. 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested. |
| 11 | CFTIF1 | B'0 | R | Channel 1 Transmit/receive FIFO Transmit Interrupt Status Flag When the CFTXIE bit in the RSCFDnCFCCk register is set to 1 (transmit/receive FIFO transmit interrupt enabled) and the CFTXIF bit in the RSCFDnCFSTSk register is set to 1 (transmit/receive FIFO transmit interrupt request), this bit is set to 1. When the CFTXIF bit is cleared to 0 under the conditions that this bit can be set to 1, this bit is cleared to 0. This flag is also cleared to 0 when the CFTXIE bit is cleared to 0. 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 10 | TQIF1 | B'0 | R | <p>Channel 1 Transmit Queue Interrupt Status Flag</p> <p>When the TXQIE bit in the RSCFDnTXQCCm register is set to 1 (transmit queue interrupt enabled) and the TXQIF bit in the RSCFDnTXQSTSm register is set to 1 (transmit queue interrupt request), this bit is set to 1.</p> <p>When the TXQIF bit (transmit queue interrupt request) in the RSCFDnTXQSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the TXQIE bit is cleared to 0.</p> <p>0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.</p> |
| 9 | TAIF1 | B'0 | R | <p>Channel 1 Transmit Buffer Abort Interrupt Status Flag</p> <p>This bit is set to 1 when the TAIE bit in the RSCFDnCMCTR register is 1 (transmit abort interrupt enabled) and the TMTRF[1:0] flags in the RSCFDnTMSTSp register are set to B'01 (transmit abort completed).</p> <p>This flag is cleared to 0 when the TMTRF[1:0] flags are cleared to B'00 after the transmit abort is completed.</p> <p>0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.</p> |
| 8 | TSIF1 | B'0 | R | <p>Channel 1 Transmit Buffer Interrupt Status Flag</p> <p>This bit is set to 1 when the TMIEp bit in the RSCFDnTMIECy register is set to 1 (transmit buffer interrupt enabled) and the TMTRF[1:0] flags in the RSCFDnTMSTSp register are set to B'10 (transmit completed without abort request) or B'11 (transmit completed with abort request).</p> <p>When the TMTRF[1:0] flags are cleared to B'00 under the condition that this bit can be set to 1, this flag is cleared to 0. In addition, clearing the TMIEp bit to 0 also clears this flag to 0.</p> <p>0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.</p> |
| 7 to 5 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 4 | THIF0 | B'0 | R | <p>Channel 0 Transmit History Interrupt Status Flag</p> <p>When the THLIE bit in the RSCFDnTHLCCm register is set to 1 (transmit history interrupt enabled) and the THLIF bit in the RSCFDnTHLSTSm register is set to 1 (transmit history interrupt request), this bit is set to 1.</p> <p>When the THLIF bit in the RSCFDnTHLSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the THLIE bit is cleared to 0.</p> <p>0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.</p> |
| 3 | CFTIF0 | B'0 | R | <p>Channel 0 Transmit/receive FIFO Transmit Interrupt Status Flag</p> <p>When the CFTXIE bit in the RSCFDnCFCCk register is set to 1 (transmit/receive FIFO transmit interrupt enabled) and the CFTXIF bit in the RSCFDnCFSTSk register is set to 1 (transmit/receive FIFO transmit interrupt request), this bit is set to 1.</p> <p>When the CFTXIF bit is cleared to 0 under the conditions that this bit can be set to 1, this bit is cleared to 0. This flag is also cleared to 0 when the CFTXIE bit is cleared to 0.</p> <p>0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | TQIF0 | B'0 | R | <p>Channel 0 Transmit Queue Interrupt Status Flag</p> <p>When the TXQIE bit in the RSCFDnTXQCCm register is set to 1 (transmit queue interrupt enabled) and the TXQIF bit in the RSCFDnTXQSTSm register is set to 1 (transmit queue interrupt request), this bit is set to 1.</p> <p>When the TXQIF bit (transmit queue interrupt request) in the RSCFDnTXQSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the TXQIE bit is cleared to 0.</p> <p>0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.</p> |
| 1 | TAIF0 | B'0 | R | <p>Channel 0 Transmit Buffer Abort Interrupt Status Flag</p> <p>This bit is set to 1 when the TAIE bit in the RSCFDnCMCTR register is 1 (transmit abort interrupt enabled) and the TMTRF[1:0] flags in the RSCFDnTMSTSp register are set to B'01 (transmit abort completed).</p> <p>This flag is cleared to 0 when the TMTRF[1:0] flags are cleared to B'00 after the transmit abort is completed.</p> <p>0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.</p> |
| 0 | TSIF0 | B'0 | R | <p>Channel 0 Transmit Buffer Interrupt Status Flag</p> <p>This bit is set to 1 when the TMIEp bit in the RSCFDnTMIECy register is set to 1 (transmit buffer interrupt enabled) and the TMTRF[1:0] flags in the RSCFDnTMSTSp register are set to B'10 (transmit completed without abort request) or B'11 (transmit completed with abort request).</p> <p>When the TMTRF[1:0] flags are cleared to B'00 under the condition that this bit can be set to 1, this flag is cleared to 0. In addition, clearing the TMIEp bit to 0 also clears this flag to 0.</p> <p>0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.</p> |

48.7.3.7 RSCFDnGTINTSTS1 — Global TX Interrupt Status Register 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnGTINTSTS1 register can be read only in 32-bit units

RSCFDnGTINTSTS1L, RSCFDnGTINTSTS1H registers can be read only in 16-bit units

RSCFDnGTINTSTS1LL, RSCFDnGTINTSTS1LH, RSCFDnGTINTSTS1HL, RSCFDnGTINTSTS1HH registers can be read only in 8-bit units

Address: RSCFDnGTINTSTS1: <RSCFDn_base> + H'0464

RSCFDnGTINTSTS1L: <RSCFDn_base> + H'0460, RSCFDnGTINTSTS0H: <RSCFDn_base> + H'0466

RSCFDnGTINTSTS1LL: <RSCFDn_base> + H'0460, RSCFDnGTINTSTS0LH: <RSCFDn_base> + H'0465,

RSCFDnGTINTSTS1HL: <RSCFDn_base> + H'0462, RSCFDnGTINTSTS0HH: <RSCFDn_base> + H'0467

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R* | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | — | All 0 | R | Reserved bits These bits are read as the value after reset. The write value should be the value after reset. |

48.7.4 Details of Receive Rule Related Registers

48.7.4.1 RSCFDnGAFLECTR — Receive Rule Entry Control Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnGAFLECTR register can be read/written in 32-bit units

RSCFDnGAFLECTRL, RSCFDnGAFLECTRH registers can be read/written in 16-bit units

RSCFDnGAFLECTRLL, RSCFDnGAFLECTRLH, RSCFDnGAFLECTRHL, RSCFDnGAFLECTRHH registers can be read/written in 8-bit units

Address: RSCFDnGAFLECTR: <RSCFDn_base> + H'0098

RSCFDnGAFLECTRL: <RSCFDn_base> + H'0098, RSCFDnGAFLECTRH: <RSCFDn_base> + H'009A

RSCFDnGAFLECTRLL: <RSCFDn_base> + H'0098, RSCFDnGAFLECTRLH: <RSCFDn_base> + H'0099,

RSCFDnGAFLECTRHL: <RSCFDn_base> + H'009A, RSCFDnGAFLECTRHH: <RSCFDn_base> + H'009B

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|------------|----|----|----|------------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | AFLDA E | — | — | — | AFLPN[4:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 9 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 8 | AFLDAE | B'0 | R/W | Receive Rule Table Write Enable Setting this bit to 0 disables the write to the receive rule table. After writes to the receive rule table are completed, set this bit to 0 to disable the write to the table. The receive rule table can be read regardless of the value of this bit. Set the AFLDAE bit to 1 only in global reset mode. 0: Receive rule table write is disabled. 1: Receive rule table write is enabled. |
| 7 to 5 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 4 to 0 | AFLPN[4:0] | B'0_0000 | R/W | Receive Rule Table Page Number Configuration These bits are used to set the page number of the receive rule table. Sixteen receive rules can be set per page. A page number can be selected from a range of page 0 (B'0_0000) to page 23 (B'1_0111). Set these bits to a value within the range of B'0_0000 to B'1_0111. |

48.7.4.2 RSCFDnGAFLCFG0 — Receive Rule Configuration Register 0

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnGAFLCFG0 register can be read/written in 32-bit units

RSCFDnGAFLCFG0L, RSCFDnGAFLCFG0H registers can be read/written in 16-bit units

RSCFDnGAFLCFG0LL, RSCFDnGAFLCFG0LH, RSCFDnGAFLCFG0HL, RSCFDnGAFLCFG0HH registers can be read/written in 8-bit units

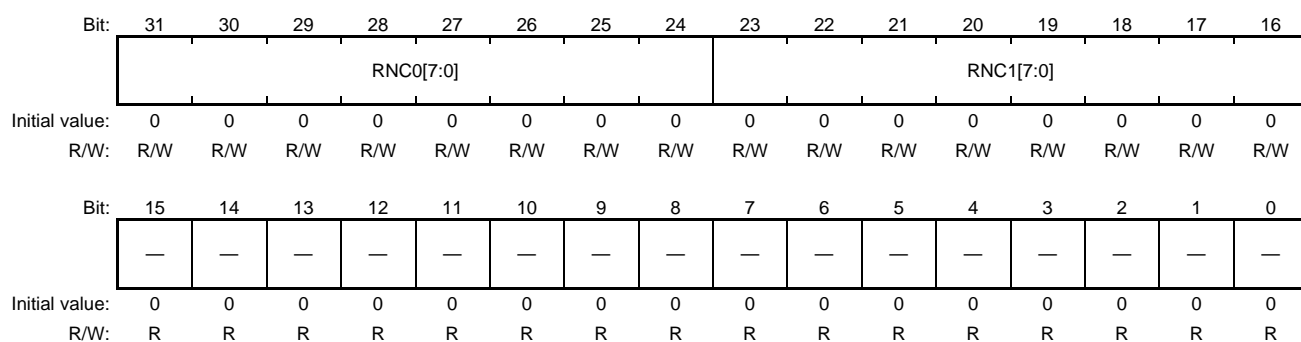
Address: RSCFDnGAFLCFG0: <RSCFDn_base> + H'009C

RSCFDnGAFLCFG0L: <RSCFDn_base> + H'009C, RSCFDnGAFLCFG0H: <RSCFDn_base> + H'009E

RSCFDnGAFLCFG0LL: <RSCFDn_base> + H'009C, RSCFDnGAFLCFG0LH: <RSCFDn_base> + H'009D,

RSCFDnGAFLCFG0HL: <RSCFDn_base> + H'009E, RSCFDnGAFLCFG0HH: <RSCFDn_base> + H'009F

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 24 | RNC0[7:0] | H'00 | R/W | Number of Rules for Channel 0 These bits are used to set the number of rules to be registered in the channel 0 receive rule table. Set these bits to a value within the range of H'00 to H'80. |
| 23 to 16 | RNC1[7:0] | H'00 | R/W | Number of Rules for Channel 1 These bits are used to set the number of rules to be registered in the channel 1 receive rule table. Set these bits to a value within the range of H'00 to H'80. |
| 15 to 0 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |

Modify the RSCFDnGAFLCFG0 register only in global reset mode.

Up to $64 \times$ (number of channels) rules can be registered in the receive rule table as the entire unit. The number of receive rules per channel should meet the following conditions.

The maximum number of rules per channel is 128.

The total of the number of rules allocated to each channel is not larger than the number of rules that can be registered in the entire unit.

48.7.4.3 RSCFDnGAFLCFG1 — Receive Rule Configuration Register 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnGAFLCFG1 register can be read/written in 32-bit units

RSCFDnGAFLCFG1L, RSCFDnGAFLCFG1H registers can be read/written in 16-bit units

RSCFDnGAFLCFG1LL, RSCFDnGAFLCFG1LH, RSCFDnGAFLCFG1HL, RSCFDnGAFLCFG1HH registers can be read/written in 8-bit units

Address: RSCFDnGAFLCFG1: <RSCFDn_base> + H'00A0

RSCFDnGAFLCFG1L: <RSCFDn_base> + H'00A0, RSCFDnGAFLCFG1H: <RSCFDn_base> + H'00A2

RSCFDnGAFLCFG1LL: <RSCFDn_base> + H'00A0, RSCFDnGAFLCFG1LH: <RSCFDn_base> + H'00A1,

RSCFDnGAFLCFG1HL: <RSCFDn_base> + H'00A2, RSCFDnGAFLCFG1HH: <RSCFDn_base> + H'00A3

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |

Modify the RSCFDnGAFLCFG1 register only in global reset mode.

48.7.4.4 RSCFDnGAFLIDj — Receive Rule ID Register (j = 0 to 15)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnGAFLIDj register can be read/written in 32-bit units

RSCFDnGAFLIDjL, RSCFDnGAFLIDjH registers can be read/written in 16-bit units

RSCFDnGAFLIDjLL, RSCFDnGAFLIDjLH, RSCFDnGAFLIDjHL, RSCFDnGAFLIDjHH registers can be read/written in 8-bit units

Address: RSCFDnGAFLIDj: <RSCFDn_base> + H'0500 + (H'10 × j)

RSCFDnGAFLIDjL: <RSCFDn_base> + H'0500 + (H'10 × j),

RSCFDnGAFLIDjH: <RSCFDn_base> + H'0502 + (H'10 × j)

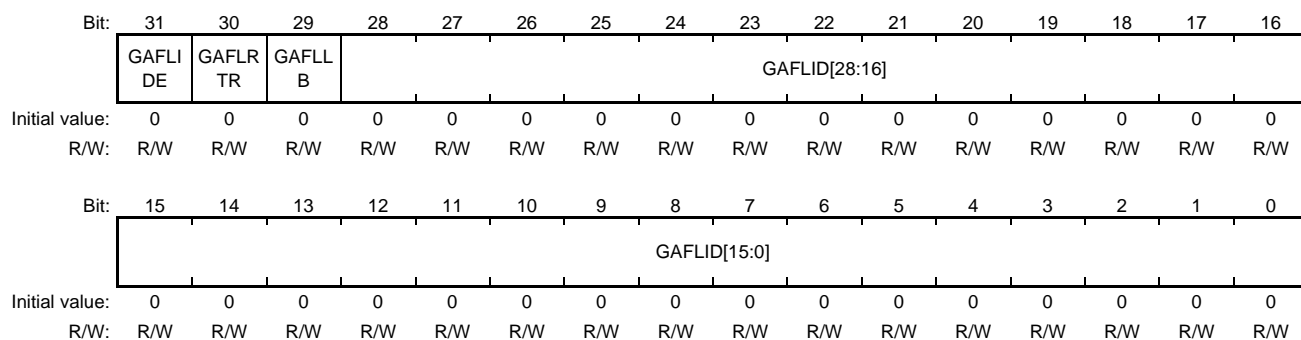
RSCFDnGAFLIDjLL: <RSCFDn_base> + H'0500 + (H'10 × j),

RSCFDnGAFLIDjLH: <RSCFDn_base> + H'0501 + (H'10 × j),

RSCFDnGAFLIDjHL: <RSCFDn_base> + H'0502 + (H'10 × j),

RSCFDnGAFLIDjHH: <RSCFDn_base> + H'0503 + (H'10 × j)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 31 | GAFLIDE | B'0 | R/W | <p>IDE Select</p> <p>This bit is used to select the ID format (standard ID or extended ID) of the receive rule. This bit is compared with the IDE bit in the received message during the acceptance filter processing.</p> <p>0: Standard ID 1: Extended ID</p> |
| 30 | GAFLRTR | B'0 | R/W | <p>RTR Select</p> <p>This bit is used to select the frame format (data frame or remote frame) of the receive rule. This bit is compared with the RTR bit in the received message during the acceptance filter processing.</p> <p>0: Data frame 1: Remote frame</p> |
| 29 | GAFLLB | B'0 | R/W | <p>Receive Rule Target Message Select</p> <p>When this bit is set to 0, data processing using the receive rule is performed when receiving messages transmitted from another CAN node.</p> <p>When this bit is set to 1 when the mirror function is used, data processing using the receive rule is performed when the CAN node is receiving its own transmitted messages.</p> <p>0: When a message transmitted from another CAN node is received 1: When the own transmitted message is received</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|--|
| 28 to 0 | GAFLID[28:0] | H'0000_0000 | R/W | <p>ID Set</p> <p>These bits are used to set the ID field of the receive rule. The ID value set by these bits is compared with the ID of the received message during the acceptance filter processing.</p> <p>For the standard ID, set the ID in bits 10 to 0 and set bits 28 to 11 to 0.</p> |

Modify the RSCFDnGAFLIDj register when the AFLDAE bit in the RSCFDnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

48.7.4.5 RSCFDnGAFLMj — Receive Rule Mask Register (j = 0 to 15)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

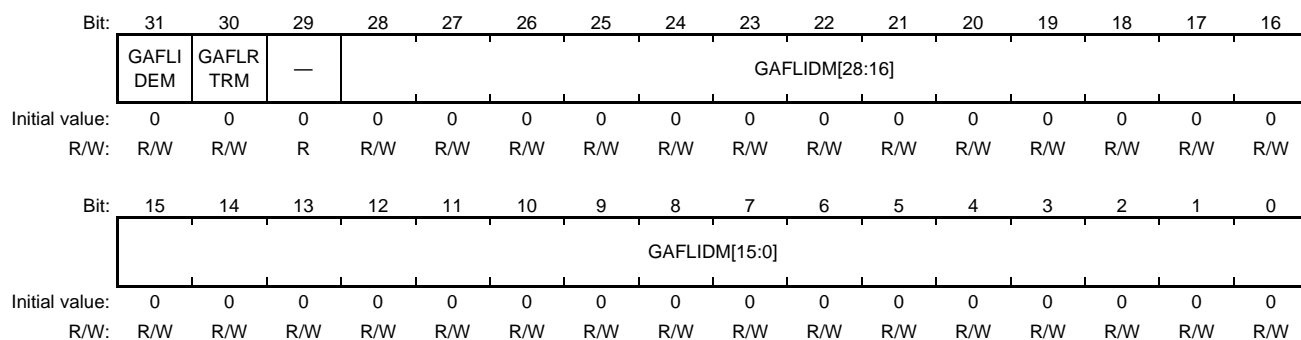
Access: RSCFDnGAFLMj register can be read/written in 32-bit units

RSCFDnGAFLMjL, RSCFDnGAFLMjH registers can be read/written in 16-bit units

RSCFDnGAFLMjLL, RSCFDnGAFLMjLH, RSCFDnGAFLMjHL, RSCFDnGAFLMjHH registers can be read/written in 8-bit units

Address: RSCFDnGAFLMj: <RSCFDn_base> + H'0504 + (H'10 × j)
 RSCFDnGAFLMjL: <RSCFDn_base> + H'0504 + (H'10 × j),
 RSCFDnGAFLMjH: <RSCFDn_base> + H'0506 + (H'10 × j)
 RSCFDnGAFLMjLL: <RSCFDn_base> + H'0504 + (H'10 × j),
 RSCFDnGAFLMjLH: <RSCFDn_base> + H'0505 + (H'10 × j),
 RSCFDnGAFLMjHL: <RSCFDn_base> + H'0506 + (H'10 × j),
 RSCFDnGAFLMjHH: <RSCFDn_base> + H'0507 + (H'10 × j)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|--|
| 31 | GAFLIDEM | B'0 | R/W | <p>IDE Mask</p> <p>When this bit is set to 1, filter processing is performed only for messages of the ID format specified by the GAFLIDE bit in the RSCFDnGAFLIDj register.</p> <p>When this bit is cleared to 0, the IDs of all the receive messages and the specified IDs are regarded as matched. To set the GAFLIDEM bit to 0, set the GAFLIDM[28:0] bits to all 0 at the same time.</p> <p>0: The IDE bit is not compared. 1: The IDE bit is compared.</p> |
| 30 | GAFLRTRM | B'0 | R/W | <p>RTR Mask</p> <p>This bit is used to mask the RTR bit of the receive rule.</p> <p>0: The RTR bit is not compared. 1: The RTR bit is compared</p> |
| 29 | Reserved | B'0 | R | <p>This bit is read as the value after reset. The write value should be the value after reset.</p> |
| 28 to 0 | GAFLIDM [28:0] | H'0000_0000 | R/W | <p>ID Mask</p> <p>These bits are used to mask the corresponding ID bit of the receive rule.</p> <p>0: The corresponding ID bit is not compared. 1: The corresponding ID bit is compared.</p> |

Modify the RSCFDnGAFLMj register when the AFLDAE bit in the RSCFDnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

48.7.4.6 RSCFDnGAFLP0_j — Receive Rule Pointer 0 Register (j = 0 to 15)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnGAFLP0_j register can be read/written in 32-bit units

RSCFDnGAFLP0_jL, RSCFDnGAFLP0_jH registers can be read/written in 16-bit units

RSCFDnGAFLP0_jLL, RSCFDnGAFLP0_jLH, RSCFDnGAFLP0_jHL, RSCFDnGAFLP0_jHH registers can be read/written in 8-bit units

Address: RSCFDnGAFLP0_j: <RSCFDn_base> + H'0508 + (H'10 × j)

RSCFDnGAFLP0_jL: <RSCFDn_base> + H'0508 + (H'10 × j),

RSCFDnGAFLP0_jH: <RSCFDn_base> + H'050A + (H'10 × j)

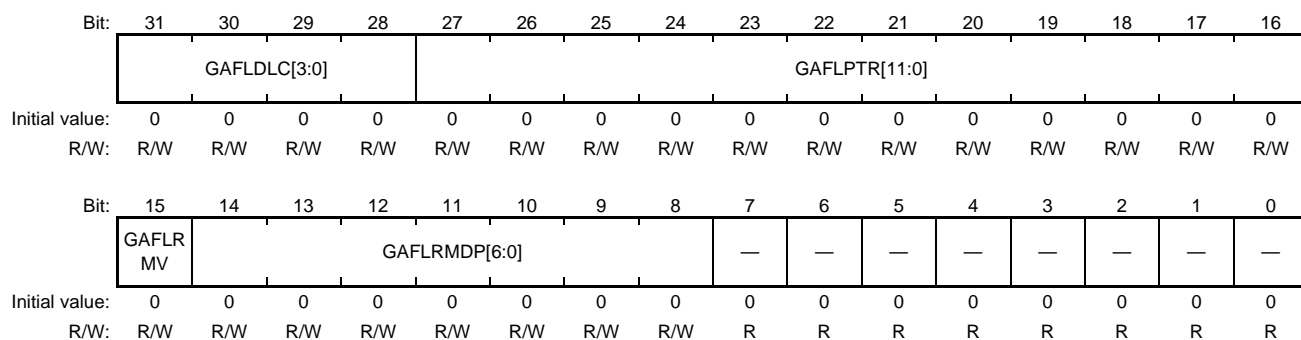
RSCFDnGAFLP0_jLL: <RSCFDn_base> + H'0508 + (H'10 × j),

RSCFDnGAFLP0_jLH: <RSCFDn_base> + H'0509 + (H'10 × j),

RSCFDnGAFLP0_jHL: <RSCFDn_base> + H'050A + (H'10 × j),

RSCFDnGAFLP0_jHH: <RSCFDn_base> + H'050B + (H'10 × j)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------|---------------|-----|---|
| 31 to 28 | GAFLDLC [3:0] | B'0000 | R/W | <p>Receive Rule DLC</p> <p>These bits are used to set the minimum data length necessary for receiving messages. If the data length of a message that is being filtered is equal to or larger than the value set by the GAFLDLC[3:0] bits, the message passes the DLC check. Setting these bits to B'0000 disables the DLC check function allowing messages with any data length to pass the DLC check.</p> <p>B'0000: DLC check is disabled.</p> <p>B'0001: 1 data byte</p> <p>B'0010: 2 data bytes</p> <p>B'0011: 3 data bytes</p> <p>B'0100: 4 data bytes</p> <p>B'0101: 5 data bytes</p> <p>B'0110: 6 data bytes</p> <p>B'0111: 7 data bytes</p> <p>B'1XXX: 8 data bytes</p> |
| 27 to 16 | GAFLPTR [11:0] | H'000 | R/W | <p>Receive Rule Label</p> <p>These bits are used to set a 12-bit label to be attached to messages that have passed through the filter. A label is attached when a message is stored in the receive buffer or the FIFO buffer.</p> <p>Set the 12-bit label information.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------------|---------------|-----|--|
| 15 | GAFLRMV | B'0 | R/W | <p>Receive Buffer Enable</p> <p>When this bit is set to 1, receive messages that have passed through the filter are stored in the receive buffer selected by the GAFLRMDP[6:0] bits.</p> <p>0: No receive buffer is used. 1: A receive buffer is used.</p> |
| 14 to 8 | GAFLRMDP [6:0] | H'00 | R/W | <p>Receive Buffer Number Select</p> <p>These bits are used to select the number of the receive buffer that stores receive messages that have passed through the filter when the GAFLRMV bit is set to 1. Set these bits to a value smaller than the value set by the NRXMB[7:0] bits in the RSCFDnRMNB register.</p> |
| 7 to 0 | Reserved | All 0 | R | <p>These bits are read as the value after reset.</p> <p>The write value should be the value after reset.</p> |

Modify the RSCFDnGAFLP0_j register when the AFLDAE bit in the RSCFDnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

48.7.4.7 RSCFDnGAFLP1_j — Receive Rule Pointer 1 Register (j = 0 to 15)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

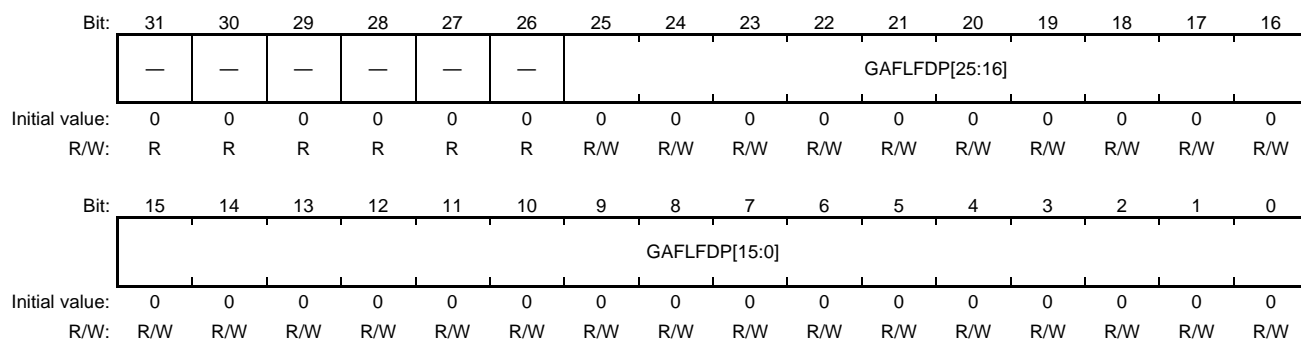
Access: RSCFDnGAFLP1_j register can be read/written in 32-bit units

RSCFDnGAFLP1_jL, RSCFDnGAFLP1_jH registers can be read/written in 16-bit units

RSCFDnGAFLP1_jLL, RSCFDnGAFLP1_jLH, RSCFDnGAFLP1_jHL, RSCFDnGAFLP1_jHH registers can be read/written in 8-bit units

Address: RSCFDnGAFLP1_j: <RSCFDn_base> + H'050C + (H'10 × j)
 RSCFDnGAFLP1_jL: <RSCFDn_base> + H'050C + (H'10 × j),
 RSCFDnGAFLP1_jH: <RSCFDn_base> + H'050E + (H'10 × j)
 RSCFDnGAFLP1_jLL: <RSCFDn_base> + H'050C + (H'10 × j),
 RSCFDnGAFLP1_jLH: <RSCFDn_base> + H'050D + (H'10 × j),
 RSCFDnGAFLP1_jHL: <RSCFDn_base> + H'050E + (H'10 × j),
 RSCFDnGAFLP1_jHH: <RSCFDn_base> + H'050F + (H'10 × j)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 26 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 25 to 8 | GAFLFDP [25:8] | H'0_0000 | R/W | Transmit/Receive FIFO Buffer k Select (Bit position –8 = target transmit/receive FIFO buffer number k) 0: Transmit/receive FIFO buffer is not selected. 1: Transmit/receive FIFO buffer is selected. |
| 7 to 0 | GAFLFDP [7:0] | H'00 | R/W | Receive FIFO Buffer x Select (Bit position = target receive FIFO buffer number x) 0: Receive FIFO buffer is not selected. 1: Receive FIFO buffer is selected. |

Modify the RSCFDnGAFLP1_j register when the AFLDAE bit in the RSCFDnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLFDP[25:0] Bits

These bits are used to specify FIFO buffers that store receive messages that have passed through the filter. Up to eight FIFO buffers are selectable. However, when the GAFLRMV bit in the RSCFDnGAFLP0_j register is set to 1 (a message is stored in the receive buffer), up to seven FIFO buffers can be selected. Only receive FIFO buffers and the transmit/receive FIFO buffer for which the CFM[1:0] bits in the RSCFDnCFCC register are set to B'00 (receive mode) or B'10 (gateway mode) are selectable.

48.7.5 Details of Receive Buffer Related Registers

48.7.5.1 RSCFDnRMNB - Receive Buffer Number Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnRMNB register can be read/written in 32-bit units

RSCFDnRMNBL, RSCFDnRMNBH registers can be read/written in 16-bit units

RSCFDnRMNBLL, RSCFDnRMNBLH, RSCFDnRMNBHL, RSCFDnRMNBHH registers can be read/written in 8-bit units

Address: RSCFDnRMNB: <RSCFDn_base> + H'00A4

RSCFDnRMNBL: <RSCFDn_base> + H'00A4, RSCFDnRMNBH: <RSCFDn_base> + H'00A6

RSCFDnRMNBLL: <RSCFDn_base> + H'00A4, RSCFDnRMNBLH: <RSCFDn_base> + H'00A5,

RSCFDnRMNBHL: <RSCFDn_base> + H'00A6, RSCFDnRMNBHH: <RSCFDn_base> + H'00A7

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | NRXMB[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 8 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 7 to 0 | NRXMB[7:0] | H'00 | R/W | Receive Buffer Number Configuration These bits are used to set the total number of receive buffers of the RS-CANFD module. The maximum value is 16 × (number of channels). Setting these bits all to 0 makes receive buffers unavailable. Set a value of 0 to 96. |

Modify the RSCFDnRMNB register only in global reset mode.

48.7.5.2 RSCFDnRMNDy — Receive Buffer New Data Register (y = 0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnRMNDy register can be read/written in 32-bit units

RSCFDnRMNDyL, RSCFDnRMNDyH registers can be read/written in 16-bit units

RSCFDnRMNDyLL, RSCFDnRMNDyLH, RSCFDnRMNDyHL, RSCFDnRMNDyHH registers can be read/written in 8-bit units

Address: RSCFDnRMNDy: <RSCFDn_base> + H'00A8 + (H'04 × y)

RSCFDnRMNDyL: <RSCFDn_base> + H'00A8 + (H'04 × y),

RSCFDnRMNDyH: <RSCFDn_base> + H'00AA + (H'04 × y)

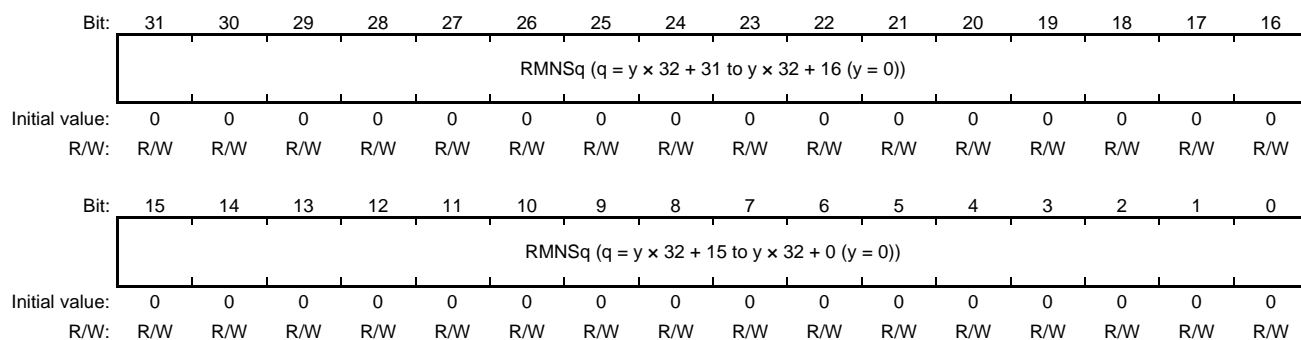
RSCFDnRMNDyLL: <RSCFDn_base> + H'00A8 + (H'04 × y),

RSCFDnRMNDyLH: <RSCFDn_base> + H'00A9 + (H'04 × y),

RSCFDnRMNDyHL: <RSCFDn_base> + H'00AA + (H'04 × y),

RSCFDnRMNDyHH: <RSCFDn_base> + H'00AB + (H'04 × y)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | RMNSq | H'0000_0000 | R/W | Receive Buffer Receive Complete Flag q (q = y × 32 + 31 to y × 32 + 0) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q. |

Write 0 to the RSCFDnRMNDy register in global operating mode or global test mode.

RMNSq Flags (q = 0 to 31)

Each RMNS flag is set to 1 when the processing for storing a message in the corresponding receive buffer starts.

To clear a flag to 0, the program must write 0 to the flag. Use a store instruction to write “0” to the flag and “1” to other flags. These bits cannot be set to 0 while a message is being stored. It takes ten clock cycles of pclk to store a message.

These flags are cleared to 0 in global reset mode.

48.7.5.3 RSCFDnRMIDq — Receive Buffer ID Register (q = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnRMIDq register can be read only in 32-bit units

RSCFDnRMIDqL, RSCFDnRMIDqH registers can be read only in 16-bit units

RSCFDnRMIDqLL, RSCFDnRMIDqLH, RSCFDnRMIDqHL, RSCFDnRMIDqHH registers can be read only in 8-bit units

Address: RSCFDnRMIDq: <RSCFDn_base> + H'0600 + (H'10 × q)

RSCFDnRMIDqL: <RSCFDn_base> + H'0600 + (H'10 × q),

RSCFDnRMIDqH: <RSCFDn_base> + H'0602 + (H'10 × q)

RSCFDnRMIDqLL: <RSCFDn_base> + H'0600 + (H'10 × q),

RSCFDnRMIDqLH: <RSCFDn_base> + H'0601 + (H'10 × q),

RSCFDnRMIDqHL: <RSCFDn_base> + H'0602 + (H'10 × q),

RSCFDnRMIDqHH: <RSCFDn_base> + H'0603 + (H'10 × q)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----------|----|-------------|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RMIDE | RMRT R | — | RMID[28:16] | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RMID[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 | RMIDE | B'0 | R | Receive Buffer IDE This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive buffer. 0: Standard ID 1: Extended ID |
| 30 | RMRT R | B'0 | R | Receive Buffer RTR This bit indicates the frame format (data frame or remote frame) of the message stored in the receive buffer. 0: Data frame 1: Remote frame |
| 29 | Reserved | B'0 | R | This bit is read as the value after reset. |
| 28 to 0 | RMID[28:0] | H'0000_0000 | R | Receive Buffer ID Data These bits contain the standard ID or extended ID of the received message. Read bits 10 to 0 for standard ID. Bits 28 to 11 are read as 0. |

48.7.5.4 RSCFDnRMPTRq — Receive Buffer Pointer Register (q = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnRMPTRq register can be read only in 32-bit units

RSCFDnRMPTRqL, RSCFDnRMPTRqH registers can be read only in 16-bit units

RSCFDnRMPTRqLL, RSCFDnRMPTRqLH, RSCFDnRMPTRqHL, RSCFDnRMPTRqHH registers can be read only in 8-bit units

Address: RSCFDnRMPTRq: <RSCFDn_base> + H'0604 + (H'10 × q)

RSCFDnRMPTRqL: <RSCFDn_base> + H'0604 + (H'10 × q),

RSCFDnRMPTRqH: <RSCFDn_base> + H'0606 + (H'10 × q)

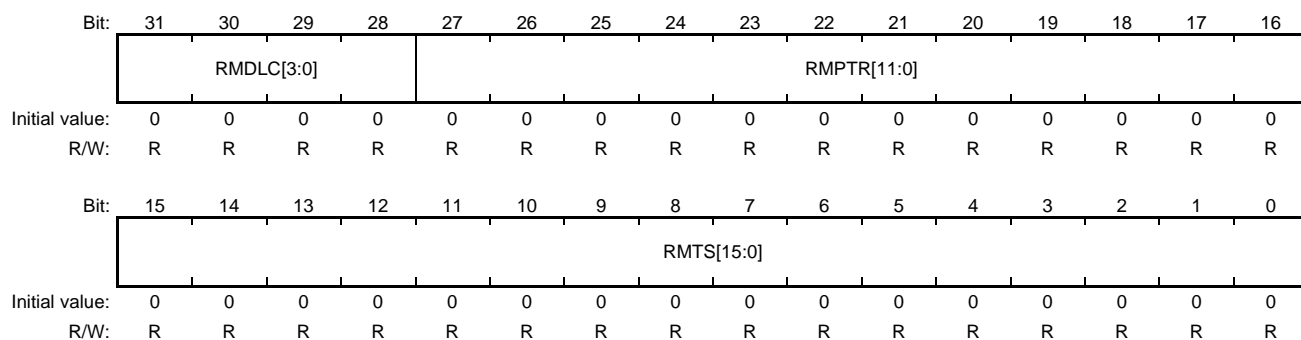
RSCFDnRMPTRqLL: <RSCFDn_base> + H'0604 + (H'10 × q),

RSCFDnRMPTRqLH: <RSCFDn_base> + H'0605 + (H'10 × q),

RSCFDnRMPTRqHL: <RSCFDn_base> + H'0606 + (H'10 × q),

RSCFDnRMPTRqHH: <RSCFDn_base> + H'0607 + (H'10 × q)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 28 | RMDLC[3:0] | B'0000 | R | Receive Buffer DLC Data These bits indicate the data length of the message stored in the receive buffer. B'0000: 0 data bytes B'0001: 1 data byte B'0010: 2 data bytes B'0011: 3 data bytes B'0100: 4 data bytes B'0101: 5 data bytes B'0110: 6 data bytes B'0111: 7 data bytes B'1XXX: 8 data bytes |
| 27 to 16 | RMPTR[11:0] | H'000 | R | Receive Buffer Label Data These bits indicate the label information of the message stored in the receive buffer. |
| 15 to 0 | RMTS[15:0] | H'0000 | R | Receive Buffer Timestamp Data These bits indicate the timestamp value of the message stored in the receive buffer. |

48.7.5.5 RSCFDnRMDF0_q — Receive Buffer Data Field 0 Register (q = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnRMDF0_q register can be read only in 32-bit units

RSCFDnRMDF0_qL, RSCFDnRMDF0_qH registers can be read only in 16-bit units

RSCFDnRMDF0_qLL, RSCFDnRMDF0_qLH, RSCFDnRMDF0_qHL, RSCFDnRMDF0_qHH registers can be read only in 8-bit units

Address: RSCFDnRMDF0_q: <RSCFDn_base> + H'0608 + (H'10 × q)

RSCFDnRMDF0_qL: <RSCFDn_base> + H'0608 + (H'10 × q),

RSCFDnRMDF0_qH: <RSCFDn_base> + H'060A + (H'10 × q)

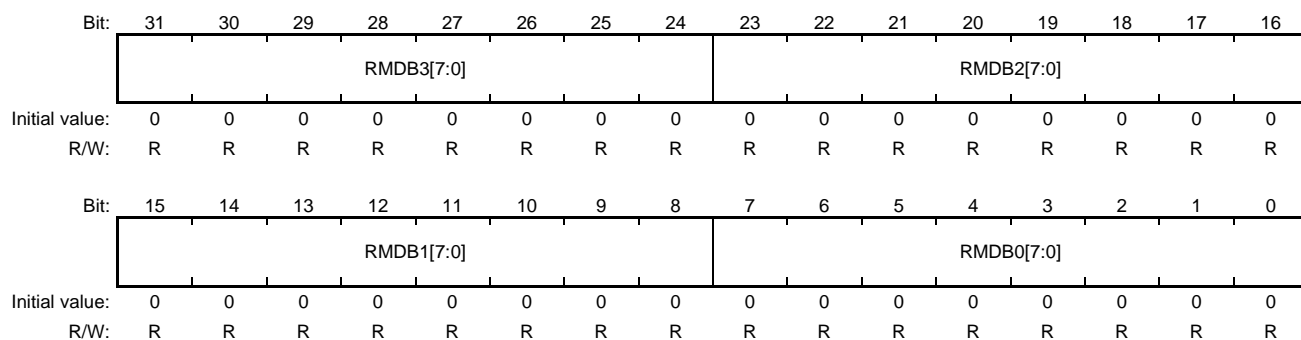
RSCFDnRMDF0_qLL: <RSCFDn_base> + H'0608 + (H'10 × q),

RSCFDnRMDF0_qLH: <RSCFDn_base> + H'0609 + (H'10 × q),

RSCFDnRMDF0_qHL: <RSCFDn_base> + H'060A + (H'10 × q),

RSCFDnRMDF0_qHH: <RSCFDn_base> + H'060B + (H'10 × q)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|----------------------------|
| 31 to 24 | RMDB3[7:0] | H'00 | R | Receive Buffer Data Byte 3 |
| 23 to 16 | RMDB2[7:0] | H'00 | R | Receive Buffer Data Byte 2 |
| 15 to 8 | RMDB1[7:0] | H'00 | R | Receive Buffer Data Byte 1 |
| 7 to 0 | RMDB0[7:0] | H'00 | R | Receive Buffer Data Byte 0 |

Data for a message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCFDnRMPTRq register is smaller than B'1000, data bytes for which no data is set are read as H'00.

48.7.5.6 RSCFDnRMDF1_q — Receive Buffer Data Field 1 Register (q = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnRMDF1_q register can be read only in 32-bit units

RSCFDnRMDF1_qL, RSCFDnRMDF1_qH register can be read only in 16-bit units

RSCFDnRMDF1_qLL, RSCFDnRMDF1_qLH, RSCFDnRMDF1_qHL, RSCFDnRMDF1_qHH registers can be read only in 8-bit units

Address: RSCFDnRMDF1_q: <RSCFDn_base> + H'060C + (H'10 × q)

RSCFDnRMDF1_qL: <RSCFDn_base> + H'060C + (H'10 × q),

RSCFDnRMDF1_qH: <RSCFDn_base> + H'060E + (H'10 × q)

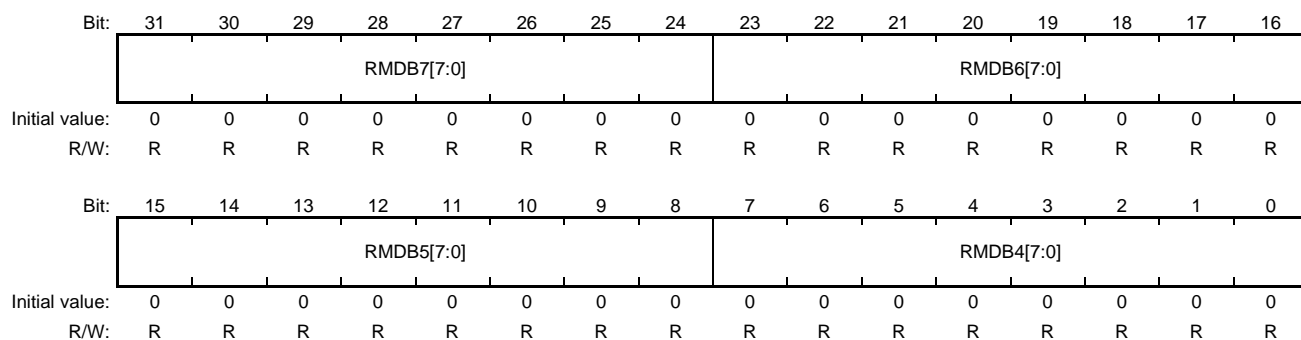
RSCFDnRMDF1_qLL: <RSCFDn_base> + H'060C + (H'10 × q),

RSCFDnRMDF1_qLH: <RSCFDn_base> + H'060D + (H'10 × q),

RSCFDnRMDF1_qHL: <RSCFDn_base> + H'060E + (H'10 × q),

RSCFDnRMDF1_qHH: <RSCFDn_base> + H'060F + (H'10 × q)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|----------------------------|
| 31 to 24 | RMDB7[7:0] | H'00 | R | Receive Buffer Data Byte 7 |
| 23 to 16 | RMDB6[7:0] | H'00 | R | Receive Buffer Data Byte 6 |
| 15 to 8 | RMDB5[7:0] | H'00 | R | Receive Buffer Data Byte 5 |
| 7 to 0 | RMDB4[7:0] | H'00 | R | Receive Buffer Data Byte 4 |

Data for a message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCFDnRMPTRq register is smaller than B'1000, data bytes for which no data is set are read as H'00.

48.7.6 Details of Receive FIFO Buffer Related Registers

48.7.6.1 RSCFDnRFCCx — Receive FIFO Buffer Configuration and Control Register (x = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnRFCCx register can be read/written in 32-bit units

RSCFDnRFCCxL, RSCFDnRFCCxH registers can be read/written in 16-bit units

RSCFDnRFCCxLL, RSCFDnRFCCxLH, RSCFDnRFCCxHL, RSCFDnRFCCxHH registers can be read/written in 8-bit units

Address: RSCFDnRFCCx: <RSCFDn_base> + H'00B8 + (H'04 × x)

RSCFDnRFCCxL: <RSCFDn_base> + H'00B8 + (H'04 × x),

RSCFDnRFCCxH: <RSCFDn_base> + H'00BA + (H'04 × x)

RSCFDnRFCCxLL: <RSCFDn_base> + H'00B8 + (H'04 × x),

RSCFDnRFCCxLH: <RSCFDn_base> + H'00B9 + (H'04 × x),

RSCFDnRFCCxHL: <RSCFDn_base> + H'00BA + (H'04 × x),

RSCFDnRFCCxHH: <RSCFDn_base> + H'00BB + (H'04 × x)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|-------------|-----|-----|------|----|-----------|-----|-----|----|----|----|----|----|----|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RFIGCV[2:0] | | | RFIM | — | RFDC[2:0] | | | — | — | — | — | — | — | RFIE | RFE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 16 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 15 to 13 | RFIGCV[2:0] | B'000 | R/W | Receive FIFO Interrupt Request Timing Select These bits are used to specify the number of received messages for generating a receive FIFO interrupt request when the RFIM bit is set to 0 with a fraction for the total number of buffers (the setting of RFDC[2:0]). When the RFDC[2:0] bits are set to B'001 (4 messages), set the RFIGCV[2:0] bits to B'001, B'011, B'101, or B'111. Modify these bits only in global reset mode. B'000: When FIFO is 1/8 full. B'001: When FIFO is 2/8 full. B'010: When FIFO is 3/8 full. B'011: When FIFO is 4/8 full. B'100: When FIFO is 5/8 full. B'101: When FIFO is 6/8 full. B'110: When FIFO is 7/8 full. B'111: When FIFO is full. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 12 | RFIM | B'0 | R/W | <p>Receive FIFO Interrupt Source Select</p> <p>This bit is used to select a FIFO interrupt source. Modify this bit only in global reset mode.</p> <p>0: An interrupt occurs when the condition set by the RFIGCV[2:0] bits is met.</p> <p>1: An interrupt occurs each time a message has been received.</p> |
| 11 | Reserved | B'0 | R | <p>This bit is read as the value after reset.</p> <p>The write value should be the value after reset.</p> |
| 10 to 8 | RFDC[2:0] | B'000 | R/W | <p>Receive FIFO Buffer Depth Configuration</p> <p>These bits are used to select the number of messages that can be stored in a single receive FIFO buffer.</p> <p>When these bits are set to B'000, no receive FIFO buffer should be used. Modify these bits only in global reset mode.</p> <p>B'000: 0 messages B'001: 4 messages B'010: 8 messages B'011: 16 messages B'100: 32 messages B'101: 48 messages B'110: 64 messages B'111: 128 messages</p> |
| 7 to 2 | Reserved | All 0 | R | <p>These bits are read as the value after reset.</p> <p>The write value should be the value after reset.</p> |
| 1 | RFIE | B'0 | R/W | <p>Receive FIFO Interrupt Enable</p> <p>Setting the RFIE bit to 1 enables receive FIFO interrupts. Modify this bit when the RFE bit set to 0 (no receive FIFO buffer is used).</p> <p>0: Receive FIFO interrupt is disabled. 1: Receive FIFO interrupt is enabled.</p> |
| 0 | RFE | B'0 | R/W | <p>Receive FIFO Buffer Enable</p> <p>Setting the RFE bit to 1 makes receive FIFO buffers available. Clearing this bit to 0 sets the RFEMP flag in the RSCFDnRFSTx register to 1 (buffer empty). Modify this bit in global operating mode or global test mode.</p> <p>0: No receive FIFO buffer is used. 1: Receive FIFO buffers are used.</p> |

48.7.6.2 RSCFDnRFSTsX — Receive FIFO Buffer Status Register (x = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnRFSTsX register can be read/written in 32-bit units

RSCFDnRFSTsXL, RSCFDnRFSTsXH registers can be read/written in 16-bit units

RSCFDnRFSTsxLL, RSCFDnRFSTsxLH, RSCFDnRFSTsxHL, RSCFDnRFSTsxHH registers can be read/written in 8-bit units

Address: RSCFDnRFSTsX: <RSCFDn_base> + H'00D8 + (H'04 × x)

RSCFDnRFSTsXL: <RSCFDn_base> + H'00D8 + (H'04 × x),

RSCFDnRFSTsXH: <RSCFDn_base> + H'00DA + (H'04 × x)

RSCFDnRFSTsxLL: <RSCFDn_base> + H'00D8 + (H'04 × x),

RSCFDnRFSTsxLH: <RSCFDn_base> + H'00D9 + (H'04 × x),

RSCFDnRFSTsxHL: <RSCFDn_base> + H'00DA + (H'04 × x),

RSCFDnRFSTsxHH: <RSCFDn_base> + H'00DB + (H'04 × x)

Value after reset: H'0000_0001

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----|----|----|----|----|----|----|----|------|-------|------|-------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RFMC[7:0] | | | | | | | — | — | — | — | RFIF | RFMLT | RFLL | RFEMP | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W* | R/W* | R | R |

Note: * The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|------|---|
| 31 to 16 | Reserved | All 0 | R | These bits are read as the value after reset. When writing to these bits, write the value after reset. |
| 15 to 8 | RFMC[7:0] | H'00 | R | Receive FIFO Unread Message Counter This flag indicates the number of unread messages in the receive FIFO buffer. This flag becomes H'00 when the RFE bit in the RSCFDnRFCCx register is set to 0. |
| 7 to 4 | Reserved | All 0 | R | These bits are read as the value after reset. When writing to these bits, write the value after reset. |
| 3 | RFIF | B'0 | R/W* | Receive FIFO Interrupt Request Flag This flag is set to 1 when the receive FIFO interrupt request generation conditions set by the RFIGCV[2:0] bits and the RFIM bit in the RSCFDnRFCCx register are met. This flag is cleared to 0 in global reset mode or by writing 0 to this flag. Modify this bit in global operating mode or global test mode. To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1". 0: No receive FIFO interrupt request is present. 1: A receive FIFO interrupt request is present. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 2 | RFMLT | B'0 | R/W* | <p>Receive FIFO Message Lost Flag</p> <p>This flag is set to 1 when an attempt is made to store a new message while the receive FIFO buffer is full. In this case, the new message is discarded.</p> <p>This flag is cleared to 0 in global reset mode or by writing 0 to this flag.</p> <p>Modify this bit in global operating mode or global test mode.</p> <p>To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.</p> <p>When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1".</p> <p>0: No receive FIFO message is lost. 1: A receive FIFO message is lost.</p> |
| 1 | RFFLL | B'0 | R | <p>Receive FIFO Buffer Full Status Flag</p> <p>This flag is set to 1 when the number of messages stored in the receive FIFO buffer matches the FIFO buffer depth set by the RFDC[2:0] bits in the RSCFDnRFCCx register.</p> <p>If the number of messages stored in the receive FIFO buffer becomes smaller than the FIFO buffer depth set by the RFDC[2:0] bits, this flag is cleared to 0. This flag is also cleared to 0 when the RFE bit in the RSCFDnRFCCx register is set to 0 (no receive FIFO buffer is used) or in global reset mode.</p> <p>0: The receive FIFO buffer is not full. 1: The receive FIFO buffer is full.</p> |
| 0 | RFEMP | B'1 | R | <p>Receive FIFO Buffer Empty Status Flag</p> <p>This flag is set to 1 when all messages in the receive FIFO buffer have been read. This flag is also set to 1 when the RFE bit in the RSCFDnRFCCx register is 0 or in global reset mode.</p> <p>This flag is cleared to 0 when even a single received message has been stored in the receive FIFO buffer.</p> <p>0: The receive FIFO buffer contains unread message. 1: The receive FIFO buffer contains no unread message (buffer empty).</p> |

Note: To clear the RFMLT or RFIF flag to 0, use a store instruction to write "0" to the given flag and "1" to the other flags.

48.7.6.3 RSCFDnRFPCTR_x — Receive FIFO Buffer Pointer Control Register (x = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnRFPCTR_x register can only be written in 32-bit units

RSCFDnRFPCTR_{xL}, RSCFDnRFPCTR_{xH} registers can only be written in 16-bit units

RSCFDnRFPCTR_{xLL}, RSCFDnRFPCTR_{xLH}, RSCFDnRFPCTR_{xHL}, RSCFDnRFPCTR_{xHH} registers can only be written in 8-bit units

Address: RSCFDnRFPCTR_x: <RSCFDn_base> + H'00F8 + (H'04 × x)

RSCFDnRFPCTR_{xL}: <RSCFDn_base> + H'00F8 + (H'04 × x),

RSCFDnRFPCTR_{xH}: <RSCFDn_base> + H'00FA + (H'04 × x)

RSCFDnRFPCTR_{xLL}: <RSCFDn_base> + H'00F8 + (H'04 × x),

RSCFDnRFPCTR_{xLH}: <RSCFDn_base> + H'00F9 + (H'04 × x),

RSCFDnRFPCTR_{xHL}: <RSCFDn_base> + H'00FA + (H'04 × x),

RSCFDnRFPCTR_{xHH}: <RSCFDn_base> + H'00FB + (H'04 × x)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-----------|----|----|----|----|----|----|----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | RFPC[7:0] | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | W | W | W | W | W | W | W | W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 8 | Reserved | All 0 | R | The write value should be the value after reset. |
| 7 to 0 | RFPC[7:0] | H'00 | W | <p>Receive FIFO Pointer Control</p> <p>When the RFPC[7:0] bits are set to H'FF, the read pointer moves to the next unread message in the receive FIFO buffer. At this time, the RFMC[7:0] (receive FIFO unread message counter) value in the RSCFDnRFST_{Sx} register is decremented. Read the RSCFDnRFID_x, RSCFDnRFPTR_x, RSCFDnRFDF0_x, and RSCFDnRFDF1_x registers to read messages in the receive FIFO buffer, and then write H'FF to the RFPC[7:0] bits.</p> <p>When writing H'FF to these bits, make sure that the RFE bit in the RSCFDnRFCC_x register is set to 1 (receive FIFO buffers are used) and the RFEMP flag in the RSCFDnRFST_{Sx} register is 0 (the receive FIFO buffer contains unread messages).</p> |

48.7.6.4 RSCFDnRFIDx — Receive FIFO Buffer Access ID Register (x = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnRFIDx register can be read only in 32-bit units

RSCFDnRFIDxL, RSCFDnRFIDxH registers can be read only in 16-bit units

RSCFDnRFIDxLL, RSCFDnRFIDxLH, RSCFDnRFIDxHL, RSCFDnRFIDxHH registers can be read only in 8-bit units

Address: RSCFDnRFIDx: <RSCFDn_base> + H'0E00 + (H'10 × x)

RSCFDnRFIDxL: <RSCFDn_base> + H'0E00 + (H'10 × x),

RSCFDnRFIDxH: <RSCFDn_base> + H'0E02 + (H'10 × x)

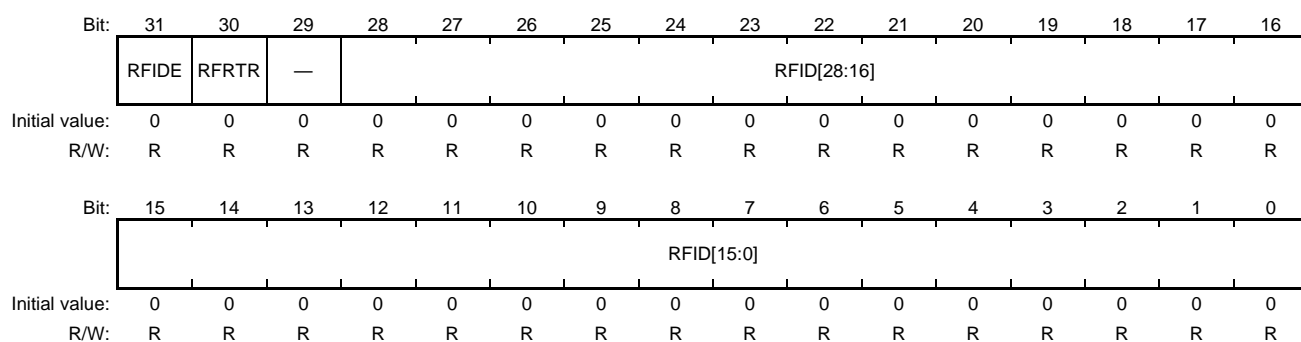
RSCFDnRFIDxLL: <RSCFDn_base> + H'0E00 + (H'10 × x),

RSCFDnRFIDxLH: <RSCFDn_base> + H'0E01 + (H'10 × x),

RSCFDnRFIDxHL: <RSCFDn_base> + H'0E02 + (H'10 × x),

RSCFDnRFIDxHH: <RSCFDn_base> + H'0E03 + (H'10 × x)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 | RFIDE | B'0 | R | Receive FIFO Buffer IDE This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive FIFO buffer. 0: Standard ID 1: Extended ID |
| 30 | RFRTR | B'0 | R | Receive FIFO Buffer RTR This bit indicates the frame format (data frame or remote frame) of the message stored in the receive FIFO buffer. 0: Data frame 1: Remote frame |
| 29 | Reserved | B'0 | R | This bit is read as the value after reset. |
| 28 to 0 | RFID[28:0] | H'0000_0000 | R | Receive FIFO Buffer ID Data These bits indicate the ID of the message stored in the receive FIFO buffer. The standard ID or extended ID of received message can be read. Read bits 10 to 0 for standard ID. Bits 28 to 11 are read as 0. |

48.7.6.5 RSCFDnRFPTRx — Receive FIFO Buffer Access Pointer Register (x = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnRFPTRx register can be read only in 32-bit units

RSCFDnRFPTRxL, RSCFDnRFPTRxH registers can be read only in 16-bit units

RSCFDnRFPTRxLL, RSCFDnRFPTRxLH, RSCFDnRFPTRxHL, RSCFDnRFPTRxHH registers can be read only in 8-bit units

Address: RSCFDnRFPTRx: <RSCFDn_base> + H'0E04 + (H'10 × x)

RSCFDnRFPTRxL: <RSCFDn_base> + H'0E04 + (H'10 × x),

RSCFDnRFPTRxH: <RSCFDn_base> + H'0E06 + (H'10 × x)

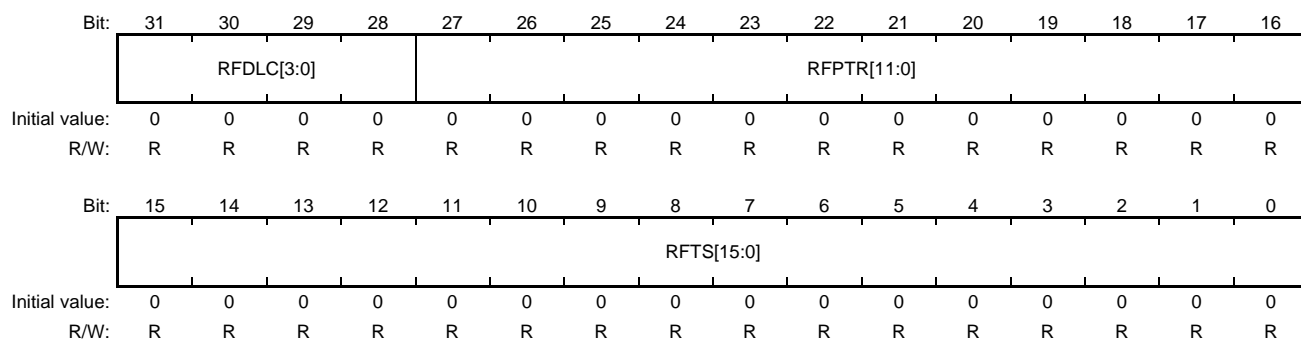
RSCFDnRFPTRxLL: <RSCFDn_base> + H'0E04 + (H'10 × x),

RSCFDnRFPTRxLH: <RSCFDn_base> + H'0E05 + (H'10 × x),

RSCFDnRFPTRxHL: <RSCFDn_base> + H'0E06 + (H'10 × x),

RSCFDnRFPTRxHH: <RSCFDn_base> + H'0E07 + (H'10 × x)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 28 | RFDLC[3:0] | B'0000 | R | Receive FIFO Buffer DLC Data These bits contain the data length of the message stored in the receive FIFO buffer. B'0000: 0 data bytes B'0001: 1 data byte B'0010: 2 data bytes B'0011: 3 data bytes B'0100: 4 data bytes B'0101: 5 data bytes B'0110: 6 data bytes B'0111: 7 data bytes B'1XXX: 8 data bytes |
| 27 to 16 | RFPTR[11:0] | H'000 | R | Receive FIFO Buffer Label Data These bits contain the label information of the message stored in the receive FIFO buffer. |
| 15 to 0 | RFTS[15:0] | H'0000 | R | Receive FIFO Buffer Timestamp Data These bits contain the timestamp value of the message stored in the receive FIFO buffer. |

48.7.6.6 RSCFDnRFDF0_x — Receive FIFO Buffer Access Data Field 0 Register (x = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnRFDF0_x register can be read only in 32-bit units

RSCFDnRFDF0_xL, RSCFDnRFDF0_xH registers can be read only in 16-bit units

RSCFDnRFDF0_xLL, RSCFDnRFDF0_xLH, RSCFDnRFDF0_xHL, RSCFDnRFDF0_xHH registers can be read only in 8-bit units

Address: RSCFDnRFDF0_x: <RSCFDn_base> + H'0E08 + (H'10 × x)

RSCFDnRFDF0_xL: <RSCFDn_base> + H'0E08 + (H'10 × x),

RSCFDnRFDF0_xH: <RSCFDn_base> + H'0E0A + (H'10 × x)

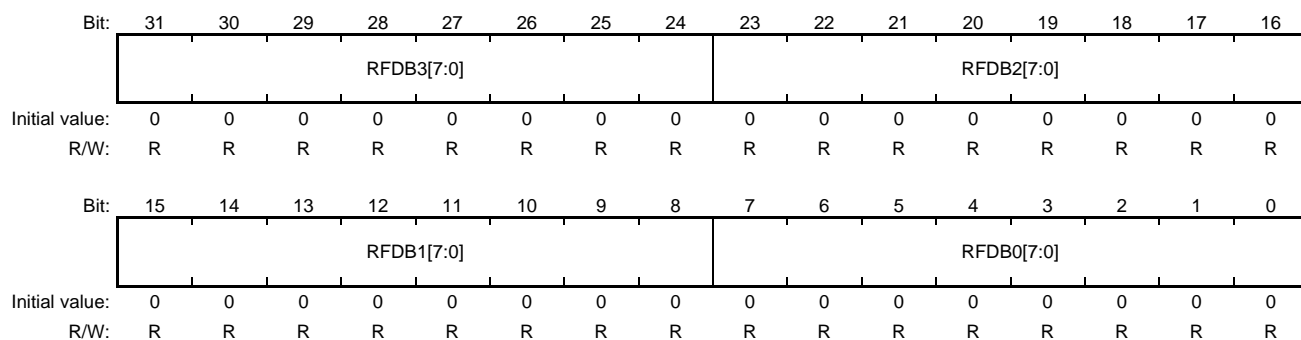
RSCFDnRFDF0_xLL: <RSCFDn_base> + H'0E08 + (H'10 × x),

RSCFDnRFDF0_xLH: <RSCFDn_base> + H'0E09 + (H'10 × x),

RSCFDnRFDF0_xHL: <RSCFDn_base> + H'0E0A + (H'10 × x),

RSCFDnRFDF0_xHH: <RSCFDn_base> + H'0E0B + (H'10 × x)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---------------------------------|
| 31 to 24 | RFDB3[7:0] | H'00 | R | Receive FIFO Buffer Data Byte 3 |
| 23 to 16 | RFDB2[7:0] | H'00 | R | Receive FIFO Buffer Data Byte 2 |
| 15 to 8 | RFDB1[7:0] | H'00 | R | Receive FIFO Buffer Data Byte 1 |
| 7 to 0 | RFDB0[7:0] | H'00 | R | Receive FIFO Buffer Data Byte 0 |

Data for a message stored in the receive FIFO buffer can be read.

When the RFDLC[3:0] value in the RSCFDnRFPTRx register is smaller than B'1000, data bytes for which no data is set are read as H'00.

48.7.6.7 RSCFDnRFDF1_x — Receive FIFO Buffer Access Data Field 1 Register (x = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnRFDF1_x register can be read only in 32-bit units

RSCFDnRFDF1_xL, RSCFDnRFDF1_xH registers can be read only in 16-bit units

RSCFDnRFDF1_xLL, RSCFDnRFDF1_xLH, RSCFDnRFDF1_xHL, RSCFDnRFDF1_xHH registers can be read only in 8-bit units

Address: RSCFDnRFDF1_x: <RSCFDn_base> + H'0E0C + (H'10 × x)

RSCFDnRFDF1_xL: <RSCFDn_base> + H'0E0C + (H'10 × x),

RSCFDnRFDF1_xH: <RSCFDn_base> + H'0E0E + (H'10 × x)

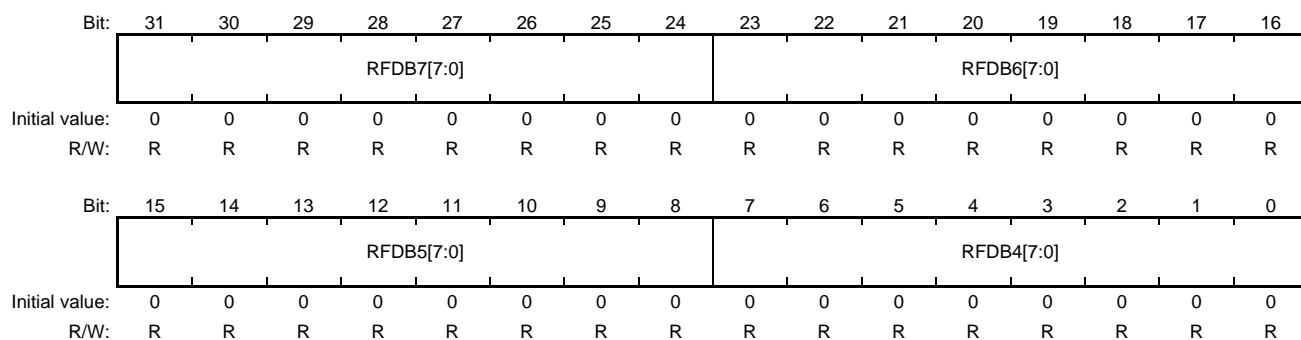
RSCFDnRFDF1_xLL: <RSCFDn_base> + H'0E0C + (H'10 × x),

RSCFDnRFDF1_xLH: <RSCFDn_base> + H'0E0D + (H'10 × x),

RSCFDnRFDF1_xHL: <RSCFDn_base> + H'0E0E + (H'10 × x),

RSCFDnRFDF1_xHH: <RSCFDn_base> + H'0E0F + (H'10 × x)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---------------------------------|
| 31 to 24 | RFDB7[7:0] | H'00 | R | Receive FIFO Buffer Data Byte 7 |
| 23 to 16 | RFDB6[7:0] | H'00 | R | Receive FIFO Buffer Data Byte 6 |
| 15 to 8 | RFDB5[7:0] | H'00 | R | Receive FIFO Buffer Data Byte 5 |
| 7 to 0 | RFDB4[7:0] | H'00 | R | Receive FIFO Buffer Data Byte 4 |

Data for a message stored in the receive FIFO buffer can be read.

When the RFDLC[3:0] value in the RSCFDnRFPTRx register is smaller than B'1000, data bytes for which no data is set are read as H'00.

48.7.7 Details of Transmit/Receive FIFO Buffer Related Registers

48.7.7.1 RSCFDnCFCCk — Transmit/receive FIFO Buffer Configuration and Control Register k (k = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFCCk register can be read/written in 32-bit units

RSCFDnCFCCkL, RSCFDnCFCCkH registers can be read/written in 16-bit units

RSCFDnCFCCkLL, RSCFDnCFCCkLH, RSCFDnCFCCkHL, RSCFDnCFCCkHH registers can be read/written in 8-bit units

Address: RSCFDnCFCCk: <RSCFDn_base> + H'0118 + (H'04 × k)

RSCFDnCFCCkL: <RSCFDn_base> + H'0118 + (H'04 × k),

RSCFDnCFCCkH: <RSCFDn_base> + H'011A + (H'04 × k)

RSCFDnCFCCkLL: <RSCFDn_base> + H'0118 + (H'04 × k),

RSCFDnCFCCkLH: <RSCFDn_base> + H'0119 + (H'04 × k),

RSCFDnCFCCkHL: <RSCFDn_base> + H'011A + (H'04 × k),

RSCFDnCFCCkHH: <RSCFDn_base> + H'011B + (H'04 × k)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|-------------|-----|-----|------|-----|-----------|-----|------------|-----|-----|-----|-------|--------|----------|------------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CFITT[7:0] | | | | | | | CFTML[3:0] | | | | CFITR | CFITSS | CFM[1:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CFGICV[2:0] | | | CFIM | — | CFDC[2:0] | | | — | — | — | — | — | CFIXIE | CFRXI E | CFE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 24 | CFITT[7:0] | H'00 | R/W | <p>Set a message transmission interval.</p> <p>These bits are used to set a message transmission interval when transmitting messages continuously from a transmit/receive FIFO buffer whose CFM[1:0] bits are set to B'01 (transmit mode) or B'10 (gateway mode).</p> <p>Clear the CFE bit to 0 (no transmit/receive FIFO buffer is used) before modifying the CFITT[7:0] bits.</p> <p>Set Value: H'00 to H'FF</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 23 to 20 | CFTML[3:0] | H'0 | R/W | <p>Transmit Buffer Link Configuration</p> <p>These bits are used to set the number of transmit buffer on the channel which will be linked to transmit/receive FIFO buffer k when the CFM[1:0] bits are set to B'01 (transmit mode) or B'10 (gateway mode).</p> <p>There are three transmit/receive FIFO buffers per channel, so channel number m of FIFO buffer k is calculated as $m = k/3$ (integer division). The actual assigned transmit buffer number p linked to FIFO buffer k will be $((16 \times m) + CFTML[3:0])$.</p> <p>See Table 48.7 and Table 48.8, as for the relationship between transmit/receive FIFO buffer k and transmit buffer p.</p> <p>Setting the CFDC[2:0] bits to B'001 or more enables the setting of the CFTML[3:0] bits.</p> <p>Do not link to any transmit buffer which is already allocated to a transmit queue on the identical channel or to another transmit/receive FIFO buffer. Modify these bits only in global reset mode.</p> |
| 19 | CFITR | B'0 | R/W | <p>Transmit/Receive FIFO Interval Timer Resolution</p> <p>This bit is enabled when the CFITSS bit is 0.</p> <p>When this bit is 0, the interval timer clock source is the pclk/2 clock divided by the value of the ITRCP[15:0] bits in the RSCFDnGCFG register.</p> <p>When this bit is 1, the interval timer clock source is the pclk/2 clock divided by (the value of the ITRCP[15:0] bits in the RSCFDnGCFG register $\times 10$).</p> <p>Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).</p> <p>0: Clock dividing pclk/2 by (ITRCP[15:0] bits) 1: Clock dividing pclk/2 by (ITRCP[15:0] bits $\times 10$)</p> |
| 18 | CFITSS | B'0 | R/W | <p>Transmit/Receive FIFO Interval Timer Clock Source Select</p> <p>When this bit is 0, the clock selected by the CFITR bit is the count source of the interval timer.</p> <p>When this bit is 1, the bit time clock of the channel to which the FIFO is linked is the count source of the interval timer.</p> <p>Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).</p> <p>0: Interval timer clock source selected by the CFITR bit 1: Interval timer clock source is the bit time clock for the channel to which the FIFO is linked.</p> |
| 17, 16 | CFM[1:0] | B'00 | R/W | <p>Transmit/Receive FIFO Mode Select</p> <p>These bits are used to select transmit/receive FIFO mode. Modify these bits only in global reset mode.</p> <p>B'00: Receive mode B'01: Transmit mode B'10: Gateway mode B'11: Setting prohibited</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 15 to 13 | CFIGCV[2:0] | B'000 | R/W | <p>Transmit/Receive FIFO Receive Interrupt Request Timing Select</p> <p>These bits are used to specify the number of received messages for generating a transmit/receive FIFO receive interrupt request when the CFM[1:0] bits are set to B'00 (receive mode) or B'10 (gateway mode) and the CFIM bit is set to 0 with a fraction for the total number of buffers (the setting of CFDC[2:0]).</p> <p>When the CFDC[2:0] bits are set to B'001 (4 messages), set the CFIGCV[2:0] bits to B'001, B'011, B'101, or B'111.</p> <p>Modify these bits only in global reset mode.</p> <p>B'000: When FIFO is 1/8 full. B'001: When FIFO is 2/8 full. B'010: When FIFO is 3/8 full. B'011: When FIFO is 4/8 full. B'100: When FIFO is 5/8 full. B'101: When FIFO is 6/8 full. B'110: When FIFO is 7/8 full. B'111: When FIFO is full.</p> |
| 12 | CFIM | B'0 | R/W | <p>Transmit/Receive FIFO Interrupt Source Select</p> <p>This bit is used to select a transmit/receive FIFO interrupt source. Modify this bit only in global reset mode.</p> <p>0:</p> <ul style="list-style-type: none"> Receive mode/gateway mode When the number of received messages has met the condition set by the CFIGCV[2:0] bits, a FIFO receive interrupt request is generated. Transmit mode/gateway mode When the buffer becomes empty upon completion of message transmission, a FIFO transmit interrupt request is generated. <p>1:</p> <ul style="list-style-type: none"> Receive mode/gateway mode A FIFO receive interrupt request is generated each time a message has been received. Transmit mode/gateway mode A FIFO transmit interrupt request is generated each time a message has been transmitted. |
| 11 | Reserved | B'0 | R | <p>This bit is read as the value after reset.</p> <p>The write value should be the value after reset.</p> |
| 10 to 8 | CFDC[2:0] | B'000 | R/W | <p>Transmit/Receive FIFO Buffer Depth Configuration</p> <p>These bits are used to set the number of messages that can be stored in a single transmit/receive FIFO buffer. When these bits are set to B'000, do not use a transmit/receive FIFO buffer. Modify these bits only in global reset mode.</p> <p>B'000: 0 messages B'001: 4 messages B'010: 8 messages B'011: 16 messages B'100: 32 messages B'101: 48 messages B'110: 64 messages B'111: 128 messages</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 to 3 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 2 | CCTXIE | B'0 | R/W | <p>Transmit/Receive FIFO Transmit Interrupt Enable</p> <p>When this bit is set to 1 and the CCTXIF flag in the RSCFDnCFSTSk register is set to 1, a transmit/receive FIFO transmit interrupt request is generated.</p> <p>Modify this bit with the CFE bit set to 0 (no transmit/receive FIFO buffer is used).</p> <p>0: Transmit/receive FIFO transmit interrupt is disabled. 1: Transmit/receive FIFO transmit interrupt is enabled.</p> |
| 1 | CFRXIE | B'0 | R/W | <p>Transmit/Receive FIFO Receive Interrupt Enable</p> <p>When this bit is set to 1 and the CFRXIF flag in the RSCFDnCFSTSk register is set to 1, a transmit/receive FIFO receive interrupt request is generated.</p> <p>Modify this bit with the CFE bit set to 0.</p> <p>0: Transmit/receive FIFO receive interrupt is disabled. 1: Transmit/receive FIFO receive interrupt is enabled.</p> |
| 0 | CFE | B'0 | R/W | <p>Transmit/Receive FIFO Buffer Enable</p> <p>Setting this bit to 1 makes transmit/receive FIFO buffers available.</p> <p>When this bit is set to 0 in transmit mode or gateway mode, if a message in the transmit/receive FIFO buffer is being transmitted or will be transmitted next, the transmit/receive FIFO buffer becomes empty after completion of transmission of that message, or upon detection of a CAN bus error, or arbitration-lost. In other cases or in receive mode, the transmit/receive FIFO buffer becomes empty immediately.</p> <p>This bit is cleared to 0 when the following conditions are met.</p> <ul style="list-style-type: none"> • Receive mode: Global reset mode • Transmit mode or gateway mode: Channel reset mode <p>Modify this bit in the following mode.</p> <ul style="list-style-type: none"> • Receive mode: Global operating mode or global test mode • Transmit mode or gateway mode: Channel communication mode or channel halt mode <p>First set all the other bits in the RSCFDnCFCCk register and then set this bit to 1 using another instruction.</p> <p>0: No transmit/receive FIFO buffer is used. 1: Transmit/receive FIFO buffers are used.</p> |

48.7.7.2 RSCFDnCFSTSk — Transmit/receive FIFO Buffer Status Register (k = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFSTSk register can be read/written in 32-bit units

RSCFDnCFSTSkL, RSCFDnCFSTSkH registers can be read/written in 16-bit units

RSCFDnCFSTSkLL, RSCFDnCFSTSkLH, RSCFDnCFSTSkHL, RSCFDnCFSTSkHH registers can be read/written in 8-bit units

Address: RSCFDnCFSTSk: <RSCFDn_base> + H'0178 + (H'04 × k)

RSCFDnCFSTSkL: <RSCFDn_base> + H'0178 + (H'04 × k),

RSCFDnCFSTSkH: <RSCFDn_base> + H'017A + (H'04 × k)

RSCFDnCFSTSkLL: <RSCFDn_base> + H'0178 + (H'04 × k),

RSCFDnCFSTSkLH: <RSCFDn_base> + H'0179 + (H'04 × k),

RSCFDnCFSTSkHL: <RSCFDn_base> + H'017A + (H'04 × k),

RSCFDnCFSTSkHH: <RSCFDn_base> + H'017B + (H'04 × k)

Value after reset: H'0000_0001

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----|----|----|----|----|----|----|--------|--------|-------|------|-------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CFMC[7:0] | | | | | | | — | — | — | CFTXIF | CFRXIF | CFMLT | CFLL | CFEMP | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W* | R/W* | R/W* | R | R |

Note: * The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 16 | Reserved | All 0 | R | These bits are read as the value after reset. When writing to these bits, write the value after reset. |
| 15 to 8 | CFMC[7:0] | H'00 | R | Transmit/Receive FIFO Message Counter The number of messages stored in the transmit/receive FIFO buffer. The CFMC[7:0] bits indicate the following values that depend on the setting of the CFM[1:0] bits in the RSCFDnCFCCk register. <ul style="list-style-type: none"> When CFM[1:0] value is B'01 (transmit mode): Number of untransmitted messages in the buffer When CFM[1:0] value is B'00 (receive mode): Number of unread received messages in the buffer When CFM[1:0] value is B'10 (gateway mode): Number of untransmitted received messages in the buffer These bits are cleared to 0 when any of the following conditions is met. <ul style="list-style-type: none"> When CFM[1:0] value is B'00: In global reset mode When CFM[1:0] value is B'01 or B'10: In channel reset mode |
| 7 to 5 | Reserved | All 0 | R | These bits are read as the value after reset. When writing to these bits, write the value after reset. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 4 | CFTXIF | B'0 | R/W* | <p>Transmit/Receive FIFO Transmit Interrupt Request Flag</p> <p>The CFTXIF flag is set to 1 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When the CFM[1:0] bits are set to B'01 or B'10, and the factor selected by the CFIM bit in the RSCFDnCFCCk register occurs <p>The CFTXIF flag is cleared to 0 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When 0 is written to the CFTXIF flag When the CFM[1:0] bits are set to B'00: In global reset mode When the CFM[1:0] bits are set to B'01 or B'10: In channel reset mode <p>Write 0 to this flag in global operating mode or global test mode.</p> <p>To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.</p> <p>When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1".</p> <p>0: No transmit/receive FIFO transmit interrupt request is present. 1: A transmit/receive FIFO transmit interrupt request is present.</p> |
| 3 | CFRXIF | B'0 | R/W* | <p>Transmit/Receive FIFO Receive Interrupt Request Flag</p> <p>The CFRXIF flag is set to 1 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When the CFM[1:0] bits are set to B'00 or B'10, and the factor selected by the CFIM bit in the RSCFDnCFCCk register occurs <p>The CFRXIF flag is cleared to 0 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When 0 is written to the CFRXIF flag When the CFM[1:0] bits are set to B'00: In global reset mode When the CFM[1:0] bits are set to B'01 or B'10: In channel reset mode <p>Write 0 to this flag in global operating mode or global test mode.</p> <p>To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.</p> <p>When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1".</p> <p>0: No transmit/receive FIFO receive interrupt request is present. 1: A transmit/receive FIFO receive interrupt request is present.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 2 | CFMLT | B'0 | R/W* | <p>Transmit/Receive FIFO Message Lost Flag</p> <p>The CFMLT flag is set to 1 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When an attempt is made to store a new message while the transmit/receive FIFO buffer is full. In this case, the new message is discarded. <p>The CFMLT flag is cleared to 0 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When 0 is written to the CFMLT flag When the CFM[1:0] bits are set to B'00: In global reset mode When the CFM[1:0] bits are set to B'01 or B'10: In channel reset mode <p>Write 0 to this flag in global operating mode or global test mode</p> <p>To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.</p> <p>When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1".</p> <p>0: No transmit/receive FIFO message is lost. 1: A transmit/receive FIFO message is lost.</p> |
| 1 | CFLL | B'0 | R | <p>Transmit/Receive FIFO Buffer Full Status Flag</p> <p>The CFLL flag is set to 1 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When the number of messages stored in the transmit/receive FIFO buffer matches the FIFO buffer depth set by the CFDC[2:0] bits in the RSCFDnCFCCk register. <p>The CFMLT flag is cleared to 0 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When the number of messages stored in the transmit/receive FIFO buffer becomes smaller than the FIFO buffer depth set by the CFDC[2:0] bits. When the CFE bit in the RSCFDnCFCCk register is 0 (no transmit/receive FIFO buffer is used): When not in the transmit abort When the CFM[1:0] bits are set to B'00: In global reset mode When the CFM[1:0] bits are set to B'01 or B'10: In channel reset mode <p>0: The transmit/receive FIFO buffer is not full. 1: The transmit/receive FIFO buffer is full.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | CFEMP | B'1 | R | <p>Transmit/Receive FIFO Buffer Empty Status Flag</p> <p>The CFEMP flag is set to 1 when any of the following conditions is met.</p> <ul style="list-style-type: none"> • When the CFM[1:0] bits are set to B'00: All messages have been read, or in global reset mode • When the CFM[1:0] bits are set to B'01 or B'10: All messages have been transmitted, or in channel reset mode • When the CFE bit is 0 (no transmit/receive FIFO buffer is used): Not in the transmit abort <p>The CFEMP flag is cleared to 0 when any of the following conditions is met.</p> <ul style="list-style-type: none"> • When the CFM[1:0] bits are set to B'00 or B'10: At least one received message has been stored in the transmit/receive FIFO buffer. • When the CFM[1:0] bits are set to B'01: A value of H'FF has been written to the RSCFDnCFPCTRk register after data was written to the RSCFDnCFIDk, RSCFDnCFPTRk, RSCFDnCFDF0_k, and RSCFDnCFDF1_k registers. <p>0: The transmit/receive FIFO buffer contains messages. 1: The transmit/receive FIFO buffer contains no message (buffer empty).</p> |

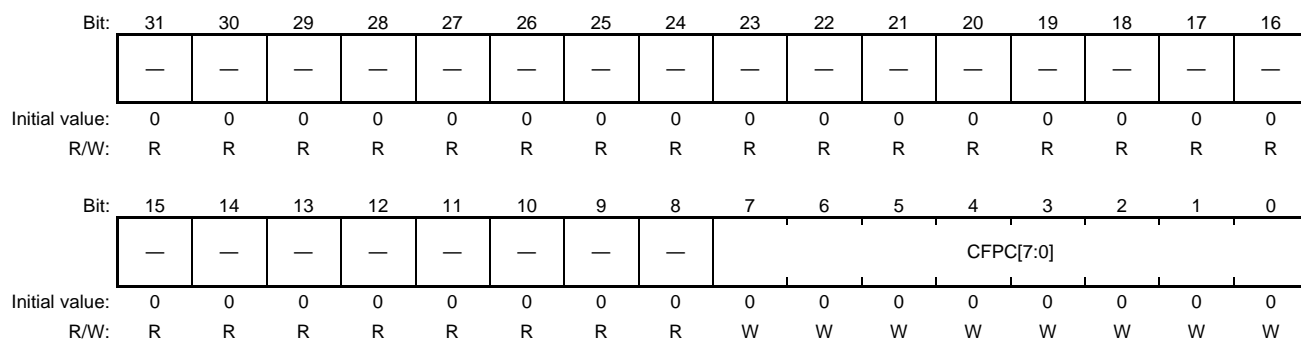
48.7.7.3 RSCFDnCFPCTRk — Transmit/receive FIFO Buffer Pointer Control Register (k = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFPCTRk register can only be written in 32-bit units
 RSCFDnCFPCTRkL, RSCFDnCFPCTRkH registers can only be written in 16-bit units
 RSCFDnCFPCTRkLL, RSCFDnCFPCTRkLH, RSCFDnCFPCTRkHL, RSCFDnCFPCTRkHH registers can only be written in 8-bit units

Address: RSCFDnCFPCTRk: <RSCFDn_base> + H'01D8 + (H'04 × k)
 RSCFDnCFPCTRkL: <RSCFDn_base> + H'01D8 + (H'04 × k),
 RSCFDnCFPCTRkH: <RSCFDn_base> + H'01DA + (H'04 × k)
 RSCFDnCFPCTRkLL: <RSCFDn_base> + H'01D8 + (H'04 × k),
 RSCFDnCFPCTRkLH: <RSCFDn_base> + H'01D9 + (H'04 × k),
 RSCFDnCFPCTRkHL: <RSCFDn_base> + H'01DA + (H'04 × k),
 RSCFDnCFPCTRkHH: <RSCFDn_base> + H'01DB + (H'04 × k)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | Reserved | All 0 | R | The write value should be the value after reset. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|--|
| 7 to 0 | CFPC[7:0] | H'00 | W | <p>Transmit/Receive FIFO Pointer Control</p> <ul style="list-style-type: none"> <p>Receive mode (CFM[1:0] value in the RSCFDnCFCCk register is B'00):</p> <p>Writing H'FF to the CFPC[7:0] bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value (transmit/receive FIFO message counter) in the RSCFDnCFSTSk register is decremented. Read the RSCFDnCFIDk, RSCFDnCFPTRk, RSCFDnCFDF0_k, and RSCFDnCFDF1_k registers to read messages from the transmit/receive FIFO buffer, and then write H'FF to the CFPC[7:0] bits.</p> <p>When writing H'FF to these bits, make sure that the CFE bit in the RSCFDnCFCCk register is set to 1 (transmit/receive FIFO buffers are used) and the CFEMP flag in the RSCFDnCFSTSk register is 0 (the transmit/receive FIFO buffer contains messages).</p> <p>Transmit mode (CFM[1:0] value in the RSCFDnCFCCk register is B'01):</p> <p>Writing H'FF to the CFPC[7:0] bits stores the data written to the RSCFDnCFIDk, RSCFDnCFPTRk, RSCFDnCFDF0_k, and RSCFDnCFDF1_k registers in the transmit/receive FIFO buffer and moves the write pointer to the next stage of the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value is incremented. Write transmit messages to the RSCFDnCFIDk, RSCFDnCFPTRk, RSCFDnCFDF0_k, and RSCFDnCFDF1_k registers before writing H'FF to the CFPC[7:0] bits.</p> <p>When writing H'FF to these bits, make sure that the CFE bit in the RSCFDnCFCCk register is set to 1 and the CFFLL flag in the RSCFDnCFSTSk register is 0 (the transmit/receive FIFO buffer is not full).</p> <p>Gateway mode (CFM[1:0] value in the RSCFDnCFCCk register is B'10):</p> <p>Setting prohibited</p> |

48.7.7.4 RSCFDnCFIDk — Transmit/receive FIFO Buffer Access ID Register (k = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFIDk register can be read/written in 32-bit units

RSCFDnCFIDkL, RSCFDnCFIDkH registers can be read/written in 16-bit units

RSCFDnCFIDkLL, RSCFDnCFIDkLH, RSCFDnCFIDkHL, RSCFDnCFIDkHH registers can be read/written in 8-bit units

Address: RSCFDnCFIDk: <RSCFDn_base> + H'0E80 + (H'10 × k)

RSCFDnCFIDkL: <RSCFDn_base> + H'0E80 + (H'10 × k),

RSCFDnCFIDkH: <RSCFDn_base> + H'0E82 + (H'10 × k)

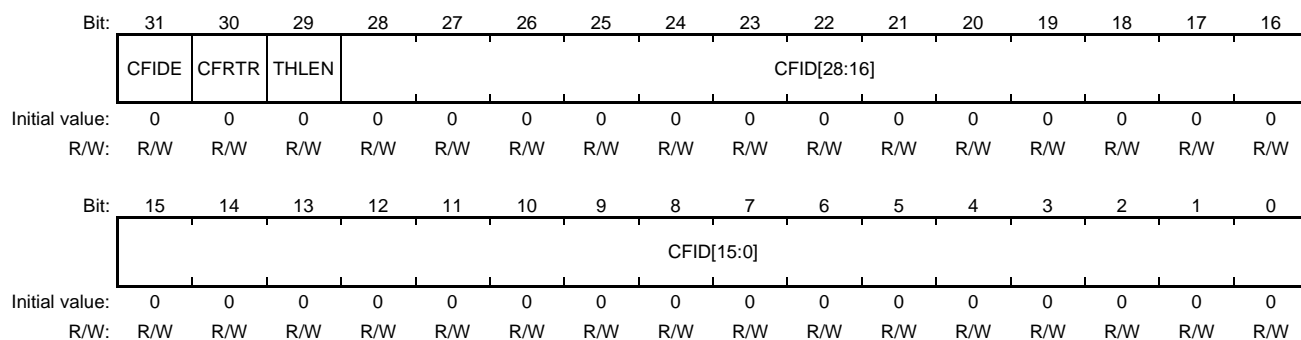
RSCFDnCFIDkLL: <RSCFDn_base> + H'0E80 + (H'10 × k),

RSCFDnCFIDkLH: <RSCFDn_base> + H'0E81 + (H'10 × k),

RSCFDnCFIDkHL: <RSCFDn_base> + H'0E82 + (H'10 × k),

RSCFDnCFIDkHH: <RSCFDn_base> + H'0E83 + (H'10 × k)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 31 | CFIDE | B'0 | R/W | Transmit/Receive FIFO Buffer IDE This bit indicates the ID format (standard ID or extended ID) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is B'00. When the CFM[1:0] value is B'01, these bits are used to set the ID format of the message to be transmitted from the transmit/receive FIFO buffer. 0: Standard ID 1: Extended ID |
| 30 | CFRTR | B'0 | R/W | Transmit/Receive FIFO Buffer RTR This bit indicates the data format (data frame or remote frame) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is B'00. When the CFM[1:0] value is B'01, this bit is used to set the data format of the message to be transmitted from the transmit/receive FIFO buffer. 0: Data frame 1: Remote frame |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 29 | THLEN | B'0 | R/W | <p>Transmit History Data Store Enable</p> <p>When this bit is set to 1, the transmit history data (label information, buffer number, and buffer type) of transmit messages is stored in the transmit history buffer after transmission is completed.</p> <p>This bit is enabled when the CFM[1:0] value is B'01 (transmit mode).</p> <p>0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.</p> |
| 28 to 0 | CFID[28:0] | H'0000_0000 | R/W | <p>Transmit/Receive FIFO Buffer ID Data</p> <ul style="list-style-type: none"> When CFM[1:0] value is B'01 (transmit mode): Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11. When CFM[1:0] value is B'00 (receive mode): Standard ID or extended ID in the received message can be read. For standard ID, read bits 10 to 0. Bits 28 to 11 are read as 0. |

This register is writable only when the CFM[1:0] value in the RSCFDnCFCCk register is B'01 (transmit mode). This register is readable only when the CFM[1:0] value is B'00 (receive mode). This register should not be read or written when the CFM[1:0] value is B'10 (gateway mode).

48.7.7.5 RSCFDnCFPTRk — Transmit/receive FIFO Buffer Access Pointer Register (k = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

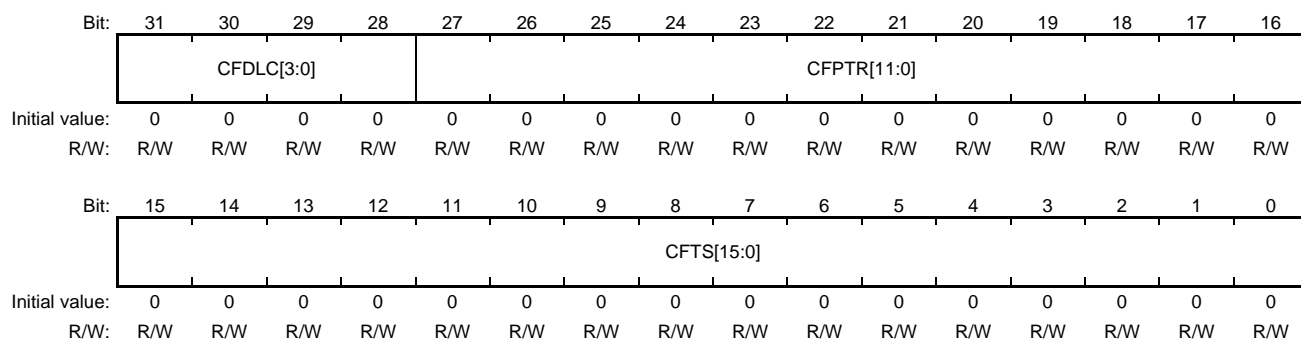
Access: RSCFDnCFPTRk register can be read/written in 32-bit units

RSCFDnCFPTRkL, RSCFDnCFPTRkH registers can be read/written in 16-bit units

RSCFDnCFPTRkLL, RSCFDnCFPTRkLH, RSCFDnCFPTRkHL, RSCFDnCFPTRkHH registers can be read/written in 8-bit units

Address: RSCFDnCFPTRk: <RSCFDn_base> + H'0E84 + (H'10 × k)
 RSCFDnCFPTRkL: <RSCFDn_base> + H'0E84 + (H'10 × k),
 RSCFDnCFPTRkH: <RSCFDn_base> + H'0E86 + (H'10 × k)
 RSCFDnCFPTRkLL: <RSCFDn_base> + H'0E84 + (H'10 × k),
 RSCFDnCFPTRkLH: <RSCFDn_base> + H'0E85 + (H'10 × k),
 RSCFDnCFPTRkHL: <RSCFDn_base> + H'0E86 + (H'10 × k),
 RSCFDnCFPTRkHH: <RSCFDn_base> + H'0E87 + (H'10 × k)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 28 | CFDLC[3:0] | B'0000 | R/W | Transmit/Receive FIFO Buffer DLC Data These bits indicate the data length of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is B'00. When the CFM[1:0] value is B'01, these bits are used to set the data length of the message to be transmitted from the transmit/receive FIFO buffer. If the data length is set to 9 bytes or more, the actual transmit data defaults to 8 bytes. B'0000: 0 data bytes B'0001: 1 data byte B'0010: 2 data bytes B'0011: 3 data bytes B'0100: 4 data bytes B'0101: 5 data bytes B'0110: 6 data bytes B'0111: 7 data bytes B'1XXX: 8 data bytes |
| 27 to 16 | CFPTR[11:0] | H'000 | R/W | Transmit/Receive FIFO Buffer Label Data <ul style="list-style-type: none"> • When CFM[1:0] value is B'01 (transmit mode): Set the label information to be stored in the transmit history buffer. Only bits CFPTR[7:0] are valid. • When CFM[1:0] value is B'00 (receive mode): The label information of the received message can be read. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 15 to 0 | CFTS[15:0] | H'0000 | R/W | Transmit/Receive FIFO Buffer Timestamp Data These bits are valid only when the CFM[1:0] value is B'00 (receive mode). The timestamp value of the received message can be read. |

This register is writable only when the CFM[1:0] value in the RSCFDnCFCCk register is B'01 (transmit mode). This register is readable only when the CFM[1:0] value is B'00 (receive mode). This register should not be read or written when the CFM[1:0] value is B'10 (gateway mode).

48.7.7.6 RSCFDnCFDF0_k — Transmit/receive FIFO Buffer Access Data Field 0 Register (k = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDF0_k register can be read/written in 32-bit units

RSCFDnCFDF0_kL, RSCFDnCFDF0_kH registers can be read/written in 16-bit units

RSCFDnCFDF0_kLL, RSCFDnCFDF0_kLH, RSCFDnCFDF0_kHL, RSCFDnCFDF0_kHH registers can be read/written in 8-bit units

Address: RSCFDnCFDF0_k: <RSCFDn_base> + H'0E88 + (H'10 × k)

RSCFDnCFDF0_kL: <RSCFDn_base> + H'0E88 + (H'10 × k),

RSCFDnCFDF0_kH: <RSCFDn_base> + H'0E8A + (H'10 × k)

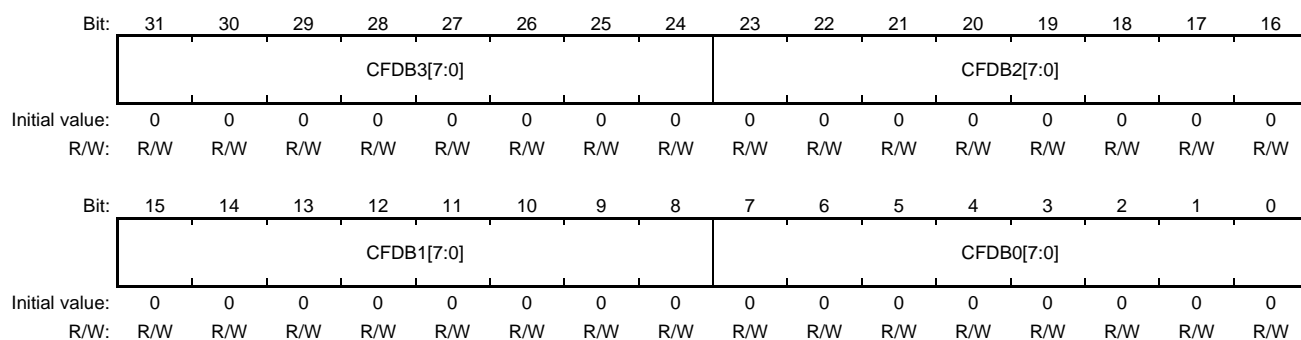
RSCFDnCFDF0_kLL: <RSCFDn_base> + H'0E88 + (H'10 × k),

RSCFDnCFDF0_kLH: <RSCFDn_base> + H'0E89 + (H'10 × k),

RSCFDnCFDF0_kHL: <RSCFDn_base> + H'0E8A + (H'10 × k),

RSCFDnCFDF0_kHH: <RSCFDn_base> + H'0E8B + (H'10 × k)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 24 | CFDB3[7:0] | H'00 | R/W | Transmit/Receive FIFO Buffer Data Byte 3 |
| 23 to 16 | CFDB2[7:0] | H'00 | R/W | Transmit/Receive FIFO Buffer Data Byte 2 |
| 15 to 8 | CFDB1[7:0] | H'00 | R/W | Transmit/Receive FIFO Buffer Data Byte 1 |
| 7 to 0 | CFDB0[7:0] | H'00 | R/W | Transmit/Receive FIFO Buffer Data Byte 0 |

- When CFM[1:0] value is B'01 (transmit mode):
Set the transmit/receive FIFO buffer data.
- When CFM[1:0] value is B'00 (receive mode):
The message data stored in the transmit/receive FIFO buffer can be read.

This register is writable only when the CFM[1:0] value in the RSCFDnCFCCk register is B'01 (transmit mode).

This register is readable only when the CFM[1:0] value is B'00 (receive mode). When the CFDLc[3:0] value in the RSCFDnCFPTRk register is smaller than B'1000, data bytes for which no data is set are read as H'00.

This register should not be read or written when the CFM[1:0] value is B'10 (gateway mode).

48.7.7.7 RSCFDnCFDF1_k — Transmit/receive FIFO Buffer Access Data Field 1 Register (k = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDF1_k register can be read/written in 32-bit units

RSCFDnCFDF1_kL, RSCFDnCFDF1_kH registers can be read/written in 16-bit units

RSCFDnCFDF1_kLL, RSCFDnCFDF1_kLH, RSCFDnCFDF1_kHL, RSCFDnCFDF1_kHH registers can be read/written in 8-bit units

Address: RSCFDnCFDF1_k: <RSCFDn_base> + H'0E8C + (H'10 × k)

RSCFDnCFDF1_kL: <RSCFDn_base> + H'0E8C + (H'10 × k),

RSCFDnCFDF1_kH: <RSCFDn_base> + H'0E8E + (H'10 × k)

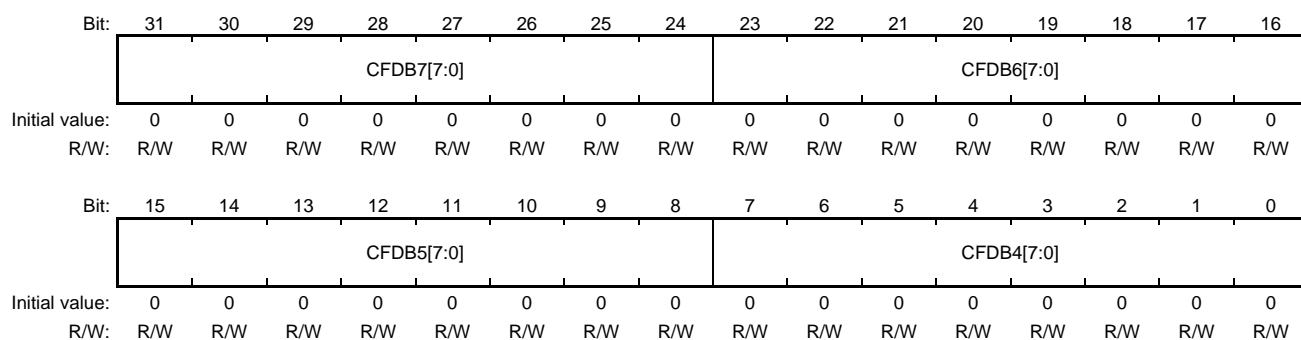
RSCFDnCFDF1_kLL: <RSCFDn_base> + H'0E8C + (H'10 × k),

RSCFDnCFDF1_kLH: <RSCFDn_base> + H'0E8D + (H'10 × k),

RSCFDnCFDF1_kHL: <RSCFDn_base> + H'0E8E + (H'10 × k),

RSCFDnCFDF1_kHH: <RSCFDn_base> + H'0E8F + (H'10 × k)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 24 | CFDB7[7:0] | H'00 | R/W | Transmit/Receive FIFO Buffer Data Byte 7 |
| 23 to 16 | CFDB6[7:0] | H'00 | R/W | Transmit/Receive FIFO Buffer Data Byte 6 |
| 15 to 8 | CFDB5[7:0] | H'00 | R/W | Transmit/Receive FIFO Buffer Data Byte 5 |
| 7 to 0 | CFDB4[7:0] | H'00 | R/W | Transmit/Receive FIFO Buffer Data Byte 4 |

- When CFM[1:0] value is B'01 (transmit mode):
Set the transmit/receive FIFO buffer data.
- When CFM[1:0] value is B'00 (receive mode):
The message data stored in the transmit/receive FIFO buffer can be read.

This register is writable only when the CFM[1:0] value in the RSCFDnCFCCk register is B'01 (transmit mode).

This register is readable only when the CFM[1:0] value is B'00 (receive mode). When the CFDLc[3:0] value in the RSCFDnCFPTRk register is smaller than B'1000, data bytes for which no data is set are read as H'00.

This register should not be read or written when the CFM[1:0] value is B'10 (gateway mode).

48.7.8 Details of FIFO Status Related Registers

48.7.8.1 RSCFDnFESTS — FIFO Empty Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnFESTS registers can be read only in 32-bit units

RSCFDnFESTSL, RSCFDnFESTSH registers can be read only in 16-bit units

RSCFDnFESTSLL, RSCFDnFESTSLH, RSCFDnFESTSHL, RSCFDnFESTSHH registers can be read only in 8-bit units

Address: RSCFDnFESTS: <RSCFDn_base> + H'0238

RSCFDnFESTSL: <RSCFDn_base> + H'0238, RSCFDnFESTSH: <RSCFDn_base> + H'023A

RSCFDnFESTSLL: <RSCFDn_base> + H'0238, RSCFDnFESTSLH: <RSCFDn_base> + H'0239,

RSCFDnFESTSHL: <RSCFDn_base> + H'023A, RSCFDnFESTSHH: <RSCFDn_base> + H'023B

Value after reset: H'03FF_FFFF

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | CF17E MP | CF16E MP | CF15E MP | CF14E MP | CF13E MP | CF12E MP | CF11E MP | CF10E MP | CF9EM P | CF8EM P |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CF7EM P | CF6EM P | CF5EM P | CF4EM P | CF3EM P | CF2EM P | CF1EM P | CF0EM P | RF7EM P | RF6EM P | RF5EM P | RF4EM P | RF3EM P | RF2EM P | RF1EM P | RF0EM P |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 26 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 25 | CF17EMP | B'1 | R | Transmit/Receive FIFO Buffer Empty Status Flag |
| 24 | CF16EMP | B'1 | R | The CFkEMP flag is set to 1 when the CFEMP flag in the RSCFDnCFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)). When the CFEMP flag is cleared to 0 (the transmit/receive FIFO buffer contains messages), the CFkEMP flag is cleared to 0. 0: Transmit/receive FIFO buffer k contains a message. 1: Transmit/receive FIFO buffer k contains no message. (k = 0 to 5) |
| 23 | CF15EMP | B'1 | R | |
| 22 | CF14EMP | B'1 | R | |
| 21 | CF13EMP | B'1 | R | |
| 20 | CF12EMP | B'1 | R | |
| 19 | CF11EMP | B'1 | R | |
| 18 | CF10EMP | B'1 | R | |
| 17 | CF9EMP | B'1 | R | |
| 16 | CF8EMP | B'1 | R | |
| 15 | CF7EMP | B'1 | R | |
| 14 | CF6EMP | B'1 | R | |
| 13 | CF5EMP | B'1 | R | |
| 12 | CF4EMP | B'1 | R | |
| 11 | CF3EMP | B'1 | R | |
| 10 | CF2EMP | B'1 | R | |
| 9 | CF1EMP | B'1 | R | |
| 8 | CF0EMP | B'1 | R | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | RF7EMP | B'1 | R | Receive FIFO Buffer Empty Status Flag |
| 6 | RF6EMP | B'1 | R | The RFxEMP flag is set to 1 when the RFEMP flag in the RSCFDnRFSTs register is set to 1 (the receive FIFO buffer contains no unread message). When the RFEMP flag is cleared to 0 (the receive FIFO buffer contains unread messages), the RFxEMP flag is cleared to 0. 0: Receive FIFO buffer x contains an unread message. 1: Receive FIFO buffer x contains no unread message (buffer empty). (x = 0 to 7) |
| 5 | RF5EMP | B'1 | R | |
| 4 | RF4EMP | B'1 | R | |
| 3 | RF3EMP | B'1 | R | |
| 2 | RF2EMP | B'1 | R | |
| 1 | RF1EMP | B'1 | R | |
| 0 | RF0EMP | B'1 | R | |

The RSCFDnRFSTs register is set to H'03FF_FFFF in global reset mode.

48.7.8.2 RSCFDnFFSTS — FIFO Full Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnFFSTS register can be read only in 32-bit units

RSCFDnFFSTS_L, RSCFDnFFSTS_H registers can be read only in 16-bit units

RSCFDnFFSTS_{LL}, RSCFDnFFSTS_{SLH}, RSCFDnFFSTS_{SHL}, RSCFDnFFSTS_{SHH} registers can be read only in 8-bit units

Address: RSCFDnFFSTS: <RSCFDn_base> + H'023C

RSCFDnFFSTS_L: <RSCFDn_base> + H'023C, RSCFDnFFSTS_H: <RSCFDn_base> + H'023E

RSCFDnFFSTS_{LL}: <RSCFDn_base> + H'023C, RSCFDnFFSTS_{SLH}: <RSCFDn_base> + H'023D,

RSCFDnFFSTS_{SHL}: <RSCFDn_base> + H'023E, RSCFDnFFSTS_{SHH}: <RSCFDn_base> + H'023F

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | CF17F LL | CF16F LL | CF15F LL | CF14F LL | CF13F LL | CF12F LL | CF11F LL | CF10F LL | CF9FL L | CF8FL L |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CF7FL L | CF6FL L | CF5FL L | CF4FL L | CF3FL L | CF2FL L | CF1FL L | CF0FL L | RF7FL L | RF6FL L | RF5FL L | RF4FL L | RF3FL L | RF2FL L | RF1FL L | RF0FL L |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 26 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 25 | CF17FLL | B'0 | R | Transmit/Receive FIFO Buffer Full Status Flag |
| 24 | CF16FLL | B'0 | R | The CFkFLL flag is set to 1 when the CFFLL flag in the RSCFDnCFSTS _k register is set to 1 (the transmit/receive FIFO buffer is full). |
| 23 | CF15FLL | B'0 | R | |
| 22 | CF14FLL | B'0 | R | When the CFFLL flag is cleared to 0 (the transmit/receive FIFO buffer is not full), the CFkFLL flag is cleared to 0. |
| 21 | CF13FLL | B'0 | R | |
| 20 | CF12FLL | B'0 | R | 0: Transmit/receive buffer k is not full. |
| 19 | CF11FLL | B'0 | R | 1: Transmit/receive buffer k is full. |
| 18 | CF10FLL | B'0 | R | (k = 0 to 5) |
| 17 | CF9FLL | B'0 | R | |
| 16 | CF8FLL | B'0 | R | |
| 15 | CF7FLL | B'0 | R | |
| 14 | CF6FLL | B'0 | R | |
| 13 | CF5FLL | B'0 | R | |
| 12 | CF4FLL | B'0 | R | |
| 11 | CF3FLL | B'0 | R | |
| 10 | CF2FLL | B'0 | R | |
| 9 | CF1FLL | B'0 | R | |
| 8 | CF0FLL | B'0 | R | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | RF7FLL | B'0 | R | Receive FIFO Buffer Full Status Flag |
| 6 | RF6FLL | B'0 | R | The RFXFLL flag is set to 1 when the RFFLL flag in the RSCFDnRFSTsX register is set to 1 (the receive FIFO buffer is full). When the RFFLL flag is cleared to 0 (the receive FIFO buffer is not full), the RFXFLL flag is cleared to 0. |
| 5 | RF5FLL | B'0 | R | |
| 4 | RF4FLL | B'0 | R | 0: Receive FIFO buffer x is not full. 1: Receive FIFO buffer x is full. (x = 0 to 7) |
| 3 | RF3FLL | B'0 | R | |
| 2 | RF2FLL | B'0 | R | |
| 1 | RF1FLL | B'0 | R | |
| 0 | RF0FLL | B'0 | R | |

The RSCFDnFFSTS register is cleared to H'0000_0000 in global reset mode.

48.7.8.3 RSCFDnFMSTS — FIFO Message Lost Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnFMSTS register can be read only in 32-bit units

RSCFDnFMSTSLL, RSCFDnFMSTSLLH registers can be read only in 16-bit units

RSCFDnFMSTSLL, RSCFDnFMSTSLLH, RSCFDnFMSTSLLH, RSCFDnFMSTSLLH registers can be read only in 8-bit units

Address: RSCFDnFMSTS: <RSCFDn_base> + H'0240

RSCFDnFMSTSLL: <RSCFDn_base> + H'0240, RSCFDnFMSTSLLH: <RSCFDn_base> + H'0242

RSCFDnFMSTSLL: <RSCFDn_base> + H'0240, RSCFDnFMSTSLLH: <RSCFDn_base> + H'0241,

RSCFDnFMSTSLLH: <RSCFDn_base> + H'0242, RSCFDnFMSTSLLHH: <RSCFDn_base> + H'0243

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | CF17ML LT | CF16ML LT | CF15ML LT | CF14ML LT | CF13ML LT | CF12ML LT | CF11ML LT | CF10ML LT | CF9ML T | CF8ML T |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CF7ML T | CF6ML T | CF5ML T | CF4ML T | CF3ML T | CF2ML T | CF1ML T | CF0ML T | RF7ML T | RF6ML T | RF5ML T | RF4ML T | RF3ML T | RF2ML T | RF1ML T | RF0ML T |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 26 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 25 | CF17MLT | B'0 | R | Transmit/Receive FIFO Buffer Message Lost Status Flag |
| 24 | CF16MLT | B'0 | R | The CFkMLT flag is set to 1 when the CFMLT flag in the RSCFDnCFSTSk register is set to 1 (a transmit/receive FIFO message is lost). When the CFMLT flag is cleared to 0, the CFkMLT flag is cleared to 0. |
| 23 | CF15MLT | B'0 | R | |
| 22 | CF14MLT | B'0 | R | |
| 21 | CF13MLT | B'0 | R | 0: No transmit/receive FIFO buffer k message is lost. |
| 20 | CF12MLT | B'0 | R | 1: A transmit/receive FIFO buffer k message is lost. |
| 19 | CF11MLT | B'0 | R | (k = 0 to 5) |
| 18 | CF10MLT | B'0 | R | |
| 17 | CF9MLT | B'0 | R | |
| 16 | CF8MLT | B'0 | R | |
| 15 | CF7MLT | B'0 | R | |
| 14 | CF6MLT | B'0 | R | |
| 13 | CF5MLT | B'0 | R | |
| 12 | CF4MLT | B'0 | R | |
| 11 | CF3MLT | B'0 | R | |
| 10 | CF2MLT | B'0 | R | |
| 9 | CF1MLT | B'0 | R | |
| 8 | CF0MLT | B'0 | R | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | RF7MLT | B'0 | R | Receive FIFO Buffer Message Lost Status Flag |
| 6 | RF6MLT | B'0 | R | The RFxMLT flag is set to 1 when the RFMLT flag in the RSCFDnRFSTsX register is set to 1 (a receive FIFO message is lost). When the RFMLT flag is cleared to 0, the RFxMLT flag is cleared to 0. |
| 5 | RF5MLT | B'0 | R | |
| 4 | RF4MLT | B'0 | R | 0: No receive FIFO buffer x message is lost. 1: A receive FIFO buffer x message is lost. (x = 0 to 7) |
| 3 | RF3MLT | B'0 | R | |
| 2 | RF2MLT | B'0 | R | |
| 1 | RF1MLT | B'0 | R | |
| 0 | RF0MLT | B'0 | R | |

The RSCFDnFMSTS register is cleared to H'0000_0000 in global reset mode.

48.7.8.4 RSCFDnRFISTS — Receive FIFO Buffer Interrupt Flag Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnRFISTS register can be read only in 32-bit units

RSCFDnRFISTS_{SL}, RSCFDnRFISTS_{SH} registers can be read only in 16-bit units

RSCFDnRFISTS_{SLL}, RSCFDnRFISTS_{SLH}, RSCFDnRFISTS_{SHL}, RSCFDnRFISTS_{SHH} registers can be read only in 8-bit units

Address: RSCFDnRFISTS: <RSCFDn_base> + H'0244

RSCFDnRFISTS_{SL}: <RSCFDn_base> + H'0244, RSCFDnRFISTS_{SH}: <RSCFDn_base> + H'0246

RSCFDnRFISTS_{SLL}: <RSCFDn_base> + H'0244, RSCFDnRFISTS_{SLH}: <RSCFDn_base> + H'0245,

RSCFDnRFISTS_{SHL}: <RSCFDn_base> + H'0246, RSCFDnRFISTS_{SHH}: <RSCFDn_base> + H'0247

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | RF7IF | RF6IF | RF5IF | RF4IF | RF3IF | RF2IF | RF1IF | RF0IF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 7 | RF7IF | B'0 | R | Receive FIFO Buffer Interrupt Request Status Flag |
| 6 | RF6IF | B'0 | R | The RFxIF flag is set to 1 when the RFIF flag in the RSCFDnRFISTS _x register is set to 1 (a receive FIFO interrupt request is present). When the RFIF flag is cleared to 0, the RFxIF flag is cleared to 0. |
| 5 | RF5IF | B'0 | R | |
| 4 | RF4IF | B'0 | R | |
| 3 | RF3IF | B'0 | R | 0: No receive FIFO buffer x interrupt request is present. |
| 2 | RF2IF | B'0 | R | 1: A receive FIFO buffer x interrupt request is present. |
| 1 | RF1IF | B'0 | R | (x = 0 to 7) |
| 0 | RF0IF | B'0 | R | |

The RSCFDnRFISTS register is cleared to H'0000_0000 in global reset mode.

48.7.8.5 RSCFDnCFRISTS — Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFRISTS register can be read only in 32-bit units

RSCFDnCFRISTSL, RSCFDnCFRISTSH registers can be read only in 16-bit units

RSCFDnCFRISTSLL, RSCFDnCFRISTSLH, RSCFDnCFRISTSHL, RSCFDnCFRISTSHH registers can be read only in 8-bit units

Address: RSCFDnCFRISTS: <RSCFDn_base> + H'0248

RSCFDnCFRISTSL: <RSCFDn_base> + H'0248, RSCFDnCFRISTSH: <RSCFDn_base> + H'024A

RSCFDnCFRISTSLL: <RSCFDn_base> + H'0248, RSCFDnCFRISTSLH: <RSCFDn_base> + H'0249,

RSCFDnCFRISTSHL: <RSCFDn_base> + H'024A, RSCFDnCFRISTSHH: <RSCFDn_base> + H'024B

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|--------------|--------------|--------------|--------------|--------------|--------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CF17R XIF | CF16R XIF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CF15R XIF | CF14R XIF | CF13R XIF | CF12R XIF | CF11R XIF | CF10R XIF | CF9RXI F | CF8RXI F | CF7RXI F | CF6RXI F | CF5RXI F | CF4RXI F | CF3RXI F | CF2RXI F | CF1RXI F | CF0RXI F |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 18 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 17 | CF17RXIF | B'0 | R | Transmit/Receive FIFO Buffer Receive Interrupt Request Status Flag |
| 16 | CF16RXIF | B'0 | R | |
| 15 | CF15RXIF | B'0 | R | The CFkRXIF flag is set to 1 when the CFRXIF flag in the RSCFDnCFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). When the CFRXIF flag is cleared to 0, the CFkRXIF flag is cleared to 0. |
| 14 | CF14RXIF | B'0 | R | |
| 13 | CF13RXIF | B'0 | R | |
| 12 | CF12RXIF | B'0 | R | 0: No transmit/receive FIFO buffer k receive interrupt request is present. |
| 11 | CF11RXIF | B'0 | R | 1: A transmit/receive FIFO buffer k receive interrupt request is present. |
| 10 | CF10RXIF | B'0 | R | |
| 9 | CF9RXIF | B'0 | R | (k = 0 to 5) |
| 8 | CF8RXIF | B'0 | R | |
| 7 | CF7RXIF | B'0 | R | |
| 6 | CF6RXIF | B'0 | R | |
| 5 | CF5RXIF | B'0 | R | |
| 4 | CF4RXIF | B'0 | R | |
| 3 | CF3RXIF | B'0 | R | |
| 2 | CF2RXIF | B'0 | R | |
| 1 | CF1RXIF | B'0 | R | |
| 0 | CF0RXIF | B'0 | R | |

The RSCFDnCFRISTS register is cleared to H'0000_0000 in global reset mode.

48.7.8.6 RSCFDnCFTISTS — Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFTISTS register can be read only in 32-bit units

RSCFDnCFTISTSL, RSCFDnCFTISTSH registers can be read only in 16-bit units

RSCFDnCFTISTSL, RSCFDnCFTISTSLH, RSCFDnCFTISTSHL, RSCFDnCFTISTSHH registers can be read only in 8-bit units

Address: RSCFDnCFTISTS: <RSCFDn_base> + H'024C

RSCFDnCFTISTSL: <RSCFDn_base> + H'024C, RSCFDnCFTISTSH: <RSCFDn_base> + H'024E

RSCFDnCFTISTSL: <RSCFDn_base> + H'024C, RSCFDnCFTISTSLH: <RSCFDn_base> + H'024D,

RSCFDnCFTISTSHL: <RSCFDn_base> + H'024E, RSCFDnCFTISTSHH: <RSCFDn_base> + H'024F

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|--------------|--------------|--------------|--------------|--------------|--------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CF17T XIF | CF16T XIF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CF15T XIF | CF14T XIF | CF13T XIF | CF12T XIF | CF11T XIF | CF10T XIF | CF9TXI F | CF8TXI F | CF7TXI F | CF6TXI F | CF5TXI F | CF4TXI F | CF3TXI F | CF2TXI F | CF1TXI F | CF0TXI F |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 18 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 17 | CF17TXIF | B'0 | R | Transmit/Receive FIFO Buffer Transmit Interrupt Request Status Flag |
| 16 | CF16TXIF | B'0 | R | |
| 15 | CF15TXIF | B'0 | R | The CFkTXIF flag is set to 1 when the CFTXIF flag in the RSCFDnCFSTSk register is set to 1 (a transmit/receive FIFO transmit interrupt request is present). When the CFTXIF flag is cleared to 0, the CFkTXIF flag is cleared to 0. |
| 14 | CF14TXIF | B'0 | R | |
| 13 | CF13TXIF | B'0 | R | |
| 12 | CF12TXIF | B'0 | R | 0: No transmit/receive FIFO buffer k transmit interrupt request is present. |
| 11 | CF11TXIF | B'0 | R | 1: A transmit/receive FIFO buffer k transmit interrupt request is present. |
| 10 | CF10TXIF | B'0 | R | |
| 9 | CF9TXIF | B'0 | R | (k = 0 to 5) |
| 8 | CF8TXIF | B'0 | R | |
| 7 | CF7TXIF | B'0 | R | |
| 6 | CF6TXIF | B'0 | R | |
| 5 | CF5TXIF | B'0 | R | |
| 4 | CF4TXIF | B'0 | R | |
| 3 | CF3TXIF | B'0 | R | |
| 2 | CF2TXIF | B'0 | R | |
| 1 | CF1TXIF | B'0 | R | |
| 0 | CF0TXIF | B'0 | R | |

The RSCFDnCFTISTS register is cleared to H'0000_0000 in global reset mode.

48.7.9 Details of Transmit Buffer Related Registers

48.7.9.1 RSCFDnTMCp — Transmit Buffer Control Register (p = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTMCp register can be read/written in 8-bit units

Address: RSCFDnTMCp: <RSCFDn_base> + H'0250 + (H'01 × p)

Value after reset: H'00

| | | | | | | | | |
|------|---|---|---|---|---|------|-------|------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | TMOM | TMTAR | TMTR |

Initial value: 0 0 0 0 0 0 0 0 0

R/W: R R R R R R/W* R/W* R/W*

Note: * The only effective value for writing to this bit is 1, which sets the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|------|---|
| 7 to 3 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 2 | TMOM | B'0 | R/W* | One-Shot Transmission Enable Setting this bit to 1 enables one-shot transmission. When transmission fails, retransmission defined in the CAN protocol is not performed. Modify the TMOM bit when the TMTRM flag in the RSCFDnTMSTSp register is set to 0. Set the TMOM bit to 1 together with the TMTR bit. 0: One-shot transmission is disabled. 1: One-shot transmission is enabled. |
| 1 | TMTAR | B'0 | R/W* | Transmit Abort Request Setting this bit to 1 generates a transmit abort request for the message stored in the transmit buffer. However, a message that is being transmitted or one that will be transmitted next cannot be aborted. The TMTAR bit can be set to 1 when TMTR bit is 1. The TMTAR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit. Transmission has been completed. Transmit abort has been completed. An error or arbitration loss has been detected. If this bit becomes 0 at the same time as the program writes 1 to this bit, this bit becomes 0. 0: Transmit abort is not requested. 1: Transmit abort is requested. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 0 | TMTR | B'0 | R/W* | <p>Transmit Request</p> <p>Setting this bit to 1 transmits the message stored in the transmit buffer.</p> <p>The TMTR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.</p> <ul style="list-style-type: none"> Transmission has been completed. Transmit abort has been completed after the TMTAR bit was set to 1. An error or arbitration-lost has been detected with the TMOM bit set to 1. <p>Set the TMTR bit to 1 when the value of TMTRF[1:0] in the RSCFDnTMSTSp register is B'00.</p> <p>0: Transmission is not requested. 1: Transmission is requested.</p> |

When the RSCFDnTMCp register meets any of the following conditions, set it to H'00.

The RSCFDnTMCp register corresponds to the transmit buffer number selected by the CFTML[3:0] bits in the RSCFDnCFCCk register ($p = m \times 16 + \text{the value of CFTML}[3:0] \text{ bits}$).

The RSCFDnTMCp register corresponds to the transmit buffer allocated to the transmit queue by the TXQDC[3:0] bits in the RSCFDnTXQCCm ($m = 0 \text{ or } 1$) register ($p = (m \times 16 + 15) \text{ to } (m \times 16 + 15 - \text{the value of TXQDC}[3:0] \text{ bits})$).

Bits in the RSCFDnTMCp register are all cleared to 0 in channel reset mode. Modify the RSCFDnTMCp register in channel communication mode or channel halt mode.

48.7.9.2 RSCFDnTMSTSp — Transmit Buffer Status Register (p = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTMSTSp register can be read/written in 8-bit units

Address: RSCFDnTMSTSp: <RSCFDn_base> + H'02D0 + (H'01 × p)

Value after reset: H'00

| | | | | | | | | |
|----------------|---|---|---|------------|-----------|------------|------------|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | TMTAR M | TMTR M | TMTRF[1:0] | TMTST S | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|---|
| 7 to 5 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 4 | TMTARM | B'0 | R | Transmit Buffer Transmit Abort Request Status Flag The TMTARM flag is set to 1 when the TMTAR bit in the RSCFDnTMCP register is set to 1. The TMTARM flag is set to 0 when the TMTAR bit in the RSCFDnTMCP register is set to 0. 0: No transmit abort request is present. 1: A transmit abort request is present. |
| 3 | TMTRM | B'0 | R | Transmit Buffer Transmit Request Status Flag The TMTRM flag is set to 1 when the TMTR bit in the RSCFDnTMCP register is set to 1. The TMTRM flag is set to 0 when the TMTR bit in the RSCFDnTMCP register is set to 0. 0: No transmit request is present. 1: A transmit request is present. |
| 2, 1 | TMTRF[1:0] | B'00 | R/W | Transmit Buffer Transmit Result Status Flag This flag indicates the result of transmission from the transmit buffer. B'00: Transmission is in progress or no transmit request is present. B'01: Transmission from the transmit buffer was aborted. B'10: Transmission has been completed with the TMTAR bit in the RSCFDnTMCP register set to 0 (transmit abort is not requested). B'11: Transmission has been completed with the TMTAR bit in the RSCFDnTMCP register set to 1 (transmit abort is requested). Write B'00 to the TMTRF[1:0] flag in channel communication mode or channel halt mode. Do not write any value other than B'00 to this flag. |
| 0 | TMTSTS | B'0 | R | Transmit Buffer Transmit Status Flag This flag is set to 1 when transmission from the transmit buffer starts, and is cleared to 0 when transmission from the transmit buffer has been completed or terminated due to a bus error or arbitration lost. 0: Transmission is not in progress. 1: Transmission is in progress. |

The RSCFDnTMSTSp register is cleared to all 0 in channel reset mode.

48.7.9.3 RSCFDnTMIDp — Transmit Buffer ID Register (p = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTMIDp register can be read/written in 32-bit units

RSCFDnTMIDpL, RSCFDnTMIDpH registers can be read/written in 16-bit units

RSCFDnTMIDpLL, RSCFDnTMIDpLH, RSCFDnTMIDpHL, RSCFDnTMIDpHH registers can be read/written in 8-bit units

Address: RSCFDnTMIDp: <RSCFDn_base> + H'1000 + (H'10 × p)

RSCFDnTMIDpL: <RSCFDn_base> + H'1000 + (H'10 × p),

RSCFDnTMIDpH: <RSCFDn_base> + H'1002 + (H'10 × p)

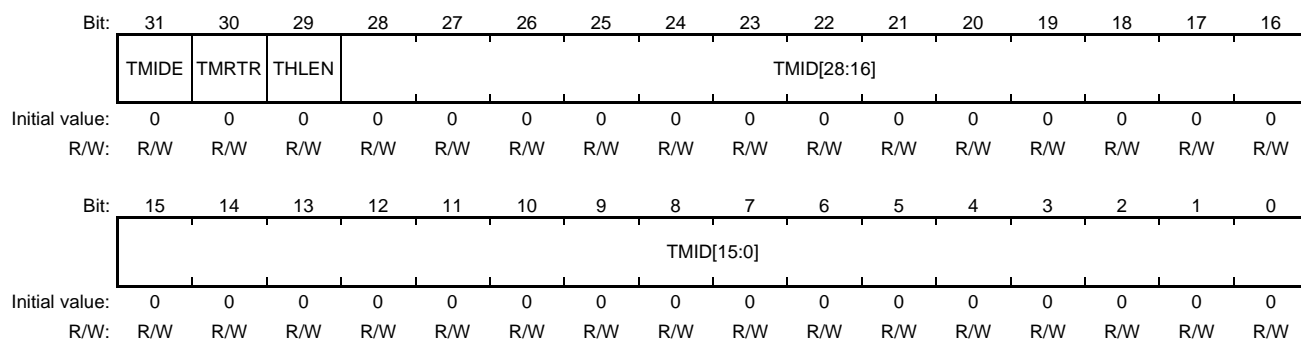
RSCFDnTMIDpLL: <RSCFDn_base> + H'1000 + (H'10 × p),

RSCFDnTMIDpLH: <RSCFDn_base> + H'1001 + (H'10 × p),

RSCFDnTMIDpHL: <RSCFDn_base> + H'1002 + (H'10 × p),

RSCFDnTMIDpHH: <RSCFDn_base> + H'1003 + (H'10 × p)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 31 | TMIDE | B'0 | R/W | Transmit Buffer IDE This bit is used to set the ID format of the message to be transmitted from the transmit buffer. 0: Standard ID 1: Extended ID |
| 30 | TMRTR | B'0 | R/W | Transmit Buffer RTR This bit is used to set the data format of the message to be transmitted from the transmit buffer. 0: Data frame 1: Remote frame |
| 29 | THLEN | B'0 | R/W | Transmit History Data Store Enable With this bit set to 1, the transmit history data of the message transmitted (label information, buffer number, and buffer type; if the TMTSCE bit in the RSCFDnGCFG register is 1, timestamp is also included) is stored in the transmit history buffer after transmission is completed. 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 28 to 0 | TMID[28:0] | H'0000_0000 | R/W | Transmit Buffer ID Data These bits are used to set the ID of the message to be transmitted from the transmit buffer. Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11. |

Modify this register when the TMTRM bit in the corresponding RSCFDnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write data to this register. If this register is allocated to the transmit queue, only write data to a transmit buffer p ($p = m \times 16 + 15$) for the corresponding channel.

48.7.9.4 RSCFDnTMPTRp — Transmit Buffer Pointer Register (p = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTMPTRp register can be read/written in 32-bit units

RSCFDnTMPTRpL, RSCFDnTMPTRpH registers can be read/written in 16-bit units

RSCFDnTMPTRpLL, RSCFDnTMPTRpLH, RSCFDnTMPTRpHL, RSCFDnTMPTRpHH registers can be read/written in 8-bit units

Address: RSCFDnTMPTRp: <RSCFDn_base> + H'1004 + (H'10 × p)

RSCFDnTMPTRpL: <RSCFDn_base> + H'1004 + (H'10 × p),

RSCFDnTMPTRpH: <RSCFDn_base> + H'1006 + (H'10 × p)

RSCFDnTMPTRpLL: <RSCFDn_base> + H'1004 + (H'10 × p),

RSCFDnTMPTRpLH: <RSCFDn_base> + H'1005 + (H'10 × p),

RSCFDnTMPTRpHL: <RSCFDn_base> + H'1006 + (H'10 × p),

RSCFDnTMPTRpHH: <RSCFDn_base> + H'1007 + (H'10 × p)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----|-----|-----|----|----|----|----|------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TMDLC[3:0] | | | | — | — | — | — | TMPTR[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 28 | TMDLC[3:0] | B'0000 | R/W | Transmit Buffer DLC Data These bits are used to set the data length of the message to be transmitted from the transmit buffer when the TMRTR bit in the RSCFDnTMIDp register is set to 0 (data frame). If the data length is set to 9 bytes or more, the transmit data is 8 bytes long. When the TMRTR bit is set to 1 (remote frame), set the data length of messages to be requested. B'0000: 0 data bytes B'0001: 1 data byte B'0010: 2 data bytes B'0011: 3 data bytes B'0100: 4 data bytes B'0101: 5 data bytes B'0110: 6 data bytes B'0111: 7 data bytes B'1XXX: 8 data bytes |
| 27 to 24 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 23 to 16 | TMPTR[7:0] | H'00 | R/W | Transmit Buffer Label Data When message transmission has been completed, the TMPTR[7:0] value is stored in the transmit history buffer. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 15 to 0 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |

Modify this register when the TMTRM bit in the corresponding RSCFDnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p ($p = m \times 16 + 15$) for the corresponding channel.

48.7.9.5 RSCFDnTMDF0_p — Transmit Buffer Data Field 0 Register (p = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

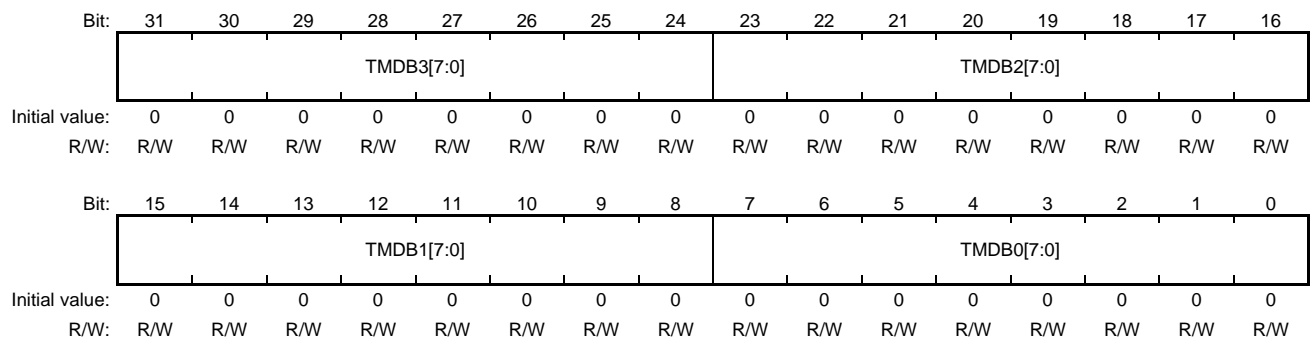
Access: RSCFDnTMDF0_p register can be read/written in 32-bit units

RSCFDnTMDF0_pL, RSCFDnTMDF0_pH registers can be read/written in 16-bit units

RSCFDnTMDF0_pLL, RSCFDnTMDF0_pLH, RSCFDnTMDF0_pHL, RSCFDnTMDF0_pHH registers can be read/written in 8-bit units

Address: RSCFDnTMDF0_p: <RSCFDn_base> + H'1008 + (H'10 × p)
 RSCFDnTMDF0_pL: <RSCFDn_base> + H'1008 + (H'10 × p),
 RSCFDnTMDF0_pH: <RSCFDn_base> + H'100A + (H'10 × p)
 RSCFDnTMDF0_pLL: <RSCFDn_base> + H'1008 + (H'10 × p),
 RSCFDnTMDF0_pLH: <RSCFDn_base> + H'1009 + (H'10 × p),
 RSCFDnTMDF0_pHL: <RSCFDn_base> + H'100A + (H'10 × p),
 RSCFDnTMDF0_pHH: <RSCFDn_base> + H'100B + (H'10 × p)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|-----------------------------|
| 31 to 24 | TMDB3[7:0] | H'00 | R/W | Transmit Buffer Data Byte 3 |
| 23 to 16 | TMDB2[7:0] | H'00 | R/W | Transmit Buffer Data Byte 2 |
| 15 to 8 | TMDB1[7:0] | H'00 | R/W | Transmit Buffer Data Byte 1 |
| 7 to 0 | TMDB0[7:0] | H'00 | R/W | Transmit Buffer Data Byte 0 |

Set the transmit buffer data.

Modify this register when the TMTRM bit in the corresponding RSCFDnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p ($p = m \times 16 + 15$) for the corresponding channel.

48.7.9.6 RSCFDnTMDF1_p — Transmit Buffer Data Field 1 Register (p = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTMDF1_p register can be read/written in 32-bit units

RSCFDnTMDF1_pL, RSCFDnTMDF1_pH registers can be read/written in 16-bit units

RSCFDnTMDF1_pLL, RSCFDnTMDF1_pLH, RSCFDnTMDF1_pHL, RSCFDnTMDF1_pHH registers can be read/written in 8-bit units

Address: RSCFDnTMDF1_p: <RSCFDn_base> + H'100C + (H'10 × p)

RSCFDnTMDF1_pL: <RSCFDn_base> + H'100C + (H'10 × p),

RSCFDnTMDF1_pH: <RSCFDn_base> + H'100E + (H'10 × p)

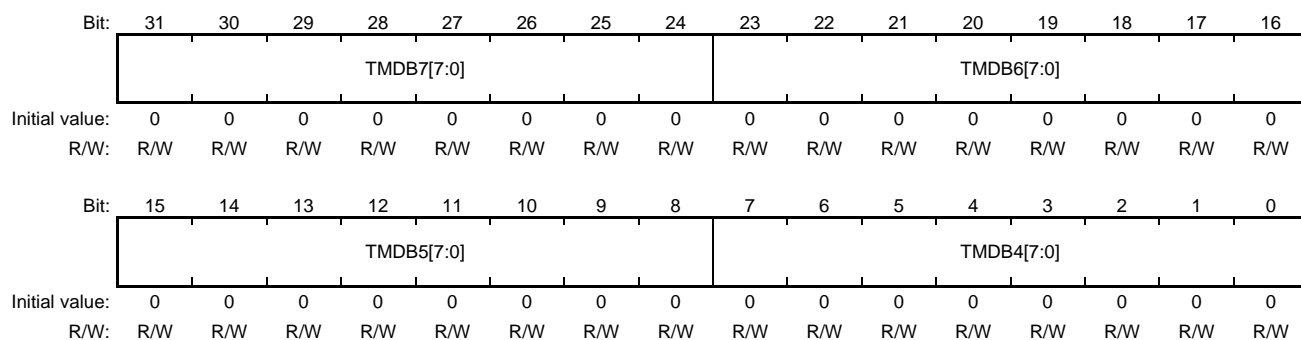
RSCFDnTMDF1_pLL: <RSCFDn_base> + H'100C + (H'10 × p),

RSCFDnTMDF1_pLH: <RSCFDn_base> + H'100D + (H'10 × p),

RSCFDnTMDF1_pHL: <RSCFDn_base> + H'100E + (H'10 × p),

RSCFDnTMDF1_pHH: <RSCFDn_base> + H'100F + (H'10 × p)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|-----------------------------|
| 31 to 24 | TMDB7[7:0] | H'00 | R/W | Transmit Buffer Data Byte 7 |
| 23 to 16 | TMDB6[7:0] | H'00 | R/W | Transmit Buffer Data Byte 6 |
| 15 to 8 | TMDB5[7:0] | H'00 | R/W | Transmit Buffer Data Byte 5 |
| 7 to 0 | TMDB4[7:0] | H'00 | R/W | Transmit Buffer Data Byte 4 |

Set the transmit buffer data.

Modify this register when the TMTRM bit in the corresponding RSCFDnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p ($p = m \times 16 + 15$) for the corresponding channel.

48.7.9.7 RSCFDnTMIECy — Transmit Buffer Interrupt Enable Configuration Register (y = 0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTMIECy register can be read/written in 32-bit units

RSCFDnTMIECyL, RSCFDnTMIECyH registers can be read/written in 16-bit units

RSCFDnTMIECyLL, RSCFDnTMIECyLH, RSCFDnTMIECyHL, RSCFDnTMIECyHH registers can be read/written in 8-bit units

Address: RSCFDnTMIECy: <RSCFDn_base> + H'0390 + (H'04 × y)

RSCFDnTMIECyL: <RSCFDn_base> + H'0390 + (H'04 × y),

RSCFDnTMIECyH: <RSCFDn_base> + H'0392 + (H'04 × y)

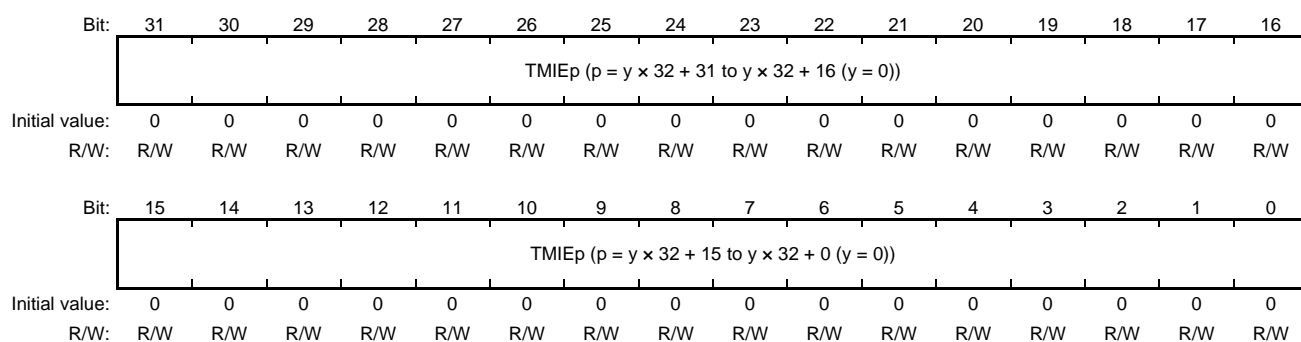
RSCFDnTMIECyLL: <RSCFDn_base> + H'0390 + (H'04 × y),

RSCFDnTMIECyLH: <RSCFDn_base> + H'0391 + (H'04 × y),

RSCFDnTMIECyHL: <RSCFDn_base> + H'0392 + (H'04 × y),

RSCFDnTMIECyHH: <RSCFDn_base> + H'0393 + (H'04 × y)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | TMIEp | H'0000_0000 | R/W | Transmit Buffer Interrupt Enable p (p = y × 32 + 31 to y × 32 + 0) 0: Transmit buffer interrupt is disabled 1: Transmit buffer interrupt is enabled |

TMIEp Bits (p = 0 to 31)

When any of these bits is set to 1 and the corresponding transmission has been completed, a transmit buffer interrupt request is generated.

Modify these bits when the TMTRM flag in the corresponding RSCFDnTMSTSp register is 0 (no transmit request is present).

Write 0 to bits corresponding to transmit buffers linked to transmit/receive FIFO buffers or transmit buffers allocated to the transmit queue.

Table 48.11 shows the bit assignment.

Table 48.11 TMIEp Bit Assignment

| Bit | Channel | Transmit Buffer Number |
|------------|----------------|-------------------------------|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| . | . | . |
| . | . | . |
| 15 | 0 | 15 |
| 16 | 1 | 0 |
| . | . | . |
| . | . | . |
| 30 | 1 | 14 |
| 31 | 1 | 15 |

48.7.10 Details of Transmit Buffer Status Related Registers

48.7.10.1 RSCFDnTMTRSTSy — Transmit Buffer Transmit Request Status Register (y = 0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTMTRSTSy register can be read only in 32-bit units

RSCFDnTMTRSTSyL, RSCFDnTMTRSTSyH registers can be read only in 16-bit units

RSCFDnTMTRSTSyLL, RSCFDnTMTRSTSyLH, RSCFDnTMTRSTSyHL, RSCFDnTMTRSTSyHH registers can be read only in 8-bit units

Address: RSCFDnTMTRSTSy: <RSCFDn_base> + H'0350 + (H'04 × y)

RSCFDnTMTRSTSyL: <RSCFDn_base> + H'0350 + (H'04 × y),

RSCFDnTMTRSTSyH: <RSCFDn_base> + H'0352 + (H'04 × y)

RSCFDnTMTRSTSyLL: <RSCFDn_base> + H'0350 + (H'04 × y),

RSCFDnTMTRSTSyLH: <RSCFDn_base> + H'0351 + (H'04 × y),

RSCFDnTMTRSTSyHL: <RSCFDn_base> + H'0352 + (H'04 × y),

RSCFDnTMTRSTSyHH: <RSCFDn_base> + H'0353 + (H'04 × y)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TMTRSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0)) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TMTRSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0)) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | TMTRSTSp | H'0000_0000 | R | Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 31 to y × 32 + 0) 0: No transmit request is present. 1: A transmit request is present. |

TMTRSTSp Flags (p = 0 to 31)

These flags indicate the status of the TMTR bit in the RSCFDnTMCp register.

When the TMTR bit is set to 1 (transmission is requested), the corresponding TMTRSTSp flag is set to 1.

The corresponding TMTRSTSp flag is cleared to 0 when the TMTR bit is set to 0 (transmission is not requested) or in channel reset mode.

Table 48.12 shows the bit assignment.

Table 48.12 TMTRSTSp Bit Assignment

| Bit | Channel | Transmit Buffer Number |
|------------|----------------|-------------------------------|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| . | . | . |
| . | . | . |
| 15 | 0 | 15 |
| 16 | 1 | 0 |
| . | . | . |
| . | . | . |
| 30 | 1 | 14 |
| 31 | 1 | 15 |

48.7.10.2 RSCFDnTMTARSTSy — Transmit Buffer Transmit Abort Request Status Register (y = 0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTMTARSTSy register can be read only in 32-bit units

RSCFDnTMTARSTSyL, RSCFDnTMTARSTSyH registers can be read only in 16-bit units

RSCFDnTMTARSTSyLL, RSCFDnTMTARSTSyLH, RSCFDnTMTARSTSyHL, RSCFDnTMTARSTSyHH registers can be read only in 8-bit units

Address: RSCFDnTMTARSTSy: <RSCFDn_base> + H'0360 + (H'04 × y)

RSCFDnTMTARSTSyL: <RSCFDn_base> + H'0360 + (H'04 × y),

RSCFDnTMTARSTSyH: <RSCFDn_base> + H'0360 + (H'04 × y)

RSCFDnTMTARSTSyLL: <RSCFDn_base> + H'0360 + (H'04 × y),

RSCFDnTMTARSTSyLH: <RSCFDn_base> + H'0361 + (H'04 × y),

RSCFDnTMTARSTSyHL: <RSCFDn_base> + H'0362 + (H'04 × y),

RSCFDnTMTARSTSyHH: <RSCFDn_base> + H'0363 + (H'04 × y)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|--|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TMTARSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0)) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TMTARSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0)) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | TMTARSTSp | H'0000_0000 | R | Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 31 to y × 32 + 0) 0: No transmit abort request is present. 1: A transmit abort request is present. |

TMTARSTSp Flags (p = 0 to 31)

These flags indicate the status of the TMTAR bit in the RSCFDnTMCp register.

When the TMTAR bit is set to 1 (transmit abort is requested), the corresponding TMTARSTSp flag is set to 1.

The corresponding TMTARSTSp flag is cleared to 0 when the TMTAR bit is set to 0 (transmit abort is not requested) or in channel reset mode.

Table 48.13 shows the bit assignment.

Table 48.13 TMTARSTSp Bit Assignment

| Bit | Channel | Transmit Buffer Number |
|------------|----------------|-------------------------------|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| . | . | . |
| . | . | . |
| 15 | 0 | 15 |
| 16 | 1 | 0 |
| . | . | . |
| . | . | . |
| 30 | 1 | 14 |
| 31 | 1 | 15 |

48.7.10.3 RSCFDnTMCSTSy — Transmit Buffer Transmit Complete Status Register (y = 0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTMCSTSy register can be read only in 32-bit units

RSCFDnTMCSTSyL, RSCFDnTMCSTSyH registers can be read only in 16-bit units

RSCFDnTMCSTSyLL, RSCFDnTMCSTSyLH, RSCFDnTMCSTSyHL, RSCFDnTMCSTSyHH registers can be read only in 8-bit units

Address: RSCFDnTMCSTSy: <RSCFDn_base> + H'0370 + (H'04 × y)

RSCFDnTMCSTSyL: <RSCFDn_base> + H'0370 + (H'04 × y),

RSCFDnTMCSTSyH: <RSCFDn_base> + H'0372 + (H'04 × y)

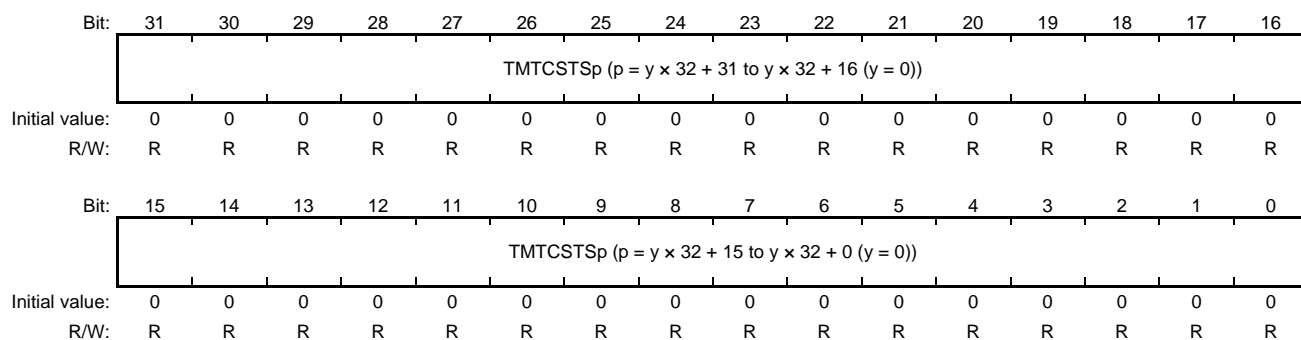
RSCFDnTMCSTSyLL: <RSCFDn_base> + H'0370 + (H'04 × y),

RSCFDnTMCSTSyLH: <RSCFDn_base> + H'0371 + (H'04 × y),

RSCFDnTMCSTSyHL: <RSCFDn_base> + H'0372 + (H'04 × y),

RSCFDnTMCSTSyHH: <RSCFDn_base> + H'0373 + (H'04 × y)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | TMCSTSp | H'0000_0000 | R | Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 31 to y × 32 + 0) 0: Transmission has not been completed. 1: Transmission has been completed. |

TMCSTSp Flags (p = 0 to 31)

When the TMTRF[1:0] flag in the RSCFDnTMSTSp register is set to B'10 (transmission has been completed (without transmit abort request)) or B'11 (transmission has been completed (with transmit abort request)), the corresponding TMCSTSp flag is set to 1.

A TMCSTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to B'00 or in channel reset mode.

Table 48.14 shows the bit assignment.

Table 48.14 TMTcSTSp Bit Assignment

| Bit | Channel | Transmit Buffer Number |
|-----|---------|------------------------|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| · | · | · |
| · | · | · |
| 15 | 0 | 15 |
| 16 | 1 | 0 |
| · | · | · |
| · | · | · |
| 30 | 1 | 14 |
| 31 | 1 | 15 |

48.7.10.4 RSCFDnTMTASTSy — Transmit Buffer Transmit Abort Status Register (y = 0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTMTASTSy register can be read only in 32-bit units

RSCFDnTMTASTSyL, RSCFDnTMTASTSyH registers can be read only in 16-bit units

RSCFDnTMTASTSyLL, RSCFDnTMTASTSyLH, RSCFDnTMTASTSyHL, RSCFDnTMTASTSyHH registers can be read only in 8-bit units

Address: RSCFDnTMTASTSy: <RSCFDn_base> + H'0380 + (H'04 × y)

RSCFDnTMTASTSyL: <RSCFDn_base> + H'0380 + (H'04 × y),

RSCFDnTMTASTSyH: <RSCFDn_base> + H'0382 + (H'04 × y)

RSCFDnTMTASTSyLL: <RSCFDn_base> + H'0380 + (H'04 × y),

RSCFDnTMTASTSyLH: <RSCFDn_base> + H'0381 + (H'04 × y),

RSCFDnTMTASTSyHL: <RSCFDn_base> + H'0382 + (H'04 × y),

RSCFDnTMTASTSyHH: <RSCFDn_base> + H'0383 + (H'04 × y)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TMTASTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0)) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TMTASTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0)) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | TMTASTSp | H'0000_0000 | R | Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 31 to y × 32 + 0) 0: Transmission is not aborted. 1: Transmission is aborted. |

TMTASTSp Flags (p = 0 to 31)

When the TMTRF[1:0] flag in the RSCFDnTMSTSp register is set to B'01 (transmit abort has been completed), the corresponding TMTASTSp flag is set to 1.

A TMTASTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to B'00 or in channel reset mode.

Table 48.15 shows the bit assignment.

Table 48.15 TMTASTSp Bit Assignment

| Bit | Channel | Transmit Buffer Number |
|------------|----------------|-------------------------------|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| · | · | · |
| · | · | · |
| 15 | 0 | 15 |
| 16 | 1 | 0 |
| · | · | · |
| · | · | · |
| 30 | 1 | 14 |
| 31 | 1 | 15 |

48.7.11 Details of Transmit Queue Related Registers

48.7.11.1 RSCFDnTXQCCm — Transmit Queue Configuration and Control Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTXQCCm register can be read/written in 32-bit units

RSCFDnTXQCCmL, RSCFDnTXQCCmH registers can be read/written in 16-bit units

RSCFDnTXQCCmLL, RSCFDnTXQCCmLH, RSCFDnTXQCCmHL, RSCFDnTXQCCmHH registers can be read/written in 8-bit units

Address: RSCFDnTXQCCm: <RSCFDn_base> + H'03A0 + (H'04 × m)

RSCFDnTXQCCmL: <RSCFDn_base> + H'03A0 + (H'04 × m),

RSCFDnTXQCCmH: <RSCFDn_base> + H'03A2 + (H'04 × m)

RSCFDnTXQCCmLL: <RSCFDn_base> + H'03A0 + (H'04 × m),

RSCFDnTXQCCmLH: <RSCFDn_base> + H'03A1 + (H'04 × m),

RSCFDnTXQCCmHL: <RSCFDn_base> + H'03A2 + (H'04 × m),

RSCFDnTXQCCmHH: <RSCFDn_base> + H'03A3 + (H'04 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|-------|-------|------------|-----|-----|-----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | TXQIM | TXQIE | TXQDC[3:0] | | | — | — | — | — | — | — | — | — | TXQE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 14 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 13 | TXQIM | 0 | R/W | Transmit Queue Interrupt Source Select This bit is used to select a transmit queue interrupt source. Modify this bit in channel reset mode. 0: When the transmit queue becomes empty upon completion of message transmission, a transmit queue interrupt request is generated. 1: A transmit queue interrupt request is generated each time a message has been transmitted. |
| 12 | TXQIE | 0 | R/W | Transmit Queue Interrupt Enable When the TXQIE bit is set to 1 and the source selected by the TXQIM bit occurs, an interrupt request is generated. Set the TXQE bit to 0 before modifying the TXQIE bit. 0: Transmit queue interrupt is disabled. 1: Transmit queue interrupt is enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 11 to 8 | TXQDC[3:0] | B'0000 | R/W | <p>Transmit Queue Depth Configuration</p> <p>These bits are used to specify the number of transmit buffers to be allocated to the transmit queues. Transmit buffers are allocated to transmit queues in descending order of buffer number, that is, from $(m \times 16 + 15)$ to $(m \times 16 + 0)$. For examples of how buffer allocation is done, see Figure 48.9. Modify these bits only in channel reset mode.</p> <p>Setting these bits to g ($g = 2$ to 15) makes the $(g + 1)$-buffer transmit queue available.</p> <p>Setting these bits to 0 disables the transmit queue.</p> <p>Setting these bits to 1 is prohibited.</p> |
| 7 to 1 | Reserved | All 0 | R | <p>When read, the value after reset is returned.</p> <p>When writing to these bits, write the value after reset.</p> |
| 0 | TXQE | B'0 | R/W | <p>Transmit Queue Enable</p> <p>Setting this bit to 1 makes the transmit queue available. Modify this bit in channel communication mode or channel halt mode. This bit is cleared to 0 in channel reset mode.</p> <p>Before setting the TXQE bit to 1, set the TXQDC[3:0] bits to B'0010 or more.</p> <p>0: The transmit queue is not used.</p> <p>1: The transmit queue is used.</p> |

48.7.11.2 RSCFDnTXQSTSm — Transmit Queue Status Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTXQSTSm register can be read/written in 32-bit units

RSCFDnTXQSTSmL, RSCFDnTXQSTSmH registers can be read/written in 16-bit units

RSCFDnTXQSTSmLL, RSCFDnTXQSTSmLH, RSCFDnTXQSTSmHL, RSCFDnTXQSTSmHH registers can be read/written in 8-bit units

Address: RSCFDnTXQSTSm: <RSCFDn_base> + H'03C0 + (H'04 × m)

RSCFDnTXQSTSmL: <RSCFDn_base> + H'03C0 + (H'04 × m),

RSCFDnTXQSTSmH: <RSCFDn_base> + H'03C2 + (H'04 × m)

RSCFDnTXQSTSmLL: <RSCFDn_base> + H'03C0 + (H'04 × m),

RSCFDnTXQSTSmLH: <RSCFDn_base> + H'03C1 + (H'04 × m),

RSCFDnTXQSTSmHL: <RSCFDn_base> + H'03C2 + (H'04 × m),

RSCFDnTXQSTSmHH: <RSCFDn_base> + H'03C3 + (H'04 × m)

Value after reset: H'0000_0001

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | TXQIF | TXQFL L | TXQEM P |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W* | R | R |

Note: * The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|------|---|
| 31 to 13 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 12 to 8 | Reserved | All 0 | R | When read, an undefined value is returned. When writing to these bits, write the value after reset. |
| 7 to 3 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 2 | TXQIF | B'0 | R/W* | <p>Transmit Queue Interrupt Request Flag</p> <p>The TXQIF flag is set to 1 when the event specified by the TXQIM bit in the RSCFDnTXQCCm register has occurred.</p> <p>The TXQIF flag is cleared to 0 in channel reset mode or by writing 0 to this flag. This flag is not cleared to 0 by setting the TXQE bit in the RSCFDnTXQCCm register to 0 (the transmit queue is not used).</p> <p>0: No transmit queue interrupt request is present. 1: A transmit queue interrupt request is present.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | TXQFLL | B'0 | R | <p>Transmit Queue Full Status Flag</p> <p>The TXQFLL flag is set to 1 when the number of messages set for the transmit queue matches the transmit queue depth set by the TXQDC[3:0] bits in the RSCFDnTXQCCm register.</p> <p>This flag is cleared to 0 in any of the following cases.</p> <ul style="list-style-type: none"> • The number of messages set for the transmit queue is smaller than the transmit queue depth set by the TXQDC[3:0] bits. • In channel reset mode <p>0: The transmit queue is not full. 1: The transmit queue is full.</p> |
| 0 | TXQEMP | B'1 | R | <p>Transmit Queue Empty Status Flag</p> <p>The TXQEMP flag is cleared to 0 when even a single message is set for the transmit queue.</p> <p>This flag is set to 1 in any of the following cases.</p> <ul style="list-style-type: none"> • The TXQE bit is set to 0 (the transmit queue is not used). • The transmit queue becomes empty. • In channel reset mode <p>0: The transmit queue contains messages. 1: The transmit queue contains no message (transmit queue empty).</p> |

48.7.11.3 RSCFDnTXQPCTRm — Transmit Queue Pointer Control Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTXQPCTRm registers can only be written in 32-bit units

RSCFDnTXQPCTRmL, RSCFDnTXQPCTRmH registers can only be written in 16-bit units

RSCFDnTXQPCTRmLL, RSCFDnTXQPCTRmLH, RSCFDnTXQPCTRmHL, RSCFDnTXQPCTRmHH registers can only be written in 8-bit units

Address: RSCFDnTXQPCTRm: <RSCFDn_base> + H'03E0 + (H'04 × m)

RSCFDnTXQPCTRmL: <RSCFDn_base> + H'03E0 + (H'04 × m),

RSCFDnTXQPCTRmH: <RSCFDn_base> + H'03E2 + (H'04 × m)

RSCFDnTXQPCTRmLL: <RSCFDn_base> + H'03E0 + (H'04 × m),

RSCFDnTXQPCTRmLH: <RSCFDn_base> + H'03E1 + (H'04 × m),

RSCFDnTXQPCTRmHL: <RSCFDn_base> + H'03E2 + (H'04 × m),

RSCFDnTXQPCTRmHH: <RSCFDn_base> + H'03RZ/G2E + (H'04 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------------|----|----|----|----|----|----|----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | TXQPC[7:0] | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | W | W | W | W | W | W | W | W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 8 | Reserved | All 0 | R | The write value should be the value after reset. |
| 7 to 0 | TXQPC[7:0] | H'00 | W | <p>Transmit Queue Pointer Control</p> <p>Writing H'FF to the TXQPC[7:0] bits moves the write pointer to the next transmit queue buffer and generates a transmit request of the message. Write transmit messages to the RSCFDnTMIDp, RSCFDnTMPTRp, RSCFDnTMDf0_p, and RSCFDnTMDf1_p registers (p = 15, 31) before writing H'FF to the TXQPC[7:0] bits.</p> <p>When writing H'FF to these bits, make sure that the TXQE bit in the RSCFDnTXQCCm register is set to 1 (the transmit queue is used) and the TXQFLL flag in the RSCFDnTXQSTSm register is 0 (the transmit queue is not full).</p> |

48.7.12 Details of Transmit History Related Registers

48.7.12.1 RSCFDnTHLCCm — Transmit History Configuration and Control Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTHLCCm register can be read/written in 32-bit units
 RSCFDnTHLCCmL, RSCFDnTHLCCmH registers can be read/written in 16-bit units
 RSCFDnTHLCCmLL, RSCFDnTHLCCmLH, RSCFDnTHLCCmHL, RSCFDnTHLCCmHH registers can be read/written in 8-bit units

Address: RSCFDnTHLCCm: <RSCFDn_base> + H'0400 + (H'04 × m)
 RSCFDnTHLCCmL: <RSCFDn_base> + H'0400 + (H'04 × m),
 RSCFDnTHLCCmH: <RSCFDn_base> + H'0402 + (H'04 × m)
 RSCFDnTHLCCmLL: <RSCFDn_base> + H'0400 + (H'04 × m),
 RSCFDnTHLCCmLH: <RSCFDn_base> + H'0401 + (H'04 × m),
 RSCFDnTHLCCmHL: <RSCFDn_base> + H'0402 + (H'04 × m),
 RSCFDnTHLCCmHH: <RSCFDn_base> + H'0403 + (H'04 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|------------|-------|-------|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | THLDT E | THLIM | THLIE | — | — | — | — | — | — | — | THLE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 11 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 10 | THLDTE | B'0 | R/W | Transmit History Target Buffer Select When this bit is set to 0, the transmit history data of messages transmitted from transmit/receive FIFO buffers and the transmit queue is stored in the transmit history buffer. When this bit is set to 1, the transmit history data of messages transmitted from transmit buffers, transmit/receive FIFO buffers, and the transmit queue is stored in the transmit history buffer. Modify this bit only in channel reset mode. 0: Entry from transmit/receive FIFO buffers and transmit queue 1: Entry from transmit buffers, transmit/receive FIFO buffers, and transmit queue |
| 9 | THLIM | B'0 | R/W | Transmit History Interrupt Source Select This bit is used to select a transmit history interrupt source. Modify this bit only in channel reset mode. 0: When 12 sets of data have been stored in the transmit history buffer 1: When a single set of transmit history data has been stored |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 8 | THLIE | B'0 | R/W | <p>Transmit History Interrupt Enable</p> <p>When the THLIE bit is set to 1 and the source selected by the THLIM bit has occurred, a transmit history interrupt request is generated. Modify the THLIE bit only when the THLE bit set to 0.</p> <p>0: Transmit history interrupt is disabled. 1: Transmit history interrupt is enabled.</p> |
| 7 to 1 | Reserved | All 0 | R | <p>When read, the value after reset is returned.</p> <p>When writing to these bits, write the value after reset.</p> |
| 0 | THLE | B'0 | R/W | <p>Transmit History Buffer Enable</p> <p>Setting this bit to 1 makes the transmit history buffer available. When data transmission from the buffer selected by the THLDTE bit has been completed, the transmit history data of transmit messages is stored in the transmit history buffer.</p> <p>Modify this bit in channel communication mode or channel halt mode.</p> <p>0: Transmit history buffer is not used. 1: Transmit history buffer is used.</p> |

48.7.12.2 RSCFDnTHLSTSm — Transmit History Status Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTHLSTSm register can be read/written in 32-bit units

RSCFDnTHLSTSmL, RSCFDnTHLSTSmH register can be read/written in 16-bit units

RSCFDnTHLSTSmLL, RSCFDnTHLSTSmLH, RSCFDnTHLSTSmHL, RSCFDnTHLSTSmHH registers can be read/written in 8-bit units

Address: RSCFDnTHLSTSm: <RSCFDn_base> + H'0420 + (H'04 × m)

RSCFDnTHLSTSmL: <RSCFDn_base> + H'0420 + (H'04 × m),

RSCFDnTHLSTSmH: <RSCFDn_base> + H'0422 + (H'04 × m)

RSCFDnTHLSTSmLL: <RSCFDn_base> + H'0420 + (H'04 × m),

RSCFDnTHLSTSmLH: <RSCFDn_base> + H'0421 + (H'04 × m),

RSCFDnTHLSTSmHL: <RSCFDn_base> + H'0422 + (H'04 × m),

RSCFDnTHLSTSmHH: <RSCFDn_base> + H'0423 + (H'04 × m)

Value after reset: H'0000_0001

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|------------|----|----|----|----|----|----|----|----|-------|------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | THLMC[4:0] | | | | — | — | — | — | — | THLIF | THLEL T | THLFL L | THLEM P |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W* | R/W* | R | R |

Note: * The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|------|--|
| 31 to 13 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 12 to 8 | THLMC[4:0] | B'0_0000 | R | Transmit History Buffer Unread Data Counter These bits indicate the number of unread data sets stored in the transmit history buffer. |
| 7 to 4 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 3 | THLIF | B'0 | R/W* | Transmit History Interrupt Request Flag The THLIF flag is set to 1 when the interrupt source specified with the THLIM bit in the RSCFDnTHLCCm register occurs. This flag is cleared to 0 in channel reset mode or by the program writing 0 to this flag. To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1". 0: No transmit history interrupt request is present. 1: A transmit history interrupt request is present. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 2 | THLELT | B'0 | R/W* | <p>Transmit History Buffer Overflow Flag</p> <p>The THLELT flag is set to 1 when an attempt is made to store new transmit history data while the transmit history buffer is full. In this case, the new data is discarded. This flag becomes 0 in channel reset mode or by the program writing 0 to this flag.</p> <p>To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.</p> <p>When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1".</p> <p>0: Transmit history buffer overflow has not occurred. 1: Transmit history buffer overflow has occurred.</p> |
| 1 | THLFLL | B'0 | R | <p>Transmit History Buffer Full Status Flag</p> <p>The THLFLL flag is set to 1 when 16 data sets have been stored in the transmit history buffer, and is cleared to 0 when the number of data sets stored in the transmit history buffer has decreased to less than 16. This bit is also cleared to 0 in channel reset mode or when the THLE bit in the RSCFDnTHLCCm register is set to 0 (transmit history buffer is not used).</p> <p>0: Transmit history buffer is not full. 1: Transmit history buffer is full.</p> |
| 0 | THLEMP | B'1 | R | <p>Transmit History Buffer Empty Status Flag</p> <p>The THLEMP flag is cleared to 0 when even a single set of transmit history data has been stored in the transmit history buffer. This flag is set to 1 when all the data in the transmit history buffer has been read. This flag is also set to 1 in channel reset mode or when the THLE bit in the RSCFDnTHLCCm register is set to 0 (transmit history buffer is not used).</p> <p>0: Transmit history buffer contains unread data. 1: Transmit history buffer contains no unread data (buffer empty).</p> |

Note: To clear THLIF or THLELT flag to 0, the program must write 0. When writing, use a store instruction to write "0" to the given flag and "1" to other flags.

48.7.12.3 RSCFDnTHLPCTRm — Transmit History Pointer Control Register (m = 0 or 1)

| | | | |
|--------|--------------------------|------------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | syuuRZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTHLPCTRm register can only be written in 32-bit units

RSCFDnTHLPCTRmL, RSCFDnTHLPCTRmH registers can only be written in 16-bit units

RSCFDnTHLPCTRmLL, RSCFDnTHLPCTRmLH, RSCFDnTHLPCTRmHL, RSCFDnTHLPCTRmHH registers can only be written in 8-bit units

Address: RSCFDnTHLPCTRm: <RSCFDn_base> + H'0440 + (H'04 × m)

RSCFDnTHLPCTRmL: <RSCFDn_base> + H'0440 + (H'04 × m),

RSCFDnTHLPCTRmH: <RSCFDn_base> + H'0442 + (H'04 × m)

RSCFDnTHLPCTRmLL: <RSCFDn_base> + H'0440 + (H'04 × m),

RSCFDnTHLPCTRmLH: <RSCFDn_base> + H'0441 + (H'04 × m),

RSCFDnTHLPCTRmHL: <RSCFDn_base> + H'0442 + (H'04 × m),

RSCFDnTHLPCTRmHH: <RSCFDn_base> + H'0443 + (H'04 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------------|----|----|----|----|----|----|----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | THLPC[7:0] | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | W | W | W | W | W | W | W | W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 8 | Reserved | All 0 | R | When writing to these bits, write the value after reset. |
| 7 to 0 | THLPC[7:0] | H'00 | W | <p>Transmit History List Pointer Control</p> <p>When the THLPC[7:0] bits are set to H'FF, the read pointer moves to the next data in the transmit history buffer. At this time, the THLMC[4:0] (transmit history buffer unread data counter) value in the RSCFDnTHLSTSm register is decremented. Write H'FF to the THLPC[7:0] bits after reading from the RSCFDnTHLACCm register.</p> <p>When writing H'FF to these bits, make sure that the THLE bit in the RSCFDnTHLCCm register is set to 1 (transmit history buffer is used) and the THLEMP flag in the RSCFDnTHLSTSm register is 0.</p> |

48.7.12.4 RSCFDnTHLACCm — Transmit History Access Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnTHLACCm register can be read only in 32-bit units

RSCFDnTHLACCmL, RSCFDnTHLACCmH registers can be read only in 16-bit units

RSCFDnTHLACCmLL, RSCFDnTHLACCmLH, RSCFDnTHLACCmHL, RSCFDnTHLACCmHH registers can be read only in 8-bit units

Address: RSCFDnTHLACCm: <RSCFDn_base> + H'1800 + (H'04 × m)

RSCFDnTHLACCmL: <RSCFDn_base> + H'1800 + (H'04 × m),

RSCFDnTHLACCmH: <RSCFDn_base> + H'1802 + (H'04 × m)

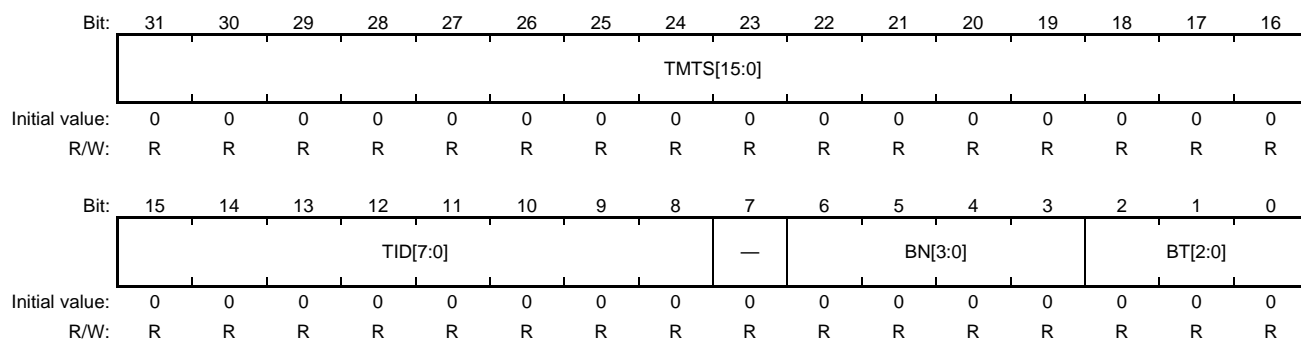
RSCFDnTHLACCmLL: <RSCFDn_base> + H'1800 + (H'04 × m),

RSCFDnTHLACCmLH: <RSCFDn_base> + H'1801 + (H'04 × m),

RSCFDnTHLACCmHL: <RSCFDn_base> + H'1802 + (H'04 × m),

RSCFDnTHLACCmHH: <RSCFDn_base> + H'1803 + (H'04 × m)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 16 | TMTS[15:0] | H'0000 | R | Timestamp Data These bits indicate the timestamp value of the transmit history data stored in the transmit history buffer when the TMTSCE bit in the RSCFDnGCFG register is 1. These bits are always read as 0 when the TMTSCE bit is 0. |
| 15 to 8 | TID[7:0] | H'00 | R | Label Data These bits indicate the label information of transmit history data stored in the transmit history buffer. |
| 7 | Reserved | B'0 | R | When read, the value after reset is returned. |
| 6 to 3 | BN[3:0] | B'0000 | R | Buffer Number Data These bits indicate the transmit source buffer number in the transmit history data stored in the transmit history buffer. |
| 2 to 0 | BT[2:0] | B'000 | R | Buffer Type Data These bits indicate the type of the transmit source buffer in the transmit history data stored in the transmit history buffer. B'001: Transmit buffer B'010: Transmit/receive FIFO buffer B'100: Transmit queue |

48.7.13 Details of Test Related Registers

48.7.13.1 RSCFDnGTSTCFG — Global Test Configuration Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnGTSTCFG register can be read/written in 32-bit units

RSCFDnGTSTCFGL, RSCFDnGTSTCFGH registers can be read/written in 16-bit units

RSCFDnGTSTCFGLL, RSCFDnGTSTCFGLH, RSCFDnGTSTCFGHL, RSCFDnGTSTCFGHH registers can be read/written in 8-bit units

Address: RSCFDnGTSTCFG: <RSCFDn_base> + H'0468

RSCFDnGTSTCFGL: <RSCFDn_base> + H'0468, RSCFDnGTSTCFGH: <RSCFDn_base> + H'046A

RSCFDnGTSTCFGLL: <RSCFDn_base> + H'0468, RSCFDnGTSTCFGLH: <RSCFDn_base> + H'0469,

RSCFDnGTSTCFGHL: <RSCFDn_base> + H'046A, RSCFDnGTSTCFGHH: <RSCFDn_base> + H'046B

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|------------|-----|-----|-----|-----|-------------|-------------|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | RTMPS[6:0] | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | C1ICB CE | C0ICB CE | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 23 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 22 to 16 | RTMPS[6:0] | H'00 | R/W | RAM Test Page Configuration These bits are used to set the RAM test target page number for RAM test. Set a value in the range of H'00 to H'38, inclusive. |
| 15 to 2 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 1 | C1ICBCE | B'0 | R/W | CAN1 Inter-Channel Communication Test Enable Setting this bit to 1 enables the channel 1 inter-channel communication test. 0: CAN1 inter-channel communication test is disabled. 1: CAN1 inter-channel communication test is enabled. |
| 0 | C0ICBCE | B'0 | R/W | CAN0 Inter-Channel Communication Test Enable Setting this bit to 1 enables the channel 0 inter-channel communication test. 0: CAN0 inter-channel communication test is disabled. 1: CAN0 inter-channel communication test is enabled. |

Modify the RSCFDnGTSTCFG register only in global test mode.

48.7.13.2 RSCFDnGTSTCTR — Global Test Control Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnGTSTCTR register can be read/written in 32-bit units

RSCFDnGTSTCTRL, RSCFDnGTSTCTRH registers can be read/written in 16-bit units

RSCFDnGTSTCTRLL, RSCFDnGTSTCTRLH, RSCFDnGTSTCTRHL, RSCFDnGTSTCTRHH registers can be read/written in 8-bit units

Address: RSCFDnGTSTCTR: <RSCFDn_base> + H'046C

RSCFDnGTSTCTRL: <RSCFDn_base> + H'046C, RSCFDnGTSTCTRH: <RSCFDn_base> + H'046E

RSCFDnGTSTCTRLL: <RSCFDn_base> + H'046C, RSCFDnGTSTCTRLH: <RSCFDn_base> + H'046D,

RSCFDnGTSTCTRHL: <RSCFDn_base> + H'046E, RSCFDnGTSTCTRHH: <RSCFDn_base> + H'046F

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|------|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | RTME | — | ICBCTME |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 3 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 2 | RTME | B'0 | R/W | RAM Test Enable Setting this bit to 1 enables the RAM test. Modify this bit only in global test mode. 1. Set the GMDC[1:0] bits in the RSCFDnGCTR register to B'10 (Global test mode). 2. Set the RTME bit to 1. 3. Check that the RTME bit is set to 1. 0: RAM test is disabled. 1: RAM test is enabled. |
| 1 | Reserved | B'0 | R | When read, the value after reset is returned. When writing to this bit, write the value after reset. |
| 0 | ICBCTME | B'0 | R/W | Communication Test between Channels Enable When this bit is set to 1, a communication test is enabled between the channels for which the CmICBCE bit (m = 0 or 1) in the RSCFDnGTSTCFG register has been set to 1. Modify the ICBCTME bit only in global test mode. 0: Communication test between channels disabled 1: Communication test between channels enabled |

48.7.13.3 RSCFDnGLOCKK — Global Lock Key Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnGLOCKK register can be written only in 32-bit units

RSCFDnGLOCKKL, RSCFDnGLOCKKH registers can be written only in 16-bit units

Address: RSCFDnGLOCKK: <RSCFDn_base> + H'047C

RSCFDnGLOCKKL: <RSCFDn_base> + H'047C, RSCFDnGLOCKKH: <RSCFDn_base> + H'047E

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | LOCK[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | W* | W* | W* | W* | W* | W* | W* | W* | W* | W* | W* | W* | W* | W* | W* | W* |

Note: * Writing to these bits is effective only when the RS-CANFD module is in global test mode.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 16 | Reserved | All 0 | R | When writing to these bits, write the value after reset. |
| 15 to 0 | LOCK[15:0] | H'0000 | W* | <p>Lock Key</p> <p>Writing the protection release data to the LOCK[15:0] bits in succession enables writing 1 to the RTME bit in the RSCFDnGTSTCTR register.</p> <p>After the protection has been released, writing to the I/O register area (<RSCFDn_base> + H'0000 to <RSCFDn_base> + H'04FF) of the CAN (except the RAM) enables the protection again.</p> <p>Reading from the I/O register area of the CAN or reading from/writing to other areas does not enable the protection.</p> |

The RSCFDnGLOCKK register releases protection of special test bits and is write only.

For the protection release data, see section 48.16.4.2, Procedure for Releasing the Protection.

48.7.13.4 RSCFDnRPGACCr — RAM Test Page Access Register (r = 0 to 63)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

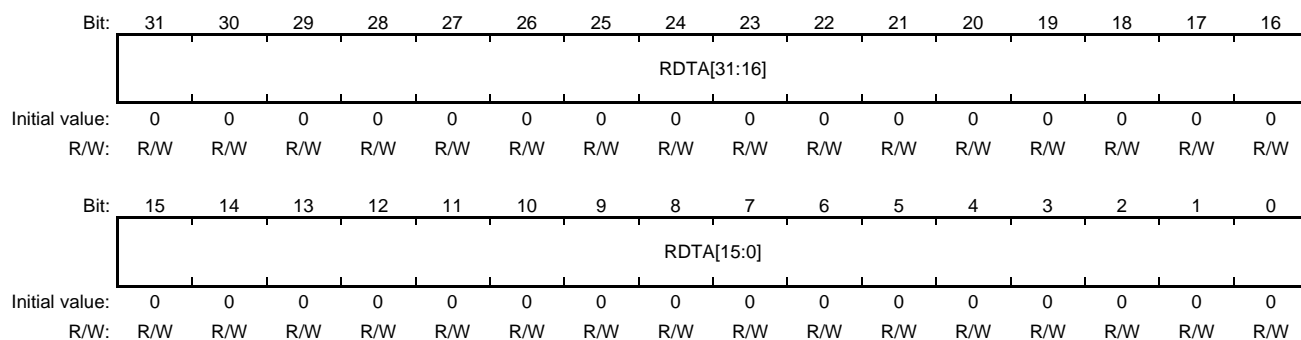
Access: RSCFDnRPGACCr register can be read/written in 32-bit units

RSCFDnRPGACCrL, RSCFDnRPGACCrH registers can be read/written in 16-bit units

RSCFDnRPGACCrLL, RSCFDnRPGACCrLH, RSCFDnRPGACCrHL, RSCFDnRPGACCrHH registers can be read/written in 8-bit units

Address: RSCFDnRPGACCr: <RSCFDn_base> + H'1900 + (H'04 × r)
 RSCFDnRPGACCrL: <RSCFDn_base> + H'1900 + (H'04 × r),
 RSCFDnRPGACCrH: <RSCFDn_base> + H'1902 + (H'04 × r)
 RSCFDnRPGACCrLL: <RSCFDn_base> + H'1900 + (H'04 × r),
 RSCFDnRPGACCrLH: <RSCFDn_base> + H'1901 + (H'04 × r),
 RSCFDnRPGACCrHL: <RSCFDn_base> + H'1902 + (H'04 × r),
 RSCFDnRPGACCrHH: <RSCFDn_base> + H'1903 + (H'04 × r)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 0 | RDTA[31:0] | H'0000_0000 | R/W | RAM Data Test Access Data can be read and written in RSCAN RAM. These bits are read as 0. The write value should be 0. |

Modify the RSCFDnRPGACCr register in global test mode with the RTME bit in the RSCFDnGTSTCTR register set to 1 (RAM test is enabled).

The RSCFDnRPGACCr register is readable and writable when the RTME bit is set to 1.

48.8 Registers Configuration (CAN FD mode)

This section describes all RS-CANFD registers used in CAN FD mode.

48.8.1 List of Registers

Table 48.16 lists the RS-CANFD registers used in CAN FD mode.

For details about <RSCFDn_base>, see section 48.5.1, Register Base Address.

Table 48.16 Registers

| Channel | Name | Abbreviation | R/W | Offset Address | Size | Second Generation RZ/G Series Products | | | |
|----------------------|--|------------------------|-----|---------------------|------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Index of Table 48.4. | Global interface mode select register | RSCFDnCFDGRMCFG | R/W | H'04FC | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Channel m normal bit rate configuration register | RSCFDnCFDCmNCFG | R/W | H'0000 + (H'10 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Channel m control register | RSCFDnCFDCmCTR | R/W | H'0004 + (H'10 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Channel m status register | RSCFDnCFDCmSTS | R/W | H'0008 + (H'10 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Channel m error flag register | RSCFDnCFDCmERFL | R/W | H'000C + (H'10 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Channel m data bit rate configuration register | RSCFDnCFDCmDCFG | R/W | H'0500 + (H'20 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Channel m CAN FD configuration register | RSCFDnCFDCmFDCF G | R/W | H'0504 + (H'20 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Channel m CAN FD control register | RSCFDnCFDCmFDCT R | R/W | H'0508 + (H'20 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Channel m CAN FD status register | RSCFDnCFDCmFDSTS | R/W | H'050C + (H'20 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Channel m CAN FD CRC register | RSCFDnCFDCmFDCR C | R | H'0510 + (H'20 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global configuration register | RSCFDnCFDGCFCG | R/W | H'0084 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global control register | RSCFDnCFDGCTR | R/W | H'0088 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global status register | RSCFDnCFDGSTS | R | H'008C | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global error flag register | RSCFDnCFDGERFL | R/W | H'0090 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global timestamp counter register | RSCFDnCFDGTSC | R | H'0094 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global TX interrupt status register 0 | RSCFDnCFDGTINTST S0 | R | H'0460 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global TX interrupt status register 1 | RSCFDnCFDGTINTST S1 | R | H'0464 | 32 | √ | √ | √ | √ |

| | | | | | | Second Generation RZ/G Series Products | | | |
|----------------------|--|------------------------|-----|--|------|---|----------------------------|--------|--------|
| Channel | Name | Abbreviation | R/W | Offset Address | Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Index of Table 48.4. | Receive rule entry control register | RSCFDnCFDGAFLLECT R | R/W | H'0098 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive rule configuration register 0 | RSCFDnCFDGAFLCFG 0 | R/W | H'009C | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive rule configuration register 1 | RSCFDnCFDGAFLCFG 1 | R | H'00A0 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive rule ID register j | RSCFDnCFDGAFLIDj | R/W | H'1000 + (H'10 × j) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive rule mask register j | RSCFDnCFDGAFLMj | R/W | H'1004 + (H'10 × j) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive rule pointer 0 register j | RSCFDnCFDGAFLP0_j | R/W | H'1008 + (H'10 × j) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive rule pointer 1 register j | RSCFDnCFDGAFLP1_j | R/W | H'100C + (H'10 × j) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive buffer number register | RSCFDnCFDRMNB | R/W | H'00A4 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive buffer new data register y | RSCFDnCFDRMNDy | R/W | H'00A8 + (H'04 × y) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive buffer ID register q | RSCFDnCFDRMIDq | R | H'2000 + (H'20 × q) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive buffer pointer register q | RSCFDnCFDRMPTRq | R | H'2004 + (H'20 × q) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive buffer CAN FD status register q | RSCFDnCFDRMFDSTS q | R | H'2008 + (H'20 × q) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive buffer data field b register q | RSCFDnCFDRMDfb_q | R | H'200C + (H'04 × b) + (H'20 × q) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer configuration and control register x | RSCFDnCFDRFCCx | R/W | H'00B8 + (H'04 × x) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer status register x | RSCFDnCFDRFSTSx | R/W | H'00D8 + (H'04 × x) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer pointer control register x | RSCFDnCFDRFPCTRx | R/W | H'00F8 + (H'04 × x) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer access ID register x | RSCFDnCFDRFIDx | R | H'3000 + (H'80 × x) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer access pointer register x | RSCFDnCFDRFPTRx | R | H'3004 + (H'80 × x) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer CAN FD status register x | RSCFDnCFDRFFDSTS x | R | H'3008 + (H'80 × x) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer access data field d register x | RSCFDnCFDRFDfd_x | R | H'300C + (H'04 × d) + (H'80 × x) | 32 | √ | √ | √ | √ |

| | | | | | | Second Generation RZ/G Series Products | | | |
|-------------------------|--|------------------------|-----|--|------|---|----------------------------|--------|--------|
| Channel | Name | Abbreviation | R/W | Offset Address | Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Index of Table 48.4. | Transmit/receive FIFO buffer configuration and control register k | RSCFDnCFDCFCCK | R/W | H'0118 + (H'04 × k) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer status register k | RSCFDnCFDCFSTSk | R/W | H'0178 + (H'04 × k) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer pointer control register k | RSCFDnCFDCFPCTRk | R/W | H'01D8 + (H'04 × k) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer access ID register k | RSCFDnCFDCFIDk | R/W | H'3400 + (H'80 × k) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer access pointer register k | RSCFDnCFDCFPTRk | R/W | H'3404 + (H'80 × k) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO CAN FD configuration and status register k | RSCFDnCFDCFFDCST Sk | R/W | H'3408 + (H'80 × k) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer access data field d register k | RSCFDnCFDCFDf_d_k | R/W | H'340C + (H'04 × d) + (H'80 × k) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | FIFO empty status register | RSCFDnCFDFESTS | R | H'0238 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | FIFO full status register | RSCFDnCFDFFSTS | R | H'023C | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | FIFO Msg lost status register | RSCFDnCFDFMSTS | R | H'0240 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Receive FIFO buffer interrupt flag status register | RSCFDnCFDRFISTS | R | H'0244 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer receive interrupt flag status register | RSCFDnCFDCFRISTS | R | H'0248 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit/receive FIFO buffer transmit interrupt flag status register | RSCFDnCFDCFTISTS | R | H'024C | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | DMA enable register | RSCFDnCFDCDTCT | R/W | H'0490 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | DMA status register | RSCFDnCFDCDTSTS | R | H'0494 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer control register p | RSCFDnCFDTMCp | R/W | H'0250 + (H'01 × p) | 8 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer status register p | RSCFDnCFDTMSTSp | R/W | H'02D0 + (H'01 × p) | 8 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer ID register p | RSCFDnCFDTMIDp | R/W | H'4000 + (H'20 × p) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer pointer register p | RSCFDnCFDTMPTRp | R/W | H'4004 + (H'20 × p) | 32 | √ | √ | √ | √ |

| Channel | Name | Abbreviation | R/W | Offset Address | Size | Second Generation RZ/G Series Products | | | |
|----------------------|---|---------------------|-----|----------------------------------|------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Index of Table 48.4. | Transmit buffer CAN FD configuration register p | RSCFDnCFDTMFDCTR p | R/W | H'4008 + (H'20 × p) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer data field b register p | RSCFDnCFDTMDFb_p | R/W | H'400C + (H'04 × b) + (H'20 × p) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer interrupt enable configuration register y | RSCFDnCFDTMIECy | R/W | H'0390 + (H'04 × y) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer transmit request status register y | RSCFDnCFDTMTRSTS y | R | H'0350 + (H'04 × y) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer transmit abort request status register y | RSCFDnCFDTMTARST Sy | R | H'0360 + (H'04 × y) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer transmit complete status register y | RSCFDnCFDTMTCSTS y | R | H'0370 + (H'04 × y) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit buffer transmit abort status register y | RSCFDnCFDTMTASTS y | R | H'0380 + (H'04 × y) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit queue configuration and control register m | RSCFDnCFDTXQCCm | R/W | H'03A0 + (H'04 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit queue status register m | RSCFDnCFDTXQSTSm | R/W | H'03C0 + (H'04 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit queue pointer control register m | RSCFDnCFDTXQPCTR m | R/W | H'03E0 + (H'04 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit history configuration and control register m | RSCFDnCFDTHLCCm | R/W | H'0400 + (H'04 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit history status register m | RSCFDnCFDTHLSTSm | R/W | H'0420 + (H'04 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit history pointer control register m | RSCFDnCFDTHLPCTR m | R/W | H'0440 + (H'04 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Transmit history access register m | RSCFDnCFDTHLACCm | R | H'6000 + (H'04 × m) | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global test configuration register | RSCFDnCFDGTSTCFG | R/W | H'0468 | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global test control register | RSCFDnCFDGTSTCTR | R/W | H'046C | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | Global lock key register | RSCFDnCFDGLCKK | R/W | H'047C | 32 | √ | √ | √ | √ |
| Index of Table 48.4. | RAM test page access register r | RSCFDnCFDRPGACCr | R/W | H'6400 + (H'04 × r) | 32 | √ | √ | √ | √ |

Table 48.17 Transmit Buffer p Allocated to Each Channel

| | CANm |
|------------------------------------|------------------------------------|
| Transmit buffer p | Transmit buffer $16 \times m + 0$ |
| | Transmit buffer $16 \times m + 1$ |
| | Transmit buffer $16 \times m + 2$ |
| | Transmit buffer $16 \times m + 3$ |
| | Transmit buffer $16 \times m + 4$ |
| | Transmit buffer $16 \times m + 5$ |
| | Transmit buffer $16 \times m + 6$ |
| | Transmit buffer $16 \times m + 7$ |
| | Transmit buffer $16 \times m + 8$ |
| | Transmit buffer $16 \times m + 9$ |
| | Transmit buffer $16 \times m + 10$ |
| | Transmit buffer $16 \times m + 11$ |
| | Transmit buffer $16 \times m + 12$ |
| | Transmit buffer $16 \times m + 13$ |
| | Transmit buffer $16 \times m + 14$ |
| Transmit buffer $16 \times m + 15$ | |

Table 48.18 Transmit/Receive FIFO Buffer k Allocated to Each Channel

| | CANm |
|--------------------------------|---|
| Transmit/receive FIFO buffer k | Transmit/receive FIFO buffer $3 \times m + 0$ |
| | Transmit/receive FIFO buffer $3 \times m + 1$ |
| | Transmit/receive FIFO buffer $3 \times m + 2$ |

Table 48.19 Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer by the Setting of Bits CFTML[3:0]

| Setting of Bits CFTML[3:0] | Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer |
|----------------------------|--|
| B'0000 | Transmit buffer $16 \times m + 0$ |
| B'0001 | Transmit buffer $16 \times m + 1$ |
| B'0010 | Transmit buffer $16 \times m + 2$ |
| B'0011 | Transmit buffer $16 \times m + 3$ |
| B'0100 | Transmit buffer $16 \times m + 4$ |
| B'0101 | Transmit buffer $16 \times m + 5$ |
| B'0110 | Transmit buffer $16 \times m + 6$ |
| B'0111 | Transmit buffer $16 \times m + 7$ |
| B'1000 | Transmit buffer $16 \times m + 8$ |
| B'1001 | Transmit buffer $16 \times m + 9$ |
| B'1010 | Transmit buffer $16 \times m + 10$ |
| B'1011 | Transmit buffer $16 \times m + 11$ |
| B'1100 | Transmit buffer $16 \times m + 12$ |
| B'1101 | Transmit buffer $16 \times m + 13$ |
| B'1110 | Transmit buffer $16 \times m + 14$ |
| B'1111 | Transmit buffer $16 \times m + 15$ |

Table 48.20 Transmit Buffer p Allocated to the Transmit Queue of Each Channel

| Setting of Bits TXQDC[3:0] | Transmit Buffer p Allocated to the Transmit Queue |
|----------------------------|--|
| B'0000 | Setting prohibited |
| B'0001 | Setting prohibited |
| B'0010 | Transmit buffer $16 \times m + 15$ to $16 \times m + 13$ |
| B'0011 | Transmit buffer $16 \times m + 15$ to $16 \times m + 12$ |
| B'0100 | Transmit buffer $16 \times m + 15$ to $16 \times m + 11$ |
| B'0101 | Transmit buffer $16 \times m + 15$ to $16 \times m + 10$ |
| B'0110 | Transmit buffer $16 \times m + 15$ to $16 \times m + 9$ |
| B'0111 | Transmit buffer $16 \times m + 15$ to $16 \times m + 8$ |
| B'1000 | Transmit buffer $16 \times m + 15$ to $16 \times m + 7$ |
| B'1001 | Transmit buffer $16 \times m + 15$ to $16 \times m + 6$ |
| B'1010 | Transmit buffer $16 \times m + 15$ to $16 \times m + 5$ |
| B'1011 | Transmit buffer $16 \times m + 15$ to $16 \times m + 4$ |
| B'1100 | Transmit buffer $16 \times m + 15$ to $16 \times m + 3$ |
| B'1101 | Transmit buffer $16 \times m + 15$ to $16 \times m + 2$ |
| B'1110 | Transmit buffer $16 \times m + 15$ to $16 \times m + 1$ |
| B'1111 | Transmit buffer $16 \times m + 15$ to $16 \times m + 0$ |

48.9 Register Description (CAN FD mode)

48.9.1 Details of Interface Mode Related Registers

48.9.1.1 RSCFDnCFDGRMCFG — Global Interface Mode Select Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGRMCFG register can be read/written in 32-bit units

RSCFDnCFDGRMCFG, RSCFDnCFDGRMCFGH registers can be read/written in 16-bit units

RSCFDnCFDGRMCFG, RSCFDnCFDGRMCFGH, RSCFDnCFDGRMCFGH, RSCFDnCFDGRMCFGH registers can be read/written in 8-bit units

Address: RSCFDnCFDGRMCFG: <RSCFDn_base> + H'04FC

RSCFDnCFDGRMCFG: <RSCFDn_base> + H'04FC,

RSCFDnCFDGRMCFGH: <RSCFDn_base> + H'04FE

RSCFDnCFDGRMCFG: <RSCFDn_base> + H'04FC,

RSCFDnCFDGRMCFGH: <RSCFDn_base> + H'04FD,

RSCFDnCFDGRMCFGH: <RSCFDn_base> + H'04FE,

RSCFDnCFDGRMCFGH: <RSCFDn_base> + H'04FF

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | RCMC |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 0 | RCMC | B'0 | R/W | Interface Mode Select Setting this bit to 1 places the CANFD into CAN FD mode. To place the RS-CANFD module from classical CAN mode to CAN FD mode, modify the RSCFDnCFDGRMCFG register after setting the post-reset values to all the registers and bits that are solely assigned to the register map used in classical CAN mode. 0: Classical CAN mode 1: CAN FD mode |

Modify the RSCFDnCFDGRMCFG register only in global reset mode. Also set the register before setting any other RS-CANFD register.

48.9.2 Details of Channel Related Registers

48.9.2.1 RSCFDnCFDCmNCFG — Channel Normal Bit Rate Configuration Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCmNCFG register can be read/written in 32-bit units

RSCFDnCFDCmNCFG, RSCFDnCFDCmNCFGH registers can be read/written in 16-bit units

RSCFDnCFDCmNCFGLL, RSCFDnCFDCmNCFGHL, RSCFDnCFDCmNCFGHL, RSCFDnCFDCmNCFGHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmNCFG: <RSCFDn_base> + H'0000 + (H'10 × m)

RSCFDnCFDCmNCFG: <RSCFDn_base> + H'0000 + (H'10 × m),

RSCFDnCFDCmNCFGH: <RSCFDn_base> + H'0002 + (H'10 × m)

RSCFDnCFDCmNCFGLL: <RSCFDn_base> + H'0000 + (H'10 × m),

RSCFDnCFDCmNCFGHL: <RSCFDn_base> + H'0001 + (H'10 × m),

RSCFDnCFDCmNCFGHL: <RSCFDn_base> + H'0002 + (H'10 × m),

RSCFDnCFDCmNCFGHH: <RSCFDn_base> + H'0003 + (H'10 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|-----------|-----|-----|-------------|-----|-----------|-----|-----|-------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | NTSEG2[4:0] | | | | — | NTSEG1[6:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | NSJW[4:0] | | | | — | NBRP[9:0] | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 29 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 28 to 24 | NTSEG2[4:0] | B'0_0000 | R/W | Normal Bit Rate Time Segment 2 Control These bits are used to specify a Tq value for the length of phase segment 2 (PHASE_SEG2) for normal bit rate. Allowed values are 2 Tq to 32 Tq, inclusive. Set a value smaller than the value of the NTSEG1[6:0] bits. B'0_0000: Setting prohibited B'0_0001: 2 Tq : : B'1_1110: 31 Tq B'1_1111: 32 Tq |
| 23 | Reserved | B'0 | R | This bit is read as the value after reset. The write value should be the value after reset. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 22 to 16 | NTSEG1[6:0] | B'000_0000 | R/W | <p>Normal Bit Rate Time Segment 1 Control</p> <p>These bits are used to specify a Tq value for the total length of the propagation segment (PROP_SEG) and phase segment 1 (PHASE_SEG1) for normal bit rate.</p> <p>Allowed values are 4 Tq to 128 Tq, inclusive.</p> <p>B'000_0000: Setting prohibited B'000_0001: Setting prohibited B'000_0010: Setting prohibited B'000_0011: 4 Tq : : B'111_1110: 127 Tq B'111_1111: 128 Tq</p> |
| 15 to 11 | NSJW[4:0] | B'0_0000 | R/W | <p>Normal Bit Rate Resynchronization Jump Width Control</p> <p>These bits are used to specify a Tq value for the resynchronization jump width for normal bit rate. Allowed values are 1 Tq to 32 Tq, inclusive. Set a value less than or equal to the value of the NTSEG2[4:0] bits.</p> <p>B'0_0000: 1 Tq B'0_0001: 2 Tq B'0_0010: 3 Tq : : B'1_1110: 31 Tq B'1_1111: 32 Tq</p> |
| 10 | Reserved | B'0 | R | <p>This bit is read as the value after reset.</p> <p>The write value should be the value after reset.</p> |
| 9 to 0 | NBRP[9:0] | H'000 | R/W | <p>Normal Bit Rate Prescaler Division Ratio Set</p> <p>The CANmTq(N) clock (fCANTQ(N)m) is calculated by dividing the CAN clock (fCAN) by the normal bit rate prescaler, ((NBRP[9:0]) + 1). One clock cycle of the CANmTq(N) clock is 1 Time Quantum (Tq).</p> <p>When these bits are set to P (0 to 1023), the normal bit rate prescaler divides fCAN by P + 1.</p> |

Modify the RSCFDnCFDCmNCFG register in channel reset mode or channel halt mode. Set this register in channel reset mode before shifting to channel communication mode or channel wait mode. For a description of the bit timing parameters and settings, see section 48.16.1, Initial Settings.

48.9.2.2 RSCFDnCFDCmCTR — Channel Control Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCmCTR register can be read/written in 32-bit units

RSCFDnCFDCmCTRL, RSCFDnCFDCmCTRHL registers can be read/written in 16-bit units

RSCFDnCFDCmCTRLL, RSCFDnCFDCmCTRLLH, RSCFDnCFDCmCTRHL, RSCFDnCFDCmCTRHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmCTR: <RSCFDn_base> + H'0004 + (H'10 × m)

RSCFDnCFDCmCTRL: <RSCFDn_base> + H'0004 + (H'10 × m),

RSCFDnCFDCmCTRHL: <RSCFDn_base> + H'0006 + (H'10 × m)

RSCFDnCFDCmCTRLL: <RSCFDn_base> + H'0004 + (H'10 × m),

RSCFDnCFDCmCTRLLH: <RSCFDn_base> + H'0005 + (H'10 × m),

RSCFDnCFDCmCTRHL: <RSCFDn_base> + H'0006 + (H'10 × m),

RSCFDnCFDCmCTRHH: <RSCFDn_base> + H'0007 + (H'10 × m)

Value after reset: H'0000_0005

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|-------|-------|-----------|------|------|----------|-----|--------|-------|-------|-------|------------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | ROM | CRCT | — | — | — | CTMS[1:0] | CTME | ERRD | BOM[1:0] | — | TDCVFI | SOCOE | EOCOE | TAIE | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | ALIE | BLIE | OLIE | BORIE | BOEIE | EPIE | EWIE | BEIE | — | — | — | — | RTBO | CSLPR | CHMDC[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 31 | ROM | B'0 | R/W | <p>Limited Operation Mode Enable</p> <p>This bit is used to enable the limited operation mode when the ROM and CTME bits in the RSCFDnCFDCmCTR register are set to 1. Use the limited operation mode only when the CTMS[1:0] bits in the SCFDnCFDCmCTR register are B'00 (standard test mode). Modify this bit only in channel halt mode. This bit is set to 0 in channel reset mode.</p> <p>0: Limited operation mode is disabled. 1: Limited operation mode is enabled.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 30 | CRCT | B'0 | R/W | <p>CRC Error Test Enable</p> <p>This bit is used to test the CRC generator circuit in the RS-CANFD module. Setting this bit to 1 allows the first bit in the ID field to be reversed when a message is received. Due to the bit reversal, the CRC calculation result does not agree with the correct CRC value of the received frame. A CRC error is thus detected (the CERR bit in the RSCFDnCFDCmERFL register is 1). Note the following when using this function:</p> <ul style="list-style-type: none"> • This function is available when the CTME bit in the RSCFDnCFDCmCTR register is 1 (communication test mode enabled). • Communicating with another CAN node is impossible. Use the function in the inter-channel communication test (the CmICBCE bit in the RSCFDnCFDGTSTCFG register is 1). • Bit reversal in the ID field may violate the bit stuffing rule. In this case, a stuff error is detected instead of a CRC error. <p>Modify this bit only in channel halt mode. This bit is set to 0 in channel reset mode.</p> <p>0: The first bit in the receive ID field is not reversed. 1: The first bit in the receive ID field is reversed.</p> |
| 29 to 27 | Reserved | All 0 | R | <p>These bits are read as the value after reset. The write value should be the value after reset.</p> |
| 26, 25 | CTMS[1:0] | B'00 | R/W | <p>Communication Test Mode Select</p> <p>These bits are used to select a communication test mode. Modify these bits only in channel halt mode.</p> <p>These bits are set to 0 in channel reset mode.</p> <p>B'00: Standard test mode B'01: Listen-only mode B'10: Self-test mode 0 (external loopback mode) B'11: Self-test mode 1 (internal loopback mode)</p> |
| 24 | CTME | B'0 | R/W | <p>Communication Test Mode Enable</p> <p>Setting this bit to 1 enables communication test mode. Modify these bits in channel halt mode. This bit is set to 0 in channel reset mode.</p> <p>0: Communication test mode is disabled. 1: Communication test mode is enabled.</p> |
| 23 | ERRD | B'0 | R/W | <p>Error Display Mode Select</p> <p>This bit is used to control the display mode of bits 14 to 8 in the RSCFDnCFDCmERFL register.</p> <p>When this bit is clear to 0, only the flags of the first error event are set to 1. If two or more errors occur in the first error event, all the flags of the detected errors are set to 1.</p> <p>When this bit is set to 1, all the flags of errors that have occurred are set to 1 regardless of the error occurrence order.</p> <p>Modify this bit only in channel reset mode or channel halt mode.</p> <p>0: Error flags are displayed only for the first error information after bits 14 to 8 in RSCFDnCFDCmERFL are all cleared. 1: Error flags for all error information are displayed.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 22, 21 | BOM[1:0] | B'00 | R/W | <p>Bus Off Recovery Mode Select</p> <p>These bits are used to select the bus off recovery mode of the RS-CANFD module.</p> <p>When the BOM[1:0] bits are set to B'00, return from the bus off state to the error active state is compliant with the CAN specifications. That is, the RS-CANFD module reenters the CAN communication (error active state) after 11 consecutive recessive bits are detected 128 times. A bus off recovery interrupt request is generated at the time of return from the bus off state. Even if the CHMDC[1:0] bits are set to B'10 (channel halt mode) before recessive bits are detected 128 times, the RS-CANFD module does not transition to channel halt mode until recessive bits are detected 128 times.</p> <p>When the module reaches the bus off state when the BOM[1:0] bits are set to B'01, the CHMDC[1:0] bits in the RSCFDnCFDCmCTR register ($m = 0$ or 1) are set to B'10 and the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated and the TEC[7:0] and REC[7:0] bits in the RSCFDnCFDCmSTS register are cleared to H'00.</p> <p>When the RS-CANFD module reaches the bus off state when the BOM[1:0] bits are set to B'10, the CHMDC[1:0] bits are set to B'10 and the RS-CANFD module transitions to channel halt mode after return from the bus off state (11 consecutive recessive bits are detected 128 times). A bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to H'00.</p> <p>When the BOM[1:0] bits are set to B'11 and the CHMDC[1:0] bits are set to B'10 while the RS-CANFD module is in the bus off state, the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to H'00. However, if 11 consecutive recessive bits are detected 128 times and the RS-CANFD module has recovered to the error active state from the bus off state before the CHMDC[1:0] bits are set to B'10, a bus off recovery interrupt request is generated.</p> <p>If a program writes to the CHMDC[1:0] bit at the same time as the RS-CANFD module transition to channel halt mode (at bus off entry when the BOM[1:0] bits are B'01 or at bus off end when the BOM[1:0] bits are B'10), the program's writing takes precedence. Modify the BOM[1:0] bits only in channel reset mode.</p> <p>B'00: ISO11898-1 compliant B'01: Entry to channel halt mode automatically at bus-off entry B'10: Entry to channel halt mode automatically at bus-off end B'11: Entry to channel halt mode (in bus-off state) by program request</p> |
| 20 | Reserved | B'0 | R | <p>This bit is read as the value after reset.</p> <p>The write value should be the value after reset.</p> |
| 19 | TDCVFIE | B'0 | R/W | <p>Transmit Delay Correction Violation Interrupt Enable</p> <p>When the TDCVF flag in the RSCFDnCFDCmFDSTS register is set to 1 with the TDCVFIE bit set to 1, an interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Transmit delay correction violation interrupt is disabled. 1: Transmit delay correction violation interrupt is enabled.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 18 | SOCOIE | B'0 | R/W | <p>Communication Success Occurrence Counter Overflow Interrupt Enable</p> <p>When the SOCO flag in the RSCFDnCFDCmFDSTS register is set to 1 with the SOCOIE bit set to 1, an interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Communication success occurrence counter overflow interrupt is disabled.</p> <p>1: Communication success occurrence counter overflow interrupt is enabled.</p> |
| 17 | EOCOIE | B'0 | R/W | <p>Error Occurrence Counter Overflow Interrupt Enable</p> <p>When the EOCO flag in the RSCFDnCFDCmFDSTS register is set to 1 with the EOCOIE bit set to 1, an interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Error occurrence counter overflow interrupt is disabled.</p> <p>1: Error occurrence counter overflow interrupt is enabled.</p> |
| 16 | TAIE | B'0 | R/W | <p>Transmit Abort Interrupt Enable</p> <p>When transmit abort of the transmit buffer is completed with the TAIE bit set to 1, an interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Transmit abort interrupt is disabled.</p> <p>1: Transmit abort interrupt is enabled.</p> |
| 15 | ALIE | B'0 | R/W | <p>Arbitration Lost Interrupt Enable</p> <p>When the ALF flag in the RSCFDnCFDCmERFL register is set to 1 with the ALIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Arbitration lost interrupt is disabled.</p> <p>1: Arbitration lost interrupt is enabled.</p> |
| 14 | BLIE | B'0 | R/W | <p>Bus Lock Interrupt Enable</p> <p>When the BLF flag in the RSCFDnCFDCmERFL register is set to 1 with the BLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Bus lock interrupt is disabled.</p> <p>1: Bus lock interrupt is enabled.</p> |
| 13 | OLIE | B'0 | R/W | <p>Overload Frame Transmit Interrupt Enable</p> <p>When the OVLF flag in the RSCFDnCFDCmERFL register is set to 1 with the OLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Overload frame transmit interrupt is disabled.</p> <p>1: Overload frame transmit interrupt is enabled.</p> |
| 12 | BORIE | B'0 | R/W | <p>Bus Off Recovery Interrupt Enable</p> <p>When the BORF flag in the RSCFDnCFDCmERFL register is set to 1 with the BORIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Bus off recovery interrupt is disabled.</p> <p>1: Bus off recovery interrupt is enabled.</p> |
| 11 | BOEIE | B'0 | R/W | <p>Bus Off Entry Interrupt Enable</p> <p>When the BOEF flag in the RSCFDnCFDCmERFL register is set to 1 with the BOEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Bus off entry interrupt is disabled.</p> <p>1: Bus off entry interrupt is enabled.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|---|
| 10 | EPIE | B'0 | R/W | <p>Error Passive Interrupt Enable</p> <p>When the EPF flag in the RSCFDnCFDCmERFL register is set to 1 with the EPIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Error passive interrupt is disabled. 1: Error passive interrupt is enabled.</p> |
| 9 | EWIE | B'0 | R/W | <p>Error Warning Interrupt Enable</p> <p>When the EWF flag in the RSCFDnCFDCmERFL register is set to 1 with the EWIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Error warning interrupt is disabled. 1: Error warning interrupt is enabled.</p> |
| 8 | BEIE | B'0 | R/W | <p>Bus Error Interrupt Enable</p> <p>When the BEF flag in the RSCFDnCFDCmERFL register is set to 1 with the BEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.</p> <p>0: Bus error interrupt is disabled. 1: Bus error interrupt is enabled.</p> |
| 7 to 4 | Reserved | All 0 | R | <p>These bits are read as the value after reset. The write value should be the value after reset.</p> |
| 3 | RTBO | B'0 | R/W | <p>Forcible Return from Bus-off</p> <p>Setting this bit to 1 in the bus off state forcibly returns the state from the bus off state to the error active state. This bit is automatically cleared to 0. Setting this bit to 1 clears the TEC[7:0] and REC[7:0] bits in the RSCFDnCFDCmSTS register to H'00 and also clears the BOSTS flag in the RSCFDnCFDCmSTS register to 0 (not in bus off state). The other registers remain unchanged. No bus off recovery interrupt request is generated upon return from the bus off state in this case. Use this bit only when the BOM[1:0] bits in the RSCFDnCFDCmCTR are B'00 (ISO11898-1 compliant).</p> <p>A delay of up to 1 CAN bit time occurs after the RTBO bit is set to 1 until the RS-CANFD module transitions to the error active state. Set this bit to 1 in channel communication mode.</p> |
| 2 | CSLPR | B'1 | R/W | <p>Channel Stop Mode</p> <p>Setting this bit to 1 places the channel into channel stop mode. Clearing this bit to 0 makes the channel exit channel stop mode. This bit should not be modified in channel communication mode or channel wait mode.</p> <p>0: Other than channel stop mode 1: Channel stop mode</p> |
| 1, 0 | CHMDC[1:0] | B'01 | R/W | <p>Mode Select</p> <p>These bits are used to select a channel mode (channel communication mode, channel reset mode, or channel halt mode). For details, see section 48.11.2, Channel Modes. Setting the CSLPR bit to 1 in channel reset mode allows transition to channel stop mode. Do not set the CHMDC[1:0] bits to B'11.</p> <p>When the RS-CANFD module has automatically transitioned to channel halt mode based on the setting of the BOM[1:0] bits, the CHMDC[1:0] bits automatically become B'10.</p> <p>B'00: Channel communication mode B'01: Channel reset mode B'10: Channel halt mode B'11: Setting prohibited</p> |

48.9.2.3 RSCFDnCFDCmSTS — Channel Status Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCmSTS register can be read/written in 32-bit units

RSCFDnCFDCmSTSL, RSCFDnCFDCmSTSH registers can be read/written in 16-bit units

RSCFDnCFDCmSTSLL, RSCFDnCFDCmSTSLH, RSCFDnCFDCmSTSHL, RSCFDnCFDCmSTSHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmSTS: <RSCFDn_base> + H'0008 + (H'10 × m)

RSCFDnCFDCmSTSL: <RSCFDn_base> + H'0008 + (H'10 × m),

RSCFDnCFDCmSTSH: <RSCFDn_base> + H'000A + (H'10 × m)

RSCFDnCFDCmSTSLL: <RSCFDn_base> + H'0008 + (H'10 × m),

RSCFDnCFDCmSTSLH: <RSCFDn_base> + H'0009 + (H'10 × m),

RSCFDnCFDCmSTSHL: <RSCFDn_base> + H'000A + (H'10 × m),

RSCFDnCFDCmSTSHH: <RSCFDn_base> + H'000B + (H'10 × m)

Value after reset: H'0000_0005

| | | | | | | | | | | | | | | | | |
|----------------|----------|----|----|----|----|----|----|------|------------|------------|------------|-------|-------|-------------|-------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TEC[7:0] | | | | | | | | REC[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | ESIF | COMS TS | RECST S | TRMST S | BOSTS | EPSTS | CSLPS TS | CHLTS TS | CRSTS TS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R/W* | R | R | R | R | R | R | R | R |

Note: * The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 24 | TEC[7:0] | H'00 | R | The transmit error counter (TEC) can be read. These bits contain the transmit error counter value. For transmit error counter increment/decrement conditions, see the CAN specification (ISO11898-1). These bits are cleared to 0 in channel reset mode. |
| 23 to 16 | REC[7:0] | H'00 | R | The receive error counter (REC) can be read. These bits contain the receive error counter value. For receive error counter increment/decrement conditions, see the CAN specifications (ISO11898-1). These bits are cleared to 0 in channel reset mode. |
| 15 to 9 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 8 | ESIF | B'0 | R/W* | <p>Error State Indicate Flag</p> <p>This flag becomes 1 when the recessive ESI bit has been detected in the message that has been normally received. In loopback mode or mirror mode, a message transmitted from the own CAN node is considered as a received message. To clear this flag to 0, the program must write 0 to the flag; this flag cannot be set to 1 by a program. If the timing when this flag becomes 1 and the program writes 0 to this flag is the same, the flag will become 1.</p> <p>This flag is cleared to 0 in channel reset mode.</p> <p>0: No CAN FD message that the ESI bit is recessive has been received.</p> <p>1: At least one CAN FD message that the ESI bit is recessive has been received.</p> |
| 7 | COMSTS | B'0 | R | <p>Communication Status Flag</p> <p>This bit indicates that communication is ready.</p> <p>This flag becomes 1 when the RS-CANFD module has detected 11 consecutive recessive bits after it has transitioned from channel reset mode or channel halt mode to channel communication mode. This flag is cleared to 0 in channel reset mode or channel halt mode.</p> <p>0: Communication is not ready.</p> <p>1: Communication is ready.</p> |
| 6 | RECSTS | B'0 | R | <p>Receive Status Flag</p> <p>This flag is set to 1 when reception has started, and is cleared to 0 when the bus has become idle or transmission has started.</p> <p>0: Bus idle, in transmission or bus off state</p> <p>1: In reception</p> |
| 5 | TRMSTS | B'0 | R | <p>Transmit Status Flag</p> <p>This flag is set to 1 when transmission has started, and is cleared to 0 when the bus has become idle or reception has started. This flag remains 1 in the bus off state.</p> <p>0: Bus idle or in reception</p> <p>1: In transmission or bus off state</p> |
| 4 | BOSTS | B'0 | R | <p>Bus Off Status Flag</p> <p>This flag is set to 1 when the bus off state ($TEC[7:0] > 255$) is entered. It is cleared to 0 when the RS-CANFD module has exited the bus off state.</p> <p>0: Not in bus off state</p> <p>1: In bus off state</p> |
| 3 | EPSTS | B'0 | R | <p>Error Passive Status Flag</p> <p>This flag is set to 1 when the RS-CANFD module has entered the error passive state ($(128 \leq TEC[7:0] \leq 255)$ or $(128 \leq REC[7:0])$). It is cleared to 0 when the RS-CANFD module has exited the error passive state or has entered channel reset mode.</p> <p>0: Not in error passive state</p> <p>1: In error passive state</p> |
| 2 | CSLPSTS | B'1 | R | <p>Channel Stop Status Flag</p> <p>This flag is set to 1 when the RS-CANFD module has transitioned to channel stop mode, and is cleared to 0 when the RS-CANFD module has returned from channel stop mode.</p> <p>0: Not in channel stop mode</p> <p>1: In channel stop mode</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 1 | CHLTSTS | B'0 | R | <p>Channel Halt Status Flag</p> <p>This flag is set to 1 when the RS-CANFD module has transitioned to channel halt mode, and is cleared to 0 when the RS-CANFD module has returned from channel halt mode.</p> <p>0: Not in channel halt mode 1: In channel halt mode</p> |
| 0 | CRSTSTS | B'1 | R | <p>Channel Reset Status Flag</p> <p>This flag is set to 1 when the RS-CANFD module has transitioned to channel reset mode, and is cleared to 0 when the RS-CANFD module has transitioned to channel communication mode or channel halt mode. This flag remains 1 when the RS-CANFD module transitions from channel reset mode to channel stop mode.</p> <p>0: Not in channel reset mode 1: In channel reset mode</p> |

48.9.2.4 RSCFDnCFDCmERFL — Channel Error Flag Status Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCmERFL register can be read/written in 32-bit units

RSCFDnCFDCmERFLL, RSCFDnCFDCmERFLH registers can be read/written in 16-bit units

RSCFDnCFDCmERFLLL, RSCFDnCFDCmERFLLH, RSCFDnCFDCmERFLHL, RSCFDnCFDCmERFLHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmERFL: <RSCFDn_base> + H'000C + (H'10 × m)

RSCFDnCFDCmERFLL: <RSCFDn_base> + H'000C + (H'10 × m),

RSCFDnCFDCmERFLH: <RSCFDn_base> + H'000E + (H'10 × m)

RSCFDnCFDCmERFLLL: <RSCFDn_base> + H'000C + (H'10 × m),

RSCFDnCFDCmERFLLH: <RSCFDn_base> + H'000D + (H'10 × m),

RSCFDnCFDCmERFLHL: <RSCFDn_base> + H'000E + (H'10 × m),

RSCFDnCFDCmERFLHH: <RSCFDn_base> + H'000F + (H'10 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|--------------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CRCREG[14:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | ADERR | BOERR | B1ERR | CERR | AERR | FERR | SERR | ALF | BLF | OVLf | BORF | BOEF | EPF | EWf | BEF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* |

Note: * The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 | Reserved | B'0 | R | When read, the value after reset is returned. When writing to this bit, write the value after reset. |
| 30 to 16 | CRCREG [14:0] | H'0000 | R | CRC Calculation Data (15-bit CRC length) When the CTME bit in the RSCFDnCFDCmCTR register is set to 1 (communication test mode is enabled), if the transmit or receive message is in the classical CAN frame (15-bit CRC length), this flag is updated and the CRC value calculated based on the transmit or receive message can be read. When the CAN FD frame is transmitted or received, the CRCREG[20:0] bits in the RSCFDnCFDCmFDCRC register are updated. When the CTME bit is set to 0 (communication test mode is disabled), these bits are always read as 0. |
| 15 | Reserved | B'0 | R | When read, the value after reset is returned. When writing to this bit, write the value after reset. |
| 14 | ADERR | B'0 | R/W | ACK Delimiter Error Flag This flag is set to 1 when a form error has been detected in the ACK delimiter during transmission. 0: No ACK delimiter error is detected. 1: ACK delimiter error is detected. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 13 | B0ERR | B'0 | R/W | <p>Dominant Bit Error Flag</p> <p>This flag is set to 1 when a recessive bit has been detected though a dominant bit was transmitted.</p> <p>0: No dominant bit error is detected.</p> <p>1: Dominant bit error is detected.</p> |
| 12 | B1ERR | B'0 | R/W | <p>Recessive Bit Error Flag</p> <p>This flag is set to 1 when a dominant bit has been detected though a recessive bit was transmitted.</p> <p>0: No recessive bit error is detected.</p> <p>1: Recessive bit error is detected.</p> |
| 11 | CERR | B'0 | R/W | <p>CRC Error Flag</p> <p>This flag is set to 1 when a CRC error has been detected.</p> <p>0: No CRC error is detected.</p> <p>1: CRC error is detected.</p> |
| 10 | AERR | B'0 | R/W | <p>ACK Error Flag</p> <p>This flag is set to 1 when an ACK error has been detected.</p> <p>0: No ACK error is detected.</p> <p>1: ACK error is detected.</p> |
| 9 | FERR | B'0 | R/W | <p>Form Error Flag</p> <p>This flag is set to 1 when a form error has been detected.</p> <p>0: No form error is detected.</p> <p>1: Form error is detected.</p> |
| 8 | SERR | B'0 | R/W | <p>Stuff Error Flag</p> <p>This flag is set to 1 when a stuff error has been detected.</p> <p>0: No stuff error is detected.</p> <p>1: Stuff error is detected.</p> |
| 7 | ALF | B'0 | R/W | <p>Arbitration-lost Flag</p> <p>This flag is set to 1 when an arbitration-lost has been detected.</p> <p>0: No arbitration-lost is detected.</p> <p>1: Arbitration-lost is detected.</p> |
| 6 | BLF | B'0 | R/W | <p>Bus Lock Flag</p> <p>This flag is set to 1 when 32 consecutive dominant bits have been detected on the CAN bus in channel communication mode. After that, detection of a dominant lock is restarted when either of the following conditions is met.</p> <ul style="list-style-type: none"> • A recessive bit is detected after the BLF bit has been cleared from 1 to 0. • The RS-CANFD module transitions to channel reset mode and returns to channel communication mode after the BLF bit has been cleared from 1 to 0. <p>0: No channel bus is detected.</p> <p>1: Channel bus is detected.</p> |
| 5 | OVLF | B'0 | R/W | <p>Overload Flag</p> <p>This flag is set to 1 when the overload frame transmit condition has been detected when performing reception or transmission.</p> <p>0: No overload is detected.</p> <p>1: Overload is detected.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 4 | BORF | B'0 | R/W | <p>Bus Off Recovery Flag</p> <p>This flag is set to 1 when 11 consecutive recessive bits have been detected 128 times and the RS-CANFD module returns from the bus off state. However, this flag is not set to 1 if the RS-CANFD module returns from the bus off state in any of the following ways before 11 consecutive recessive bits are detected 128 times.</p> <ul style="list-style-type: none"> The CHMDC[1:0] bits in the RSCFDnCFDCmCTR register are set to B'01 (channel reset mode). The RTBO bit in the RSCFDnCFDCmCTR register is set to 1 (forcible return from the bus off state is made). The BOM[1:0] bits in the RSCFDnCFDCmCTR register are set to B'01 (transition to channel halt mode at bus off entry). The CHMDC[1:0] bits in the RSCFDnCFDCmCTR register are set to B'10 (channel halt mode) before 11 consecutive recessive bits are detected 128 times with the BOM[1:0] bits set to B'11 (transition to channel halt mode upon a request from the program during bus off). <p>0: No bus off recovery is detected. 1: Bus off recovery is detected.</p> |
| 3 | BOEF | B'0 | R/W | <p>Bus Off Entry Flag</p> <p>This flag is set to 1 when the bus off state is reached (TEC[7:0] value > 255). This flag is also set to 1 if the bus off state is reached when the BOM[1:0] bits in the RSCFDnCFDCmCTR register (m = 0 or 1) set to B'01 (transition to channel halt mode at bus off entry).</p> <p>0: No bus off entry is detected. 1: Bus off entry is detected.</p> |
| 2 | EPF | B'0 | R/W | <p>Error Passive Flag</p> <p>This flag becomes 1 when the error passive state is reached (REC[7:0] or TEC[7:0] value > 127). This flag becomes 1 only when the REC[7:0] or TEC[7:0] value first exceeds 127. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 127, this bit is not set to 1 until both REC[7:0] and TEC[7:0] values become 127 or less and then the REC[7:0] or TEC[7:0] value exceeds 127 again.</p> <p>0: No error passive is detected. 1: Error passive is detected.</p> |
| 1 | EWF | B'0 | R/W | <p>Error Warning Flag</p> <p>This flag is set to 1 only when the REC[7:0] or TEC[7:0] value first exceeds 95. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 95, this bit is not set to 1 until both REC[7:0] and TEC[7:0] values become 95 or less and then the REC[7:0] or TEC[7:0] value exceeds 95 again.</p> <p>0: No error warning is detected. 1: Error warning is detected.</p> |
| 0 | BEF | B'0 | R/W | <p>Bus Error Flag</p> <p>This flag is set to 1 when any one of the ADERR, B0ERR, B1ERR, CERR, AERR, FERR, and SERR flags in the RSCFDnCFDCmERFL register is set to 1.</p> <p>0: No channel bus error is detected. 1: Channel bus error is detected.</p> |

Note: To clear the flag of this register to 0, use a store instruction to write "0" to the given flag and "1" to the other flags.

See the CAN specification (ISO11898-1) for a description of error occurrence conditions. To clear each flag of this register, the program must write a 0 to the corresponding bit. These flags cannot be set to 1 by the program. If any of these flags is set to 0 at the same time that the program writes 0 to the flag, the flag is still set to 1. The channel reset mode transition clears all of these flags to 0.

If the ERRD bit in the RSCFDnCFDCmCTR register is set to 0 (ie, only the flags for the first error event are displayed) and an error related to bits 14 to 8 of RSCFDnCFDCmERFL is detected, the flag bits are only set by the error event if bits 14 to 8 were all 0 at the when time the error occurred.

48.9.2.5 RSCFDnCFDCmDCFG — Channel Data Bit Rate Configuration Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCmDCFG register can be read/written in 32-bit units

RSCFDnCFDCmDCFGL, RSCFDnCFDCmDCFGLH registers can be read/written in 16-bit units

RSCFDnCFDCmDCFGLL, RSCFDnCFDCmDCFGLH, RSCFDnCFDCmDCFGLH, RSCFDnCFDCmDCFGLH, RSCFDnCFDCmDCFGLH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmDCFG: <RSCFDn_base> + H'0500 + (H'20 × m)

RSCFDnCFDCmDCFGL: <RSCFDn_base> + H'0500 + (H'20 × m),

RSCFDnCFDCmDCFGLH: <RSCFDn_base> + H'0502 + (H'20 × m)

RSCFDnCFDCmDCFGLL: <RSCFDn_base> + H'0500 + (H'20 × m),

RSCFDnCFDCmDCFGLH: <RSCFDn_base> + H'0501 + (H'20 × m),

RSCFDnCFDCmDCFGLH: <RSCFDn_base> + H'0502 + (H'20 × m),

RSCFDnCFDCmDCFGLH: <RSCFDn_base> + H'0503 + (H'20 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|-----------|-----|-----|-----------|-------------|-----|-----|-------------|-----|-----|-----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | DSJW[2:0] | | | — | DTSEG2[2:0] | | | DTSEG1[3:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | DBRP[7:0] | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 27 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 26 to 24 | DSJW[2:0] | B'000 | R/W | Data Bit Rate Resynchronization Jump Width Control These bits are used to specify a Tq value for the resynchronization jump width for data bit rate. Allowed values are 1 Tq to 8 Tq, inclusive. Set a value less than or equal to the value of the DTSEG2[2:0] bits. B'000: 1 Tq B'001: 2 Tq B'010: 3 Tq B'011: 4 Tq B'100: 5 Tq B'101: 6 Tq B'110: 7 Tq B'111: 8 Tq |
| 23 | Reserved | B'0 | R | This bit is read as the value after reset. The write value should be the value after reset. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 22 to 20 | DTSEG2[2:0] | B'000 | R/W | <p>Data Bit Rate Time Segment 2 Control</p> <p>These bits are used to specify a Tq value for the length of phase segment 2 (PHASE_SEG2) for data bit rate.</p> <p>Allowed values are 2 Tq to 8 Tq, inclusive.</p> <p>Set a value smaller than the value of the DTSEG1[3:0] bits.</p> <p>B'000: Setting prohibited B'001: 2 Tq B'010: 3 Tq B'011: 4 Tq B'100: 5 Tq B'101: 6 Tq B'110: 7 Tq B'111: 8 Tq</p> |
| 19 to 16 | DTSEG1[3:0] | B'0000 | R/W | <p>Data Bit Rate Time Segment 1 Control</p> <p>These bits are used to specify a Tq value for the total length of the propagation segment (PROP_SEG) and phase segment 1 (PHASE_SEG1) for data bit rate.</p> <p>Allowed values are 2 Tq to 16 Tq, inclusive.</p> <p>B'0000: Setting prohibited B'0001: 2 Tq B'0010: 3 Tq B'0011: 4 Tq B'0100: 5 Tq B'0101: 6 Tq B'0110: 7 Tq B'0111: 8 Tq B'1000: 9 Tq B'1001: 10 Tq B'1010: 11 Tq B'1011: 12 Tq B'1100: 13 Tq B'1101: 14 Tq B'1110: 15 Tq B'1111: 16 Tq</p> |
| 15 to 8 | Reserved | All 0 | R | <p>These bits are read as the value after reset.</p> <p>The write value should be the value after reset.</p> |
| 7 to 0 | DBRP[7:0] | H'00 | R/W | <p>Data Bit Rate Prescaler Division Ratio Set</p> <p>The CANmTq(D) clock (fCANTQ(D)m) is calculated by dividing the CAN clock (fCAN) by the data bit rate prescaler, ((DBRP[7:0]) + 1). One clock cycle of the CANmTq(D) clock is 1 Time Quantum (Tq).</p> |

Modify the RSCFDnCFDCmDCFG register in channel reset mode or channel halt mode. Set this register in channel reset mode before shifting to channel communication mode or channel halt mode. If the classical CAN frame is only used, set the same value to the RSCFDnCFDCmDCFG register as the set value in the RSCFDnCFDCmNCFG register. For a description of the bit timing parameters and settings, see section 48.16.1, Initial Settings.

48.9.2.6 RSCFDnCFDCmFDCFG — Channel CAN FD Configuration Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCmFDCFG register can be read/written in 32-bit units
 RSCFDnCFDCmFDCFGL, RSCFDnCFDCmFDCFGH registers can be read/written in 16-bit units
 RSCFDnCFDCmFDCFGLL, RSCFDnCFDCmFDCFGLH, RSCFDnCFDCmFDCFGHL,
 RSCFDnCFDCmFDCFGHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmFDCFG: <RSCFDn_base> + H'0504 + (H'20 × m)
 RSCFDnCFDCmFDCFGL: <RSCFDn_base> + H'0504 + (H'20 × m),
 RSCFDnCFDCmFDCFGH: <RSCFDn_base> + H'0506 + (H'20 × m)
 RSCFDnCFDCmFDCFGLL: <RSCFDn_base> + H'0504 + (H'20 × m),
 RSCFDnCFDCmFDCFGLH: <RSCFDn_base> + H'0505 + (H'20 × m),
 RSCFDnCFDCmFDCFGHL: <RSCFDn_base> + H'0506 + (H'20 × m),
 RSCFDnCFDCmFDCFGHH: <RSCFDn_base> + H'0507 + (H'20 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|------|-----------|-----------|-------|----|-----------|-----|-----|-----|-------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | TMME | GWBR S | GWFD F | GWEN | — | TDCO[6:0] | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | ESIC | TDCE | TDCOC | — | — | — | — | — | EOCCFG[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 28 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 27 | TMME | B'0 | R/W | Transmit Buffer Merge Mode Enable When this bit is set to 1, a transmit buffer merge mode is enabled. Modify this bit only in channel reset mode or channel halt mode. 0: The transmit buffer merge mode is disabled. 1: The transmit buffer merge mode is enabled. |
| 26 | GWBR S | B'0 | R/W | Gateway BRS Set This bit is used to set the BRS bit in the CAN FD frame which is transmitted by the gateway function when the GWEN bit is 1. Write 0 to this bit when the GWFDF bit is cleared to 0. Modify this bit only in channel reset mode. 0: The BRS bit in the receive frame is cleared to 0 and transmitted. 1: The BRS bit in the receive frame is set to 1 and transmitted. |
| 25 | GWFD F | B'0 | R/W | Gateway FDF Set This bit is used to set the FDF bit in the CAN FD frame which is transmitted by the gateway function when the GWEN bit is 1. Modify this bit only in channel reset mode. 0: The receive frame is transmitted as the classical CAN frame. 1: The receive frame is transmitted as the CAN FD frame. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 24 | GWEN | B'0 | R/W | <p>CAN-CAN FD Gateway Enable</p> <p>This bit is used to control operation of the transmit/receive FIFO buffer that the CFM[1:0] bits in the RSCFDnCFDCFCCK register are set to B'10 (gateway mode).</p> <p>Setting this bit to 1 enables the CAN-CAN FD gateway and transmission by the format which is different from that received by the gateway function. The receive frame is replaced according to the settings of GWFDF and GWBRS bits. When the DLC value of the received classical CAN frame is equal to or larger than B'1001 and the GWFDF bit is 1 (CAN FD frame), the DLC value is replaced with B'1000.</p> <p>When this bit is 1, do not route the following frames by the gateway function.</p> <ul style="list-style-type: none"> • CAN FD frame of which payload length is longer than 8 bytes • Remote frame <p>Modify this bit only in channel reset mode.</p> <p>Table 48.21 shows the format and setting of the transmit/receive frame when the CAN-CAN FD gateway is enabled.</p> <p>0: The CAN-CAN FD gateway is disabled. 1: The CAN-CAN FD gateway is enabled.</p> |
| 23 | Reserved | B'0 | R | <p>This bit is read as the value after reset. The write value should be the value after reset.</p> |
| 22 to 16 | TDCO[6:0] | H'00 | R/W | <p>Transmit Delay Correction Offset Set</p> <p>These bits are used to set the SSP offset value. The usage of the value depends on the TDCOC bit in the RSCFDnCFDCmFDCFG register.</p> <p>When the TDCOC bit is 0, the result of transmit delay correction is equal to the sum of the values of the measured delay and the TDCO[6:0] bits (it is rounded down to T_q which is the approximate integer).</p> <p>When the TDCOC bit is 1, the result of transmit delay correction is equal to the value of the TDCO[6:0] bits.</p> <p>The SSP offset value is the set value of TDCO[6:0] bits + 1.</p> <p>Modify these bits only in channel reset mode or channel halt mode.</p> |
| 15 to 11 | Reserved | All 0 | R | <p>This bit is read as the value after reset. The write value should be the value after reset.</p> |
| 10 | ESIC | B'0 | R/W | <p>Error State Indicate Mode Select</p> <p>When this bit is set to 1 and a channel is error-active, the ESI bit value that has been set in the transmit/receive FIFO buffer or the transmit buffer (the CFESI bit in the RSCFDnCFDCFFDCSTSk register or the TMESI bit in the RSCFDnCFDTMFDCTRp register) is transmitted as the ESI bit value in the transmit message. When a channel is error-passive or the ESIC bit is 0, the channel status is transmitted as the ESI bit value. Modify this bit only in channel reset mode or channel halt mode.</p> <p>See Table 48.22.</p> <p>0: The error state of a channel is always transmitted as the ESI bit for the frame.</p> <p>1: When a channel is not error-passive, the error state of the message buffer is transmitted as the ESI bit for the frame.</p> <p>When a channel is error-passive, the error state of the channel is transmitted as the ESI bit for the frame.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|--------------|---------------|-----|---|
| 9 | TDCE | B'0 | R/W | <p>Transmit Delay Correction Enable</p> <p>When this bit is set to 1, transmit delay correction is enabled. Modify this bit only in channel reset mode or channel halt mode.</p> <p>0: Transmit delay correction is disabled. 1: Transmit delay correction is enabled.</p> |
| 8 | TDCOC | B'0 | R/W | <p>Transmit Delay Correction Measurement Select</p> <p>When this bit is cleared to 0, the SSP position is defined by the sum of the measured delay and the SSP offset value (fixed value). When this bit is set to 1, the SSP position is only defined by the SSP offset value.</p> <p>Modify this bit only in channel reset mode or channel halt mode.</p> <p>0: Measurement and offset 1: Offset only</p> |
| 7 to 3 | Reserved | All 0 | R | <p>These bits are read as the value after reset.</p> <p>The write value should be the value after reset.</p> |
| 2 to 0 | EOCCFG[2:0] | B'000 | R/W | <p>Error Occurrence Count Method Select</p> <p>These bits are used to select the frame format and transmit/receive direction when a CAN bus error is counted by an error occurrence counter.</p> <p>Modify these bits only in channel reset mode or channel halt mode.</p> <p>B'000: All transmit and receive messages B'001: All transmit messages B'010: All receive messages B'011: Setting prohibited B'100: Only the data phase of the transmitted or received CAN FD message B'101: Only the data phase of the transmitted CAN FD message B'110: Only the data phase of the received CAN FD message B'111: Setting prohibited</p> |

Table 48.21 Operation when the CAN-CAN FD Gateway is Enabled

| Receive Frame | | | GWDFDF Bit | Transmit Frame | | |
|---------------|----------|----------------------------------|------------|----------------|---|--------------------------------------|
| Format | BRS Bit | Receive DLC Value | | Format | BRS Bit | Transmit DLC Value |
| Classical CAN | None | DLC ≤ "B'1000" DLC > "B'1000" | B'0 | Classical CAN | None | Not replaced |
| CAN FD | Optional | DLC ≤ "B'1000" | | | | |
| Classical CAN | None | DLC ≤ "B'1000" DLC > "B'1000" | B'1 | CAN FD | Depends on the setting of the GWBRS bit | Not replaced Replaced with B'1000 |
| CAN FD | Optional | DLC ≤ "B'1000" | | | | Not replaced |

Table 48.22 Transmit ESI Value

| ESIC Bit | Channel Status | Transmit ESI Value |
|----------|----------------|--|
| B'0 | Error-active | 0 (A node is error-active.) |
| | Error-passive | 1 (A node is error-passive.) |
| B'1 | Error-active | The ESI bit value that has been set in the transmit/receive FIFO buffer or the transmit buffer (the CFESI bit in the RSCFDnCFDCFFDCSTSk register or the TMESI bit in the RSCFDnCFDTMFDCTRp register) |
| | Error-passive | 1 (A node is error-passive.) |

48.9.2.7 RSCFDnCFDCmFDCTR — Channel CAN FD Control Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCmFDCTR register can be read/written in 32-bit units
 RSCFDnCFDCmFDCTRL, RSCFDnCFDCmFDCTRH registers can be read/written in 16-bit units
 RSCFDnCFDCmFDCTRLLL, RSCFDnCFDCmFDCTRLH, RSCFDnCFDCmFDCTRHL,
 RSCFDnCFDCmFDCTRHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCmFDCTR: <RSCFDn_base> + H'0508 + (H'20 × m)
 RSCFDnCFDCmFDCTRL: <RSCFDn_base> + H'0508 + (H'20 × m),
 RSCFDnCFDCmFDCTRH: <RSCFDn_base> + H'050A + (H'20 × m)
 RSCFDnCFDCmFDCTRLLL: <RSCFDn_base> + H'0508 + (H'20 × m),
 RSCFDnCFDCmFDCTRLH: <RSCFDn_base> + H'0509 + (H'20 × m),
 RSCFDnCFDCmFDCTRHL: <RSCFDn_base> + H'050A + (H'20 × m),
 RSCFDnCFDCmFDCTRHH: <RSCFDn_base> + H'050B + (H'20 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SOCCLR | EOCCLR |
| | | | | | | | | | | | | | | | R | R |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 2 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 1 | SOCCLR | B'0 | R/W | Communication Success Occurrence Counter Clear When this bit is set to 1, the communication success occurrence counter (the SOC[7:0] bits in the RSCFDnCFDCmFDSTS register) is cleared. This bit is automatically cleared to 0. |
| 0 | EOCCLR | B'0 | R/W | Error Occurrence Counter Clear When this bit is set to 1, the error occurrence counter (the EOC[7:0] bits in the RSCFDnCFDCmFDSTS register) is cleared. This bit is automatically cleared to 0. |

48.9.2.8 RSCFDnCFDCmFDSTS — Channel CAN FD Status Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCmFDSTS register can be read only in 32-bit units

RSCFDnCFDCmFDSTSL, RSCFDnCFDCmFDSTSH registers can be read only in 16-bit units

RSCFDnCFDCmFDSTSLL, RSCFDnCFDCmFDSTSLH, RSCFDnCFDCmFDSTSHL,

RSCFDnCFDCmFDSTSHH registers can be read only in 8-bit units

Address: RSCFDnCFDCmFDSTS: <RSCFDn_base> + H'050C + (H'20 × m)

RSCFDnCFDCmFDSTSL: <RSCFDn_base> + H'050C + (H'20 × m),

RSCFDnCFDCmFDSTSH: <RSCFDn_base> + H'050E + (H'20 × m)

RSCFDnCFDCmFDSTSLL: <RSCFDn_base> + H'050C + (H'20 × m),

RSCFDnCFDCmFDSTSLH: <RSCFDn_base> + H'050D + (H'20 × m),

RSCFDnCFDCmFDSTSHL: <RSCFDn_base> + H'050E + (H'20 × m),

RSCFDnCFDCmFDSTSHH: <RSCFDn_base> + H'050F + (H'20 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----------|----|----|----|----|----|------|------|----------|-----------|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SOC[7:0] | | | | | | | | EOC[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | SOCO | EOCO | TDCVF | TDCR[6:0] | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W* | R/W* | R/W* | R | R | R | R | R | R | R |

Note: * The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 24 | SOC[7:0] | H'00 | R | <p>Communication Success Occurrence Counter</p> <p>These bits indicate a value of the communication success occurrence counter. This counter is incremented when reception or transmission of a message completed without any errors. When this counter reaches H'FF, updating is stopped. In loopback mode, this counter is incremented twice.</p> <p>These bits are cleared to 0 when 1 is written to the SOCCLR bit in the RSCFDnCFDCmCTR register. In channel reset mode, these bits are cleared to 0.</p> |
| 23 to 16 | EOC[7:0] | H'00 | R | <p>Error Occurrence Counter</p> <p>These bits indicate a value of the error occurrence counter. This counter is incremented when an error occurs according to the condition that has been set for the EOCCFG[2:0] bits in the RSCFDnCFDCmFDCFG register. When this counter reaches H'FF, updating is stopped.</p> <p>These bits are cleared to 0 when 1 is written to the EOCCLR bit in the RSCFDnCFDCmCTR register. In channel reset mode, these bits are cleared to 0.</p> |
| 15 to 10 | Reserved | All 0 | R | These bits are read as the value after reset. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|--|
| 9 | SOCO | B'0 | R/W | <p>Communication Success Occurrence Counter Overflow Flag</p> <p>This bit indicates that an overflow was generated in the communication success occurrence counter.</p> <p>When the SOC[7:0] bits reached H'FF and reception or transmission of a message completed, this bit is set to 1. In channel reset mode, this bit is cleared to 0.</p> <p>0: An overflow is not generated in the communication success occurrence counter.</p> <p>1: An overflow is generated in the communication success occurrence counter.</p> |
| 8 | EOCO | B'0 | R/W | <p>Error Occurrence Counter Overflow Flag</p> <p>This bit indicates that an overflow was generated in the error occurrence counter.</p> <p>When the SOC[7:0] bits reached H'FF and a CAN bus error was detected under the condition that has been set for the EOCCFG[2:0] bits in the RSCFDnCFDCmFDCFG register, this bit is set to 1. In channel reset mode, this bit is cleared to 0.</p> <p>0: An overflow is not generated in the error occurrence counter.</p> <p>1: An overflow is generated in the error occurrence counter.</p> |
| 7 | TDCVF | B'0 | R/W | <p>Transmit Delay Correction Violation Flag</p> <p>This bit indicates violation of transmit delay correction.</p> <p>The transmit data is compared to the receive CAN bus level that has been delayed due to the loop delay of a transceiver. This delay changes depending on the physical factor such as a temperature. Since the TDCR[6:0] flags are updated in each message, the temporary maximum delay cannot be checked.</p> <p>This bit is set to 1 when the maximum transmit delay correction exceeds 3-CANm bit time – 2 Tq (the CANm bit time and Tq are values of the data bit rate).</p> <p>In channel reset mode, this bit is cleared to 0.</p> <p>0: Transmit delay correction violation does not occur.</p> <p>1: Transmit delay correction violation occurs.</p> |
| 6 to 0 | TDCR[6:0] | H'00 | R | <p>Transmit Delay Correction Result Status</p> <p>These bits indicate the transmit delay correction result with a multiple of the CAN clock (fCAN).</p> <p>The result depends on the setting of the TDCOC and TDCO[6:0] bits in the CFDCmFDCFG register.</p> <p>In channel reset mode, these bits are cleared to 0.</p> |

48.9.2.9 RSCFDnCFDCmFDCRC — Channel CAN FD CRC Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCmFDCRC register can be read only in 32-bit units
 RSCFDnCFDCmFDCRCL, RSCFDnCFDCmFDCRCH registers can be read only in 16-bit units
 RSCFDnCFDCmFDCRCLL, RSCFDnCFDCmFDCRCLH, RSCFDnCFDCmFDCRCHL,
 RSCFDnCFDCmFDCRCHH registers can be read only in 8-bit units

Address: RSCFDnCFDCmFDCRC: <RSCFDn_base> + H'0510 + (H'20 × m)
 RSCFDnCFDCmFDCRCL: <RSCFDn_base> + H'0510 + (H'20 × m),
 RSCFDnCFDCmFDCRCH: <RSCFDn_base> + H'0512 + (H'20 × m)
 RSCFDnCFDCmFDCRCLL: <RSCFDn_base> + H'0510 + (H'20 × m),
 RSCFDnCFDCmFDCRCLH: <RSCFDn_base> + H'0511 + (H'20 × m),
 RSCFDnCFDCmFDCRCHL: <RSCFDn_base> + H'0512 + (H'20 × m),
 RSCFDnCFDCmFDCRCHH: <RSCFDn_base> + H'0513 + (H'20 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|--------------|----|----|----|----|----|----|----|----|----|----|---------------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | | | | | | | | | | | CRCREG[20:16] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CRCREG[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 21 | Reserved | All 0 | R | Reserved These bits are read as 0. The write value should be 0. |
| 20 to 0 | CRCREG [20:0] | H'00_0000 | R | CRC Calculation Data (17-bit or 21-bit CRC length) When the CTME bit in the RSCFDnCFDCmCTR register is set to 1 (communication test mode is enabled), if the transmit or receive message is in the CAN FD frame (17-bit or 21-bit CRC length), these flags are updated and the CRC value calculated based on the transmit or receive message can be read. When the message has the 17-bit CRC length, bits 20 to 17 are always read as 0. When the classical CAN frame is transmitted or received, the CRCREG[14:0] bits in the RSCFDnCFDCmERFL register are updated. When the CTME bit is cleared to 0 (communication test mode is disabled), these bits are always read as 0. |

48.9.3 Details of Global Related Registers

48.9.3.1 RSCFDnCFDGCFCFG — Global Configuration Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGCFCFG register can be read/written in 32-bit units

RSCFDnCFDGCFCFGL, RSCFDnCFDGCFCFGH registers can be read/written in 16-bit units

RSCFDnCFDGCFCFGLL, RSCFDnCFDGCFCFGLH, RSCFDnCFDGCFCFGLH, RSCFDnCFDGCFCFGLH, RSCFDnCFDGCFCFGLH registers can be read/written in 8-bit units

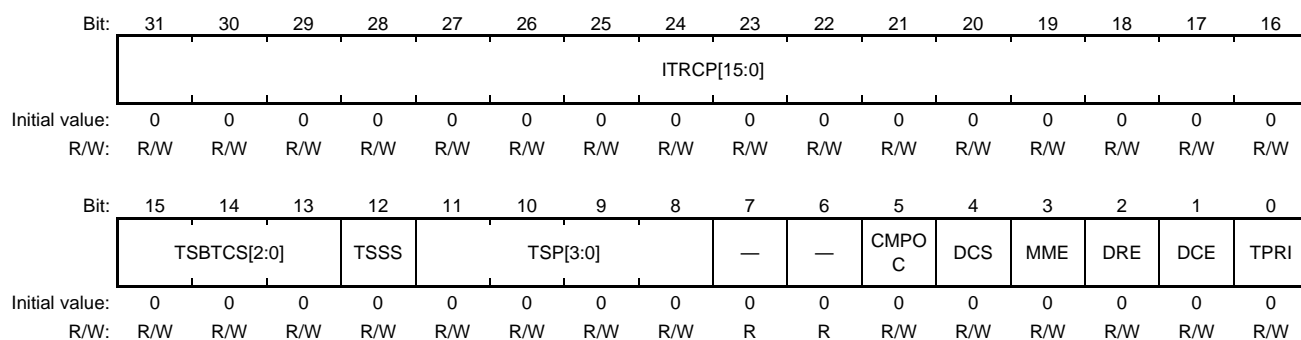
Address: RSCFDnCFDGCFCFG: <RSCFDn_base> + H'0084

RSCFDnCFDGCFCFGL: <RSCFDn_base> + H'0084, RSCFDnCFDGCFCFGH: <RSCFDn_base> + H'0086

RSCFDnCFDGCFCFGLL: <RSCFDn_base> + H'0084, RSCFDnCFDGCFCFGLH: <RSCFDn_base> + H'0085,

RSCFDnCFDGCFCFGLH: <RSCFDn_base> + H'0086, RSCFDnCFDGCFCFGLH: <RSCFDn_base> + H'0087

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 16 | ITRCP[15:0] | H'0000 | R/W | Interval Timer Prescaler Set These bits are used to set a clock source division value of the interval timer for FIFO buffers. See section 48.13.3.1, Interval Transmission Function. When these bits are set to M, the pclk is divided by M. Setting H'0000 is prohibited when the interval timer is in use. |
| 15 to 13 | TSBTCS[2:0] | B'000 | R/W | Timestamp Clock Source Select When the TSSS bit is 1, these bits are used to select the channel of the normal bit time clock that will be the clock source of the timestamp counter. However, do not select the channel which handles the CAN FD frame. B'000: Channel 0 bit time clock B'001: Channel 1 bit time clock B'010: Setting prohibited B'011: Setting prohibited B'100: Setting prohibited B'101: Setting prohibited B'110: Setting prohibited B'111: Setting prohibited |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 12 | TSSS | B'0 | R/W | <p>Timestamp Source Select</p> <p>This bit is used to select a clock source of the timestamp counter. If there is no channel which handles the classical CAN frame only, select pclk.</p> <p>0: pclk/2*1 1: Normal bit time clock</p> |
| 11 to 8 | TSP[3:0] | B'0000 | R/W | <p>Timestamp Clock Source Division</p> <p>A clock obtained by dividing the clock source selected with the TSBTCS[2:0] bits and TSSS bit according to the TSP[3:0] bits is used as the timestamp counter count source.</p> <p>B'0000: Not divided B'0001: Divided by 2 B'0010: Divided by 4 B'0011: Divided by 8 B'0100: Divided by 16 B'0101: Divided by 32 B'0110: Divided by 64 B'0111: Divided by 128 B'1000: Divided by 256 B'1001: Divided by 512 B'1010: Divided by 1024 B'1011: Divided by 2048 B'1100: Divided by 4096 B'1101: Divided by 8192 B'1110: Divided by 16384 B'1111: Divided by 32768</p> |
| 7, 6 | Reserved | All 0 | R | <p>These bits are read as the value after reset. The write value should be the value after reset.</p> |
| 5 | CMPOC | B'0 | R/W | <p>Payload Overflow Mode Select</p> <p>This bit is used to select the operation when the payload length of the received message exceeds the size for storing the payload in the storage buffer.</p> <p>When this bit is 0, the received message that the payload overflows is not stored in the buffer.</p> <p>When this bit is 1, the received message that the payload overflows is stored in the buffer. The payload which exceeds the size for storing the payload in the buffer is truncated.</p> <p>The size for storing the payload in the buffer is specified with the following bits.</p> <ul style="list-style-type: none"> • Receive buffer: RMPLS[1:0] bits in the RSCFDnCFDRMNB register • Receive FIFO buffer: RFPLS[2:0] bits in the RSCFDnCFDRFCCx register • Transmit/receive FIFO buffer: CFPLS[2:0] bits in the RSCFDnCFDCFCCk register <p>0: Messages are not stored. 1: Messages are stored and the payload which exceeds the buffer size is truncated.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 4 | DCS | B'0 | R/W | <p>CAN Clock Source Select*2</p> <p>When this bit is set to 0, clk is used as the clock source of the CAN clock (fCAN).</p> <p>When this bit is set to 1, clk_xincan is used as the clock source of the CAN clock (fCAN).</p> <p>For the CAN clock frequency settings, see section 48.16.1.3, Setting Communication Speed.</p> <p>0: clk 1: clk_xincan</p> |
| 3 | MME | B'0 | R/W | <p>Mirror Function Enable</p> <p>Setting this bit to 1 makes the mirror function available.</p> <p>0: Mirror function is disabled. 1: Mirror function is enabled.</p> |
| 2 | DRE | B'0 | R/W | <p>DLC Replacement Enable</p> <p>When the DRE bit is set to 1, the DLC value of the receive rule is stored in the buffer instead of the DLC value of the received message after the DLC value has passed through the DLC filter. In this case, a value of H'00 is stored in each data byte beyond the DLC value of the receive rule.</p> <p>The DLC replacement function is only available when the DCE bit is set to 1 (DLC check is enabled).</p> <p>0: DLC replacement is disabled. 1: DLC replacement is enabled.</p> |
| 1 | DCE | B'0 | R/W | <p>DLC Check Enable</p> <p>Setting this bit to 1 makes the DLC check function available. When disabling the DLC check function, set the GAFLDLC[3:0] bits in the RSCFDnCFDGAFLP0_j register to B'0000 before clearing the DCE bit in the RSCFDnCFDGCFG register to 0.</p> <p>0: DLC check is disabled. 1: DLC check is enabled.</p> |
| 0 | TPRI | B'0 | R/W | <p>Transmit Priority Select</p> <p>This bit is used to set the transmit priority.</p> <p>When this bit is set to 0, ID priority is selected and the transmit priority complies with the CAN bus arbitration rule (ISO11898-1 specifications). When this bit is set to 1, transmit buffer number priority is selected and the lowest transmit buffer number of those has the highest priority.</p> <p>While the transmit queue is in use, this bit should be set to 0.</p> <p>0: ID priority 1: Transmit buffer number priority</p> |

Notes: 1. When specifying pclk/2 as the timestamp counter count source, set bits TSBTCS[2:0] to B'000.
2. For the CAN clock frequency settings, see section 48.16.1.3, Setting Communication Speed.

Modify the RSCFDnCFDGCFG register only in global reset mode.

48.9.3.2 RSCFDnCFDGCTR — Global Control Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGCTR register can be read/written in 32-bit units

RSCFDnCFDGCTRL, RSCFDnCFDGCTRHL registers can be read/written in 16-bit units

RSCFDnCFDGCTRLL, RSCFDnCFDGCTRLLH, RSCFDnCFDGCTRHL, RSCFDnCFDGCTRHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGCTR: <RSCFDn_base> + H'0088

RSCFDnCFDGCTRL: <RSCFDn_base> + H'0088, RSCFDnCFDGCTRHL: <RSCFDn_base> + H'008A

RSCFDnCFDGCTRLL: <RSCFDn_base> + H'0088, RSCFDnCFDGCTRLLH: <RSCFDn_base> + H'0089,

RSCFDnCFDGCTRHL: <RSCFDn_base> + H'008A, RSCFDnCFDGCTRHH: <RSCFDn_base> + H'008B

Value after reset: H'0000_0005

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|---------|--------|------|------|----|----|----|----|----|-------|-----------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | TSRST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | CMPOFIE | THLEIE | MEIE | DEIE | — | — | — | — | — | GSLPR | GMDC[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 17 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 16 | TSRST | B'0 | R/W | Timestamp Counter Reset This bit is used to reset the timestamp counter. When this bit is set to 1, the RSCFDnCFDGTSC register is cleared to H'0000. This bit is always read as 0. |
| 15 to 12 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 11 | CMPOFIE | B'0 | R/W | Payload Overflow Interrupt Enable When the CMPOFIE bit is set to 1 and the CMPOF flag in the RSCFDnCFDGERFL register becomes 1, an interrupt request is generated. Modify this bit only in global reset mode. 0: Payload overflow interrupt is disabled. 1: Payload overflow interrupt is enabled. |
| 10 | THLEIE | B'0 | R/W | Transmit History Buffer Overflow Interrupt Enable When the THLEIE bit is set to 1 and the THLES flag in the RSCFDnCFDGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode. 0: Transmit history buffer overflow interrupt is disabled. 1: Transmit history buffer overflow interrupt is enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|--|
| 9 | MEIE | B'0 | R/W | <p>FIFO Message Lost Interrupt Enable</p> <p>When the MEIE bit is set to 1 and the MES flag in the RSCFDnCFDGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.</p> <p>0: FIFO message lost interrupt is disabled. 1: FIFO message lost interrupt is enabled.</p> |
| 8 | DEIE | B'0 | R/W | <p>DLC Error Interrupt Enable</p> <p>When the DEIE bit is set to 1 and the DEF flag in the RSCFDnCFDGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.</p> <p>0: DLC error interrupt is disabled. 1: DLC error interrupt is enabled.</p> |
| 7 to 3 | Reserved | All 0 | R | <p>When read, the value after reset is returned.</p> <p>When writing to these bits, write the value after reset.</p> |
| 2 | GSLPR | B'1 | R/W | <p>Global Stop Mode</p> <p>Setting this bit to 1 places the RS-CANFD module into global stop mode.</p> <p>Clearing this bit to 0 makes the RS-CANFD module leave from global stop mode.</p> <p>This bit should not be modified in global operating mode or global test mode.</p> <p>0: Not in global stop mode 1: In global stop mode</p> |
| 1, 0 | GMDC[1:0] | B'01 | R/W | <p>Global Mode Select</p> <p>These bits are used to select the mode of entire RS-CANFD module (global operating mode, global reset mode, or global test mode). For details, see section 48.11.1, Global Modes. Setting the GSLPR bit to 1 when in global reset mode places the RS-CANFD module into global stop mode.</p> <p>B'00: Global operating mode B'01: Global reset mode B'10: Global test mode B'11: Setting prohibited</p> |

48.9.3.3 RSCFDnCFDGSTS — Global Status Register

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

Access: RSCFDnCFDGSTS register can be read only in 32-bit units

RSCFDnCFDGSTSL, RSCFDnCFDGSTSH registers can be read only in 16-bit units

RSCFDnCFDGSTSL, RSCFDnCFDGSTSLH, RSCFDnCFDGSTSHL, RSCFDnCFDGSTSHH registers can be read only in 8-bit units

Address: RSCFDnCFDGSTS: <RSCFDn_base> + H'008C

RSCFDnCFDGSTSL: <RSCFDn_base> + H'008C, RSCFDnCFDGSTSH: <RSCFDn_base> + H'008E

RSCFDnCFDGSTSL: <RSCFDn_base> + H'008C, RSCFDnCFDGSTSLH: <RSCFDn_base> + H'008D,

RSCFDnCFDGSTSHL: <RSCFDn_base> + H'008E, RSCFDnCFDGSTSHH: <RSCFDn_base> + H'008F

Value after reset: H'0000_000D

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------------|----|----|----|----|----|----|---|---|---|---|---|---|----------|---------|---------|---------|
| | — | — | — | — | — | — | — | — | — | — | — | — | GRAMINIT | GSLPSTS | GHLTSTS | GRSTSTS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 4 | Reserved | All 0 | R | When read, the value after reset is returned. |
| 3 | GRAMINIT | B'1 | R | <p>CAN RAM Initialization Status Flag</p> <p>This flag indicates the initialization status of the CAN RAM.</p> <p>This flag is set to 1 after the MCU has been reset, and is cleared to 0 when CAN RAM initialization is completed.</p> <p>0: CAN RAM initialization is completed.</p> <p>1: CAN RAM initialization is ongoing.</p> |
| 2 | GSLPSTS | B'1 | R | <p>Global Stop Status Flag</p> <p>This flag is set to 1 when the RS-CANFD module has transitioned to global stop mode, and is cleared to 0 when the RS-CANFD module has returned from global stop mode.</p> <p>0: Not in global stop mode</p> <p>1: In global stop mode</p> |
| 1 | GHLTSTS | B'0 | R | <p>Global Test Status Flag</p> <p>This flag is set to 1 when the RS-CANFD module has transitioned to global test mode, and is cleared to 0 when the RS-CANFD module has exited global test mode.</p> <p>0: Not in global test mode</p> <p>1: In global test mode</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | GRSTSTS | B'1 | R | <p>Global Reset Status Flag</p> <p>This flag is set to 1 when the RS-CANFD module has transitioned to global reset mode, and is cleared to 0 when the RS-CANFD module has exited global reset mode. This flag remains 1 even when the RS-CANFD module has transitioned from global reset mode to global stop mode.</p> <p>0: Not in global reset mode 1: In global reset mode</p> |

48.9.3.4 RSCFDnCFDGERFL — Global Error Flag Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGERFL register can be read/written in 32-bit units

RSCFDnCFDGERFLL, RSCFDnCFDGERFLH registers can be read/written in 16-bit units

RSCFDnCFDGERFLLL, RSCFDnCFDGERFLLH, RSCFDnCFDGERFLHL, RSCFDnCFDGERFLHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGERFL: <RSCFDn_base> + H'0090

RSCFDnCFDGERFLL: <RSCFDn_base> + H'0090, RSCFDnCFDGERFLH: <RSCFDn_base> + H'0092

RSCFDnCFDGERFLLL: <RSCFDn_base> + H'0090, RSCFDnCFDGERFLLH: <RSCFDn_base> + H'0091,

RSCFDnCFDGERFLHL: <RSCFDn_base> + H'0092, RSCFDnCFDGERFLHH: <RSCFDn_base> + H'0093

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|-----------|-------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | EEF1 | EEF0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W* | R/W* |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CMPO F | THLES | MES | DEF |
| Initial value: | 0 | 0 | — | — | — | — | — | — | 0 | 0 | — | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W* | R | R | R/W* |

Note: * The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 18 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 17 | EEF1 | B'0 | R/W | ECC Error Flag for Channel 1 The EEF1 flag is set to 1 disabling message transmission if an ECC2 bit error is detected in transmit priority selection of channel 1. To clear this flag to 0, the program must write 0 to the flag. 0: No 2-bit ECC error has occurred in transmit priority selection. 1: A 2-bit ECC error has occurred in transmit priority selection. |
| 16 | EEF0 | B'0 | R/W | ECC Error Flag for Channel 0 The EEF0 flag is set to 1 disabling message transmission if an ECC2 bit error is detected in transmit priority selection of channel 0. To clear this flag to 0, the program must write 0 to the flag. 0: No 2-bit ECC error has occurred in transmit priority selection. 1: A 2-bit ECC error has occurred in transmit priority selection. |
| 15, 14 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 13 to 8 | Reserved | Undefined | R | When read, an undefined value is returned. When writing to these bits, write the value after reset. |
| 7, 6 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 5 | Reserved | Undefined | R | When read, an undefined value is returned. When writing to this bit, write the value after reset. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 4 | Reserved | B'0 | R | When read, the value after reset is returned. When writing to this bit, write the value after reset. |
| 3 | CMPOF | B'0 | R/W | Payload Overflow Flag If a payload overflow occurs in channel m (m = 0 or 1), the CMPOF flag is set to 1. To clear this flag to 0, the program must write 0 to the flag. 0: No payload overflow occurs. 1: A payload overflow occurs. |
| 2 | THLES | B'0 | R | Transmit History Buffer Overflow Status Flag The THLES flag is set to 1 when any one of the THLELT flags in the RSCFDnCFDTHLSTSm register (m = 0, 1) is set to 1. This flag is cleared to 0 when the THLELT flags of all channels are set to 0. 0: No transmit history buffer overflow has occurred. 1: A transmit history buffer overflow has occurred. |
| 1 | MES | B'0 | R | FIFO Message Lost Status Flag The MES flag is set to 1 when any one of the RFMLT flags in the RSCFDnCFDRFSTSx register (x = 0 to 7) or the CFMLT flags in the RSCFDnCFDCFSTSk register (k = 0 to 5) is set to 1. This flag is cleared to 0 when all RFMLT flags and CFMLT flags are set to 0. 0: No FIFO message lost error has occurred. 1: A FIFO message lost error has occurred. |
| 0 | DEF | B'0 | R/W | DLC Error Flag The DEF flag is set to 1 when an error has been detected during the DLC check. The program can clear this flag by writing 0 to this bit. 0: No DLC error has occurred. 1: A DLC error has occurred. |

Note: To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1".

All flags in the RSCFDnCFDGERFL register are cleared to 0 in global reset mode.

48.9.3.5 RSCFDnCFDGTSC — Global Timestamp Counter Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGTSC register can be read only in 32-bit units

RSCFDnCFDGTSC, RSCFDnCFDGTSC registers can be read only in 16-bit units

Address: RSCFDnCFDGTSC: <RSCFDn_base> + H'0094

RSCFDnCFDGTSC: <RSCFDn_base> + H'0094, RSCFDnCFDGTSC: <RSCFDn_base> + H'0096

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TS[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 15 to 0 | TS[15:0] | H'0000 | R | <p>Timestamp Value</p> <p>The timestamp counter value can be read.</p> <p>Counter Value: H'0000 to H'FFFF</p> <p>When the TS[15:0] bits are read, the read value shows the timestamp counter (16-bit free-running counter) value at that time. When the SOF is detected, the TS[15:0] value is captured and later stored in the receive buffer or the FIFO buffer. When the TMTSCE bit in the RSCFDnGCFG register is 1, the TS[15:0] value is stored in the transmit history buffer. The timestamp counter is initialized in global reset mode.</p> <p>The timestamp counter starts and stops counting differently, depending on the count source.</p> <ul style="list-style-type: none"> When the TSSS bit in the RSCFDnCFDGCFCFG register is 0 (pclk): <ul style="list-style-type: none"> The timestamp counter starts counting when the RS-CANFD module has transitioned to global operating mode. This counter stops counting when the RS-CANFD module has transitioned to global stop mode or global test mode. When the TSSS bit is 1 (CANm normal bit time clock): <ul style="list-style-type: none"> The timestamp counter starts counting when the corresponding channel has transitioned to channel communication mode. This counter stops counting when the corresponding channel has transitioned to channel reset mode or channel halt mode. |

48.9.3.6 RSCFDnCFDGTINTSTS0 — Global TX Interrupt Status Register 0

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGTINTSTS0 register can be read only in 32-bit units

RSCFDnCFDGTINTSTS0L, RSCFDnCFDGTINTSTS0H registers can be read only in 16-bit units

RSCFDnCFDGTINTSTS0LL, RSCFDnCFDGTINTSTS0LH, RSCFDnCFDGTINTSTS0HL,

RSCFDnCFDGTINTSTS0HH registers can be read only in 8-bit units

Address: RSCFDnCFDGTINTSTS0: <RSCFDn_base> + H'0460

RSCFDnCFDGTINTSTS0L: <RSCFDn_base> + H'0460,

RSCFDnCFDGTINTSTS0H: <RSCFDn_base> + H'0462

RSCFDnCFDGTINTSTS0LL: <RSCFDn_base> + H'0460,

RSCFDnCFDGTINTSTS0LH: <RSCFDn_base> + H'0461,

RSCFDnCFDGTINTSTS0HL: <RSCFDn_base> + H'0462,

RSCFDnCFDGTINTSTS0HH: <RSCFDn_base> + H'0463

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-------|--------|-------|-------|-------|----|----|----|-------|--------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R* | R* | R* | R* | R* | R | R | R | R* | R* | R* | R* | R* |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | THIF1 | CFTIF1 | TQIF1 | TAIF1 | TSIF1 | — | — | — | THIF0 | CFTIF0 | TQIF0 | TAIF0 | TSIF0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R* | R* | R* | R* | R* | R | R | R | R* | R* | R* | R* | R* |

Note: * This bit is automatically cleared in the global reset or channel reset mode.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 13 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 12 | THIF1 | B'0 | R | <p>Channel 1 Transmit History Interrupt Status Flag</p> <p>When the THLIE bit in the RSCFDnCFDTHLCCm register is set to 1 (transmit history interrupt enabled) and the THLIF bit in the RSCFDnCFDTHLSTSm register is set to 1 (transmit history interrupt request), this bit is set to 1.</p> <p>When the THLIF bit in the RSCFDnCFDTHLSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the THLIE bit is cleared to 0.</p> <p>0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.</p> |
| 11 | CFTIF1 | B'0 | R | <p>Channel 1 Transmit/receive FIFO Transmit Interrupt Status Flag</p> <p>When the CFTXIE bit in the RSCFDnCFDCFCCK register is set to 1 (transmit/receive FIFO transmit interrupt enabled) and the CFTXIF bit in the RSCFDnCFDCFSTSk register is set to 1 (transmit/receive FIFO transmit interrupt request), this bit is set to 1.</p> <p>When the CFTXIF bit is cleared to 0 under the conditions that this bit can be set to 1, this bit is cleared to 0. This flag is also cleared to 0 when the CFTXIE bit is cleared to 0.</p> <p>0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 10 | TQIF1 | B'0 | R | <p>Channel 1 Transmit Queue Interrupt Status Flag</p> <p>When the TXQIE bit in the RSCFDnTXQCCm register is set to 1 (transmit queue interrupt enabled) and the TXQIF bit in the RSCFDnCFDnTXQSTSm register is set to 1 (transmit queue interrupt request), this bit is set to 1.</p> <p>When the TXQIF bit (transmit queue interrupt request) in the RSCFDnCFDnTXQSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the TXQIE bit is cleared to 0.</p> <p>0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.</p> |
| 9 | TAIF1 | B'0 | R | <p>Channel 1 Transmit Buffer Abort Interrupt Status Flag</p> <p>This bit is set to 1 when the TAIE bit in the RSCFDnCFDCmCTR register is 1 (transmit abort interrupt enabled) and the TMTRF[1:0] flags in the RSCFDnCFDTMSTSp register are set to B'01 (transmit abort completed).</p> <p>This flag is cleared to 0 when the TMTRF[1:0] flags are cleared to B'00 after the transmit abort is completed.</p> <p>0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.</p> |
| 8 | TSIF1 | B'0 | R | <p>Channel 1 Transmit Buffer Interrupt Status Flag</p> <p>This bit is set to 1 when the TMIEp bit in the RSCFDnCFDTMIECy register is set to 1 (transmit buffer interrupt enabled) and the TMTRF[1:0] flags in the RSCFDnCFDTMSTSp register are set to B'10 (transmit completed without abort request) or B'11 (transmit completed with abort request).</p> <p>When the TMTRF[1:0] flags are cleared to B'00 under the condition that this bit can be set to 1, this flag is cleared to 0. In addition, clearing the TMIEp bit to 0 also clears this flag to 0.</p> <p>0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.</p> |
| 7 to 5 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 4 | THIF0 | B'0 | R | <p>Channel 0 Transmit History Interrupt Status Flag</p> <p>When the THLIE bit in the RSCFDnCFDnTHLCCm register is set to 1 (transmit history interrupt enabled) and the THLIF bit in the RSCFDnCFDnTHLSTSm register is set to 1 (transmit history interrupt request), this bit is set to 1.</p> <p>When the THLIF bit in the RSCFDnCFDnTHLSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the THLIE bit is cleared to 0.</p> <p>0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.</p> |
| 3 | CFTIF0 | B'0 | R | <p>Channel 0 Transmit/receive FIFO Transmit Interrupt Status Flag</p> <p>When the CFTXIE bit in the RSCFDnCFDCFCCK register is set to 1 (transmit/receive FIFO transmit interrupt enabled) and the CFTXIF bit in the RSCFDnCFDCFSTSk register is set to 1 (transmit/receive FIFO transmit interrupt request), this bit is set to 1.</p> <p>When the CFTXIF bit is cleared to 0 under the conditions that this bit can be set to 1, this bit is cleared to 0. This flag is also cleared to 0 when the CFTXIE bit is cleared to 0.</p> <p>0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | TQIF0 | B'0 | R | <p>Channel 0 Transmit Queue Interrupt Status Flag</p> <p>When the TXQIE bit in the RSCFDnCFDnTXQCCm register is set to 1 (transmit queue interrupt enabled) and the TXQIF bit in the RSCFDnCFDnTXQSTSm register is set to 1 (transmit queue interrupt request), this bit is set to 1.</p> <p>When the TXQIF bit (transmit queue interrupt request) in the RSCFDnCFDnTXQSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the TXQIE bit is cleared to 0.</p> <p>0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.</p> |
| 1 | TAIF0 | B'0 | R | <p>Channel 0 Transmit Buffer Abort Interrupt Status Flag</p> <p>This bit is set to 1 when the TAIE bit in the RSCFDnCFDnCTR register is 1 (transmit abort interrupt enabled) and the TMTRF[1:0] flags in the RSCFDnCFDnMSTSp register are set to B'01 (transmit abort completed).</p> <p>This flag is cleared to 0 when the TMTRF[1:0] flags are cleared to B'00 after the transmit abort is completed.</p> <p>0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.</p> |
| 0 | TSIF0 | B'0 | R | <p>Channel 0 Transmit Buffer Interrupt Status Flag</p> <p>This bit is set to 1 when the TMIEp bit in the RSCFDnCFDnMIECy register is set to 1 (transmit buffer interrupt enabled) and the TMTRF[1:0] flags in the RSCFDnCFDnMSTSp register are set to B'10 (transmit completed without abort request) or B'11 (transmit completed with abort request).</p> <p>When the TMTRF[1:0] flags are cleared to B'00 under the condition that this bit can be set to 1, this flag is cleared to 0. In addition, clearing the TMIEp bit to 0 also clears this flag to 0.</p> <p>0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.</p> |

48.9.3.7 RSCFDnCFDGTINTSTS1 — Global TX Interrupt Status Register 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGTINTSTS1 register can be read only in 32-bit units
 RSCFDnCFDGTINTSTS1L, RSCFDnCFDGTINTSTS1H registers can be read only in 16-bit units
 RSCFDnCFDGTINTSTS1LL, RSCFDnCFDGTINTSTS1LH, RSCFDnCFDGTINTSTS1HL,
 RSCFDnCFDGTINTSTS1HH registers can be read only in 8-bit units

Address: RSCFDnCFDGTINTSTS1: <RSCFDn_base> + H'0464
 RSCFDnCFDGTINTSTS1L: <RSCFDn_base> + H'0464,
 RSCFDnCFDGTINTSTS1H: <RSCFDn_base> + H'0466
 RSCFDnCFDGTINTSTS1LL: <RSCFDn_base> + H'0464,
 RSCFDnCFDGTINTSTS1LH: <RSCFDn_base> + H'0465,
 RSCFDnCFDGTINTSTS1HL: <RSCFDn_base> + H'0466,
 RSCFDnCFDGTINTSTS1HH: <RSCFDn_base> + H'0467

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R* | R* | R* | R* | R* | R | R | R | R* | R* | R* | R* | R* |

Note: * This bit is automatically cleared in the global reset or channel reset mode.

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | Reserved | All 0 | R | These bits are read as the value after reset. |

48.9.4 Details of Receive Rule Related Registers

48.9.4.1 RSCFDnCFDGAFLECTR — Receive Rule Entry Control Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGAFLECTR register can be read/written in 32-bit units

RSCFDnCFDGAFLECTRL, RSCFDnCFDGAFLECTRH registers can be read/written in 16-bit units

RSCFDnCFDGAFLECTRLL, RSCFDnCFDGAFLECTRLH, RSCFDnCFDGAFLECTRHL,

RSCFDnCFDGAFLECTRHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGAFLECTR: <RSCFDn_base> + H'0098

RSCFDnCFDGAFLECTRL: <RSCFDn_base> + H'0098,

RSCFDnCFDGAFLECTRH: <RSCFDn_base> + H'009A

RSCFDnCFDGAFLECTRLL: <RSCFDn_base> + H'0098,

RSCFDnCFDGAFLECTRLH: <RSCFDn_base> + H'0099,

RSCFDnCFDGAFLECTRHL: <RSCFDn_base> + H'009A,

RSCFDnCFDGAFLECTRHH: <RSCFDn_base> + H'009B

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|--------|----|----|----|------------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | AFLDAE | — | — | — | AFLPN[4:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 9 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 8 | AFLDAE | B'0 | R/W | Receive Rule Table Write Enable Setting this bit to 0 disables the write to the receive rule table. After writes to the receive rule table are completed, set this bit to 0 to disable the write to the table. The receive rule table can be read regardless of the value of this bit. Set the AFLDAE bit to 1 only in global reset mode. 0: Receive rule table write is disabled. 1: Receive rule table write is enabled. |
| 7 to 5 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 4 to 0 | AFLPN[4:0] | B'0_0000 | R/W | Receive Rule Table Page Number Configuration These bits are used to set the page number of the receive rule table. Sixteen receive rules can be set per page. A page number can be selected from a range of page 0 (B'0_0000) to page 23 (B'1_0111). Set these bits to a value within the range of B'0_0000 to B'1_0111. |

48.9.4.2 RSCFDnCFDGAFLCFG0 — Receive Rule Configuration Register 0

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGAFLCFG0 register can be read/written in 32-bit units

RSCFDnCFDGAFLCFG0L, RSCFDnCFDGAFLCFG0H registers can be read/written in 16-bit units

RSCFDnCFDGAFLCFG0LL, RSCFDnCFDGAFLCFG0LH, RSCFDnCFDGAFLCFG0HL,

RSCFDnCFDGAFLCFG0HH registers can be read/written in 8-bit units

Address: RSCFDnCFDGAFLCFG0: <RSCFDn_base> + H'009C

RSCFDnCFDGAFLCFG0L: <RSCFDn_base> + H'009C,

RSCFDnCFDGAFLCFG0H: <RSCFDn_base> + H'009E

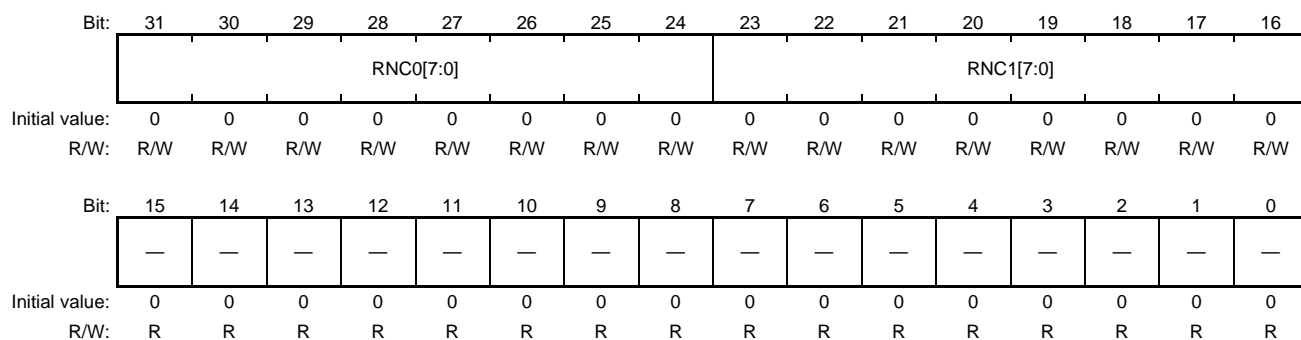
RSCFDnCFDGAFLCFG0LL: <RSCFDn_base> + H'009C,

RSCFDnCFDGAFLCFG0LH: <RSCFDn_base> + H'009D,

RSCFDnCFDGAFLCFG0HL: <RSCFDn_base> + H'009E,

RSCFDnCFDGAFLCFG0HH: <RSCFDn_base> + H'009F

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 24 | RNC0[7:0] | H'00 | R/W | Number of Rules for Channel 0 These bits are used to set the number of rules to be registered in the channel 0 receive rule table. Set these bits to a value within the range of H'00 to H'80. |
| 23 to 16 | RNC1[7:0] | H'00 | R/W | Number of Rules for Channel 1 These bits are used to set the number of rules to be registered in the channel 1 receive rule table. Set these bits to a value within the range of H'00 to H'80. |
| 15 to 0 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |

Modify the RSCFDnCFDGAFLCFG0 register only in global reset mode.

Up to $64 \times$ (number of channels) rules can be registered in the receive rule table as the entire unit. The number of receive rules per channel should meet the following conditions.

- The maximum number of rules per channel is 128.
- The total of the number of rules allocated to each channel is not larger than the number of rules that can be registered in the entire unit.

48.9.4.3 RSCFDnCFDGAFLCFG1 — Receive Rule Configuration Register 1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGAFLCFG1 register can be read/written in 32-bit units

RSCFDnCFDGAFLCFG1L, RSCFDnCFDGAFLCFG1H registers can be read/written in 16-bit units

RSCFDnCFDGAFLCFG1LL, RSCFDnCFDGAFLCFG1LH, RSCFDnCFDGAFLCFG1HL,

RSCFDnCFDGAFLCFG1HH registers can be read/written in 8-bit units

Address: RSCFDnCFDGAFLCFG1: <RSCFDn_base> + H'00A0H

RSCFDnCFDGAFLCFG1L: <RSCFDn_base> + H'00A0,

RSCFDnCFDGAFLCFG1H: <RSCFDn_base> + H'00A2

RSCFDnCFDGAFLCFG1LL: <RSCFDn_base> + H'00A0,

RSCFDnCFDGAFLCFG1LH: <RSCFDn_base> + H'00A1,

RSCFDnCFDGAFLCFG1HL: <RSCFDn_base> + H'00A2,

RSCFDnCFDGAFLCFG1HH: <RSCFDn_base> + H'00A3

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |

48.9.4.4 RSCFDnCFDGAFLIDj — Receive Rule ID Register (j = 0 to 15)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGAFLIDj register can be read/written in 32-bit units

RSCFDnCFDGAFLIDjL, RSCFDnCFDGAFLIDjH registers can be read/written in 16-bit units

RSCFDnCFDGAFLIDjLL, RSCFDnCFDGAFLIDjLH, RSCFDnCFDGAFLIDjHL, RSCFDnCFDGAFLIDjHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGAFLIDj: <RSCFDn_base> + H'1000 + (H'10 × j)

RSCFDnCFDGAFLIDjL: <RSCFDn_base> + H'1000 + (H'10 × j),

RSCFDnCFDGAFLIDjH: <RSCFDn_base> + H'1002 + (H'10 × j)

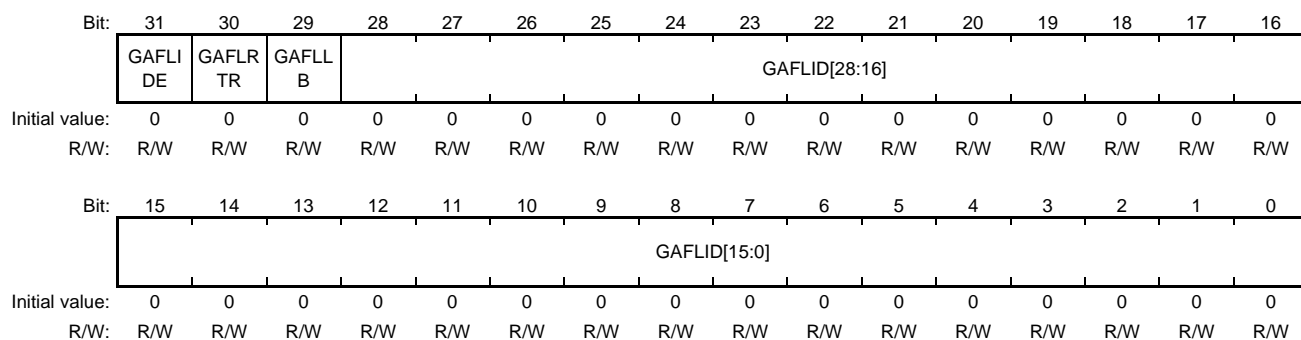
RSCFDnCFDGAFLIDjLL: <RSCFDn_base> + H'1000 + (H'10 × j),

RSCFDnCFDGAFLIDjLH: <RSCFDn_base> + H'1001 + (H'10 × j),

RSCFDnCFDGAFLIDjHL: <RSCFDn_base> + H'1002 + (H'10 × j),

RSCFDnCFDGAFLIDjHH: <RSCFDn_base> + H'1003 + (H'10 × j)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 31 | GAFLIDE | B'0 | R/W | IDE Select This bit is used to select the ID format (standard ID or extended ID) of the receive rule. This bit is compared with the IDE bit in the received message during the acceptance filter processing. 0: Standard ID 1: Extended ID |
| 30 | GAFLRTR | B'0 | R/W | RTR Select This bit is used to select the frame format (data frame or remote frame) of the receive rule. This bit is compared with the RTR bit in the received message during the acceptance filter processing. 0: Data frame 1: Remote frame |
| 29 | GAFLLB | B'0 | R/W | Receive Rule Target Message Select When this bit is set to 0, data processing using the receive rule is performed when receiving messages transmitted from another CAN node. When this bit is set to 1 when the mirror function is used, data processing using the receive rule is performed when the CAN node is receiving its own transmitted messages. 0: When a message transmitted from another CAN node is received 1: When the own transmitted message is received |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|---|
| 28 to 0 | GAFLID[28:0] | H'0000_0000 | R/W | ID Set These bits are used to set the ID field of the receive rule. The ID value set by these bits is compared with the ID of the received message during the acceptance filter processing. For the standard ID, set the ID in bits 10 to 0 and set bits 28 to 11 to 0. |

Modify the RSCFDnCFDGAFLIDj register when the AFLDAE bit in the RSCFDnCFDGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

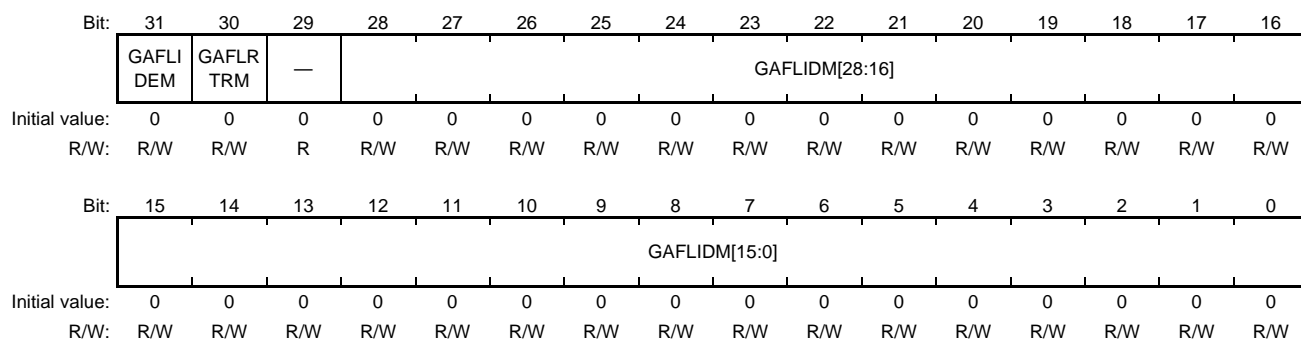
48.9.4.5 RSCFDnCFDGAFLMj — Receive Rule Mask Register (j = 0 to 15)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGAFLMj register can be read/written in 32-bit units
 RSCFDnCFDGAFLMjL, RSCFDnCFDGAFLMjH registers can be read/written in 16-bit units
 RSCFDnCFDGAFLMjLL, RSCFDnCFDGAFLMjLH, RSCFDnCFDGAFLMjHL, RSCFDnCFDGAFLMjHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGAFLMj: <RSCFDn_base> + H'1004 + (H'10 × j)
 RSCFDnCFDGAFLMjL: <RSCFDn_base> + H'1004 + (H'10 × j),
 RSCFDnCFDGAFLMjH: <RSCFDn_base> + H'1006 + (H'10 × j)
 RSCFDnCFDGAFLMjLL: <RSCFDn_base> + H'1004 + (H'10 × j),
 RSCFDnCFDGAFLMjLH: <RSCFDn_base> + H'1005 + (H'10 × j),
 RSCFDnCFDGAFLMjHL: <RSCFDn_base> + H'1006 + (H'10 × j),
 RSCFDnCFDGAFLMjHH: <RSCFDn_base> + H'1007 + (H'10 × j)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|---|
| 31 | GAFLIDEM | B'0 | R/W | <p>IDE Mask</p> <p>When this bit is set to 1, filter processing is performed only for messages of the ID format specified by the GAFLIDE bit in the RSCFDnCFDGAFLIDj register.</p> <p>When this bit is cleared to 0, the IDs of all the receive messages and the specified IDs are regarded as matched. To set the GAFLIDEM bit to 0, set the GAFLIDM[28:0] bits to all 0 at the same time.</p> <p>0: The IDE bit is not compared. 1: The IDE bit is compared.</p> |
| 30 | GAFLRTRM | B'0 | R/W | <p>RTR Mask</p> <p>This bit is used to mask the RTR bit of the receive rule.</p> <p>0: The RTR bit is not compared. 1: The RTR bit is compared</p> |
| 29 | Reserved | B'0 | R | <p>This bit is read as the value after reset. The write value should be the value after reset.</p> |
| 28 to 0 | GAFLIDM [28:0] | H'0000_0000 | R/W | <p>ID Mask</p> <p>These bits are used to mask the corresponding ID bit of the receive rule.</p> <p>0: The corresponding ID bit is not compared. 1: The corresponding ID bit is compared.</p> |

Modify the RSCFDnCFDGAFLMj register when the AFLDAE bit in the RSCFDnCFDGAFLLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

48.9.4.6 RSCFDnCFDGAFLP0_j — Receive Rule Pointer 0 Register (j = 0 to 15)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGAFLP0_j register can be read/written in 32-bit units

RSCFDnCFDGAFLP0_jL, RSCFDnCFDGAFLP0_jH registers can be read/written in 16-bit units

RSCFDnCFDGAFLP0_jLL, RSCFDnCFDGAFLP0_jLH, RSCFDnCFDGAFLP0_jHL, RSCFDnCFDGAFLP0_jHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGAFLP0_j: <RSCFDn_base> + H'1008 + (H'10 × j)

RSCFDnCFDGAFLP0_jL: <RSCFDn_base> + H'1008 + (H'10 × j),

RSCFDnCFDGAFLP0_jH: <RSCFDn_base> + H'100A + (H'10 × j)

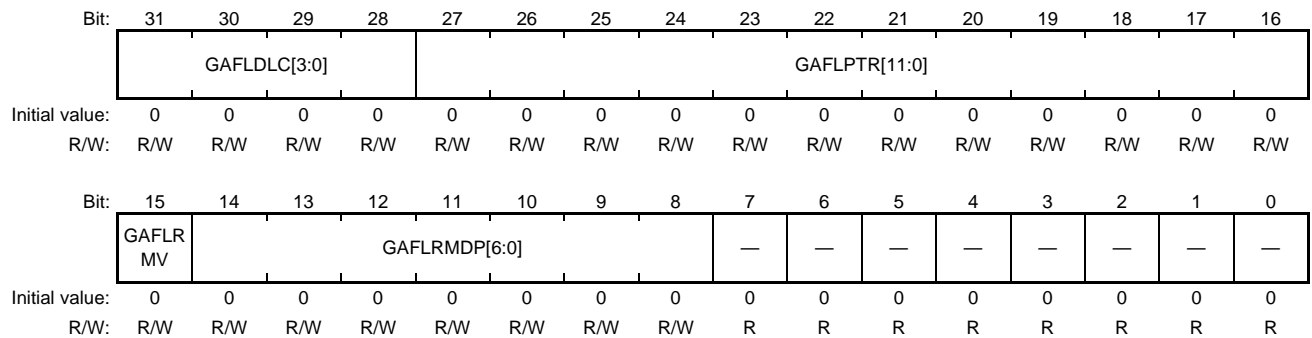
RSCFDnCFDGAFLP0_jLL: <RSCFDn_base> + H'1008 + (H'10 × j),

RSCFDnCFDGAFLP0_jLH: <RSCFDn_base> + H'1009 + (H'10 × j),

RSCFDnCFDGAFLP0_jHL: <RSCFDn_base> + H'100A + (H'10 × j),

RSCFDnCFDGAFLP0_jHH: <RSCFDn_base> + H'100B + (H'10 × j)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------|---------------------|---------------|-----|--|--|---------------------|--------------|---------|--------------|--|---------|-------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|
| 31 to 28 | GAFLDLC [3:0] | B'0000 | R/W | <p>Receive Rule DLC</p> <p>These bits are used to set the minimum data length necessary for receiving messages. If the data length of a message that is being filtered is equal to or larger than the value set by the GAFLDLC[3:0] bits, the message passes the DLC check. Setting these bits to B'0000 disables the DLC check function allowing messages with any data length to pass the DLC check.</p> <table border="1"> <thead> <tr> <th></th> <th>Classical CAN Frame</th> <th>CAN FD Frame</th> </tr> </thead> <tbody> <tr> <td>B'0000:</td> <td>0 data bytes</td> <td></td> </tr> <tr> <td>B'0001:</td> <td>1 data byte</td> <td></td> </tr> <tr> <td>B'0010:</td> <td>2 data bytes</td> <td></td> </tr> <tr> <td>B'0011:</td> <td>3 data bytes</td> <td></td> </tr> <tr> <td>B'0100:</td> <td>4 data bytes</td> <td></td> </tr> <tr> <td>B'0101:</td> <td>5 data bytes</td> <td></td> </tr> <tr> <td>B'0110:</td> <td>6 data bytes</td> <td></td> </tr> <tr> <td>B'0111:</td> <td>7 data bytes</td> <td></td> </tr> <tr> <td>B'1000:</td> <td>8 data bytes</td> <td></td> </tr> <tr> <td>B'1001:</td> <td>8 data bytes</td> <td>12 data bytes</td> </tr> <tr> <td>B'1010:</td> <td></td> <td>16 data bytes</td> </tr> <tr> <td>B'1011:</td> <td></td> <td>20 data bytes</td> </tr> <tr> <td>B'1100:</td> <td></td> <td>24 data bytes</td> </tr> <tr> <td>B'1101:</td> <td></td> <td>32 data bytes</td> </tr> <tr> <td>B'1110:</td> <td></td> <td>48 data bytes</td> </tr> <tr> <td>B'1111:</td> <td></td> <td>64 data bytes</td> </tr> </tbody> </table> | | Classical CAN Frame | CAN FD Frame | B'0000: | 0 data bytes | | B'0001: | 1 data byte | | B'0010: | 2 data bytes | | B'0011: | 3 data bytes | | B'0100: | 4 data bytes | | B'0101: | 5 data bytes | | B'0110: | 6 data bytes | | B'0111: | 7 data bytes | | B'1000: | 8 data bytes | | B'1001: | 8 data bytes | 12 data bytes | B'1010: | | 16 data bytes | B'1011: | | 20 data bytes | B'1100: | | 24 data bytes | B'1101: | | 32 data bytes | B'1110: | | 48 data bytes | B'1111: | | 64 data bytes |
| | Classical CAN Frame | CAN FD Frame | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0000: | 0 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0001: | 1 data byte | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0010: | 2 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0011: | 3 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0100: | 4 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0101: | 5 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0110: | 6 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0111: | 7 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1000: | 8 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1001: | 8 data bytes | 12 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1010: | | 16 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1011: | | 20 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1100: | | 24 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1101: | | 32 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1110: | | 48 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1111: | | 64 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 27 to 16 | GAFLPTR [11:0] | H'000 | R/W | <p>Receive Rule Label</p> <p>Set the 12-bit label information.</p> <p>These bits are used to set a 12-bit label to be attached to messages that have passed through the filter. A label is attached when a message is stored in the receive buffer or the FIFO buffer.</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 15 | GAFLRMV | B'0 | R/W | <p>Receive Buffer Enable</p> <p>When this bit is set to 1, receive messages that have passed through the filter are stored in the receive buffer selected by the GAFLRMDP[6:0] bits.</p> <p>0: No receive buffer is used. 1: A receive buffer is used.</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 14 to 8 | GAFLRMDP [6:0] | H'00 | R/W | <p>Receive Buffer Number Select</p> <p>Set the receive buffer number to store receive messages.</p> <p>These bits are used to select the number of the receive buffer that stores receive messages that have passed through the filter when the GAFLRMV bit is set to 1. Set these bits to a value smaller than the value set by the NRXMB[7:0] bits in the RSCFDnCFDRMNB register.</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 7 to 0 | Reserved | All 0 | R | <p>These bits are read as the value after reset.</p> <p>The write value should be the value after reset.</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Modify the RSCFDnCFDGAFLP0_j register when the AFLDAE bit in the RSCFDnCFDGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

48.9.4.7 RSCFDnCFDGAFLP1_j — Receive Rule Pointer 1 Register (j = 0 to 15)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGAFLP1_j register can be read/written in 32-bit units

RSCFDnCFDGAFLP1_jL, RSCFDnCFDGAFLP1_jH registers can be read/written in 16-bit units

RSCFDnCFDGAFLP1_jLL, RSCFDnCFDGAFLP1_jLH, RSCFDnCFDGAFLP1_jHL, RSCFDnCFDGAFLP1_jHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGAFLP1_j: <RSCFDn_base> + H'100C + (H'10 × j)

RSCFDnCFDGAFLP1_jL: <RSCFDn_base> + H'100C + (H'10 × j),

RSCFDnCFDGAFLP1_jH: <RSCFDn_base> + H'100E + (H'10 × j)

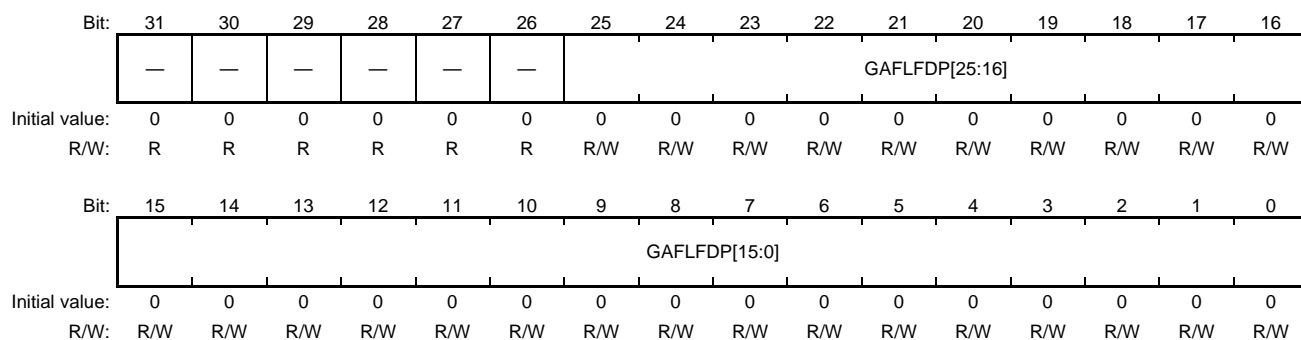
RSCFDnCFDGAFLP1_jLL: <RSCFDn_base> + H'100C + (H'10 × j),

RSCFDnCFDGAFLP1_jLH: <RSCFDn_base> + H'100D + (H'10 × j),

RSCFDnCFDGAFLP1_jHL: <RSCFDn_base> + H'100E + (H'10 × j),

RSCFDnCFDGAFLP1_jHH: <RSCFDn_base> + H'100F + (H'10 × j)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------|---------------|-----|--|
| 31 to 26 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 25 to 16 | GAFLFDP [25:16] | H'000 | R/W | Transmit/Receive FIFO Buffer k Select (Bit position – 8 = target transmit/receive FIFO buffer number k) 0: Transmit/receive FIFO buffer is not selected. 1: Transmit/receive FIFO buffer is selected. |
| 15 to 0 | GAFLFDP [15:0] | H'0000 | R/W | Receive FIFO Buffer x Select (Bit position = target receive FIFO buffer number x) 0: Receive FIFO buffer is not selected. 1: Receive FIFO buffer is selected. |

Modify the RSCFDnCFDGAFLP1_j register when the AFLDAE bit in the RSCFDnCFDGAFLP1_j register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLFDP[25:0] Bits

These bits are used to specify FIFO buffers that store receive messages that have passed through the filter. Up to eight FIFO buffers are selectable. However, when the GAFLRMV bit in the RSCFDnCFDGAFLP0_j register is set to 1 (a message is stored in the receive buffer), up to seven FIFO buffers can be selected. Only receive FIFO buffers and the transmit/receive FIFO buffer for which the CFM[1:0] bits in the RSCFDnCFDGFCCk register are set to B'00 (receive mode) or B'10 (gateway mode) are selectable.

48.9.5 Details of Receive Buffer Related Registers

48.9.5.1 RSCFDnCFDRMNB — Receive Buffer Number Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRMNB register can be read/written in 32-bit units

RSCFDnCFDRMNB, RSCFDnCFDRMNBH registers can be read/written in 16-bit units

RSCFDnCFDRMNBLL, RSCFDnCFDRMNBHL, RSCFDnCFDRMNBHLL, RSCFDnCFDRMNBHHL registers can be read/written in 8-bit units

Address: RSCFDnCFDRMNB: <RSCFDn_base> + H'00A4

RSCFDnCFDRMNB: <RSCFDn_base> + H'00A4, RSCFDnCFDRMNBH: <RSCFDn_base> + H'00A6

RSCFDnCFDRMNBLL: <RSCFDn_base> + H'00A4, RSCFDnCFDRMNBHL: <RSCFDn_base> + H'00A5,

RSCFDnCFDRMNBHLL: <RSCFDn_base> + H'00A6, RSCFDnCFDRMNBHHL: <RSCFDn_base> + H'00A7

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|------------|-----|------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | RMPLS[1:0] | | NRXMB[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 10 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 9, 8 | RMPLS[1:0] | B'00 | R/W | Receive Buffer Payload Storage Size Select Select the upper-limited size of a payload that can be stored in the receive buffer. B'00: 8 bytes B'01: 12 bytes B'10: 16 bytes B'11: 20 bytes |
| 7 to 0 | NRXMB[7:0] | H'00 | R/W | Receive Buffer Number Configuration Set the number of receive buffers. Set a value of 0 to 96. These bits are used to set the total number of receive buffers of the RS-CANFD module. The maximum value is 16 × (number of channels). Setting these bits all to 0 makes receive buffers unavailable. |

Modify the RSCFDnCFDRMNB register only in global reset mode.

48.9.5.2 RSCFDnCFDRMNDy — Receive Buffer New Data Register (y = 0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRMNDy register can be read/written in 32-bit units

RSCFDnCFDRMNDyL, RSCFDnCFDRMNDyH registers can be read/written in 16-bit units

RSCFDnCFDRMNDyLL, RSCFDnCFDRMNDyLH, RSCFDnCFDRMNDyHL, RSCFDnCFDRMNDyHH registers can be read/written in 8-bit units

Address: RSCFDnCFDRMNDy: <RSCFDn_base> + H'00A8 + (H'04 × y)

RSCFDnCFDRMNDyL: <RSCFDn_base> + H'00A8 + (H'04 × y),

RSCFDnCFDRMNDyH: <RSCFDn_base> + H'00AA + (H'04 × y)

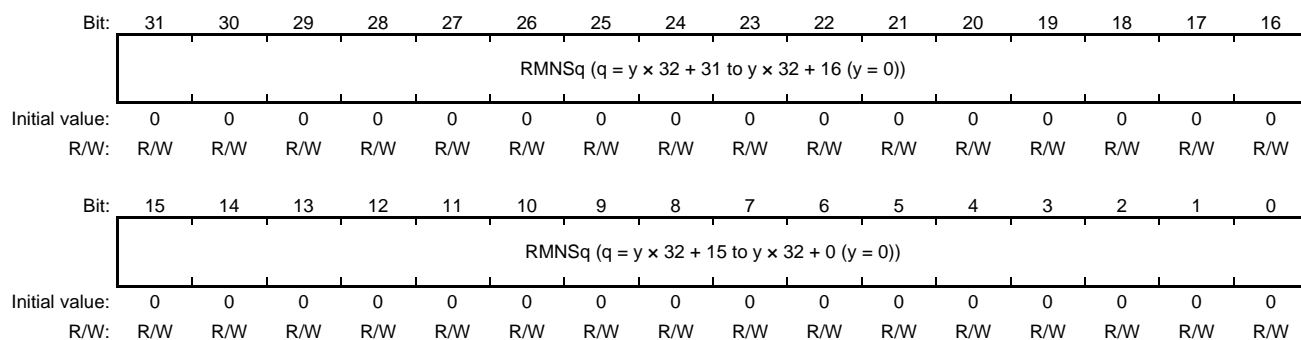
RSCFDnCFDRMNDyLL: <RSCFDn_base> + H'00A8 + (H'04 × y),

RSCFDnCFDRMNDyLH: <RSCFDn_base> + H'00A9 + (H'04 × y),

RSCFDnCFDRMNDyHL: <RSCFDn_base> + H'00AA + (H'04 × y),

RSCFDnCFDRMNDyHH: <RSCFDn_base> + H'00AB + (H'04 × y)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | RMNSq | H'0000_0000 | R/W | Receive Buffer Receive Complete Flag q (q = y × 32 + 31 to y × 32 + 0) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q. |

Write 0 to the RSCFDnCFDRMNDy register in global operating mode or global test mode.

RMNSq Flags (q = 0 to 31)

Each RMNS flag is set to 1 when the processing for storing a message in the corresponding receive buffer starts.

To clear a flag to 0, the program must write 0 to the flag. Use a store instruction to write “0” to the flag and “1” to other flags. These bits cannot be cleared to 0 while a message is being stored. Time for storing a message differs depending on the size for storing the payload in the receive buffer; it takes 12 clock cycles of pclk when the RMPLS[1:0] bits in the RSCFDnCFDRMNB register are B'00 (8 bytes) and 18 clock cycles of pclk when the RMPLS[1:0] bits are B'11 (20 bytes) (two clock cycles of pclk for four bytes of the size for storing the payload).

These flags are cleared to 0 in global reset mode.

48.9.5.3 RSCFDnCFDRMIDq — Receive Buffer ID Register (q = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRMIDq register can be read only in 32-bit units

RSCFDnCFDRMIDqL, RSCFDnCFDRMIDqH registers can be read only in 16-bit units

RSCFDnCFDRMIDqLL, RSCFDnCFDRMIDqLH, RSCFDnCFDRMIDqHL, RSCFDnCFDRMIDqHH registers can be read only in 8-bit units

Address: RSCFDnCFDRMIDq: <RSCFDn_base> + H'2000 + (H'20 × q)

RSCFDnCFDRMIDqL: <RSCFDn_base> + H'2000 + (H'20 × q),

RSCFDnCFDRMIDqH: <RSCFDn_base> + H'2002 + (H'20 × q)

RSCFDnCFDRMIDqLL: <RSCFDn_base> + H'2000 + (H'20 × q),

RSCFDnCFDRMIDqLH: <RSCFDn_base> + H'2001 + (H'20 × q),

RSCFDnCFDRMIDqHL: <RSCFDn_base> + H'2002 + (H'20 × q),

RSCFDnCFDRMIDqHH: <RSCFDn_base> + H'2003 + (H'20 × q)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----------|----|-------------|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RMIDE | RMRT R | — | RMID[28:16] | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RMID[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 | RMIDE | B'0 | R | Receive Buffer IDE This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive buffer. 0: Standard ID 1: Extended ID |
| 30 | RMRT R | B'0 | R | Receive Buffer RTR/RRS When the received message is in the classical CAN frame, this bit indicates the frame format (data frame or remote frame) of the message stored in the receive buffer. When the received message is in the CAN FD frame, this bit indicates the RRS bit value of the message. <ul style="list-style-type: none"> When the received message is in the classical CAN frame: <ul style="list-style-type: none"> 0: Data frame 1: Remote frame When the received message is in the CAN FD frame: <ul style="list-style-type: none"> The RRS bit value of the received message is read. |
| 29 | Reserved | B'0 | R | This bit is read as the value after reset. |
| 28 to 0 | RMID[28:0] | H'0000_0000 | R | Receive Buffer ID Data These bits contain the standard ID or extended ID of the received message. Read bits 10 to 0 for standard ID. Bits 28 to 11 are read as 0. |

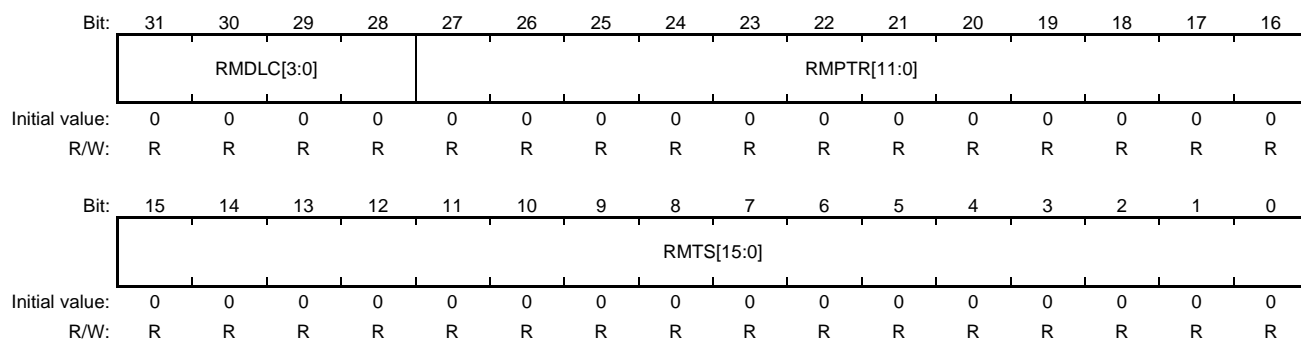
48.9.5.4 RSCFDnCFDRMPTRq — Receive Buffer Pointer Register (q = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRMPTRq register can be read only in 32-bit units
 RSCFDnCFDRMPTRqL, RSCFDnCFDRMPTRqH registers can be read only in 16-bit units
 RSCFDnCFDRMPTRqLL, RSCFDnCFDRMPTRqLH, RSCFDnCFDRMPTRqHL, RSCFDnCFDRMPTRqHH registers can be read only in 8-bit units

Address: RSCFDnCFDRMPTRq: <RSCFDn_base> + H'2004 + (H'20 × q)
 RSCFDnCFDRMPTRqL: <RSCFDn_base> + H'2004 + (H'20 × q),
 RSCFDnCFDRMPTRqH: <RSCFDn_base> + H'2006 + (H'20 × q)
 RSCFDnCFDRMPTRqLL: <RSCFDn_base> + H'2004 + (H'20 × q),
 RSCFDnCFDRMPTRqLH: <RSCFDn_base> + H'2005 + (H'20 × q),
 RSCFDnCFDRMPTRqHL: <RSCFDn_base> + H'2006 + (H'20 × q),
 RSCFDnCFDRMPTRqHH: <RSCFDn_base> + H'2007 + (H'20 × q)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------|---------------------|---------------|-----|---|--|---------------------|--------------|---------|--------------|--|---------|-------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|
| 31 to 28 | RMDLC[3:0] | B'0000 | R | Receive Buffer DLC Data These bits indicate the data length of the message stored in the receive buffer. The number of bytes of the payload stored in the receive buffer depends on the RMPLS[1:0] bits in the RSCFDnCFDRMNB register. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | <table border="1"> <thead> <tr> <th></th> <th>Classical CAN Frame</th> <th>CAN FD Frame</th> </tr> </thead> <tbody> <tr> <td>B'0000:</td> <td>0 data bytes</td> <td></td> </tr> <tr> <td>B'0001:</td> <td>1 data byte</td> <td></td> </tr> <tr> <td>B'0010:</td> <td>2 data bytes</td> <td></td> </tr> <tr> <td>B'0011:</td> <td>3 data bytes</td> <td></td> </tr> <tr> <td>B'0100:</td> <td>4 data bytes</td> <td></td> </tr> <tr> <td>B'0101:</td> <td>5 data bytes</td> <td></td> </tr> <tr> <td>B'0110:</td> <td>6 data bytes</td> <td></td> </tr> <tr> <td>B'0111:</td> <td>7 data bytes</td> <td></td> </tr> <tr> <td>B'1000:</td> <td>8 data bytes</td> <td></td> </tr> <tr> <td>B'1001:</td> <td>8 data bytes</td> <td>12 data bytes</td> </tr> <tr> <td>B'1010:</td> <td></td> <td>16 data bytes</td> </tr> <tr> <td>B'1011:</td> <td></td> <td>20 data bytes</td> </tr> <tr> <td>B'1100:</td> <td></td> <td>24 data bytes</td> </tr> <tr> <td>B'1101:</td> <td></td> <td>32 data bytes</td> </tr> <tr> <td>B'1110:</td> <td></td> <td>48 data bytes</td> </tr> <tr> <td>B'1111:</td> <td></td> <td>64 data bytes</td> </tr> </tbody> </table> | | Classical CAN Frame | CAN FD Frame | B'0000: | 0 data bytes | | B'0001: | 1 data byte | | B'0010: | 2 data bytes | | B'0011: | 3 data bytes | | B'0100: | 4 data bytes | | B'0101: | 5 data bytes | | B'0110: | 6 data bytes | | B'0111: | 7 data bytes | | B'1000: | 8 data bytes | | B'1001: | 8 data bytes | 12 data bytes | B'1010: | | 16 data bytes | B'1011: | | 20 data bytes | B'1100: | | 24 data bytes | B'1101: | | 32 data bytes | B'1110: | | 48 data bytes | B'1111: | | 64 data bytes |
| | Classical CAN Frame | CAN FD Frame | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0000: | 0 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0001: | 1 data byte | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0010: | 2 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0011: | 3 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0100: | 4 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0101: | 5 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0110: | 6 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0111: | 7 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1000: | 8 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1001: | 8 data bytes | 12 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1010: | | 16 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1011: | | 20 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1100: | | 24 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1101: | | 32 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1110: | | 48 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1111: | | 64 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 27 to 16 | RMPTR[11:0] | H'000 | R | Receive Buffer Label Data These bits indicate the label information of the message stored in the receive buffer. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 15 to 0 | RMTS[15:0] | H'0000 | R | Receive Buffer Timestamp Data These bits indicate the timestamp value of the message stored in the receive buffer. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

48.9.5.5 RSCFDnCFDRMFDSTSq — Receive Buffer CAN FD Status Register (q = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRMFDSTSq register can be read only in 32-bit units

RSCFDnCFDRMFDSTSqL, RSCFDnCFDRMFDSTSqH registers can be read only in 16-bit units

RSCFDnCFDRMFDSTSqLL, RSCFDnCFDRMFDSTSqLH, RSCFDnCFDRMFDSTSqHL,

RSCFDnCFDRMFDSTSqHH registers can be read only in 8-bit units

Address: RSCFDnCFDRMFDSTSq: <RSCFDn_base> + H'2008 + (H'20 × q)

RSCFDnCFDRMFDSTSqL: <RSCFDn_base> + H'2008 + (H'20 × q),

RSCFDnCFDRMFDSTSqH: <RSCFDn_base> + H'200A + (H'20 × q)

RSCFDnCFDRMFDSTSqLL: <RSCFDn_base> + H'2008 + (H'20 × q),

RSCFDnCFDRMFDSTSqLH: <RSCFDn_base> + H'2009 + (H'20 × q),

RSCFDnCFDRMFDSTSqHL: <RSCFDn_base> + H'200A + (H'20 × q),

RSCFDnCFDRMFDSTSqHH: <RSCFDn_base> + H'200B + (H'20 × q)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | RMFDF | RMBRS | RMESI |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 3 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 2 | RMFDF | B'0 | R | <p>FDF</p> <p>This bit indicates the FD format (classical CAN frame or CAN FD frame) of the message stored in the receive buffer.</p> <p>0: Classical CAN frame</p> <p>1: CAN FD frame</p> |
| 1 | RMBRS | B'0 | R | <p>BRS</p> <p>When the RMFDF bit is 1, this bit indicates the BRS bit value of the message stored in the receive buffer. When the RMFDF bit is 0, this bit is always read as 0.</p> <p>0: The bit rate in the data area is not changed.</p> <p>1: The bit rate in the data area is changed.</p> |
| 0 | RMESI | B'0 | R | <p>ESI</p> <p>When the RMFDF bit is 1, this bit indicates the ESI bit value of the message stored in the receive buffer. When the RMFDF bit is 0, this bit is always read as 0.</p> <p>0: Error-active node</p> <p>1: Error-passive node</p> |

48.9.5.6 RSCFDnCFDRMDFb_q — Receive Buffer Data Field b Register (b = 0 to 4, q = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRMDFb_q register can be read only in 32-bit units

RSCFDnCFDRMDFb_qL, RSCFDnCFDRMDFb_qH registers can be read only in 16-bit units

RSCFDnCFDRMDFb_qLL, RSCFDnCFDRMDFb_qLH, RSCFDnCFDRMDFb_qHL, RSCFDnCFDRMDFb_qHH registers can be read only in 8-bit units

Address: RSCFDnCFDRMDFb_q: <RSCFDn_base> + H'200C + (H'04 × b) + (H'20 × q)

RSCFDnCFDRMDFb_qL: <RSCFDn_base> + H'200C + (H'04 × b) + (H'20 × q),

RSCFDnCFDRMDFb_qH: <RSCFDn_base> + H'200E + (H'04 × b) + (H'20 × q)

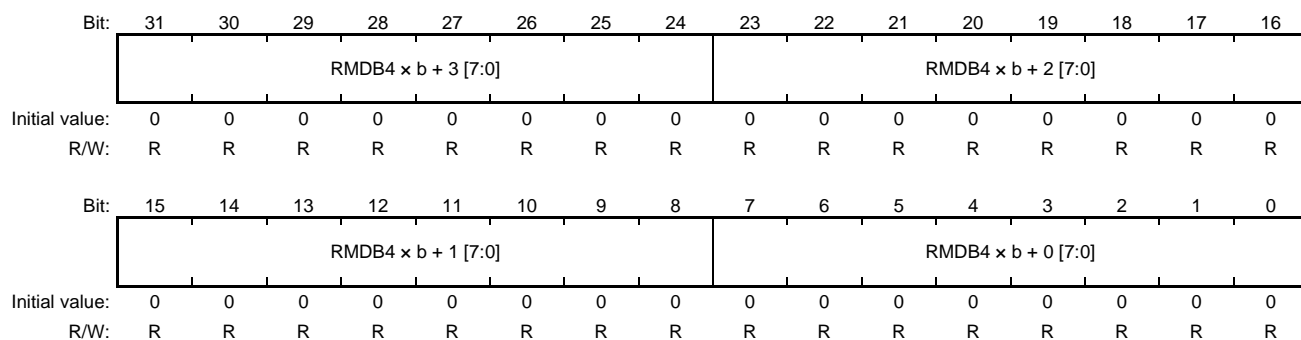
RSCFDnCFDRMDFb_qLL: <RSCFDn_base> + H'200C + (H'04 × b) + (H'20 × q),

RSCFDnCFDRMDFb_qLH: <RSCFDn_base> + H'200D + (H'04 × b) + (H'20 × q),

RSCFDnCFDRMDFb_qHL: <RSCFDn_base> + H'200E + (H'04 × b) + (H'20 × q),

RSCFDnCFDRMDFb_qHH: <RSCFDn_base> + H'200F + (H'04 × b) + (H'20 × q)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|--|
| 31 to 24 | RMDB4 × b + 3 [7:0] | H'00 | R | Receive Buffer Data Byte 4 × b + 3 |
| | | | | Receive Buffer Data Byte 4 × b + 2 |
| 23 to 16 | RMDB4 × b + 2 [7:0] | H'00 | R | Receive Buffer Data Byte 4 × b + 1 |
| | | | | Receive Buffer Data Byte 4 × b + 0 |
| 15 to 8 | RMDB4 × b + 1 [7:0] | H'00 | R | Data for a message stored in the receive buffer can be read. |
| 7 to 0 | RMDB4 × b + 0 [7:0] | H'00 | R | |

When the RMDLC[3:0] bit values in the RSCFDnCFDRMPTRq register are smaller than the size for storing the payload in the receive buffer, data bytes for which no data is set are read as H'00.

The size for storing the payload in the receive buffer is specified with the RMPLS[1:0] bits in the RSCFDnCFDRMNB register. The RSCFDnCFDRMDFb_q register corresponding to the area which exceeds the specified size must not be read or written.

48.9.6 Details of Receive FIFO Buffer Related Registers

48.9.6.1 RSCFDnCFDRFCCx — Receive FIFO Buffer Configuration and Control Register (x = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRFCCx register can be read/written in 32-bit units

RSCFDnCFDRFCCxL, RSCFDnCFDRFCCxH registers can be read/written in 16-bit units

RSCFDnCFDRFCCxLL, RSCFDnCFDRFCCxLH, RSCFDnCFDRFCCxHL, RSCFDnCFDRFCCxHH registers can be read/written in 8-bit units

Address: RSCFDnCFDRFCCx: <RSCFDn_base> + H'00B8 + (H'04 × x)

RSCFDnCFDRFCCxL: <RSCFDn_base> + H'00B8 + (H'04 × x),

RSCFDnCFDRFCCxH: <RSCFDn_base> + H'00BA + (H'04 × x)

RSCFDnCFDRFCCxLL: <RSCFDn_base> + H'00B8 + (H'04 × x),

RSCFDnCFDRFCCxLH: <RSCFDn_base> + H'00B9 + (H'04 × x),

RSCFDnCFDRFCCxHL: <RSCFDn_base> + H'00BA + (H'04 × x),

RSCFDnCFDRFCCxHH: <RSCFDn_base> + H'00BB + (H'04 × x)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|-------------|-----|-----|------|----|-----------|-----|-----|----|------------|-----|-----|----|----|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RFIGCV[2:0] | | | RFIM | — | RFDC[2:0] | | | — | RFPLS[2:0] | | | — | — | RFIE | RFE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 16 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 15 to 13 | RFIGCV[2:0] | B'000 | R/W | Receive FIFO Interrupt Request Timing Select These bits are used to specify the number of received messages for generating a receive FIFO interrupt request when the RFIM bit is set to 0 with a fraction for the total number of buffers (the setting of RFDC[2:0]). When the RFDC[2:0] bits are set to B'001 (4 messages), set the RFIGCV[2:0] bits to B'001, B'011, B'101, or B'111. Modify these bits only in global reset mode. B'000: When FIFO is 1/8 full. B'001: When FIFO is 2/8 full. B'010: When FIFO is 3/8 full. B'011: When FIFO is 4/8 full. B'100: When FIFO is 5/8 full. B'101: When FIFO is 6/8 full. B'110: When FIFO is 7/8 full. B'111: When FIFO is full. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 12 | RFIM | B'0 | R/W | <p>Receive FIFO Interrupt Source Select</p> <p>This bit is used to select a FIFO interrupt source. Modify this bit only in global reset mode.</p> <p>0: An interrupt occurs when the condition set by the RFIGCV[2:0] bits is met.</p> <p>1: An interrupt occurs each time a message has been received.</p> |
| 11 | Reserved | B'0 | R | <p>This bit is read as the value after reset.</p> <p>The write value should be the value after reset.</p> |
| 10 to 8 | RFDC[2:0] | B'000 | R/W | <p>Receive FIFO Buffer Depth Configuration</p> <p>These bits are used to select the number of messages that can be stored in a single receive FIFO buffer.</p> <p>When these bits are set to B'000, no receive FIFO buffer should be used. Modify these bits only in global reset mode.</p> <p>B'000: 0 messages B'001: 4 messages B'010: 8 messages B'011: 16 messages B'100: 32 messages B'101: 48 messages B'110: 64 messages B'111: 128 messages</p> |
| 7 | Reserved | B'0 | R | <p>This bit is read as the value after reset.</p> <p>The write value should be the value after reset.</p> |
| 6 to 4 | RFPLS[2:0] | B'000 | R/W | <p>Receive FIFO Buffer Payload Storing Size Select</p> <p>These bits are used to select the upper-limited size of the payload which can be stored in the receive FIFO buffer. Modify these bits only in global reset mode.</p> <p>B'000: 8 bytes B'001: 12 bytes B'010: 16 bytes B'011: 20 bytes B'100: 24 bytes B'101: 32 bytes B'110: 48 bytes B'111: 64 bytes</p> |
| 3, 2 | Reserved | All 0 | R | <p>These bits are read as the value after reset.</p> <p>The write value should be the value after reset.</p> |
| 1 | RFIE | B'0 | R/W | <p>Receive FIFO Interrupt Enable</p> <p>Setting the RFIE bit to 1 enables receive FIFO interrupts. Modify this bit when the RFE bit set to 0 (no receive FIFO buffer is used).</p> <p>0: Receive FIFO interrupt is disabled. 1: Receive FIFO interrupt is enabled.</p> |
| 0 | RFE | B'0 | R/W | <p>Receive FIFO Buffer Enable</p> <p>Setting the RFE bit to 1 makes receive FIFO buffers available. Clearing this bit to 0 sets the RFEMP flag in the RSCFDnCFDRFSTSx register to 1 (buffer empty). Modify this bit in global operating mode or global test mode.</p> <p>0: No receive FIFO buffer is used. 1: Receive FIFO buffers are used.</p> |

48.9.6.2 RSCFDnCFDRFSTSx — Receive FIFO Buffer Status Register (x = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRFSTSx register can be read/written in 32-bit units

RSCFDnCFDRFSTSxL, RSCFDnCFDRFSTSxH registers can be read/written in 16-bit units

RSCFDnCFDRFSTSxLL, RSCFDnCFDRFSTSxLH, RSCFDnCFDRFSTSxHL, RSCFDnCFDRFSTSxHH registers can be read/written in 8-bit units

Address: RSCFDnCFDRFSTSx: <RSCFDn_base> + H'00D8 + (H'04 × x)

RSCFDnCFDRFSTSxL: <RSCFDn_base> + H'00D8 + (H'04 × x),

RSCFDnCFDRFSTSxH: <RSCFDn_base> + H'00DA + (H'04 × x)

RSCFDnCFDRFSTSxLL: <RSCFDn_base> + H'00D8 + (H'04 × x),

RSCFDnCFDRFSTSxLH: <RSCFDn_base> + H'00D9 + (H'04 × x),

RSCFDnCFDRFSTSxHL: <RSCFDn_base> + H'00DA + (H'04 × x),

RSCFDnCFDRFSTSxHH: <RSCFDn_base> + H'00DB + (H'04 × x)

Value after reset: H'0000_0001

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----|----|----|----|----|----|----|----|------|-------|------|-------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RFMC[7:0] | | | | | | | — | — | — | — | RFIF | RFMLT | RFLL | RFEMP | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W* | R/W* | R | R |

Note: * The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 16 | Reserved | All 0 | R | These bits are read as the value after reset. When writing to these bits, write the value after reset. |
| 15 to 8 | RFMC[7:0] | H'00 | R | Receive FIFO Unread Message Counter This flag indicates the number of unread messages in the receive FIFO buffer. This flag becomes H'00 when the RFE bit in the RSCFDnCFDRFCCx register is set to 0. |
| 7 to 4 | Reserved | All 0 | R | These bits are read as the value after reset. When writing to these bits, write the value after reset. |
| 3 | RFIF | B'0 | R/W | Receive FIFO Interrupt Request Flag This flag is set to 1 when the receive FIFO interrupt request generation conditions set by the RFIGCV[2:0] bits and the RFIM bit in the RSCFDnCFDRFCCx register are met. This flag is cleared to 0 in global reset mode or by writing 0 to this flag. Modify this bit in global operating mode or global test mode. To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1". 0: No receive FIFO interrupt request is present. 1: A receive FIFO interrupt request is present. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 2 | RFMLT | B'0 | R/W | <p>Receive FIFO Message Lost Flag</p> <p>This flag is set to 1 when an attempt is made to store a new message while the receive FIFO buffer is full. In this case, the new message is discarded.</p> <p>This flag is cleared to 0 in global reset mode or by writing 0 to this flag.</p> <p>Modify this bit in global operating mode or global test mode.</p> <p>To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.</p> <p>When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1".</p> <p>0: No receive FIFO message is lost. 1: A receive FIFO message is lost.</p> |
| 1 | RFFLL | B'0 | R | <p>Receive FIFO Buffer Full Status Flag</p> <p>This flag is set to 1 when the number of messages stored in the receive FIFO buffer matches the FIFO buffer depth set by the RFDC[2:0] bits in the RSCFDnCFDRFCCx register.</p> <p>If the number of messages stored in the receive FIFO buffer becomes smaller than the FIFO buffer depth set by the RFDC[2:0] bits, this flag is cleared to 0. This flag is also cleared to 0 when the RFE bit in the RSCFDnCFDRFCCx register is set to 0 (no receive FIFO buffer is used) or in global reset mode.</p> <p>0: The receive FIFO buffer is not full. 1: The receive FIFO buffer is full.</p> |
| 0 | RFEMP | B'1 | R | <p>Receive FIFO Buffer Empty Status Flag</p> <p>This flag is set to 1 when all messages in the receive FIFO buffer have been read. This flag is also set to 1 when the RFE bit in the RSCFDnCFDRFCCx register is 0 or in global reset mode.</p> <p>This flag is cleared to 0 when even a single received message has been stored in the receive FIFO buffer.</p> <p>0: The receive FIFO buffer contains unread message. 1: The receive FIFO buffer contains no unread message (buffer empty).</p> |

Note: To clear the RFMLT or RFIF flag to 0, use a store instruction to write "0" to the given flag and "1" to the other flags.

48.9.6.3 RSCFDnCFDRFPCTR_x — Receive FIFO Buffer Pointer Control Register (x = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRFPCTR_x register can only be written in 32-bit units

RSCFDnCFDRFPCTR_{xL}, RSCFDnCFDRFPCTR_{xH} registers can only be written in 16-bit units

RSCFDnCFDRFPCTR_{xLL}, RSCFDnCFDRFPCTR_{xLH}, RSCFDnCFDRFPCTR_{xHL}, RSCFDnCFDRFPCTR_{xHH} registers can only be written in 8-bit units

Address: RSCFDnCFDRFPCTR_x: <RSCFDn_base> + H'00F8 + (H'04 × x)

RSCFDnCFDRFPCTR_{xL}: <RSCFDn_base> + H'00F8 + (H'04 × x),

RSCFDnCFDRFPCTR_{xH}: <RSCFDn_base> + H'00FA + (H'04 × x)

RSCFDnCFDRFPCTR_{xLL}: <RSCFDn_base> + H'00F8 + (H'04 × x),

RSCFDnCFDRFPCTR_{xLH}: <RSCFDn_base> + H'00F9 + (H'04 × x),

RSCFDnCFDRFPCTR_{xHL}: <RSCFDn_base> + H'00FA + (H'04 × x),

RSCFDnCFDRFPCTR_{xHH}: <RSCFDn_base> + H'00FB + (H'04 × x)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-----------|----|----|----|----|----|----|----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | RFPC[7:0] | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | W | W | W | W | W | W | W | W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 8 | Reserved | All 0 | R | The write value should be the value after reset. |
| 7 to 0 | RFPC[7:0] | H'00 | W | <p>Receive FIFO Pointer Control</p> <p>When the RFPC[7:0] bits are set to H'FF, the read pointer moves to the next unread message in the receive FIFO buffer. At this time, the RFMC[7:0] (receive FIFO unread message counter) value in the RSCFDnCFDRFSTS_x register is decremented. Read the RSCFDnCFDRFID_x, RSCFDnCFDRFPTR_x, RSCFDnCFDRFFDSTS_x, and RSCFDnCFDRFDFd_x registers to read messages in the receive FIFO buffer, and then write H'FF to the RFPC[7:0] bits.</p> <p>When writing H'FF to these bits, make sure that the RFE bit in the RSCFDnCFDRFCC_x register is set to 1 (receive FIFO buffers are used) and the RFEMP flag in the RSCFDnCFDRFSTS_x register is 0 (the receive FIFO buffer contains unread messages).</p> |

When the RFDMAEx bit in the RSCFDnCFDCDTCT register is 1 (the DMA transfer request is enabled), this register must not be written.

48.9.6.4 RSCFDnCFDRFIDx — Receive FIFO Buffer Access ID Register (x = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRFIDx register can be read only in 32-bit units

RSCFDnCFDRFIDxL, RSCFDnCFDRFIDxH registers can be read only in 16-bit units

RSCFDnCFDRFIDxLL, RSCFDnCFDRFIDxLH, RSCFDnCFDRFIDxHL, RSCFDnCFDRFIDxHH registers can be read only in 8-bit units

Address: RSCFDnCFDRFIDx: <RSCFDn_base> + H'3000 + (H'80 × x)

RSCFDnCFDRFIDxL: <RSCFDn_base> + H'3000 + (H'80 × x),

RSCFDnCFDRFIDxH: <RSCFDn_base> + H'3002 + (H'80 × x)

RSCFDnCFDRFIDxLL: <RSCFDn_base> + H'3000 + (H'80 × x),

RSCFDnCFDRFIDxLH: <RSCFDn_base> + H'3001 + (H'80 × x),

RSCFDnCFDRFIDxHL: <RSCFDn_base> + H'3002 + (H'80 × x),

RSCFDnCFDRFIDxHH: <RSCFDn_base> + H'3003 + (H'80 × x)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|------------|--------|----|-------------|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RFIDE | RFRTTR | — | RFID[28:16] | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RFID[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 | RFIDE | B'0 | R | Receive FIFO Buffer IDE This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive FIFO buffer. 0: Standard ID 1: Extended ID |
| 30 | RFRTTR | B'0 | R | Receive FIFO Buffer RTR/RRS When the received message is in the classical CAN frame, this bit indicates the frame format (data frame or remote frame) of the message stored in the receive FIFO buffer. When the received message is in the CAN FD frame, this bit indicates the RRS bit value of the message. <ul style="list-style-type: none"> When the received message is in the classical CAN frame: <ul style="list-style-type: none"> 0: Data frame 1: Remote frame When the received message is in the CAN FD frame: <ul style="list-style-type: none"> The RRS bit value of the received message is read. |
| 29 | Reserved | B'0 | R | This bit is read as the value after reset. |
| 28 to 0 | RFID[28:0] | H'0000_0000 | R | Receive FIFO Buffer ID Data These bits indicate the ID of the message stored in the receive FIFO buffer. The standard ID or extended ID of received message can be read. Read bits 10 to 0 for standard ID. Bits 28 to 11 are read as 0. |

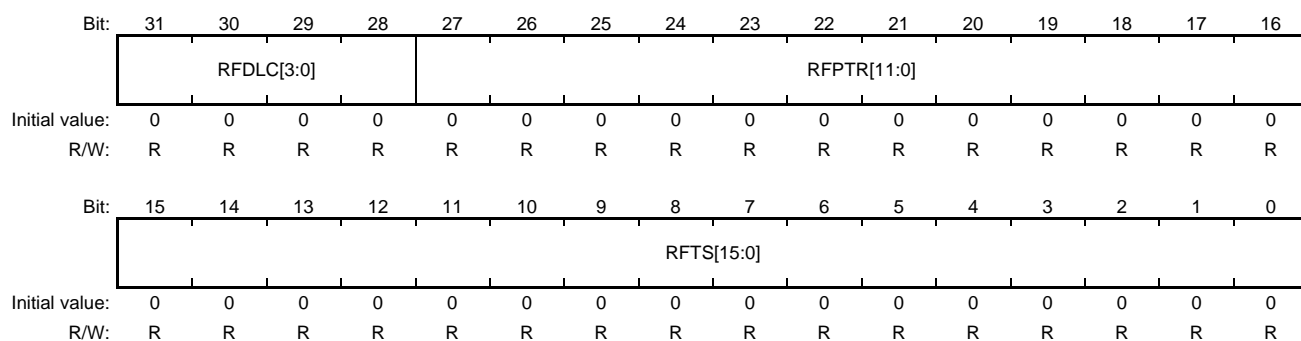
48.9.6.5 RSCFDnCFDRFPTRx — Receive FIFO Buffer Access Pointer Register (x = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRFPTRx register can be read only in 32-bit units
 RSCFDnCFDRFPTRxL, RSCFDnCFDRFPTRxH registers can be read only in 16-bit units
 RSCFDnCFDRFPTRxLL, RSCFDnCFDRFPTRxLH, RSCFDnCFDRFPTRxHL, RSCFDnCFDRFPTRxHH registers can be read only in 8-bit units

Address: RSCFDnCFDRFPTRx: <RSCFDn_base> + H'3004 + (H'80 × x)
 RSCFDnCFDRFPTRxL: <RSCFDn_base> + H'3004 + (H'80 × x),
 RSCFDnCFDRFPTRxH: <RSCFDn_base> + H'3006 + (H'80 × x)
 RSCFDnCFDRFPTRxLL: <RSCFDn_base> + H'3004 + (H'80 × x),
 RSCFDnCFDRFPTRxLH: <RSCFDn_base> + H'3005 + (H'80 × x),
 RSCFDnCFDRFPTRxHL: <RSCFDn_base> + H'3006 + (H'80 × x),
 RSCFDnCFDRFPTRxHH: <RSCFDn_base> + H'3007 + (H'80 × x)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------|---------------------|---------------|-----|---|--|---------------------|--------------|---------|--------------|--|---------|-------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|
| 31 to 28 | RFDLC[3:0] | B'0000 | R | Receive FIFO Buffer DLC Data These bits contain the data length of the message stored in the receive FIFO buffer. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | <table border="1"> <thead> <tr> <th></th> <th>Classical CAN Frame</th> <th>CAN FD Frame</th> </tr> </thead> <tbody> <tr> <td>B'0000:</td> <td>0 data bytes</td> <td></td> </tr> <tr> <td>B'0001:</td> <td>1 data byte</td> <td></td> </tr> <tr> <td>B'0010:</td> <td>2 data bytes</td> <td></td> </tr> <tr> <td>B'0011:</td> <td>3 data bytes</td> <td></td> </tr> <tr> <td>B'0100:</td> <td>4 data bytes</td> <td></td> </tr> <tr> <td>B'0101:</td> <td>5 data bytes</td> <td></td> </tr> <tr> <td>B'0110:</td> <td>6 data bytes</td> <td></td> </tr> <tr> <td>B'0111:</td> <td>7 data bytes</td> <td></td> </tr> <tr> <td>B'1000:</td> <td>8 data bytes</td> <td></td> </tr> <tr> <td>B'1001:</td> <td>8 data bytes</td> <td>12 data bytes</td> </tr> <tr> <td>B'1010:</td> <td></td> <td>16 data bytes</td> </tr> <tr> <td>B'1011:</td> <td></td> <td>20 data bytes</td> </tr> <tr> <td>B'1100:</td> <td></td> <td>24 data bytes</td> </tr> <tr> <td>B'1101:</td> <td></td> <td>32 data bytes</td> </tr> <tr> <td>B'1110:</td> <td></td> <td>48 data bytes</td> </tr> <tr> <td>B'1111:</td> <td></td> <td>64 data bytes</td> </tr> </tbody> </table> | | Classical CAN Frame | CAN FD Frame | B'0000: | 0 data bytes | | B'0001: | 1 data byte | | B'0010: | 2 data bytes | | B'0011: | 3 data bytes | | B'0100: | 4 data bytes | | B'0101: | 5 data bytes | | B'0110: | 6 data bytes | | B'0111: | 7 data bytes | | B'1000: | 8 data bytes | | B'1001: | 8 data bytes | 12 data bytes | B'1010: | | 16 data bytes | B'1011: | | 20 data bytes | B'1100: | | 24 data bytes | B'1101: | | 32 data bytes | B'1110: | | 48 data bytes | B'1111: | | 64 data bytes |
| | Classical CAN Frame | CAN FD Frame | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0000: | 0 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0001: | 1 data byte | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0010: | 2 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0011: | 3 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0100: | 4 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0101: | 5 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0110: | 6 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0111: | 7 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1000: | 8 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1001: | 8 data bytes | 12 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1010: | | 16 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1011: | | 20 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1100: | | 24 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1101: | | 32 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1110: | | 48 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1111: | | 64 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 27 to 16 | RFPTR[11:0] | H'000 | R | Receive FIFO Buffer Label Data These bits contain the label information of the message stored in the receive FIFO buffer. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 15 to 0 | RFTS[15:0] | H'0000 | R | Receive FIFO Buffer Timestamp Data These bits contain the timestamp value of the message stored in the receive FIFO buffer. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

48.9.6.6 RSCFDnCFDRFFDSTSx — Receive FIFO CAN FD Status Register (x = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRFFDSTSx register can be read only in 32-bit units
 RSCFDnCFDRFFDSTSxL, RSCFDnCFDRFFDSTSxH registers can be read only in 16-bit units
 RSCFDnCFDRFFDSTSxLL, RSCFDnCFDRFFDSTSxLH, RSCFDnCFDRFFDSTSxHL,
 RSCFDnCFDRFFDSTSxHH registers can be read only in 8-bit units

Address: RSCFDnCFDRFFDSTSx: <RSCFDn_base> + H'3008 + (H'80 × x)
 RSCFDnCFDRFFDSTSxL: <RSCFDn_base> + H'3008 + (H'80 × x),
 RSCFDnCFDRFFDSTSxH: <RSCFDn_base> + H'300A + (H'80 × x)
 RSCFDnCFDRFFDSTSxLL: <RSCFDn_base> + H'3008 + (H'80 × x),
 RSCFDnCFDRFFDSTSxLH: <RSCFDn_base> + H'3009 + (H'80 × x),
 RSCFDnCFDRFFDSTSxHL: <RSCFDn_base> + H'300A + (H'80 × x),
 RSCFDnCFDRFFDSTSxHH: <RSCFDn_base> + H'300B + (H'80 × x)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | RFFDF | RFBRS | RFESI |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 3 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 2 | RFFDF | B'0 | R | RFFDF This bit indicates the FD format (classical CAN frame or CAN FD frame) of the message stored in the receive FIFO buffer. 0: Classical CAN frame 1: CAN FD frame |
| 1 | RFBRS | B'0 | R | RFBRS When the RFFDF bit is 1, this bit indicates the BRS bit value of the message stored in the receive FIFO buffer. When the RFFDF bit is 0, this bit is always read as 0. 0: The bit rate in the data area is not changed. 1: The bit rate in the data area is changed. |
| 0 | RFESI | B'0 | R | RFESI When the RFFDF bit is 1, this bit indicates the ESI bit value of the message stored in the receive FIFO buffer. When the RFFDF bit is 0, this bit is always read as 0. |

48.9.6.7 RSCFDnCFDRFDFd_x — Receive FIFO Buffer Access Data Field d Register (d = 0 to 15, x = 0 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRFDFd_x register can be read only in 32-bit units

RSCFDnCFDRFDFd_xL, RSCFDnCFDRFDFd_xH registers can be read only in 16-bit units

RSCFDnCFDRFDFd_xLL, RSCFDnCFDRFDFd_xLH, RSCFDnCFDRFDFd_xHL, RSCFDnCFDRFDFd_xHH registers can be read only in 8-bit units

Address: RSCFDnCFDRFDFd_x: <RSCFDn_base> + H'300C + (H'04 × d) + (H'80 × x)

RSCFDnCFDRFDFd_xL: <RSCFDn_base> + H'300C + (H'04 × d) + (H'80 × x),

RSCFDnCFDRFDFd_xH: <RSCFDn_base> + H'300E + (H'04 × d) + (H'80 × x)

RSCFDnCFDRFDFd_xLL: <RSCFDn_base> + H'300C + (H'04 × d) + (H'80 × x),

RSCFDnCFDRFDFd_xLH: <RSCFDn_base> + H'300D + (H'04 × d) + (H'80 × x),

RSCFDnCFDRFDFd_xHL: <RSCFDn_base> + H'300E + (H'04 × d) + (H'80 × x),

RSCFDnCFDRFDFd_xHH: <RSCFDn_base> + H'300F + (H'04 × d) + (H'80 × x)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|---------------------|----|----|----|----|----|----|----|---------------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RFDB4 × d + 3 [7:0] | | | | | | | | RFDB4 × d + 2 [7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RFDB4 × d + 1 [7:0] | | | | | | | | RFDB4 × d + 0 [7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|---|
| 31 to 24 | RFDB4 × d + 3 [7:0] | H'00 | R | Receive FIFO Buffer Data Byte 4 × d + 3 |
| 23 to 16 | RFDB4 × d + 2 [7:0] | H'00 | R | Receive FIFO Buffer Data Byte 4 × d + 2 |
| 15 to 8 | RFDB4 × d + 1 [7:0] | H'00 | R | Receive FIFO Buffer Data Byte 4 × d + 1 |
| 7 to 0 | RFDB4 × d + 0 [7:0] | H'00 | R | Receive FIFO Buffer Data Byte 4 × d + 0 |

Data for a message stored in the receive FIFO buffer can be read.

When the RFDLC[3:0] bit values in the RSCFDnCFDRFPTRx register are smaller than the size for storing the payload in the receive FIFO buffer, data bytes for which no data is set are read as H'00.

The size for storing the payload in the receive FIFO buffer is specified with the RFPLS[2:0] bits in the RSCFDnCFDRFCCx register. The RSCFDnCFDRFDFd_x register corresponding to the area which exceeds the specified size must not be read or written.

48.9.7 Details of Transmit/Receive FIFO Buffer Related Registers

48.9.7.1 RSCFDnCFDCFCCK — Transmit/receive FIFO Buffer Configuration and Control Register k (k = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCFCCK register can be read/written in 32-bit units

RSCFDnCFDCFCCKL, RSCFDnCFDCFCCKH registers can be read/written in 16-bit units

RSCFDnCFDCFCCKLL, RSCFDnCFDCFCCKLH, RSCFDnCFDCFCCKHL, RSCFDnCFDCFCCKHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCFCCK: <RSCFDn_base> + H'0118 + (H'04 × k)

RSCFDnCFDCFCCKL: <RSCFDn_base> + H'0118 + (H'04 × k),

RSCFDnCFDCFCCKH: <RSCFDn_base> + H'011A + (H'04 × k)

RSCFDnCFDCFCCKLL: <RSCFDn_base> + H'0118 + (H'04 × k),

RSCFDnCFDCFCCKLH: <RSCFDn_base> + H'0119 + (H'04 × k),

RSCFDnCFDCFCCKHL: <RSCFDn_base> + H'011A + (H'04 × k),

RSCFDnCFDCFCCKHH: <RSCFDn_base> + H'011B + (H'04 × k)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----|-----|------|-----|-----------|-----|------------|------------|-----|-----|-------|--------|------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CFITT[7:0] | | | | | | | CFTML[3:0] | | | | CFITR | CFITSS | CFM[1:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CFGCV[2:0] | | | CFIM | — | CFDC[2:0] | | — | CFPLS[2:0] | | | — | CFTXIE | CFRXI E | CFE | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 24 | CFITT[7:0] | H'00 | R/W | Set a message transmission interval. These bits are used to set a message transmission interval when transmitting messages continuously from a transmit/receive FIFO buffer whose CFM[1:0] bits are set to B'01 (transmit mode) or B'10 (gateway mode). Clear the CFE bit to 0 (no transmit/receive FIFO buffer is used) before modifying the CFITT[7:0] bits. Set Value: H'00 to H'FF |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 23 to 20 | CFTML[3:0] | H'0 | R/W | <p>Transmit Buffer Link Configuration</p> <p>These bits are used to set the number of transmit buffer on the channel which will be linked to transmit/receive FIFO buffer k when the CFM[1:0] bits are set to B'01 (transmit mode) or B'10 (gateway mode).</p> <p>There are three transmit/receive FIFO buffers per channel, so channel number m of FIFO buffer k is calculated as $m = k/3$ (integer division). The actual assigned transmit buffer number p linked to FIFO buffer k will be $((16 \times m) + CFTML[3:0])$.</p> <p>See Table 48.17 and Table 48.18, as for the relationship between transmit/receive FIFO buffer k and transmit buffer p.</p> <p>Setting the CFDC[2:0] bits to B'001 or more enables the setting of the CFTML[3:0] bits.</p> <p>Do not link to any transmit buffer which is already allocated to a transmit queue on the identical channel or to another transmit/receive FIFO buffer. Modify these bits only in global reset mode.</p> |
| 19 | CFITR | B'0 | R/W | <p>Transmit/Receive FIFO Interval Timer Resolution</p> <p>This bit is enabled when the CFITSS bit is 0.</p> <p>When this bit is 0, the interval timer clock source is the pclk/2 clock divided by the value of the ITRCP[15:0] bits in the RSCFDnCFDGCFG register.</p> <p>When this bit is 1, the interval timer clock source is the pclk/2 clock divided by (the value of the ITRCP[15:0] bits in the RSCFDnCFDGCFG register $\times 10$).</p> <p>Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).</p> <p>0: Clock dividing pclk/2 by (ITRCP[15:0] bits) 1: Clock dividing pclk/2 by (ITRCP[15:0] bits $\times 10$)</p> |
| 18 | CFITSS | B'0 | R/W | <p>Transmit/Receive FIFO Interval Timer Clock Source Select</p> <p>When this bit is 0, the clock selected by the CFITR bit is the count source of the interval timer.</p> <p>When this bit is 1, the normal bit time clock of the channel to which the FIFO is linked is the count source of the interval timer.</p> <p>Use this bit only for a channel which does not handle the CAN FD frame.</p> <p>Modify this bit while the CFE bit is cleared to 0 (no transmit/receive FIFO buffer is used).</p> <p>0: Interval timer clock source selected by the CFITR bit 1: Interval timer clock source is the normal bit time clock for the channel to which the FIFO is linked.</p> |
| 17, 16 | CFM[1:0] | B'00 | R/W | <p>Transmit/Receive FIFO Mode Select</p> <p>These bits are used to select transmit/receive FIFO mode. Modify these bits only in global reset mode.</p> <p>B'00: Receive mode B'01: Transmit mode B'10: Gateway mode B'11: Setting prohibited</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 15 to 13 | CFIGCV[2:0] | B'000 | R/W | <p>Transmit/Receive FIFO Receive Interrupt Request Timing Select</p> <p>These bits are used to specify the number of received messages for generating a transmit/receive FIFO receive interrupt request when the CFM[1:0] bits are set to B'00 (receive mode) or B'10 (gateway mode) and the CFIM bit is set to 0 with a fraction for the total number of buffers (the setting of CFDC[2:0]).</p> <p>When the CFDC[2:0] bits are set to B'001 (4 messages), set the CFIGCV[2:0] bits to B'001, B'011, B'101, or B'111.</p> <p>Modify these bits only in global reset mode.</p> <p>B'000: When FIFO is 1/8 full. B'001: When FIFO is 2/8 full. B'010: When FIFO is 3/8 full. B'011: When FIFO is 4/8 full. B'100: When FIFO is 5/8 full. B'101: When FIFO is 6/8 full. B'110: When FIFO is 7/8 full. B'111: When FIFO is full.</p> |
| 12 | CFIM | B'0 | R/W | <p>Transmit/Receive FIFO Interrupt Source Select</p> <p>This bit is used to select a transmit/receive FIFO interrupt source. Modify this bit only in global reset mode.</p> <p>0:</p> <ul style="list-style-type: none"> Receive mode/gateway mode When the number of received messages has met the condition set by the CFIGCV[2:0] bits, a FIFO receive interrupt request is generated. Transmit mode/gateway mode When the buffer becomes empty upon completion of message transmission, a FIFO transmit interrupt request is generated. <p>1:</p> <ul style="list-style-type: none"> Receive mode/gateway mode A FIFO receive interrupt request is generated each time a message has been received. Transmit mode/gateway mode A FIFO transmit interrupt request is generated each time a message has been transmitted. |
| 11 | Reserved | B'0 | R | <p>This bit is read as the value after reset.</p> <p>The write value should be the value after reset.</p> |
| 10 to 8 | CFDC[2:0] | B'000 | R/W | <p>Transmit/Receive FIFO Buffer Depth Configuration</p> <p>These bits are used to set the number of messages that can be stored in a single transmit/receive FIFO buffer. When these bits are set to B'000, do not use a transmit/receive FIFO buffer. Modify these bits only in global reset mode.</p> <p>B'000: 0 messages B'001: 4 messages B'010: 8 messages B'011: 16 messages B'100: 32 messages B'101: 48 messages B'110: 64 messages B'111: 128 messages</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|--|
| 7 | Reserved | B'0 | R | This bit is read as the value after reset. The write value should be the value after reset. |
| 6 to 4 | CFPLS[2:0] | B'000 | R/W | Transmit/Receive FIFO Buffer Payload Storing Size Select These bits are used to select the upper-limited size of the payload which can be stored in the transmit/receive FIFO buffer. Modify these bits only in global reset mode. B'000: 8 bytes B'001: 12 bytes B'010: 16 bytes B'011: 20 bytes B'100: 24 bytes B'101: 32 bytes B'110: 48 bytes B'111: 64 bytes |
| 3 | Reserved | B'0 | R | This bit is read as the value after reset. The write value should be the value after reset. |
| 2 | CCTXIE | B'0 | R/W | Transmit/Receive FIFO Transmit Interrupt Enable When this bit is set to 1 and the CCTXIF flag in the RSCFDnCFDCFSTSk register is set to 1, a transmit/receive FIFO transmit interrupt request is generated. Modify this bit with the CFE bit set to 0 (no transmit/receive FIFO buffer is used). 0: Transmit/receive FIFO transmit interrupt is disabled. 1: Transmit/receive FIFO transmit interrupt is enabled. |
| 1 | CFRXIE | B'0 | R/W | Transmit/Receive FIFO Receive Interrupt Enable When this bit is set to 1 and the CFRXIF flag in the RSCFDnCFDCFSTSk register is set to 1, a transmit/receive FIFO receive interrupt request is generated. Modify this bit with the CFE bit set to 0. 0: Transmit/receive FIFO receive interrupt is disabled. 1: Transmit/receive FIFO receive interrupt is enabled. |
| 0 | CFE | B'0 | R/W | Transmit/Receive FIFO Buffer Enable Setting this bit to 1 makes transmit/receive FIFO buffers available. When this bit is set to 0 in transmit mode or gateway mode, if a message in the transmit/receive FIFO buffer is being transmitted or will be transmitted next, the transmit/receive FIFO buffer becomes empty after completion of transmission of that message, or upon detection of a CAN bus error, or arbitration-lost. In other cases or in receive mode, the transmit/receive FIFO buffer becomes empty immediately. This bit is cleared to 0 when the following conditions are met. <ul style="list-style-type: none"> • Receive mode: Global reset mode • Transmit mode or gateway mode: Channel reset mode Modify this bit in the following mode. <ul style="list-style-type: none"> • Receive mode: Global operating mode or global test mode • Transmit mode or gateway mode: Channel communication mode or channel halt mode First set all the other bits in the RSCFDnCFDCFCCk register and then set this bit to 1 using another instruction. 0: No transmit/receive FIFO buffer is used. 1: Transmit/receive FIFO buffers are used. |

48.9.7.2 RSCFDnCFDCFSTSk — Transmit/receive FIFO Buffer Status Register (k = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCFSTSk register can be read/written in 32-bit units

RSCFDnCFDCFSTSkL, RSCFDnCFDCFSTSkH registers can be read/written in 16-bit units

RSCFDnCFDCFSTSkLL, RSCFDnCFDCFSTSkLH, RSCFDnCFDCFSTSkHL, RSCFDnCFDCFSTSkHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCFSTSk: <RSCFDn_base> + H'0178 + (H'04 × k)

RSCFDnCFDCFSTSkL: <RSCFDn_base> + H'0178 + (H'04 × k),

RSCFDnCFDCFSTSkH: <RSCFDn_base> + H'017A + (H'04 × k)

RSCFDnCFDCFSTSkLL: <RSCFDn_base> + H'0178 + (H'04 × k),

RSCFDnCFDCFSTSkLH: <RSCFDn_base> + H'0179 + (H'04 × k),

RSCFDnCFDCFSTSkHL: <RSCFDn_base> + H'017A + (H'04 × k),

RSCFDnCFDCFSTSkHH: <RSCFDn_base> + H'017B + (H'04 × k)

Value after reset: H'0000_0001

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----|----|----|----|----|----|----|--------|--------|-------|-------|-------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CFMC[7:0] | | | | | | | — | — | — | CFTXIF | CFRXIF | CFMLT | CFPLL | CFEMP | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W* | R/W* | R/W* | R | R |

Note: * The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 16 | Reserved | All 0 | R | These bits are read as the value after reset. When writing to these bits, write the value after reset. |
| 15 to 8 | CFMC[7:0] | H'00 | R | <p>Transmit/Receive FIFO Message Counter</p> <p>The number of messages stored in the transmit/receive FIFO buffer.</p> <p>The CFMC[7:0] bits indicate the following values that depend on the setting of the CFM[1:0] bits in the RSCFDnCFDCFCCk register.</p> <ul style="list-style-type: none"> When CFM[1:0] value is B'01 (transmit mode): Number of untransmitted messages in the buffer When CFM[1:0] value is B'00 (receive mode): Number of unread received messages in the buffer When CFM[1:0] value is B'10 (gateway mode): Number of untransmitted received messages in the buffer <p>These bits are cleared to 0 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When CFM[1:0] value is B'00: In global reset mode When CFM[1:0] value is B'01 or B'10: In channel reset mode |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 to 5 | Reserved | All 0 | R | These bits are read as the value after reset. When writing to these bits, write the value after reset. |
| 4 | CFTXIF | B'0 | R/W | <p>Transmit/Receive FIFO Transmit Interrupt Request Flag</p> <p>The CFTXIF flag is set to 1 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When the CFM[1:0] bits are set to B'01 or B'10, and the factor selected by the CFIM bit in the RSCFDnCFDCFCCK register occurs <p>The CFTXIF flag is cleared to 0 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When 0 is written to the CFTXIF flag When the CFM[1:0] bits are set to B'00: In global reset mode When the CFM[1:0] bits are set to B'01 or B'10: In channel reset mode <p>Write 0 to this flag in global operating mode or global test mode.</p> <p>To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.</p> <p>When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1".</p> <p>0: No transmit/receive FIFO transmit interrupt request is present. 1: A transmit/receive FIFO transmit interrupt request is present.</p> |
| 3 | CFRXIF | B'0 | R/W | <p>Transmit/Receive FIFO Receive Interrupt Request Flag</p> <p>The CFRXIF flag is set to 1 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When the CFM[1:0] bits are set to B'00 or B'10, and the factor selected by the CFIM bit in the RSCFDnCFDCFCCK register occurs <p>The CFRXIF flag is cleared to 0 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When 0 is written to the CFRXIF flag When the CFM[1:0] bits are set to B'00: In global reset mode When the CFM[1:0] bits are set to B'01 or B'10: In channel reset mode <p>Write 0 to this flag in global operating mode or global test mode.</p> <p>To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.</p> <p>When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1".</p> <p>0: No transmit/receive FIFO receive interrupt request is present. 1: A transmit/receive FIFO receive interrupt request is present.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 2 | CFMLT | B'0 | R/W | <p>Transmit/Receive FIFO Message Lost Flag</p> <p>The CFMLT flag is set to 1 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When an attempt is made to store a new message while the transmit/receive FIFO buffer is full. In this case, the new message is discarded. <p>The CFMLT flag is cleared to 0 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When 0 is written to the CFMLT flag When the CFM[1:0] bits are set to B'00: In global reset mode When the CFM[1:0] bits are set to B'01 or B'10: In channel reset mode <p>Write 0 to this flag in global operating mode or global test mode</p> <p>To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.</p> <p>When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1".</p> <p>0: No transmit/receive FIFO message is lost. 1: A transmit/receive FIFO message is lost.</p> |
| 1 | CFLL | B'0 | R | <p>Transmit/Receive FIFO Buffer Full Status Flag</p> <p>The CFLL flag is set to 1 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When the number of messages stored in the transmit/receive FIFO buffer matches the FIFO buffer depth set by the CFDC[2:0] bits in the RSCFDnCFDCFCCK register. <p>The CFLL flag is cleared to 0 when any of the following conditions is met.</p> <ul style="list-style-type: none"> When the number of messages stored in the transmit/receive FIFO buffer becomes smaller than the FIFO buffer depth set by the CFDC[2:0] bits. When the CFE bit in the RSCFDnCFDCFCCK register is 0 (no transmit/receive FIFO buffer is used): When not in the transmit abort When the CFM[1:0] bits are set to B'00: In global reset mode When the CFM[1:0] bits are set to B'01 or B'10: In channel reset mode <p>0: The transmit/receive FIFO buffer is not full. 1: The transmit/receive FIFO buffer is full.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | CFEMP | B'1 | R | <p>Transmit/Receive FIFO Buffer Empty Status Flag</p> <p>The CFEMP flag is set to 1 when any of the following conditions is met.</p> <ul style="list-style-type: none"> • When the CFM[1:0] bits are set to B'00: All messages have been read, or in global reset mode • When the CFM[1:0] bits are set to B'01 or B'10: All messages have been transmitted, or in channel reset mode • When the CFE bit is 0 (no transmit/receive FIFO buffer is used): Not in the transmit abort <p>The CFEMP flag is cleared to 0 when any of the following conditions is met.</p> <ul style="list-style-type: none"> • When the CFM[1:0] bits are set to B'00 or B'10: At least one received message has been stored in the transmit/receive FIFO buffer. • When the CFM[1:0] bits are set to B'01: A value of H'FF has been written to the RSCFDnCFDCFPCTRk register after data was written to the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDf_k registers. <p>0: The transmit/receive FIFO buffer contains messages. 1: The transmit/receive FIFO buffer contains no message (buffer empty).</p> |

Note: To clear the CCTXIF, CFRXIF, or CFMLT flag to 0, the program must write 0 to the flag. Use a store instruction to write "0" to the flag and "1" to other flags.

48.9.7.3 RSCFDnCFDCFPCTRk — Transmit/receive FIFO Buffer Pointer Control Register (k = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCFPCTRk register can only be written in 32-bit units
 RSCFDnCFDCFPCTRkL, RSCFDnCFDCFPCTRkH registers can only be written in 16-bit units
 RSCFDnCFDCFPCTRkLL, RSCFDnCFDCFPCTRkLH, RSCFDnCFDCFPCTRkHL, RSCFDnCFDCFPCTRkHH registers can only be written in 8-bit units

Address: RSCFDnCFDCFPCTRk: <RSCFDn_base> + H'01D8 + (H'04 × k)
 RSCFDnCFDCFPCTRkL: <RSCFDn_base> + H'01D8 + (H'04 × k),
 RSCFDnCFDCFPCTRkH: <RSCFDn_base> + H'01DA + (H'04 × k)
 RSCFDnCFDCFPCTRkLL: <RSCFDn_base> + H'01D8 + (H'04 × k),
 RSCFDnCFDCFPCTRkLH: <RSCFDn_base> + H'01D9 + (H'04 × k),
 RSCFDnCFDCFPCTRkHL: <RSCFDn_base> + H'01DA + (H'04 × k),
 RSCFDnCFDCFPCTRkHH: <RSCFDn_base> + H'01DB + (H'04 × k)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-----------|----|----|----|----|----|----|----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | CFPC[7:0] | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | W | W | W | W | W | W | W | W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | Reserved | All 0 | R | The write value should be the value after reset. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|---|
| 7 to 0 | CFPC[7:0] | H'00 | W | <p>Transmit/Receive FIFO Pointer Control</p> <ul style="list-style-type: none"> <p>Receive mode (CFM[1:0] value in the RSCFDnCFDCFCCK register is B'00):</p> <p>Writing H'FF to the CFPC[7:0] bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value (transmit/receive FIFO message counter) in the RSCFDnCFDCFSTSk register is decremented. Read the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDf_k registers to read messages from the transmit/receive FIFO buffer, and then write H'FF to the CFPC[7:0] bits.</p> <p>When writing H'FF to these bits, make sure that the CFE bit in the RSCFDnCFDCFCCK register is set to 1 (transmit/receive FIFO buffers are used) and the CFEMP flag in the RSCFDnCFDCFSTSk register is 0 (the transmit/receive FIFO buffer contains messages).</p> <p>Transmit mode (CFM[1:0] value in the RSCFDnCFDCFCCK register is B'01):</p> <p>Writing H'FF to the CFPC[7:0] bits stores the data written to the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDf_k registers in the transmit/receive FIFO buffer and moves the write pointer to the next stage of the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value is incremented. Write transmit messages to the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDf_k registers before writing H'FF to the CFPC[7:0] bits.</p> <p>When writing H'FF to these bits, make sure that the CFE bit in the RSCFDnCFDCFCCK register is set to 1 and the CFFLL flag in the RSCFDnCFDCFSTSk register is 0 (the transmit/receive FIFO buffer is not full).</p> <p>Gateway mode (CFM[1:0] value in the RSCFDnCFDCFCCK register is B'10):</p> <p>Setting prohibited</p> |

When the corresponding transmit/receive FIFO buffer is the first transmit/receive FIFO buffer which has been assigned to channel m ($k = 3 \times m$) and the CFDMAEm bit in the RSCFDnCFDCDTCT register is 1 (the DMA transfer request is enabled), this register must not be written.

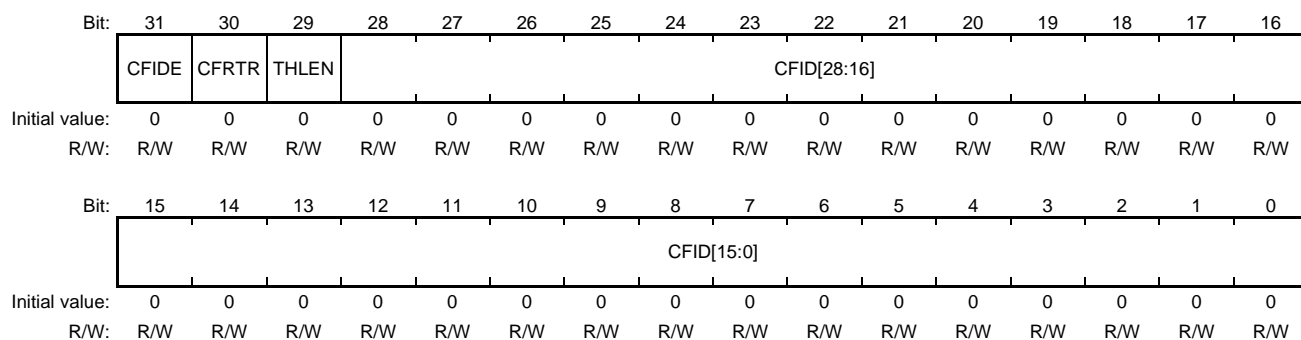
48.9.7.4 RSCFDnCFDCFIDk — Transmit/receive FIFO Buffer Access ID Register (k = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCFIDk register can be read/written in 32-bit units
 RSCFDnCFDCFIDkL, RSCFDnCFDCFIDkH registers can be read/written in 16-bit units
 RSCFDnCFDCFIDkLL, RSCFDnCFDCFIDkLH, RSCFDnCFDCFIDkHL, RSCFDnCFDCFIDkHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCFIDk: <RSCFDn_base> + H'3400 + (H'80 × k)
 RSCFDnCFDCFIDkL: <RSCFDn_base> + H'3400 + (H'80 × k),
 RSCFDnCFDCFIDkH: <RSCFDn_base> + H'3402 + (H'80 × k)
 RSCFDnCFDCFIDkLL: <RSCFDn_base> + H'3400 + (H'80 × k),
 RSCFDnCFDCFIDkLH: <RSCFDn_base> + H'3401 + (H'80 × k),
 RSCFDnCFDCFIDkHL: <RSCFDn_base> + H'3402 + (H'80 × k),
 RSCFDnCFDCFIDkHH: <RSCFDn_base> + H'3403 + (H'80 × k)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 31 | CFIDE | B'0 | R/W | Transmit/Receive FIFO Buffer IDE This bit indicates the ID format (standard ID or extended ID) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is B'00. When the CFM[1:0] value is B'01, these bits are used to set the ID format of the message to be transmitted from the transmit/receive FIFO buffer. 0: Standard ID 1: Extended ID |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 30 | CFRTR | B'0 | R/W | <p>Transmit/Receive FIFO Buffer RTR/RRS</p> <p>When the CFM[1:0] bits are B'00 and the received message is in the classical CAN frame, this bit indicates the data format (data frame or remote frame) of the received message stored in the transmit/receive FIFO buffer. When the received message is in the CAN FD frame, this bit indicates the RRS bit value of the message.</p> <p>When the CFM[1:0] bits are B'01, this bit is used to set the data format of the message transmitted from the transmit/receive FIFO buffer. When the CFFDF bit in the RSCFDnCFDCFFDCSTSk register is 1 (CAN FD frame), this bit must be cleared to 0.</p> <ul style="list-style-type: none"> • When the CFM[1:0] bits are B'01 (transmit mode): <ul style="list-style-type: none"> — When the transmit message is in the classical CAN frame: <ul style="list-style-type: none"> 0: Data frame 1: Remote frame — When the transmit message is in the CAN FD frame: <ul style="list-style-type: none"> Write 0 to this bit. • When the CFM[1:0] bits are B'00 (receive mode): <ul style="list-style-type: none"> — When the received message is in the classical CAN frame: <ul style="list-style-type: none"> 0: Data frame 1: Remote frame — When the received message is in the CAN FD frame: <ul style="list-style-type: none"> The RRS bit value of the received message is read. |
| 29 | THLEN | B'0 | R/W | <p>Transmit History Data Store Enable</p> <p>When this bit is set to 1, the transmit history data (label information, buffer number, buffer type, and timestamp) of transmit messages is stored in the transmit history buffer after transmission is completed. This bit is enabled when the CFM[1:0] value is B'01 (transmit mode).</p> <p>0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.</p> |
| 28 to 0 | CFID[28:0] | H'0000_0000 | R/W | <p>Transmit/Receive FIFO Buffer ID Data</p> <ul style="list-style-type: none"> • When CFM[1:0] value is B'01 (transmit mode): <ul style="list-style-type: none"> Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11. • When CFM[1:0] value is B'00 (receive mode): <ul style="list-style-type: none"> Standard ID or extended ID in the received message can be read. For standard ID, read bits 10 to 0. Bits 28 to 11 are read as 0. |

This register is writable only when the CFM[1:0] bits in the RSCFDnCFDCFCCK register are B'01 (transmit mode). This register is readable only when the CFM[1:0] bits are B'00 (receive mode). This register should not be read or written when the CFM[1:0] bits are B'10 (gateway mode).

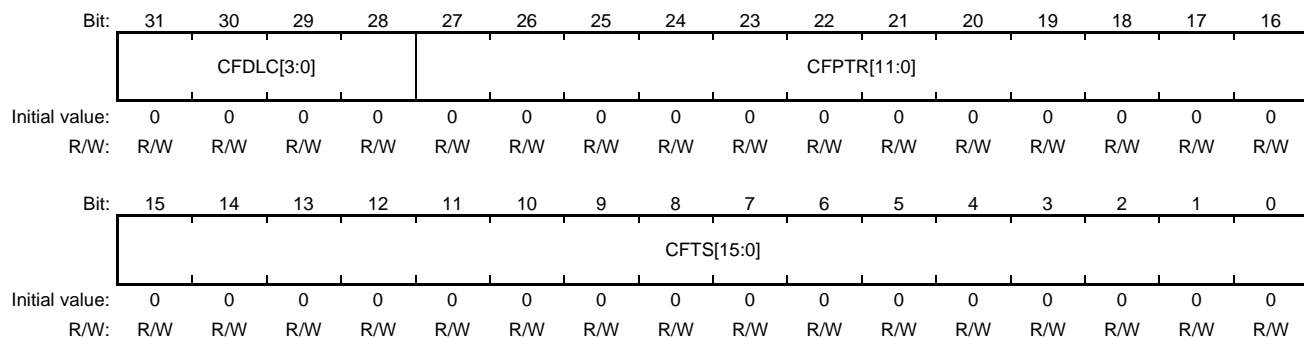
48.9.7.5 RSCFDnCFDCFPTRk — Transmit/receive FIFO Buffer Access Pointer Register (k = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCFPTRk register can be read/written in 32-bit units
 RSCFDnCFDCFPTRkL, RSCFDnCFDCFPTRkH registers can be read/written in 16-bit units
 RSCFDnCFDCFPTRkLL, RSCFDnCFDCFPTRkLH, RSCFDnCFDCFPTRkHL, RSCFDnCFDCFPTRkHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCFPTRk: <RSCFDn_base> + H'3404 + (H'80 × k)
 RSCFDnCFDCFPTRkL: <RSCFDn_base> + H'3404 + (H'80 × k),
 RSCFDnCFDCFPTRkH: <RSCFDn_base> + H'3406 + (H'80 × k)
 RSCFDnCFDCFPTRkLL: <RSCFDn_base> + H'3404 + (H'80 × k),
 RSCFDnCFDCFPTRkLH: <RSCFDn_base> + H'3405 + (H'80 × k),
 RSCFDnCFDCFPTRkHL: <RSCFDn_base> + H'3406 + (H'80 × k),
 RSCFDnCFDCFPTRkHH: <RSCFDn_base> + H'3407 + (H'80 × k)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------|---------------------|---------------|-----|---|--|---------------------|--------------|---------|--------------|--|---------|-------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|
| 31 to 28 | CFDLC[3:0] | B'0000 | R/W | <p>Transmit/Receive FIFO Buffer DLC Data</p> <p>These bits indicate the data length of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is B'00. When the CFM[1:0] value is B'01, these bits are used to set the data length of the message to be transmitted from the transmit/receive FIFO buffer. When the CFFDF bit in the RSCFDnCFDCFFDCSTSk register is 0 (CAN frame) and a value equal to or larger than B'1001 is set to the CFDLC[3:0] bits, 8-byte data is actually transmitted. When the CFFDF bit is 1 (CAN FD frame), the range of values to be set differs depending on the TMME bit in the RSCFDnCFDCmFDCFG register and the CFPLS[2:0] bits in the RSCFDnCFDCFCCK register.</p> <ul style="list-style-type: none"> When the TMME bit is 0 (the transmit buffer merge mode is disabled): All values from B'0000 to B'1111 can be set. If the data length exceeds the size for storing the payload which has been set by the CFPLS[2:0] bits, the exceeded payload will be padded by CCH. When the TMME bit is 1 (the transmit buffer merge mode is enabled): The data length must not exceed the size for storing the payload which has been set by the CFPLS[2:0] bits. <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th></th> <th>Classical CAN frame</th> <th>CAN FD frame</th> </tr> </thead> <tbody> <tr><td>B'0000:</td><td>0 data bytes</td><td></td></tr> <tr><td>B'0001:</td><td>1 data byte</td><td></td></tr> <tr><td>B'0010:</td><td>2 data bytes</td><td></td></tr> <tr><td>B'0011:</td><td>3 data bytes</td><td></td></tr> <tr><td>B'0100:</td><td>4 data bytes</td><td></td></tr> <tr><td>B'0101:</td><td>5 data bytes</td><td></td></tr> <tr><td>B'0110:</td><td>6 data bytes</td><td></td></tr> <tr><td>B'0111:</td><td>7 data bytes</td><td></td></tr> <tr><td>B'1000:</td><td>8 data bytes</td><td></td></tr> <tr><td>B'1001:</td><td>8 data bytes</td><td>12 data bytes</td></tr> <tr><td>B'1010:</td><td></td><td>16 data bytes</td></tr> <tr><td>B'1011:</td><td></td><td>20 data bytes</td></tr> <tr><td>B'1100:</td><td></td><td>24 data bytes</td></tr> <tr><td>B'1101:</td><td></td><td>32 data bytes</td></tr> <tr><td>B'1110:</td><td></td><td>48 data bytes</td></tr> <tr><td>B'1111:</td><td></td><td>64 data bytes</td></tr> </tbody> </table> | | Classical CAN frame | CAN FD frame | B'0000: | 0 data bytes | | B'0001: | 1 data byte | | B'0010: | 2 data bytes | | B'0011: | 3 data bytes | | B'0100: | 4 data bytes | | B'0101: | 5 data bytes | | B'0110: | 6 data bytes | | B'0111: | 7 data bytes | | B'1000: | 8 data bytes | | B'1001: | 8 data bytes | 12 data bytes | B'1010: | | 16 data bytes | B'1011: | | 20 data bytes | B'1100: | | 24 data bytes | B'1101: | | 32 data bytes | B'1110: | | 48 data bytes | B'1111: | | 64 data bytes |
| | Classical CAN frame | CAN FD frame | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0000: | 0 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0001: | 1 data byte | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0010: | 2 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0011: | 3 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0100: | 4 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0101: | 5 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0110: | 6 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0111: | 7 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1000: | 8 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1001: | 8 data bytes | 12 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1010: | | 16 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1011: | | 20 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1100: | | 24 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1101: | | 32 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1110: | | 48 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1111: | | 64 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 27 to 16 | CFPTR[11:0] | H'000 | R/W | <p>Transmit/Receive FIFO Buffer Label Data</p> <ul style="list-style-type: none"> When the CFM[1:0] bits are B'01 (transmit mode): When a message is transmitted, the CFPTR[7:0] bit values are stored in the transmit history. Only the CFPTR[7:0] bits are valid. When the CFM[1:0] bits are B'00 (receive mode): The label information, which has been attached to the receive message in the transmit/receive FIFO buffer, is indicated. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 15 to 0 | CFTS[15:0] | H'0000 | R/W | Transmit/Receive FIFO Buffer Timestamp Data These bits indicate the timestamp value of the message stored in the transmit/receive FIFO buffer. These bits are valid when the CFM[1:0] bits are B'00. |

This register is writable only when the CFM[1:0] bits in the RSCFDnCFDCFCCK register are B'01 (transmit mode). This register is readable only when the CFM[1:0] bits are B'00 (receive mode). This register should not be read or written when the CFM[1:0] bits are B'10 (gateway mode).

48.9.7.6 RSCFDnCFDCFFDCSTSk — Transmit/receive FIFO CAN FD Configuration and Status Register (k = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCFFDCSTSk register can be read/written in 32-bit units

RSCFDnCFDCFFDCSTSkL, RSCFDnCFDCFFDCSTSkH registers can be read/written in 16-bit units

RSCFDnCFDCFFDCSTSkLL, RSCFDnCFDCFFDCSTSkLH, RSCFDnCFDCFFDCSTSkHL,

RSCFDnCFDCFFDCSTSkHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCFFDCSTSk: <RSCFDn_base> + H'3408 + (H'80 × k)

RSCFDnCFDCFFDCSTSkL: <RSCFDn_base> + H'3408 + (H'80 × k),

RSCFDnCFDCFFDCSTSkH: <RSCFDn_base> + H'340A + (H'80 × k)

RSCFDnCFDCFFDCSTSkLL: <RSCFDn_base> + H'3408 + (H'80 × k),

RSCFDnCFDCFFDCSTSkLH: <RSCFDn_base> + H'3409 + (H'80 × k),

RSCFDnCFDCFFDCSTSkHL: <RSCFDn_base> + H'340A + (H'80 × k),

RSCFDnCFDCFFDCSTSkHH: <RSCFDn_base> + H'340B + (H'80 × k)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | CFFDF | CFBRS | CFESI |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 3 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 2 | CFFDF | B'0 | R/W | <p>CFFDF</p> <p>When the CFM[1:0] bits are B'00, this bit indicates the FD format (classical CAN frame or CAN FD frame) of the message stored in the transmit/receive FIFO buffer. When the CFM[1:0] bits are B'01, this bit is used to set the FD format of the message to be transmitted from the transmit/receive FIFO buffer.</p> <p>0: Classical CAN frame 1: CAN FD frame</p> |
| 1 | CFBRS | B'0 | R/W | <p>CFBRS</p> <p>When the CFM[1:0] bits are B'00, this bit indicates the BRS bit value of the receive message stored in the transmit/receive FIFO buffer if the CFFDF bit is 1. This bit is always read as 0 if the CFFDF bit is 0.</p> <p>When the CFM[1:0] bits are B'01, this bit is used to set the BRS bit value of the message transmitted from the transmit/receive FIFO buffer if the CFFDF bit is 1. Write 0 to this bit if the CFFDF bit is 0.</p> <p>0: The bit rate in the data area is not changed. 1: The bit rate in the data area is changed.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | CFESI | B'0 | R/W | <p>CFESI</p> <p>When the CFM[1:0] bits are B'00, this bit indicates the ESI bit value of the receive message stored in the transmit/receive FIFO buffer if the CFFDF bit is 1. This bit is always read as 0 if the CFFDF bit is 0.</p> <p>When the CFM[1:0] bits are B'01, this bit is used to set the ESI bit value of the message transmitted from the transmit/receive FIFO buffer if the CFFDF bit is 1. The set value is transmitted when the ESIC bit in the RSCFDnCFDCmFDCFG register is 1 and the channel is in error-active state. When the channel is in error-passive state, the ESI bit value indicating the error-passive node is transmitted regardless of this bit. Write 0 to this bit if the CFFDF bit is 0.</p> <p>0: Error-active node 1: Error-passive node</p> |

This register is writable only when the CFM[1:0] bits in the RSCFDnCFDCFCCK register are B'01 (transmit mode). This register is readable only when the CFM[1:0] bits are B'00 (receive mode). This register should not be read or written when the CFM[1:0] bits are B'10 (gateway mode).

48.9.7.7 RSCFDnCFDCFDf_k — Transmit/receive FIFO Buffer Access Data Field d Register (d = 0 to 15, k = 0 to 5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCFDf_k register can be read/written in 32-bit units

RSCFDnCFDCFDf_kL, RSCFDnCFDCFDf_kH registers can be read/written in 16-bit units

RSCFDnCFDCFDf_kLL, RSCFDnCFDCFDf_kLH, RSCFDnCFDCFDf_kHL, RSCFDnCFDCFDf_kHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCFDf_k: <RSCFDn_base> + H'340C + (H'04 × d) + (H'80 × k)

RSCFDnCFDCFDf_kL: <RSCFDn_base> + H'340C + (H'04 × d) + (H'80 × k),

RSCFDnCFDCFDf_kH: <RSCFDn_base> + H'340E + (H'04 × d) + (H'80 × k)

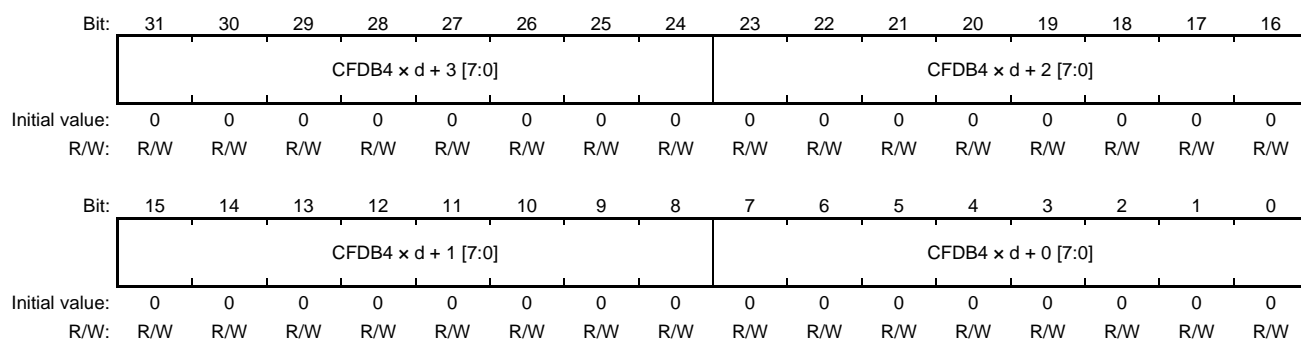
RSCFDnCFDCFDf_kLL: <RSCFDn_base> + H'340C + (H'04 × d) + (H'80 × k),

RSCFDnCFDCFDf_kLH: <RSCFDn_base> + H'340D + (H'04 × d) + (H'80 × k),

RSCFDnCFDCFDf_kHL: <RSCFDn_base> + H'340E + (H'04 × d) + (H'80 × k),

RSCFDnCFDCFDf_kHH: <RSCFDn_base> + H'340F + (H'04 × d) + (H'80 × k)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|---|
| 31 to 24 | CFDB4 × d + 3 [7:0] | H'00 | R/W | Transmit/Receive FIFO Buffer Data Byte 4 × d + 3 Transmit/Receive FIFO Buffer Data Byte 4 × d + 2 |
| 23 to 16 | CFDB4 × d + 2 [7:0] | H'00 | R/W | Transmit/Receive FIFO Buffer Data Byte 4 × d + 1 Transmit/Receive FIFO Buffer Data Byte 4 × d + 0 |
| 15 to 8 | CFDB4 × d + 1 [7:0] | H'00 | R/W | • When CFM[1:0] value is B'01 (transmit mode): Set the transmit/receive FIFO buffer data. |
| 7 to 0 | CFDB4 × d + 0 [7:0] | H'00 | R/W | • When CFM[1:0] value is B'00 (receive mode): The message data stored in the transmit/receive FIFO buffer can be read. |

This register is writable only when the CFM[1:0] bits in the RSCFDnCFDCFCCK register are B'01.

This register is readable only when the CFM[1:0] bits are B'00. When the CFDLC[3:0] bit values in the RSCFDnCFDCFPTRk register are smaller than the size for storing the payload in the transmit/receive FIFO buffer, data bytes for which no data is set are read as H'00.

The size for storing the payload in the transmit/receive FIFO buffer is specified with the CFPLS[2:0] bits in the RSCFDnCFDCFCCK register. The RSCFDnCFDCFDf_k register, which corresponds to the larger area than the specified size, should not be read or written.

This register should not be read or written when the CFM[1:0] bits are B'10 (gateway mode).

48.9.8 Details of FIFO Status Related Registers

48.9.8.1 RSCFDnCFDFESTS — FIFO Empty Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDFESTS register can be read only in 32-bit units

RSCFDnCFDFESTSL, RSCFDnCFDFESTSH registers can be read only in 16-bit units

RSCFDnCFDFESTSLL, RSCFDnCFDFESTSLH, RSCFDnCFDFESTSHL, RSCFDnCFDFESTSHH registers can be read only in 8-bit units

Address: RSCFDnCFDFESTS: <RSCFDn_base> + H'0238

RSCFDnCFDFESTSL: <RSCFDn_base> + H'0238, RSCFDnCFDFESTSH: <RSCFDn_base> + H'023A

RSCFDnCFDFESTSLL: <RSCFDn_base> + H'0238, RSCFDnCFDFESTSLH: <RSCFDn_base> + H'0239,

RSCFDnCFDFESTSHL: <RSCFDn_base> + H'023A, RSCFDnCFDFESTSHH: <RSCFDn_base> + H'023B

Value after reset: H'03FF_FFFF

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | CF17E MP | CF16E MP | CF15E MP | CF14E MP | CF13E MP | CF12E MP | CF11E MP | CF10E MP | CF9EM P | CF8EM P |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CF7EM P | CF6EM P | CF5EM P | CF4EM P | CF3EM P | CF2EM P | CF1EM P | CF0EM P | RF7EM P | RF6EM P | RF5EM P | RF4EM P | RF3EM P | RF2EM P | RF1EM P | RF0EM P |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 26 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 25 | CF17EMP | B'1 | R | Transmit/Receive FIFO Buffer Empty Status Flag |
| 24 | CF16EMP | B'1 | R | The CFkEMP flag is set to 1 when the CFEMP flag in the RSCFDnCFDFESTS register is set to 1 (the transmit/receive FIFO buffer contains no message). When the CFEMP flag is cleared to 0 (the transmit/receive FIFO buffer contains messages), the CFkEMP flag is cleared to 0. |
| 23 | CF15EMP | B'1 | R | |
| 22 | CF14EMP | B'1 | R | |
| 21 | CF13EMP | B'1 | R | |
| 20 | CF12EMP | B'1 | R | |
| 19 | CF11EMP | B'1 | R | 0: Transmit/receive FIFO buffer k contains a message. 1: Transmit/receive FIFO buffer k contains no message. (k = 0 to 5) |
| 18 | CF10EMP | B'1 | R | |
| 17 | CF9EMP | B'1 | R | |
| 16 | CF8EMP | B'1 | R | |
| 15 | CF7EMP | B'1 | R | |
| 14 | CF6EMP | B'1 | R | |
| 13 | CF5EMP | B'1 | R | |
| 12 | CF4EMP | B'1 | R | |
| 11 | CF3EMP | B'1 | R | |
| 10 | CF2EMP | B'1 | R | |
| 9 | CF1EMP | B'1 | R | |
| 8 | CF0EMP | B'1 | R | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | RF7EMP | B'1 | R | Receive FIFO Buffer Empty Status Flag |
| 6 | RF6EMP | B'1 | R | The RFxEMP flag is set to 1 when the RFEMP flag in the RSCFDnCFDRFSTSx register is set to 1 (the receive FIFO buffer contains no unread message). When the RFEMP flag is cleared to 0 (the receive FIFO buffer contains unread messages), the RFxEMP flag is cleared to 0. 0: Receive FIFO buffer x contains an unread message. 1: Receive FIFO buffer x contains no unread message. (x = 0 to 7) |
| 5 | RF5EMP | B'1 | R | |
| 4 | RF4EMP | B'1 | R | |
| 3 | RF3EMP | B'1 | R | |
| 2 | RF2EMP | B'1 | R | |
| 1 | RF1EMP | B'1 | R | |
| 0 | RF0EMP | B'1 | R | |

The RSCFDnCFDFESTS register is set to H'03FF_FFFF in global reset mode.

48.9.8.2 RSCFDnCFDFFSTS — FIFO Full Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDFFSTS register can be read only in 32-bit units

RSCFDnCFDFFSTSL, RSCFDnCFDFFSTSH registers can be read only in 16-bit units

RSCFDnCFDFFSTSL, RSCFDnCFDFFSTSLH, RSCFDnCFDFFSTSHL, RSCFDnCFDFFSTSHH registers can be read only in 8-bit units

Address: RSCFDnCFDFFSTS: <RSCFDn_base> + H'023C

RSCFDnCFDFFSTSL: <RSCFDn_base> + H'023C, RSCFDnCFDFFSTSH: <RSCFDn_base> + H'023E

RSCFDnCFDFFSTSL: <RSCFDn_base> + H'023C, RSCFDnCFDFFSTSLH: <RSCFDn_base> + H'023D,

RSCFDnCFDFFSTSHL: <RSCFDn_base> + H'023E, RSCFDnCFDFFSTSHH: <RSCFDn_base> + H'023F

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | CF17F LL | CF16F LL | CF15F LL | CF14F LL | CF13F LL | CF12F LL | CF11F LL | CF10F LL | CF9FL L | CF8FL L |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CF7FL L | CF6FL L | CF5FL L | CF4FL L | CF3FL L | CF2FL L | CF1FL L | CF0FL L | RF7FL L | RF6FL L | RF5FL L | RF4FL L | RF3FL L | RF2FL L | RF1FL L | RF0FL L |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 26 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 25 | CF17FLL | B'0 | R | Transmit/Receive FIFO Buffer Full Status Flag |
| 24 | CF16FLL | B'0 | R | The CFkFLL flag is set to 1 when the CFFLL flag in the RSCFDnCFDFFSTS register is set to 1 (the transmit/receive FIFO buffer is full). |
| 23 | CF15FLL | B'0 | R | |
| 22 | CF14FLL | B'0 | R | When the CFFLL flag is cleared to 0 (the transmit/receive FIFO buffer is not full), the CFkFLL flag is cleared to 0. |
| 21 | CF13FLL | B'0 | R | |
| 20 | CF12FLL | B'0 | R | 0: Transmit/receive buffer k is not full. |
| 19 | CF11FLL | B'0 | R | 1: Transmit/receive buffer k is full. |
| 18 | CF10FLL | B'0 | R | (k = 0 to 5) |
| 17 | CF9FLL | B'0 | R | |
| 16 | CF8FLL | B'0 | R | |
| 15 | CF7FLL | B'0 | R | |
| 14 | CF6FLL | B'0 | R | |
| 13 | CF5FLL | B'0 | R | |
| 12 | CF4FLL | B'0 | R | |
| 11 | CF3FLL | B'0 | R | |
| 10 | CF2FLL | B'0 | R | |
| 9 | CF1FLL | B'0 | R | |
| 8 | CF0FLL | B'0 | R | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | RF7FLL | B'0 | R | Receive FIFO Buffer Full Status Flag |
| 6 | RF6FLL | B'0 | R | The RFXFLL flag is set to 1 when the RFFLL flag in the RSCFDnCFDRFSTSx register is set to 1 (the receive FIFO buffer is full). When the RFFLL flag is cleared to 0 (the receive FIFO buffer is not full), the RFXFLL flag is cleared to 0. |
| 5 | RF5FLL | B'0 | R | |
| 4 | RF4FLL | B'0 | R | 0: Receive FIFO buffer x is not full. 1: Receive FIFO buffer x is full. (x = 0 to 7) |
| 3 | RF3FLL | B'0 | R | |
| 2 | RF2FLL | B'0 | R | |
| 1 | RF1FLL | B'0 | R | |
| 0 | RF0FLL | B'0 | R | |

The RSCFDnCFDRFSTS register is cleared to H'0000_0000 in global reset mode.

48.9.8.3 RSCFDnCFDFMSTS — FIFO Message Lost Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDFMSTS register can be read only in 32-bit units

RSCFDnCFDFMSTSL, RSCFDnCFDFMSTSH registers can be read only in 16-bit units

RSCFDnCFDFMSTSL, RSCFDnCFDFMSTSLH, RSCFDnCFDFMSTSHL, RSCFDnCFDFMSTSHH registers can be read only in 8-bit units

Address: RSCFDnCFDFMSTS: <RSCFDn_base> + H'0240

RSCFDnCFDFMSTSL: <RSCFDn_base> + H'0240, RSCFDnCFDFMSTSH: <RSCFDn_base> + H'0242

RSCFDnCFDFMSTSL: <RSCFDn_base> + H'0240, RSCFDnCFDFMSTSLH: <RSCFDn_base> + H'0241,

RSCFDnCFDFMSTSHL: <RSCFDn_base> + H'0242, RSCFDnCFDFMSTSHH: <RSCFDn_base> + H'0243

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|------------|------------|------------|------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | CF17M LT | CF16M LT | CF15M LT | CF14M LT | CF13M LT | CF12M LT | CF11M LT | CF10M LT | CF9ML T | CF8ML T |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CF7ML T | CF6ML T | CF5ML T | CF4ML T | CF3ML T | CF2ML T | CF1ML T | CF0ML T | RF7ML T | RF6ML T | RF5ML T | RF4ML T | RF3ML T | RF2ML T | RF1ML T | RF0ML T |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 26 | Reserved | All 0 | R/W | These bits are read as the value after reset. |
| 25 | CF17MLT | B'0 | R | Transmit/Receive FIFO Buffer Message Lost Status Flag |
| 24 | CF16MLT | B'0 | R | The CFkMLT flag is set to 1 when the CFMLT flag in the RSCFDnCFDFMSTSL register is set to 1 (a transmit/receive FIFO message is lost). When the CFMLT flag is cleared to 0, the CFkMLT flag is cleared to 0. |
| 23 | CF15MLT | B'0 | R | |
| 22 | CF14MLT | B'0 | R | |
| 21 | CF13MLT | B'0 | R | 0: No transmit/receive FIFO buffer k message is lost. |
| 20 | CF12MLT | B'0 | R | 1: A transmit/receive FIFO buffer k message is lost. |
| 19 | CF11MLT | B'0 | R | (k = 0 to 5) |
| 18 | CF10MLT | B'0 | R | |
| 17 | CF9MLT | B'0 | R | |
| 16 | CF8MLT | B'0 | R | |
| 15 | CF7MLT | B'0 | R | |
| 14 | CF6MLT | B'0 | R | |
| 13 | CF5MLT | B'0 | R | |
| 12 | CF4MLT | B'0 | R | |
| 11 | CF3MLT | B'0 | R | |
| 10 | CF2MLT | B'0 | R | |
| 9 | CF1MLT | B'0 | R | |
| 8 | CF0MLT | B'0 | R | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | RF7MLT | B'0 | R | Receive FIFO Buffer Message Lost Status Flag |
| 6 | RF6MLT | B'0 | R | The RFxMLT flag is set to 1 when the RFMLT flag in the RSCFDnCFDRFSTSx register is set to 1 (a receive FIFO message is lost). When the RFMLT flag is cleared to 0, the RFxMLT flag is cleared to 0. |
| 5 | RF5MLT | B'0 | R | |
| 4 | RF4MLT | B'0 | R | 0: No receive FIFO buffer x message is lost. 1: A receive FIFO buffer x message is lost. (x = 0 to 7) |
| 3 | RF3MLT | B'0 | R | |
| 2 | RF2MLT | B'0 | R | |
| 1 | RF1MLT | B'0 | R | |
| 0 | RF0MLT | B'0 | R | |

The RSCFDnCFDFMSTS register is cleared to H'0000_0000 in global reset mode.

48.9.8.4 RSCFDnCFDRFISTS — Receive FIFO Buffer Interrupt Flag Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRFISTS register can be read only in 32-bit units

RSCFDnCFDRFISTS_L, RSCFDnCFDRFISTS_H registers can be read only in 16-bit units

RSCFDnCFDRFISTS_{LL}, RSCFDnCFDRFISTS_{LH}, RSCFDnCFDRFISTS_{SHL}, RSCFDnCFDRFISTS_{SHH} registers can be read only in 8-bit units

Address: RSCFDnCFDRFISTS: <RSCFDn_base> + H'0244

RSCFDnCFDRFISTS_L: <RSCFDn_base> + H'0244, RSCFDnCFDRFISTS_H: <RSCFDn_base> + H'0246

RSCFDnCFDRFISTS_{LL}: <RSCFDn_base> + H'0244, RSCFDnCFDRFISTS_{LH}: <RSCFDn_base> + H'0245,

RSCFDnCFDRFISTS_{SHL}: <RSCFDn_base> + H'0246, RSCFDnCFDRFISTS_{SHH}: <RSCFDn_base> + H'0247

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | RF7IF | RF6IF | RF5IF | RF4IF | RF3IF | RF2IF | RF1IF | RF0IF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | Reserved | All 0 | R/W | These bits are read as the value after reset. |
| 7 | RF7IF | B'0 | R | Receive FIFO Buffer Interrupt Request Status Flag |
| 6 | RF6IF | B'0 | R | The RFxIF flag is set to 1 when the RFIF flag in the RSCFDnCFDRFISTS _x register is set to 1 (a receive FIFO interrupt request is present). When the RFIF flag is cleared to 0, the RFxIF flag is cleared to 0. |
| 5 | RF5IF | B'0 | R | |
| 4 | RF4IF | B'0 | R | |
| 3 | RF3IF | B'0 | R | 0: No receive FIFO buffer x interrupt request is present. |
| 2 | RF2IF | B'0 | R | 1: A receive FIFO buffer x interrupt request is present. |
| 1 | RF1IF | B'0 | R | (x = 0 to 7) |
| 0 | RF0IF | B'0 | R | |

The RSCFDnCFDRFISTS register is cleared to H'0000_0000 in global reset mode.

48.9.8.5 RSCFDnCFDCFRISTS — Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCFRISTS register can be read only in 32-bit units

RSCFDnCFDCFRISTS_L, RSCFDnCFDCFRISTS_H registers can be read only in 16-bit units

RSCFDnCFDCFRISTS_{LL}, RSCFDnCFDCFRISTS_{SLH}, RSCFDnCFDCFRISTS_{SHL}, RSCFDnCFDCFRISTS_{SHH} registers can be read only in 8-bit units

Address: RSCFDnCFDCFRISTS: <RSCFDn_base> + H'0248

RSCFDnCFDCFRISTS_L: <RSCFDn_base> + H'0248, RSCFDnCFDCFRISTS_H: <RSCFDn_base> + H'024A

RSCFDnCFDCFRISTS_{LL}: <RSCFDn_base> + H'0248, RSCFDnCFDCFRISTS_{SLH}: <RSCFDn_base> + H'0249,

RSCFDnCFDCFRISTS_{SHL}: <RSCFDn_base> + H'024A, RSCFDnCFDCFRISTS_{SHH}: <RSCFDn_base> + H'024B

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|--------------|--------------|--------------|--------------|--------------|--------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CF17R XIF | CF16R XIF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CF15R XIF | CF14R XIF | CF13R XIF | CF12R XIF | CF11R XIF | CF10R XIF | CF9RXI F | CF8RXI F | CF7RXI F | CF6RXI F | CF5RXI F | CF4RXI F | CF3RXI F | CF2RXI F | CF1RXI F | CF0RXI F |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 18 | Reserved | All 0 | R/W | These bits are read as the value after reset. |
| 17 | CF17RXIF | B'0 | R | Transmit/Receive FIFO Buffer Receive Interrupt Request Status Flag |
| 16 | CF16RXIF | B'0 | R | |
| 15 | CF15RXIF | B'0 | R | The CFkRXIF flag is set to 1 when the CFRXIF flag in the RSCFDnCFDCFRISTS _k register is set to 1 (a transmit/receive FIFO receive interrupt request is present). When the CFRXIF flag is cleared to 0, the CFkRXIF flag is cleared to 0. |
| 14 | CF14RXIF | B'0 | R | |
| 13 | CF13RXIF | B'0 | R | |
| 12 | CF12RXIF | B'0 | R | 0: No transmit/receive FIFO buffer k receive interrupt request is present. |
| 11 | CF11RXIF | B'0 | R | 1: A transmit/receive FIFO buffer k receive interrupt request is present. |
| 10 | CF10RXIF | B'0 | R | |
| 9 | CF9RXIF | B'0 | R | (k = 0 to 5) |
| 8 | CF8RXIF | B'0 | R | |
| 7 | CF7RXIF | B'0 | R | |
| 6 | CF6RXIF | B'0 | R | |
| 5 | CF5RXIF | B'0 | R | |
| 4 | CF4RXIF | B'0 | R | |
| 3 | CF3RXIF | B'0 | R | |
| 2 | CF2RXIF | B'0 | R | |
| 1 | CF1RXIF | B'0 | R | |
| 0 | CF0RXIF | B'0 | R | |

The RSCFDnCFDCFRISTS register is cleared to H'0000_0000 in global reset mode.

48.9.8.6 RSCFDnCFDCFTISTS — Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCFTISTS register can be read only in 32-bit units

RSCFDnCFDCFTISTS_L, RSCFDnCFDCFTISTS_H registers can be read only in 16-bit units

RSCFDnCFDCFTISTS_{LL}, RSCFDnCFDCFTISTS_{LH}, RSCFDnCFDCFTISTS_{HL}, RSCFDnCFDCFTISTS_{HH} registers can be read only in 8-bit units

Address: RSCFDnCFDCFTISTS: <RSCFDn_base> + H'024C

RSCFDnCFDCFTISTS_L: <RSCFDn_base> + H'024C, RSCFDnCFDCFTISTS_H: <RSCFDn_base> + H'024E

RSCFDnCFDCFTISTS_{LL}: <RSCFDn_base> + H'024C, RSCFDnCFDCFTISTS_{LH}: <RSCFDn_base> + H'024D,

RSCFDnCFDCFTISTS_{HL}: <RSCFDn_base> + H'024E, RSCFDnCFDCFTISTS_{HH}: <RSCFDn_base> + H'024F

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|--------------|--------------|--------------|--------------|--------------|--------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CF17T XIF | CF16T XIF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CF15T XIF | CF14T XIF | CF13T XIF | CF12T XIF | CF11T XIF | CF10T XIF | CF9TXI F | CF8TXI F | CF7TXI F | CF6TXI F | CF5TXI F | CF4TXI F | CF3TXI F | CF2TXI F | CF1TXI F | CF0TXI F |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 18 | Reserved | All 0 | R | These bits are read as the value after reset. |
| 17 | CF17TXIF | B'0 | R | Transmit/Receive FIFO Buffer Transmit Interrupt Request Status Flag |
| 16 | CF16TXIF | B'0 | R | |
| 15 | CF15TXIF | B'0 | R | The CFkTXIF flag is set to 1 when the CFTXIF flag in the RSCFDnCFDCFSTSk register is set to 1 (a transmit/receive FIFO transmit interrupt request is present). When the CFTXIF flag is cleared to 0, the CFkTXIF flag is cleared to 0. |
| 14 | CF14TXIF | B'0 | R | |
| 13 | CF13TXIF | B'0 | R | 0: No transmit/receive FIFO buffer k transmit interrupt request is present. 1: A transmit/receive FIFO buffer k transmit interrupt request is present. |
| 12 | CF12TXIF | B'0 | R | |
| 11 | CF11TXIF | B'0 | R | (k = 0 to 5) |
| 10 | CF10TXIF | B'0 | R | |
| 9 | CF9TXIF | B'0 | R | |
| 8 | CF8TXIF | B'0 | R | |
| 7 | CF7TXIF | B'0 | R | |
| 6 | CF6TXIF | B'0 | R | |
| 5 | CF5TXIF | B'0 | R | |
| 4 | CF4TXIF | B'0 | R | |
| 3 | CF3TXIF | B'0 | R | |
| 2 | CF2TXIF | B'0 | R | |
| 1 | CF1TXIF | B'0 | R | |
| 0 | CF0TXIF | B'0 | R | |

The RSCFDnCFDCFTISTS register is cleared to H'0000_0000 in global reset mode.

48.9.9 Details of FIFO DMA Related Registers

48.9.9.1 RSCFDnCFDCDTCT — DMA Enable Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCDTCT register can be read/written in 32-bit units

RSCFDnCFDCDTCTL, RSCFDnCFDCDTCTH registers can be read/written in 16-bit units

RSCFDnCFDCDTCTL, RSCFDnCFDCDTCTLH, RSCFDnCFDCDTCTHL, RSCFDnCFDCDTCTHH registers can be read/written in 8-bit units

Address: RSCFDnCFDCDTCT: <RSCFDn_base> + H'0490

RSCFDnCFDCDTCTL: <RSCFDn_base> + H'0490

RSCFDnCFDCDTCTH: <RSCFDn_base> + H'0492

RSCFDnCFDCDTCTL: <RSCFDn_base> + H'0490,

RSCFDnCFDCDTCTLH: <RSCFDn_base> + H'0491,

RSCFDnCFDCDTCTHL: <RSCFDn_base> + H'0492,

RSCFDnCFDCDTCTHH: <RSCFDn_base> + H'0493

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | CFDMA E5 | CFDMA E4 | CFDMA E3 | CFDMA E2 | CFDMA E1 | CFDMA E0 | RFDMA E7 | RFDMA E6 | RFDMA E5 | RFDMA E4 | RFDMA E3 | RFDMA E2 | RFDMA E1 | RFDMA E0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 14 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 13 | CFDMAE5 | B'0 | R/W | Transmit/Receive FIFO Buffer 15 DMA Enable 0: The DMA transfer request for transmit/receive FIFO buffer 15 is disabled. 1: The DMA transfer request for transmit/receive FIFO buffer 15 is enabled. |
| 12 | CFDMAE4 | B'0 | R/W | Transmit/Receive FIFO Buffer 12 DMA Enable 0: The DMA transfer request for transmit/receive FIFO buffer 12 is disabled. 1: The DMA transfer request for transmit/receive FIFO buffer 12 is enabled. |
| 11 | CFDMAE3 | B'0 | R/W | Transmit/Receive FIFO Buffer 9 DMA Enable 0: The DMA transfer request for transmit/receive FIFO buffer 9 is disabled. 1: The DMA transfer request for transmit/receive FIFO buffer 9 is enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 10 | CFDMAE2 | B'0 | R/W | Transmit/Receive FIFO Buffer 6 DMA Enable 0: The DMA transfer request for transmit/receive FIFO buffer 6 is disabled. 1: The DMA transfer request for transmit/receive FIFO buffer 6 is enabled. |
| 9 | CFDMAE1 | B'0 | R/W | Transmit/Receive FIFO Buffer 3 DMA Enable 0: The DMA transfer request for transmit/receive FIFO buffer 3 is disabled. 1: The DMA transfer request for transmit/receive FIFO buffer 3 is enabled. |
| 8 | CFDMAE0 | B'0 | R/W | Transmit/Receive FIFO Buffer 0 DMA Enable 0: The DMA transfer request for transmit/receive FIFO buffer 0 is disabled. 1: The DMA transfer request for transmit/receive FIFO buffer 0 is enabled. |
| 7 | RFDMAE7 | B'0 | R/W | Receive FIFO Buffer x DMA Enable |
| 6 | RFDMAE6 | B'0 | R/W | These bits enable the DMA transfer for receive FIFO buffer x. |
| 5 | RFDMAE5 | B'0 | R/W | 0: The DMA transfer request for receive FIFO buffer x is disabled. |
| 4 | RFDMAE4 | B'0 | R/W | 1: The DMA transfer request for receive FIFO buffer x is enabled. |
| 3 | RFDMAE3 | B'0 | R/W | (x = 0 to 7) |
| 2 | RFDMAE2 | B'0 | R/W | |
| 1 | RFDMAE1 | B'0 | R/W | |
| 0 | RFDMAE0 | B'0 | R/W | |

Modify the RSCFDnCFDCDTCT register only in global operating mode or global test mode.

CFDMAEm Bits

The DMA transfer for transmit/receive FIFO buffer $3 \times m$ (the first transmit/receive FIFO buffer assigned to channel m) is enabled. The DMA transfer is only enabled for the transmit/receive FIFO buffer in which the CFM[1:0] bits in the RSCFDnCFDCFCCK register have been set to B'00 (receive mode). Set these bits to 0 when the CFM[1:0] bits are B'01 (transmit mode) or B'10 (gateway mode).

48.9.9.2 RSCFDnCFDCDTSTS — DMA Status Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDCDTSTS register can be read only in 32-bit units

RSCFDnCFDCDTSTSL, RSCFDnCFDCDTSTSH registers can be read only in 16-bit units

RSCFDnCFDCDTSTSL, RSCFDnCFDCDTSTSLH, RSCFDnCFDCDTSTSHL, RSCFDnCFDCDTSTSHH registers can be read only in 8-bit units

Address: RSCFDnCFDCDTSTS: <RSCFDn_base> + H'0494

RSCFDnCFDCDTSTSL: <RSCFDn_base> + H'0494,

RSCFDnCFDCDTSTSH: <RSCFDn_base> + H'0496

RSCFDnCFDCDTSTSL: <RSCFDn_base> + H'0494,

RSCFDnCFDCDTSTSLH: <RSCFDn_base> + H'0495,

RSCFDnCFDCDTSTSHL: <RSCFDn_base> + H'0496,

RSCFDnCFDCDTSTSHH: <RSCFDn_base> + H'0497

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | CFDMA STS5 | CFDMA STS4 | CFDMA STS3 | CFDMA STS2 | CFDMA STS1 | CFDMA STS0 | RFDMA STS7 | RFDMA STS6 | RFDMA STS5 | RFDMA STS4 | RFDMA STS3 | RFDMA STS2 | RFDMA STS1 | RFDMA STS0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 14 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 13 | CFDMASTS5 | B'0 | R | Transmit/Receive FIFO Buffer 15 DMA Status 0: The DMA for transmit/receive FIFO buffer 15 is not being transferred. 1: The DMA for transmit/receive FIFO buffer 15 is being transferred. |
| 12 | CFDMASTS4 | B'0 | R | Transmit/Receive FIFO Buffer 12 DMA Status 0: The DMA for transmit/receive FIFO buffer 12 is not being transferred. 1: The DMA for transmit/receive FIFO buffer 12 is being transferred. |
| 11 | CFDMASTS3 | B'0 | R | Transmit/Receive FIFO Buffer 9 DMA Status 0: The DMA for transmit/receive FIFO buffer 9 is not being transferred. 1: The DMA for transmit/receive FIFO buffer 9 is being transferred. |
| 10 | CFDMASTS2 | B'0 | R | Transmit/Receive FIFO Buffer 6 DMA Status 0: The DMA for transmit/receive FIFO buffer 6 is not being transferred. 1: The DMA for transmit/receive FIFO buffer 6 is being transferred. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------|---------------|-----|---|
| 9 | CFDMASTS1 | B'0 | R | Transmit/Receive FIFO Buffer 3 DMA Status 0: The DMA for transmit/receive FIFO buffer 3 is not being transferred. 1: The DMA for transmit/receive FIFO buffer 3 is being transferred. |
| 8 | CFDMASTS0 | B'0 | R | Transmit/Receive FIFO Buffer 0 DMA Status 0: The DMA for transmit/receive FIFO buffer 0 is not being transferred. 1: The DMA for transmit/receive FIFO buffer 0 is being transferred. |
| 7 | RFDMASTS7 | B'0 | R | Receive FIFO Buffer x DMA Status |
| 6 | RFDMASTS6 | B'0 | R | When the DMA transfer for receive FIFO buffer x is enabled (the RFDMAEx bit corresponding to the RSCFDnCFDCDTCT register is 1) and there is a message in the receive FIFO buffer, the RFDMAEx bit becomes 1 and indicates that the DMA is being transferred. When all messages in receive FIFO buffer x are transferred or the DMA transfer is disabled (the RFDMAEx bit is 0), the RFDMASTSx bit becomes 0 and indicates that the DMA transfer completed. When the RFDMAEx bit is cleared to 0 during DMA transfer, the RFDMASTSx bit becomes 0 after the DMA transfer in execution has completed (the message has been transferred to the last byte in the payload storage area). 0: The DMA for receive FIFO buffer x is not being transferred. 1: The DMA for receive FIFO buffer x is being transferred. (x = 0 to 7) |
| 5 | RFDMASTS5 | B'0 | R | |
| 4 | RFDMASTS4 | B'0 | R | |
| 3 | RFDMASTS3 | B'0 | R | |
| 2 | RFDMASTS2 | B'0 | R | |
| 1 | RFDMASTS1 | B'0 | R | |
| 0 | RFDMASTS0 | B'0 | R | |

CFDMASTS_m Bits

When the DMA transfer for transmit/receive FIFO buffer $3 \times m$ (the first transmit/receive FIFO buffer assigned to channel m) is enabled (the CFDMAEm bit in the RSCFDnCFDCDTCT register is 1) and there is a message in the transmit/receive FIFO buffer, the CFDMASTS_m bit becomes 1 and indicates that the DMA is being transferred.

When all messages in the transmit/receive FIFO buffer are transferred or the DMA transfer is disabled (the CFDMAEm bit is 0), the CFDMASTS_m bit becomes 0 and indicates that the DMA transfer completed. When the CFDMAEm bit is cleared to 0 during DMA transfer, the CFDMASTS_m bit becomes 0 after the DMA transfer in execution has completed (the message has been transferred to the last byte in the payload storage area).

48.9.10 Details of Transmit Buffer Related Registers

48.9.10.1 RSCFDnCFDTMCp — Transmit Buffer Control Register (p = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDTMCp register can be read/written in 8-bit units

Address: RSCFDnCFDTMCp: <RSCFDn_base> + H'0250 + (H'01 × p)

Value after reset: H'00

| | | | | | | | | |
|------|---|---|---|---|---|------|-------|------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | TMOM | TMTAR | TMTR |

Initial value: 0 0 0 0 0 0 0 0

R/W: R R R R R R/W R/W* R/W*

Note: * The only effective value for writing to this bit is 1, which sets the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 to 3 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 2 | TMOM | B'0 | R/W | One-Shot Transmission Enable Setting this bit to 1 enables one-shot transmission. When transmission fails, retransmission defined in the CAN protocol is not performed. Modify the TMOM bit when the TMTRM flag in the RSCFDnCFDTMSTSp register is set to 0. Set the TMOM bit to 1 together with the TMTR bit. 0: One-shot transmission is disabled. 1: One-shot transmission is enabled. |
| 1 | TMTAR | B'0 | R/W | Transmit Abort Request Setting this bit to 1 generates a transmit abort request for the message stored in the transmit buffer. However, a message that is being transmitted or one that will be transmitted next cannot be aborted. The TMTAR bit can be set to 1 when TMTR bit is 1. The TMTAR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit. <ul style="list-style-type: none"> • Transmission has been completed. • Transmit abort has been completed. • An error or arbitration loss has been detected. If this bit becomes 0 at the same time as the program writes 1 to this bit, this bit becomes 0. 0: Transmit abort is not requested. 1: Transmit abort is requested. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | TMTR | B'0 | R/W | <p>Transmit Request</p> <p>Setting this bit to 1 transmits the message stored in the transmit buffer.</p> <p>The TMTR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.</p> <ul style="list-style-type: none"> • Transmission has been completed. • Transmit abort has been completed after the TMTAR bit was set to 1. • An error or arbitration-lost has been detected with the TMOM bit set to 1. <p>Set the TMTR bit to 1 when the value of TMTRF[1:0] in the RSCFDnCFDTMSTSp register is B'00.</p> <p>0: Transmission is not requested. 1: Transmission is requested.</p> |

When the RSCFDnCFDTMCp register meets any of the following conditions, set it to H'00.

- The RSCFDnCFDTMCp register corresponds to the transmit buffer number selected by the CFTML[3:0] bits in the RSCFDnCFDCFCCK register ($p = m \times 16 + \text{the value of CFTML}[3:0] \text{ bits}$).
- The RSCFDnCFDTMCp register corresponds to the transmit buffer allocated to the transmit queue by the TXQDC[3:0] bits in the RSCFDnCFDTXQCCm ($m = 0 \text{ or } 1$) register ($p = (m \times 16 + 15) \text{ to } (m \times 16 + 15 - \text{the value of TXQDC}[3:0] \text{ bits})$).
- The TMME bit in the RSCFDnCFDCmFDCFG register is 1 (transmit buffer merge mode) and the RSCFDnCFDTMCp register corresponds to the transmit buffer allocated to the payload storage area ($p = (m \times 16) + 1, (m \times 16) + 2, (m \times 16) + 4, (m \times 16) + 5$).

Bits in the RSCFDnCFDTMCp register are all cleared to 0 in channel reset mode. Modify the RSCFDnCFDTMCp register in channel communication mode or channel halt mode.

48.9.10.2 RSCFDnCFDTMSTSp — Transmit Buffer Status Register (p = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDTMSTSp register can be read/written in 8-bit units

Address: RSCFDnCFDTMSTSp: <RSCFDn_base> + H'02D0 + (H'01 × p)

Value after reset: H'00

| | | | | | | | | |
|----------------|---|---|---|------------|-----------|------------|------------|---|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | TMTAR M | TMTR M | TMTRF[1:0] | TMTST S | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|---|
| 7 to 5 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 4 | TMTARM | B'0 | R | Transmit Buffer Transmit Abort Request Status Flag The TMTARM flag is set to 1 when the TMTAR bit in the RSCFDnCFDTMCp register is set to 1. The TMTARM flag is set to 0 when the TMTAR bit in the RSCFDnCFDTMCp register is set to 0. 0: No transmit abort request is present. 1: A transmit abort request is present. |
| 3 | TMTRM | B'0 | R | Transmit Buffer Transmit Request Status Flag The TMTRM flag is set to 1 when the TMTR bit in the RSCFDnCFDTMCp register is set to 1. The TMTRM flag is set to 0 when the TMTR bit in the RSCFDnCFDTMCp register is set to 0. 0: No transmit request is present. 1: A transmit request is present. |
| 2, 1 | TMTRF[1:0] | B'00 | R/W | Transmit Buffer Transmit Result Status Flag This flag indicates the result of transmission from the transmit buffer. B'00: Transmission is in progress or no transmit request is present. B'01: Transmission from the transmit buffer was aborted. B'10: Transmission has been completed with the TMTAR bit in the RSCFDnCFDTMCp register set to 0 (transmit abort is not requested). B'11: Transmission has been completed with the TMTAR bit in the RSCFDnCFDTMCp register set to 1 (transmit abort is requested). Write B'00 to the TMTRF[1:0] flag in channel communication mode or channel halt mode. Do not write any value other than B'00 to this flag. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | TMTSTS | B'0 | R | <p>Transmit Buffer Transmit Status Flag</p> <p>This flag is set to 1 when transmission from the transmit buffer starts, and is cleared to 0 when transmission from the transmit buffer has been completed or terminated due to a bus error or arbitration lost.</p> <p>0: Transmission is not in progress. 1: Transmission is in progress.</p> |

The RSCFDnCFDTMSTSp register is cleared to all 0 in channel reset mode.

48.9.10.3 RSCFDnCFDTMIDp — Transmit Buffer ID Register (p = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDTMIDp register can be read/written in 32-bit units

RSCFDnCFDTMIDpL, RSCFDnCFDTMIDpH registers can be read/written in 16-bit units

RSCFDnCFDTMIDpLL, RSCFDnCFDTMIDpLH, RSCFDnCFDTMIDpHL, RSCFDnCFDTMIDpHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTMIDp: <RSCFDn_base> + H'4000 + (H'20 × p)

RSCFDnCFDTMIDpL: <RSCFDn_base> + H'4000 + (H'20 × p),

RSCFDnCFDTMIDpH: <RSCFDn_base> + H'4002 + (H'20 × p)

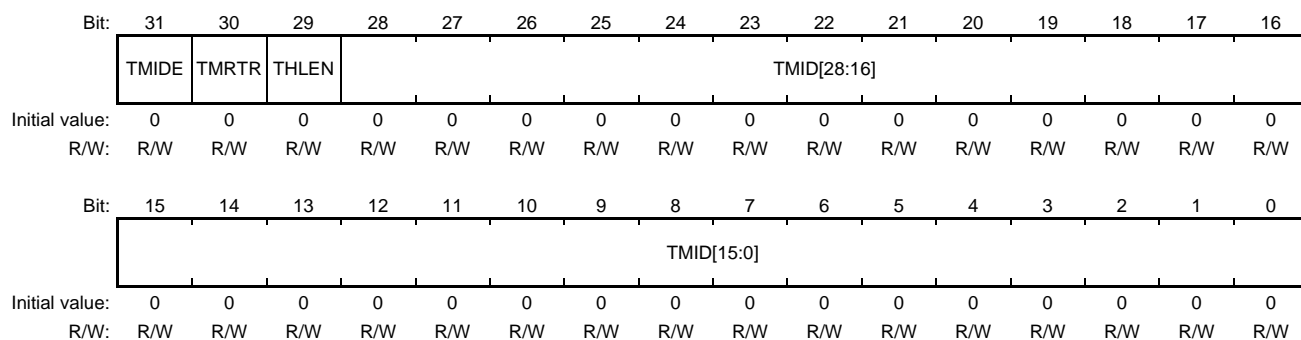
RSCFDnCFDTMIDpLL: <RSCFDn_base> + H'4000 + (H'20 × p),

RSCFDnCFDTMIDpLH: <RSCFDn_base> + H'4001 + (H'20 × p),

RSCFDnCFDTMIDpHL: <RSCFDn_base> + H'4002 + (H'20 × p),

RSCFDnCFDTMIDpHH: <RSCFDn_base> + H'4003 + (H'20 × p)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 31 | TMIDE | B'0 | R/W | Transmit Buffer IDE This bit is used to set the ID format of the message to be transmitted from the transmit buffer. 0: Standard ID 1: Extended ID |
| 30 | TMRTR | B'0 | R/W | Transmit Buffer RTR/RRS This bit is used to set the data format of the message to be transmitted from the transmit buffer. When the TMFDF bit in the RSCFDnCFDTMFDCTRp register is 1 (CAN FD frame), this bit must be cleared to 0. <ul style="list-style-type: none"> • When the transmit message is in the classical CAN frame: <ul style="list-style-type: none"> 0: Data frame 1: Remote frame • When the transmit message is in the CAN FD frame: <ul style="list-style-type: none"> Write 0 to this bit. |
| 29 | THLEN | B'0 | R/W | Transmit History Data Store Enable With this bit set to 1, the transmit history data of the message transmitted (label information, buffer number, buffer type, and timestamp) is stored in the transmit history buffer after transmission is completed. 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 28 to 0 | TMID[28:0] | H'0000_0000 | R/W | Transmit Buffer ID Data These bits are used to set the ID of the message to be transmitted from the transmit buffer. Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11. |

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write data to this register. If this register is allocated to the transmit queue, only write data to a transmit buffer p ($p = m \times 16 + 15$) for the corresponding channel.

48.9.10.4 RSCFDnCFDMPTRp — Transmit Buffer Pointer Register (p= 0 to 31)

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

Access: RSCFDnCFDMPTRp register can be read/written in 32-bit units

RSCFDnCFDMPTRpL, RSCFDnCFDMPTRpH registers can be read/written in 16-bit units

RSCFDnCFDMPTRpLL, RSCFDnCFDMPTRpLH, RSCFDnCFDMPTRpHL, RSCFDnCFDMPTRpHH registers can be read/written in 8-bit units

Address: RSCFDnCFDMPTRp: <RSCFDn_base> + H'4004 + (H'20 × p)

RSCFDnCFDMPTRpL: <RSCFDn_base> + H'4004 + (H'20 × p),

RSCFDnCFDMPTRpH: <RSCFDn_base> + H'4006 + (H'20 × p)

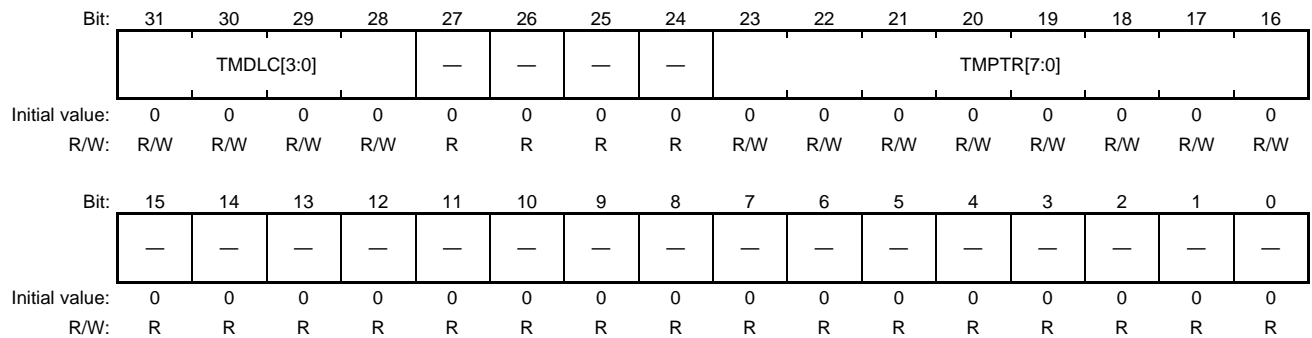
RSCFDnCFDMPTRpLL: <RSCFDn_base> + H'4004 + (H'20 × p),

RSCFDnCFDMPTRpLH: <RSCFDn_base> + H'4005 + (H'20 × p),

RSCFDnCFDMPTRpHL: <RSCFDn_base> + H'4006 + (H'20 × p),

RSCFDnCFDMPTRpHH: <RSCFDn_base> + H'4007 + (H'20 × p)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------|---------------------|---------------|-----|--|--|---------------------|--------------|---------|--------------|--|---------|-------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|--|---------|--------------|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|---------|--|---------------|
| 31 to 28 | TMDLC[3:0] | H'0 | R/W | <p>Transmit Buffer DLC Data</p> <p>These bits are used to set the data length of the message to be transmitted from the transmit buffer when the TMRTR bit in the RSCFDnCFDTMIDp register is set to 0 (data frame). When the TMFDF bit in the RSCFDnCFDTMFDCTRp register is 0 (classical CAN frame) and the value equal to or larger than B'1001 is set to the TMDLC[3:0] bits, 8-byte data is transmitted. When the TMFDF bit is 1 (CAN FD frame), the range of values to be set differs depending on the TMME bit in the RSCFDnCFDCmFDCFG register.</p> <ul style="list-style-type: none"> When the TMME bit is 0 (the transmit buffer merge mode is disabled): All values from B'0000 to B'1111 can be set. If B'1100 or larger value is set, the payload which exceeds 20 bytes will be padded by CCH. When the TMME bit is 1 (the transmit buffer merge mode is enabled): When the number of the corresponding transmit buffer is $p = (m \times 16) + 0$ or $(m \times 16) + 3$, all values from B'0000 to B'1111 can be set. For other cases, set values up to B'1011 (20 data bytes). <p>When the TMRTR bit is set to 1 (remote frame), set the data length of messages to be requested.</p> <table border="1"> <thead> <tr> <th></th> <th>Classical CAN frame</th> <th>CAN FD frame</th> </tr> </thead> <tbody> <tr><td>B'0000:</td><td>0 data bytes</td><td></td></tr> <tr><td>B'0001:</td><td>1 data byte</td><td></td></tr> <tr><td>B'0010:</td><td>2 data bytes</td><td></td></tr> <tr><td>B'0011:</td><td>3 data bytes</td><td></td></tr> <tr><td>B'0100:</td><td>4 data bytes</td><td></td></tr> <tr><td>B'0101:</td><td>5 data bytes</td><td></td></tr> <tr><td>B'0110:</td><td>6 data bytes</td><td></td></tr> <tr><td>B'0111:</td><td>7 data bytes</td><td></td></tr> <tr><td>B'1000:</td><td>8 data bytes</td><td></td></tr> <tr><td>B'1001:</td><td>8 data bytes</td><td>12 data bytes</td></tr> <tr><td>B'1010:</td><td></td><td>16 data bytes</td></tr> <tr><td>B'1011:</td><td></td><td>20 data bytes</td></tr> <tr><td>B'1100:</td><td></td><td>24 data bytes</td></tr> <tr><td>B'1101:</td><td></td><td>32 data bytes</td></tr> <tr><td>B'1110:</td><td></td><td>48 data bytes</td></tr> <tr><td>B'1111:</td><td></td><td>64 data bytes</td></tr> </tbody> </table> | | Classical CAN frame | CAN FD frame | B'0000: | 0 data bytes | | B'0001: | 1 data byte | | B'0010: | 2 data bytes | | B'0011: | 3 data bytes | | B'0100: | 4 data bytes | | B'0101: | 5 data bytes | | B'0110: | 6 data bytes | | B'0111: | 7 data bytes | | B'1000: | 8 data bytes | | B'1001: | 8 data bytes | 12 data bytes | B'1010: | | 16 data bytes | B'1011: | | 20 data bytes | B'1100: | | 24 data bytes | B'1101: | | 32 data bytes | B'1110: | | 48 data bytes | B'1111: | | 64 data bytes |
| | Classical CAN frame | CAN FD frame | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0000: | 0 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0001: | 1 data byte | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0010: | 2 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0011: | 3 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0100: | 4 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0101: | 5 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0110: | 6 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'0111: | 7 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1000: | 8 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1001: | 8 data bytes | 12 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1010: | | 16 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1011: | | 20 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1100: | | 24 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1101: | | 32 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1110: | | 48 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| B'1111: | | 64 data bytes | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 27 to 24 | Reserved | All 0 | R | <p>When read, the value after reset is returned.</p> <p>When writing to these bits, write the value after reset.</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 23 to 16 | TMPTR[7:0] | H'00 | R/W | <p>Transmit Buffer Label Data</p> <p>Set the label information to be stored in the transmit history buffer. When message transmission has been completed, the TMPTR[7:0] value is stored in the transmit history buffer.</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 15 to 0 | Reserved | All 0 | R | <p>When read, the value after reset is returned.</p> <p>When writing to these bits, write the value after reset.</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p ($p = m \times 16 + 15$) for the corresponding channel.

48.9.10.5 RSCFDnCFDTMFDCTR_p — Transmit Buffer CAN FD Configuration Register (p = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDTMFDCTR_p register can be read/written in 32-bit units
 RSCFDnCFDTMFDCTR_{pL}, RSCFDnCFDTMFDCTR_{pH} registers can be read/written in 16-bit units
 RSCFDnCFDTMFDCTR_{pLL}, RSCFDnCFDTMFDCTR_{pLH}, RSCFDnCFDTMFDCTR_{pHL},
 RSCFDnCFDTMFDCTR_{pHH} registers can be read/written in 8-bit units

Address: RSCFDnCFDTMFDCTR_p: <RSCFDn_base> + H'4008 + (H'20 × p)
 RSCFDnCFDTMFDCTR_{pL}: <RSCFDn_base> + H'4008 + (H'20 × p),
 RSCFDnCFDTMFDCTR_{pH}: <RSCFDn_base> + H'400A + (H'20 × p)
 RSCFDnCFDTMFDCTR_{pLL}: <RSCFDn_base> + H'4008 + (H'20 × p),
 RSCFDnCFDTMFDCTR_{pLH}: <RSCFDn_base> + H'4009 + (H'20 × p),
 RSCFDnCFDTMFDCTR_{pHL}: <RSCFDn_base> + H'400A + (H'20 × p),
 RSCFDnCFDTMFDCTR_{pHH}: <RSCFDn_base> + H'400B + (H'20 × p)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | TMFDF | TMBRS | TMESI |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 3 | Reserved | All 0 | R | These bits are read as the value after reset. The write value should be the value after reset. |
| 2 | TMFDF | B'0 | R/W | FD This bit is used to set the FD format of the message to be transmitted from the transmit buffer. 0: Classical CAN frame 1: CAN FD frame |
| 1 | TMBRS | B'0 | R/W | BRS When the TMFDF bit is 1 and this bit is set to 1, the data area for transmit messages is transmitted with data bit rate. Write 0 to this bit if the TMFDF bit is 0. 0: The bit rate in the data area is not changed. 1: The bit rate in the data area is changed. |
| 0 | TMESI | B'0 | R/W | ESI When the TMFDF bit is 1, this bit is used to set the ESI bit value of the message transmitted from the transmit buffer. The set value is transmitted when the ESIC bit in the RSCFDnCFDCmFDCFG register is 1 and the channel is in error-active state. When the channel is in error-passive state, the ESI bit value indicating the error-passive node is transmitted regardless of this bit. Write 0 to this bit if the TMFDF bit is 0. 0: Error-active node 1: Error-passive node |

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p ($p = m \times 16 + 15$) for the corresponding channel.

48.9.10.6 RSCFDnCFDTMDFb_p — Transmit Buffer Data Field b Register (b = 0 to 4, p = 0 to 31)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDTMDFb_p register can be read/written in 32-bit units

RSCFDnCFDTMDFb_pL, RSCFDnCFDTMDFb_pH registers can be read/written in 16-bit units

RSCFDnCFDTMDFb_pLL, RSCFDnCFDTMDFb_pLH, RSCFDnCFDTMDFb_pHL, RSCFDnCFDTMDFb_pHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTMDFb_p: <RSCFDn_base> + H'400C + (H'04 × b) + (H'20 × p)

RSCFDnCFDTMDFb_pL: <RSCFDn_base> + H'400C + (H'04 × b) + (H'20 × p),

RSCFDnCFDTMDFb_pH: <RSCFDn_base> + H'400E + (H'04 × b) + (H'20 × p)

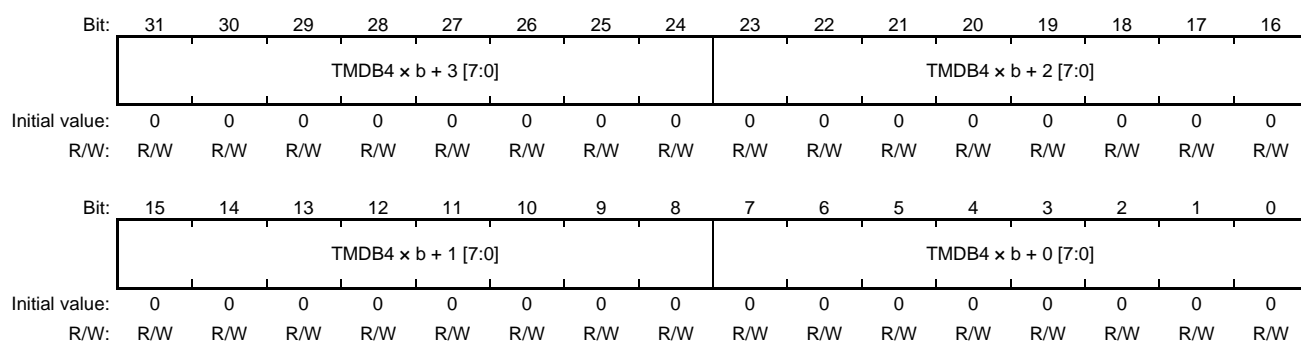
RSCFDnCFDTMDFb_pLL: <RSCFDn_base> + H'400C + (H'04 × b) + (H'20 × p),

RSCFDnCFDTMDFb_pLH: <RSCFDn_base> + H'400D + (H'04 × b) + (H'20 × p),

RSCFDnCFDTMDFb_pHL: <RSCFDn_base> + H'400E + (H'04 × b) + (H'20 × p),

RSCFDnCFDTMDFb_pHH: <RSCFDn_base> + H'400F + (H'04 × b) + (H'20 × p)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|-------------------------------------|
| 31 to 24 | TMDB4 × b + 3 [7:0] | H'00 | R/W | Transmit Buffer Data Byte 4 × b + 3 |
| | | | | Transmit Buffer Data Byte 4 × b + 2 |
| 23 to 16 | TMDB4 × b + 2 [7:0] | H'00 | R/W | Transmit Buffer Data Byte 4 × b + 1 |
| | | | | Transmit Buffer Data Byte 4 × b + 0 |
| 15 to 8 | TMDB4 × b + 1 [7:0] | H'00 | R/W | Set the transmit buffer data. |
| 7 to 0 | TMDB4 × b + 0 [7:0] | H'00 | R/W | |

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

48.9.10.7 RSCFDnCFDTMIECy — Transmit Buffer Interrupt Enable Configuration Register (y = 0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDTMIECy register can be read/written in 32-bit units

RSCFDnCFDTMIECyL, RSCFDnCFDTMIECyH registers can be read/written in 16-bit units

RSCFDnCFDTMIECyLL, RSCFDnCFDTMIECyLH, RSCFDnCFDTMIECyHL, RSCFDnCFDTMIECyHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTMIECy: <RSCFDn_base> + H'0390 + (H'04 × y)

RSCFDnCFDTMIECyL: <RSCFDn_base> + H'0390 + (H'04 × y),

RSCFDnCFDTMIECyH: <RSCFDn_base> + H'0392 + (H'04 × y)

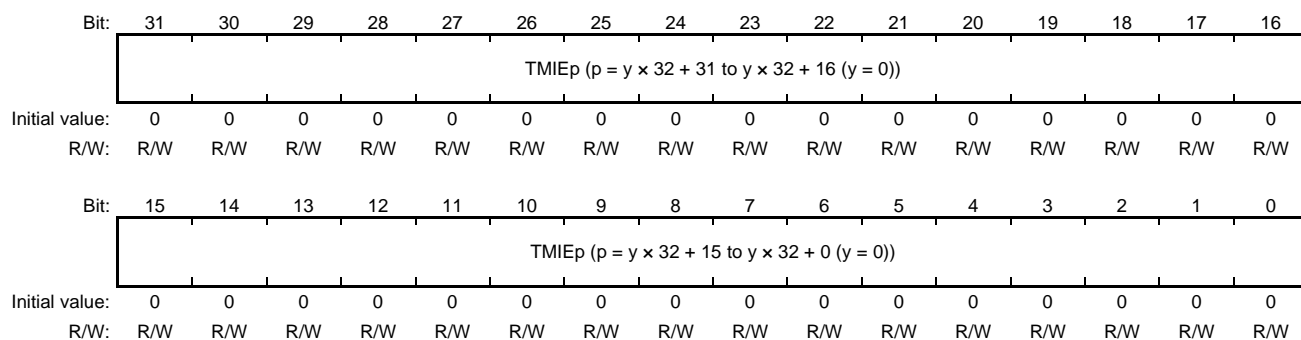
RSCFDnCFDTMIECyLL: <RSCFDn_base> + H'0390 + (H'04 × y),

RSCFDnCFDTMIECyLH: <RSCFDn_base> + H'0391 + (H'04 × y),

RSCFDnCFDTMIECyHL: <RSCFDn_base> + H'0392 + (H'04 × y),

RSCFDnCFDTMIECyHH: <RSCFDn_base> + H'0393 + (H'04 × y)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | TMIEp | H'0000_0000 | R/W | Transmit Buffer Interrupt Enable p (p = y × 32 + 31 to y × 32 + 0) 0: Transmit buffer interrupt is disabled. 1: Transmit buffer interrupt is enabled. |

TMIEp Bits (p = 0 to 31)

When any of these bits is set to 1 and the corresponding transmission has been completed, a transmit buffer interrupt request is generated.

Modify these bits when the TMTRM flag in the corresponding RSCFDnCFDTMSTSp register is 0 (no transmit request is present).

Write 0 to bits corresponding to transmit buffers linked to transmit/receive FIFO buffers or transmit buffers allocated to the transmit queue. When the TMME bit in the RSCFDnCFDCmFDCFG register is 1 (the transmit buffer merge mode is enabled), write 0 to bits corresponding to transmit buffers allocated to the payload storage area.

Table 48.23 shows the bit assignment.

Table 48.23 TMIEp Bit Assignment

| Bit | Channel | Transmit Buffer Number |
|------------|----------------|-------------------------------|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| · | · | · |
| · | · | · |
| 15 | 0 | 15 |
| 16 | 1 | 0 |
| · | · | · |
| · | · | · |
| 30 | 1 | 14 |
| 31 | 1 | 15 |

48.9.11 Details of Transmit Buffer Status Related Registers

48.9.11.1 RSCFDnCFDTMTRSTSy — Transmit Buffer Transmit Request Status Register (y = 0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDTMTRSTSy register can be read only in 32-bit units

RSCFDnCFDTMTRSTSyL, RSCFDnCFDTMTRSTSyH registers can be read only in 16-bit units

RSCFDnCFDTMTRSTSyLL, RSCFDnCFDTMTRSTSyLH, RSCFDnCFDTMTRSTSyHL,

RSCFDnCFDTMTRSTSyHH registers can be read only in 8-bit units

Address: RSCFDnCFDTMTRSTSy: <RSCFDn_base> + H'0350 + (H'04 × y)

RSCFDnCFDTMTRSTSyL: <RSCFDn_base> + H'0350 + (H'04 × y),

RSCFDnCFDTMTRSTSyH: <RSCFDn_base> + H'0352 + (H'04 × y)

RSCFDnCFDTMTRSTSyLL: <RSCFDn_base> + H'0350 + (H'04 × y),

RSCFDnCFDTMTRSTSyLH: <RSCFDn_base> + H'0351 + (H'04 × y),

RSCFDnCFDTMTRSTSyHL: <RSCFDn_base> + H'0352 + (H'04 × y),

RSCFDnCFDTMTRSTSyHH: <RSCFDn_base> + H'0353 + (H'04 × y)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TMTRSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0)) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TMTRSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0)) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | TMTRSTSp | H'0000_0000 | R | Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 31 to y × 32 + 0) 0: No transmit request is present. 1: A transmit request is present. |

TMTRSTSp Flags (p = 0 to 31)

These flags indicate the status of the TMTR bit in the RSCFDnCFDTMCP register.

When the TMTR bit is set to 1 (transmission is requested), the corresponding TMTRSTSp flag is set to 1.

The corresponding TMTRSTSp flag is cleared to 0 when the TMTR bit is set to 0 (transmission is not requested) or in channel reset mode.

Table 48.24 shows the bit assignment.

Table 48.24 TMTRSTSp Bit Assignment

| Bit | Channel | Transmit Buffer Number |
|-----|---------|------------------------|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| · | · | · |
| · | · | · |
| 15 | 0 | 15 |
| 16 | 1 | 0 |
| · | · | · |
| · | · | · |
| 30 | 1 | 14 |
| 31 | 1 | 15 |

48.9.11.2 RSCFDnCFDnTMTARSTSy — Transmit Buffer Transmit Abort Request Status Register (y = 0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDnTMTARSTSy register can be read only in 32-bit units

RSCFDnCFDnTMTARSTSyL, RSCFDnCFDnTMTARSTSyH registers can be read only in 16-bit units

RSCFDnCFDnTMTARSTSyLL, RSCFDnCFDnTMTARSTSyLH, RSCFDnCFDnTMTARSTSyHL,

RSCFDnCFDnTMTARSTSyHH registers can be read only in 8-bit units

Address: RSCFDnCFDnTMTARSTSy: <RSCFDn_base> + H'0360 + (H'04 × y)

RSCFDnCFDnTMTARSTSyL: <RSCFDn_base> + H'0360 + (H'04 × y),

RSCFDnCFDnTMTARSTSyH: <RSCFDn_base> + H'0362 + (H'04 × y)

RSCFDnCFDnTMTARSTSyLL: <RSCFDn_base> + H'0360 + (H'04 × y),

RSCFDnCFDnTMTARSTSyLH: <RSCFDn_base> + H'0361 + (H'04 × y),

RSCFDnCFDnTMTARSTSyHL: <RSCFDn_base> + H'0362 + (H'04 × y),

RSCFDnCFDnTMTARSTSyHH: <RSCFDn_base> + H'0363 + (H'04 × y)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|--|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TMTARSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0)) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TMTARSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0)) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | TMTARSTSp | H'0000_0000 | R | Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 31 to y × 32 + 0) 0: No transmit abort request is present. 1: A transmit abort request is present. |

TMTARSTSp Flags (p = 0 to 31)

These flags indicate the status of the TMTAR bit in the RSCFDnCFDnTMCp register.

When the TMTAR bit is set to 1 (transmit abort is requested), the corresponding TMTARSTSp flag is set to 1.

The corresponding TMTARSTSp flag is cleared to 0 when the TMTAR bit is set to 0 (transmit abort is not requested) or in channel reset mode.

Table 48.25 shows the bit assignment.

Table 48.25 TMTARSTSp Bit Assignment

| Bit | Channel | Transmit Buffer Number |
|------------|----------------|-------------------------------|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| · | · | · |
| · | · | · |
| 15 | 0 | 15 |
| 16 | 1 | 0 |
| · | · | · |
| · | · | · |
| 30 | 1 | 14 |
| 31 | 1 | 15 |

48.9.11.3 RSCFDnCFDTMTCSTSy — Transmit Buffer Transmit Complete Status Register (y = 0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDTMTCSTSy register can be read only in 32-bit units

RSCFDnCFDTMTCSTSyL, RSCFDnCFDTMTCSTSyH registers can be read only in 16-bit units

RSCFDnCFDTMTCSTSyLL, RSCFDnCFDTMTCSTSyLH, RSCFDnCFDTMTCSTSyHL,

RSCFDnCFDTMTCSTSyHH registers can be read only in 8-bit units

Address: RSCFDnCFDTMTCSTSy: <RSCFDn_base> + H'0370 + (H'04 × y)

RSCFDnCFDTMTCSTSyL: <RSCFDn_base> + H'0370 + (H'04 × y),

RSCFDnCFDTMTCSTSyH: <RSCFDn_base> + H'0372 + (H'04 × y)

RSCFDnCFDTMTCSTSyLL: <RSCFDn_base> + H'0370 + (H'04 × y),

RSCFDnCFDTMTCSTSyLH: <RSCFDn_base> + H'0371 + (H'04 × y),

RSCFDnCFDTMTCSTSyHL: <RSCFDn_base> + H'0372 + (H'04 × y),

RSCFDnCFDTMTCSTSyHH: <RSCFDn_base> + H'0373 + (H'04 × y)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|--|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TMTTCSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0)) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TMTTCSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0)) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | TMTTCSTSp | H'0000_0000 | R | Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 31 to y × 32 + 0) 0: Transmission has not been completed. 1: Transmission has been completed. |

TMTTCSTSp Flags (p = 0 to 31)

When the TMTRF[1:0] flag in the RSCFDnCFDTMSTSp register is set to B'10 (transmission has been completed (without transmit abort request)) or B'11 (transmission has been completed (with transmit abort request)), the corresponding TMTTCSTSp flag is set to 1.

A TMTTCSTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to B'00 or in channel reset mode.

Table 48.26 shows the bit assignment.

Table 48.26 TMTcSTSp Bit Assignment

| Bit | Channel | Transmit Buffer Number |
|-----|---------|------------------------|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| · | · | · |
| · | · | · |
| 15 | 0 | 15 |
| 16 | 1 | 0 |
| · | · | · |
| · | · | · |
| 30 | 1 | 14 |
| 31 | 1 | 15 |

48.9.11.4 RSCFDnCFDnTMTASTSy — Transmit Buffer Transmit Abort Status Register (y = 0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDnTMTASTSy register can be read only in 32-bit units
 RSCFDnCFDnTMTASTSyL, RSCFDnCFDnTMTASTSyH registers can be read only in 16-bit units
 RSCFDnCFDnTMTASTSyLL, RSCFDnCFDnTMTASTSyLH, RSCFDnCFDnTMTASTSyHL,
 RSCFDnCFDnTMTASTSyHH registers can be read only in 8-bit units

Address: RSCFDnCFDnTMTASTSy: <RSCFDn_base> + H'0380 + (H'04 × y)
 RSCFDnCFDnTMTASTSyL: <RSCFDn_base> + H'0380 + (H'04 × y),
 RSCFDnCFDnTMTASTSyH: <RSCFDn_base> + H'0382 + (H'04 × y)
 RSCFDnCFDnTMTASTSyLL: <RSCFDn_base> + H'0380 + (H'04 × y),
 RSCFDnCFDnTMTASTSyLH: <RSCFDn_base> + H'0381 + (H'04 × y),
 RSCFDnCFDnTMTASTSyHL: <RSCFDn_base> + H'0382 + (H'04 × y),
 RSCFDnCFDnTMTASTSyHH: <RSCFDn_base> + H'0383 + (H'04 × y)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TMTASTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0)) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TMTASTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0)) | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | TMTASTSp | H'0000_0000 | R | Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 31 to y × 32 + 0) 0: Transmission is not aborted. 1: Transmission is aborted. |

TMTASTSp Flags (p = 0 to 31)

When the TMTRF[1:0] flag in the RSCFDnCFDnTMTASTSp register is set to B'01 (transmit abort has been completed), the corresponding TMTASTSp flag is set to 1.

A TMTASTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to B'00 or in channel reset mode.

Table 48.27 shows the bit assignment.

Table 48.27 TMTASTSp Bit Assignment

| Bit | Channel | Transmit Buffer Number |
|------------|----------------|-------------------------------|
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| · | · | · |
| · | · | · |
| 15 | 0 | 15 |
| 16 | 1 | 0 |
| · | · | · |
| · | · | · |
| 30 | 1 | 14 |
| 31 | 1 | 15 |

48.9.12 Details of Transmit Queue Related Registers

48.9.12.1 RSCFDnCFDTXQCCm — Transmit Queue Configuration and Control Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDTXQCCm register can be read/written in 32-bit units

RSCFDnCFDTXQCCmL, RSCFDnCFDTXQCCmH registers can be read/written in 16-bit units

RSCFDnCFDTXQCCmLL, RSCFDnCFDTXQCCmLH, RSCFDnCFDTXQCCmHL, RSCFDnCFDTXQCCmHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTXQCCm: <RSCFDn_base> + H'03A0 + (H'04 × m)

RSCFDnCFDTXQCCmL: <RSCFDn_base> + H'03A0 + (H'04 × m),

RSCFDnCFDTXQCCmH: <RSCFDn_base> + H'03A2 + (H'04 × m)

RSCFDnCFDTXQCCmLL: <RSCFDn_base> + H'03A0 + (H'04 × m),

RSCFDnCFDTXQCCmLH: <RSCFDn_base> + H'03A1 + (H'04 × m),

RSCFDnCFDTXQCCmHL: <RSCFDn_base> + H'03A2 + (H'04 × m),

RSCFDnCFDTXQCCmHH: <RSCFDn_base> + H'03A3 + (H'04 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|-------|-------|------------|-----|-----|-----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | TXQIM | TXQIE | TXQDC[3:0] | | | — | — | — | — | — | — | — | — | TXQE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 14 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 13 | TXQIM | B'0 | R/W | Transmit Queue Interrupt Source Select This bit is used to select a transmit queue interrupt source. Modify this bit in channel reset mode. 0: When the transmit queue becomes empty upon completion of message transmission, a transmit queue interrupt request is generated. 1: A transmit queue interrupt request is generated each time a message has been transmitted. |
| 12 | TXQIE | B'0 | R/W | Transmit Queue Interrupt Enable When the TXQIE bit is set to 1 and the source selected by the TXQIM bit occurs, an interrupt request is generated. Set the TXQE bit to 0 before modifying the TXQIE bit. 0: Transmit queue interrupt is disabled. 1: Transmit queue interrupt is enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 11 to 8 | TXQDC[3:0] | B'0000 | R/W | <p>Transmit Queue Depth Configuration</p> <p>Setting these bits to g (g = 2 to 15) makes the (g + 1)-buffer transmit queue available.</p> <p>Setting these bits to 0 disables the transmit queue.</p> <p>Setting these bits to 1 is prohibited.</p> <p>For transmit buffer merge mode, set g = 2 to 9.</p> <p>These bits are used to specify the number of transmit buffers to be allocated to the transmit queues. Transmit buffers are allocated to transmit queues in descending order of buffer number, that is, from (m × 16 + 15) to (m × 16 + 0). For examples of how buffer allocation is done, see Figure 48.9.</p> <p>When the TMME bit in the RSCFDnCFDCmFDCFG register is 1 (transmit buffer merge mode), transmit buffers from (m × 16 + 5) to (m × 16 + 0) are merged and cannot be allocated to the transmit queues. Do not specify 10 to 15 to the TXQDC[3:0] bits. Modify these bits only in channel reset mode.</p> |
| 7 to 1 | Reserved | All 0 | R | <p>When read, the value after reset is returned.</p> <p>When writing to these bits, write the value after reset.</p> |
| 0 | TXQE | B'0 | R/W | <p>Transmit Queue Enable</p> <p>Setting this bit to 1 makes the transmit queue available. Modify this bit in channel communication mode or channel halt mode. This bit is cleared to 0 in channel reset mode.</p> <p>Before setting the TXQE bit to 1, set the TXQDC[3:0] bits to B'0010 or more.</p> <p>0: The transmit queue is not used. 1: The transmit queue is used.</p> |

48.9.12.2 RSCFDnCFDnTXQSTSm — Transmit Queue Status Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDnTXQSTSm register can be read/written in 32-bit units

RSCFDnCFDnTXQSTSmL, RSCFDnCFDnTXQSTSmH registers can be read/written in 16-bit units

RSCFDnCFDnTXQSTSmLL, RSCFDnCFDnTXQSTSmLH, RSCFDnCFDnTXQSTSmHL, RSCFDnCFDnTXQSTSmHH registers can be read/written in 8-bit units

Address: RSCFDnCFDnTXQSTSm: <RSCFDn_base> + H'03C0 + (H'04 × m)

RSCFDnCFDnTXQSTSmL: <RSCFDn_base> + H'03C0 + (H'04 × m),

RSCFDnCFDnTXQSTSmH: <RSCFDn_base> + H'03C2 + (H'04 × m)

RSCFDnCFDnTXQSTSmLL: <RSCFDn_base> + H'03C0 + (H'04 × m),

RSCFDnCFDnTXQSTSmLH: <RSCFDn_base> + H'03C1 + (H'04 × m),

RSCFDnCFDnTXQSTSmHL: <RSCFDn_base> + H'03C2 + (H'04 × m),

RSCFDnCFDnTXQSTSmHH: <RSCFDn_base> + H'03C3 + (H'04 × m)

Value after reset: H'0000_0001

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | TXQIF | TXQFL L | TXQEM P |
| Initial value: | 0 | 0 | 0 | — | — | — | — | — | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W* | R | R |

Note: * The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 13 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 12 to 8 | Reserved | Undefined | R | When read, an undefined value is returned. When writing to these bits, write the value after reset. |
| 7 to 3 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 2 | TXQIF | B'0 | R/W | Transmit Queue Interrupt Request Flag The TXQIF flag is set to 1 when the event specified by the TXQIM bit in the RSCFDnCFDnTXQCCm register has occurred. The TXQIF flag is cleared to 0 in channel reset mode or by writing 0 to this flag. This flag is not cleared to 0 by setting the TXQE bit in the RSCFDnCFDnTXQCCm register to 0 (the transmit queue is not used). 0: No transmit queue interrupt request is present. 1: A transmit queue interrupt request is present. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | TXQFLL | B'0 | R | <p>Transmit Queue Full Status Flag</p> <p>The TXQFLL flag is set to 1 when the number of messages set for the transmit queue matches the transmit queue depth set by the TXQDC[3:0] bits in the RSCFDnCFDnTXQCCm register.</p> <p>This flag is cleared to 0 in any of the following cases.</p> <ul style="list-style-type: none"> • The number of messages set for the transmit queue is smaller than the transmit queue depth set by the TXQDC[3:0] bits. • In channel reset mode <p>0: The transmit queue is not full. 1: The transmit queue is full.</p> |
| 0 | TXQEMP | B'1 | R | <p>Transmit Queue Empty Status Flag</p> <p>The TXQEMP flag is cleared to 0 when even a single message is set for the transmit queue.</p> <p>This flag is set to 1 in any of the following cases.</p> <ul style="list-style-type: none"> • The TXQE bit is set to 0 (the transmit queue is not used). • The transmit queue becomes empty. • In channel reset mode <p>0: The transmit queue contains messages. 1: The transmit queue contains no message (transmit queue empty).</p> |

48.9.12.3 RSCFDnCFDnTXQPCTRM — Transmit Queue Pointer Control Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDnTXQPCTRM register can only be written in 32-bit units
 RSCFDnCFDnTXQPCTRM_L, RSCFDnCFDnTXQPCTRM_H registers can only be written in 16-bit units
 RSCFDnCFDnTXQPCTRM_{LL}, RSCFDnCFDnTXQPCTRM_{LH}, RSCFDnCFDnTXQPCTRM_{HL},
 RSCFDnCFDnTXQPCTRM_{HH} registers can only be written in 8-bit units

Address: RSCFDnCFDnTXQPCTRM: <RSCFDn_base> + H'03E0 + (H'04 × m)
 RSCFDnCFDnTXQPCTRM_L: <RSCFDn_base> + H'03E0 + (H'04 × m),
 RSCFDnCFDnTXQPCTRM_H: <RSCFDn_base> + H'03E2 + (H'04 × m)
 RSCFDnCFDnTXQPCTRM_{LL}: <RSCFDn_base> + H'03E0 + (H'04 × m),
 RSCFDnCFDnTXQPCTRM_{LH}: <RSCFDn_base> + H'03E1 + (H'04 × m),
 RSCFDnCFDnTXQPCTRM_{HL}: <RSCFDn_base> + H'03E2 + (H'04 × m),
 RSCFDnCFDnTXQPCTRM_{HH}: <RSCFDn_base> + H'03RZ/G2E + (H'04 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------------|----|----|----|----|----|----|----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | TXQPC[7:0] | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | W | W | W | W | W | W | W | W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 8 | Reserved | All 0 | R | The write value should be the value after reset. |
| 7 to 0 | TXQPC[7:0] | H'00 | W | <p>Transmit Queue Pointer Control</p> <p>Writing H'FF to the TXQPC[7:0] bits moves the write pointer to the next transmit queue buffer and generates a transmit request of the message. Write transmit messages to the RSCFDnCFDnTXQPCp, RSCFDnCFDnTXQPCRp, RSCFDnCFDnTXQPCFp, and RSCFDnCFDnTXQPCFb_p registers (p = 15, 31) before writing H'FF to the TXQPC[7:0] bits.</p> <p>When writing H'FF to these bits, make sure that the TXQE bit in the RSCFDnCFDnTXQCCm register is set to 1 (the transmit queue is used) and the TXQFLL flag in the RSCFDnCFDnTXQSTSm register is 0 (the transmit queue is not full).</p> |

48.9.13 Details of Transmit History Related Registers

48.9.13.1 RSCFDnCFDTHLCCm — Transmit History Configuration and Control Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDTHLCCm register can be read/written in 32-bit units

RSCFDnCFDTHLCCmL, RSCFDnCFDTHLCCmH registers can be read/written in 16-bit units

RSCFDnCFDTHLCCmLL, RSCFDnCFDTHLCCmLH, RSCFDnCFDTHLCCmHL, RSCFDnCFDTHLCCmHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTHLCCm: <RSCFDn_base> + H'0400 + (H'04 × m)

RSCFDnCFDTHLCCmL: <RSCFDn_base> + H'0400 + (H'04 × m),

RSCFDnCFDTHLCCmH: <RSCFDn_base> + H'0402 + (H'04 × m)

RSCFDnCFDTHLCCmLL: <RSCFDn_base> + H'0400 + (H'04 × m),

RSCFDnCFDTHLCCmLH: <RSCFDn_base> + H'0401 + (H'04 × m),

RSCFDnCFDTHLCCmHL: <RSCFDn_base> + H'0402 + (H'04 × m),

RSCFDnCFDTHLCCmHH: <RSCFDn_base> + H'0403 + (H'04 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|------------|-------|-------|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | THLDT E | THLIM | THLIE | — | — | — | — | — | — | — | THLE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 11 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 10 | THLDTE | B'0 | R/W | Transmit History Target Buffer Select When this bit is set to 0, the transmit history data of messages transmitted from transmit/receive FIFO buffers and the transmit queue is stored in the transmit history buffer. When this bit is set to 1, the transmit history data of messages transmitted from transmit buffers, transmit/receive FIFO buffers, and the transmit queue is stored in the transmit history buffer. Modify this bit only in channel reset mode. 0: Entry from transmit/receive FIFO buffers and transmit queue 1: Entry from transmit buffers, transmit/receive FIFO buffers, and transmit queue |
| 9 | THLIM | B'0 | R/W | Transmit History Interrupt Source Select This bit is used to select a transmit history interrupt source. Modify this bit only in channel reset mode. 0: When 12 sets of data have been stored in the transmit history buffer 1: When a single set of transmit history data has been stored |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 8 | THLIE | B'0 | R/W | <p>Transmit History Interrupt Enable</p> <p>When the THLIE bit is set to 1 and the source selected by the THLIM bit has occurred, a transmit history interrupt request is generated. Modify the THLIE bit only when the THLE bit set to 0.</p> <p>0: Transmit history interrupt is disabled. 1: Transmit history interrupt is enabled.</p> |
| 7 to 1 | Reserved | All 0 | R | <p>When read, the value after reset is returned.</p> <p>When writing to these bits, write the value after reset.</p> |
| 0 | THLE | B'0 | R/W | <p>Transmit History Buffer Enable</p> <p>Setting this bit to 1 makes the transmit history buffer available. When data transmission from the buffer selected by the THLDTE bit has been completed, the transmit history data of transmit messages is stored in the transmit history buffer.</p> <p>Modify this bit in channel communication mode or channel halt mode.</p> <p>0: Transmit history buffer is not used. 1: Transmit history buffer is used.</p> |

48.9.13.2 RSCFDnCFDTHLSTSm — Transmit History Status Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDTHLSTSm register can be read/written in 32-bit units

RSCFDnCFDTHLSTSmL, RSCFDnCFDTHLSTSmH registers can be read/written in 16-bit units

RSCFDnCFDTHLSTSmLL, RSCFDnCFDTHLSTSmLH, RSCFDnCFDTHLSTSmHL, RSCFDnCFDTHLSTSmHH registers can be read/written in 8-bit units

Address: RSCFDnCFDTHLSTSm: <RSCFDn_base> + H'0420 + (H'04 × m)

RSCFDnCFDTHLSTSmL: <RSCFDn_base> + H'0420 + (H'04 × m),

RSCFDnCFDTHLSTSmH: <RSCFDn_base> + H'0422 + (H'04 × m)

RSCFDnCFDTHLSTSmLL: <RSCFDn_base> + H'0420 + (H'04 × m),

RSCFDnCFDTHLSTSmLH: <RSCFDn_base> + H'0421 + (H'04 × m),

RSCFDnCFDTHLSTSmHL: <RSCFDn_base> + H'0422 + (H'04 × m),

RSCFDnCFDTHLSTSmHH: <RSCFDn_base> + H'0423 + (H'04 × m)

Value after reset: H'0000_0001

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|------------|----|----|----|----|----|----|----|----|-------|------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | THLMC[4:0] | | | | — | — | — | — | — | THLIF | THLEL T | THLFL L | THLEM P |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W* | R/W* | R | R |

Note: * The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 13 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 12 to 8 | THLMC[4:0] | B'0_0000 | R | Transmit History Buffer Unread Data Counter These bits indicate the number of unread data sets stored in the transmit history buffer. |
| 7 to 4 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 3 | THLIF | B'0 | R/W | Transmit History Interrupt Request Flag The THLIF flag is set to 1 when the interrupt source specified with the THLIM bit in the RSCFDnCFDTHLCCm register occurs. This flag is cleared to 0 in channel reset mode or by the program writing 0 to this flag. To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1". 0: No transmit history interrupt request is present. 1: A transmit history interrupt request is present. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | THLELT | B'0 | R/W | <p>Transmit History Buffer Overflow Flag</p> <p>The THLELT flag is set to 1 when an attempt is made to store new transmit history data while the transmit history buffer is full. In this case, the new data is discarded. This flag becomes 0 in channel reset mode or by the program writing 0 to this flag.</p> <p>To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared.</p> <p>When writing 0, using store instruction, set the bit to be set to "0" to "0", and the bits not to be set to "0" to "1".</p> <p>0: Transmit history buffer overflow has not occurred. 1: Transmit history buffer overflow has occurred.</p> |
| 1 | THLFLL | B'0 | R | <p>Transmit History Buffer Full Status Flag</p> <p>The THLFLL flag is set to 1 when 16 data sets have been stored in the transmit history buffer, and is cleared to 0 when the number of data sets stored in the transmit history buffer has decreased to less than 16. This bit is also cleared to 0 in channel reset mode or when the THLE bit in the RSCFDnCFDTHLCCm register is set to 0 (transmit history buffer is not used).</p> <p>0: Transmit history buffer is not full. 1: Transmit history buffer is full.</p> |
| 0 | THLEMP | B'1 | R | <p>Transmit History Buffer Empty Status Flag</p> <p>The THLEMP flag is cleared to 0 when even a single set of transmit history data has been stored in the transmit history buffer.</p> <p>This flag is set to 1 when all the data in the transmit history buffer has been read. This flag is also set to 1 in channel reset mode or when the THLE bit in the RSCFDnCFDTHLCCm register is set to 0 (transmit history buffer is not used).</p> <p>0: Transmit history buffer contains unread data. 1: Transmit history buffer contains no unread data (buffer empty).</p> |

Note: To clear THLIF or THLELT flag to 0, the program must write 0. When writing, use a store instruction to write "0" to the given flag and "1" to other flags.

48.9.13.3 RSCFDnCFDTHLPCTRm — Transmit History Pointer Control Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDTHLPCTRm register can only be written in 32-bit units
 RSCFDnCFDTHLPCTRmL, RSCFDnCFDTHLPCTRmH registers can only be written in 16-bit units
 RSCFDnCFDTHLPCTRmLL, RSCFDnCFDTHLPCTRmLH, RSCFDnCFDTHLPCTRmHL,
 RSCFDnCFDTHLPCTRmHH registers can only be written in 8-bit units

Address: RSCFDnCFDTHLPCTRm: <RSCFDn_base> + H'0440 + (H'04 × m)
 RSCFDnCFDTHLPCTRmL: <RSCFDn_base> + H'0440 + (H'04 × m),
 RSCFDnCFDTHLPCTRmH: <RSCFDn_base> + H'0442 + (H'04 × m)
 RSCFDnCFDTHLPCTRmLL: <RSCFDn_base> + H'0440 + (H'04 × m),
 RSCFDnCFDTHLPCTRmLH: <RSCFDn_base> + H'0441 + (H'04 × m),
 RSCFDnCFDTHLPCTRmHL: <RSCFDn_base> + H'0442 + (H'04 × m),
 RSCFDnCFDTHLPCTRmHH: <RSCFDn_base> + H'0443 + (H'04 × m)

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | THLPC[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | W | W | W | W | W | W | W | W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 8 | Reserved | All 0 | R | When writing to these bits, write the value after reset. |
| 7 to 0 | THLPC[7:0] | H'00 | W | <p>Transmit History List Pointer Control</p> <p>When the THLPC[7:0] bits are set to H'FF, the read pointer moves to the next data in the transmit history buffer. At this time, the THLMC[4:0] (transmit history buffer unread data counter) value in the RSCFDnCFDTHLSTSm register is decremented. Write H'FF to the THLPC[7:0] bits after reading from the RSCFDnCFDTHLACCm register.</p> <p>When writing H'FF to these bits, make sure that the THLE bit in the RSCFDnCFDTHLCCm register is set to 1 (transmit history buffer is used) and the THLEMP flag in the RSCFDnCFDTHLSTSm register is 0.</p> |

48.9.13.4 RSCFDnCFDTHLACCm — Transmit History Access Register (m = 0 or 1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDTHLACCm register can be read only in 32-bit units

RSCFDnCFDTHLACCmL, RSCFDnCFDTHLACCmH registers can be read only in 16-bit units

RSCFDnCFDTHLACCmLL, RSCFDnCFDTHLACCmLH, RSCFDnCFDTHLACCmHL, RSCFDnCFDTHLACCmHH registers can be read only in 8-bit units

Address: RSCFDnCFDTHLACCm: <RSCFDn_base> + H'6000 + (H'04 × m)

RSCFDnCFDTHLACCmL: <RSCFDn_base> + H'6000 + (H'04 × m),

RSCFDnCFDTHLACCmH: <RSCFDn_base> + H'6002 + (H'04 × m)

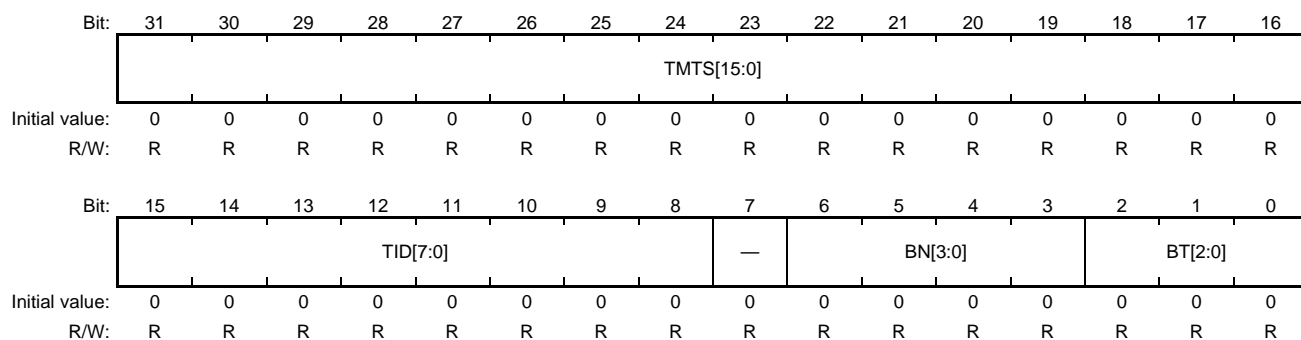
RSCFDnCFDTHLACCmLL: <RSCFDn_base> + H'6000 + (H'04 × m),

RSCFDnCFDTHLACCmLH: <RSCFDn_base> + H'6001 + (H'04 × m),

RSCFDnCFDTHLACCmHL: <RSCFDn_base> + H'6002 + (H'04 × m),

RSCFDnCFDTHLACCmHH: <RSCFDn_base> + H'6003 + (H'04 × m)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 16 | TMTS[15:0] | H'0000 | R | Timestamp Data These bits indicate the timestamp value of the transmit history data stored in the transmit history buffer. These bits are always read as 0 when the TMTSCE bit is 0. |
| 15 to 8 | TID[7:0] | H'00 | R | Label Data These bits indicate the label information of transmit history data stored in the transmit history buffer. |
| 7 | Reserved | B'0 | R | When read, the value after reset is returned. |
| 6 to 3 | BN[3:0] | H'0 | R | Buffer Number Data These bits indicate the transmit source buffer number in the transmit history data stored in the transmit history buffer. |
| 2 to 0 | BT[2:0] | B'000 | R | Buffer Type Data These bits indicate the type of the transmit source buffer in the transmit history data stored in the transmit history buffer. B'001: Transmit buffer B'010: Transmit/receive FIFO buffer B'100: Transmit queue |

48.9.14 Details of Test Related Registers

48.9.14.1 RSCFDnCFDGTSTCFG — Global Test Configuration Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGTSTCFG register can be read/written in 32-bit units

RSCFDnCFDGTSTCFGL, RSCFDnCFDGTSTCFGH registers can be read/written in 16-bit units

RSCFDnCFDGTSTCFGLL, RSCFDnCFDGTSTCFGLH, RSCFDnCFDGTSTCFGHL, RSCFDnCFDGTSTCFGHH registers can be read/written in 8-bit units

Address: RSCFDnCFDGTSTCFG: <RSCFDn_base> + H'0468

RSCFDnCFDGTSTCFGL: <RSCFDn_base> + H'0468,

RSCFDnCFDGTSTCFGH: <RSCFDn_base> + H'046A

RSCFDnCFDGTSTCFGLL: <RSCFDn_base> + H'0468,

RSCFDnCFDGTSTCFGLH: <RSCFDn_base> + H'0469,

RSCFDnCFDGTSTCFGHL: <RSCFDn_base> + H'046A,

RSCFDnCFDGTSTCFGHH: <RSCFDn_base> + H'046B

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|------------|-----|-----|-----|-----|-------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | RTMPS[6:0] | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | C1ICB CE | C0ICB CE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 23 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 22 to 16 | RTMPS[6:0] | H'00 | R/W | RAM Test Page Configuration These bits are used to set the RAM test target page number for RAM test. Set a value in the range of H'00 to H'53, inclusive. |
| 15 to 2 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 1 | C1ICBCE | B'0 | R/W | CAN1 Inter-Channel Communication Test Enable Setting this bit to 1 enables the channel 1 inter-channel communication test. 0: CAN1 inter-channel communication test is disabled. 1: CAN1 inter-channel communication test is enabled. |
| 0 | C0ICBCE | B'0 | R/W | CAN0 Inter-Channel Communication Test Enable Setting this bit to 1 enables the channel 0 inter-channel communication test. 0: CAN0 inter-channel communication test is disabled. 1: CAN0 inter-channel communication test is enabled. |

Modify the RSCFDnCFDGTSTCFG register only in global test mode.

48.9.14.2 RSCFDnCFDGTSTCTR — Global Test Control Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGTSTCTR register can be read/written in 32-bit units,
 RSCFDnCFDGTSTCTRL, RSCFDnCFDGTSTCTRH registers can be read/written in 16-bit units
 RSCFDnCFDGTSTCTRLL, RSCFDnCFDGTSTCTRLH, RSCFDnCFDGTSTCTRHL, RSCFDnCFDGTSTCTRHH
 registers can be read/written in 8-bit units

Address: RSCFDnCFDGTSTCTR: <RSCFDn_base> + H'046C
 RSCFDnCFDGTSTCTRL: <RSCFDn_base> + H'046C,
 RSCFDnCFDGTSTCTRH: <RSCFDn_base> + H'046E
 RSCFDnCFDGTSTCTRLL: <RSCFDn_base> + H'046C,
 RSCFDnCFDGTSTCTRLH: <RSCFDn_base> + H'046D,
 RSCFDnCFDGTSTCTRHL: <RSCFDn_base> + H'046E,
 RSCFDnCFDGTSTCTRHH: <RSCFDn_base> + H'046F

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|------|----|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | RTME | — | ICBCT ME |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 3 | Reserved | All 0 | R | When read, the value after reset is returned. When writing to these bits, write the value after reset. |
| 2 | RTME | B'0 | R/W | RAM Test Enable Setting this bit to 1 enables the RAM test. Modify this bit only in global test mode. <ol style="list-style-type: none"> Set the GMDC[1:0] bits in the RSCFDnCFDGTSTCTR register to B'10 (Global test mode). Set the RTME bit to 1. Check that the RTME bit is set to 1. 0: RAM test is disabled. 1: RAM test is enabled. |
| 1 | Reserved | B'0 | R | When read, the value after reset is returned. When writing to this bit, write the value after reset. |
| 0 | ICBCTME | B'0 | R/W | Communication Test between Channels Enable When this bit is set to 1, a communication test is enabled between the channels for which the CmICBCE bit (m = 0 or 1) in the RSCFDnCFDGTSTCFG register has been set to 1. Modify the ICBCTME bit only in global test mode. 0: Communication test between channels disabled 1: Communication test between channels enabled |

48.9.14.3 RSCFDnCFDGLCKK — Global Lock Key Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDGLCKK register can only be written in 32-bit units

RSCFDnCFDGLCKKL, RSCFDnCFDGLCKKH registers can only be written in 16-bit units

Address: RSCFDnCFDGLCKK: <RSCFDn_base> + H'047C

RSCFDnCFDGLCKKL: <RSCFDn_base> + H'047C, RSCFDnCFDGLCKKH: <RSCFDn_base> + H'047E

Value after reset: H'0000_0000

| | | | | | | | | | | | | | | | | |
|----------------|------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | LOCK[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | W* | W* | W* | W* | W* | W* | W* | W* | W* | W* | W* | W* | W* | W* | W* | W* |

Note: * Writing to these bits is effective only when the RS-CANFD module is in global test mode.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 16 | Reserved | All 0 | R | When writing to these bits, write the value after reset. |
| 15 to 0 | LOCK[15:0] | H'0000 | W | <p>Lock Key</p> <p>Writing the protection release data to the LOCK[15:0] bits in succession enables writing 1 to the RTME bit in the RSCFDnCFDGTSTCTR register.</p> <p>After the protection has been released, writing to the I/O register area (<RSCFDn_base> + H'0000 to <RSCFDn_base> + H'0FFF) of the CAN (except the RAM) enables the protection again.</p> <p>Reading from the I/O register area of the CAN or reading from/writing to other areas does not enable the protection.</p> |

The RSCFDnCFDGLCKK register releases protection of special test bits and is write only.

For the protection release data, see section 48.16.4.2, Procedure for Releasing the Protection.

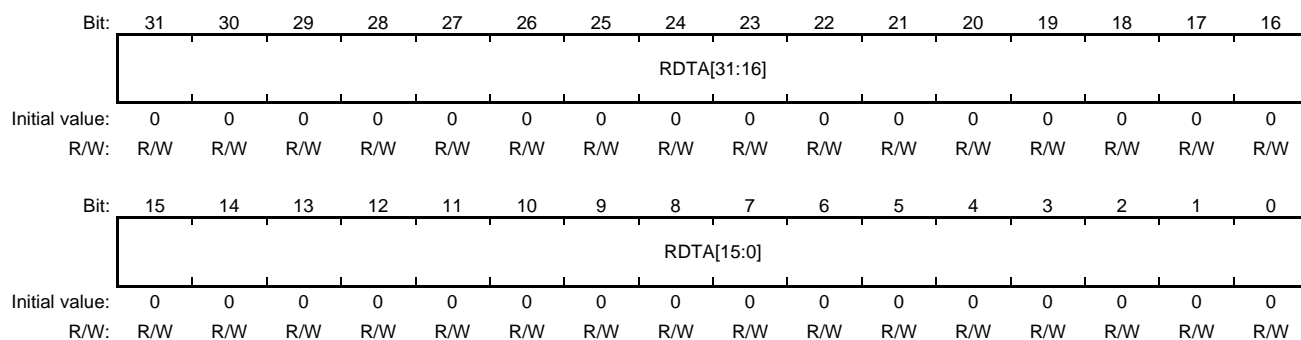
48.9.14.4 RSCFDnCFDRPGACCr — RAM Test Page Access Register (r = 0 to 63)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Access: RSCFDnCFDRPGACCr register can be read/written in 32-bit units
 RSCFDnCFDRPGACCrL, RSCFDnCFDRPGACCrH registers can be read/written in 16-bit units
 RSCFDnCFDRPGACCrLL, RSCFDnCFDRPGACCrLH, RSCFDnCFDRPGACCrHL, RSCFDnCFDRPGACCrHH registers can be read/written in 8-bit units

Address: RSCFDnCFDRPGACCr: <RSCFDn_base> + H'6400 + (H'04 × r)
 RSCFDnCFDRPGACCrL: <RSCFDn_base> + H'6400 + (H'04 × r),
 RSCFDnCFDRPGACCrH: <RSCFDn_base> + H'6402 + (H'04 × r)
 RSCFDnCFDRPGACCrLL: <RSCFDn_base> + H'6400 + (H'04 × r),
 RSCFDnCFDRPGACCrLH: <RSCFDn_base> + H'6401 + (H'04 × r),
 RSCFDnCFDRPGACCrHL: <RSCFDn_base> + H'6402 + (H'04 × r),
 RSCFDnCFDRPGACCrHH: <RSCFDn_base> + H'6403 + (H'04 × r)

Value after reset: H'0000_0000



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 0 | RDTA[31:0] | H'0000_0000 | R/W | RAM Data Test Access Data can be read and written in RSCAN RAM. |

Modify the RSCFDnCFDRPGACCr register in global test mode with the RTME bit in the RSCFDnCFDGTSTCTR register set to 1 (RAM test is enabled).

The RSCFDnCFDRPGACCr register is readable and writable when the RTME bit is set to 1.

48.10 Interrupt Sources and DMA Triggers

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

48.10.1 Interrupt Sources

Interrupt sources are classified into global and channel interrupts.

- Global interrupts (2 sources):
 - Receive FIFO interrupt
 - Global error interrupt
- Channel interrupts (3 sources/channel):
 - CANm transmit interrupt (m = 0 or 1)
 - CANm transmit complete interrupt
 - CANm transmit abort interrupt
 - CANm transmit/receive FIFO transmit complete interrupt (in transmit mode, gateway mode)
 - CANm transmit history interrupt
 - CANm transmit queue interrupt
 - CANm transmit/receive FIFO receive complete interrupt (in receive mode, gateway mode)
 - CANm error interrupt

When an interrupt request is generated, the corresponding interrupt request flag is set to 1 (interrupt request present). In that case, when the interrupt enable bit is set to 1 (enabling interrupts), an interrupt request is output from the RS-CANFD module. (Generation of interrupts also depends on the interrupt control register settings of the interrupt controller.)

Setting the interrupt request flag to 0 (no interrupt request present) or setting the interrupt enable bit to 0 (disabling interrupts) clears the current interrupt request. The current interrupt request is still output until the interrupt request flag is cleared.

Table 48.28 lists the CAN interrupt sources. Figure 48.2 shows the CAN global interrupt block diagram. Figure 48.3 shows the CAN channel interrupt block diagram.

Table 48.28 List of CAN Interrupt Sources

| | Interrupt Source | Corresponding Interrupt Request Flag | Corresponding Interrupt Enable Bit | |
|---------------------------------|-------------------------|--|---|---|
| Global interrupts | Receive FIFO | Receive FIFO 0 | RFIF in the RSCFDn(CFD)RFSTS0 register | RFIE in the RSCFDn(CFD)RFCC0 register |
| | | Receive FIFO 1 | RFIF in the RSCFDn(CFD)RFSTS1 register | RFIE in the RSCFDn(CFD)RFCC1 register |
| | | Receive FIFO 2 | RFIF in the RSCFDn(CFD)RFSTS2 register | RFIE in the RSCFDn(CFD)RFCC2 register |
| | | Receive FIFO 3 | RFIF in the RSCFDn(CFD)RFSTS3 register | RFIE in the RSCFDn(CFD)RFCC3 register |
| | | Receive FIFO 4 | RFIF in the RSCFDn(CFD)RFSTS4 register | RFIE in the RSCFDn(CFD)RFCC4 register |
| | | Receive FIFO 5 | RFIF in the RSCFDn(CFD)RFSTS5 register | RFIE in the RSCFDn(CFD)RFCC5 register |
| | | Receive FIFO 6 | RFIF in the RSCFDn(CFD)RFSTS6 register | RFIE in the RSCFDn(CFD)RFCC6 register |
| | | Receive FIFO 7 | RFIF in the RSCFDn(CFD)RFSTS7 register | RFIE in the RSCFDn(CFD)RFCC7 register |
| | Global error | <ul style="list-style-type: none"> • DEF in the RSCFDn(CFD)GERFL register • MES in the RSCFDn(CFD)GERFL register • THLES in the RSCFDn(CFD)GERFL register • CMPOF in the RSCFDnCFDGERFL register | <ul style="list-style-type: none"> • DEIE in the RSCFDn(CFD)GCTR register • MEIE in the RSCFDn(CFD)GCTR register • THLEIE in the RSCFDn(CFD)GCTR register • CMPOFIE in the RSCFDnCFDGCTR register | |
| Channel interrupts (m = 0 or 1) | CANm transmit | CANm transmit complete | TMTRF[1:0] in the RSCFDn(CFD)TMSTSp register | TMIEp in the RSCFDn(CFD)TMIECy register |
| | | CANm transmit abort | TMTRF[1:0] in the RSCFDn(CFD)TMSTSp register | TAIE in the RSCFDn(CFD)CmCTR register |
| | | CANm transmit/receive FIFO transmit complete | CFTXIF in the RSCFDn(CFD)CFSTSk register | CFTXIE in the RSCFDn(CFD)CFCCk register |
| | | CANm transmit queue | TXQIF in the RSCFDn(CFD)TXQSTSm register | TXQIE in the RSCFDn(CFD)TXQCCm register |
| | | CANm transmit history | THLIF in the RSCFDn(CFD)THLSTSm register | THLIE in the RSCFDn(CFD)THLCCm register |
| | | CANm transmit/receive FIFO receive complete | CFRXIF in the RSCFDn(CFD)CFSTSk register | CFRXIE in the RSCFDn(CFD)CFCCk register |

| Interrupt Source | Corresponding Interrupt Request Flag | Corresponding Interrupt Enable Bit | |
|------------------------------------|--------------------------------------|--|--|
| Channel interrupts (m = 0 or 1) | CANm error | <ul style="list-style-type: none"> • BEF in the RSCFDn(CFD)CmERFL register • ALF in the RSCFDn(CFD)CmERFL register • BLF in the RSCFDn(CFD)CmERFL register • OVLf in the RSCFDn(CFD)CmERFL register • BORF in the RSCFDn(CFD)CmERFL register • BOEF in the RSCFDn(CFD)CmERFL register • EPF in the RSCFDn(CFD)CmERFL register • EWF in the RSCFDn(CFD)CmERFL register • SOCO in the RSCFDn(CFD)CmERFL register • EOCO in the RSCFDn(CFD)CmERFL register • TDCVF in the RSCFDn(CFD)CmERFL register | <ul style="list-style-type: none"> • BEIE in the RSCFDn(CFD)CmCTR register • ALIE in the RSCFDn(CFD)CmCTR register • BLIE in the RSCFDn(CFD)CmCTR register • OLIE in the RSCFDn(CFD)CmCTR register • BORIE in the RSCFDn(CFD)CmCTR register • BOEIE in the RSCFDn(CFD)CmCTR register • EPIE in the RSCFDn(CFD)CmCTR register • EWIE in the RSCFDn(CFD)CmCTR register • SOCOIE in the RSCFDnCFDCmCTR register • EOCOIE in the RSCFDnCFDCmCTR register • TDCVFIE in the RSCFDnCFDCmCTR register |

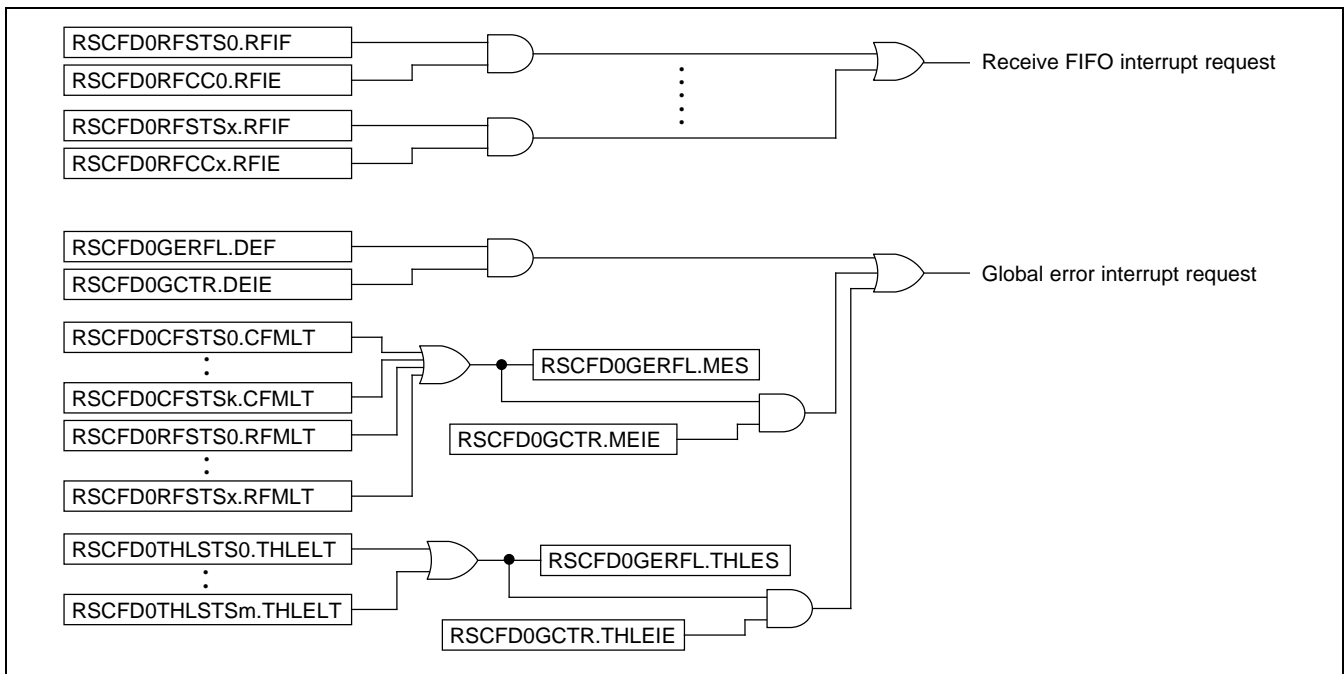


Figure 48.2 CAN Global Interrupt Block Diagram

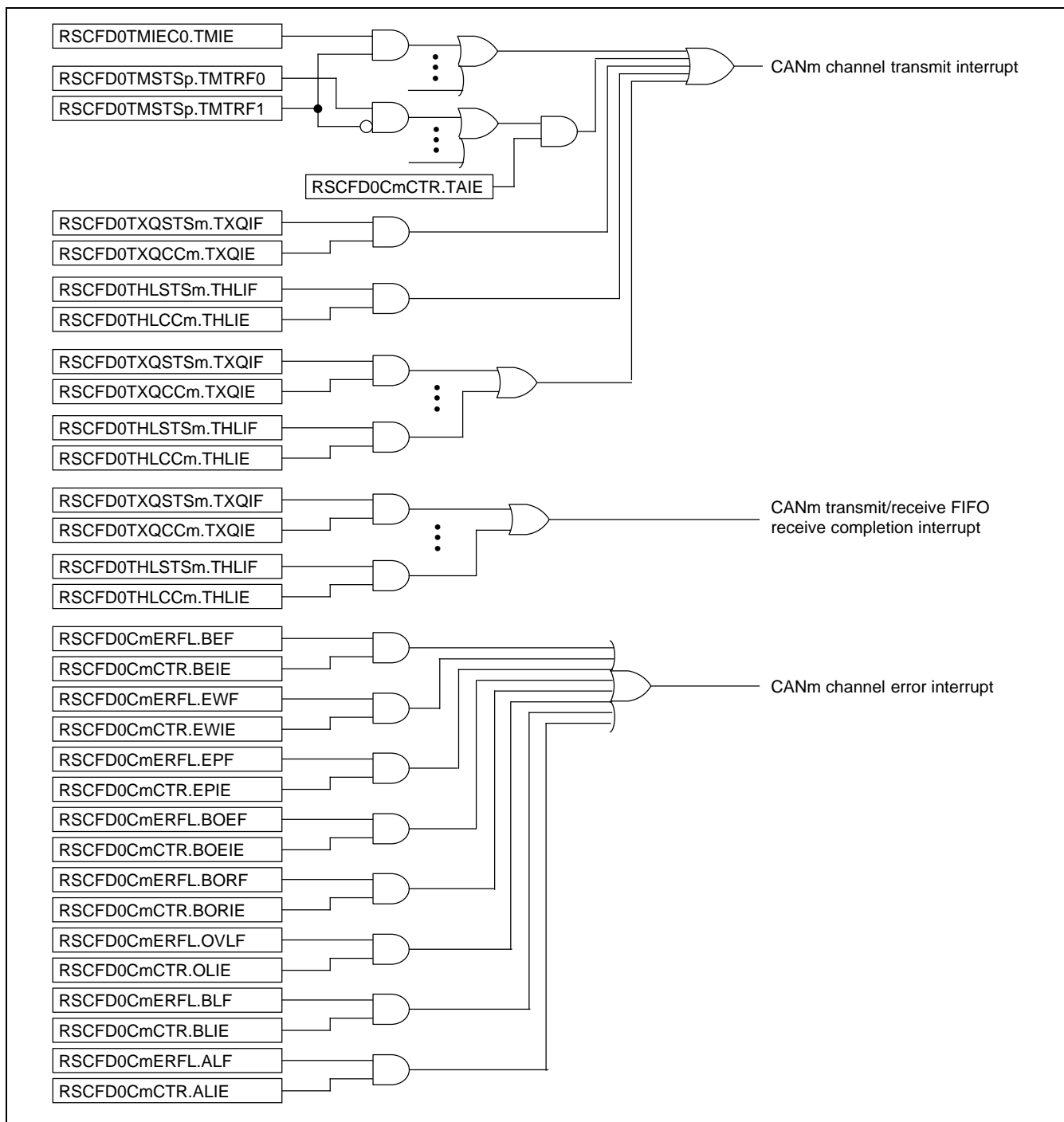


Figure 48.3 CAN Channel Interrupt Block Diagram

48.10.2 DMA Triggers (CAN FD Mode Only)

In CAN FD mode, the FIFO buffer for reception can be related to DMA channels. The following 10 FIFO buffers are available.

- All receive FIFO buffers x (x = 0 to 7)
- The first transmit/receive FIFO buffer k ($k = 3 \times m$, m = 0 or 1) assigned to channel m

When the DMA enable bit (the RFDMAEx or CFDMAEm bit in the RSCFDnCFDCDTCT register) is set to 1 and there are unread messages in the related FIFO, a DMA transfer request trigger is generated.

48.11 CAN Modes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The RS-CANFD module has four global modes to control the entire RS-CANFD module status and four channel modes to control individual channel status. Details of global modes are described in section 48.11.1, Global Modes, and details of channel modes are described in section 48.11.2, Channel Modes.

- Global stop mode: Stops the clocks of the entire module to achieve low power consumption.
- Global reset mode: Performs initial settings for the entire module.
- Global test mode: Performs test settings and performs the RAM test.
- Global operating mode: Makes the entire module operable.
- Channel stop mode: Stops the channel clock.
- Channel reset mode: Performs initial settings for the channels.
- Channel halt mode: Stops CAN communication and allows channel testing.
- Channel communication mode: Performs CAN communication.

48.11.1 Global Modes

Figure 48.4 shows the transitions of global modes.

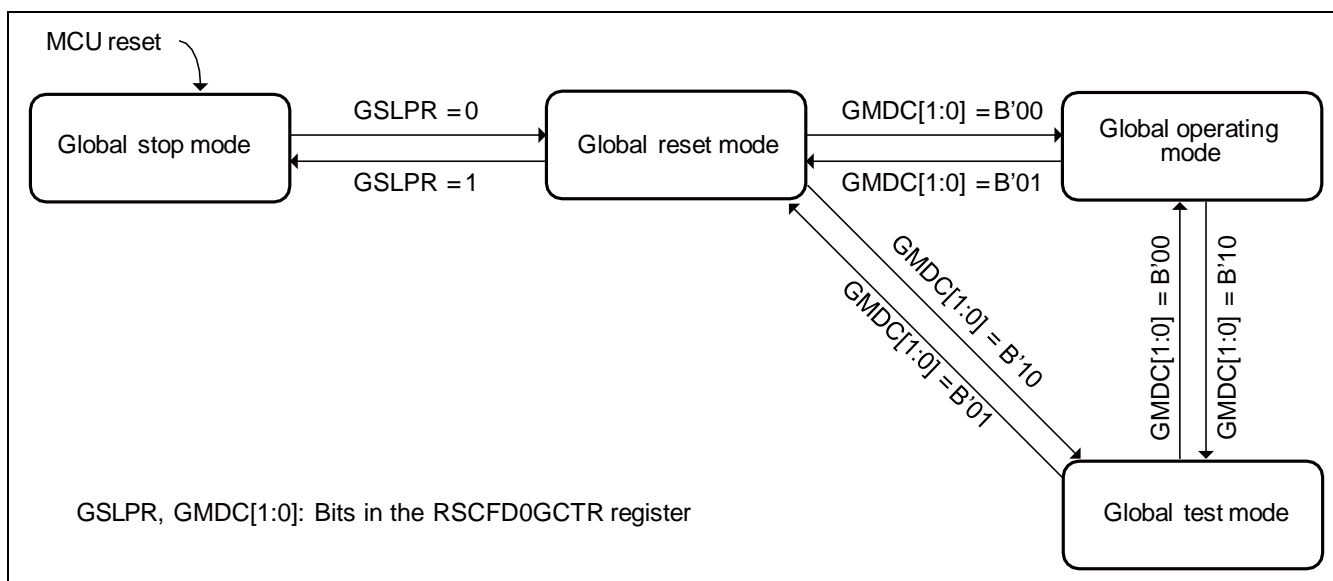


Figure 48.4 Transitions of Global Modes

In some cases, global mode transitions also force channel mode transitions. Table 48.29 shows the channel mode transitions depending on the global mode setting dictated by the GMDC[1:0] bits and the GSLPR bit.

Table 48.29 Transitions of Channel Modes Depending on Global Mode Setting (GMDC[1:0] and GSLPR Bits)

| Channel Mode before Setting | Channel Mode after Setting | | | |
|-----------------------------|---|--|---|--|
| | GMDC[1:0] = B'00 GSLPR = 0 (Global Operation) | GMDC[1:0] = B'10 GSLPR = 0 (Global Test) | GMDC[1:0] = B'01 GSLPR = 0 (Global Reset) | GMDC[1:0] = B'01 GSLPR = 1 (Global Stop) |
| Channel communication | Channel communication | Channel halt | Channel reset | Transition prohibited |
| Channel halt | Channel halt | Channel halt | Channel reset | Transition prohibited |
| Channel reset | Channel reset | Channel reset | Channel reset | Channel stop |
| Channel stop | Channel stop | Channel stop | Channel stop | Channel stop |

Note: GMDC[1:0], GSLPR: Bits in the RSCFDn(CFD)GCTR register

Table 48.30 shows the global mode transition time.

Table 48.30 Global Mode Transition Time

| Mode before Transition | Mode after Transition | Maximum Transition Time |
|------------------------|-----------------------|-----------------------------------|
| Global stop | Global reset | Three pclk cycles |
| Global reset | Global stop | Three pclk cycles |
| Global reset | Global test | Ten pclk cycles |
| Global reset | Global operating | Ten pclk cycles |
| Global test | Global reset | Two CAN bit times ^{*1,2} |
| Global test | Global operating | Three pclk cycles |
| Global operating | Global reset | Two CAN bit times ^{*1,2} |
| Global operating | Global test | Two CAN frames ^{*1} |

Notes: 1. It is the CAN bit time and CAN frame time with the lowest communication speed of the channels in use.
2. In CAN FD mode, it is the CAN bit time with normal bit rate.

48.11.1.1 Global Stop Mode

In global stop mode, clocks of the CAN do not run and therefore power consumption is reduced. CAN registers can be read, but writing data to them is prohibited. Register values are retained. Only the clock used by the CPU for writing to the GSLPR bit runs in this mode.

After the MCU is reset, the RS-CANFD module transitions to global stop mode. Setting the GSLPR bit in the RSCFDn(CFD)GCTR register to 1 (in global stop mode) in global reset mode sets the CSLPR bit in each of the RSCFDn(CFD)CmCTR register to 1 (channel stop mode). Afterwards, if all channels are forced to transition to channel stop mode, the RS-CANFD module transitions to global stop mode. The GSLPR bit should not be modified in global operating mode or global test mode.

48.11.1.2 Global Reset Mode

In global reset mode, RS-CANFD module settings are performed. When the RS-CANFD module transitions to global reset mode, some registers are initialized. Table 48.33 and Table 48.34 list the registers to be initialized.

Setting the GMDC[1:0] bits in the RSCFDn(CFD)GCTR register to B'01 sets the CHMDC[1:0] bits in each of the RSCFDn(CFD)CmCTR registers (m = 0 or 1) to B'01 (channel reset mode). If all channels are forced to transition to channel reset mode, the RS-CANFD module transitions to global reset mode. Channels that are already in channel reset mode or channel stop mode do not transition (because the CHMDC[1:0] bits have already been set to B'01).

48.11.1.3 Global Test Mode

In global test mode, settings for test-related registers are performed. When the CAN module transitions to global test mode, all CAN communications are disabled.

Setting the GMDC[1:0] bits in the RSCFDn(CFD)GCTR register to B'10 sets the CHMDC[1:0] bits in each of the RSCFDn(CFD)CmCTR register to B'10 (channel halt mode). If all channels are forced to transition to channel halt mode, the CAN module transitions to global test mode. Channels that are in channel stop mode, channel reset mode, or channel halt mode do not transition.

48.11.1.4 Global Operating Mode

The RS-CANFD module operates in global operating mode.

When the GMDC[1:0] bits in the RSCFDn(CFD)GCTR register are set to B'00, the RS-CAN module transitions to global operating mode.

48.11.2 Channel Modes

Figure 48.5 shows a channel mode state transition chart. Table 48.31 shows the channel mode transition time.

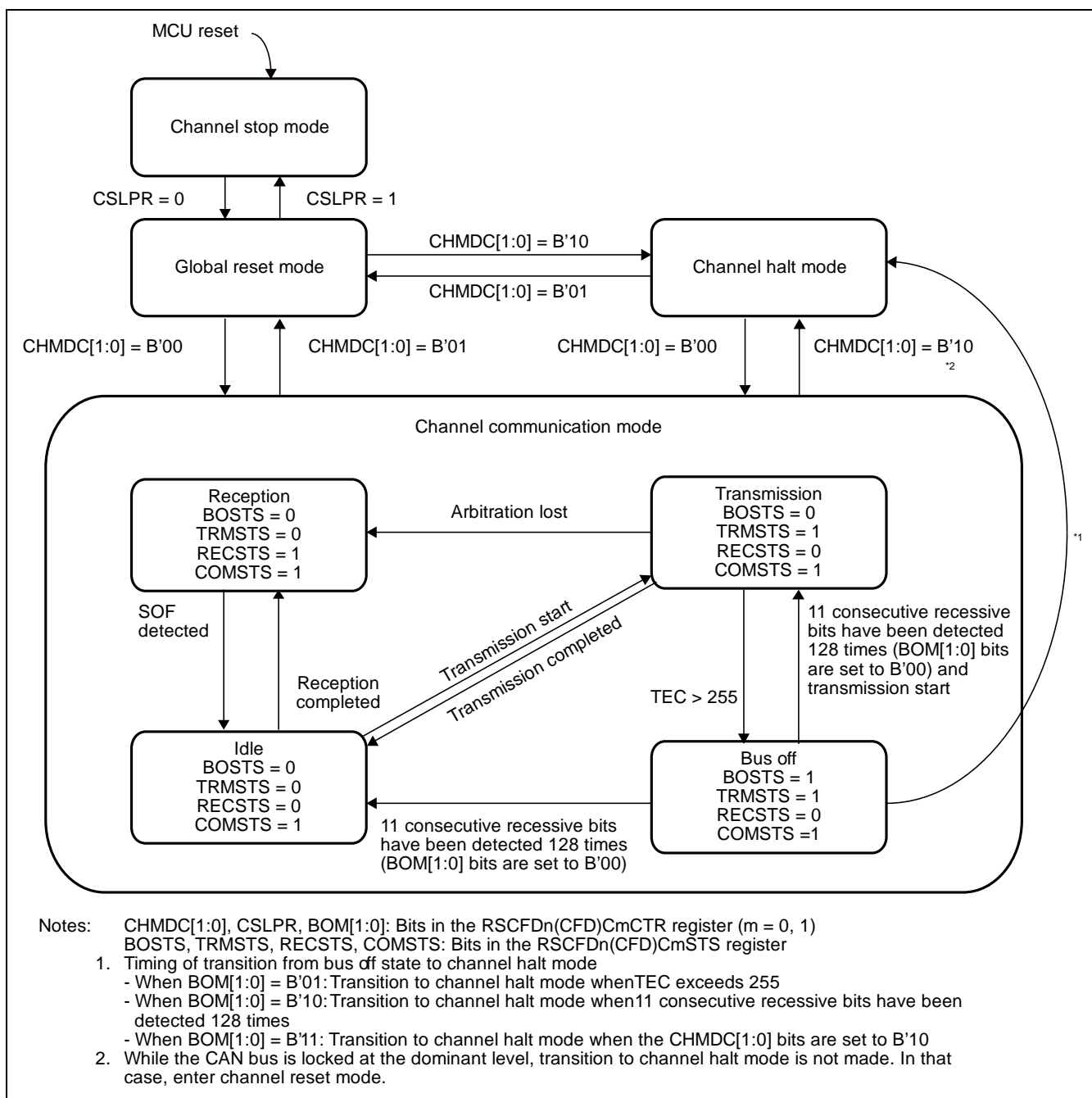


Figure 48.5 Channel Mode State Transition Chart

Table 48.31 Channel Mode Transition Time

| Mode before Transition | Mode after Transition | Maximum Transition Time |
|------------------------|-----------------------|-------------------------|
| Channel stop | Channel reset | Three pclk cycles |
| Channel reset | Channel stop | Three pclk cycles |
| Channel reset | Channel halt | Three CANm bit times* |
| Channel reset | Channel communication | Three CANm bit times* |

| Mode before Transition | Mode after Transition | Maximum Transition Time |
|------------------------|-----------------------|-------------------------|
| Channel halt | Channel reset | Two CANm bit times* |
| Channel halt | Channel communication | Four CANm bit times* |
| Channel communication | Channel reset | Two CANm bit times* |
| Channel communication | Channel halt | Two CANm frames |

Note: * In CAN FD mode, it is the CANm bit time with normal bit rate.

48.11.2.1 Channel Stop Mode

In channel stop mode, clocks are not supplied to channels and therefore power consumption is reduced. CAN registers can be read, but writing data to them is prohibited. Register values are retained.

Each channel enters channel stop mode after the MCU is reset. Channels also transition to channel stop mode when the CSLPR bit in the RSCFDn(CFD)CmCTR register (m = 0 or 1) is set to 1 (channel stop mode) in channel reset mode. The CSLPR bit should not be modified in channel communication mode and channel halt mode.

48.11.2.2 Channel Reset Mode

In channel reset mode, channel settings are performed. When a channel transitions to channel reset mode, some channel-related registers are initialized. Table 48.33 lists the registers to be initialized.

When the CHMDC[1:0] bits in the RSCFDn(CFD)CmCTR register are set to B'01 (channel reset mode) during CAN communication, communication is terminated before it is completed and the channel transitions to channel reset mode. Table 48.32 shows the operation when the CHMDC[1:0] bits are set to B'01 (channel reset mode) during CAN communication.

48.11.2.3 Channel Halt Mode

In channel halt mode, settings for test-related registers of channels are performed. When a channel transitions to channel halt mode, CAN communication of the channel stops.

Table 48.32 shows operation when the CHMDC[1:0] bits are set to B'10 (channel halt mode) during CAN communication.

Table 48.32 Operation a Channel Transitions to Channel Reset Mode/Channel Halt Mode

| Mode | During Reception | During Transmission | Bus Off State |
|---|--|---|---|
| Channel reset (CHMDC[1:0] = B'01) | Transitions to channel reset mode before reception is completed.* ¹ | Transitions to channel reset mode before transmission is completed.* ¹ | Transitions to channel reset mode before bus off recovery. |
| Channel halt* ³ (CHMDC[1:0] = B'10) | Transitions to channel halt mode after reception is completed.* ² | Transitions to channel halt mode after transmission is completed. | [When BOM[1:0] = B'00] Transitions to channel halt mode (CHMDC[1:0] = B'10) only after bus off recovery. [When BOM[1:0] = B'01] Transitions to channel halt mode automatically when the condition for transition to bus off state is met. [When BOM[1:0] = B'10] Transitions to channel halt mode automatically after bus off recovery. [When BOM[1:0] = B'11] Transitions to channel halt mode immediately after the CHMDC[1:0] bits are set to B'10 before bus off recovery. |

- Notes: 1. To allow transition to channel reset mode after communication is completed, set the CHMDC[1:0] bits to B'10 and confirm that communication has been completed and transition to channel halt mode has been made, and then set the CHMDC[1:0] bits to B'01.
2. While the CAN bus is locked at the dominant level, transition to channel halt mode is not made. In that case, enter channel reset mode. The CAN bus status can be confirmed with the BLF flag of the RSCFDn(CFD)CmERFL register that becomes 1 when dominant lock is detected.
3. When the transition from channel reset mode to channel halt mode is to be made, in classical CAN mode, set the RSCFDn(CFD)CmCFG register in channel reset mode and then shift to channel halt mode. In CAN FD mode, make transition to channel halt mode after setting the RSCFDn(CFD)CmNCFG and RSCFDn(CFD)CmDCFG registers.

48.11.2.4 Channel Communication Mode

In channel communication mode, CAN communication is performed. Each channel has the following communication states during CAN communication.

- Idle: Neither reception nor transmission is in progress.
- Reception: Receiving a message sent from another node.
- Transmission: Transmitting a message.
- Bus off: Isolated from CAN communication.

When the CHMDC[1:0] bits in the RSCFDn(CFD)CmCTR register are set to B'00, the channel transitions to channel communication mode. After that, once 11 consecutive recessive bits have been detected, the COMSTS flag in the RSCFDn(CFD)CmSTS register (m = 0 or 1) is set to 1 (communication is ready) and transmission and reception are enabled on the CAN network as an active node. At this time, transmission and reception of messages can be started.

48.11.2.5 Bus Off State

A channel transitions to the bus off state according to the transmit/receive error counter increment/decrement rules of the CAN specifications.

The conditions for returning from the bus off state are determined by the BOM[1:0] bits in the RSCFDn(CFD)CmCTR register.

- When BOM[1:0] = B'00:

Bus off recovery is compliant with the CAN specifications. After 11 consecutive recessive bits have been detected 128 times, a channel returns from the bus off state to the CAN communication ready state (error active state). At that time, the TEC[7:0] and REC[7:0] bits in the RSCFDn(CFD)CmERFL register are initialized to H'00, the BORF flag in the RSCFDn(CFD)CmERFL register is set to 1 (bus off recovery is detected), and a bus off recovery interrupt request is generated. When the CHMDC[1:0] bits in the RSCFDn(CFD)CmCTR register are set to B'10 (channel halt mode) in the bus off state, the channel transitions to channel halt mode after bus off recovery has been completed (11 consecutive recessive bits have been detected 128 times).

- When BOM[1:0] = B'01:

When a channel transitions to the bus off state, the CHMDC[1:0] bits are set to B'10 and the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to H'00. The BORF flag is not set to 1, and bus off recovery interrupt request is not generated.

- When BOM[1:0] = B'10:

When a channel has transitioned to the bus off state, the CHMDC[1:0] bits are set to B'10. After bus off recovery has been completed (11 consecutive recessive bits have been detected 128 times), the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to H'00, the BORF flag is set to 1, and a bus off recovery interrupt request is generated.

- When BOM[1:0] = B'11:

When the CHMDC[1:0] bits are set to B'10 in the bus off state, the channel transitions to channel halt mode before bus off recovery is completed. At that time, the TEC[7:0] and REC[7:0] bits are initialized to H'00, but the BORF flag is not set to 1. Also, a bus off recovery interrupt is not generated.

However, the BORF flag becomes 1 and a bus off recovery interrupt request is generated if a CAN module transitions to error active state (by detecting 128 times of 11 consecutive recessive bits) before CHMDC[1:0] bits are set to B'10.

If the RS-CANFD module causes the channel to transition to channel halt mode simultaneously with a program write to the CHMDC[1:0] bits, the program write takes precedence. An automatic transition to channel halt mode when the BOM[1:0] bits are set to B'01 or B'10 is made only when the CHMDC[1:0] bits are B'00 (channel communication mode).

Furthermore, setting the RTBO bit in the RSCFDn(CFD)CmCTR register to 1 allows a forced return from the bus off state. As soon as the RTBO bit is set to 1, the state changes to the error active state. After 11 consecutive recessive bits have been detected, the CAN module becomes ready for communication. In this case, the BORF flag is not set to 1 and the TEC[7:0] and REC[7:0] bits are initialized to H'00. Write 1 to the RTBO bit only when the BOM[1:0] value is B'00. Writing the RTBO bit to 1 in a state other than the bus off state is ignored, and the RTBO bit is immediately set to 0.

48.11.3 Registers Initialized by Transition to CAN Mode

Table 48.33 shows bits and flags initialized by transition to channel reset mode. These bits and flags are also initialized by transition to global reset mode. Table 48.34 shows bits and flags that are only initialized by transition to global reset mode.

Table 48.33 Registers Initialized in Global Reset Mode or Channel Reset Mode

| Register | Bit / Flag |
|-------------------------------|---|
| RSCFDn(CFD)CmCTR register | (ROM), CRCT, CTMS[1:0], CTME, CHMDC[1:0] |
| RSCFDn(CFD)CmSTS register | CHLTSTS, EPSTS, BOSTS, TRMSTS, RECSTS, COMSTS, (ESIF), REC[7:0], TEC[7:0] |
| RSCFDn(CFD)CmERFL register | CRCREG[14:0], ADERR, B0ERR, B1ERR, CERR, AERR, FERR, SERR, ALF, BLF, OVLF, BORF, BOEF, EPF, EWF, BEF |
| RSCFDnCFDCmFDCTR register | EOCCLR, SOCCLR |
| RSCFDnCFDCmFDSTS register | SOC[7:0], EOC[7:0], SOCO, EOCO, TDCVF, TDCR[6:0] |
| RSCFDnCFDCmFDCRC register | CRCREG[20:0] |
| RSCFDn(CFD)CFCK register | When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFE |
| RSCFDn(CFD)CFSTSk register | When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFMC[7:0], CFFLL, CFEMP, CFMLT, CFRXIF, CFTXIF |
| RSCFDn(CFD)CFTISTS register | CFkTXIF |
| RSCFDn(CFD)TMCp register | TMOM, TMTAR, TMTR |
| RSCFDn(CFD)TMSTSp register | TMTARM, TMTRM, TMTRF[1:0], TMTSTS |
| RSCFDn(CFD)TMTRSTSy register | TMTRSTSp (Bits of corresponding channel are initialized in channel reset mode.) |
| RSCFDn(CFD)TMTARSTSy register | TMTARSTSp (Bits of corresponding channel are initialized in channel reset mode.) |
| RSCFDn(CFD)TMTCSTSy register | TMTCSTSp (Bits of corresponding channel are initialized in channel reset mode.) |
| RSCFDn(CFD)TMTASTSy register | TMTASTSp (Bits of corresponding channel are initialized in channel reset mode.) |
| RSCFDn(CFD)TXQCCm register | TXQE |
| RSCFDn(CFD)TXQSTSm register | TXQIF, TXQFLL, TXQEMP |
| RSCFDn(CFD)THLCCm register | THLE |
| RSCFDn(CFD)THLSTSm register | THLMC[4:0], THLIF, THLELT, THLFLL, THLEMP |
| RSCFDn(CFD)GTINTSTS0 register | TSIFm, TAIEm, TQIFm, CFTIFm, THIFm (m = 0 or 1) |
| RSCFDn(CFD)GTINTSTS1 register | TSIFm, TAIEm, TQIFm, CFTIFm, THIFm (m = 4 or 5) |

Note: Bits and flags enclosed by () only exist in registers for CAN FD mode.

Table 48.34 Registers Initialized Only in Global Reset Mode

| Register | Bit / Flag |
|-----------------------------|--|
| RSCFDn(CFD)GSTS register | GHLTSTS |
| RSCFDn(CFD)GERFL register | EEF0, EEF1, EEF2, EEF3, EEF4, EEF5, (CMPOF), THLES, MES, DEF |
| RSCFDn(CFD)GTSC register | TS[15:0] |
| RSCFDn(CFD)RMNDy register | RMNSq |
| RSCFDn(CFD)RFCCx register | RFE |
| RSCFDn(CFD)RFSTsx register | RFMC[7:0], RFIF, RFMLT, RFFLL, RFEMP |
| RSCFDn(CFD)CFCCk register | When transmit/receive FIFO buffer is in receive mode: CFE |
| RSCFDn(CFD)CFSTSk register | When transmit/receive FIFO buffer is in receive mode: CFMC[7:0], CFFLL, CFEMP, CFTXIF, CFRXIF, CFMLT |
| RSCFDn(CFD)FESTS register | CFkEMP, RFxEMP |
| RSCFDn(CFD)FFSTS register | CFkFLL, RFxFLL |
| RSCFDn(CFD)FMSTS register | CFkMLT, RFxMLT |
| RSCFDn(CFD)RFISTS register | RFxIF |
| RSCFDn(CFD)CFRISTS register | CFkRXIF |
| RSCFDnCFDCDTCT register | CFDMAEm, RFDMAEx |
| RSCFDnCFDCDTSTS register | CFDMASTSm, RFDMASTsx |
| RSCFDn(CFD)GTSTCFG register | RTMPS[6:0], C0ICBCE, C1ICBCE, C2ICBCE, C3ICBCE, C4ICBCE, C5ICBCE |
| RSCFDn(CFD)GTSTCTR register | RTME, ICBCTME |

Note: Bits and flags enclosed by () only exist in registers for CAN FD mode.

48.12 Reception Function

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

There are two reception types.

Reception by receive buffers:

- 0 to 31 receive buffers can be shared by all channels. Since messages stored in receive buffers are overwritten at each reception, the latest receive data can always be read.
- Reception by receive FIFO buffers and transmit/receive FIFO buffers (receive mode):
Eight receive FIFO buffers can be shared by all channels and three dedicated transmit/receive FIFO buffers are provided for each channel. Messages of up to the number of buffer stages specified with the RFDC[2:0] and CFDC[2:0] bits can be stored in FIFO buffers and can be read sequentially from the oldest.

48.12.1 Data Processing Using the Receive Rule Table

Data processing using the receive rule table allows dispatching of selected messages to the specified buffer. Data processing includes acceptance filter processing, DLC filter processing, routing processing, label addition processing, and mirror function processing.

Up to 128 receive rules can be registered per channel and up to (64 × number of channels) total receive rules can be registered in the entire module. (Up to 128 receive rules can be registered in this module that has two channels.) Set receive rules for each channel. Receive rules cannot be shared with other channels. If receive rules are not set, no messages can be received. Figure 48.6 illustrates how receive rules are registered.

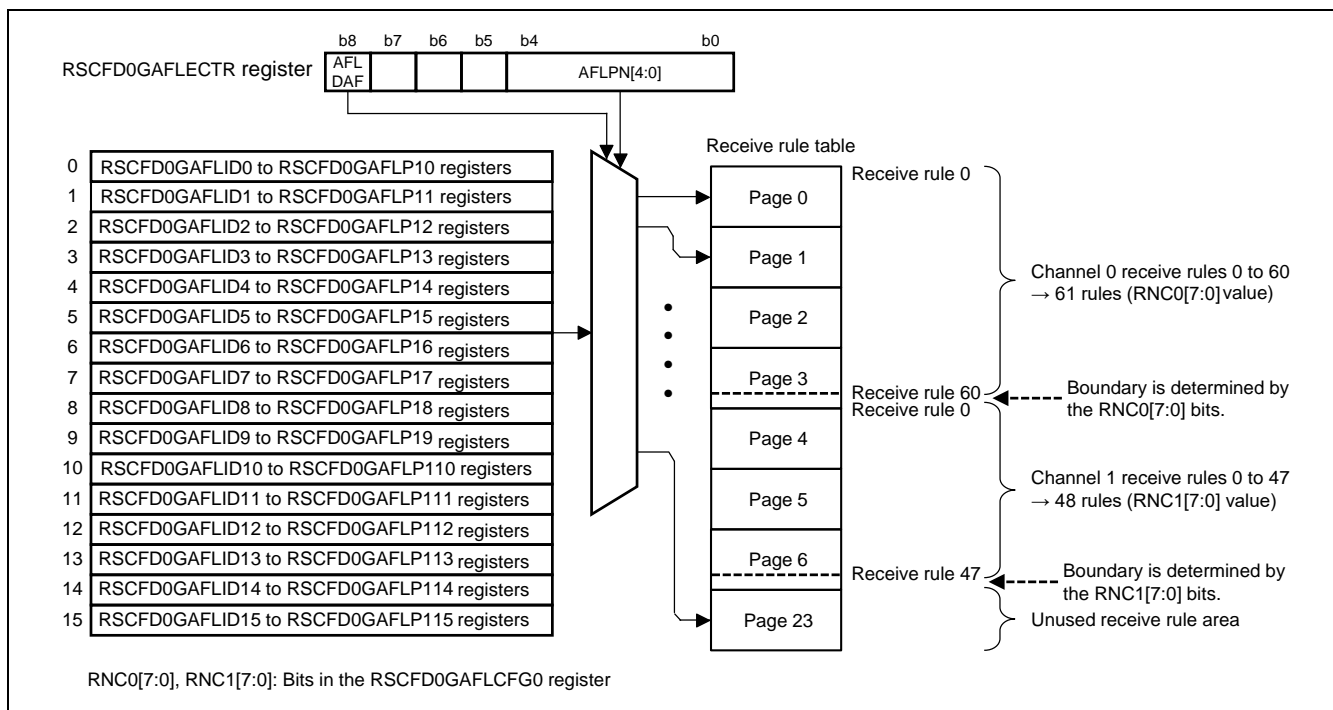


Figure 48.6 Entry of Receive Rules (for Setting Channels 0 and 1)

Note: Receive rules for each channel must be set in contiguous blocks. Channel 1 rules and channel 0 rules must be set separately.

Each receive rule consists of 16 bytes in the RSCFDn(CFD)GAFLIDj, RSCFDn(CFD)GAFLMj, RSCFDn(CFD)GAFLP0_j, and RSCFDn(CFD)GAFLP1_j registers (j = 0 to 15). The RSCFDn(CFD)GAFLIDj register is used to set GAFLID, GAFLIDE bit, GAFLRTR bit, and the mirror function, the RSCFDn(CFD)GAFLMj register is used to set mask, the RSCFDn(CFD)GAFLP0_j register is used to set label information to be added, DLC value, and storage receive buffer, and the RSCFDn(CFD)GAFLP1_j register is used to set storage FIFO buffer. Up to 16 receive rules can be set per page.

48.12.1.1 Acceptance Filter Processing

In the acceptance filter processing, the ID data, IDE bit, and RTR bit in a received message are compared with the ID data, IDE bit, and RTR bit set in the receive rule of the corresponding channel. When all these bits match, the message passes through the acceptance filter processing. The ID data, IDE bit, and RTR bit in the received message which correspond to the bits set to 0 (bits are not compared) in the RSCFDn(CFD)GAFLMj register are not compared and are regarded as matched.

Check begins with the receive rule of the minimum number for the corresponding channel. When all the bits to be compared in a received message match the bits set in the receive rule or when all the receive rules are compared without any match, filter processing stops. If there is no matching receive rule, the received message is not stored in the receive buffer or FIFO buffer.

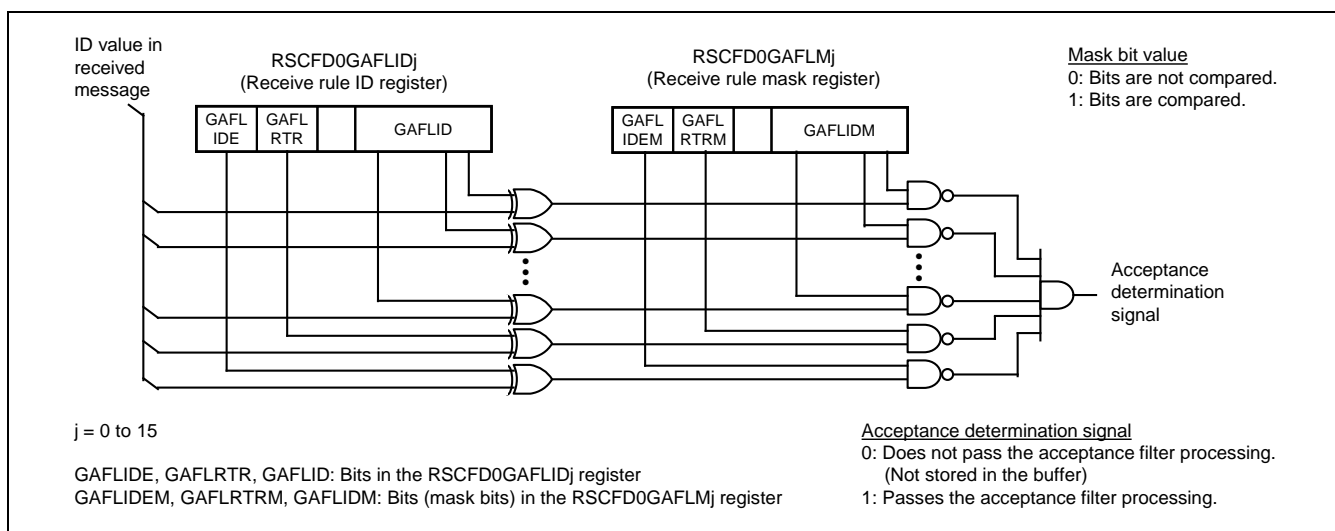


Figure 48.7 Acceptance Filter Function

48.12.1.2 DLC Filter Processing

When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), DLC filter processing is added to messages that passed through the acceptance filter processing. When the DLC value in a message is equal to or larger than the DLC value set in the receive rule, the message passes through the DLC filter processing.

When a message has passed through the DLC filter processing with the DRE bit in the RSCFDn(CFD)GCFG register set to 0 (DLC replacement is disabled), the DLC value in the received message is stored in the buffer. In this case, all the data bytes in the received message are stored in the buffer.

When a message has passed through the DLC filter processing with the DRE bit in the RSCFDn(CFD)GCFG register set to 1 (DLC replacement is enabled), the DLC value in the receive rule is stored in the buffer instead of the DLC value in the received message. In this case, a value of H'00 is stored in each data byte beyond the number of bytes which is indicated by the DLC value in the receive rule.

When the DLC value in the received message is smaller than that in the receive rule, the message does not pass through the DLC filter processing. In this case, the message is not stored in the receive buffer or the FIFO buffer and the DEF flag in the RSCFDn(CFD)GERFL register is set to 1 (a DLC error is present).

48.12.1.3 Routing Processing

Messages that passed through the acceptance filter processing and the DLC filter processing are stored in receive buffers, receive FIFO buffers, or transmit/receive FIFO buffers (set to receive mode or gateway mode). Message storage destination is set by the GAFLRMV and GAFLRMDP[6:0] bits in the RSCFDn(CFD)GAFLP0_j register (j = 0 to 15) and by the RSCFDn(CFD)GAFLP1_j register. Messages that passed through the acceptance filter processing and the DLC filter processing can be stored in up to eight buffers.

In CAN FD mode, when the payload length of the received message exceeds the size for storing the payload in the storage buffer, the CMPOF flag in the RSCFDnCFDGERFL register is set to 1 (payload overflow) and handled according to the CMPOC bit in the RSCFDnCFDGCFG register. When the CMPOC bit is 0, a message which exceeds the size for storing the payload is not stored in the buffer. When the CMPOC bit is 1, the payload which exceeds the storage size is truncated and a message is stored in the buffer.

48.12.1.4 Label Addition Processing

It is possible to add 12-bit label information to messages that passed through the filter processing and store them in buffers. This label information is set in the GAFLPTR[11:0] bits in the RSCFDn(CFD)GAFLP0_j register.

48.12.1.5 Mirror Function Processing

The mirror function allows the CAN node to receive its own transmitted messages. The mirror function is made available by setting the MME bit in the RSCFDn(CFD)GCFG register to 1 (mirror function is enabled).

When the mirror function is in use, receive rules for which the GAFLLB bit in the RSCFDn(CFD)GAFLIDj register is set to 0 are used for data processing when receiving messages transmitted from other CAN nodes. When the CAN node is receiving its own transmitted messages, receive rules for which the GAFLLB bit is set to 1 are used for data processing.

48.12.1.6 Timestamp

The timestamp counter is a 16-bit free-running counter used for recording the time for receiving and transmitting messages. The timestamp counter value is fetched at the start-of-frame (SOF) timing of a message and stored in a receive buffer or a FIFO buffer with a message ID and data at reception. The clock source of the timestamp counter is selected by the TSBTCS[2:0] and TSSS bits in the RSCAN0GCFG register; it is selected by pclk/2 or the CANm bit time clock (m = 0 or 1) in classical CAN mode and pclk/2 or the CANm normal bit time clock in CAN FD mode. However, do not select the CANm normal bit time clock of a channel which handles the CAN FD frame. The count source of the timestamp counter is obtained by a clock which divides the selected clock source by the TSP[3:0] bits in the RSCAN0GCFG register.

When the CANm bit time clock or CANm normal bit time clock is used as a clock source, the timestamp counter stops when the corresponding channel transitions to channel reset mode or channel halt mode. When the pclk/2 is used as a clock source, the timestamp function is not affected by channel mode.

The timestamp counter value is reset to H'0000 by setting the TSRST bit in the RSCFDn(CFD)GCTR register to 1.

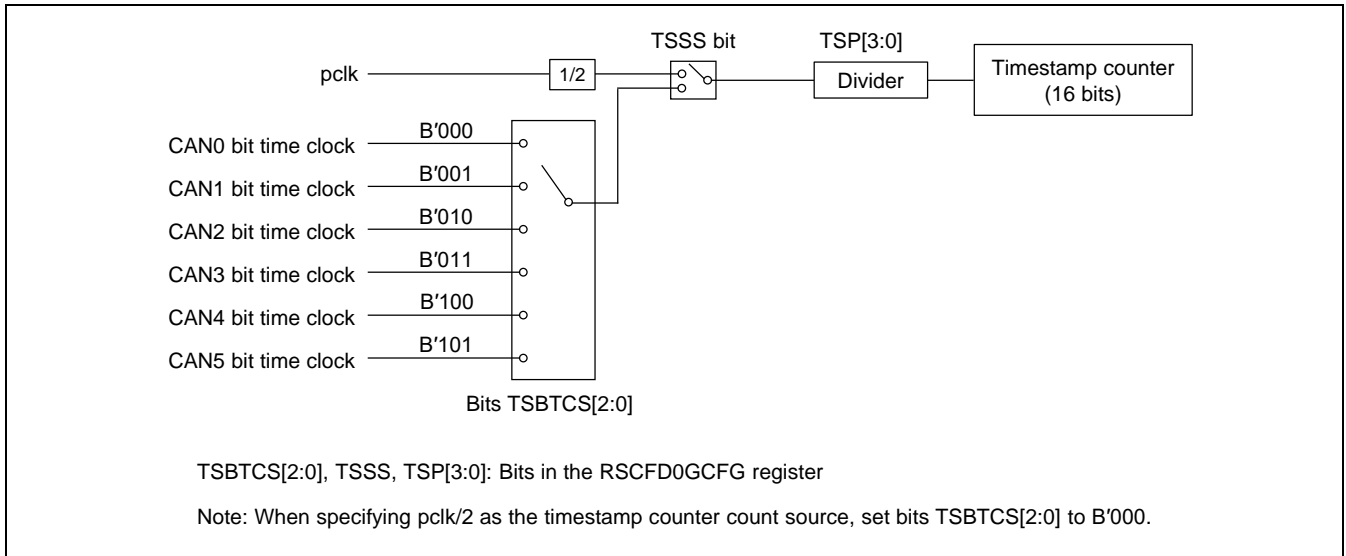


Figure 48.8 Timestamp Function Block Diagram

48.13 Transmission Function

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

There are three types of transmission. In classical CAN mode, the 8-byte payload length can be transmitted in each type of transmission. In CAN FD mode, the payload length to be transmitted differs depending on the type of transmission.

- Transmission using transmit buffers:**
 Each channel has 16 buffers. In CAN FD mode, the 20-byte payload length can be transmitted. However, when transmit buffer merge mode is used, four of 16 buffers are allocated as storage areas for the payload and two buffers can be used to transmit the payload which exceeds 20 bytes.
- Transmission using transmit/receive FIFO buffers (transmit mode):**
 Each channel has three FIFO buffers. Up to 128 messages can be contained in a single FIFO buffer. In CAN FD mode, the 64-byte payload length can be transmitted. Each FIFO buffer is used with a link to a transmit buffer. Only the message to be transmitted next in a FIFO buffer becomes the target of transmit priority determination. Messages are transmitted sequentially on a first-in, first-out basis.
- Transmission using transmit queues:**
 Up to 16 transmit buffers per channel can be allocated to the transmit queues. In CAN FD mode, the 20-byte payload length can be transmitted. Transmit buffer $((16 \times m) + 15)$ is used as an access window of a corresponding channel. Transmit buffers are allocated to transmit queues in descending order of buffer number. All messages in transmit queues, which are targets of priority determination, are transmitted in the order of ID number.

Figure 48.9 shows the allocation of transmit queues and transmit/receive FIFO buffer link.

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--|-------------------|--------------------------------|-------------------|--------------------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------------------|---|--------------------------------|-------------------|-------------------|-------------------|-------------------|--------------------------------|-------------------|-------------------|-------------------|-------------------|--------------------|--------------------------------|----------------|----------------|----------------|----------------|--|-------------------|-------------------|-------------------|-------------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| <table border="1"> <tr><td>Transmit buffer 0</td></tr> <tr><td>Transmit buffer 1</td></tr> <tr><td>Transmit buffer 2</td></tr> <tr><td>Transmit buffer 3</td></tr> <tr><td>Transmit buffer 4</td></tr> <tr><td>Transmit buffer 5</td></tr> <tr><td>Transmit buffer 6</td></tr> <tr><td>Transmit buffer 7</td></tr> <tr><td>Transmit buffer 8</td></tr> <tr><td>Transmit buffer 9</td></tr> <tr><td>Transmit buffer 10</td></tr> <tr><td>Transmit buffer 11</td></tr> <tr><td>Transmit buffer 12</td></tr> <tr><td>Transmit buffer 13</td></tr> <tr><td>Transmit buffer 14</td></tr> <tr><td>Transmit buffer 15</td></tr> </table> <p>Only transmit buffers are used</p> | Transmit buffer 0 | Transmit buffer 1 | Transmit buffer 2 | Transmit buffer 3 | Transmit buffer 4 | Transmit buffer 5 | Transmit buffer 6 | Transmit buffer 7 | Transmit buffer 8 | Transmit buffer 9 | Transmit buffer 10 | Transmit buffer 11 | Transmit buffer 12 | Transmit buffer 13 | Transmit buffer 14 | Transmit buffer 15 | <table border="1"> <tr><td>Transmit buffer 0</td></tr> <tr><td>Transmit/receive FIFO buffer 0</td></tr> <tr><td>Transmit buffer 2</td></tr> <tr><td>Transmit/receive FIFO buffer 1</td></tr> <tr><td>Transmit buffer 4</td></tr> <tr><td>Transmit buffer 5</td></tr> <tr><td>Transmit buffer 6</td></tr> <tr><td>Transmit buffer 7</td></tr> <tr><td>Transmit buffer 8</td></tr> <tr><td>Transmit buffer 9</td></tr> <tr><td>Transmit buffer 10</td></tr> <tr><td>Transmit buffer 11</td></tr> <tr><td>Transmit buffer 12</td></tr> <tr><td>Transmit buffer 13</td></tr> <tr><td>Transmit buffer 14</td></tr> <tr><td>Transmit/receive FIFO buffer 2</td></tr> </table> <p>Transmit buffers and transmit/receive FIFO buffers are used (Transmit/receive FIFO buffers are linked to transmit buffers 1, 3, and 15)</p> | Transmit buffer 0 | Transmit/receive FIFO buffer 0 | Transmit buffer 2 | Transmit/receive FIFO buffer 1 | Transmit buffer 4 | Transmit buffer 5 | Transmit buffer 6 | Transmit buffer 7 | Transmit buffer 8 | Transmit buffer 9 | Transmit buffer 10 | Transmit buffer 11 | Transmit buffer 12 | Transmit buffer 13 | Transmit buffer 14 | Transmit/receive FIFO buffer 2 | <table border="1"> <tr><td>Transmit/receive FIFO buffer 0</td></tr> <tr><td>Transmit buffer 1</td></tr> <tr><td>Transmit buffer 2</td></tr> <tr><td>Transmit buffer 3</td></tr> <tr><td>Transmit buffer 4</td></tr> <tr><td>Transmit/receive FIFO buffer 1</td></tr> <tr><td>Transmit buffer 6</td></tr> <tr><td>Transmit buffer 7</td></tr> <tr><td>Transmit buffer 8</td></tr> <tr><td>Transmit buffer 9</td></tr> <tr><td>Transmit buffer 10</td></tr> <tr><td>Transmit/receive FIFO buffer 2</td></tr> <tr><td>Transmit queue</td></tr> <tr><td>Transmit queue</td></tr> <tr><td>Transmit queue</td></tr> <tr><td>Transmit queue</td></tr> </table> <p>Transmit buffers, transmit/receive FIFO buffers, and a transmit queue are used (Transmit/receive FIFO buffers are linked to transmit buffers 0, 5, and 11; Four transmit queue elements are allocated to transmit buffers)</p> | Transmit/receive FIFO buffer 0 | Transmit buffer 1 | Transmit buffer 2 | Transmit buffer 3 | Transmit buffer 4 | Transmit/receive FIFO buffer 1 | Transmit buffer 6 | Transmit buffer 7 | Transmit buffer 8 | Transmit buffer 9 | Transmit buffer 10 | Transmit/receive FIFO buffer 2 | Transmit queue | Transmit queue | Transmit queue | Transmit queue | <table border="1"> <tr><td>Transmit buffer 0</td></tr> <tr><td>Transmit buffer 1</td></tr> <tr><td>Transmit buffer 2</td></tr> <tr><td>Transmit buffer 3</td></tr> <tr><td>Transmit queue</td></tr> <tr><td>Transmit queue</td></tr> <tr><td>Transmit queue</td></tr> <tr><td>Transmit queue</td></tr> <tr><td>Transmit queue</td></tr> <tr><td>Transmit queue</td></tr> <tr><td>Transmit queue</td></tr> <tr><td>Transmit queue</td></tr> <tr><td>Transmit queue</td></tr> <tr><td>Transmit queue</td></tr> <tr><td>Transmit queue</td></tr> <tr><td>Transmit queue</td></tr> </table> <p>Transmit buffers and a transmit queue are used (12 transmit queue elements are allocated to transmit buffers)</p> | Transmit buffer 0 | Transmit buffer 1 | Transmit buffer 2 | Transmit buffer 3 | Transmit queue | Transmit queue | Transmit queue | Transmit queue | Transmit queue | Transmit queue | Transmit queue | Transmit queue | Transmit queue | Transmit queue | Transmit queue | Transmit queue |
| Transmit buffer 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 8 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 9 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 10 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 11 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 12 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 13 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 14 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 15 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit/receive FIFO buffer 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit/receive FIFO buffer 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 8 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 9 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 10 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 11 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 12 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 13 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 14 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit/receive FIFO buffer 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit/receive FIFO buffer 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit/receive FIFO buffer 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 8 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 9 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 10 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit/receive FIFO buffer 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit buffer 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Transmit queue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Figure 48.9 Allocation of Transmit Queues and Transmit/Receive FIFO Buffer Links

48.13.1 Transmit Priority Determination

If transmit requests are issued from multiple buffers or from the queue on the same channel, transmit priority is determined using one of the following methods.

The priority is determined by using one of the following methods.

- ID priority (TPRI bit = 0)
- Transmit buffer number priority (TPRI bit = 1)

All CAN channels use the setting of the TPRI bit in the RSCFDn(CFD)GCFG register.

When the TPRI bit is set to 0, messages are transmitted according to the priority of stored message IDs. ID priority conforms to the CAN bus arbitration specification defined in the CAN specifications. All IDs of pending transmit messages are targets of priority determination, regardless of whether they are stored in transmit buffers, transmit/receive FIFO buffers (set to transmit mode or gateway mode), or the transmit queue. If even a single transmit queue is used, select ID priority. When transmit/receive FIFO buffers are used, the oldest message in a FIFO buffer becomes the target of priority determination.

When a message is being transmitted from a transmit/receive FIFO buffer, the next message in the FIFO buffer becomes the target of priority determination. When a transmit queue is used, all messages in the transmit queue are targets of priority determination. If the same ID is set for two or more buffers, the buffer with the smaller buffer number takes precedence. When the TPRI bit is set to 1, the message in the transmit buffer with the minimum buffer number among all buffers with a transmit request is transmitted first. When transmit/receive FIFO buffers are linked to transmit buffers, transmit priority is determined according to linked transmit buffer numbers.

When messages are retransmitted due to an arbitration-lost or an error, transmit priority determination is made again regardless of the TPRI bit. If the ECC2 bit error is detected in priority determination, transmission will not be performed (only when the EEFE bit in the RSCFDnGCFG register is 1 in classical CAN mode).

48.13.2 Transmission Using Transmit Buffers

Setting the transmit request bit (TMTR bit in the RSCFDn(CFD)TMCp register) in a transmit buffer to 1 (transmission is requested) allows transmission of data frames or remote frames.

The transmit result is shown by the TMTRF[1:0] flag in the corresponding RSCFDn(CFD)TMSTSp register (p = 0 to 31). When transmit completes successfully, the TMTRF[1:0] flag is set to B'10 (transmission has been completed (without transmit abort request)) or B'11 (transmission has been completed (with transmit abort request)).

48.13.2.1 Transmit Abort Function

With respect to transmit buffers for which the TMTRM bit in the RSCFDn(CFD)TMSTSp register is set to 1 (a transmit request is present), when the TMTAR bit in the RSCFDn(CFD)TMCp register is set to 1 (transmit abort is requested), the transmit request is canceled. When transmit abort is completed, the TMTRF[1:0] flag in the RSCFDn(CFD)TMSTSp register is set to B'01 (transmit abort has been completed) and the transmit request is canceled (clearing the TMTRM bit to 0).

A message that is being transmitted or a message to be transmitted next according to the transmit priority determination cannot be aborted. However, when an arbitration-lost or an error occurs during transmission of a message for which the TMTAR bit is set to 1, retransmission is not performed.

48.13.2.2 One-Shot Transmission Function (Retransmission Disabling Function)

When the TMOM bit in the RSCFDn(CFD)TMCp register is set to 1 (one-shot transmission is enabled), transmission is performed only once. Even if an arbitration-lost or an error occurs, retransmission is not performed.

The one-shot transmit result is shown by the TMTRF[1:0] flag in the corresponding RSCFDn(CFD)TMSTSp register. When one-shot transmission completes successfully, the TMTRF[1:0] flag is set to B'10 or B'11. When an arbitration-lost or an error occurs, the TMTRF[1:0] flag is set to B'01 (transmit abort has been completed).

48.13.2.3 Transmit Buffer Merge Mode (CAN FD Mode Only)

Although the 20-byte payload length can be transmitted from the transmit buffer, it is possible to transmit messages having a maximum of 64-byte payload length after merging three transmit buffers in transmit buffer merge mode.

When the TMME bit in the RSCFDnCFDCmFDCFG register is set to 1, transmit buffer merge mode is enabled. In this mode, six buffers are used as merge areas in each channel and two transmit buffers $((16 \times m) + 0$ to $(16 \times m) + 2$ and $(16 \times m) + 3$ to $(16 \times m) + 5$) are merged. Transmission is requested in the first transmit buffer and the following two buffers are used as the payload storage area. For other transmit buffers than the first buffer, do not set 1 to the transmission request bit (the TMTR bit in the RSCFDnCFDTMCp register) and the transmission abort request bit (the TMTAR bit in the RSCFDnCFDTMCp register).

While transmit buffer merge mode is being enabled, do not link the transmit/receive FIFO buffer to six merged buffers or allocate the buffer to the transmit queue.

48.13.3 Transmission Using FIFO Buffers

Multiple messages can be stored in a single transmit/receive FIFO buffers, up to the number specified by the FIFO buffer depth, which is set by the CFDC[2:0] bits in the RSCFDn(CFD)CFCCk register ($k = 0$ to 5). Messages are transmitted sequentially on a first-in, first-out basis.

Each transmit/receive FIFO buffer is linked to a transmit buffer selected by the CFTML[3:0] bits in the RSCFDn(CFD)CFCCk register. When the CFE bit in the RSCFDn(CFD)CFCCk register is set to 1 (transmit/receive FIFO buffers are used), transmit/receive FIFO buffers become targets of transmit priority determination. Priority of only the next transmit message is determined in the FIFO buffer.

When the CFE bit is set to 0 (no transmit/receive FIFO buffer is used), the CFEMP flag is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)) at the timing below.

- The transmit/receive FIFO buffer becomes empty immediately if the message in it is not being transmitted or is not to be transmitted next.
- The transmit/receive FIFO buffer becomes empty after transmission completion, CAN bus error detection, or arbitration-lost in the case that a message in it is being transmitted or to be transmitted next.

When the CFE bit is cleared to 0, all messages in transmit/receive FIFO buffers are lost and messages cannot be stored in FIFO buffers. Confirm that the CFEMP flag is set to 1 before setting the CFE bit to 1 again.

48.13.3.1 Interval Transmission Function

A message transmission interval time can be set to space the transmission of messages from the same FIFO buffer when using a transmit/receive FIFO buffer set to transmit mode or gateway mode.

Immediately after the first message has been transmitted successfully from the FIFO buffer with the CFE bit in the RSCFDn(CFD)CFCCk register set to 1, the interval timer starts counting (after EOF7 of the CAN protocol). After that, when the interval time has passed, the next message is transmitted. The interval timer stops in channel reset mode or by clearing the CFE bit to 0.

The interval time is set by the CFITT[7:0] bits in the RSCFDn(CFD)CFCCk register. When the interval timer is not used, set the CFITT[7:0] bits to H'00.

Select an interval timer count source using the CFITR and CFITSS bits in the RSCFDn(CFD)CFCCk register. When the CFITR and CFITSS bits are set to B'00, the count source is obtained by dividing $plk/2$ by the value of the ITRCP[15:0] bits. When the CFITR and CFITSS bits are set to B'10, the count source is obtained by dividing $plk/2$ by (the value of

the ITRCP[15:0] bits in the RSCFDn(CFD)GCFG register $\times 10$). When the CFITR and CFITSS bits are set to B'x1, the CANm bit time clock and the CANm normal bit time clock are used as a count source in classical CAN mode and CAN FD mode, respectively.

The interval time is calculated by the following equations where M is the value of ITRCP[15:0] and N is the value of CFITT[7:0].

- When CFITR and CFITSS = B'00:

$$\frac{1}{\text{Frequency of pclk}} \times 2 \times M \times N$$

- When CFITR and CFITSS = B'10:

$$\frac{1}{\text{Frequency of pclk}} \times 2 \times M \times 10 \times N$$

- When CFITR and CFITSS = B'x1:

$$\text{Classical CAN mode} \quad : \quad \frac{1}{\text{Frequency of CANm bit time clock}} \times N$$

$$\text{CAN FD mode} \quad : \quad \frac{1}{\text{Frequency of CANm normal bit time clock}} \times N$$

Figure 48.10 shows the interval timer block diagram.

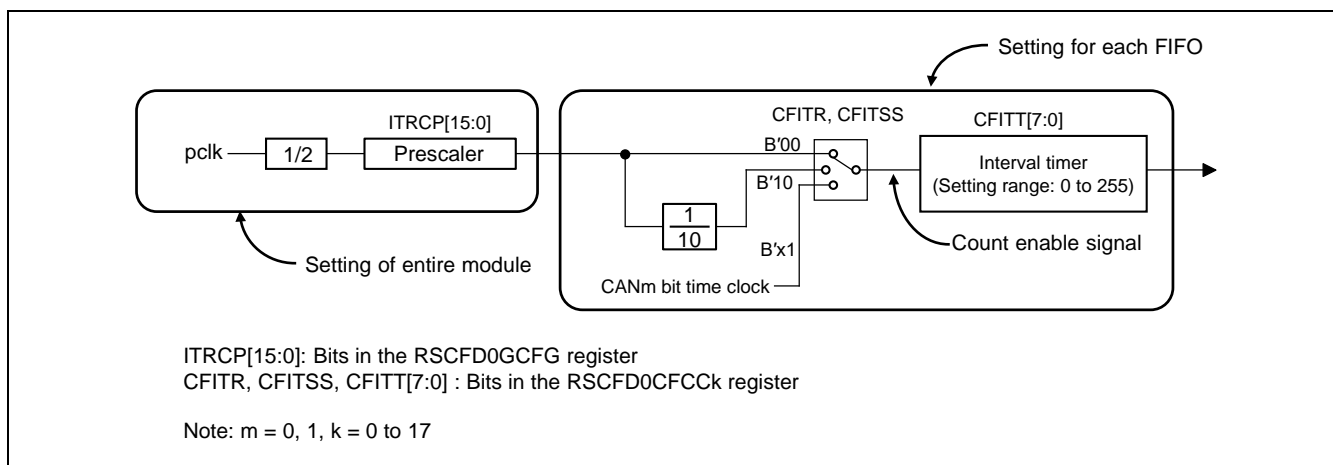


Figure 48.10 Interval Timer Block Diagram

Figure 48.11 shows the interval timer timing diagram.

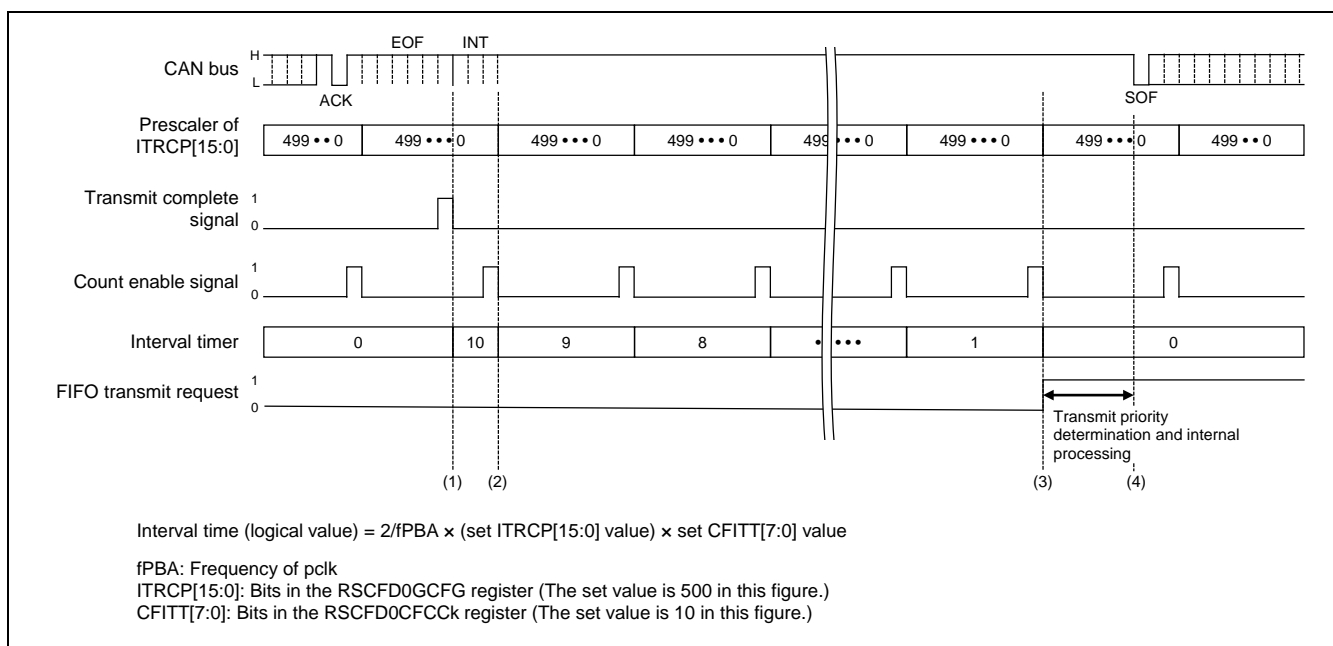


Figure 48.11 Interval Timer Timing Chart

- (1) The interval timer starts counting upon completion of transmission. Since the prescaler is not initialized at the time of transmission completion, the first interval time contains an error of up to one count of the interval timer.
- (2) The interval timer is decremented by the next count enable signal.
- (3) When the interval timer has decreased to 0, the transmit/receive FIFO buffer issues a transmit request.
- (4) The transmit/receive FIFO buffer is determined for the next transmission by the priority determination, it starts transmitting data. Transmission starts usually with a delay of three CANm bit time clock cycles or less from the issue of transmit request. If multiple internal processes (such as receive filter processing, message routing, and transmit priority determination) take place in all channels, a delay of up to 1164 cycles of the pclk may be generated.

48.13.4 Transmission Using Transmit Queues

Three to sixteen buffers (up to ten buffers in transmit buffer merge mode) are allocated to a transmit queue for each channel, and a transmit buffer $((16 \times m) + 15)$ is used as an access window of a corresponding channel. All messages in a transmit queue are targets of transmit priority determination and are transmitted in the ID priority order regardless of storage sequence. If two messages having the same ID are stored in a transmit queue, these messages are not always transmitted in the order of their storage in the transmit queue. Setting the TXQE bit in the RSCFDn(CFD)TXQCCm register to 0 disables transmit queues. When the TXQE bit is set to 0, the TXQEMP flag in the RSCFDn(CFD)TXQSTSm register is set to 1 (the transmit queue contains no messages (transmit queue empty)) at the timing below.

- The transmit queue becomes empty immediately when no message in it is being transmitted or will be transmitted next.
- The transmit queue becomes empty after transmission completion, CAN bus error detection, or arbitration-lost when a message in it is being transmitted or will be transmitted next.

When the TXQE bit is cleared to 0, all messages in transmit queues are lost and messages cannot be stored in transmit queues. Confirm that the TXQEMP flag is set to 1 before setting the TXQE bit to 1 again.

48.13.5 Padding Transmit Data (CAN FD Mode Only)

If the payload length indicated by the DLC value of the specified transmission message exceeds the size of the payload storage area of the transmit buffer, the exceeded payload will be padded by H'CC.

The processing is performed at the following cases when transmit merge mode is not enabled (the TMME bit in the RSCFDnCFDCmFDCFG register is 0).

- Transmit/receive FIFO which has been set for transmit or gateway mode:
The payload length of the transmission DLC exceeds the size of the payload storage area of the transmit/receive FIFO buffer which has been set for the CFPLS[2:0] bits in the RSCFDnCFDCFCCK register.
- Transmit buffer (including transmit queues):
The payload length of the transmit DLC exceeds 20 bytes.

When transmit buffer merge mode is enabled, the transmit data is not padded even if transmission is performed by using transmit buffers, transmit/receive buffers, or transmit queues. In this case, do not set the payload length, which exceeds the size of the payload storage area of the transmit buffer, to the DLC value of the transmission message.

48.13.6 Transmit History Function

Information on the transmitted messages can be stored in the transmit history buffer. Each channel has a single transmit history buffer that can contain 16 sets of transmit history data.

Types of message transmit source buffers can be selected by the THLDTE bit in the RSCFDn(CFD)THLCCm register. The THLEN bit in the RSCFDn(CFD)CFIDk register ($k = 0$ to 5) determines whether or not the transmit history data is stored for each message.

In classical CAN mode, the TMTSCE bit in the RSCFDnGCFG register determines whether or not a timestamp value is included in the transmit history data. In CAN FD mode, a timestamp value is always included.

The following information on a transmitted message will be stored in the transmission history buffer after transmission has successfully completed.

Storage of the transmission history data after successful completion of transmission may take up to 152 clock cycles of pclk.

- **Buffer type** B'001: Transmit buffer
B'010: Transmit/receive FIFO buffer
B'100: Transmit queue
- **Buffer number** Number of a transmit buffer, a transmit queue, or a transmit/receive FIFO buffer in the transmit source. This number depends on the buffer types. See Table 48.35.
- **Label data** Label information of the transmit message
- **Timestamp** Timestamp value of the transmit message
(In classical CAN mode, the TMTSCE bit is 1.)

Table 48.35 Transmit History Data Buffer Numbers

| Buffer No. | Buffer type | | |
|------------|------------------------------------|---|---|
| | B'001 | B'010 | B'100 |
| B'0000 | Transmit buffer $16 \times m + 0$ | Buffer numbers of the transmit buffer linked to the transmit/receive FIFO buffer by the CFTML[3:0] bits in the RSCFDn(CFD)CFCCk register (k = 0 to 5) | Buffer numbers of the transmit buffer allocated to the transmit queue that performed transmission |
| B'0001 | Transmit buffer $16 \times m + 1$ | | |
| B'0010 | Transmit buffer $16 \times m + 2$ | | |
| B'0011 | Transmit buffer $16 \times m + 3$ | | |
| B'0100 | Transmit buffer $16 \times m + 4$ | | |
| B'0101 | Transmit buffer $16 \times m + 5$ | | |
| B'0110 | Transmit buffer $16 \times m + 6$ | | |
| B'0111 | Transmit buffer $16 \times m + 7$ | | |
| B'1000 | Transmit buffer $16 \times m + 8$ | | |
| B'1001 | Transmit buffer $16 \times m + 9$ | | |
| B'1010 | Transmit buffer $16 \times m + 10$ | | |
| B'1011 | Transmit buffer $16 \times m + 11$ | | |
| B'1100 | Transmit buffer $16 \times m + 12$ | | |
| B'1101 | Transmit buffer $16 \times m + 13$ | | |
| B'1110 | Transmit buffer $16 \times m + 14$ | | |
| B'1111 | Transmit buffer $16 \times m + 15$ | | |

Label data is used to identify each message. Unique label data can be added to each message transmitted from a transmit buffer, a transmit queue, or a transmit/receive FIFO buffer.

A timestamp value is obtained from the timestamp counter at the timing of the start-of-frame (SOF) of a message. For details on the timestamp counter, refer to section 48.12.1.6, Timestamp.

Transmit history data can be read from the RSCFDn(CFD)THLACCm register. If an attempt is made to store new transmit history data while the buffer is full, the buffer overflows and the new data is discarded.

48.14 Gateway Function

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

When a transmit/receive FIFO buffer is set to gateway mode, receive messages can be transmitted from an arbitrary channel without CPU intervention.

When the CFM [1:0] bits in the RSCFDn(CFD)CFCCk register are set to B'10 (gateway mode) for the transmit/receive FIFO buffer selected by the RSCFDn(CFD)GAFLP1_j register of a channel being used for transmission, messages that pass through filter processing according to the reception rule are stored in the specified transmit/receive FIFO buffer and are automatically transmitted from the buffer.

Messages stored in a transmit/receive FIFO buffer are transmitted sequentially on a first-in, first-out basis. Only the message to be transmitted next becomes the target of transmit priority determination.

Transmit/receive FIFO buffers in the gateway mode are disabled by setting the CFE bit in the RSCFDn(CFD)CFCCk register to 0 and the CFEMP flag becomes 1 according to the timing below.

- The transmit/receive FIFO buffer becomes empty immediately when the message in it is not being transmitted and will not be transmitted next.
- The transmit/receive FIFO buffer becomes empty after transmission completion, CAN bus error detection, or arbitration-lost when the message in it is being transmitted or will be transmitted next.

When the CFE bit is cleared to 0, all messages in transmit/receive FIFO buffers are lost and messages can no longer be stored in transmit/receive FIFO buffers. Confirm that the CFEMP flag is set to 1 before setting the CFE bit to 1 again.

48.14.1 CAN-CAN FD Gateway (CAN FD Mode Only)

When the gateway function is used in CAN FD mode, the transmit frame can be replaced with the classical CAN frame or CAN FD frame.

When the GWEN bit in the RSCFDnCFDCmFDCFG register is set to 1, the CAN-CAN FD gateway is enabled. The GWDF and GWBRS bits in the RSCFDnCFDCmFDCFG register can be used to select the FDF and BRS bits of the transmit frame. When the DLC value of the received classical CAN frame is equal to or larger than B'1001 and the GWDF bit is 1 (CAN FD frame), the DLC value is replaced with B'1000.

When the CAN-CAN FD gateway is enabled, do not route the following frames.

- CAN FD frame of which payload length is longer than 8 bytes
- Remote frame

48.15 Test Function

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Test functions are classified into communication tests and global tests.

- Communication tests: Performed for each channel.
 - Standard test mode
 - Listen-only mode
 - Self-test mode 0 (external loopback mode)
 - Self-test mode 1 (internal loopback mode)
 - Limited operation mode (CAN FD mode only)
- Global tests: Performed for the entire module
 - RAM test (read/write test)
 - Inter-channel communication test [CRC error test is enabled.]

48.15.1 Standard Test Mode

Standard test mode allows CRC test. The CRC value, which has been calculated by the RS-CANFD module according to the transmit or receive message, is stored in the register; the CRCREG[14:0] bits in the RSCFDn(CFD)CmERFL register when the message is in the classical CAN frame (15-bit CRC length) and the CRCREG[20:0] bits in the RSCFDnCFDCmFDCRC register when the message is in the CAN FD frame (17-bit or 21-bit CRC length). Use the inter-channel communication test function for the CRC error test. For details, refer to section 48.15.6.1, CRC Error Test.

48.15.2 Listen-Only Mode

Listen-only mode allows reception of data frames and remote frames. Only recessive bits are transmitted on the CAN bus, and the ACK bit, overload flag, and active error flag are not transmitted. Listen-only mode is available for detecting the communication speed. Do not make a transmit request from any buffer or queue in listen-only mode.

Figure 48.12 shows the connection when listen-only mode is selected.

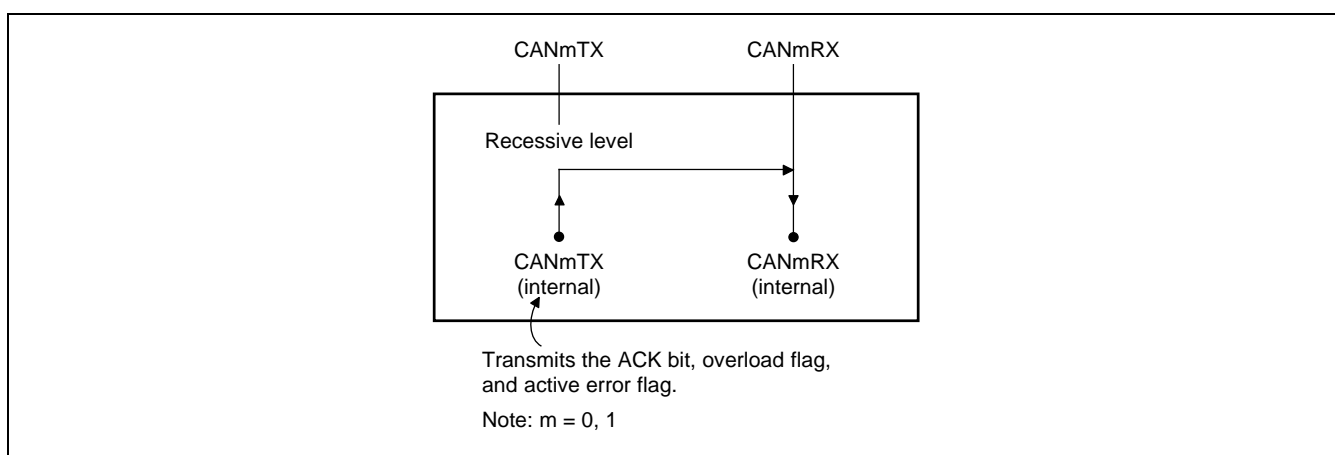


Figure 48.12 Connection when Listen-Only Mode is Selected

48.15.3 Self-Test Mode (Loopback Mode)

In self-test mode, transmitted messages are compared with the receive rule of the own channel and the messages are stored in a buffer if they have passed through the filter processing. Messages transmitted from other CAN nodes are compared only with the receive rule for which the GAFLLB bit in the RSCFDn(CFD)GAFLIDj register ($j = 0$ to 15) is set to 0 (when a message transmitted from another CAN node is received).

If the mirror function and self-test mode are both enabled, the self-test mode setting takes precedence.

48.15.3.1 Self-Test Mode 0 (External Loopback Mode)

Self-test mode 0 is used to perform a loopback test within a channel including the CAN transceiver.

In self-test mode 0, transmitted messages are handled as messages received through the CAN transceiver and are stored in a buffer. An ACK bit is generated to receive messages transmitted from the own CAN node.

Figure 48.13 shows the connection when self-test mode 0 is selected.

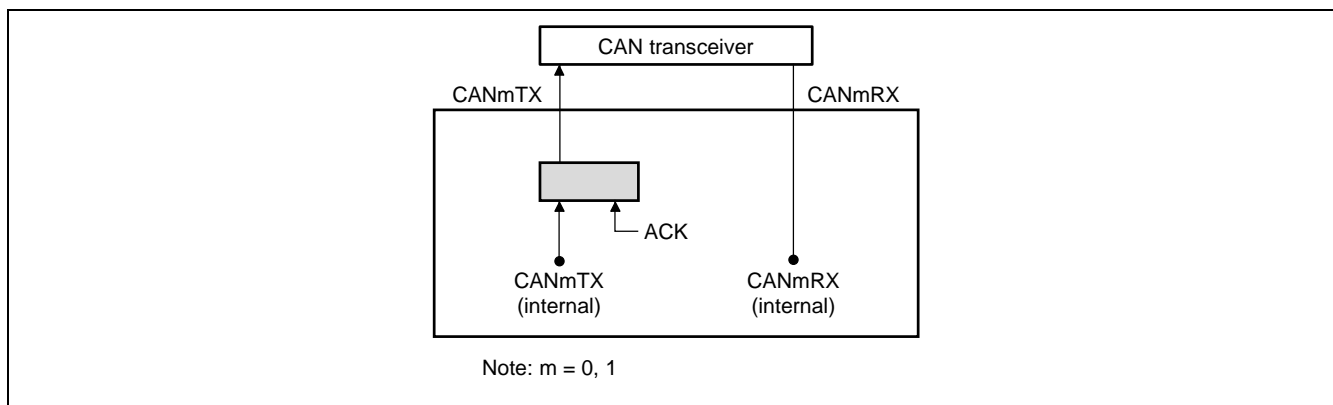


Figure 48.13 Connection when Self-Test Mode 0 is Selected

48.15.3.2 Self-Test Mode 1 (Internal Loopback Mode)

In self-test mode 1, transmitted messages are handled as received messages and are stored in a buffer. An ACK bit is generated to receive messages transmitted from the own CAN node.

In self-test mode 1, internal feedback from the internal CANmTX pin ($m = 0$ or 1) to the internal CANmRX pin is performed. The external CANmRX pin input is isolated. The external CANmTX pin outputs only recessive bits.

Figure 48.14 shows the connection when self-test mode 1 is selected.

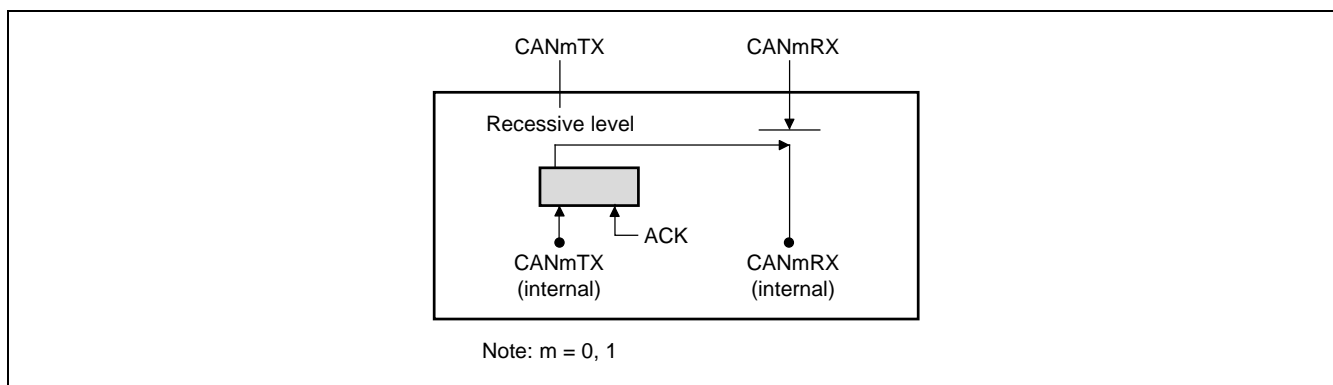


Figure 48.14 Connection when Self-Test Mode 1 is Selected

48.15.4 Limited Operation Mode (CAN FD Mode Only)

When the enabled data and remote frames are received in limited operation mode, an ACK bit is generated. However, these frames are not transmitted even if the error frame or overload frame transmit condition is detected. When the condition is detected, CANFD waits for the bus idling state to resynchronize CAN communications. The receive error counter (REC) and the transmit error counter (TEC) are not changed if an error occurs.

For transmission, any transmit request can be issued with no limitations.

48.15.5 RAM Test

The RAM test function allows accesses to all CAN RAM addresses.

When the RAM test function is used, the RAM is divided into pages of 256 bytes each. RAM test page is set by the RTMPS[6:0] bits in the RSCFDn(CFD)GTSTCFG register. Data in the set page can be read from and written to the RSCFDn(CFD)RPGACCr register (r = 0 to 63). The available total RAM size is 15360 bytes (H'3C00) in classical CAN mode and 21312 bytes (H'53D0) in CAN FD mode.

48.15.6 Inter-Channel Communication Test

The inter-channel communication test function allows communication test by internally connecting CAN channels to each other. During this test, channels are isolated from the external CAN bus.

Before starting data transmission/reception in channel communication mode, make transmission/reception settings for each channel.

Figure 48.15 shows the connection for inter-channel communication test.

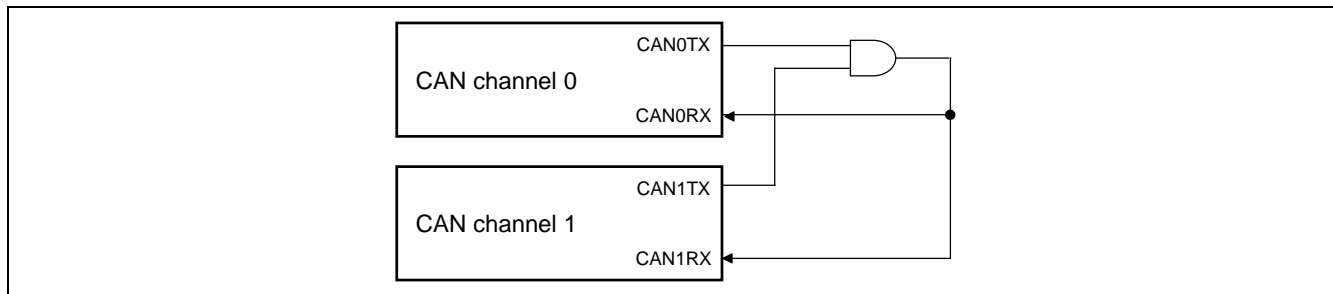


Figure 48.15 Connection for Inter-Channel Communication Test

48.15.6.1 CRC Error Test

During inter-channel communication test, a CRC error test can be performed. The following shows an example of procedure when a CRC error test on channel 0 is performed during communication test between channels 0 and 1.

Conditions:

- Inter-channel communication test is enabled.
- Channels 0 and 1 are in standard test mode.

Procedure:

1. Make a setting that a message is transmitted from transmit buffer p in channel 1.
2. Set 1 to the CRCT bit in the RSCFDn(CFD)C0CTR register (inversion of the start bit in the receive ID field is enabled).
3. Set 1 to the TMTR bit in the RSCFDn(CFD)TMCp register (a transmission request is issued to transmit buffer p in channel 1).
4. Wait for generation of the CAN0 error interrupt due to a channel bus error.
5. Read the CRCREG[15:0] bits in the RSCFDn(CFD)CmERFL register or the CRCREG[20:0] bits in the RSCFDnCFDCmFDCRC register on both channels and check that the CRC values are different in the transmit and receive sides.
6. Check that the CERR bit in the RSCFDm(CFD)C0ERFL register is 1 (a CRC error is detected).

Using the CRC error test function generates an illegal CRC value by inverting the start bit in the receive ID field. Therefore, if the higher-order five bits of the ID are B'1_0000 or the higher-order six bits are B'01_1111 in the received message, note that a stuff error (consecutive 6-bit data with the same level) will be detected.

Since the CRC generation circuit in the RS-CANFD module is incorporated in each protocol controller of channels and the same circuit is shared in transmission and reception, a CRC calculation test needs not be performed at transmission.

48.16 RS-CANFD Setting Procedure

RZ/G2H

RZ/G2M V1.3

RZ/G2M V3.0

48.16.1 Initial Settings

RZ/G2N

RZ/G2E

The RS-CAN module initializes the CAN RAM after the MCU is reset. The RAM initialization time is 7586 cycles of the pclk. The GRAMINIT flag in the RSCFDn(CFD)GSTS register is set to 1 (CAN RAM initialization is ongoing) during the RAM initialization and is cleared to 0 (CAN RAM initialization is finished) when the initialization is completed. Make CAN settings after the GRAMINIT flag is cleared to 0. Figure 48.16 shows the CAN setting procedure after the MCU is reset.

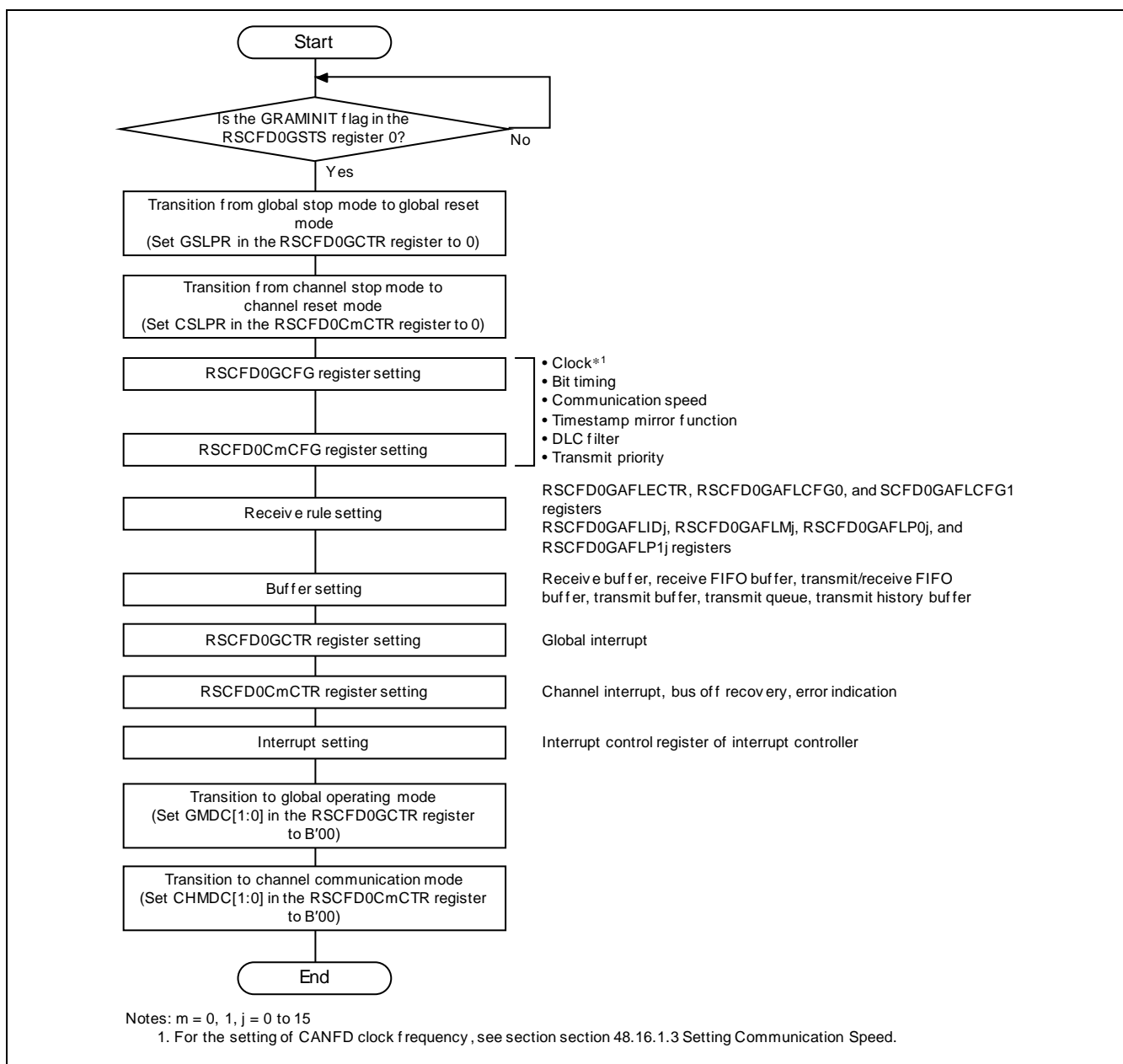


Figure 48.16 CAN Setting Procedure after the MCU is Reset

48.16.1.1 Setting a Clock

Set the CAN clock (fCAN) as a clock source of the RS-CANFD module. Select the clk_xincan or clk_c using the DCS bit in the RSCFDn(CFD)GCFG register.

48.16.1.2 Setting Bit Timing

In the CAN protocol, one bit of a communication frame consists of three segments SS, TSEG1, and TSEG2, of which two segments TSEG1 and TSEG2 can be set by the register for each channel. These segments are set by the RSCFDnCmCFG register in classical CAN mode. In CAN FD mode, these segments have two types of bit rates (normal and data bit rates) and each bit rate is set by the RSCFDnCFDCmNCFG and RSCFDnCFDCmDCFG registers. Sample point timing can be determined by setting these two segments. This timing can be adjusted in units of 1 Time Quantum (hereafter referred to as Tq). 1 Tq is equal to one CANmTq clock cycle. A 1Tq is a clock cycle which divides the clock selected with the DCS bit in the RSCFDnGCFG register. The division ratio is set by the BRP[9:0] bits in the RSCFDnCmCFG register (CANmTq clock cycle) in classical CAN mode and set by the NBRP[9:0] bits in the RSCFDnCFDCmNCFG register and the DBRP[7:0] bits in the RSCFDnCFDCmDCFG register (CANmTq(N) and CANmTq(D) clock cycles) in CAN FD mode. ISO 11898-1 allows the normal and data bit rates individually have each division ratio. However, if different values are set to each division ratio, resynchronization between CAN nodes may be lost at the timing when the normal bit rate is switched to the data bit rate. Therefore, it is recommended that the NBRP[9:0] and DBRP[7:0] bits are set to the same value and two bit rates are set to the different values depending on the segment value.

Figure 48.17 shows the bit timing chart. Table 48.36 shows an example of bit timing setting.

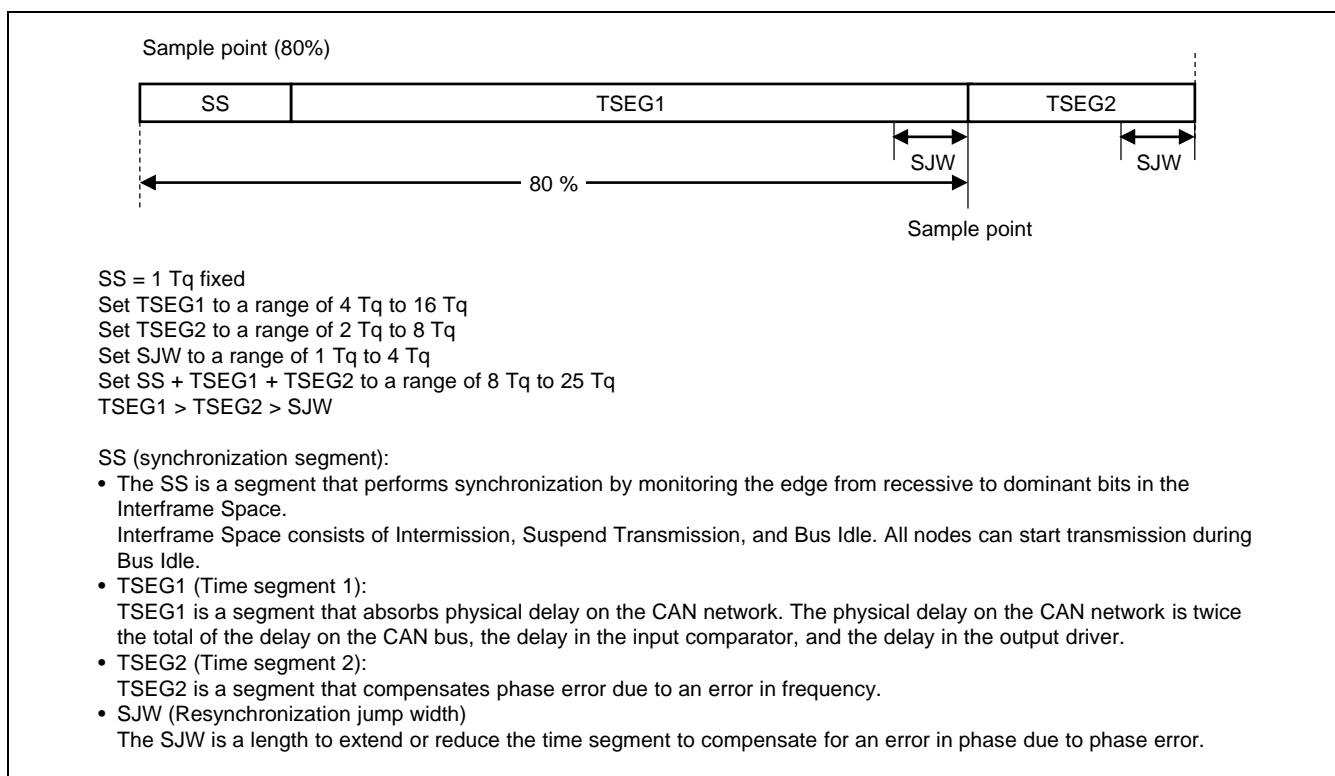


Figure 48.17 Bit Timing Chart

Table 48.36 Example of Bit Timing Setting

| 1 Bit | Set Value (Tq) | | | | Sample Point (%) |
|--------|----------------|-------|-------|-----|-------------------------|
| | SS | TSEG1 | TSEG2 | SJW | Note: See Figure 48.17. |
| 5 Tq* | 1 | 2 | 2 | 1 | 60.00 |
| 8 Tq | 1 | 4 | 3 | 1 | 62.50 |
| | 1 | 5 | 2 | 1 | 75.00 |
| 10 Tq | 1 | 6 | 3 | 1 | 70.00 |
| | 1 | 7 | 2 | 1 | 80.00 |
| 16 Tq | 1 | 10 | 5 | 1 | 68.75 |
| | 1 | 11 | 4 | 1 | 75.00 |
| 20 Tq | 1 | 12 | 7 | 1 | 65.00 |
| | 1 | 13 | 6 | 1 | 70.00 |
| 50 Tq* | 1 | 39 | 10 | 4 | 80.00 |

Note: * CAN FD mode only.

48.16.1.3 Setting Communication Speed

Set the CAN communication speed for each channel using the fCAN, baud rate prescaler division value, and Tq count per bit time. In CAN FD mode, set communication speed for each channel in arbitration and data phases.

Figure 48.18 shows the CAN clock control block diagram, and Table 48.37 shows an example of the communication speed setting.

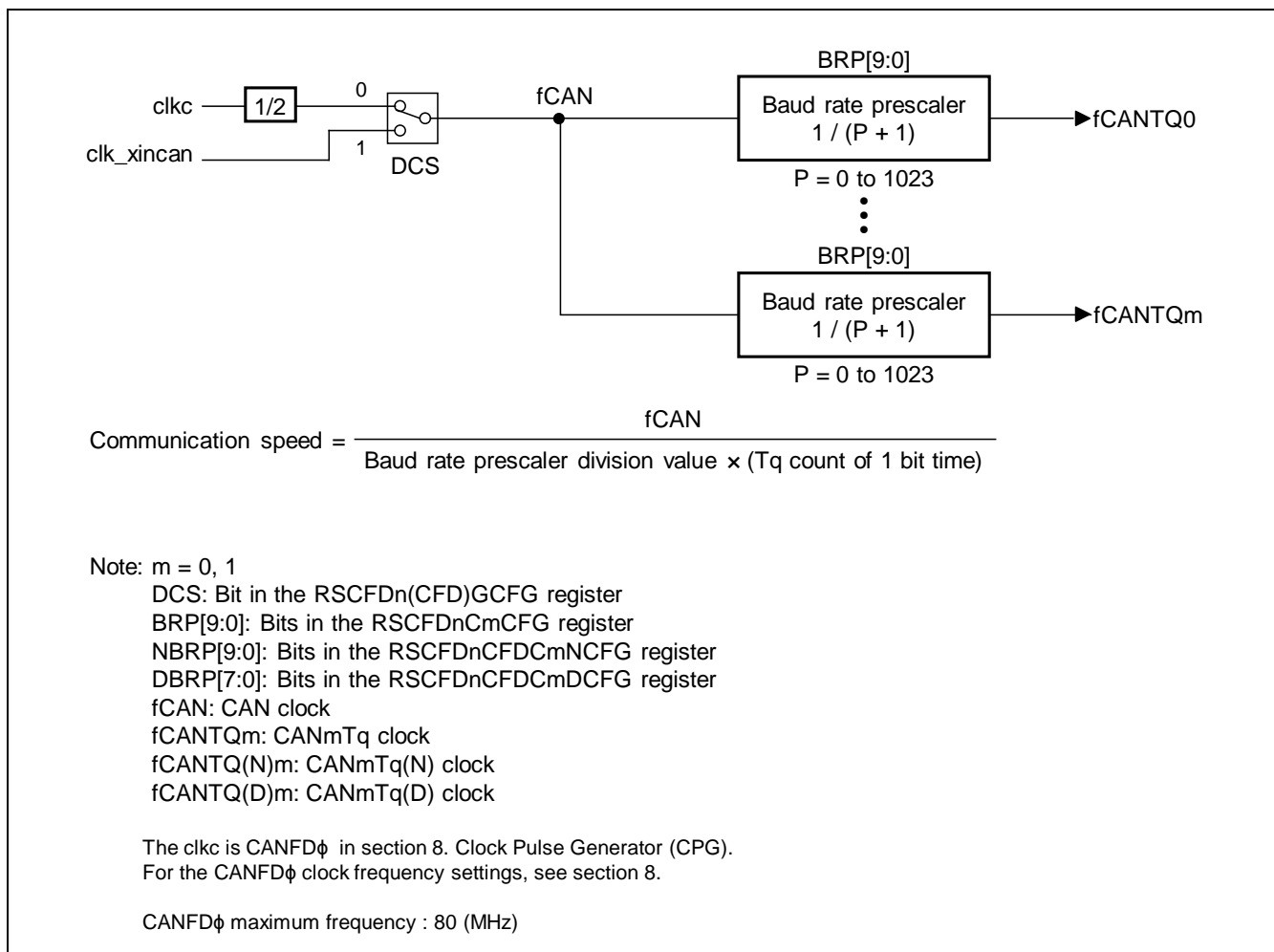


Figure 48.18 CAN Clock Control Block Diagram

Table 48.37 Example of Communication Speed Setting (Classical CAN Mode)

| Communication speed | fCAN | | | | | |
|---------------------|------------|------------|------------|------------|-----------|-----------|
| | 40 MHz | 32 MHz | 24 MHz | 20 MHz | 16 MHz | 8 MHz |
| 1 Mbps | 8 Tq (5) | 8 Tq (4) | 8 Tq (3) | 10 Tq (2) | 8 Tq (2) | 8 Tq (1) |
| | 20 Tq (2) | 16 Tq (2) | 12 Tq (2) | 20 Tq (1) | 16 Tq (1) | |
| | | | 24 Tq (1) | | | |
| 500 Kbps | 8 Tq (10) | 8 Tq (8) | 8 Tq (6) | 10 Tq (4) | 8 Tq (4) | 8 Tq (2) |
| | 20 Tq (4) | 16 Tq (4) | 12 Tq (4) | 20 Tq (2) | 16 Tq (2) | 16 Tq (1) |
| | | | 24 Tq (2) | | | |
| 250 Kbps | 8 Tq (20) | 8 Tq (16) | 8 Tq (12) | 10 Tq (8) | 8 Tq (8) | 8 Tq (4) |
| | 20 Tq (8) | 16 Tq (8) | 12 Tq (8) | 20 Tq (4) | 16 Tq (4) | 16 Tq (2) |
| | | | 24 Tq (4) | | | |
| 125 Kbps | 8 Tq (40) | 8 Tq (32) | 8 Tq (24) | 10 Tq (16) | 8 Tq (16) | 8 Tq (8) |
| | 20 Tq (16) | 16 Tq (16) | 12 Tq (16) | 20 Tq (8) | 16 Tq (8) | 16 Tq (4) |
| | | | 24 Tq (8) | | | |

Table 48.38 Example of Communication Speed Setting (CAN FD Mode, Normal Bit Rate and Data Bit Rate)

| Communication speed | fCAN | |
|-----------------------|-------------------|-------------------|
| | 40 MHz | 20 MHz |
| Normal: 1-Mbps data | Normal: 40 Tq (1) | None |
| 5 Mbps | Data: 8 Tq (1) | |
| Normal: 500-Kbps data | Normal: 80 Tq (1) | Normal: 40 Tq (1) |
| 2 Mbps | Data: 20 Tq (1) | Data: 10 Tq (1) |

Note: Values in () are baud rate prescaler division values.

The clk_c is CANFD ϕ in section 11. Clock Pulse Generator (CPG).

For the CANFD ϕ clock frequency settings, see section 11.

CANFD ϕ maximum frequency are as follows.

RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E: 80 (MHz)

48.16.1.4 Setting Receive Rules

Receive rules can be set using registers related to receive rules.

Up to 16 receive rules can be registered per page. Specify pages 0 to 23 (when a unit incorporates six channels) by the AFLPN[4:0] bits in the RSCFDn(CFD)GAFLECTR register. Use the AFLDAE bit to set enabling or disabling writing to the receive rule table.

Figure 48.19 shows the receive rule setting procedure.

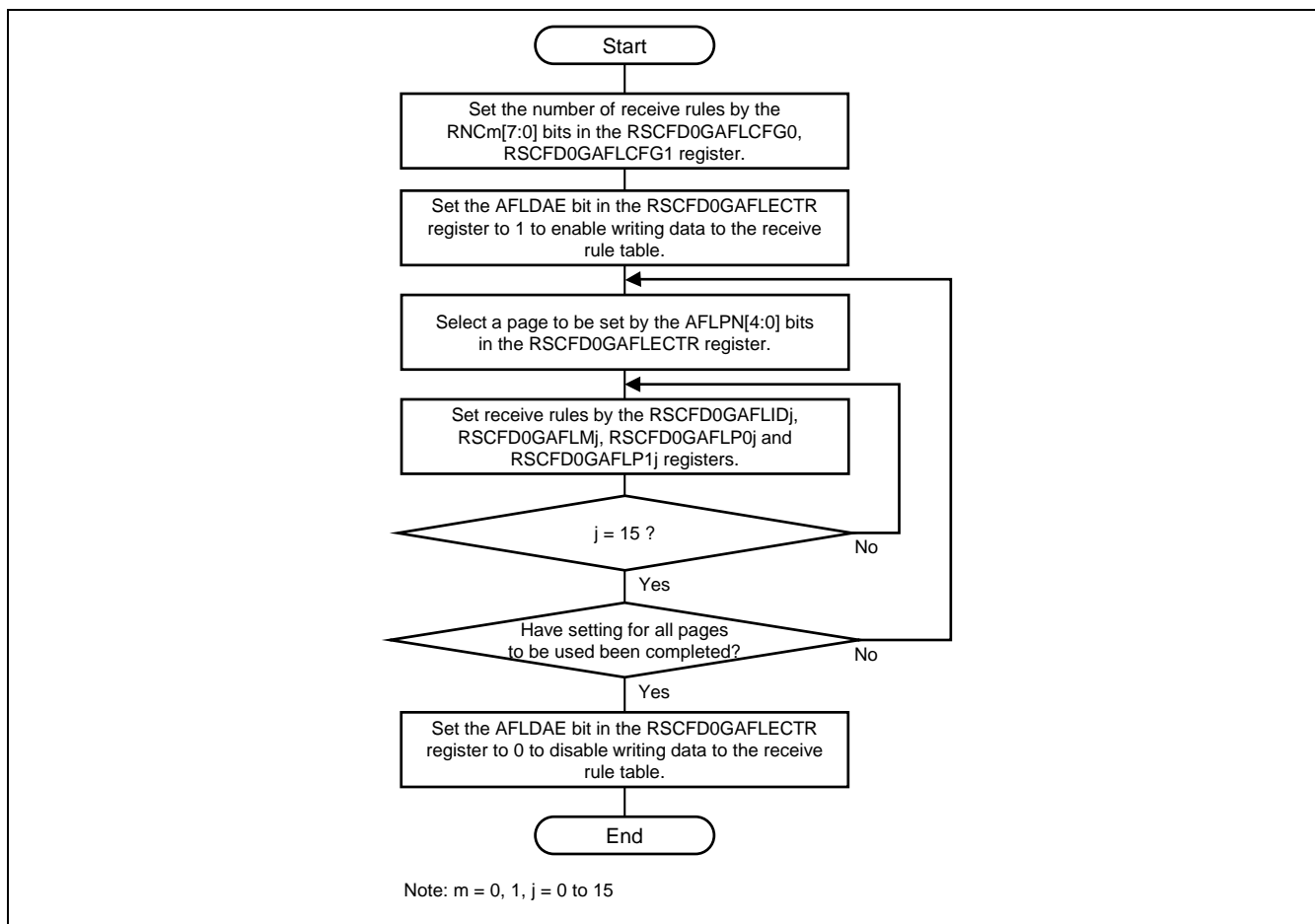


Figure 48.19 Receive Rule Setting Procedure

48.16.1.5 Setting Buffers

Set the number of buffers (messages to be stored) to be used and interrupt sources of buffers. In CAN FD mode, the size for storing the payload is also set. For transmit/receive FIFO buffers that are set to transmit mode, set transmit buffers to be linked.

In classical CAN mode, receive and FIFO buffers can use a maximum of 2048-byte RAM. Up to 128 buffers can be used and 16 bytes are used per buffer. The following conditions must be satisfied.

The number of receive buffers

+ The total of depth of receive FIFO buffer x

+ The total of depth of transmit/receive FIFO buffer k \leq 128 buffers

In CAN FD mode, receive and FIFO buffers can use a maximum of 10752-byte RAM. The following conditions must be satisfied.

The number of receive buffers \times (12 + the size for storing the payload)

+ The total of (depth \times (12 + the size for storing the payload) of receive FIFO buffer x

+ The total of (depth \times (12 + the size for storing the payload) of transmit/receive FIFO buffer k

\leq 3584 bytes

Figure 48.20 shows the buffer configuration. Figure 48.21 shows the buffer setting procedure.

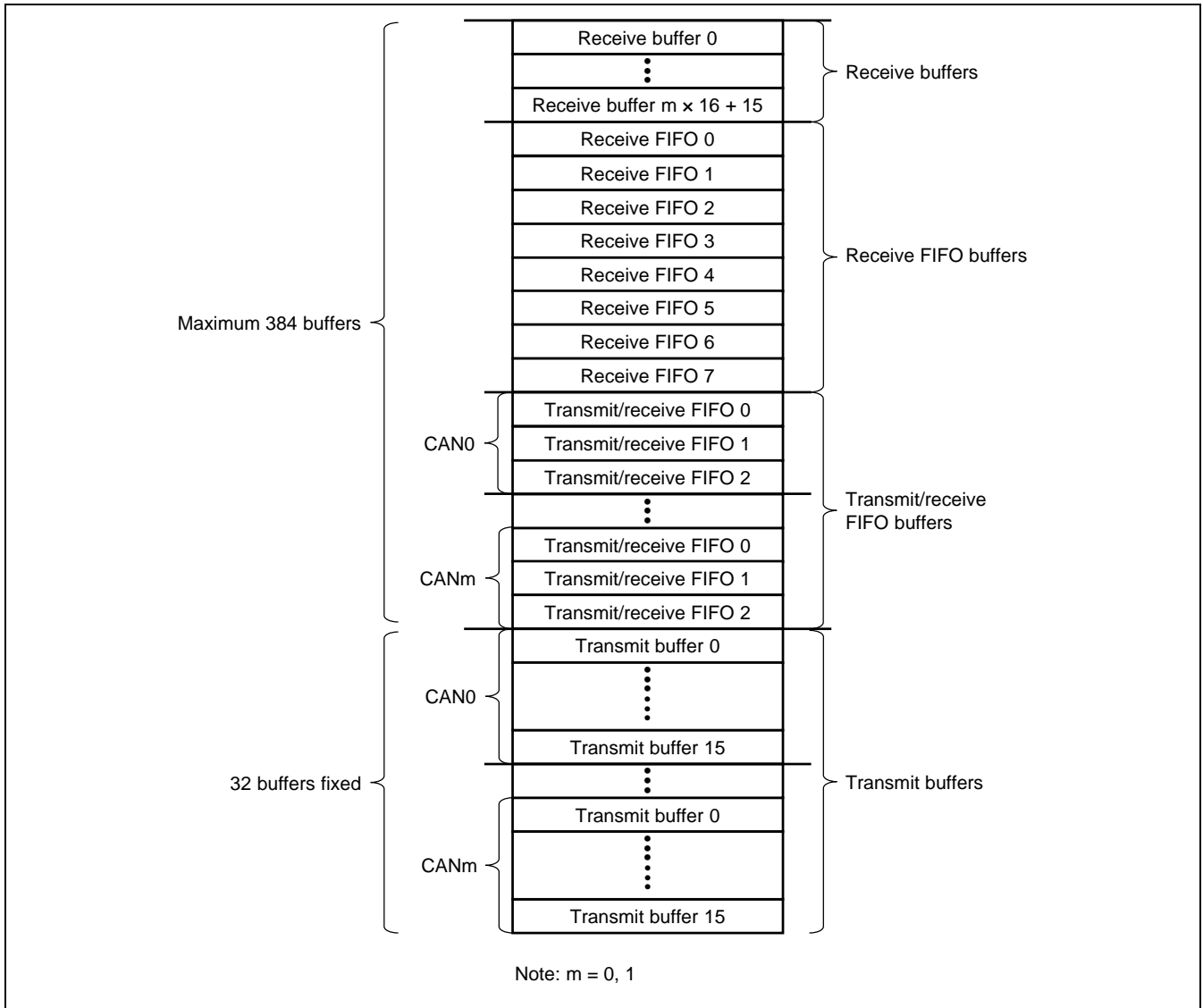


Figure 48.20 Buffer Configuration

Note: Receive buffers, receive FIFO buffers, transmit/receive FIFO buffers, and transmit buffers are located in succession.

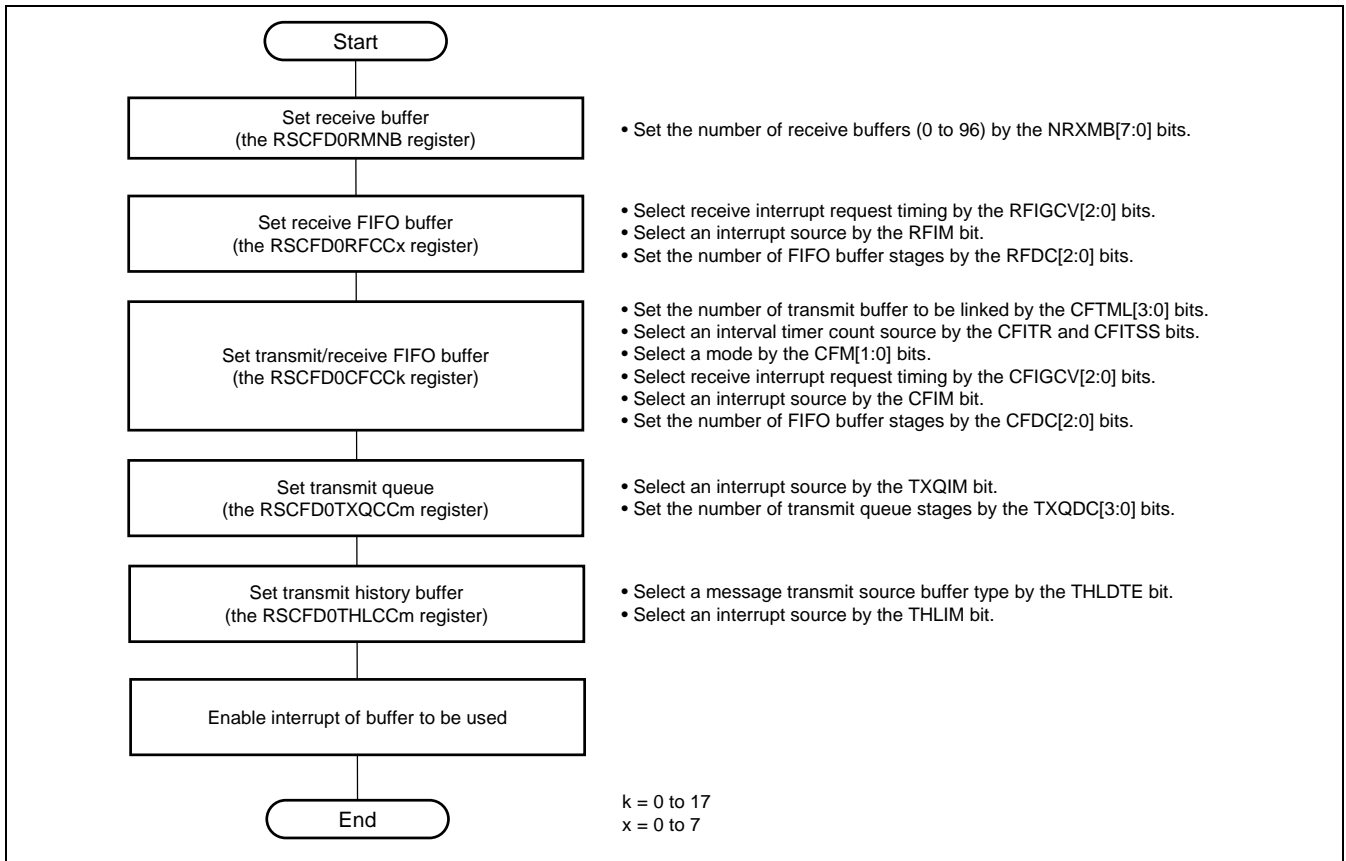


Figure 48.21 Buffer Setting Procedure

48.16.1.6 Transmit Delay Correction (CAN FD Mode Only)

A high baud rate is used for the data phase in CAN FD mode. At this time, the transmit delay correction function is used to enable transmit delay.

When this function is used, set the TDCE bit in the RSCFDnCFDCmFDCFG register to 1. The timing of the secondary sample point (SSP) used for a data phase must be set by the TDCOC and TDCO[6:0] bits in the RSCFDnCFDCmFDCFG register.

When the TDCOC bit is 0, the SSP timing is equal to the sum of the values of the delay measured by the RS-CANFD module and the TDCO[6:0] bits (it is rounded down to T_q which is the approximate integer). The value of the TDCO[6:0] bits must always be the sum of the sample point timing SS and TSEG1.

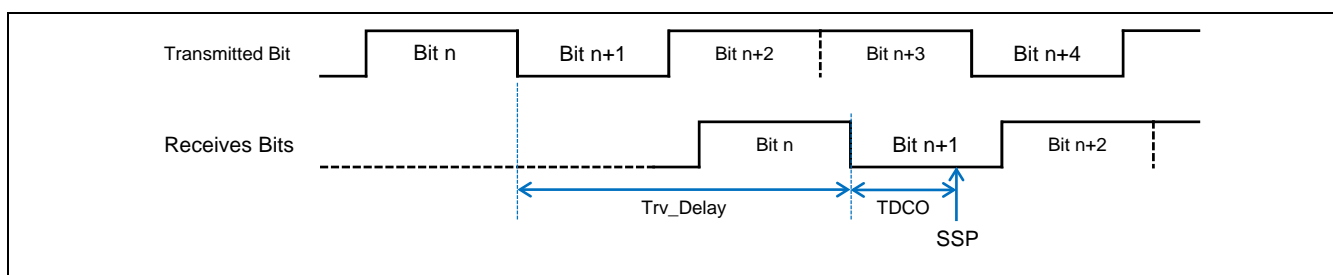


Figure 48.22 SSP Timing

When the TDCOC bit is 1, the SSP timing is only determined by the value of the TDCO[6:0] bits (if the DBRP[7:0] bits in the RSCFDnCFDCmDCFG register are larger than 0, the value of the TDCO[6:0] bits are also rounded down to T_q which is the approximate integer).

The RS-CANFD module corrects the delay of a maximum of $3CAN_m$ bit time - $2T_q$ (the CAN_m bit time and T_q are the values of data bit rate).

48.16.2 Reception Procedure

48.16.2.1 Receive Buffer Reading Procedure

When the processing to store received messages in a receive buffer starts, the RMNSq flag in the RSCFDn(CFD)RMNDy register (y = 0 to 2, q = 0 to 31) is set to 1 (receive buffer q contains a new message). Messages can be read from the RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq (CAN FD mode only), and RSCFDn(CFD)RMDFb_q (b = 0 or 1 in classical CAN mode, and b = 0 to 4 in CAN FD mode) registers. If the next message has been received before the current message is read from the receive buffer, the message is overwritten. Figure 48.23 shows the receive buffer reading procedure. Using this procedure maintains consistency of the message read from the RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq, and RSCFDn(CFD)RMDFb_q registers.

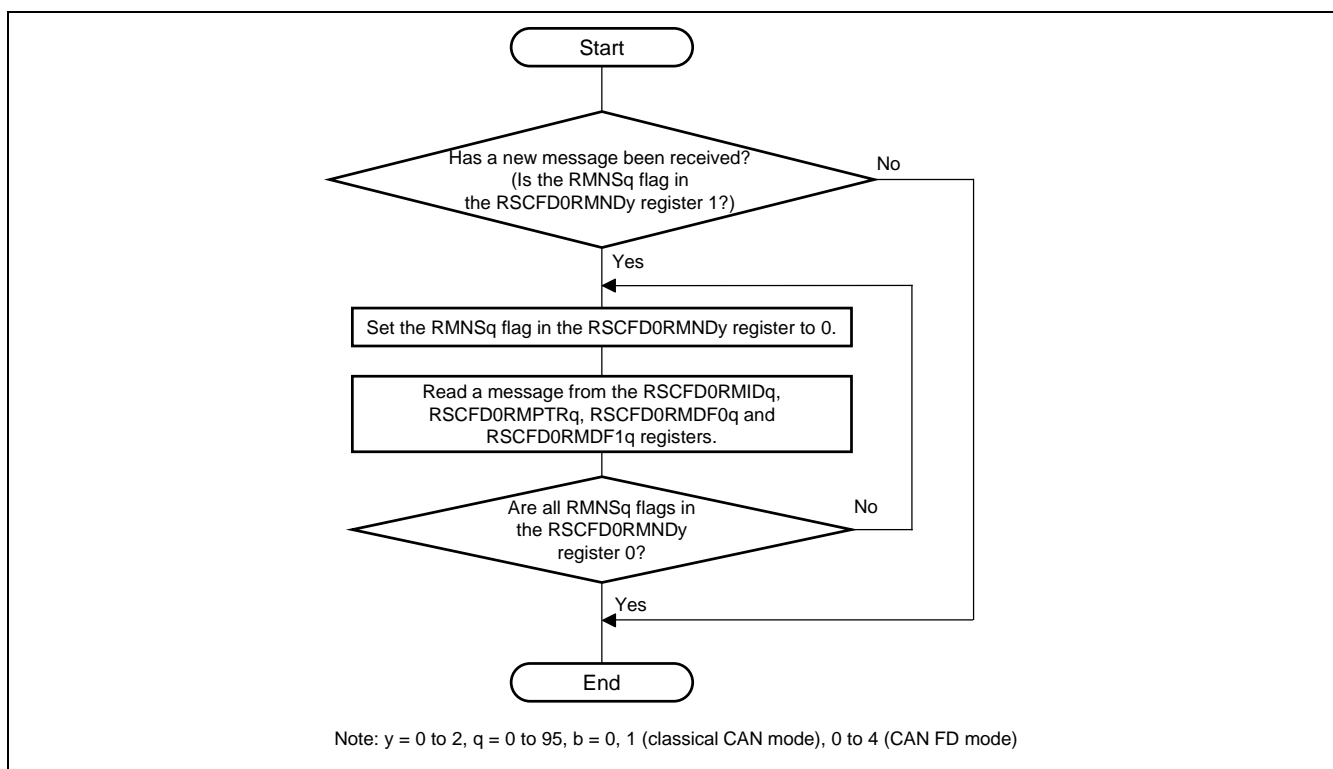


Figure 48.23 Receive Buffer Reading Procedure

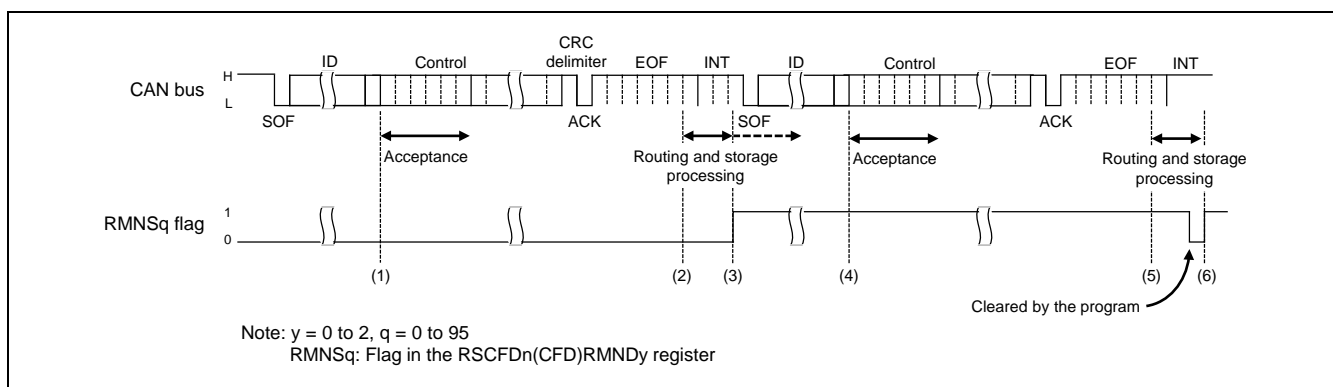


Figure 48.24 Receive Buffer Reception Timing Chart

- (1) When the ID field in a message has been received, the acceptance filter processing starts.
- (2) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.
- (3) When the message has passed through the DLC filter processing, the processing to store the message in the specified receive buffer starts. When the message storage processing starts, the RMNSq flag in the corresponding RSCFDn(CFD)RMNDy register is set to 1 (the receive buffer contains a new message). If other channels are performing filter processing or transmit priority determination processing, the routing processing and the storage processing may be delayed.
- (4) When the ID field of the next message has been received, the acceptance filter processing starts.
- (5) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.
- (6) When the corresponding RMNSq flag is cleared to 0 (the receive buffer contains no new message), this flag is set to 1 again when the message storage processing starts. Even if the RMNSq flag remains 1, a new message is overwritten to the receive buffer. The RMNSq flag should not be cleared to 0 during storage of messages.

48.16.2.2 FIFO Buffer Reading Procedure

When received messages have been stored in one or more receive FIFO buffers or a transmit/receive FIFO buffer that is set to receive mode or gateway mode, the corresponding message count display counter (RFMC[7:0] bits in the RSCFDn(CFD)RFSTSx register (x = 0 to 7) or CFMC[7:0] bits in the RSCFDn(CFD)CFSTS_k register (k = 0 to 5)) is incremented. At this time, when the RFIE bit (receive FIFO interrupt is enabled) in the RSCFDn(CFD)RFCCx register or the CFRXIE bit (transmit/receive FIFO receive interrupt is enabled) in the RSCFDn(CFD)CFCC_k register is set to 1, an interrupt request is generated. Received messages can be read from the RSCFDn(CFD)RFIDx, RSCFDn(CFD)RFPTRx, RSCFDn(CFD)RFFDSTSx (CAN FD mode only), and RSCFDn(CFD)RFDFd_x (d = 0 or 1 in classical CAN mode, and d = 0 to 15 in CAN FD mode) registers for receive FIFO buffers, or from the RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, RSCFDn(CFD)CFDCFFDCSTS_k (CAN FD mode only), and RSCFDn(CFD)CFDFd_k registers for transmit/receive FIFO buffers. Messages in FIFO buffers can be read sequentially on a first-in, first-out basis.

When the message count display counter value matches the FIFO buffer depth (a value set by the RFDC[2:0] bits in the RSCFDn(CFD)RFCCx register or the CFDC[2:0] bits in the RSCFDn(CFD)CFCC_k register), the RFFLL or CFFLL flag is set to 1 (the receive FIFO buffer is full).

When all messages have been read out of the FIFO buffer, the RFEMP flag in the RSCFDn(CFD)RFSTSx register or the CFEMP flag in the RSCFDn(CFD)CFSTS_k register is set to 1 (the receive FIFO buffer contains no unread message (buffer empty)).

If the RFE bit or the CFE bit is cleared to 0 (no receive FIFO buffer is used) with the interrupt request flag (RFIF flag in the RSCFDn(CFD)RFSTSx register or CFRXIF flag in the RSCFDn(CFD)CFSTS_k register) set to 1 (a receive FIFO interrupt request is present), the interrupt request flag is not automatically cleared to 0. The program must clear the interrupt request flag to 0.

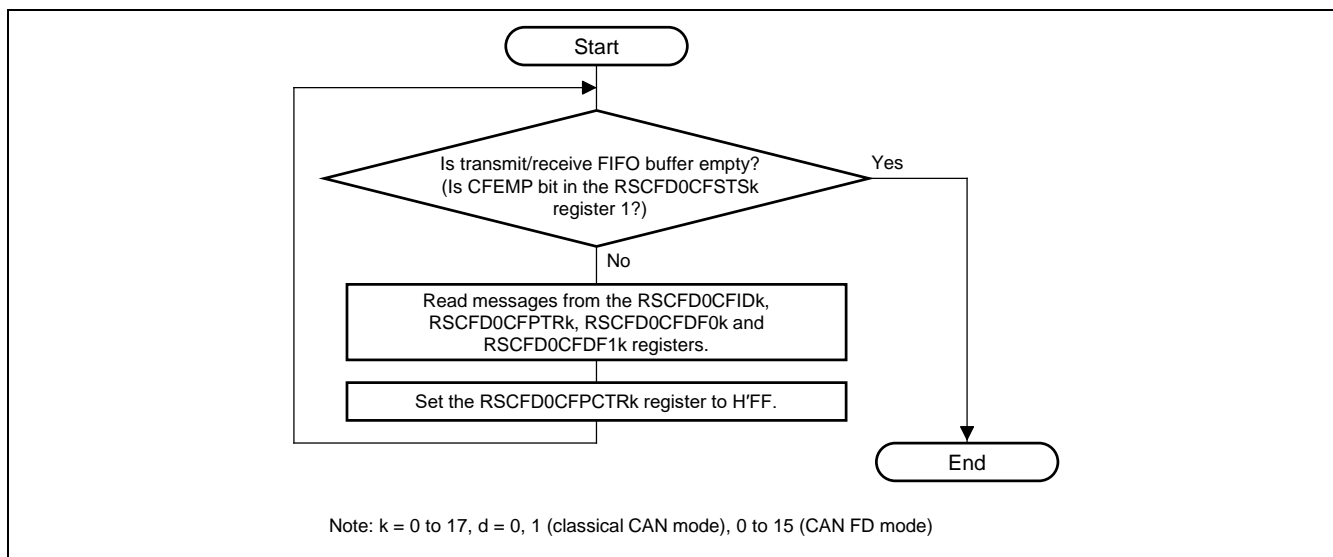


Figure 48.25 Transmit/Receive FIFO Buffer Reading Procedure

When a message is read in CAN FD mode, do not read the RSCFDnCFDRFDFd_x or RSCFDnCFDCFDFd_k register corresponding to the area which exceeds the payload storage size set by the RFPLS[2:0] bits in the RSCFDnCFDRFCCx register or the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register.

Table 48.39 Payload Storage Area in the Receive FIFO Buffer

| Value Set by the RFPLS[2:0] Bits | Payload Storage Size | Corresponding Data Field Register |
|-------------------------------------|----------------------|---------------------------------------|
| B'000 | 8 bytes | RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF1_x |
| B'001 | 12 bytes | RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF2_x |
| B'010 | 16 bytes | RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF3_x |
| B'011 | 20 bytes | RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF4_x |
| B'100 | 24 bytes | RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF5_x |
| B'101 | 32 bytes | RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF7_x |
| B'110 | 48 bytes | RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF11_x |
| B'111 | 64 bytes | RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF15_x |

Table 48.40 Payload Storage Area in the Transmit/Receive FIFO Buffer

| Value Set by the CFPLS[2:0] Bits | Payload Storage Size | Corresponding Data Field Register |
|-------------------------------------|----------------------|---------------------------------------|
| B'000 | 8 bytes | RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF1_k |
| B'001 | 12 bytes | RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF2_k |
| B'010 | 16 bytes | RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF3_k |
| B'011 | 20 bytes | RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF4_k |
| B'100 | 24 bytes | RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF5_k |
| B'101 | 32 bytes | RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF7_k |
| B'110 | 48 bytes | RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF11_k |
| B'111 | 64 bytes | RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF15_k |

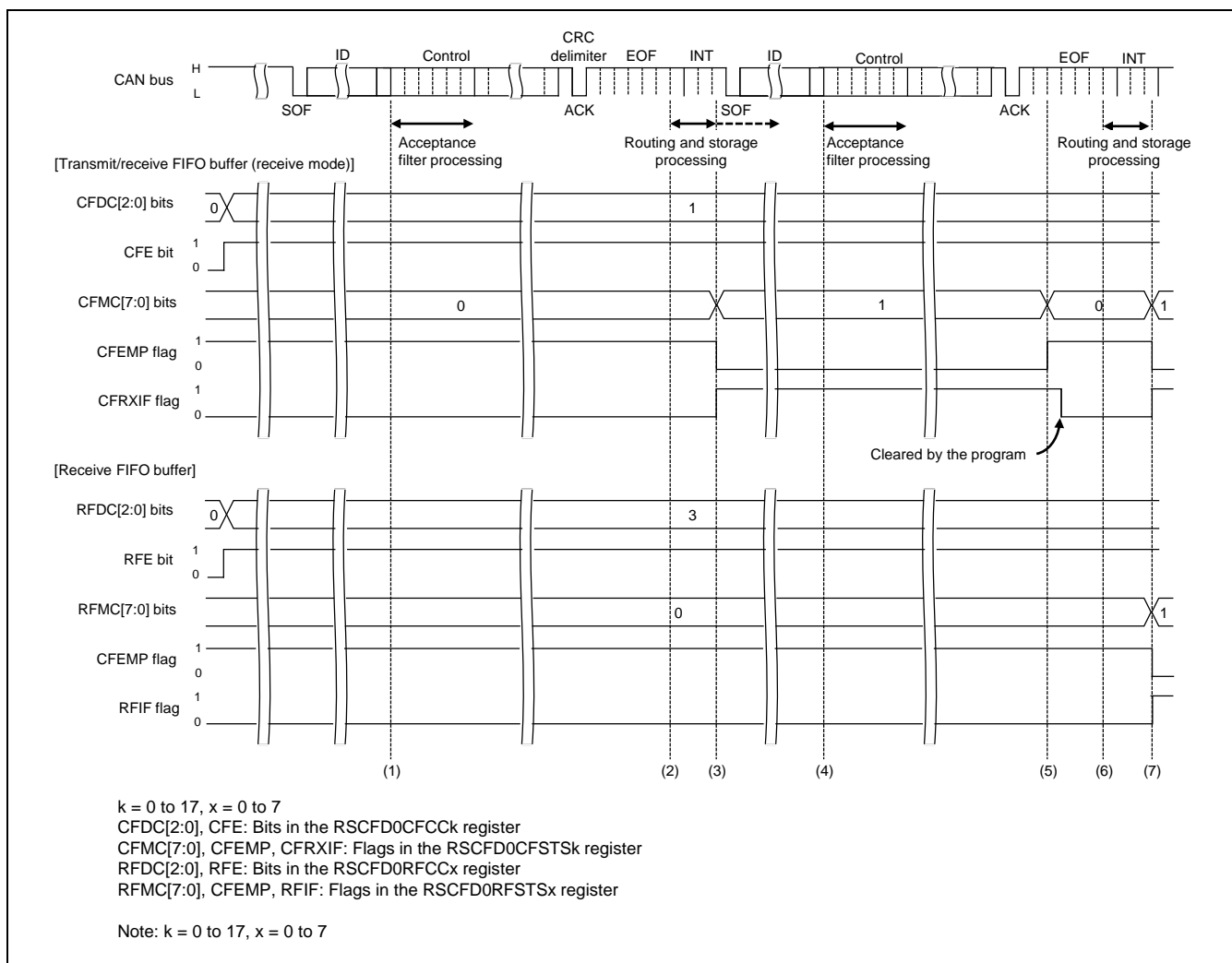


Figure 48.26 FIFO Buffer Reception Timing Chart

- (1) When the ID field in a message has been received, the acceptance filter processing starts.
- (2) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.
- (3) When the message has passed through the DLC filter processing and the CFE bit in the RSCFDn(CFD)CFCCk register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCFDn(CFD)CFCCk register is B'001 or more, the message is stored in the transmit/receive FIFO buffer that is set to receive mode. The CFMC[7:0] value in the RSCFDn(CFD)CFSTSk register is incremented and becomes H'01. When the CFIM bit in the RSCFDn(CFD)CFCCk register is set to 1 (a FIFO receive interrupt request is generated each time a message has been received), the CFRXIF flag in the RSCFDn(CFD)CFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). The CFRXIF flag can be reset to 0 by the program.
- (4) When the ID field of the next message has been received, the acceptance filter processing starts.
- (5) Read received messages from the RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, and RSCFDn(CFD)CFDFd_k registers and write H'FF to the RSCFDn(CFD)CFPCTRk register. This causes the CFMC[7:0] bits in the RSCFDn(CFD)CFSTSk register to be decremented. When CFMC[7:0] becomes H'00, the CFEMP flag in the RSCFDn(CFD)CFSTSk register becomes 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).
- (6) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(7) The message is stored in the transmit/receive FIFO buffer set in receive mode when the message has passed through the DLC filter process if the CFE bit is set to 1 (transmit/receive FIFO buffers are used), and the CFDC[2:0] bits are set to B'001 or more. The CFMC[7:0] bit value is incremented by 1 to be H'01. When the CFIM bit is set to 1 (an interrupt occurs each time a message has been received), the CFRXIF flag is set to 1 (a transmit/receive FIFO receive interrupt request is present). The message is stored in the receive FIFO buffer if the RFE bit in the RSCFDn(CFD)RFCCx register is set to 1 (receive FIFO buffers are used), and the RFDC[2:0] bits in the RSCFDn(CFD)RFCCx register are set to B'001 or more. The RFMC[7:0] bits in the RSCFDn(CFD)RFSTStx register are set to H'01 by being incremented by 1. When the RFIM bit in the RSCFDn(CFD)RFCCx register is set to 1 (an interrupt occurs each time a message has been received), the RFIF flag in the RSCFDn(CFD)RFSTStx register is set to 1 (a receive FIFO interrupt request is present).

48.16.2.3 Procedure for Reading FIFO Buffers by the DMA Transfer

In CAN FD mode, the following FIFO buffers can be read by the DMA transfer.

- All receive FIFO buffers x (x = 0 to 7)
- The first transmit/receive FIFO buffer k (k = 3 × m, m = 0 or 1) assigned to channel m

The DMA enable bit (the RFDMAEx or CFDMAEm bit in the RSCFDnCFDCDTC register) can be set at any timing. However, before this bit is set to 1 (the DMA transfer request is enabled), clear the receive interrupt enable bit of the related FIFO (the RFIE bit in the RSCFDnCFDRFCCx register or the CFRXIE bit in the RSCFDnCFDCFCCK register) to 0 (interrupts are disabled). When the DMA transfer request is enabled, do not write the FIFO control register (the RSCFDnCFDRFCCx or RSCFDnCFDCFCCK register).

If there are unread messages in the FIFO buffer that the DMA transfer is enabled, a DMA transfer request trigger will be generated. Specify the address for the FIFO access register to the transfer source address and adjust the transfer size so that the end of the payload storage area is read with a trigger. The end of the payload depends on the payload storage size set by the RFPLS[2:0] bits in the RSCFDnCFDRFCCx register or the CFPLS[2:0] bits in the RSCFDnCFDCFCCK register.

When the end of the payload stored in the FIFO buffer is read, the RFMC[7:0] bits in the RSCFDnCFDRFSTStx register or the CFMC[7:0] bits in the RSCFDnCFDCFSTSk register are automatically decremented by 1. If there is an unread message in the FIFO buffer after the end of the payload is read, a trigger will be generated again.

If the RFDMAEx or CFDMAEm bit is cleared to 0 (the DMA transfer request is disabled) during DMA transfer, wait for the DMA transfer state (the RFDMASTStx or CFDMASTSm bit in the RSCFDnCFDCDTSTStx register) is cleared to 0 (DMA is not being transferred) and perform the next processing (the DMA transfer is enabled again, etc.). When the DMA transfer is disabled, consider handling of messages that have remained in the FIFO buffer and are newly received. When the FIFO buffer is enabled, reception by the FIFO buffer will proceed.

The calculation of payload size should add the ID and PTR, STS. The ID and PTR, STS is the total 12byte.

If the payload size and transfer size of SYS-DMA is mismatch, use the PIO transfer.

The choices of SYS-DMA are following.

- 1byte
- 2byte
- 4byte
- 8byte
- 16byte
- 32byte
- 64byte

48.16.3 Transmission Procedure

48.16.3.1 Procedure for Transmission from Transmit Buffers

Figure 48.27 shows the procedure for transmission from transmit buffers.

Figure 48.28 shows a timing chart where messages are transmitted from two transmit buffers in the same channel and transmission has been successfully completed. Figure 48.29 shows a timing chart where messages are transmitted from two transmit buffers in the same channel and transmit abort has been completed.

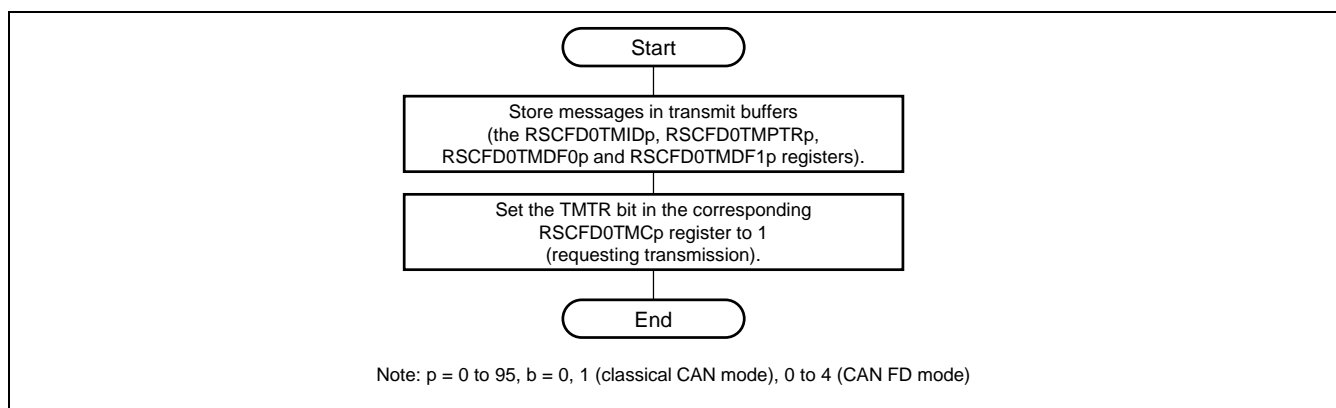


Figure 48.27 Procedure for Transmission from Transmit Buffers

In CAN FD mode and transmit buffer merge mode, transmit buffers (" $(16 \times m) + 0$ " and " $(16 \times m) + 3$ ") can transmit a message which has a payload of 20 bytes or more. In this case, transmit buffers (" $(16 \times m) + 1$ to $(16 \times m) + 2$ " and " $(16 \times m) + 4$ to $(16 \times m) + 5$ ") are assigned to the payload storage areas. The RSCFDnCFDTMIDp, RSCFDnCFDTMPTRp, and RSCFDnCFDTMFDCTRp registers corresponding to those buffers, similarly to the RSCFDnCFDTMDFb_p register, can be used as data field registers which store the 4-byte data byte (payload). Table 48.41 shows message storage registers when a message which has a payload of 20 bytes or more is transmitted in transmit buffer 0.

Table 48.41 Message Storage Registers in Transmit Buffer Merge Mode (Transmit Buffer 0)

| Transmit Buffer | Offset from Base Address | Abbreviation | Register Function for Transmit Buffer Merge Mode |
|-------------------|--------------------------|--------------------------------------|--|
| Transmit buffer 0 | H'4000 | RSCFDnCFDTMID0 | ID data, transmit history data storage enable, RTR and IDE bits in transmit buffer 0 |
| | H'4004 | RSCFDnCFDTMPTR0 | Label data and DLC data in transmit buffer 0 |
| | H'4008 | RSCFDnCFDTMFDCTR0 | ESI, BRS, and FDF bits in transmit buffer 0 |
| | H'400C to H'401C | RSCFDnCFDTMDF0_0 to RSCFDnCFDTMDF4_0 | Data bytes 0, 1, 2, and 3 in transmit buffer 0 to data bytes 16, 17, 18, and 19 in transmit buffer 0 |
| Transmit buffer 1 | H'4020 | RSCFDnCFDTMID1 | Data bytes 20, 21, 22, and 23 in transmit buffer 0 |
| | H'4024 | RSCFDnCFDTMPTR1 | Data bytes 24, 25, 26, and 27 in transmit buffer 0 |
| | H'4028 | RSCFDnCFDTMFDCTR1 | Data bytes 28, 29, 30, and 31 in transmit buffer 0 |
| | H'402C to H'403C | RSCFDnCFDTMDF0_1 to RSCFDnCFDTMDF4_1 | Data bytes 32, 33, 34, and 35 in transmit buffer 0 to data bytes 48, 49, 50, and 51 in transmit buffer 0 |
| Transmit buffer 2 | H'4040 | RSCFDnCFDTMID2 | Data bytes 52, 53, 54, and 55 in transmit buffer 0 |
| | H'4044 | RSCFDnCFDTMPTR2 | Data bytes 56, 57, 58, and 59 in transmit buffer 0 |
| | H'4048 | RSCFDnCFDTMFDCTR2 | Data bytes 60, 61, 62, and 63 in transmit buffer 0 |
| | H'404C to H'405C | RSCFDnCFDTMDF0_2 to RSCFDnCFDTMDF4_2 | Not used |

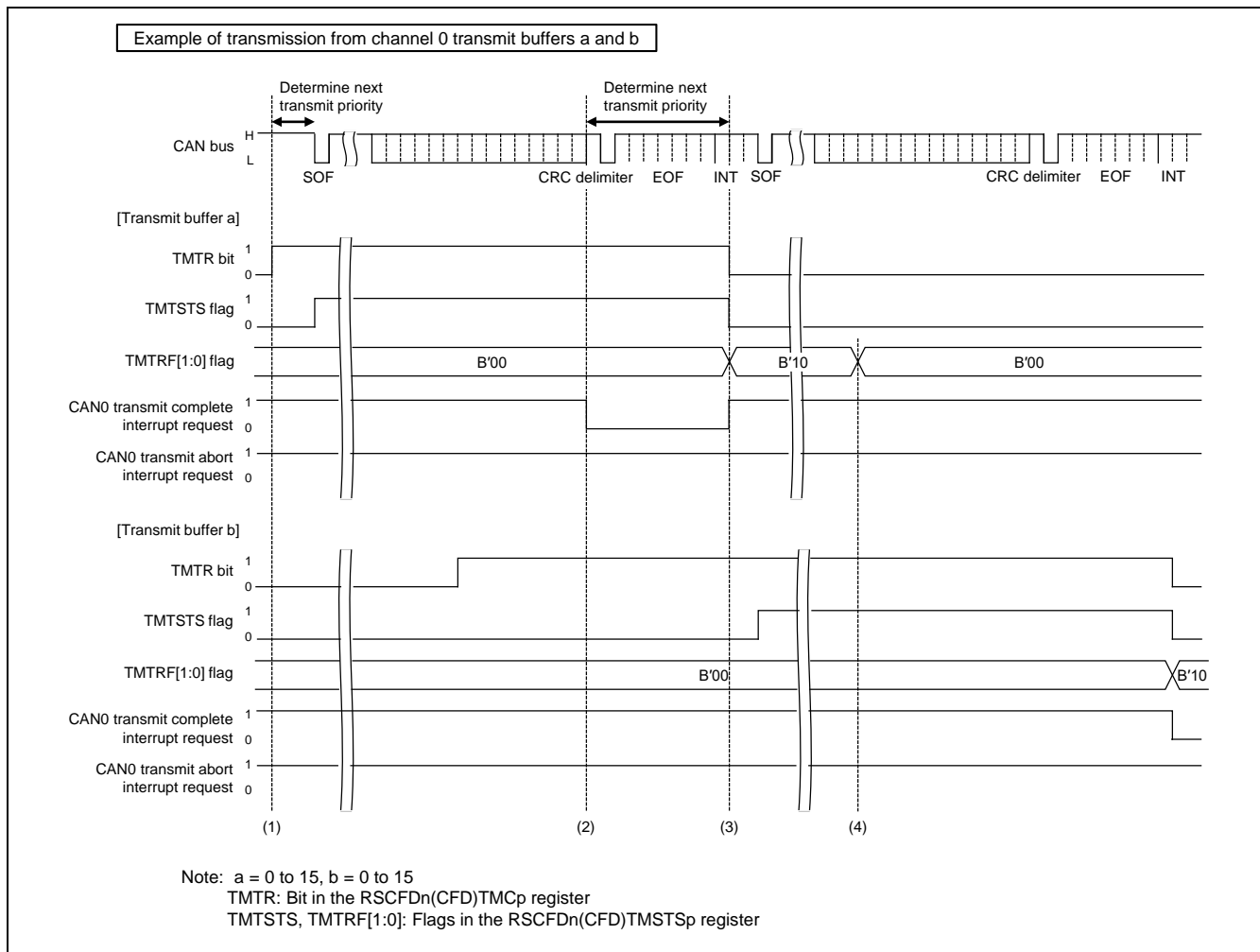


Figure 48.28 Transmit Buffer Transmission Timing Chart (Transmission Completed Successfully)

- (1) When the TMTR bit in the RSCFDn(CFD)TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding RSCFDn(CFD)TMSTSa register is set to 1 (transmission is in progress) and the CAN channel starts transmitting data.
- (2) When a transmit request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.
- (3) When transmission completes successfully, the TMTRF[1:0] flag in the RSCFDn(CFD)TMSTSa register is set to B'10 (transmission has been completed (without transmit abort request)) and the TMTSTS flag and the TMTR bit in the RSCFDn(CFD)TMIEC0 register are cleared to 0. When the TMIEa bit in the RSCFDn(CFD)TMIEC0 register is 1 (transmit buffer interrupt is enabled), a CAN0 transmit complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to B'00 (transmission is in progress or no transmit request is present).
- (4) Before starting the next transmission, set the TMTRF[1:0] flag to B'00. Write the next message to the transmit buffer, and then set the TMTR bit to 1 (transmission is requested). The TMTR bit can be set to 1 only when the TMTRF[1:0] flag value is B'00.

If an arbitration lost has occurred after transmission is started, the TMTSTS flag is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC delimiter to search the highest-priority transmit buffer. If an error has occurred during transmission or after arbitration lost, the priority determination processing is reexecuted during transmission of an error frame.

If an ECC2 bit error is detected in the priority determination processing, transmission will not be performed (for classical CAN mode, the EEFEBIT bit in the RSCFDnGCFG register is 1).

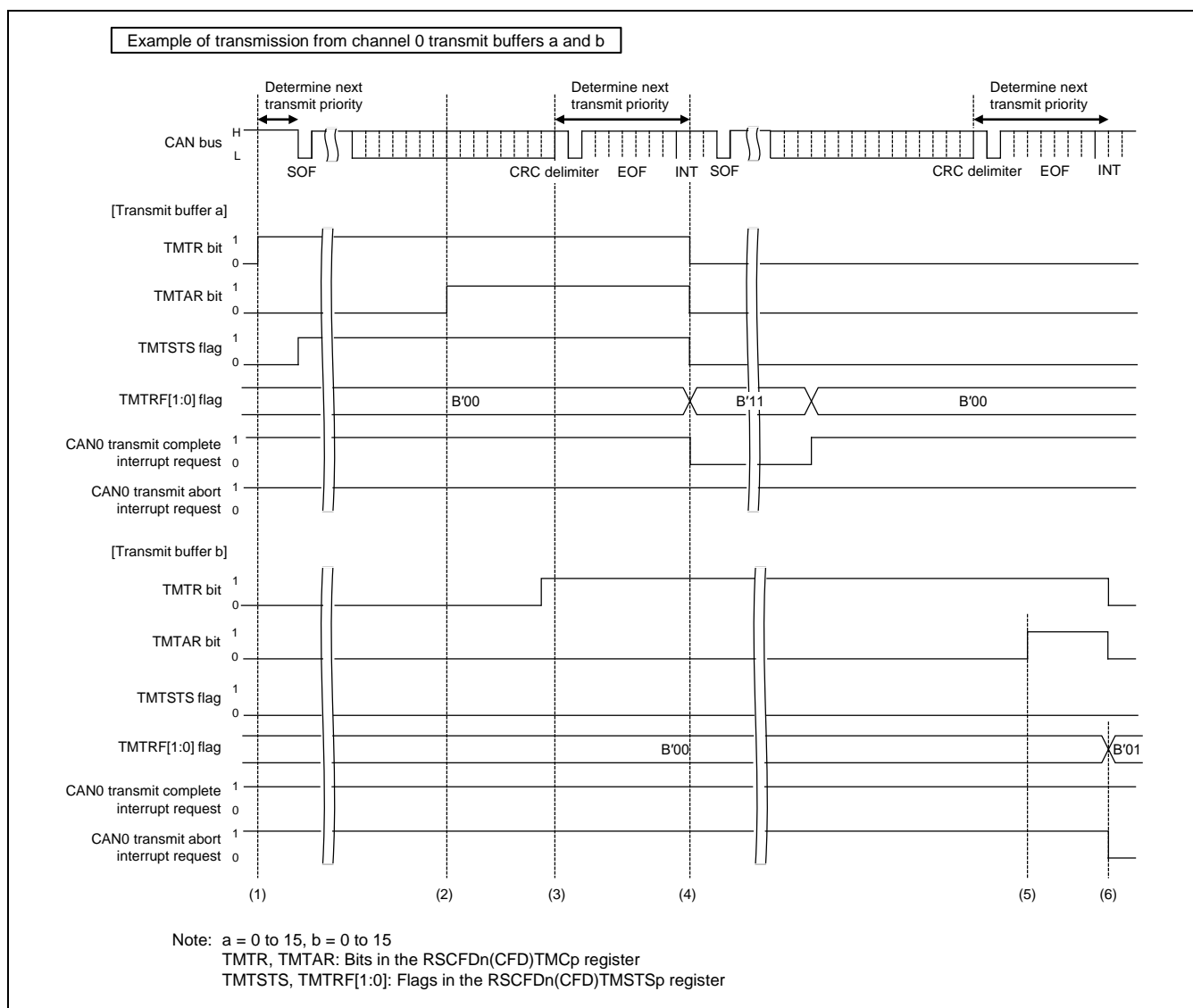


Figure 48.29 Transmit Buffer Transmission Timing Chart (Transmit Abort Completed)

- (1) When the TMTR bit in the RSCFDn(CFD)TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding RSCFDn(CFD)TMSTSa register is set to 1 (transmission is in progress) and the CAN channel starts transmitting data.
- (2) When it is determined that the transmit buffer is used for the next transmission or transmission is in progress, message transmission is not aborted unless an error or arbitration loss occurs even if the TMTAR bit is set to 1 (transmit abort is requested).
- (3) The priority determination starts with the CRC delimiter for the next transmission. In this timing chart, buffer b is not selected as the next transmit buffer. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.
- (4) When transmission completes successfully, the TMTRF[1:0] flag in the RSCFDn(CFD)TMSTSa register is set to B'11 (transmission has been completed (with transmit abort request)) and the TMTSTS flag and the TMTR bit in the RSCFDn(CFD)TMCa register are cleared to 0. When the TMIEa value in the RSCFDn(CFD)TMIEC0 register is 1

(transmit buffer interrupt is enabled), a CAN0 transmit complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to B'00 (transmission is in progress or no transmit request is present).

- (5) While another CAN node is transmitting data on the CAN bus (TMTSTS flag = 0), if the TMTAR bit is set to 1 while the corresponding channel is determining transmit priority, the TMTR bit cannot be cleared to 0.
- (6) After the internal processing time has passed, the transmission is terminated and the TMTRF[1:0] flag is set to B'01. When the transmit buffer is not transmitting data and is not selected as the next transmit buffer and priority determination is not being made, an abort request is immediately accepted and the TMTRF[1:0] flag is set to B'01. At this time, the TMTR and TMTAR bits are cleared to 0. When transmit abort is completed with the TAIE bit in the RSCFDn(CFD)CmCTR register set to 1 (transmit abort interrupt is enabled), an interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to B'00.

If an arbitration lost has occurred after the CAN channel started transmission, the TMTSTS bit is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC delimiter to find the highest-priority transmit buffer. If an error has occurred during transmission or after arbitration lost, the priority determination processing is reexecuted during transmission of an error frame.

If an ECC2 bit error is detected in the priority determination processing, transmission will not be performed (for classical CAN mode, the EEFE bit in the RSCFDnGCFG register is 1).

48.16.3.2 Procedure for Transmission from Transmit/Receive FIFO Buffers

Figure 48.30 shows the procedure for transmission from transmit/receive FIFO buffers.

Figure 48.31 shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers in the same channel and transmission has been successfully completed. Figure 48.32 shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers in the same channel and transmit abort has been completed.

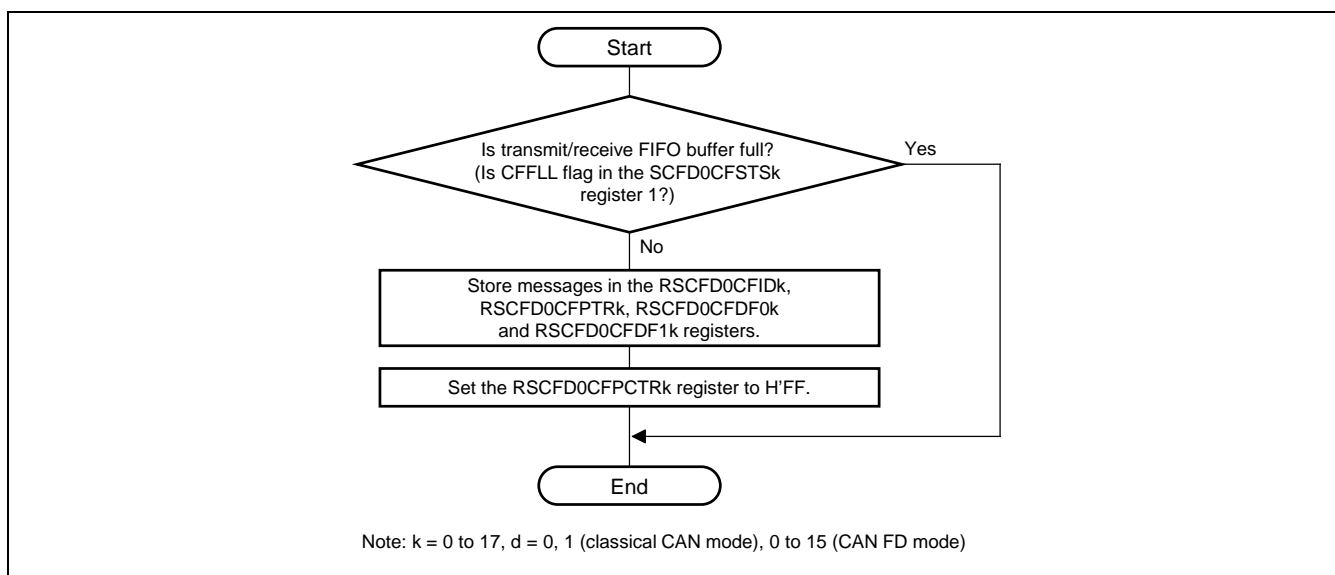


Figure 48.30 Procedure for Transmission from Transmit/Receive FIFO Buffers

When a message is stored, do not write the RSCFDnCFDCFDf_d_k register corresponding to the area which exceeds the payload storage size set by the CFPLS[2:0] bits in the RSCFDnCFDCFCCK register.

Table 48.42 Payload Storage Area in the Transmit/Receive FIFO Buffer

| Value Set by the CFPLS[2:0] Bits | Payload Storage Size | Corresponding Data Field Register |
|----------------------------------|----------------------|---------------------------------------|
| B'000 | 8 bytes | RSCFDnCFDCFDf0_k to RSCFDnCFDCFDf1_k |
| B'001 | 12 bytes | RSCFDnCFDCFDf0_k to RSCFDnCFDCFDf2_k |
| B'010 | 16 bytes | RSCFDnCFDCFDf0_k to RSCFDnCFDCFDf3_k |
| B'011 | 20 bytes | RSCFDnCFDCFDf0_k to RSCFDnCFDCFDf4_k |
| B'100 | 24 bytes | RSCFDnCFDCFDf0_k to RSCFDnCFDCFDf5_k |
| B'101 | 32 bytes | RSCFDnCFDCFDf0_k to RSCFDnCFDCFDf7_k |
| B'110 | 48 bytes | RSCFDnCFDCFDf0_k to RSCFDnCFDCFDf11_k |
| B'111 | 64 bytes | RSCFDnCFDCFDf0_k to RSCFDnCFDCFDf15_k |

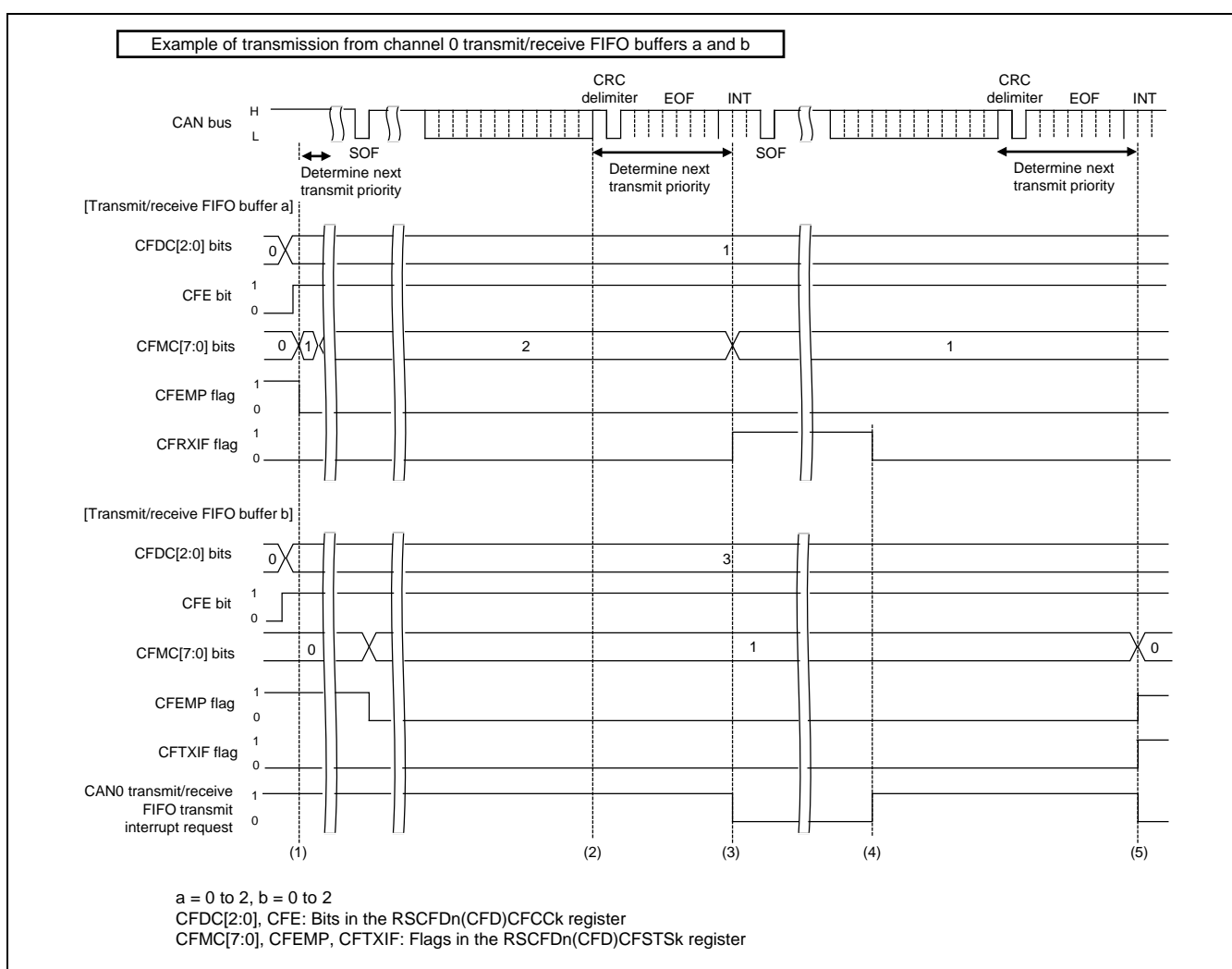
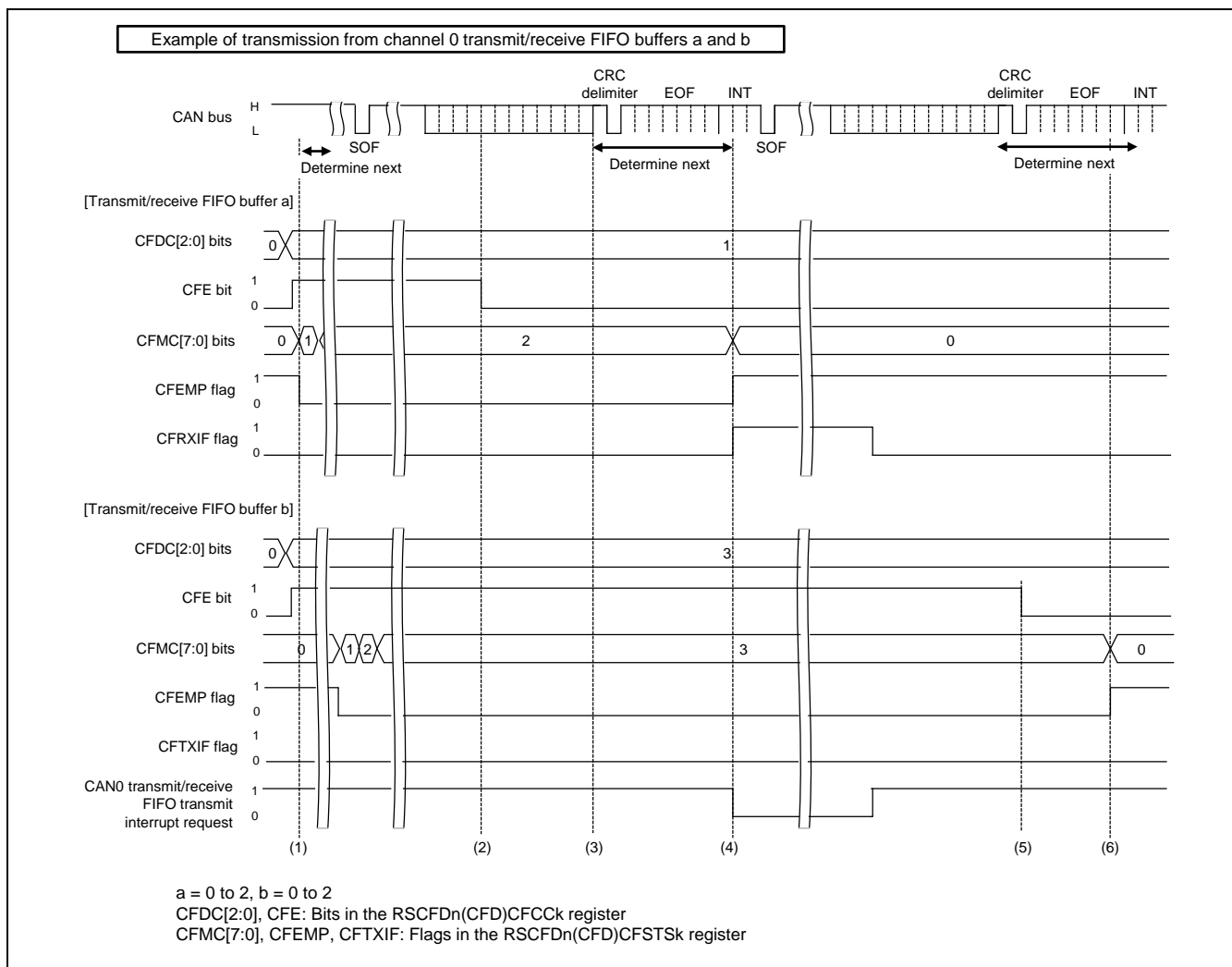


Figure 48.31 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmission Completed Successfully)

(1) While the CAN bus is idle, when the CFE bit in the RSCFDn(CFD)CFCCa register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCFDn(CFD)CFCCa register is B'001 (4 messages) or more and the CFMC[7:0] value in the RSCFDn(CFD)CFSTSa register is H'01 or more, the priority determination processing starts to determine the highest-priority transmit message. When the highest-priority transmit message has been determined, transmission of the message starts. In this figure, the message is transmitted from transmit/receive FIFO buffer a of channel 0.

- (2) When a transmit request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.
- (3) When transmission completes successfully, the CFMC[7:0] value in the RSCFDn(CFD)CFSTSa register is decremented. Setting the CFIM bit in the RSCFDn(CFD)CFCCa register to 1 (a FIFO transmit interrupt request is generated each time a message has been transmitted) sets the CFTXIF flag in the RSCFDn(CFD)CFSTSk register to 1 (a transmit/receive FIFO transmit interrupt request is present).
- (4) The program can clear the CFTXIF flag.
- (5) Message transmission from transmit/receive FIFO buffer b of channel 0 has been completed and the CFMC[7:0] value in the RSCFDn(CFD)CFSTSB register is decremented. The CFMC[7:0] bits are cleared to H'00 and therefore the CFEMP flag in the RSCFDn(CFD)CFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).

Transmission is continued until the CFEMP flag is set to 1. It is possible to continuously store transmit messages in FIFO buffers until the CFFLL flag in the RSCFDn(CFD)CFSTSa and RSCFDn(CFD)CFSTSB register is set to 1 (the transmit/receive FIFO buffer is full).



**Figure 48.32 Transmit/Receive FIFO Buffer Transmission Timing Chart
(Transmission Abort Completed)**

- (1) While the CAN bus is idle, when the CFE bit in the RSCFDn(CFD)CFCCa register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCFDn(CFD)CFCCa register is B'001 (4 messages) or more and the CFMC[7:0] value in the RSCFDn(CFD)CFSTSa register is H'01 or more, the priority determination processing starts to determine the highest-priority transmit message. When the highest-priority transmit message has been determined, transmission of the message starts. In this figure, the message is transmitted from transmit/receive FIFO buffer a of channel 0.
- (2) When transmission is in progress or it is determined that the transmit/receive FIFO buffer is used for the next transmission, message transmission is not aborted unless an error or arbitration loss occurs even if the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).
- (3) When a transmit request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission. In this figure, transmit/receive FIFO buffer b is not selected as a buffer for the next transmission. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.
- (4) When transmit completes successfully, the CFMC[7:0] value is cleared to H'00. Setting the CFIM bit to 1 (a FIFO transmit interrupt request is generated each time a message has been transmitted) sets the CFTXIF flag in the RSCFDn(CFD)CFSTSa register to 1 (a transmit/receive FIFO transmit interrupt request is present). The program can clear the CFTXIF flag.
- (5) If another CAN node on the CAN bus is transmitting data (not from transmit/receive FIFO buffer b), transmit/receive FIFO buffers cannot be disabled immediately even if the CFE bit in the RSCFDn(CFD)CFCCb register is cleared to 0 (no transmit/receive FIFO buffer is used) during transmit priority determination. (The CFEMP flag in the RSCFDn(CFD)CFSTSa register is not set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)) immediately.)
- (6) After the internal processing time has passed, transmit/receive FIFO buffers are disabled and the CFMC[7:0] bits in the RSCFDn(CFD)CFSTSa register are cleared to H'00 and the CFEMP flag is set to 1. When the transmit/receive FIFO buffer is not transmitting data and is not selected as the next transmit buffer and priority determination is not in progress, the transmit/receive FIFO buffer is immediately disabled. (The CFMC[7:0] bits are cleared to H'00 and the CFEMP flag is set to 1.)

48.16.3.3 Procedure for Transmission from the Transmit Queue

Figure 48.33 shows the procedure for transmission from the transmit queue.

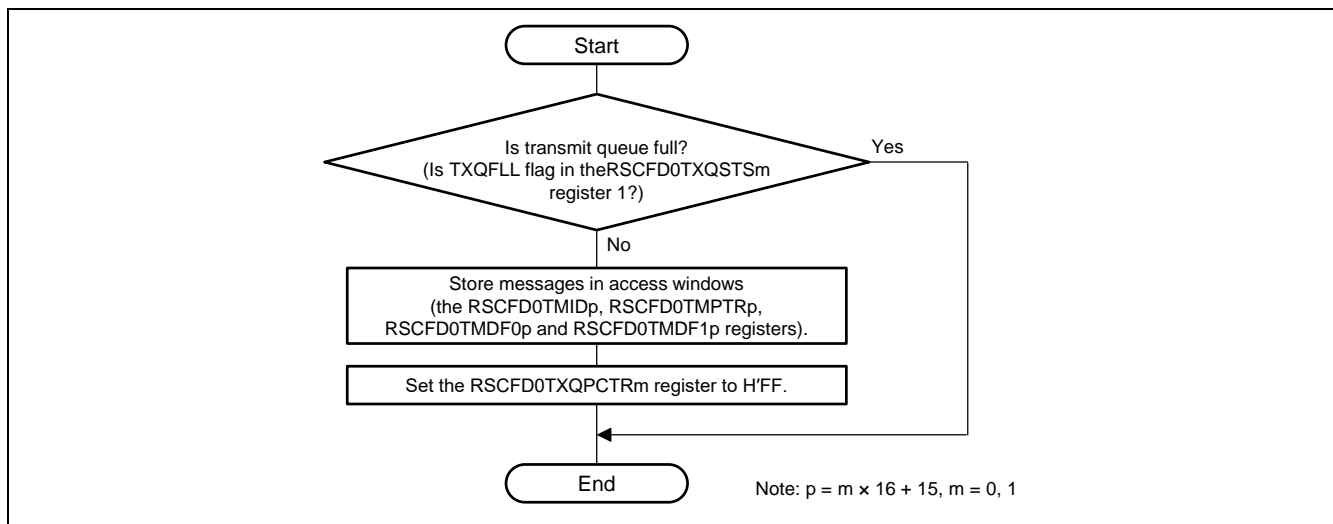


Figure 48.33 Procedure for Transmission from the Transmit Queue

48.16.3.4 Transmit History Buffer Reading Procedure

Transmit history data can be read from the RSCFDn(CFD)THLACCm register. The next data can be accessed by writing H'FF to the corresponding RSCFDn(CFD)THLPCTRm register (m = 0 or 1) after reading a set of data. Figure 48.34 shows the transmit history buffer reading procedure.

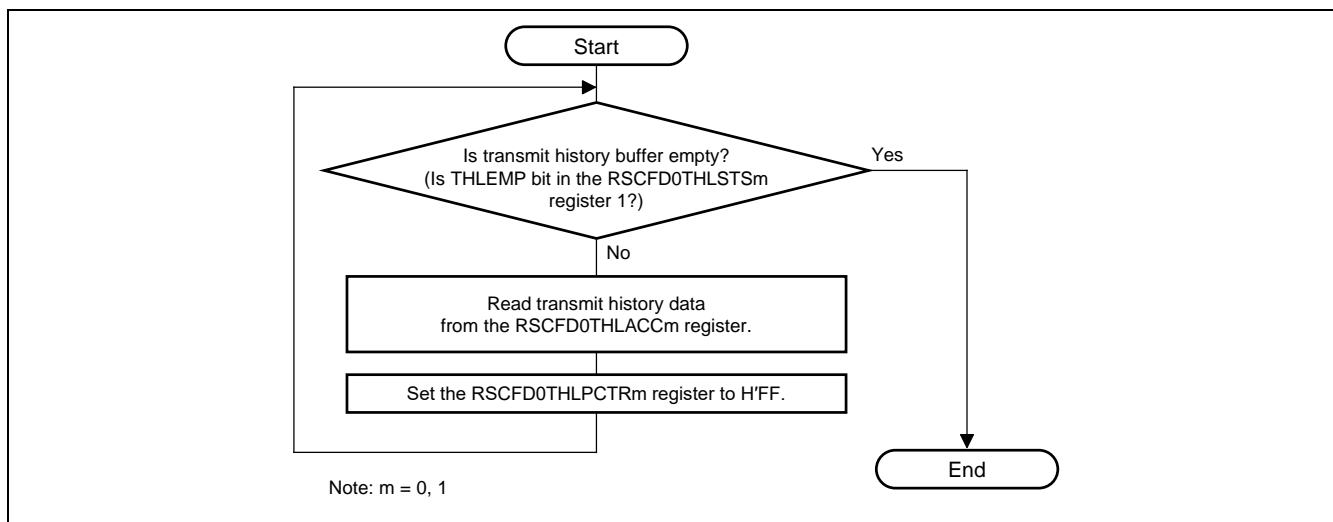


Figure 48.34 Transmit History Buffer Reading Procedure

48.16.4 Test Settings

48.16.4.1 Self-Test Mode Setting Procedure

Self-test mode allows communication test on a channel basis by enabling a CAN node to receive its own transmitted messages.

Figure 48.35 shows the self-test mode setting procedure.

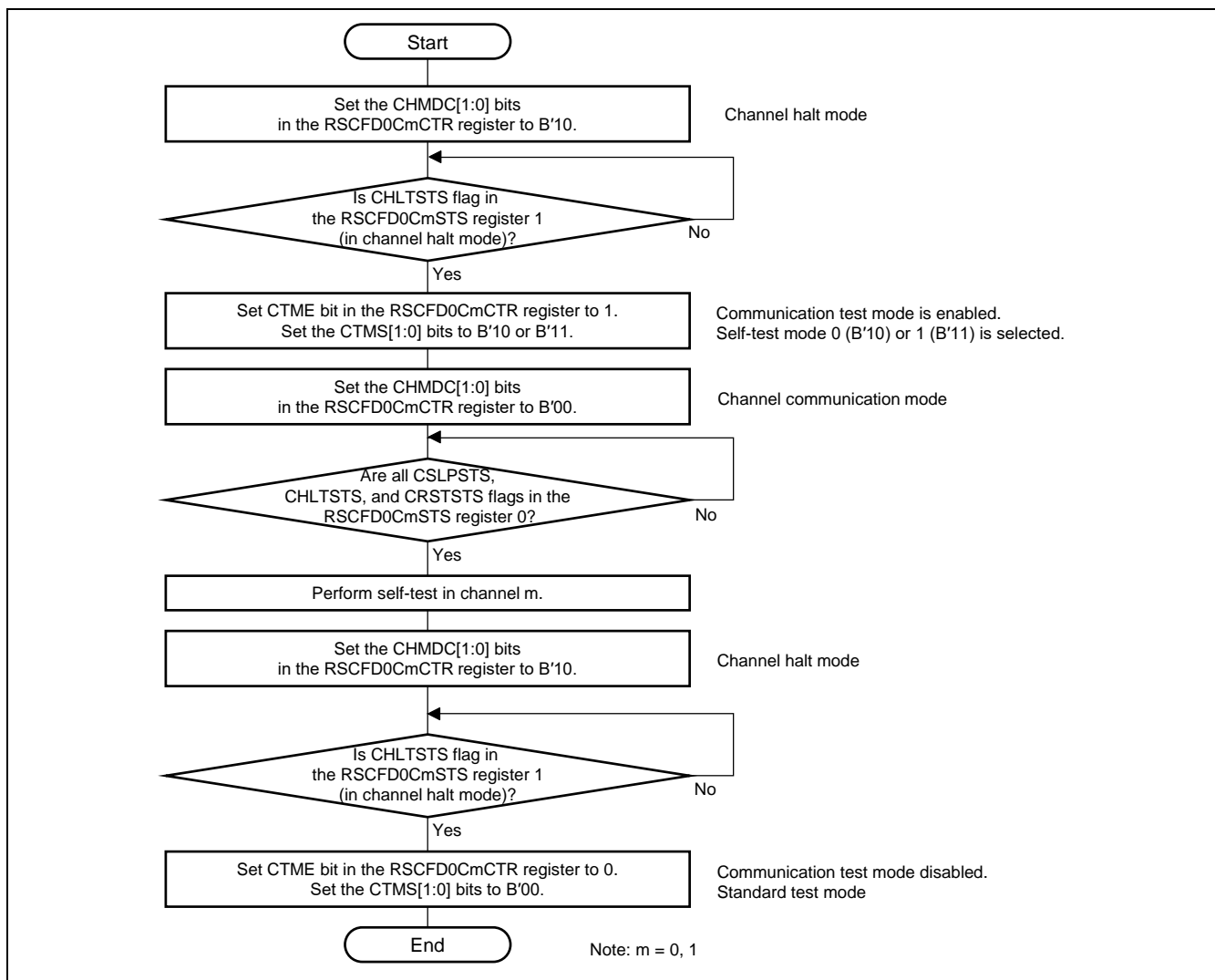


Figure 48.35 Self-Test Mode Setting Procedure

48.16.4.2 Procedure for Releasing the Protection

Since the global test function in Table 48.43 is protected, write the protection release data 1 and release data 2 in succession to the LOCK[15:0] bits in the RSCFDn(CFD)GLOCKK register, then set the target test bit to 1.

Table 48.43 Protection Release Data for Test Function

| Test Function | Protection Release Data 1 | Protection Release Data 2 | Target Bit |
|---------------|---------------------------|---------------------------|---|
| RAM test | H'7575 | H'8A8A | RTME bit in the RSCFDn(CFD)GTSTCTR register |

If an incorrect value is written to the LOCK[15:0] bits, restart from writing the protection release data 1. Figure 48.36 shows the procedure for releasing the protection.

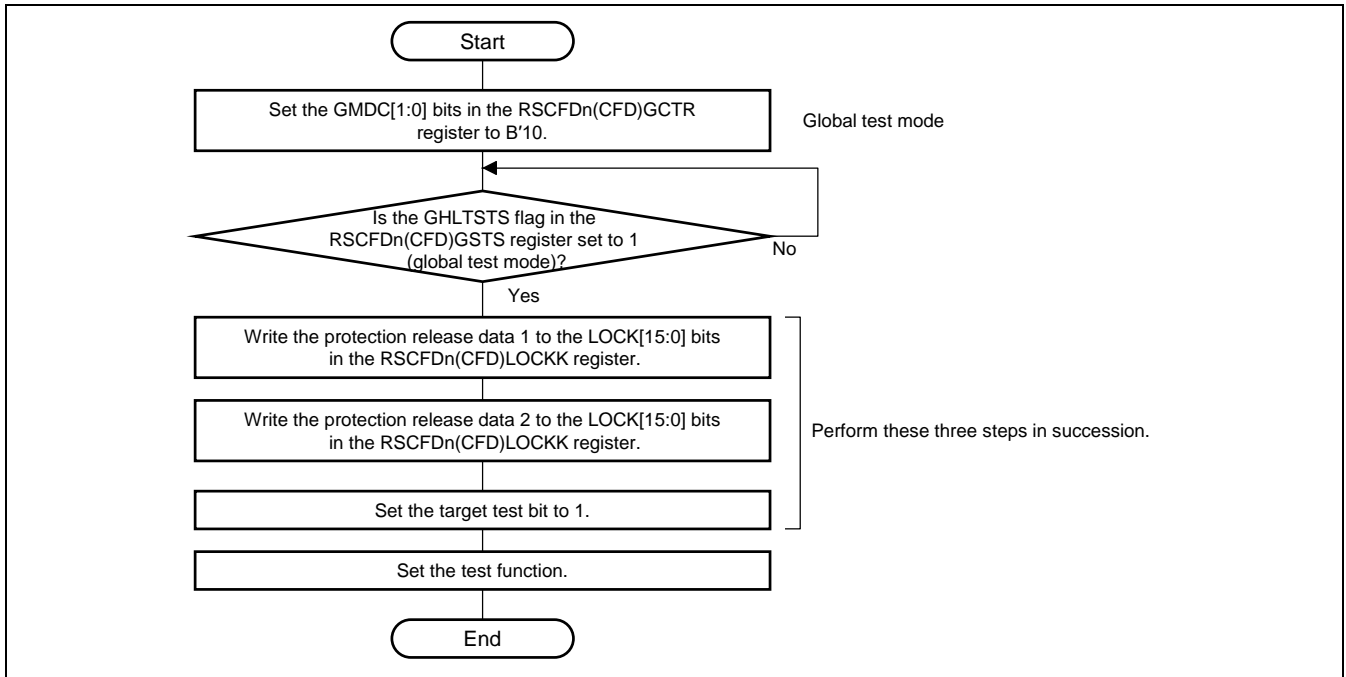


Figure 48.36 Protection Release Procedure

48.16.4.3 RAM Test Setting Procedure

RAM tests include CAN RAM read/write test. The read/write test verifies that data written to the RAM is read correctly. Before closing the RAM test, write H'0000_0000 to all pages of the CAN RAM.

Figure 48.37 shows the RAM test setting procedure.

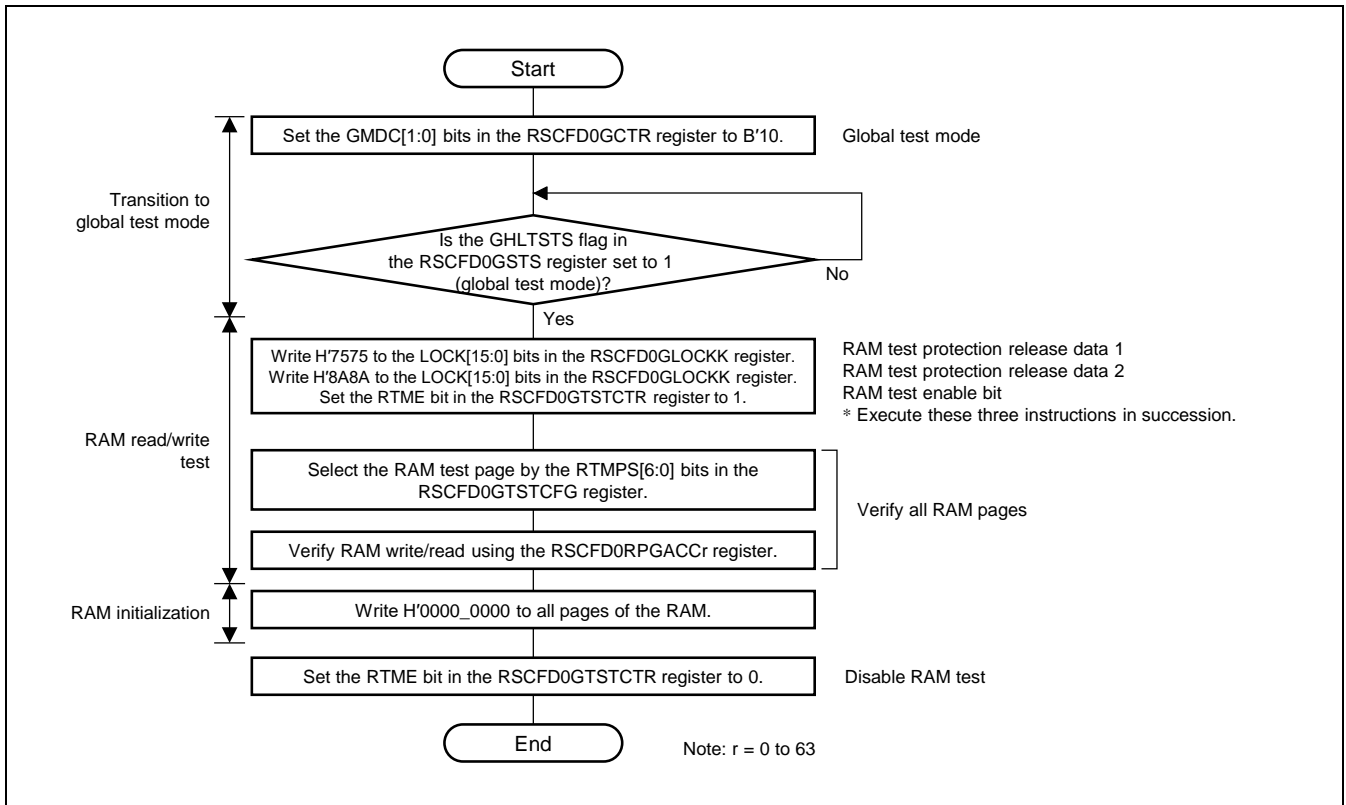


Figure 48.37 RAM Test Setting Procedure

48.16.4.4 Inter-Channel Communication Test Setting Procedure

Communication testing can be performed by transmitting and receiving data between different channels.

Figure 48.38 shows the inter-channel communication test setting procedure.

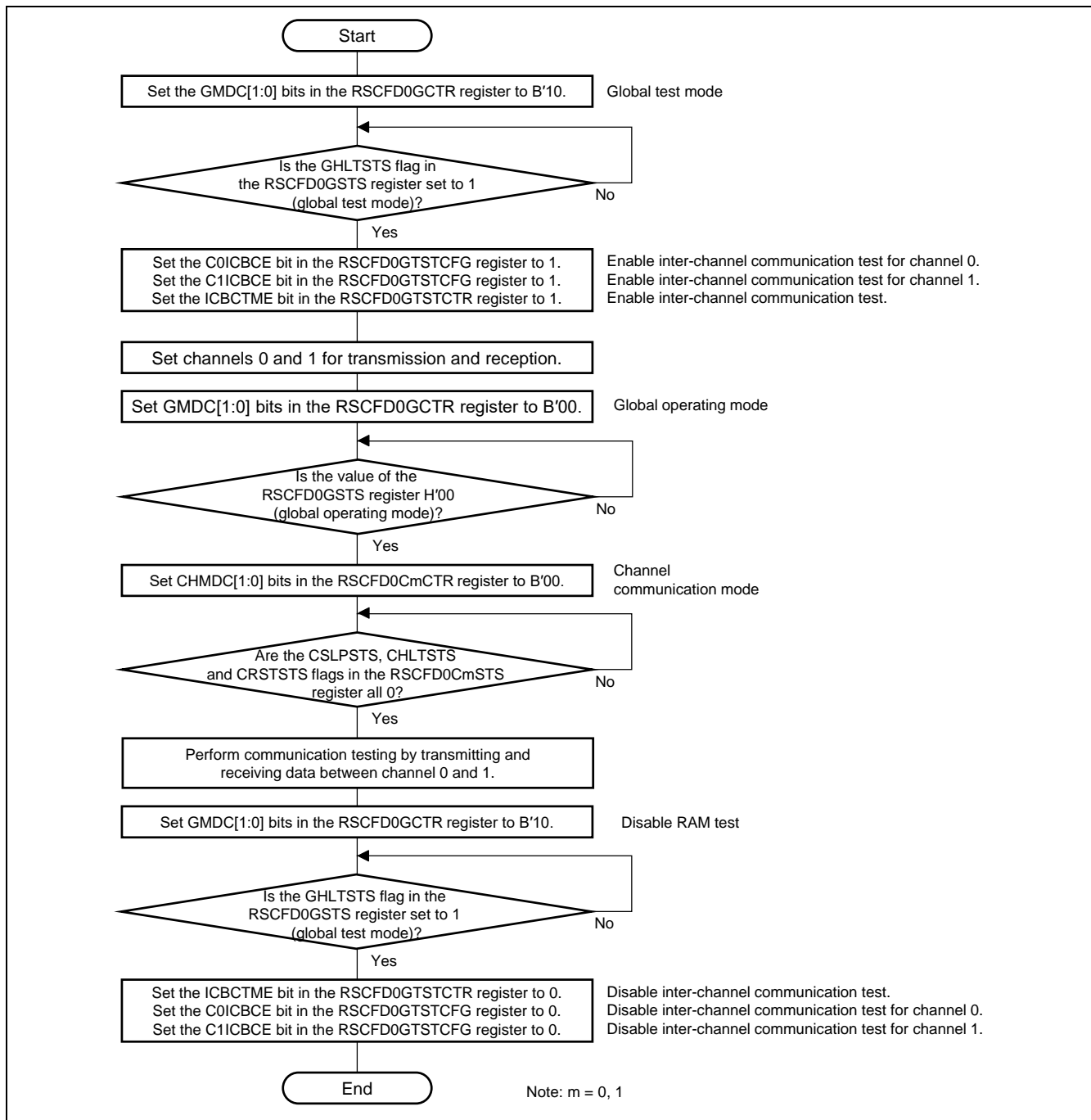


Figure 48.38 Inter-Channel Communication Test Setting Procedure (Example of Communication Test between Channel 0 and Channel 1)

48.17 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

- When the interface mode is changed without resetting RS-CANFD, write values after a reset to all registers and bits that are not assigned to the switched register map and rewrite the RSCFDnCFDGRMCFG register.
- When changing a global mode, check the GSLPSTS, GHLTSTS, and GRSTSTS flags in the RSCFDn(CFD)GSTS register for transitions. When changing a channel mode, check the CSLPSTS, CHLTSTS, and CRSTSTS flags in the RSCFDn(CFD)CmSTS register (m = 0 or 1) for transitions.
- If a classical CAN frame is only used in CAN FD mode, specify the same value to the RSCFDnCFDCmDCFG register as that specified for the RSCFDnCFDCmNCFG register.
- The acceptance filter processing checks receive rules sequentially in ascending order from the minimum rule number. If the same ID, IDE bit, or RTR bit value is set for multiple receive rules, the minimum number of receive rule is used for the acceptance filter processing. If the message does not pass through the subsequent DLC filter processing, the data processing is terminated without returning to the acceptance filter processing and the message is not stored in the buffer.
- When linking transmit buffers to transmit/receive FIFO buffers or allocating transmit buffers to transmit queues, set the control register (RSCFDn(CFD)TMCp) of the corresponding transmit buffer to H'00. The status register (RSCFDn(CFD)TMSTSp) of the corresponding transmit buffer should not be used. Flags in other status registers (registers RSCFDn(CFD)TMTRSTS0 to RSCFDn(CFD)TMTRSTS2, RSCFDn(CFD)TMTARSTS0 to RSCFDn(CFD)TMTARSTS2, RSCFDn(CFD)TMTASTS0 to RSCFDn(CFD)TMTASTS2), which correspond to transmit buffers linked to transmit/receive FIFO buffers or allocated to transmit queues remain unchanged. Set the enable bit in the corresponding interrupt enable register (registers RSCFDn(CFD)TMIEC0 to RSCFDn(CFD)TMIEC2) to 0 (transmit buffer interrupt is disabled).
- When the transmit buffer merge mode (for CAN FD mode) is used, set H'00 to the control register (the RSCFDn(CFD)TMCp register) of the transmit buffer corresponding to the transmit buffer that has been assigned to the payload storage area. The enable bit of the corresponding interrupt enable registers (RSCFDn(CFD)TMIEC0 to RSCFDn(CFD)TMIEC2 registers) must be cleared to 0 (interrupts are disabled).
- Transmit buffers that are linked to transmit/receive FIFO buffers must not be assigned to transmit queues. Additionally, transmit buffers that are assigned to the payload storage area in buffer merge mode (CAN FD mode) must not be assigned to transmit queues.
- Only a single transmit/receive FIFO buffer can be linked to a transmit buffer. Do not link two or more transmit/receive FIFO buffers to transmit buffers of the same number.
- When the CANm bit time clock is selected as a timestamp counter clock source in classical CAN mode, the timestamp counter stops when the corresponding channel has transitioned to channel reset mode or channel halt mode.
- In case of an attempt to store a new received message when the receive FIFO buffer and the transmit/receive FIFO buffer are full, the new message is discarded. If you wish to store a new transmit message in the transmit/receive FIFO buffer or the transmit queue, check that the transmit/receive FIFO buffer or the transmit queue is not full.
- When registers access RAM, values after a reset described in chapter 48.3, Registers (classical CAN mode), and chapter 48.4, Registers (CAN FD mode), indicate values that have been cleared by initialization of RAM for CAN. Values that have not been cleared are undefined. The registers are shown below.
 - Receive rule (RSCFDn(CFD)GAFLIDj, RSCFDn(CFD)GAFLMj, RSCFDn(CFD)GAFLP0_j, and RSCFDn(CFD)GAFLP1_j registers)
 - Receive buffer (RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq, and RSCFDn(CFD)RMDFb_q registers)
 - Receive FIFO buffer access (RSCFDn(CFD)RFIDx, RSCFDn(CFD)RFPTRx, RSCFDnCFDRFFDSTSx, and RSCFDn(CFD)RFDFd_x registers)

- Transmit/receive FIFO buffer access (RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDn(CFD)CFDFd_k registers)
- Transmit buffer (RSCFDn(CFD)TMIDp, RSCFDn(CFD)TMPTRp, RSCFDnCFDTMFDCTRp, and RSCFDn(CFD)TMDFb_p registers)
- Transmit history access (RSCFDn(CFD)THLACCm register)
- RAM test page access (RSCFDn(CFD)RPGACCr register)
- The values of unused receive buffers (RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq, and RSCFDn(CFD)RMDFb_q registers), receive FIFO buffer access registers (RSCFDn(CFD)RFIDx, RSCFDn(CFD)RFPTRx, RSCFDnCFDRFFDSTsx, and RSCFDn(CFD)RFDFd_x registers), and transmit/receive FIFO buffer access registers (RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDn(CFD)CFDFd_k registers) are undefined when the RS-CAN module transitions to global operation mode or global test mode after exiting from global reset mode.

49. PCIe Controller

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

49.1 Overview

This module performs PCIe controls, and transfers data between this LSI internal bus (AXI bus) and the PCI devices connected to the PCIe Controller. This module facilitates the design of systems using the PCI Express while at the same time permitting high-speed data transfers in a compact system.

This module functions as a bus bridge connecting the PCI Express to this LSI internal bus (AXI bus), and provides transfer channels between PCI Express devices and devices connected to the LSI internal bus (AXI bus).

This LSI incorporates a PCI Express physical module that supports one lane (×1).

This module has two modes: a Root Port and an Endpoint and can operate as a Root Port or Endpoint that is defined in the PCI Express specification.

49.1.1 Features

This module has the following features.

- The AXI bus clock and APB bus clock that are used in PCIEC are as below.
 - S3D1 ϕ and S3D2 ϕ (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N)
 - S3D1 ϕ (RZ/G2E)
- Functions as a bridge that connects the PCI Express to the LSI internal bus (AXI bus)
 - Generates a PCIe packet from the AXI bus transaction to the PCI area
 - Generates a AXI bus transaction from a PCI Express packet
- Supports high-speed data transfer between PCI Express devices and internal bus (AXI bus) via the internal DMAC
- Supports the subset of PCI Express Base Specification Revision 2.0 (Dec. 20, 2006)
- Operates as a PCI Express Endpoint or a PCI Express Root Port
- Supports the requester and completer functions for the PCIe transaction
- Incorporates configuration registers and the following capability structures
 - Power management capability structure
 - MSI capability structure (supports masking function and pending bits.)
 - PCI Express capability structure
 - Virtual channel capability structure
- Supports interrupts by INTx and MSI
- Supports one lane (×1)
- Supports multiple speeds (2.5 GT/s and 5.0 GT/s).
- Automatically performs link training and link configurations
- Supports the D+/D-line automatic swap function (lane polarity inversion)
- Achieves QoS through transaction ordering.
- Automatically performs flow control by managing six types of credits
- Automatic handling function by hardware
- Buffer for holding error packet headers and error indication status provided
- Supports the link power control function (L0, L0s, and L1 states)
- Supports device power control function (D0, D3 hot, and D3 cold states).

- Supports 128-byte max payload size.
- Supports completion timeout disable.

Note: This module does not support the following PCI Express functions.

- Expansion ROM for system boot
- Card bus
- Phantom function
- Power-management event (PME) handling by hardware
- Slot function (including Hot Plug)
- AER function
- BIST function
- Surprise Down error function
- ARI forwarding function
- Port arbitration function
- Peer-to-peer transfer
- Separate Refclk Architecture

49.1.2 Block Diagram

Figure 49.1 shows a block diagram of the PCIE Controller.

The PCIE Controller comprises a bridge, PCIE Controller, and PCIE Controller physical unit.

The bridge provides the bridge function of connecting the internal bus (AXI bus) to the PCI Express controller.

The PCIE Controller is a block that controls the PCI Express packet transmission and reception, and it implements the functions of transaction and data link layers that are defined by the PCI Express standards. The configuration registers defined by the PCI Express standards are installed in this block.

The PCIE Controller physical unit is a block that controls PCI Express transmission paths, and implements the physical layer function that is defined by the PCI Express standards.

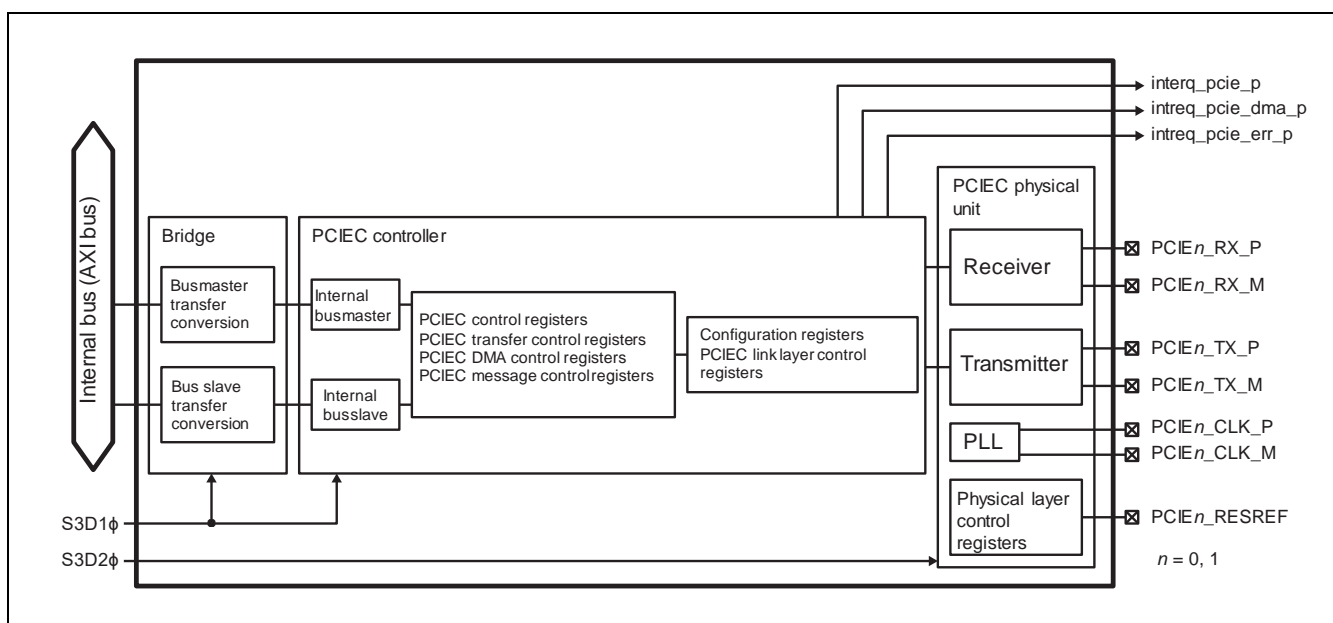


Figure 49.1 Block Diagram

49.1.3 External Pins

Table 49.1 shows the pin configuration.

Table 49.1 Pin Configuration

| Signal Name | Signal Conforming to PCI Standard | I/O | Description | Second Generation RZ/G Series Products | | | | |
|--|-----------------------------------|--------|--|--|----------------------------|--------|--------|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| PCIE _n _CLK_P* PCIE _n _CLK_M* (n = 0, 1) | REFCLK+ | Input | Reference clock inputs to the PLL incorporated in the PCIE Controller module (differential inputs). 100-MHz clock should be applied. | √ | √ | √ | √ | |
| PCIE _n _TX_P* PCIE _n _TX_M* (n = 0, 1) | PETp0 PETn0 | Output | Transmit data pins used by a physical module, where 2.5- or 5.0-GHz signals are propagated. (Differential output) | √ | √ | √ | √ | |
| PCIE _n _RX_P* PCIE _n _RX_M* (n = 0, 1) | PERp0 PERn0 | Input | Receive data pins used by a physical module, where 2.5- or 5.0-GHz signals are propagated. (Differential input) | √ | √ | √ | √ | |
| PCIE _n _RESREF* (n = 0, 1) | — | InOut | Reference Resistor Connection Attach a 200-Ω ±1% ±100 ppm/°C precision resistor-to-ground on the board. | √ | √ | √ | — | |

Note: RZ/G2E supports only one channel (n = 0).

49.1.4 Register Configuration

Table 49.4 shows the PCIE Controller registers and Table 49.5 shows the PCIE Controller phy registers. In these table, the Address Offset columns show an offset for each register from the PCIE Controller base address or the PCIE Controller phy base address. The actual access address is obtained by adding each offset to the PCIE Controller base address or the PCIE Controller phy base address. Table 49.2 shows the PCIE Controller base address and Table 49.3 shows the PCIE Controller phy base address. These registers should be accessed through the AXI bus. Only the little endian mode can be used for this module.

Table 49.2 PCIE Controller Base Address

| | PCI Express |
|--------------------|-------------|
| PCIE0 Base Address | H'FE00_0000 |
| PCIE1 BaseAddress | H'EE80_0000 |

Table 49.3 PCIE Controller Phy Base Address

| | PCI Express |
|------------------------|-------------|
| PCIE0 Phy Base Address | H'E65D_0000 |
| PCIE1 Phy Base Address | H'E65D_8000 |

Table 49.4 PCIE Controller Register Configuration

| Register Name | Abbreviation | AX*1 R/W | PCI*1 R/W | Address Offset | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|--|---------------|-------------|--------------|-------------------|---------------|----------------|---|----------------------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| PCIEC control registers | | | | | | | | | | |
| Configuration transmission address register | PCIECAR | R/W | — | H'0_0010 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Configuration transmission control register | PCIECCTLR | R/W | — | H'0_0018 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Configuration transmission data register | PCIECDR | R/W | — | H'0_0020 | — | 32 | √ | √ | √ | √ |
| Mode setting register | PCIEMSR | R/W | — | H'0_0028 | H'0000_0001 | 32 | √ | √ | √ | √ |
| Unlock transmission control register | PCIEUNLOCK CR | R/W | — | H'0_0048 | H'0000_0000 | 32 | √ | √ | √ | √ |
| INTx register | PCIEINTXR | R/W | — | H'0_0400 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Message reception status register | PCIERMSGR | R/W | — | H'0_0410 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Message reception interrupt enable register | PCIERMSGIER | R/W | — | H'0_0440 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Power management message transmission control register | PCIEPMMSG CR | R/W | — | H'0_0500 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Physical layer status register | PCIEPHYSR | R/W | — | H'0_07F0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MSI transmission register | PCIEMSITXR | R/W | — | H'0_0840 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC transfer control registers | | | | | | | | | | |
| Transfer control register | PCIETCTLR | R/W | — | H'0_2000 | H'0000_0008 | 32 | √ | √ | √ | √ |
| Transfer status register | PCIETSTR | R/W | — | H'0_2004 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Interrupt register | PCIEINTR | R/W | — | H'0_2008 | H'2000_0000 | 32 | √ | √ | √ | √ |
| Interrupt enable register | PCIEINTER | R/W | — | H'0_200C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Error factor register | PCIEERRFR | R/W | — | H'0_2020 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Error interrupt enable register | PCIEERRFER | R/W | — | H'0_2024 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Error factor register 2 | PCIEERRFR2 | R/W | — | H'0_2028 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Transfer interrupt enable register | PCIETIER | R/W | — | H'0_2030 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Power management state status register | PCIEPMSR | R/W | — | H'0_2034 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Power management state interrupt enable register | PCIEPMSCIER | R/W | — | H'0_2038 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MSIF register | PCIEMSIFR | R/W | — | H'0_2044 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MSI address lower register | PCIEMSIALR | R/W | — | H'0_2048 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MSI address upper register | PCIEMSIAUR | R/W | — | H'0_204C | H'0000_0000 | 32 | √ | √ | √ | √ |

| Register Name | Abbreviation | AX*1 R/W | PCI*1 R/W | Address Offset | Initial Value | Access Size | Second Generation RZ/G Series Products | | | | |
|--------------------------------------|--------------|-------------|--------------|-------------------|---------------|----------------|---|----------------------------|--------|--------|--|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| MSI interrupt enable register | PCIEMSIER | R/W | — | H'0_2050 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI EC Root Port address register 0 | PCIEPRAR0 | R/W | — | H'0_2080 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI EC Root Port address register 1 | PCIEPRAR1 | R/W | — | H'0_2084 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI EC Root Port address register 2 | PCIEPRAR2 | R/W | — | H'0_2088 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI EC Root Port address register 3 | PCIEPRAR3 | R/W | — | H'0_208C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI EC Root Port address register 4 | PCIEPRAR4 | R/W | — | H'0_2090 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI EC Root Port address register 5 | PCIEPRAR5 | R/W | — | H'0_2094 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Local address register 0 | PCIELAR0 | R/W | — | H'0_2200 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Local address mask register 0 | PCIELAMR0 | R/W | — | H'0_2208 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Local address register 1 | PCIELAR1 | R/W | — | H'0_2220 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Local address mask register 1 | PCIELAMR1 | R/W | — | H'0_2228 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Local address register 2 | PCIELAR2 | R/W | — | H'0_2240 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Local address mask register 2 | PCIELAMR2 | R/W | — | H'0_2248 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Local address register 3 | PCIELAR3 | R/W | — | H'0_2260 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Local address mask register 3 | PCIELAMR3 | R/W | — | H'0_2268 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Local address register 4 | PCIELAR4 | R/W | — | H'0_2280 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Local address mask register 4 | PCIELAMR4 | R/W | — | H'0_2288 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Local address register 5 | PCIELAR5 | R/W | — | H'0_22A0 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Local address mask register 5 | PCIELAMR5 | R/W | — | H'0_22A8 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI EC address lower register 0 | PCIEPALR0 | R/W | — | H'0_3400 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI EC address upper register 0 | PCIEPAUR0 | R/W | — | H'0_3404 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI EC address mask register 0 | PCIEPAMR0 | R/W | — | H'0_3408 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI EC conversion control register 0 | PCIEPTCLR0 | R/W | — | H'0_340C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI EC address lower register 1 | PCIEPALR1 | R/W | — | H'0_3420 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI EC address upper register 1 | PCIEPAUR1 | R/W | — | H'0_3424 | H'0000_0000 | 32 | √ | √ | √ | √ | |

| Register Name | Abbreviation | AX*1 R/W | PCI*1 R/W | Address Offset | Initial Value | Access Size | Second Generation RZ/G Series Products | | | | |
|---|---------------|-------------|--------------|-------------------|---------------|----------------|---|----------------------------|--------|--------|--|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| PCIEC address mask register 1 | PCIEPAMR1 | R/W | — | H'0_3428 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC conversion control register 1 | PCIEPTCTLR 1 | R/W | — | H'0_342C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC address lower register 2 | PCIEPALR2 | R/W | — | H'0_3440 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC address upper register 2 | PCIEPAUR2 | R/W | — | H'0_3444 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC address mask register 2 | PCIEPAMR2 | R/W | — | H'0_3448 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC conversion control register 2 | PCIEPTCTLR 2 | R/W | — | H'0_344C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC address lower register 3 | PCIEPALR3 | R/W | — | H'0_3460 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC address upper register 3 | PCIEPAUR3 | R/W | — | H'0_3464 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC address mask register 3 | PCIEPAMR3 | R/W | — | H'0_3468 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC conversion control register 3 | PCIEPTCTLR 3 | R/W | — | H'0_346C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMA control registers | | | | | | | | | | | |
| PCIEC DMAC DMA operation register | PCIEDMAOR | R/W | — | H'0_4000 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC PCI Express address lower register 0 | PCIEDMPALR 0 | R/W | — | H'0_4100 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC PCI Express address upper register 0 | PCIEDMPAUR 0 | R/W | — | H'0_4104 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC internal bus address register 0 | PCIEDMIAR0 | R/W | — | H'0_4108 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC byte count register 0 | PCIEDMBCNT R0 | R/W | — | H'0_4110 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC command chain address register 0 | PCIEDMCCAR 0 | R/W | — | H'0_4120 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel control register 0 | PCIEDMCHC R0 | R/W | — | H'0_4128 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel status register 0 | PCIEDMCHSR 0 | R/W | — | H'0_412C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel control 2 register 0 | PCIEDMCHC2 R0 | R/W | — | H'0_4130 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC PCI Express address lower register 1 | PCIEDMPALR 1 | R/W | — | H'0_4140 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC PCI Express address upper register 1 | PCIEDMPAUR 1 | R/W | — | H'0_4144 | H'0000_0000 | 32 | √ | √ | √ | √ | |

| Register Name | Abbreviation | AX*1 R/W | PCI*1 R/W | Address Offset | Initial Value | Access Size | Second Generation RZ/G Series Products | | | | |
|---|---------------|-------------|--------------|-------------------|---------------|----------------|---|----------------------------|--------|--------|--|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| PCIEC DMAC internal bus address register 1 | PCIEDMIAR1 | R/W | — | H'0_4148 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC byte count register 1 | PCIEDMBCNT R1 | R/W | — | H'0_4150 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC command chain address register 1 | PCIEDMCCAR 1 | R/W | — | H'0_4160 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel control register 1 | PCIEDMCHC R1 | R/W | — | H'0_4168 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel status register 1 | PCIEDMCHSR 1 | R/W | — | H'0_416C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel control 2 register 1 | PCIEDMCHC2 R1 | R/W | — | H'0_4170 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC PCI Express address lower register 2 | PCIEDMPALR 2 | R/W | — | H'0_4180 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC PCI Express address upper register 2 | PCIEDMPAUR 2 | R/W | — | H'0_4184 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC internal bus address register 2 | PCIEDMIAR2 | R/W | — | H'0_4188 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC byte count register 2 | PCIEDMBCNT R2 | R/W | — | H'0_4190 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC command chain address register 2 | PCIEDMCCAR 2 | R/W | — | H'0_41A0 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel control register 2 | PCIEDMCHC R2 | R/W | — | H'0_41A8 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel status register 2 | PCIEDMCHSR 2 | R/W | — | H'0_41AC | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel control 2 register 2 | PCIEDMCHC2 R2 | R/W | — | H'0_41B0 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC PCI Express address lower register 3 | PCIEDMPALR 3 | R/W | — | H'0_41C0 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC PCI Express address upper register 3 | PCIEDMPAUR 3 | R/W | — | H'0_41C4 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC internal bus address register 3 | PCIEDMIAR3 | R/W | — | H'0_41C8 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC byte count register 3 | PCIEDMBCNT R3 | R/W | — | H'0_41D0 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC command chain address register 3 | PCIEDMCCAR 3 | R/W | — | H'0_41E0 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel control register 3 | PCIEDMCHC R3 | R/W | — | H'0_41E8 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel status register 3 | PCIEDMCHSR 3 | R/W | — | H'0_41EC | H'0000_0000 | 32 | √ | √ | √ | √ | |

| Register Name | Abbreviation | AX*1 R/W | PCI*1 R/W | Address Offset | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|---------------|-------------|--------------|-------------------|---------------|----------------|---|----------------------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| PCIEC DMAC channel control 2 register 3 | PCIEDMCHC2 R3 | R/W | — | H'0_41F0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC PCI Express address lower register 4 | PCIEDMPALR 4 | R/W | — | H'0_4200 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC PCI Express address upper register 4 | PCIEDMPAUR 4 | R/W | — | H'0_4204 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC internal bus address register 4 | PCIEDMIAR4 | R/W | — | H'0_4208 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC byte count register 4 | PCIEDMBCNT R4 | R/W | — | H'0_4210 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC command chain address register 4 | PCIEDMCCAR 4 | R/W | — | H'0_4220 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC channel control register 4 | PCIEDMCHC R4 | R/W | — | H'0_4228 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC channel status register 4 | PCIEDMCHSR 4 | R/W | — | H'0_422C | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC channel control 2 register 4 | PCIEDMCHC2 R4 | R/W | — | H'0_4230 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC PCI Express address lower register 5 | PCIEDMPALR 5 | R/W | — | H'0_4240 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC PCI Express address upper register 5 | PCIEDMPAUR 5 | R/W | — | H'0_4244 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC internal bus address register 5 | PCIEDMIAR5 | R/W | — | H'0_4248 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC byte count register 5 | PCIEDMBCNT R5 | R/W | — | H'0_4250 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC command chain address register 5 | PCIEDMCCAR 5 | R/W | — | H'0_4260 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC channel control register 5 | PCIEDMCHC R5 | R/W | — | H'0_4268 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC channel status register 5 | PCIEDMCHSR 5 | R/W | — | H'0_426C | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC channel control 2 register 5 | PCIEDMCHC2 R5 | R/W | — | H'0_4270 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC PCI Express address lower register 6 | PCIEDMPALR 6 | R/W | — | H'0_4280 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC PCI Express address upper register 6 | PCIEDMPAUR 6 | R/W | — | H'0_4284 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC DMAC internal bus address register 6 | PCIEDMIAR6 | R/W | — | H'0_4288 | H'0000_0000 | 32 | √ | √ | √ | √ |

| | | | | | | | Second Generation RZ/G Series Products | | | | |
|---|---------------|--|--|-------------------|---|----------------|---|----------------------------|--------|--------|--|
| Register Name | Abbreviation | AX*1 R/W | PCI*1 R/W | Address Offset | Initial Value | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| PCIEC DMAC byte count register 6 | PCIEDMBCNT R6 | R/W | — | H'0_4290 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC command chain address register 6 | PCIEDMCCAR 6 | R/W | — | H'0_42A0 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel control register 6 | PCIEDMCHC R6 | R/W | — | H'0_42A8 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel status register 6 | PCIEDMCHSR 6 | R/W | — | H'0_42AC | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel control 2 register 6 | PCIEDMCHC2 R6 | R/W | — | H'0_42B0 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC PCI Express address lower register 7 | PCIEDMPALR 7 | R/W | — | H'0_42C0 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC PCI Express address upper register 7 | PCIEDMPAUR 7 | R/W | — | H'0_42C4 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC internal bus address register 7 | PCIEDMIAR7 | R/W | — | H'0_42C8 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC byte count register 7 | PCIEDMBCNT R7 | R/W | — | H'0_42D0 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC command chain address register 7 | PCIEDMCCAR 7 | R/W | — | H'0_42E0 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel control register 7 | PCIEDMCHC R7 | R/W | — | H'0_42E8 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel status register 7 | PCIEDMCHSR 7 | R/W | — | H'0_42EC | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCIEC DMAC channel control 2 register 7 | PCIEDMCHC2 R7 | R/W | — | H'0_42F0 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Configuration registers | | | | | | | | | | | |
| PCI configuration register 0 | PCICONF0 | R | R | H'1_0000 | H'0025_1912 | 32 | √ | √ | √ | √ | |
| PCI configuration register 1 | PCICONF1 | R/W | R/W | H'1_0004 | H'0010_0000 | 32 | √ | √ | √ | √ | |
| PCI configuration register 2 | PCICONF2 | R | R | H'1_0008 | H'FF00_0000 | 32 | √ | √ | √ | √ | |
| PCI configuration register 3 | PCICONF3 | R/W | R/W | H'1_000C | H'0001_0000 | 32 | √ | √ | √ | √ | |
| PCI configuration register 4 | PCICONF4 | *2 | *3 | H'1_0010 | *2 | 32 | √ | √ | √ | √ | |
| PCI configuration register 5 | PCICONF5 | *4 | *3 | H'1_0014 | *4 | 32 | √ | √ | √ | √ | |
| PCI configuration register 6 | PCICONF6 | Header TYPE00 :*5 Header TYPE01 : R/W | Header TYPE00: *3 Header TYPE01 : — | H'1_0018 | Header TYPE00:*5 Header TYPE01: H'0000_0000 | 32 | √ | √ | √ | √ | |

| Register Name | Abbreviation | AX*1 R/W | PCI*1 R/W | Address Offset | Initial Value | Access Size | Second Generation RZ/G Series Products | | | | |
|----------------------------------|--------------|--|--|-------------------|--|----------------|---|----------------------------|--------|--------|--|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| PCI configuration register 7 | PCICONF7 | Header TYPE00 :*6 Header TYPE01 : R/W | Header TYPE00: *3 Header TYPE01: — | H'1_001C | Header TYPE00:*6 Header TYPE01: H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI configuration register 8 | PCICONF8 | Header TYPE00 :*7 Header TYPE01 : R/W | Header TYPE00: *3 Header TYPE01: — | H'1_0020 | Header TYPE00:*7 Header TYPE01: H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI configuration register 9 | PCICONF9 | Header TYPE00 :*8 Header TYPE01 : R/W | Header TYPE00: *3 Header TYPE01: — | H'1_0024 | Header TYPE00:*8 Header TYPE01: H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI configuration register 10 | PCICONF10 | Header TYPE00 : R Header TYPE01 : R/W | Header TYPE00 : R Header TYPE01 : — | H'1_0028 | Header TYPE00: H'0000_0000 Header TYPE01: H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI configuration register 11 | PCICONF11 | Header TYPE00 : R Header TYPE01 : R/W | Header TYPE00 : R Header TYPE01 : — | H'1_002C | Header TYPE00: H'0000_0000 Header TYPE01: H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI configuration register 12 | PCICONF12 | Header TYPE00 : R Header TYPE01 : R/W | Header TYPE00 : R Header TYPE01 : — | H'1_0030 | Header TYPE00: H'0000_0000 Header TYPE01: H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI configuration register 13 | PCICONF13 | R/W | R | H'1_0034 | H'0000_0040 | 32 | √ | √ | √ | √ | |
| PCI configuration register 14 | PCICONF14 | Header TYPE00 : R Header TYPE01 : R | Header TYPE00 : R Header TYPE01 : — | H'1_0038 | Header TYPE00: H'0000_0000 Header TYPE01: H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI configuration register 15 | PCICONF15 | Header TYPE00 : R/W Header TYPE01 : R/W | Header TYPE00 : R/W Header TYPE01 : — | H'1_003C | Header TYPE00: H'0000_00FF Header TYPE01: H'0000_00FF | 32 | √ | √ | √ | √ | |

| Register Name | Abbreviation | AX*1 R/W | PCI*1 R/W | Address Offset | Initial Value | Access Size | Second Generation RZ/G Series Products | | | | |
|--|--------------|--|--|-------------------|--|----------------|---|----------------------------|--------|--------|--|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| PCI power management capability register 0 | PMCAP0 | R/W | R | H'1_0040 | H'C803_5001 | 32 | √ | √ | √ | √ | |
| PCI power management capability register 1 | PMCAP1 | R/W | R/W | H'1_0044 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| MSI capability register 0 | MSICAP0 | R/W | R/W | H'1_0050 | H'0180_7005 | 32 | √ | √ | √ | √ | |
| MSI capability register 1 | MSICAP1 | R/W | R/W | H'1_0054 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| MSI capability register 2 | MSICAP2 | R/W | R/W | H'1_0058 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| MSI capability register 3 | MSICAP3 | R/W | R/W | H'1_005C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| MSI capability register 4 | MSICAP4 | R/W | R/W | H'1_0060 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| MSI capability register 5 | MSICAP5 | R | R | H'1_0064 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI Express capability register 0 PCI Express capability list register PCI Express capabilities register | EXPCAP0 | R/W | R | H'1_0070 | H'0042_0010 | 32 | √ | √ | √ | √ | |
| PCI Express capability register 1 Device capabilities register | EXPCAP1 | R/W | R | H'1_0074 | H'0000_8020 | 32 | √ | √ | √ | √ | |
| CI Express capability register 2 Device control register Device status register | EXPCAP2 | R/W | R/W | H'1_0078 | H'0000_2810 | 32 | √ | √ | √ | √ | |
| PCI Express capability register 3 Link capabilities register | EXPCAP3 | R/W | R | H'1_007C | H'0003_F412 | 32 | √ | √ | √ | √ | |
| PCI Express capability register 4 Link control register Link status register | EXPCAP4 | R/W | R/W | H'1_0080 | H'0041_0000 | 32 | √ | √ | √ | √ | |
| PCI Express capability register 5 Slot capabilities register | EXPCAP5 | Header TYPE00 : R Header TYPE01 : R | Header TYPE00 : R Header TYPE01 : — | H'1_0084 | Header TYPE00: H'0000_0000 Header TYPE01: H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI Express capability register 6 Slot control register Slot status register | EXPCAP6 | Header TYPE00 : R Header TYPE01 : R | Header TYPE00 : R Header TYPE01 : — | H'1_0088 | Header TYPE00: H'0000_0000 Header TYPE01: H'0040_0000 | 32 | √ | √ | √ | √ | |

| Register Name | Abbreviation | AX*1 R/W | PCI*1 R/W | Address Offset | Initial Value | Access Size | Second Generation RZ/G Series Products | | | | |
|---|--------------|-------------------------|-------------------------|-------------------|----------------------------------|----------------|---|----------------------------|--------|--------|--|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| PCI Express capability register 7 Root control register Root capabilities register | EXPCAP7 | Header TYPE00 : R | Header TYPE00 : R | H'1_008C | Header TYPE00: H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI Express capability register 8 Root status register | EXPCAP8 | Header TYPE00 : R | Header TYPE00 : R | H'1_0090 | Header TYPE00: H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI Express capability register 9 Device capabilities 2 register | EXPCAP9 | R/W | R | H'1_0094 | H'0000_0010 | 32 | √ | √ | √ | √ | |
| PCI Express capability register 10 Device control 2 register Device status 2 register | EXPCAP10 | R/W | R/W | H'1_0098 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI Express capability register 11 Link capabilities 2 register | EXPCAP11 | R | R | H'1_009C | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI Express capability register 12 Link control 2 register Link status 2 register | EXPCAP12 | R/W | R/W | H'1_00A0 | H'0001_0002 | 32 | √ | √ | √ | √ | |
| PCI Express capability register 13 Slot capabilities 2 register | EXPCAP13 | R | R | H'1_00A4 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| PCI Express capability register 14 Slot control 2 register Slot status 2 register | EXPCAP14 | R | R | H'1_00A8 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| VC capability register 0 Virtual channel enhanced capability header | VCCAP0 | R/W | R | H'1_0100 | H'1B01_0002 | 32 | √ | √ | √ | √ | |
| VC capability register 1 Port VC capability register 1 | VCCAP1 | R/W | R | H'1_0104 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| VC capability register 2 Port VC capability register 2 | VCCAP2 | R | R | H'1_0108 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| VC capability register 3 Port VC status register Port VC control register | VCCAP3 | R | R | H'1_010C | H'0000_0000 | 32 | √ | √ | √ | √ | |

| Register Name | Abbreviation | AX*1 R/W | PCI*1 R/W | Address Offset | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|-------------------|-------------|--------------|-------------------|---------------|----------------|---|----------------------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| VC capability register 4 VC0 resource capability register | VCCAP4 | R/W | R | H'1_0110 | H'0000_0000 | 32 | √ | √ | √ | √ |
| VC capability register 5 VC0 resource control register | VCCAP5 | R/W | R/W | H'1_0114 | H'8000_00FF | 32 | √ | √ | √ | √ |
| VC capability register 6 VC0 resource status register | VCCAP6 | R | R | H'1_0118 | H'0002_0000 | 32 | √ | √ | √ | √ |
| Device serial number capability register 0 | SERNUMCAP 0 | R/W | R | H'1_01B0 | H'0001_0003 | 32 | √ | √ | √ | √ |
| Device serial number capability register 1 | SERNUMCAP 1 | R | R | H'1_01B4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Device serial number capability register 2 | SERNUMCAP 2 | R | R | H'1_01B8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PCIEC link layer control registers | | | | | | | | | | |
| ID setting register 0 | IDSETR0 | R/W | — | H'1_1000 | H'0025_1912 | 32 | √ | √ | √ | √ |
| ID setting register 1 | IDSETR1 | R/W | — | H'1_1004 | H'FF00_0000 | 32 | √ | √ | √ | √ |
| SUBID setting register | SUBIDSETR | R/W | — | H'1_1024 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Device serial number setting register 0 | DSERSETR0 | R/W | — | H'1_102C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Device serial number setting register 1 | DSERSETR1 | R/W | — | H'1_1030 | H'0000_0000 | 32 | √ | √ | √ | √ |
| TL control register | TLCTLR | R/W | — | H'1_1048 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MAC status register | MACSR | R/W | — | H'1_1054 | H'0041_0000 | 32 | √ | √ | √ | √ |
| MAC control register | MACCTLR | R/W | — | H'1_1058 | H'80FF_0001 | 32 | √ | √ | √ | √ |
| PM status register | PMSR | R/W | — | H'1_105C | H'0000_0000 | 32 | √ | √ | √ | √ |
| PM control register | PMCTLR | R/W | — | H'1_1060 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MAC interrupt enable register | MACINTENR | R/W | — | H'1_106C | H'0000_0000 | 32 | √ | √ | √ | √ |
| PM interrupt enable register | PMINTENR | R/W | — | H'1_1070 | H'0000_0000 | 32 | √ | √ | √ | √ |
| MAC status register 2 | MACS2R | R | — | H'1_1078 | H'0041_0001 | 32 | √ | √ | √ | √ |
| MAC control register 2 | MACCTL2R | R/W | — | H'1_107C | H'0000_0002 | 32 | √ | √ | √ | √ |
| MAC speed change setting register | MACCGSPSE TR | R/W | — | H'1_1084 | H'0002_0000 | 32 | √ | √ | √ | √ |
| AXI cache control register | AXI_CACHE_ REG | R/W | — | H'1_1200 | H'8000_0000 | 32 | √ | √ | — | √ |

Notes: 1. AX: AXI bus (internal bus)

PCI: PCI local bus

'—' in R/W column: Access prohibited.

For details of R/W attribute for each bit, see section 49.2, Register Description.

2. Depends on the setting of the local address mask register 0.

3. R/W changes by the state. RootPort:R/W.EndPoint:R.

4. Depends on the setting of the local address mask register 1.
5. Depends on the setting of the local address mask register 2.
6. Depends on the setting of the local address mask register 3.
7. Depends on the setting of the local address mask register 4.
8. Depends on the setting of the local address mask register 5.

Do not write to any addresses other than listed above. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed above are undefined.

Table 49.5 PCIE Controller Phy Register Configuration

| Register Name | Abbreviation | AX*1 R/W | PCI*1 R/W | Address Offset | Initial Value | Access Size | Second Generation RZ/G Series Products | | | | |
|----------------------------|--------------|-------------|--------------|-------------------|---------------|----------------|---|----------------------------|--------|--------|--|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| PCIEC Phy registers | | | | | | | | | | | |
| PHY clock register | PHY_CLK | R/W | — | H' 4004 | H'0000_0119 | 32 | √ | √ | √ | — | |

49.1.5 Connected Module

Table 49.6 Connected modules

| Module name | Connected module name | Function of connected module |
|-------------|-----------------------|------------------------------|
| PCIEC | AP-System Core | Access the Registers |
| | CPG | Output clocks |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |
| | INTC-AP | Control to interrupt |

49.2 Register Description

[Legend for the Register Description]

A bit assignment figure is shown for each register. The initial value and R/W attribute are indicated for each bit.

Bit: Bit number or bit range

Bit Name: Bit name or field name

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

RW1C: Readable/writable. The bit is initialized when 1 is written. Writing 0 is ignored.

R: Read-only. The write value should always be 0. (If there is a direction on a read or write value in the Description column, set the bit as directed.)

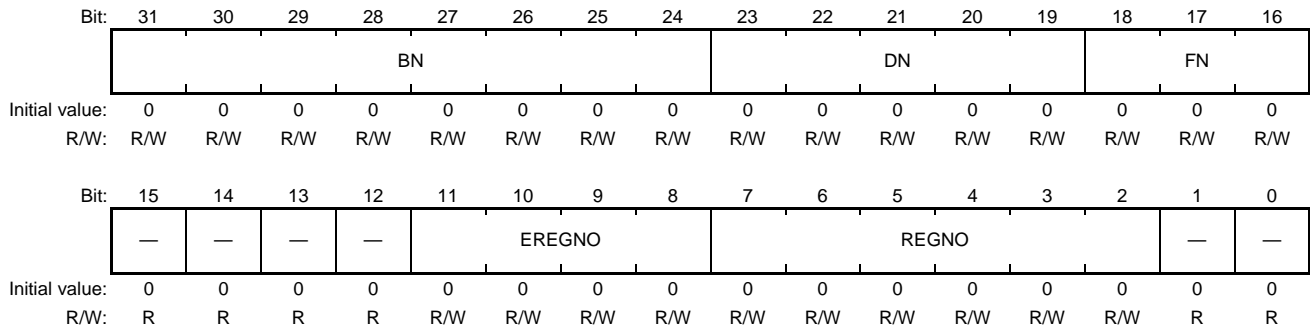
RWS*: Sticky-read write bit. Readable/writable. The bit is not initialized by a hot reset.

RW1CS*: Sticky-read-only status bit. The bit is initialized when 1 is written. Writing 0 is ignored. The bit is not initialized by a hot reset.

Note: * RWS and RW1CS are only used for the configuration registers. For details, see section 49.4.1, Write/Read Attributes of Configuration Registers.

49.2.1 Configuration Transmission Address Register (PCIECAR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 24 | BN | H'00 | R/W | Bus Number Specifies the number of the bus to be accessed when a configuration request is issued. |
| 23 to 19 | DN | B'0_0000 | R/W | Device Number Specifies the number of the device to be accessed when a configuration request is issued. Note: A type 0 configuration request can be issued only to device #0; do not send a type 0 configuration request to the device with the other number. |
| 18 to 16 | FN | B'000 | R/W | Function Number Specifies the number of the function to be accessed when a configuration request is issued. |
| 15 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 8 | EREGNO | H'0 | R/W | Extended Register No. Specifies the number of the extended register to be accessed when a configuration request is issued. |
| 7 to 2 | REGNO | H'00 | R/W | Register No. Specifies the number of the register to be accessed when a configuration request is issued. |
| 1, 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.2 Configuration Transmission Control Register (PCIECCTLR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

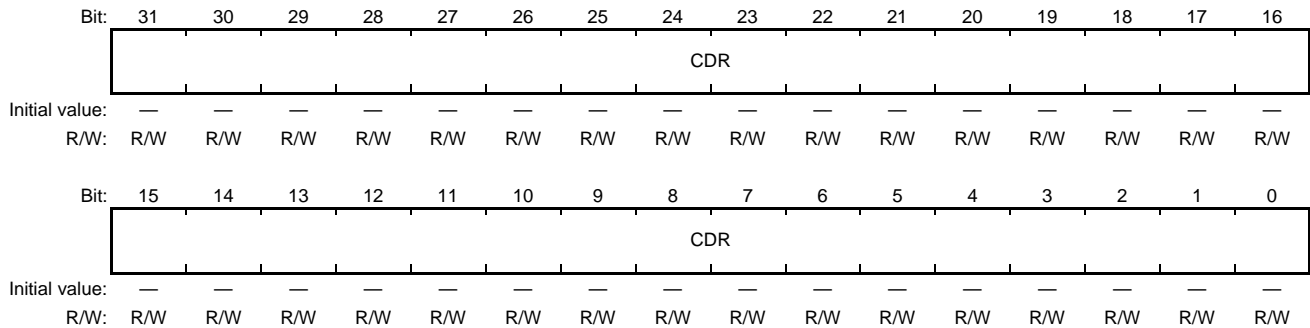
| | | | | | | | | | | | | | | | | |
|----------------|------|----|----|----|----|----|----|------|----|----|------|------|----|----|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CCIE | — | — | — | — | — | — | — | — | — | WUR | WCRS | — | — | RUR | RCRS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R | R | RW1C | RW1C | R | R | RW1C | RW1C |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | TYPE | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|------|---|
| 31 | CCIE | B'0 | R/W | Configuration Send Enable Enables issuance of a configuration request. Set to 1 when the configuration transmission address register and configuration transmission data register are used to issue the configuration cycle. |
| 30 to 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 | WUR | B'0 | RW1C | Write Unsupported Request Indicates that an unsupported request has been received. This bit is set to 1 when a completion with the unsupported request status is received in response to the configuration write request transmitted. |
| 20 | WCRS | B'0 | RW1C | Write CRS Indicates that CRS (configuration retry status) has been received. This bit is set to 1 when a completion with the CRS status is received in response to the configuration write request transmitted. |
| 19, 18 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 17 | RUR | B'0 | RW1C | Read Unsupported Request Indicates that an unsupported request has been received. This bit is set to 1 when a completion with the unsupported request status is received in response to the configuration read request transmitted. |
| 16 | RCRS | B'0 | RW1C | Read CRS Indicates that CRS (configuration retry status) has been received. This bit is set to 1 when a Completion with the CRS status is received in response to the configuration read request transmitted. |
| 15 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 8 | TYPE | B'0 | R/W | TYPE Specifies the type of a configuration request issued by accessing the PCIECDR register. 0: Type 0 configuration request 1: Type 1 configuration request Note: A type 0 configuration request can be issued only to device #0; do not send a type 0 configuration request to the device with the other number. |
| 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.3 Configuration Transmission Data Register (PCIECDR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | CDR | Undefined | R/W | <p>Configuration Data Register</p> <p>Allows a configuration request to be issued.</p> <p>A write access to this field allows issuance of a configuration write request containing the information written to this field as data.</p> <p>A read access to this field allows issuance of a configuration read request and reading of the data contained in the response.</p> <p>When the peer device returns an unsupported request response, the read value is H'FFFF_FFFF.</p> <p>When EXPCAP7[4].CRSVISE = 0:</p> <p>When the peer device returns a CRS response, the read value is H'FFFF_FFFF (error response).</p> <p>When EXPCAP7[4].CRSVISE = 1:</p> <p>Until the peer device returns a response other than a CRS response, the first access to the peer device should be a read access to register address 0.</p> <p>When the peer device returns a CRS response, the read value is H'FFFF_0001.</p> <p>In Endpoint mode, this field must not be accessed.</p> <p>Except when PCIECTLR.CCIE = 1 and PCIETCTLR.CFINIT = 1, the read value is H'FFFF_FFFF.</p> |

49.2.4 Mode Setting Register (PCIEMSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | PEM |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | PEM | B'1 | R/W | PCI Express Mode Sets operating mode of the PCI Express. 0: Operates as a PCI Express Endpoint. 1: Operates as a PCI Express Root Port. |

49.2.5 Unlock Transmission Control Register (PCIEUNLOCKCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ASTUNLOCK |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | ASTUNLOCK | B'0 | R/W | Assert Unlock Specifies issuance of an unlock message. Writing 1 to this bit allows issuance of an unlock message (message code = B'0000_0000; routing = B'011). Transmission of an unlock message through a write access to this bit can be used only in Root Port mode. Transmission of an unlock message from an Endpoint is prohibited by the standard. While a link is not established, do not write 1 to this bit. This bit is always read as 0. |

49.2.6 INTx Register (PCIEINTXR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-------|-------|-------|-------|----|----|----|----|------|------|------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ASTINT X |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | INTDE | INTCE | INTBE | INTAE | — | — | — | — | INTD | INTC | INTB | INTA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | ASTINTX | B'0 | R/W | Assert INTx Specifies generation of an INTx interrupt. Writing 1 to this bit asserts the INTx interrupt specified by the PCICONF15[15:8].Interrupt Pin bits and writing 0 deasserts the INTx interrupt. When generating an INTx interrupt using this bit, the PCICONF1[10].INTDIS bit must be 0. This bit is cleared if the PCICONF1[10].Interrupt Disable bit is set to 1 while this bit is 1. When MSI is used, an INTx interrupt cannot be used. Therefore, when the MSICAP0[16].MSI Enable bit is 1, 1 cannot be written to this bit. This bit is cleared if the MSICAP0[16].MSI Enable bit is set to 1 while this bit is 1. This bit is valid only in Endpoint mode. When a link goes down, this bit is cleared. A write access is ignored while a link is down. |
| 15 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 | INTDE | B'0 | R/W | INTD Enable Enables generation of an interrupt by INTD assertion. This bit is valid only in Root Port mode. |
| 10 | INTCE | B'0 | R/W | INTC Enable Enables generation of an interrupt by INTC assertion. This bit is valid only in Root Port mode. |
| 9 | INTBE | B'0 | R/W | INTB Enable Enables generation of an interrupt by INTB assertion. This bit is valid only in Root Port mode. |
| 8 | INTAE | B'0 | R/W | INTA Enable Enables generation of an interrupt by INTA assertion. This bit is valid only in Root Port mode. |
| 7 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | INTD | B'0 | R | Indicates that INTD has been asserted by a PCI Express interrupt. This bit is asserted upon reception of an Assert_INTD message and deasserted upon reception of a Deassert_INTD message. When this bit is asserted while PCICONF15[7:0].Interrupt Line is not H'FF and PCIEINTXR[11].INTDE is 1, the interrupt is generated. This bit is valid only in Root Port mode. |
| 2 | INTC | B'0 | R | Indicates that INTC has been asserted by a PCI Express interrupt. This bit is asserted upon reception of an Assert_INTC message and deasserted upon reception of a Deassert_INTC message. When this bit is asserted while PCICONF15[7:0].Interrupt Line is not H'FF and PCIEINTXR[10].INTCE is 1, the interrupt is generated. This bit is valid only in Root Port mode. |
| 1 | INTB | B'0 | R | Indicates that INTB has been asserted by a PCI Express interrupt. This bit is asserted upon reception of an Assert_INTB message and deasserted upon reception of a Deassert_INTB message. When this bit is asserted while PCICONF15[7:0].Interrupt Line is not H'FF and PCIEINTXR[9].INTBE is 1, the interrupt is generated. This bit is valid only in Root Port mode. |
| 0 | INTA | B'0 | R | Indicates that INTA has been asserted by a PCI Express interrupt. This bit is asserted upon reception of an Assert_INTA message and deasserted upon reception of a Deassert_INTA message. When this bit is asserted while PCICONF15[7:0].Interrupt Line is not H'FF and PCIEINTXR[8].INTAE is 1, the interrupt is generated. This bit is valid only in Root Port mode. |

49.2.7 Message Reception Status Register (PCIERMSGR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|------------|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | SLOT_POWER | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | RW1C | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|------|--|
| 31 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | SLOT_POWER | B'0 | RW1C | SLOT_POWER Indicates that the Set_Slot_Power_Limit message has been received. This bit is valid only in Endpoint mode. |
| 11 to 0 | — | All 0 | R | Reserved Ignore the value in this field. The read value is undefined. The write value should always be 0. |

49.2.8 Message Reception Interrupt Enable Register (PCIERMSGIER)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----------------|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | SLOT_P OWERE | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | SLOT_POWERE | B'0 | R/W | SLOT_POWERE Enables generation of an interrupt upon reception of a Set_Slot_Power_Limit message. |
| 11 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.9 Power Management Message Transmission Control Register (PCIEPMMMSGCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|--------|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | PM_PME | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | PM_PME | B'0 | R/W | Specifies issuance of a PM_PME message. Writing 1 to this bit allows issuance of a PM_PME Message (message code = B'0001 1000; routing = B'000). This bit is always read as 0. Transmission of a PM_PME message through a write access to this bit can be used only in Endpoint mode. Transmission of a PM_PME message from a Root Port is prohibited by the standard. While a link is not established, do not write 1 to this bit. |
| 8 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.10 Physical Layer Status Register (PCIEPHYSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|--------------|----|----|----|----|----|----|----|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | PHYRD YCE | — | — | — | — | — | — | — | PHYRD YC |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | RW1C |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | PHYRD Y |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|------|--|
| 31 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | PHYRDYCE | B'0 | R/W | PHYRDY Change Enable Enables generation of an INT_PCISC interrupt upon change of PHYRDY. |
| 23 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | PHYRDYC | B'0 | RW1C | PHYRDY Change Indicates that PHYRDY has changed. |
| 15 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | PHYRDY | B'0 | R | 0: Indicates that the clock supplied by the physical layer has not been stabilized and that it is not ready to be used. 1: Indicates that the clock supplied by the physical layer has been stabilized and that it is ready to be used. |

49.2.11 MSI Transmission Register (PCIEMSITXR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------|----|----|----|----|----|----|----|----|----|----|--------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MSIE | — | — | — | — | — | — | — | — | — | — | MMENUM | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MSIPC LR | — | — | — | — | — | — | — | — | — | — | MSIAST | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 | MSIE | B'0 | R | MSI Enable Indicates the MSICAP0[16]. MSIE value. |
| 30 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 to 16 | MMENUM | B'0_0000 | R | Multiple Message Enable Number Indicates the number of sources specified with MSICAP0[22:20].MMESE. B'0_0000: 1 source (only source number 0 is available) B'0_0001: 2 sources (source numbers 0 and 1 are available) B'0_0011: 4 sources (source numbers 0 to 3 are available) B'0_0111: 8 sources (source numbers 0 to 7 are available) B'0_1111: 16 sources (source numbers 0 to 15 are available) B'1_1111: 32 sources (source numbers 0 to 31 are available) |
| 15 | MSIPC LR | B'0 | R/W | MSI Pending Clear Forcibly clears the MSI interrupt sources to be transmitted. If the MSI source is canceled because of the user side reason (internal source), set the information of the MSI interrupt to be cleared to the MSI Assert field, and simultaneously write 1 to this bit. In the other cases, hold this bit to 0. This bit is always read as 0. |
| 14 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 4 to 0 | MSIAST | B'0_0000 | R/W | <p>MSI Assert</p> <p>Specifies generation of an MSI interrupt.</p> <p>This field is valid only when the MSI is enabled in Endpoint mode.</p> <p>A write access to this field allows setting of the relevant bits to 1 in MSICAP5.Message Pending. After that, among the MSI interrupts whose corresponding bits are 0 in MSICAP4.Message Mask, the MSI interrupts are allowed to be generated one by one in ascending order of the assigned number.</p> <p>The maximum source number that may be written to this field is indicated by the Multiple Message Enable Number field.</p> <p>These bits are always read as 0.</p> <p>While communication with the connection destination device is not established, do not write to this field.</p> |

49.2.12 Transfer Control Register (PCIETCLR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|---------|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | DL_DOWN | — | — | CF INIT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 | DL_DOWN | B'1 | R | DL_DOWN Indicates whether connection in the data link layer is lost. This bit is set to 1 after cancellation of a reset. This bit is cleared to 0 when a connection with the target device is established, and set to 1 when a connection in the data link layer is lost due to communication failure. |
| 2, 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | CFINIT | B'0 | R/W | Configuration Initialization Specifies initialization of registers to establish a PCI Express link. Set this bit to 1 after setting the appropriate values in the following registers. PCIELAR0 to PCIELAR 5 PCIELAMR0 to PCIELAMR5 The above registers must not be written to while this bit is 1. Setting this bit starts PCI Express link establishment process. Only writing 1 to this bit is valid; writing 0 is invalid. Write 1 at initialization only once and do not write any value to this bit after that. 0: Initialization has not yet been done. 1: Initialization has been completed. In Endpoint mode: Triggered by this bit, hardware sets the BARn registers based on PCIELAMRn. |

49.2.13 Transfer Status Register (PCIETSTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|--------|----|----|----|----|----|----|----|----|----|----|----|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | MSIEC | — | — | — | — | — | — | — | — | — | — | — | DLLACTC |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | RW1C | R | R | R | R | R | R | R | R | R | R | R | RW1C |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | INTXDC | — | — | — | — | — | — | — | — | — | — | — | DLLACT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | RW1C | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|------|--|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | MSIEC | B'0 | RW1C | MSI Enable Change Indicates that MSICAP0[16].MSIE has changed. |
| 27 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | DLLACTC | B'0 | RW1C | DLLACT Change Indicates that DLLACT has changed. |
| 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | INTXDC | B'0 | RW1C | INTx Disable Change Indicates that PCICONF1[10].INTDIS has changed. |
| 11 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | DLLACT | B'0 | R | Data Link Layer Active Indicates that the data link layer is active. |

49.2.14 Interrupt Register (PCIEINTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----------------|--------------|-----------|----|----|----|----|----|----|-------------|---------------|--------------|---------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | INT_PCITE | INT_PM_PME_RCV | INT_TXALLEMP | INT_PCIBW | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | RW1C | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | INT_MAC | INT_PM | — | — | — | — | — | — | INT_PCI_MES | INT_PCI_POWER | INT_PCI_CERR | INT_PCI_NFERR | INT_PCI_FERR | INT_PCI_SERR |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | RW1C | * | * | * | R |

Note: * R/W changes by the state. Root Port: R, EndPoint: RW1C

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|------|---|
| 31 | INT_PCITE | B'0 | RW1C | <p>Interrupt PCI Express Transfer Error</p> <p>Indicates that the interrupt by a PCI Express transfer error has been generated.</p> <p>The error can be either of the following.</p> <ul style="list-style-type: none"> When PCIECTLR[31].CCIE = 0, PCIECDR is accessed. When PCIEPCTLRn[31].PARE = 0, the corresponding PIO space is accessed. Transfer is executed with TC being set that is not mapped to VC0. When PCICONF1[2].BME = 0 in Endpoint mode, PIO transfer is executed (MRd, MWr). Transfer is executed with the attribute being set that is not enabled with the configuration register. Transfer other than PME message (PM_PME) transfer is executed in the non-D0 state. When PMCAP1[8].PMEE = 0 in Endpoint mode, PM_PME message transfer is executed. |
| 30 | INT_PM_PME_RCV | B'0 | R | <p>Interrupt PM_PME Receive</p> <p>Generates an interrupt when PM_PME is received.</p> <p>This bit is valid only in Root Port mode. To clear the interrupt, write 1 to EXPCAP8[16].PMEST.</p> |
| 29 | INT_TXALLEMP | B'1 | R | <p>Interrupt TX All Empty</p> <p>Generates an interrupt when all the transmission buffers are empty for PCI Express.</p> |
| 28 | INT_PCIBW | B'0 | R | <p>Interrupt PCI Express Bandwidth</p> <p>Indicates that the interrupt by change of the link bandwidth (link width × link speed) has been generated.</p> <p>This bit is valid only in Root Port mode. To clear the interrupt, write 1 to EXPCAP4[31].LKAUTOBWSTS or EXPCAP4[30].LKBWMNGSTS.</p> |
| 27 to 14 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|------|---|
| 13 | INT_MAC | B'0 | R | <p>Interrupt MAC</p> <p>Indicates that the MAC interrupt has been generated.</p> <p>To clear the interrupt, write 1 to the corresponding field in MACSR.</p> |
| 12 | INT_PM | B'0 | R | <p>Interrupt PM</p> <p>Indicates that the PM interrupt has been generated.</p> <p>To clear the interrupt, write 1 to the corresponding field in PMSR.</p> |
| 11 to 6 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 5 | INT_PCIMES | B'0 | R | <p>Interrupt PCI Message</p> <p>Indicates that the interrupt has been generated upon reception of a PCI Express message.</p> <p>However, if the interrupt corresponding to individual message is not enabled, this interrupt is not generated; process the message through software when a message is received.</p> |
| 4 | INT_PCIPOWER | B'0 | RW1C | <p>Interrupt PCI Express Power</p> <p>Indicates that a configuration write request has been received in Endpoint mode thus generating a request for a power down sequence to the L1 state.</p> |
| 3 | INT_PCICERR | B'0 | R | <p>Interrupt PCI Correctable Error</p> <p>Root Port:</p> <p>Indicates that the interrupt has been generated by detection of the correctable error or reception of the ERR_COR message. This bit is a read-only bit; to clear this bit, use the configuration register.</p> |
| | | | RW1C | <p>Interrupt PCI Correctable Error</p> <p>Endpoint:</p> <p>Indicates that an error classified as ERR_COR has been generated. This bit is a read-write-1-to-clear bit; writing 1 clears this bit only.</p> |
| 2 | INT_PCINFERR | B'0 | R | <p>Interrupt PCI Non-Fatal Error</p> <p>Root Port:</p> <p>Indicates that the interrupt has been generated by detection of the non-fatal error or reception of the ERR_NONFATAL message. This bit is a read-only bit; to clear this bit, use the configuration register.</p> |
| | | | RW1C | <p>Interrupt PCI Non-Fatal Error</p> <p>Endpoint:</p> <p>Indicates that an error classified as ERR_NONFATAL has been generated. This bit is a read-write-1-to-clear bit; writing 1 clears this bit only.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-------------|---------------|------|---|
| 1 | INT_PCIFERR | B'0 | R | <p>Interrupt PCI Fatal Error</p> <p>Root Port:</p> <p>Indicates that the interrupt has been generated by detection of the fatal error or reception of the ERR_FATAL message. This bit is a read-only bit; to clear this bit, use the configuration register.</p> |
| | | | RW1C | <p>Interrupt PCI Fatal Error</p> <p>Endpoint:</p> <p>Indicates that an error classified as ERR_FATAL has been generated. This bit is a read-write-1-to-clear bit; writing 1 clears this bit only.</p> |
| 0 | INT_PCISERR | B'0 | R | <p>Interrupt PCI System Error</p> <p>Indicates that a system error has occurred.</p> <p>When a system error occurs, check the related registers through software and execute error recovery processing.</p> <p>A system error is:</p> <p>A PCIEERRFR error has occurred (when enabled).</p> |

49.2.15 Interrupt Enable Register (PCIEINTER)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----------------|---------------|------------|----|----|----|----|----|----|-------------|---------------|--------------|---------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | INT_PCITEE | INT_PM_PME_RCVE | INT_TXALLEMPE | INT_PCIBWE | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | INT_MACE | INT_PME | — | — | — | — | — | — | INT_PCIMESE | INT_PCIPOWERE | INT_PCICERRE | INT_PCINFERRE | INT_PCIFERRE | INT_PCISERRE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 | INT_PCITEE | B'0 | R/W | Interrupt PCI Express Transfer Error Enable Enables generation of an interrupt by a PCI Express transfer error. |
| 30 | INT_PM_PME_RCVE | B'0 | R/W | Interrupt PM_PME Receive Enable Enables generation of an interrupt by INT_PM_PME_RCV. |
| 29 | INT_TXALLEMPE | B'0 | R/W | Interrupt TX All Empty Enable Enables generation of an interrupt when all the transmission buffers are empty for PCI Express. |
| 28 | INT_PCIBWE | B'0 | R/W | Interrupt PCI Express Bandwidth Enable Enables generation of an interrupt by change of the link bandwidth. |
| 27 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 | INT_MACE | B'0 | R/W | Interrupt MAC Enable Enables generation of the MAC interrupt. |
| 12 | INT_PME | B'0 | R/W | Interrupt PM Enable Enables generation of the PM interrupt. |
| 11 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 | INT_PCIMESE | B'0 | R/W | Interrupt Message Enable Enables generation of the PCI Express message reception interrupt. |
| 4 | INT_PCIPOWERE | B'0 | R/W | Interrupt PCI Express Power Enable Enables generation of an interrupt by a request for a power down sequence to the L1 state. |
| 3 | INT_PCICERRE | B'0 | R/W | Interrupt Correctable Error Enable Enables generation of an interrupt by a correctable error. |
| 2 | INT_PCINFERRE | B'0 | R/W | Interrupt Non-Fatal Error Enable Enables generation of an interrupt by a non-fatal error. |
| 1 | INT_PCIFERRE | B'0 | R/W | Interrupt Fatal Error Enable Enables generation of an interrupt by a fatal error. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------------|-----------------|----------------------|------------|--|
| 0 | INT_PCISERRE | B'0 | R/W | Interrupt System Error Enable Enables generation of an interrupt by a system error. |

49.2.16 Error Factor Register (PCIEERRFR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|--------------|--------------|------|------|---------------|-------------|----|---------------|--------------|--------------|----|----|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | IBERR | MTLP | UR | PTLP | — | — | — | POVF | NPOVF | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | RW1C | RW1C | RW1C | RW1C | R | R | R | RW1C | RW1C | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | RCVCP LLK | UNEXP CPL | — | — | RCVSZ ECPL | CPLTO UT | — | RCVCR SCPL | RCVCA CPL | RCVUR CPL | — | — | SENDC ACPL | SENDU RCPL |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | RW1C | RW1C | R | R | RW1C | RW1C | R | RW1C | RW1C | RW1C | R | R | RW1C | RW1C |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|------|---|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | IBERR | B'0 | RW1C | IBERR Indicates that a transfer error has occurred on the internal bus. |
| 27 | MTLP | B'0 | RW1C | MTLP Indicates that a malformed TLP has been received. |
| 26 | UR | B'0 | RW1C | UR Indicates that an unsupported request has been received. |
| 25 | PTLP | B'0 | RW1C | PTLP Indicates that a poisoned TLP has been received. |
| 24 to 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 | POVF | B'0 | RW1C | POVF Indicates that the posted request packet reception buffer has overflowed. |
| 20 | NPOVF | B'0 | RW1C | NPOVF Indicates that the non-posted request packet reception buffer has overflowed. |
| 19 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 | RCVCP LLK | B'0 | RW1C | Receive Completion Lock Indicates that a completion lock (without data) has been received. This completion is returned when memory locking of the device supporting memory lock has failed. |
| 12 | UNEXP CPL | B'0 | RW1C | Unexpected Completion Indicates that an unexpected completion has been received. |
| 11, 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|-----------|---------------|------|---|
| 9 | RCVSZECPL | B'0 | RW1C | Receive Size Error Completion Indicates that the size of the completion received is different from the requested size. |
| 8 | CPLTOUT | B'0 | RW1C | CPL Timeout Indicates that a completion timeout has occurred. |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 | RCVCRSCPL | B'0 | RW1C | Receive Configuration Retry Status Completion Indicates that a completion with the configuration retry status has been received. |
| 5 | RCVCACPL | B'0 | RW1C | Receive Completion Abort Status Completion Indicates that a completion with the completion abort status has been received. |
| 4 | RCVURCPL | B'0 | RW1C | Receive Unsupported Request Status Completion Indicates that a completion with the unsupported request status has been received. |
| 3, 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | SENDACPL | B'0 | RW1C | Send Completion Abort Status Completion Indicates that a completion with the completion abort status has been transmitted. |
| 0 | SENDURCPL | B'0 | RW1C | Send Unsupported Request Status Completion Indicates that a completion with the unsupported request status has been transmitted. |

49.2.17 Error Interrupt Enable Register (PCIEERRFER)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register determines whether to reflect the occurrence of errors to PCIEINTR.INT_PCISERR.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|---------------|--------------|----|----|----------------|--------------|----|----------------|--------------|--------------|----|----|----------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | IBERR E | — | — | — | — | — | — | POVFE | NPOVFE | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R | R | R | R/W | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | RCVCP LLKE | UNEXP CPL | — | — | RCVSZ ECPLE | CPLTO UTE | — | RCVCR SCPLE | RCVCA CPL | RCVUR CPL | — | — | SENDC ACPLE | SENDU RCPL |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R | R | R/W | R/W | R | R/W | R/W | R/W | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | IBERRE | B'0 | R/W | IBERRE Enables generation of the INT_PCISERR interrupt upon occurrence of a transfer error on the internal bus. |
| 27 to 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 | POVFE | B'0 | R/W | POVFE Enables generation of the INT_PCISERR interrupt upon overflow of a posted request packet reception buffer. |
| 20 | NPOVFE | B'0 | R/W | NPOVFE Enables generation of the INT_PCISERR interrupt upon overflow of a non-posted request packet reception buffer. |
| 19 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 | RCVCP LLKE | B'0 | R/W | Receive Completion Lock Enables generation of the INT_PCISERR interrupt upon reception of the completion lock (without data). |
| 12 | UNEXP CPL | B'0 | R/W | Unexpected Completion Enabled Enables generation of the INT_PCISERR interrupt upon reception of the unexpected completion. |
| 11, 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | RCVSZ ECPLE | B'0 | R/W | Receive Size Error Completion Enabled Enables generation of the INT_PCISERR interrupt upon reception of the completion whose size is different from the requested size. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|------------|---------------|-----|---|
| 8 | CPLTOUTE | B'0 | R/W | Completion Timeout Interrupt Enable Enables generation of the INT_PCISERR interrupt upon detection of a completion timeout. |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 | RCVCRSCPLE | B'0 | R/W | Configuration Retry Status Completion Receive Interrupt Enable Enables generation of the INT_PCISERR interrupt upon reception of a completion with the configuration retry status. |
| 5 | RCVCACPLE | B'0 | R/W | Completion Abort Status Completion Receive Interrupt Enable Enables generation of the INT_PCISERR interrupt upon reception of a completion with the completion abort status. |
| 4 | RCVURCPLE | B'0 | R/W | Unsupported Request Status Completion Receive Interrupt Enable Enables generation of the INT_PCISERR interrupt upon reception of a completion with the unsupported request status. |
| 3, 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | SENDACPLE | B'0 | R/W | Send Completion Abort Status Completion Interrupt Enable Enables generation of the INT_PCISERR interrupt upon transmission of a completion with the completion abort status. |
| 0 | SENDURCPLE | B'0 | R/W | Send Unsupported Request Status Completion Interrupt Enable Enables generation of the INT_PCISERR interrupt upon transmission of a completion with the unsupported request status. |

49.2.18 Error Factor Register 2 (PCIEERRFR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------|----|-------|------|------|--------|---------|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | DLLPE | RTO | RNR | BADTLP | BADDLLP | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | RW1C | RW1C | RW1C | RW1C | RW1C | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RE | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | RW1C | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|------|---|
| 31, 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 | DLLPE | B'0 | RW1C | Data Link Layer Protocol Error Indicates that a data link layer protocol error has occurred. |
| 28 | RTO | B'0 | RW1C | Replay Timeout Indicates that a replay timeout has occurred. |
| 27 | RNR | B'0 | RW1C | Replay Number Rollover Indicates that a replay number rollover has occurred. |
| 26 | BADTLP | B'0 | RW1C | Bad TLP Indicates that a Bad TLP has been detected. |
| 25 | BADDLLP | B'0 | RW1C | Bad DLLP Indicates that a Bad DLLP has been detected. |
| 24 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | RE | B'0 | RW1C | Receiver Error Indicates that a receiver error has been detected. |
| 14 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.19 Transfer Interrupt Enable Register (PCIETIER)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|---------|----|----|----|----|----|----|----|----|----|----|----|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | MSIECE | — | — | — | — | — | — | — | — | — | — | — | DLLACTCE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | INTXDCE | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | MSIECE | B'0 | R/W | MSI Enable Change Enable Enables generation of the INT_PCISC interrupt upon change of MSICAP0[16].MSIE. |
| 27 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | DLLACTCE | B'0 | R/W | DLLACT Change Enable Enables generation of the INT_PCISC interrupt upon change of DLLACT. |
| 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | INTXDCE | B'0 | R/W | INTx Disable Change Enable Enables generation of the INT_PCISC interrupt upon change of PCICONF1[10].INTDIS. |
| 11 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.20 Power Management State Status Register (PCIEPMSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | PSTC | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | RW1C | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | PST | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|------|--|
| 31 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | PSTC | B'0 | RW1C | Power State Change Indicates that PST has changed. This bit is valid only in Endpoint mode. |
| 23 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | PST | B'0 | R | Power State Indicates the power state. 0: D0 1: Non-D0 Do not issue a request in the non-D0 state. This bit is valid only in Endpoint mode. |
| 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.21 Power Management State Interrupt Enable Register (PCIEPMSCIER)

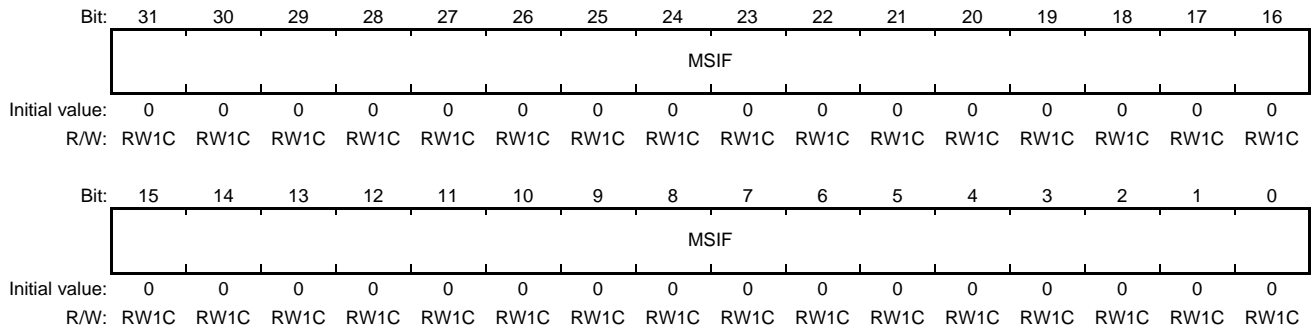
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | PSTCE | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | PSTCE | B'0 | R/W | Power State Change Enable Enables generation of the INT_PCISC interrupt upon change of PST. This bit is valid only in Endpoint mode. |
| 23 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.22 MSIF Register (PCIEMSIFR)

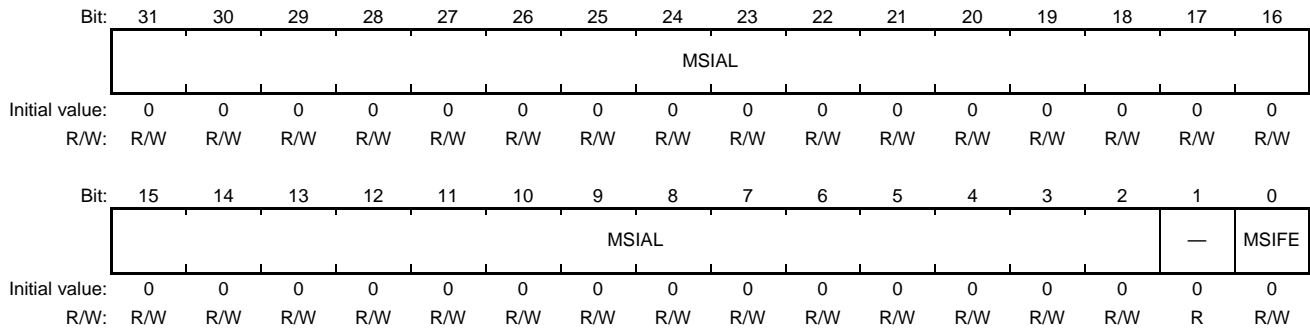
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|------|---|
| 31 to 0 | MSIF | All 0 | RW1C | <p>MSIF</p> <p>Indicates that an MSI interrupt has occurred.</p> <p>This field is valid only when PCIEMSIALR.MSIFE = 1 in Root Port mode.</p> <p>A write from the internal bus is interpreted as interrupt cancellation thus clearing the bit to 0 to which 1 has been written. A read from the internal bus is processed as a normal read.</p> <p>A 1DW write to the address specified with PCIEMSIAUR and PCIEMSIALR from the PCI Express side is interpreted as an MSI interrupt.</p> <p>This IP sets 1 to the bit determined by the sum of the values in bits [4:0] and [12:8] in the write data in order to convert the MSI message data into a flag. Set the function appropriately so that the interrupt should not be reflected to the same bit. A read from the address specified with PCIEMSIAUR and PCIEMSIALR from the PCI Express side is processed as a normal read.</p> |

49.2.23 MSI Address Lower Register (PCIEMSIALR)

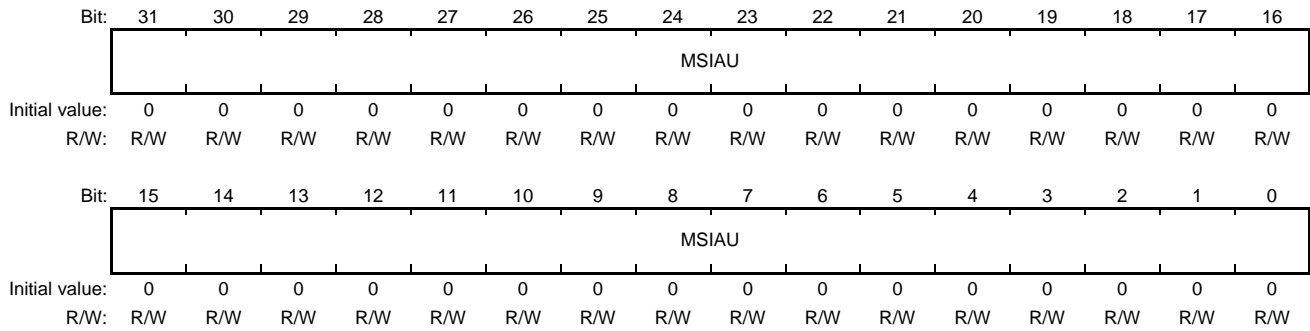
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 2 | MSIAL | All 0 | R/W | MSI Address Lower Specifies the lower address bits [31:2] of the memory write by MSI in the PCI space in Root Port mode. (Refer to section 49.3.3 (6), MSI Reception.) This field is invalid when PCIEMSIALR.MSIFE = 0. |
| 1 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 0 | MSIFE | B'0 | R/W | MSIF Enable Enables address decoding of the MSI interrupt. Be sure to set this bit to 0 in Endpoint mode. |

49.2.24 MSI Address Upper Register (PCIEMSIUR)

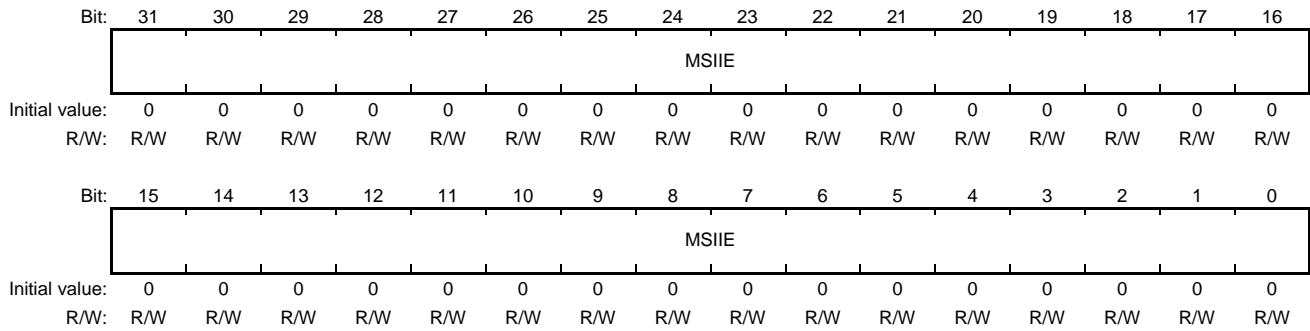
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | MSIAU | All 0 | R/W | MSI Address Upper Specifies the upper address bits [63:32] of the memory write by MSI in the PCI space in Root Port mode. (Refer to section 49.3.3 (6), MSI Reception.) This field is invalid when PCIEMSIALR.MSIFE = 0. |

49.2.25 MSI Interrupt Enable Register (PCIEMSIER)

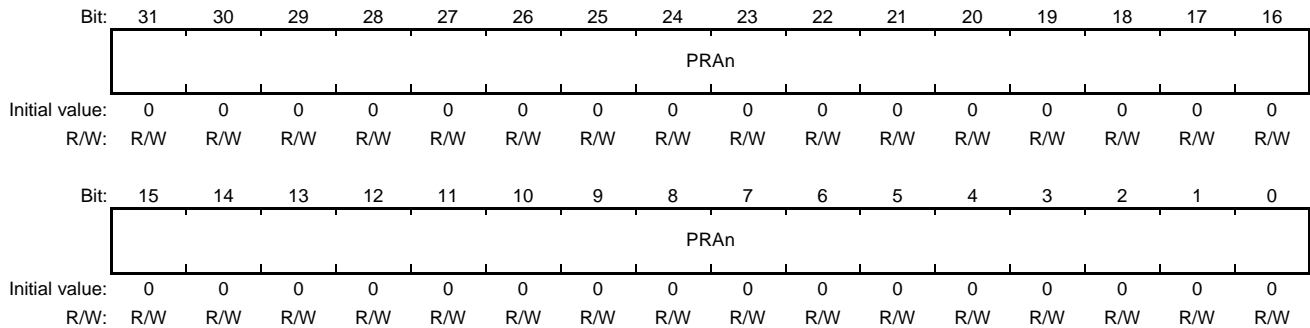
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | MSIIE | All 0 | R/W | MSI Interrupt Enable Individually enables the interrupts by MSI in Root Port mode. This field is invalid when PCIEMSIALR.MSIFE = 0. |

49.2.26 PCI Express Root Port Address Register 0 to 5 (PCIEPRAR0 to PCIEPRAR5)

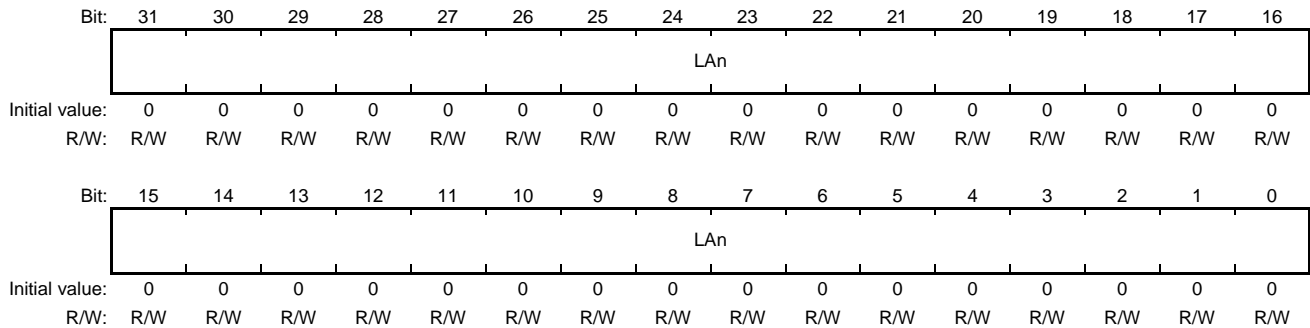
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | PRAn | All 0 | R/W | <p>PCI Express Root Port Address n</p> <p>Specifies the PCI Express address for transferring a transaction from the PCI Express to the internal bus in Root Port mode.</p> <p>This field is invalid in Endpoint mode.</p> <p>Set the appropriate value corresponding to the local address registers and local address mask registers. Modify this field only when no access should occur to the corresponding PCI Express, for example, while a link is down and the master enable of the peer is off.</p> |

49.2.27 Local Address Register 0 to 5 (PCIELAR0 to PCIELAR5)

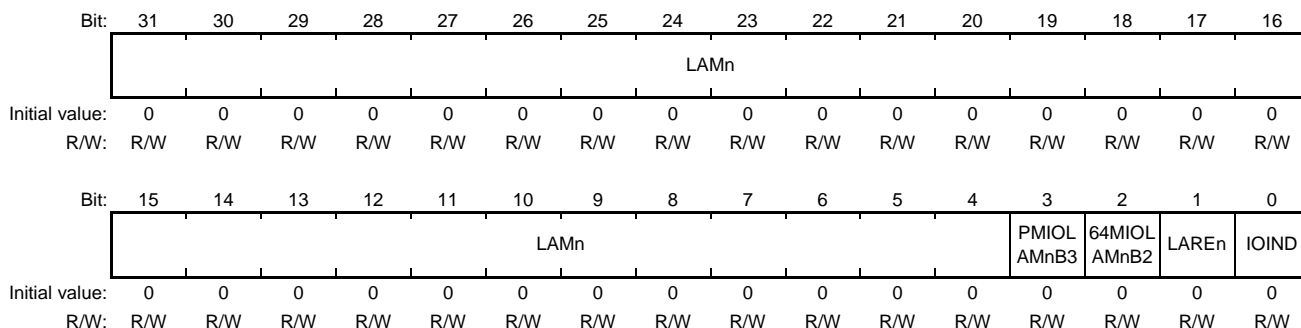
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | LAn | All 0 | R/W | <p>Local Address n</p> <p>Indicates the local (internal bus) address for transferring a transaction from the PCI Express to the internal bus.</p> <p>When the addresses in the space allocated by the BARn of the PCI Express are translated into the internal bus space addresses, this field is used as the upper address bits. The lower address bits of the PCI Express packets are used as the lower address bits. The boundary between upper and lower bits is determined by PCIELAMR0 to PCIELAMR5.</p> <p>Writing to this field is prohibited while PCIETCTLR.CFINIT is 1.</p> <p>When a 64-bit access is enabled with PCIELAMR0 to PCIELAMR5, write 0 to all the bits in PCIELARn + 1.</p> <p>The write value for bits 1 and 0 should always be 0.</p> |

49.2.28 Local Address Mask Register 0 to 5 (PCIELAMR0 to PCIELAMR5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



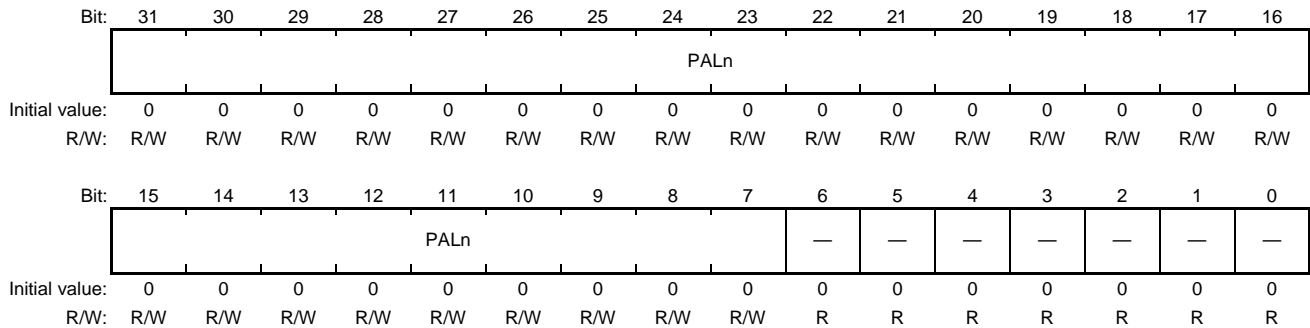
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 4 | LAMn | All 0 | R/W | <p>Local Address Mask n</p> <p>Masks the local (internal bus) addresses when transactions are transferred from PCI Express to the internal bus.</p> <p>When the addresses in the space allocated by the BARn of the PCI Express are translated into the internal bus space addresses, the PCIELARn.LAn value is used as the upper bits while the address in the PCI Express packet header is used as the lower bits. This field determines the boundary between upper and lower bits. Therefore, this field also determines the size of the space to be allocated to the PCI Express space.</p> <p>B'0000_0000_0000_0000_0000_0000_0000: 4, 8, or 16 bytes B'0000_0000_0000_0000_0000_0000_0001: 32 bytes B'0000_0000_0000_0000_0000_0000_0011: 64 bytes B'0000_0000_0000_0000_0000_0000_0111: 128 bytes B'0000_0000_0000_0000_0000_0000_1111: 256 bytes (Omitted) B'0000_1111_1111_1111_1111_1111_1111: 256 Mbytes B'0001_1111_1111_1111_1111_1111_1111: 512 Mbytes B'0011_1111_1111_1111_1111_1111_1111: 1 Gbyte B'0111_1111_1111_1111_1111_1111_1111: 2 Gbytes B'1111_1111_1111_1111_1111_1111_1111: 4 Gbytes</p> <p>This field must not be set to any value other than those given above. Writing to this field is prohibited when PCIETCTLR.CFINIT is 1.</p> <p>When using the space as a memory space, set a size of 128 bytes or more with BAR.</p> <p>When setting a 4-Gbyte space, set 64M_IOLAMB2 to 1.</p> <p>When 64-bit addresses are used with PCIELAMRn, PCIELAMRn + 1 is used as the upper address of PCIELAMRn, and so PCIELAMRn cannot be used as an independent space. In this case, set this field as follows.</p> <p>B'0000_0000_0000_0000_0000_0000_0000_0000: 4 Gbytes or less</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-------------|---------------|-----|---|
| 3 | PMIOLAMnB3 | B'0 | R/W | <p>Prefetchable Memory or IO Local Address Mask n Bit 3</p> <ul style="list-style-type: none"> When IO indicator is 0: <ul style="list-style-type: none"> Specifies memory space prefetch. 0: A non-prefetchable memory space is allocated. 1: A prefetchable memory space is allocated. When IO indicator is 1: <ul style="list-style-type: none"> Masks bit 3 as an extension of local address mask n. 0: A space of 4 or 8 bytes is allocated. 1: A space of 16 bytes or more is allocated. <p>Writing to this field is prohibited when PCIETCTLR.CFINIT is 1. When 64-bit addresses are used with PCIELAMRn, PCIELAMRn + 1 is used as the upper address of PCIELAMRn, and so PCIELAMRn cannot be used as an independent space. Set all the bits to 0.</p> |
| 2 | 64MIOLAMnB2 | B'0 | R/W | <p>64-bit Access Space Memory or IO Local Address Mask n Bit 2</p> <ul style="list-style-type: none"> When IO indicator is 0: <ul style="list-style-type: none"> Specifies the width of the addresses to define BAR in the memory space. 0: Indicates that 32-bit addresses are used with PCIELAMRn. 1: Indicates that 64-bit addresses are used with PCIELAMRn. When IO indicator is 1: <ul style="list-style-type: none"> Masks bit 2 as an extension of local address mask n. 0: A space of 4 bytes is allocated. 1: A space of 8 bytes or more is allocated. <p>Writing to this field is prohibited when PCIETCTLR.CFINIT is 1. When 64-bit addresses are used with PCIELAMRn, PCIELAMRn + 1 is used as the upper address of PCIELAMRn, and so PCIELAMRn cannot be used as an independent space. Set all the bits to 0.</p> |
| 1 | LAREn | B'0 | R/W | <p>Local Address Enable</p> <p>Enables local address.</p> <p>Setting this field to 1 enables transaction transfer in the area specified by PCIELARn and PCIELAMRn. Set this field to 1 to enable address translation by LAn and LAMn.</p> <p>Writing to this field is prohibited when PCIETCTLR.CFINIT is 1. When 64-bit addresses are used with PCIELAMRn, PCIELAMRn + 1 is used as the upper address of PCIELAMRn, and so PCIELAMRn cannot be used as an independent space. Set all the bits to 0.</p> |
| 0 | IOIND | B'0 | R/W | <p>IO Indicator</p> <p>Specifies the type of space on the PCI Express side allocated for transaction transfer from the PCI Express to the internal bus.</p> <p>0: Memory space 1: IO space</p> <p>Writing to this field is prohibited when PCIETCTLR.CFINIT is 1. When 64-bit addresses are used with PCIELAMRn, PCIELAMRn + 1 is used as the upper address of PCIELAMRn, and so PCIELAMRn cannot be used as an independent space. Set all the bits to 0.</p> |

Note: Setting the memory space type of 1 Mbyte or less (defined in the PCI Express standard 2.2 and the earlier versions) is not supported.

49.2.29 PCIEC Address Lower Register 0 to 3 (PCIEPALR0 to PCIEPALR3)

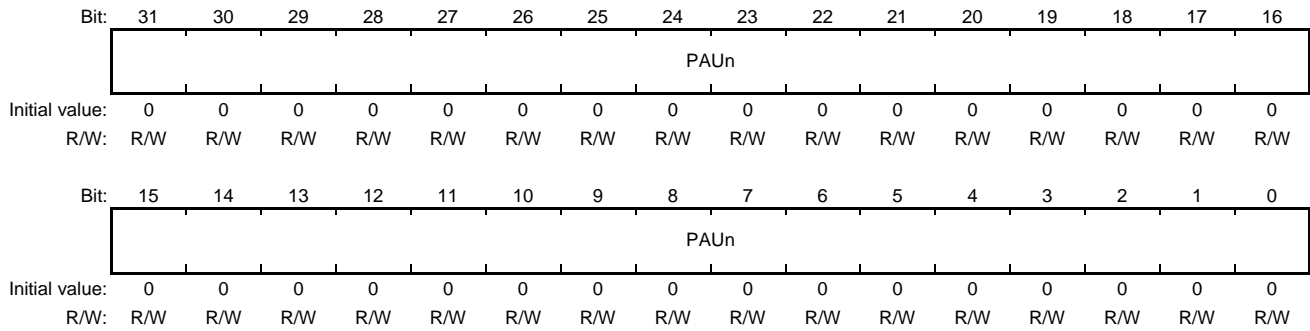
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 7 | PALn | All 0 | R/W | PCI Express Address Lower Sets the PCI Express address (lower) for PIO transfer from the internal bus to the PCI Express. |
| 6 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.30 PCIEC Address Upper Register 0 to 3 (PCIEPAUR0 to PCIEPAUR3)

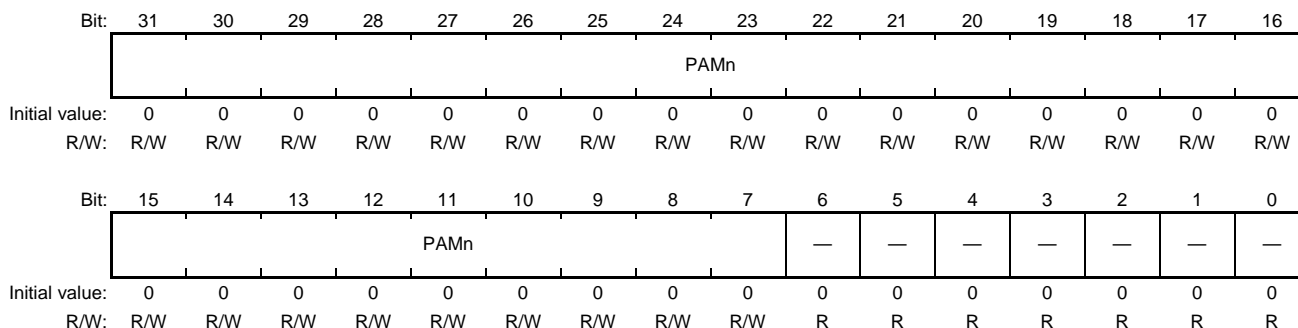
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | PAUn | All 0 | R/W | PCI Express Address Upper Sets the PCI Express address (upper) for PIO transfer from the internal bus to the PCI Express. |

49.2.31 PCIEC Address Mask Register 0 to 3 (PCIEPAMR0 to PCIEPAMR3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 7 | PAMn | All 0 | R/W | <p>PCI Express Address Mask</p> <p>Masks the PCI Express addresses.</p> <p>When the packet accessing the internal bus space n is converted to the PCI Express packet, PAUn and PALn are used as the upper bits while the address on the internal bus packet is used as the lower bits. Here, PAMn determines the boundary between them. That is, the packet address addr[31:0] on the PCI Express is generated as follows.</p> <p>Addr[31:7] = PALn[31:7] when the corresponding bit in PAMn[31:7] is 0, and Addr[31:7] = internal bus - addr[31:7] when the corresponding bit in PAMn[31:7] is 1.</p> <p>Addr[6:0] = internal bus - addr[6:0]</p> <p>According to the setting of this field, the size of the space to which the PCI Express side transactions is transferred is determined as follows.</p> <p>B'0000_0000_0000_0000_0000_0000_0: 128 bytes B'0000_0000_0000_0000_0000_0000_1: 256 bytes B'0000_0000_0000_0000_0000_0001_1: 512 bytes B'0000_0000_0000_0000_0000_0011_1: 1 Kbyte B'0000_0000_0000_0000_0000_0111_1: 2 Kbytes (Omitted) B'0000_0111_1111_1111_1111_1111_1: 128 Mbytes B'0000_1111_1111_1111_1111_1111_1: 256 Mbytes B'0001_1111_1111_1111_1111_1111_1: 512 Mbytes B'0011_1111_1111_1111_1111_1111_1: 1 Gbyte B'0111_1111_1111_1111_1111_1111_1: 2 Gbytes</p> <p>The area size exceeding the size defined in the address map must not be set.</p> <p>Any value other than those given above must not be set.</p> |
| 6 to 0 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

49.2.32 PCIEC Conversion Control Register 0 to 3 (PCIEPTCTLR0 to PCIEPTCTLR3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------|----|----|-------|----|----|----|------|----|-----|-----|-----|----|----|-------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | PARE | — | — | — | — | — | — | — | — | TCn | | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | LOCKn | — | — | — | SPCn | — | — | — | ZLR | — | — | ATTRn | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R/W | R | R | R | R/W | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | PARE | B'0 | R | PAR Enable Indicates that the PCI Express address upper and lower registers are enabled. |
| 30 to 23 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 22 to 20 | TCn | All 0 | R/W | Traffic Class Specifies the traffic class (TC) of the PCI Express packet to be transferred. |
| 19 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | LOCKn | B'0 | R/W | LOCK Sets the lock of PCI Express packets to be transmitted. When this bit is set, locked transactions are transmitted. In Endpoint mode, this bit must not be set. Before making access requiring exclusive processing, confirm that the locked transaction has not caused an error. |
| 11 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | SPCn | B'0 | R/W | SPC Specifies a transfer destination space. 0: Memory 1: IO (A request cannot be issued to the IO space in Endpoint mode.) When issuing a request to the IO space, be sure to separate the accesses at 1DW boundaries. |
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 4 | ZLR | B'0 | R/W | <p>ZLR</p> <p>Sets the zero-length read.</p> <p>0: A read access to this space is output as a read access as is.</p> <p>1: A 1DW read access to this space is output after converted to a zero-length read.</p> <p>Do not make any read access other than 1DW read access when this bit is set.</p> <p>The read value depends on the dummy data used by PCI Express for zero-length read.</p> <p>According to the standard, the value is undefined.</p> <p>Setting this bit has no effect for write access.</p> |
| 3, 2 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 1, 0 | ATTRn | All 0 | R/W | <p>ATTR</p> <p>Specifies the attribute of the PCI Express packet to be transferred.</p> <p>ATTR[0]: No snoop</p> <p>ATTR[1]: Relaxed ordering</p> <p>Setting ATTR[0] (no snoop bit) to 1 disables snooping in the transfer destination.</p> <p>Setting ATTR[1] (relaxed ordering bit) to 1 relaxes ordering in the transfer destination.</p> |

49.2.33 PCIEC DMAC DMA Operation Register (PCIEDMAOR)

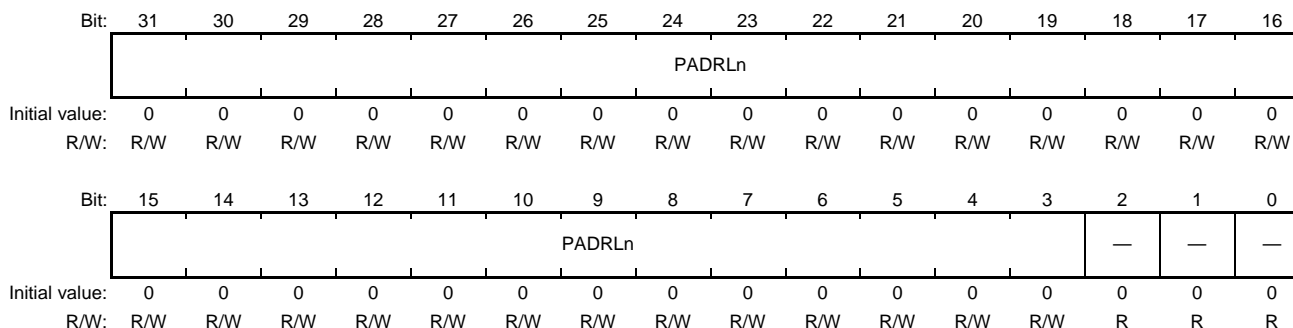
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | DMAE | — | — | — | — | — | — | — | — | — | — | — | — | — | — | DMAACT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ABT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | DMAE | B'0 | R/W | DMA Enable Enables DMA function. Set this bit to 1 to use DMA function. This bit must not be cleared to 0 during DMA transfer (PCIEDMCHCR0-7.CHE = 1). |
| 30 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | DMAACT | B'0 | R | DMA Active Indicates whether there is a currently operating DMA channel. 0: No DMA channel is currently operating. 1: At least one DMA channel is currently operating. |
| 15 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | ABT | B'0 | R/W | Arbitration Specifies the method of arbitration among the channels. 0: CH0 > CH1 > CH2 > CH3 > CH4 > CH5 > CH6 > CH7 1: Round robin mode |

49.2.34 PCIEC DMAC PCIEC Address Lower Register 0 to 7 (PCIEDMPALR0 to PCIEDMPALR 7)

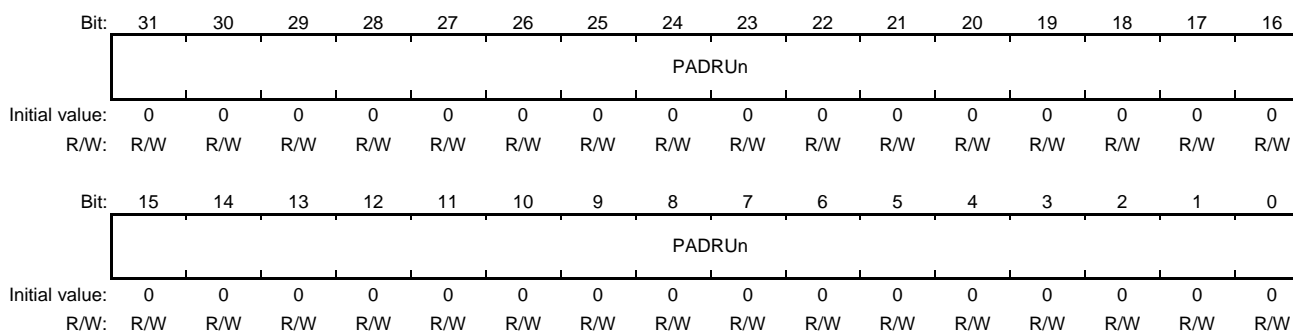
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 3 | PADRLn | All 0 | R/W | PCI Express Address Lower Sets the lower 32 bits of the PCI Express address for DMA transfer. Only an 8-byte boundary address can be set. |
| 2 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.35 PCIEC DMAC PCIEC Address Upper Register 0 to 7 (PCIEDMPAUR0 to PCIEDMPAUR7)

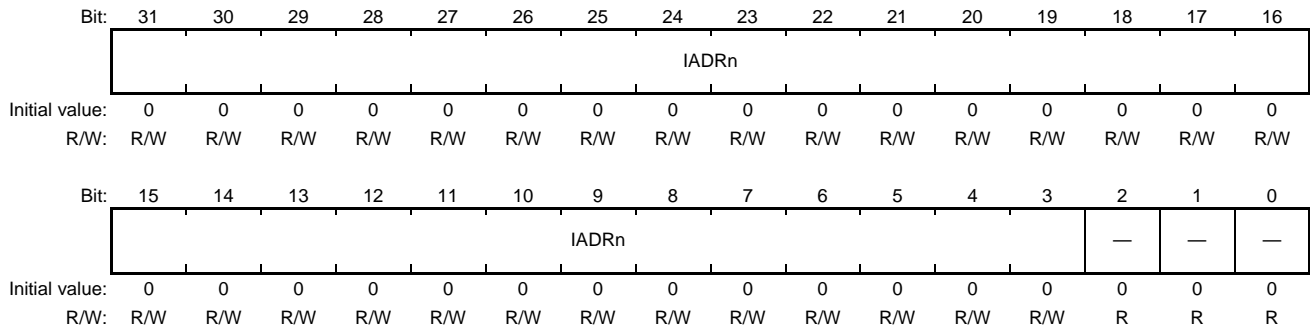
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | PADRUn | All 0 | R/W | PCI Express Address Upper Sets the upper 32 bits of the PCI Express address for DMA transfer. |

49.2.36 PCIEC DMAC Internal Bus Address Register 0 to 7 (PCIEDMIAR0 to PCIEDMIAR7)

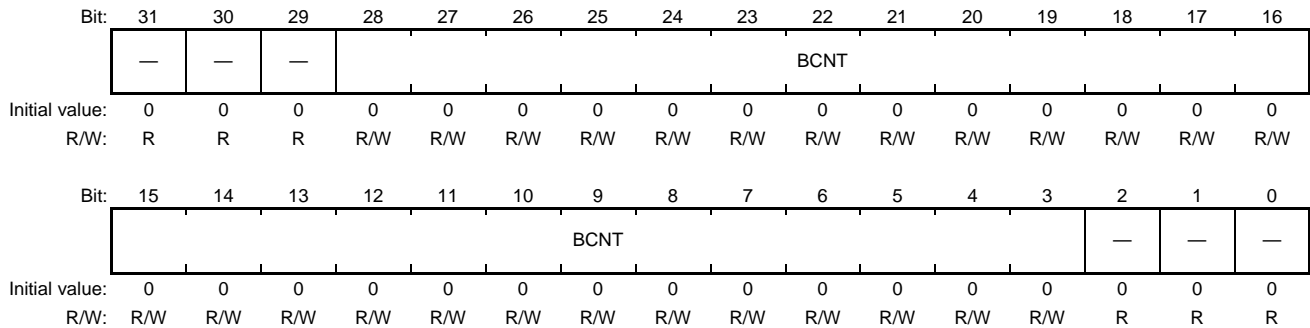
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 3 | IADRn | All 0 | R/W | Internal Address Lower Sets the internal bus address for DMA transfer. Only an 8-byte boundary address can be set. Internal bus addresses for registers or PIO transfer must not be set. |
| 2 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.37 PCIEC DMAC Byte Count Register 0 to 7 (PCIEDMBCNTR0 to PCIEDMBCNTR7)

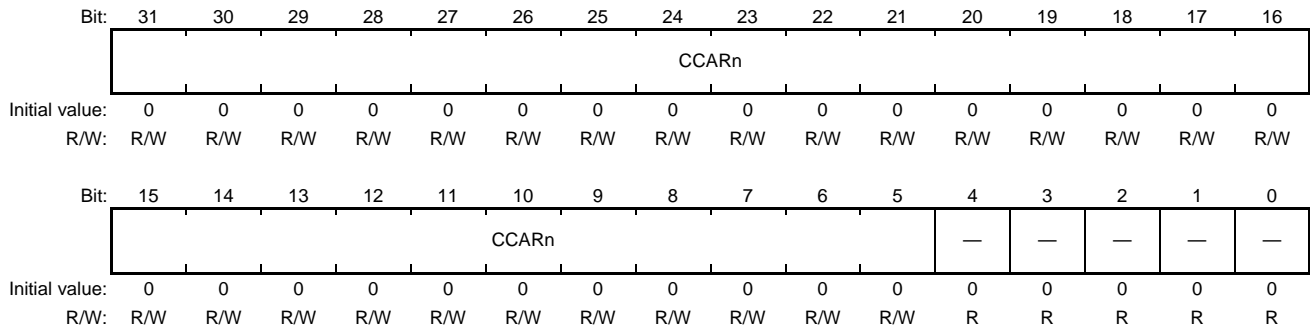
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 3 | BCNT | All 0 | R/W | Byte Count Specifies the transfer byte count. Only a multiple of 8 can be specified. In this field, set one-eighth of the number to be specified (omit the lower three bits from the number). When 0 is set, 2 ²⁹ bytes are transferred. Note: If the transfer direction is changed after a transfer, the byte count read value will be undefined until the next transfer byte count is set (written). Be sure to write the byte count after initialization or completion of the previous transfer and before the start of the next transfer. The value before modification cannot be referenced or used. |
| 2 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.38 PCIEC DMAC Command Chain Address Register 0 to 7 (PCIEDMCCAR0 to PCIEDMCCAR7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 5 | CCARn | All 0 | R/W | Command Chain Address Specifies a command chain address for DMA transfer. Only a 32-byte boundary address can be set. |
| 4 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.39 PCIEC DMAC Channel Control Register 0 to 7 (PCIEDMCHCR0 to PCIEDMCHCR7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|-----|-----|-----|----|----|----|-------|-----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CHEn | DIRn | CCEn | CHTn | — | — | — | — | — | — | ATTRn | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R/W | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | TC | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 31 | CHEn | B'0 | R/W | <p>Channel Enable</p> <p>Enables a DMA channel.</p> <p>Setting this bit to 1 starts DMA transfer on the corresponding channel. Note that, DMA transfer is not performed while the bit that indicates a transfer end (PCIEDMCHSRn.TE), a transfer error (PCIEDMCHSRn.PEE/IBE), or completion of DMA transfer suspension processing (PCIEDMCHSRn.CHTC) is 1.</p> <p>This bit must not be cleared during DMA transfer.</p> <p>This bit is not cleared to 0 by a transfer end or suspension.</p> <p>0: Disables data transfer. 1: Enables data transfer.</p> |
| 30 | DIRn | B'0 | R/W | <p>Direction</p> <p>Specifies the data transfer direction.</p> <p>0: PCI Express → internal bus 1: Internal bus → PCI Express</p> |
| 29 | CCEn | B'0 | R/W | <p>Command Chain Enable</p> <p>Enables a command chain.</p> <p>If data transfer is requested with this bit set to 1, the data is transferred by reading a command from an address set in PCIEDMCCARn.</p> <p>0: Disables a command chain. 1: Enables a command chain.</p> |
| 28 | CHTn | B'0 | R/W | <p>Channel Terminate</p> <p>Forcibly suspends DMA transfer in normal transfer.</p> <p>Setting this bit to 1 during DMA transfer suspends DMA transfer forcibly. Completion of DMA transfer suspension processing can be checked with PCIEDMCHSRn.CHTC. After transfer suspension, initialize the registers for the channel suspended.</p> <p>When setting this bit to 1, set other bits to the same value.</p> <p>This bit can be set to 1 only when a command chain is disabled (PCIEDMCHCRn.CCE = 0).</p> <p>When PCIEDMCHCRn.CCE is 1, this bit must be cleared to 0.</p> <p>This bit is always read as 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 27 to 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21, 20 | ATTRn | All 0 | R/W | ATTR Specifies the attribute of the PCI Express packet to be transferred. ATTR[0]: No snoop ATTR[1]: Relaxed ordering |
| 19 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 9 | TC | All 0 | R/W | TC Specifies the traffic class (TC) of the PCI Express packet to be transferred. |
| 8 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.40 PCIEC DMAC Channel Status Register 0 to 7 (PCIEDMCHSR0 to PCIEDMCHSR7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|--------------------|-------|----|--------|----|----|----|----|----|-----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | CHTCE _n | PEEEn | — | IBEEEn | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R | R/W | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | CHTCn | PEEn | — | IBEn | — | — | — | — | — | IEn | — | — | TEn |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | RW1C | RW1C | R | RW1C | R | R | R | R | R | R/W | R | R | RW1C |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------|---------------|------|---|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | CHTCE _n | B'0 | R/W | Channel Terminate Complete Interrupt Enable Enables an interrupt caused by completion of forced suspension. 0: Disables generation of an interrupt. 1: Enables generation of an interrupt. |
| 27 | PEEEn | B'0 | R/W | PCIE Error Interrupt Enable Enables an interrupt caused by a transfer error on the PCI Express side. 0: Disables generation of an interrupt. 1: Enables generation of an interrupt. |
| 26 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 25 | IBEEEn | B'0 | R/W | Internal Bus Error Interrupt Enable Enables an interrupt caused by a transfer error on the internal bus side. 0: Disables generation of an interrupt. 1: Enables generation of an interrupt. |
| 24 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | CHTCn | B'0 | RW1C | Channel Terminate Complete Indicates that the forced suspension processing started by setting PCIEDMCHCRn.CHT or PCIEDMCHC2Rn.CHT has been completed. With this bit set, DMA transfer is not performed by setting PCIEDMCHCRn.CHE to 1. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|------|---|
| 11 | PEEn | B'0 | RW1C | <p>PCIE Error</p> <p>Indicates that a transfer error has occurred on the PCI Express side.</p> <p>This error is caused in the following cases:</p> <ul style="list-style-type: none"> • PCIEDMCHCRn.CHE is set to 1 while PCIEDMAOR.DMAE is 0. • DMA transfer is started while a link is not established, or DL_Down is caused during DMA transfer. • DMA transfer is started in the Non-D0 state. • DMA transfer is started with TC being set that is not mapped to VC0. • DMA transfer is started with PCICONF1[2].BME = 0 in Endpoint mode, or the bit is deasserted during DMA transfer. • DMA transfer is started with an attribute being set that is not enabled in the configuration registers. • A completion other than SC is received when PCI Express → internal bus. • A completion timeout occurs when PCI Express → internal bus. • A malformed completion is received when PCI Express → internal bus. • A poisoned completion is received when PCI Express → internal bus. • With this bit set, DMA transfer is not performed by setting PCIEDMCHCRn.CHE to 1. |
| 10 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 9 | IBEn | B'0 | RW1C | <p>Internal Bus Error</p> <p>Indicates that a transfer error has occurred on the internal bus side.</p> <p>With this bit set, DMA transfer is not performed by setting PCIEDMCHCRn.CHE to 1.</p> |
| 8 to 4 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 3 | IEn | B'0 | R/W | <p>Interrupt Enable</p> <p>Enables an interrupt request.</p> <p>When this bit is 1, setting TE bit requests an interrupt.</p> <p>0: Disables an interrupt request. 1: Enables an interrupt request.</p> |
| 2, 1 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|---|
| 0 | TEn | B'0 | RW1C | <p>Transfer End</p> <p>Indicates a transfer end.</p> <p>This bit is set to 1 when data transfer ends with the value in PCIEDMBCNTR0- PCIEDMBCNTR7 cleared to 0. This bit is not set to 1 when a transfer end is caused by a transfer error or data transfer is forcibly terminated by setting the PCIEDMCHCRn.CHT bit.</p> <p>With this bit set, DMA transfer is not performed by setting PCIEDMCHCRn.CHE to 1.</p> <p>0: Indicates that data transfer is being performed or has been suspended.</p> <p>1: Indicates that data transfer has ended (with PCIEDMBCNTRn cleared to 0).</p> |

49.2.41 PCIEC DMAC Channel Control 2 Register 0 to 7 (PCIEDMCHC2R0 to PCIEDMCHC2R7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|------|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | CHTn | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | CHTn | B'0 | R/W | Command Chain Channel Terminate Forcibly suspends DMA transfer when command chain is used. Setting this bit to 1 during DMA transfer suspends DMA transfer forcibly. Completion of DMA transfer suspension processing can be checked with PCIEDMCHSRn.CHTC. After transfer suspension, initialize the registers for the channel suspended. This bit can be set to 1 only when a command chain is enabled (PCIEDMCHCRn.CCE = 1). When PCIEDMCHCRn.CCE is 0, this bit must be cleared to 0. This bit is always read as 0. |
| 27 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.42 PCI Configuration Register 0 (PCICONF0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

[RZ/G2H]

Common to Header TYPE00/01

| | | | | | | | | | | | | | | | | |
|------------------|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Device ID | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Vendor ID | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-----------|---------------|---------|------------------|---|
| 31 to 16 | Device ID | H'0025 | R | R | Device ID Indicates the device ID. Reflects a value set in the IDSETR0.Device ID set. Indicates the device ID assigned to the PCI device vendor. |
| 15 to 0 | Vendor ID | H'1912 | R | R | Vendor ID Indicates the vendor ID. Reflects a value set in the IDSETR0.Vendor ID set. Indicates the PCI device vendor ID. |

[RZ/G2M V1.3, RZ/G2M V3.0]

Common to Header TYPE00/01

| | | | | | | | | | | | | | | | | |
|------------------|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Device ID | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Vendor ID | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-----------|---------------|---------|------------------|---|
| 31 to 16 | Device ID | H'0028 | R | R | Device ID Indicates the device ID. Reflects a value set in the IDSETR0.Device ID set. Indicates the device ID assigned to the PCI device vendor. |
| 15 to 0 | Vendor ID | H'1912 | R | R | Vendor ID Indicates the vendor ID. Reflects a value set in the IDSETR0.Vendor ID set. Indicates the PCI device vendor ID. |

[RZ/G2N]

Common to Header TYPE00/01

| | | | | | | | | | | | | | | | | |
|------------------|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Device ID | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Vendor ID | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-----------|---------------|---------|------------------|---|
| 31 to 16 | Device ID | H'002B | R | R | Device ID Indicates the device ID. Reflects a value set in the IDSETR0.Device ID set. Indicates the device ID assigned to the PCI device vendor. |
| 15 to 0 | Vendor ID | H'1912 | R | R | Vendor ID Indicates the vendor ID. Reflects a value set in the IDSETR0.Vendor ID set. Indicates the PCI device vendor ID. |

[RZ/G2E]

Common to Header TYPE00/01

| | | | | | | | | | | | | | | | | |
|------------------|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Device ID | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Vendor ID | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-----------|---------------|---------|------------------|---|
| 31 to 16 | Device ID | H'002D | R | R | Device ID Indicates the device ID. Reflects a value set in the IDSETR0.Device ID set. Indicates the device ID assigned to the PCI device vendor. |
| 15 to 0 | Vendor ID | H'1912 | R | R | Vendor ID Indicates the vendor ID. Reflects a value set in the IDSETR0.Vendor ID set. Indicates the PCI device vendor ID. |

49.2.43 PCI Configuration Register 1 (PCICONF1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

| | | | | | | | | | | | | | | | | |
|------------------|------|------|------|------|------|---------------|--------|----------|----------|---------|-------|-------|----|-----|-----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | DPE | SSE | RMA | RTA | STA | DEVSEL Timing | MDPE | FBBTC AP | — | 66MCA P | CAPL | INTST | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| PCI R/W | RW1C | RW1C | RW1C | RW1C | RW1C | R | R | RW1C | R | R | R | R | R | R | R | R |
| Internal bus R/W | RW1C | RW1C | RW1C | RW1C | RW1C | R | R | RW1C | R | R | R | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | INTDIS | FBTBTE | SERRE | IDSS/WCC | PERS | VGAPS | MWIE | SC | BME | MSE | IOSE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R/W | R | R/W | R | R/W | R | R | R | R/W | R/W | R/W |
| Internal bus R/W | R | R | R | R | R | R/W | R | R/W | R | R/W | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|-----|----------|---------------|---------|------------------|---|
| 31 | DPE | B'0 | RW1C | RW1C | <p>Detected Parity Error</p> <p>Indicates that a parity error has been detected on the primary bus side.</p> <ul style="list-style-type: none"> Root Port: This bit is set to 1 when a poisoned TLP is transmitted. Endpoint: This bit is set to 1 when a poisoned TLP is received. |
| 30 | SSE | B'0 | RW1C | RW1C | <p>Signaled System Error</p> <p>Indicates that a system error has been detected.</p> <ul style="list-style-type: none"> Root Port: This bit is set to 1 if a fatal/non-fatal error is detected while PCICONF1[8].SERRE = 1 or if a ERR_FATAL/ERR_NONFATAL message is received while PCICONF1[8].SERRE = 1 and PCICONF15[17].SERRE = 1. Endpoint: This bit is set to 1 if a ERR_FATAL/ERR_NONFATAL message is transmitted while PCICONF1[8].SERRE = 1. |
| 29 | RMA | B'0 | RW1C | RW1C | <p>Received Master Abort</p> <p>Indicates that a master abort has been detected on the primary bus side.</p> <ul style="list-style-type: none"> Root Port: This bit is set to 1 when a completion with the unsupported request completion status is transmitted. Endpoint: This bit is set to 1 when a completion with the unsupported request completion status is received. |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|--------|---------------|---------------|---------|------------------|---|
| 28 | RTA | B'0 | RW1C | RW1C | <p>Received Target Abort</p> <p>Indicates that a target abort has been detected on the primary bus side.</p> <ul style="list-style-type: none"> • Root Port: This bit is set to 1 when a completion with the completer abort completion status is transmitted. • Endpoint: This bit is set to 1 when a completion with the completer abort completion status is received. |
| 27 | STA | B'0 | RW1C | RW1C | <p>Signaled Target Abort</p> <p>Indicates that a target abort has been transmitted to the primary bus side.</p> <ul style="list-style-type: none"> • Root Port: This bit is set to 1 when a completion with the completer abort completion status is received. • Endpoint: This bit is set to 1 when a completion with the completer abort completion status is transmitted. |
| 26, 25 | DEVSEL Timing | All 0 | R | R | <p>DEVSEL Timing</p> <p>Fixed to 0. These bits are not used for PCI Express.</p> |
| 24 | MDPE | B'0 | RW1C | RW1C | <p>Master Data Parity Error</p> <p>Indicates that a master data parity error has occurred on the primary bus side. This bit is set to 1 when any of the following conditions are satisfied while PCICONF1[6].PERS is 1.</p> <ul style="list-style-type: none"> • Root Port: <ol style="list-style-type: none"> 1. A poisoned completion is transmitted. 2. A poisoned request is received. • Endpoint: <ol style="list-style-type: none"> 1. A poisoned completion is received. 2. A poisoned request is transmitted. |
| 23 | FBBTCAP | B'0 | R | R | <p>Fast Back to Back Transaction Capable</p> <p>Fixed to 0. This bit is not used for PCI Express.</p> |
| 22 | — | B'0 | R | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 21 | 66MCAP | B'0 | R | R | <p>66 MHz Capable</p> <p>This bit is always read as 0. The write value should always be 0.</p> <p>This bit is not used for PCI Express.</p> |
| 20 | CAPL | B'1 | R | R/W | <p>Capabilities List</p> <p>Indicates that the extended capabilities list exists in the PCI compatible configuration space. This IP supports the extended capabilities list.</p> <p>This bit must not be set to a value other than 1.</p> |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|----------|---------------|---------|------------------|--|
| 19 | INTST | B'0 | R | R | <p>Interrupt Status</p> <p>This bit is set to 1 when an interrupt is generated by INTx message transmission. It is cleared to 0 when a requested interrupt processing ends. When MSI is used by the device, this field is not set to 1.</p> <p>Fixed to 0 in Root Port mode.</p> |
| 18 to 16 | — | All 0 | R | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 15 to 11 | — | All 0 | R | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 10 | INTDIS | B'0 | R/W | R/W | <p>Interrupt Disable</p> <p>Disables an INTx message transmission.</p> <p>Setting this bit to 1 disables an INTx message transmission.</p> |
| 9 | FBSTE | B'0 | R | R | <p>Fast Back to Back Transaction Enable</p> <p>This bit is always read as 0. The write value should always be 0.</p> <p>This bit is not used for PCI Express.</p> |
| 8 | SERRE | B'0 | R/W | R/W | <p>System Error Enable</p> <p>Enables an error message transmission when a non-fatal/fatal error is detected.</p> <p>Setting this bit to 1 enables an error message transmission.</p> |
| 7 | IDSS/WCC | B'0 | R | R | <p>IDSel Stepping/Wait Cycle Control</p> <p>This bit is always read as 0. The write value should always be 0.</p> <p>This bit is not used for PCI Express.</p> |
| 6 | PERS | B'0 | R/W | R/W | <p>Parity Error Response</p> <p>Enables setting the PCICONF1[24].MDPE bit.</p> |
| 5 | VGAPS | B'0 | R | R | <p>VGA Plate Snoop</p> <p>This bit is always read as 0. The write value should always be 0.</p> <p>This bit is not used for PCI Express.</p> |
| 4 | MWIE | B'0 | R | R | <p>Memory Write and Invalidate Enable</p> <p>This bit is always read as 0. The write value should always be 0.</p> <p>This bit is not used for PCI Express.</p> |
| 3 | SC | B'0 | R | R | <p>Special Cycle</p> <p>This bit is always read as 0. The write value should always be 0.</p> <p>This bit is not used for PCI Express.</p> |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|-----|----------|---------------|---------|------------------|---|
| 2 | BME | B'0 | R/W | R/W | Bus Master Enable <ul style="list-style-type: none"> • Root Port: Enables software to transfer the memory/IO request received from the endpoint device to the upper layer. When this bit is 0, the memory/IO request received is handled as an unsupported request. • Endpoint: When this bit is cleared to 0, transmission of a memory/IO request is disabled. |
| 1 | MSE | B'0 | R/W | R/W | Memory Space Enable Enables an access to memory space. When this bit is 0, the memory request to this device is handled as an unsupported request. |
| 0 | IOSE | B'0 | R/W | R/W | IO Space Enable Enables an access to I/O space. When this bit is 0, the IO request to this device is handled as an unsupported request. |

49.2.44 PCI Configuration Register 2 (PCICONF2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

| | | | | | | | | | | | | | | | | |
|------------------|------------|----|----|----|----|----|----|----|----------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Class Code | | | | | | | | Sub Class Code | | | | | | | |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Prog IF | | | | | | | | Revision ID | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|----------------|---------------|---------|------------------|--|
| 31 to 24 | Class Code | H'FF | R | R | Class Code Indicates the class code. The value set in the IDSETR1.Class code set field is reflected on these bits. |
| 23 to 16 | Sub Class Code | All 0 | R | R | Sub Class Code Indicates the sub-class code. The value set in the IDSETR1.Sub Class Code Set field is reflected on these bits. |
| 15 to 8 | Prog IF | All 0 | R | R | Prog IF Indicates the prog IF. The value set in the IDSETR1.PROG IF Set field is reflected on these bits. |
| 7 to 0 | Revision ID | All 0 | R | R | Revision ID Indicates the revision ID. The value set in the IDSETR1.Rev ID Set field is reflected on these bits. |

49.2.45 PCI Configuration Register 3 (PCICONF3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

| | | | | | | | | | | | | | | | | |
|------------------|----------------------|-------|----|----|---------|----|----|----|-----------------|-------------|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | BSTCA P | STBST | — | — | BSTCODE | | | | SFMF | Header Type | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Master Latency Timer | | | | | | | | Cache Line Size | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Internal bus R/W | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|----------|---------------|---------|------------------|--|
| 31 | BSTCAP | B'0 | R | R | BIST Capable Indicates the BIST function support status. 0: BIST function is not supported. 1: BIST function is supported. This IP does not support the BIST function. |
| 30 | STBST | B'0 | R | R | Start BIST Indicates the BIST function execution status. 0: BIST function has ended. 1: BIST function execution is in progress. This IP does not support the BIST function. |
| 29, 28 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 to 24 | BSTCODE | H'0 | R | R | BIST Completion Code Indicates the BIST completion status code. H'0: Passed BIST. H'1 to H'F: Failure has been detected. This IP does not support the BIST function. |
| 23 | SFMF | B'0 | R | R/W | Single Function/ Multi-function Indicates whether single function device or multi-function device is used. 0: Single function device 1: Multi -function device (setting prohibited) This IP supports only single function. This bit must not be set to a value other than 0. |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|----------------------|---------------|---------|------------------|--|
| 22 to 16 | Header Type | H'01 | R | R/W | <p>Header Type</p> <p>Specifies a layout of the configuration registers.</p> <p>H'00: Type00 layout</p> <p>H'01: Type01 layout</p> <p>H'02: Type02 layout (setting prohibited)</p> <ul style="list-style-type: none"> • Root Port: Set Type01. • Endpoint: Set Type00. <p>This IP supports Type00 and Type01, and does not support Type02.</p> |
| 15 to 8 | Master Latency Timer | All 0 | R | R | <p>Master Latency Timer</p> <p>Fixed to 0. These bits are not used for PCI Express.</p> |
| 7 to 0 | Cache Line Size | All 0 | R/W | R/W | <p>Cache Line Size</p> <p>These bits are not used for PCI Express functions.</p> <p>Updates the value on receiving a configuration write request for this field.</p> |

49.2.46 PCI Configuration Register 4 (PCICONF4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

| | | | | | | | | | | | | | | | | |
|------------------|------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | BAR0 | | | | | | | | | | | | | | | |
| Initial value: | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 |
| PCI R/W | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 |
| Internal bus R/W | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | BAR0 | | | | | | | | | | | | | | | |
| Initial value: | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 |
| PCI R/W | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 |
| Internal bus R/W | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 |

- Notes: 1. R/W changes by the state. RootPort:R/W.EndPoint:R
 2. Depends on the value of local address mask register 0.

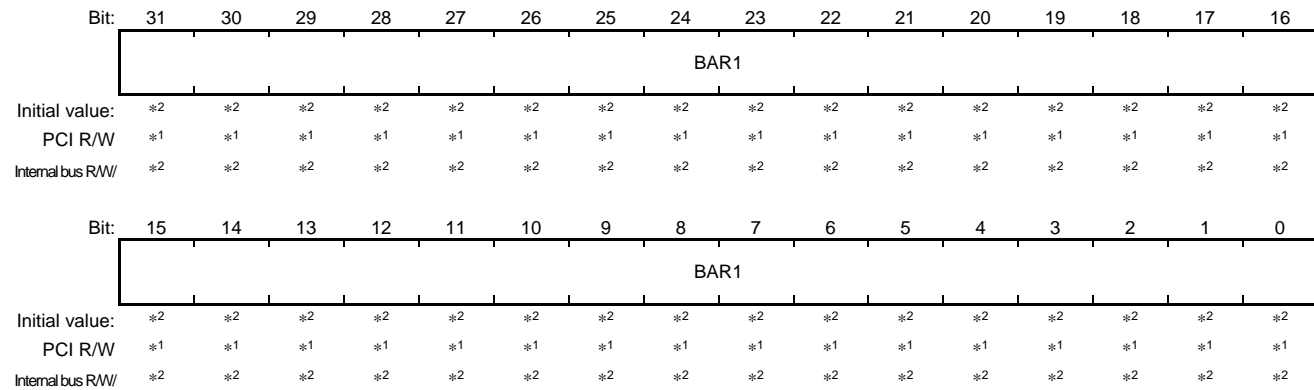
| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------|---------------|---------|------------------|---|
| 31 to 0 | BAR0 | *3 | R/W | *3 | BAR0 Indicates the value of base address register 0. <ul style="list-style-type: none"> Root Port: This field has no effect in Root Port mode. |
| | | | R | *3 | BAR0 Indicates the value of base address register 0. <ul style="list-style-type: none"> Endpoint: This field is set by local address mask register 0 in Endpoint mode. |

Notes: 3. It is the same as *2.

49.2.47 PCI Configuration Register 5 (PCICONF5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01



- Notes: 1. R/W changes by the state. RootPort:R/W.EndPoint:R
 2. Depends on the value of local address mask register 1.

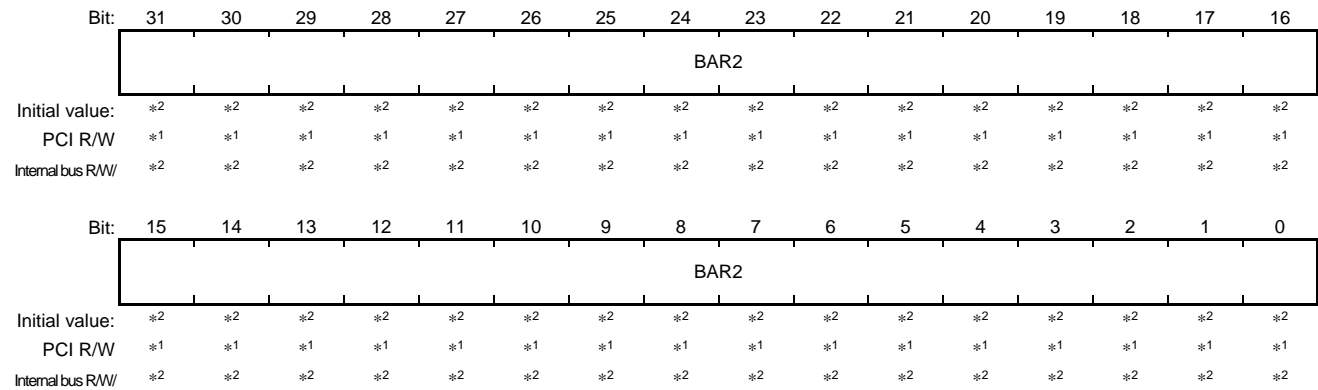
| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------|---------------|---------|------------------|---|
| 31 to 0 | BAR1 | *3 | R/W | *3 | BAR1 Indicates the value of base address register 1. <ul style="list-style-type: none"> Root Port: This field has no effect in Root Port mode. |
| | | | R | *3 | BAR1 Indicates the value of base address register 1. <ul style="list-style-type: none"> Endpoint: This field is set by local address mask register 1 in Endpoint mode. |

Notes: 3. It is the same as *2.

49.2.48 PCI Configuration Register 6 (PCICONF6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00

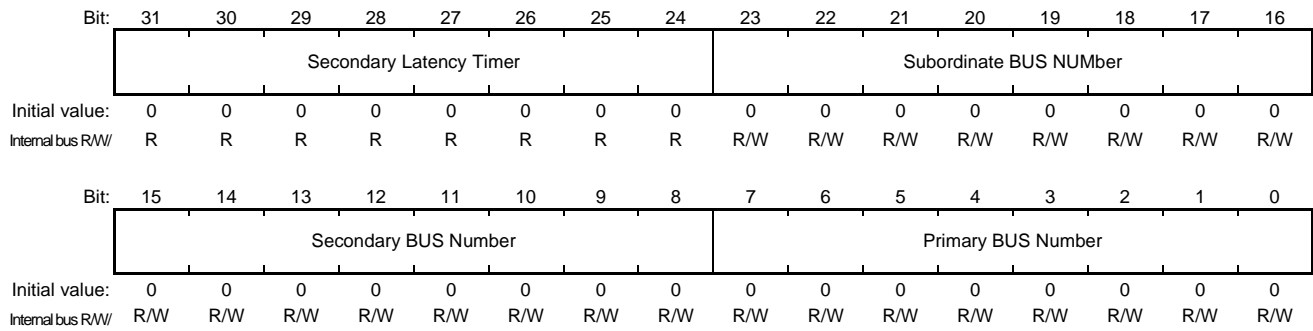


- Notes: 1. R/W changes by the state. RootPort:R/W.EndPoint:R
 2. Depends on the value of local address mask register 2.

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------|---------------|---------|------------------|---|
| 31 to 0 | BAR2 | *3 | R/W | *3 | BAR2 Indicates the value of base address register 2. <ul style="list-style-type: none"> Root Port: This field has no effect in Root Port mode. |
| | | | R | *3 | BAR2 Indicates the value of base address register 2. <ul style="list-style-type: none"> Endpoint: This field is set by local address mask register 2 in Endpoint mode. |

Notes: 3 It is the same as *2.

Header TYPE01

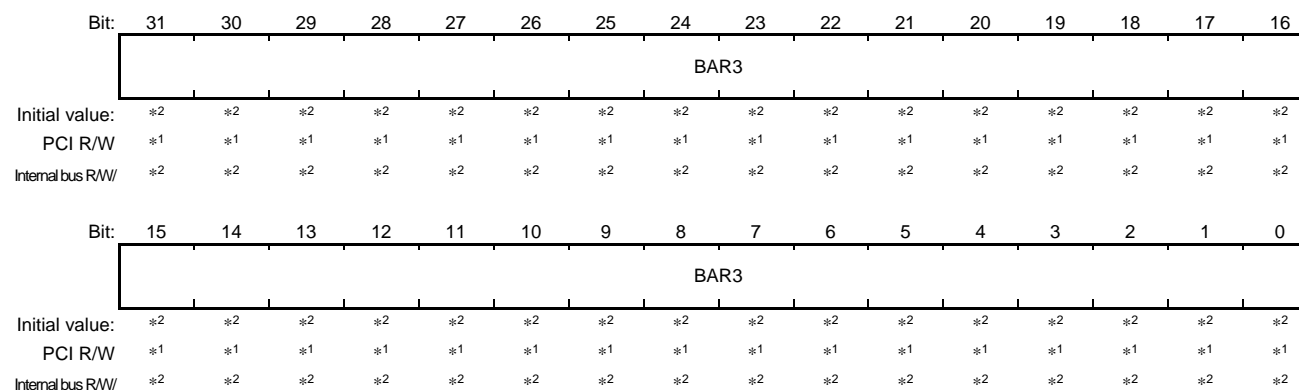


| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|----------|-------------------------|---------------|------------------|---|
| 31 to 24 | Secondary Latency Timer | All 0 | R | Secondary Latency Timer Fixed to 0. These bits are not used for PCI Express. |
| 23 to 16 | Subordinate BUS NUMBER | All 0 | R/W | Subordinate Bus Number According to the PCI standard, indicates the maximum bus number of the devices connected to the downstream link. In this IP, this field has no effect on the hardware. |
| 15 to 8 | Secondary BUS Number | All 0 | R/W | Secondary Bus Number According to the PCI standard, indicates the number of the bus directly connected to the secondary IF. In this IP, this field has no effect on the hardware. |
| 7 to 0 | Primary BUS Number | All 0 | R/W | Primary Bus Number According to the PCI standard, indicates the number of the bus directly connected to the primary IF. In this IP, this field has no effect on the hardware. |

49.2.49 PCI Configuration Register 7 (PCICONF7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00



- Notes: 1. R/W changes by the state. RootPort:R/W.EndPoint:R
 2. Depends on the value of local address mask register 3.

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------|---------------|---------|------------------|---|
| 31 to 0 | BAR3 | *3 | R/W | *3 | BAR3 Indicates the value of base address register 3. <ul style="list-style-type: none"> Root Port: This field has no effect in Root Port mode. |
| | | | R | *3 | BAR3 Indicates the value of base address register 3. <ul style="list-style-type: none"> Endpoint: This field is set by local address mask register 3 in Endpoint mode. |

Notes: 3 It is the same as *2.

Header TYPE01

- Secondary status register

| | | | | | | | | | | | | | | | | |
|-------------------|------|------|------|------|------|--------|------|-------------|----|------------|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | DPE | RSE | RMA | RTA | STA | DEVSEL | MDPE | FBBTC AP | — | 66MCA P | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W: | RW1C | RW1C | RW1C | RW1C | RW1C | R | R | RW1C | R | R | R | R | R | R | R | R |

| | | | | | | | | | | | | | | | | |
|-------------------|-------|-----|-----|-----|------|-----|-----|-----|-------|-----|-----|-----|------|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | IOLEA | | | | IOLT | | | | IOBEA | | | | IOBT | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|--------|----------|---------------|------------------|---|
| 31 | DPE | B'0 | RW1C | Detected Parity Error Indicates that a parity error has been detected on the secondary bus side. This bit is set to 1 upon reception of a poisoned TLP. |
| 30 | RSE | B'0 | RW1C | Received System Error Indicates that a system error has been detected. This bit is set to 1 upon reception of an ERR_FATAL/NON_FATAL message. |
| 29 | RMA | B'0 | RW1C | Received Master Abort Indicates that a master abort has been detected on the secondary bus side. This bit is set to 1 upon reception of a completion with the unsupported request completion status. |
| 28 | RTA | B'0 | RW1C | Received Target Abort Indicates that a target abort has been detected on the secondary bus side. This bit is set to 1 upon reception of a completion with the completer abort completion status. |
| 27 | STA | B'0 | RW1C | Signaled Target Abort Indicates that a target abort has been transmitted to the secondary bus side. This bit is set to 1 upon transmission of a completion with the completer abort completion status. |
| 26, 25 | DEVSEL | All 0 | R | DEVSEL Timing Fixed to 0. These bits are not used for PCI Express. |
| 24 | MDPE | B'0 | RW1C | MDPE Indicates that a master data parity error has occurred on the secondary bus side. This bit is set when either of the following conditions is satisfied while PCICONF15[16].PERS is 1. 1. A poisoned completion has been received. 2. A poisoned request has been transmitted. |
| 23 | FBTCA | B'0 | R | Fast Back to Back Transaction Capable Fixed to 0. This bit is not used for PCI Express. |
| 22 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|----------|----------|---------------|------------------|---|
| 21 | 66MCAP | B'0 | R | 66 MHz Capable This bit is always read as 0. The write value should always be 0. This bit is not used for PCI Express. |
| 20 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 12 | IOLEA | All 0 | R/W | IO Limit End Address According to the PCI standard, sets the address[15:12] of the upper limit address of the IO transaction to be transferred to the primary bus. In this IP, this field has no effect on the hardware. |
| 11 to 8 | IOLT | All 0 | R/W | IO Limit Type According to the PCI standard, indicates the upper limit address decoding format of the IO transactions to be transferred to the primary bus. H'0: 16-bit I/O address H'1: 32-bit I/O address In this IP, this field has no effect on the hardware. |
| 7 to 4 | IOBEA | All 0 | R/W | IO Base End Address According to the PCI standard, sets the address[15:12] of the base address of the IO transaction to be transferred to the primary bus. In this IP, this field has no effect on the hardware. |
| 3 to 0 | IOBT | All 0 | R/W | IO Base Type According to the PCI standard, indicates the base address decoding format of the IO transactions to be transferred to the primary bus. H'0: 16-bit I/O address H'1: 32-bit I/O address In this IP, this field has no effect on the hardware. |

49.2.50 PCI Configuration Register 8 (PCICONF8)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00

| | | | | | | | | | | | | | | | | |
|------------------|------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | BAR4 | | | | | | | | | | | | | | | |
| Initial value: | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 |
| PCI R/W | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 |
| Internal bus R/W | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | BAR4 | | | | | | | | | | | | | | | |
| Initial value: | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 |
| PCI R/W | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 |
| Internal bus R/W | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 |

- Notes: 1. R/W changes by the state. RootPort:R/W.EndPoint:R
 2. Depends on the value of local address mask register 4.

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------|---------------|---------|------------------|---|
| 31 to 0 | BAR4 | *3 | R/W | *3 | BAR4 Indicates the value of base address register 4. <ul style="list-style-type: none"> Root Port: This field has no effect in Root Port mode. |
| | | | R | *3 | BAR4 Indicates the value of base address register 4. <ul style="list-style-type: none"> Endpoint: This field is set by local address mask register 4 in Endpoint mode. |

Notes: 3. It is the same as *2.

Header TYPE01

| | | | | | | | | | | | | | | | | |
|-------------------|--------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|----|----|----|----|
| Bit: | 3 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Memory Limit | | | | | | | | | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Memory Base | | | | | | | | | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|----------|--------------|---------------|------------------|---|
| 31 to 20 | Memory Limit | All 0 | R/W | <p>Memory Limit</p> <p>According to the PCI standard, sets the upper limit address of the memory mapped IO performing transfer between the primary IF and the secondary IF.</p> <p>This field corresponds to address[31:20].</p> <p>In this IP, this field has no effect on the hardware.</p> |
| 19 to 16 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 15 to 4 | Memory Base | All 0 | R/W | <p>Memory Base</p> <p>According to the PCI standard, sets the base address of the memory mapped IO performing transfer between the primary IF and the secondary IF.</p> <p>This field corresponds to address[31:20].</p> <p>In this IP, this field has no effect on the hardware.</p> |
| 3 to 0 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

49.2.51 PCI Configuration Register 9 (PCICONF9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00

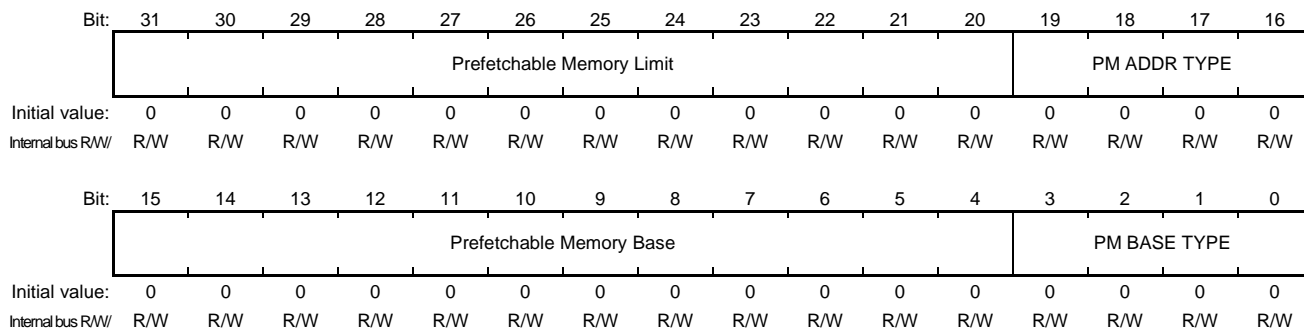
| | | | | | | | | | | | | | | | | |
|------------------|------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | BAR5 | | | | | | | | | | | | | | | |
| Initial value: | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 |
| PCI R/W | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 |
| Internal bus R/W | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | BAR5 | | | | | | | | | | | | | | | |
| Initial value: | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 |
| PCI R/W | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 |
| Internal bus R/W | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 | *2 |

- Notes: 1. R/W changes by the state. RootPort:R/W.EndPoint:R
 2. Depends on the value of local address mask register 5.

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------|---------------|---------|------------------|---|
| 31 to 0 | BAR5 | *3 | R/W | *3 | BAR5 Indicates the value of base address register 5. <ul style="list-style-type: none"> Root Port: This field has no effect in Root Port mode. |
| | | | R | *3 | BAR5 Indicates the value of base address register 5. <ul style="list-style-type: none"> Endpoint: This field is set by local address mask register 5 in Endpoint mode. |

Notes: 3. It is the same as *2.

Header TYPE01



| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|----------|---------------------------|---------------|------------------|--|
| 31 to 20 | Prefetchable Memory Limit | H'000 | R/W | <p>Prefetchable Memory Limit</p> <p>According to the PCI standard, sets the upper limit address of the prefetchable memory performing transfer between the primary IF and the secondary IF.</p> <p>This field corresponds to address[31:20]. In this IP, this field has no effect on the hardware.</p> |
| 19 to 16 | PM ADDR TYPE | H'0 | R/W | <p>Prefetchable Memory Address Decode TYPE</p> <p>According to the PCI standard, indicates the upper limit address decoding format of the prefetchable memory performing transfer between the primary IF and the secondary IF.</p> <p>H'0: 32-bit address H'1: 64-bit address</p> <p>In this IP, this field has no effect on the hardware.</p> |
| 15 to 4 | Prefetchable Memory Base | H'000 | R/W | <p>Prefetchable Memory Base</p> <p>According to the PCI standard, sets the base address of the prefetchable memory performing transfer between the primary IF and the secondary IF.</p> <p>This field corresponds to address[31:20]. In this IP, this field has no effect on the hardware.</p> |
| 3 to 0 | PM BASE TYPE | H'0 | R/W | <p>Prefetchable Memory Base Type</p> <p>According to the PCI standard, indicates the base address decoding format of the prefetchable memory performing transfer between the primary IF and the secondary IF.</p> <p>H'0: 32-bit address H'1: 64-bit address</p> <p>In this IP, this field has no effect on the hardware.</p> |

49.2.52 PCI Configuration Register 10 (PCICONF10)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00

| | | | | | | | | | | | | | | | | |
|------------------|----------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Card Bus CIS Pointer | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Card Bus CIS Pointer | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------------------|---------------|---------|------------------|--|
| 31 to 0 | Card Bus CIS Pointer | All 0 | R | R | Card Bus CIS Pointer According to the PCI standard, indicates the value of the card bus CIS pointer. This IP does not support the card bus CIS pointer. Fixed to 0. |

Header TYPE01

| | | | | | | | | | | | | | | | | |
|------------------|----------------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Prefetchable Memory Base Upper32 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Prefetchable Memory Base Upper32 | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|---------|----------------------------------|---------------|------------------|--|
| 31 to 0 | Prefetchable Memory Base Upper32 | All 0 | R/W | Prefetchable Memory Base Upper 32 According to the PCI standard, sets the upper limit address of the prefetchable memory performing transfer between the primary IF and the secondary IF. This field corresponds to address[63:32]. This field is valid when PCICONF9[3:0].PM BASE TYPE = H'1 (64-bit address decode format). In this IP, this field has no effect on the hardware. |

49.2.53 PCI Configuration Register 11 (PCICONF11)

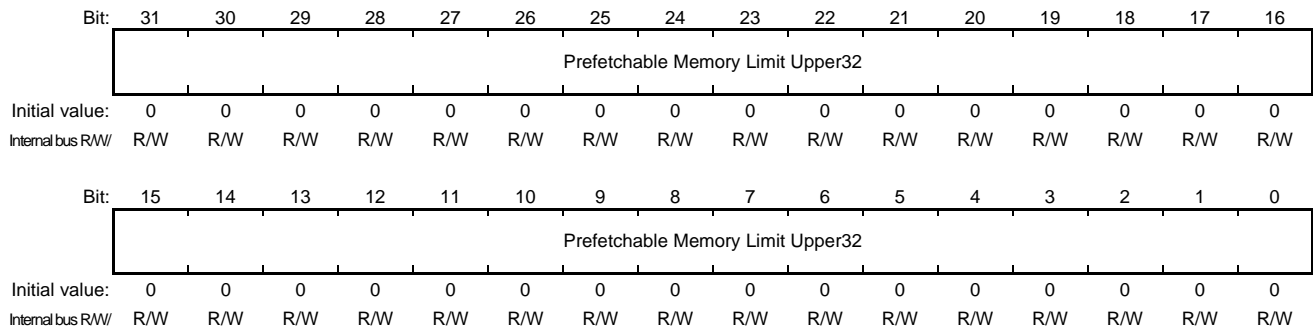
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00

| | | | | | | | | | | | | | | | | |
|------------------|----------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Sub System ID | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Sub System Vendor ID | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|----------------------|---------------|---------|------------------|--|
| 31 to 16 | Sub System ID | All 0 | R | R | Sub System ID I Indicates the sub-system ID. The value of SUBIDSETR.Sub System ID Set field is reflected on this field. |
| 15 to 0 | Sub System Vendor ID | All 0 | R | R | Sub System Vendor ID Indicates the sub-system vendor ID. The value of SUBIDSETR.Sub System Vendor ID Set field is reflected on this field. |

Header TYPE01



| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|---------|-----------------------------------|---------------|------------------|---|
| 31 to 0 | Prefetchable Memory Limit Upper32 | All 0 | R/W | <p>Prefetchable Memory Limit Upper 32</p> <p>According to the PCI standard, sets the base address of the prefetchable memory performing transfer between the primary IF and the secondary IF. This field corresponds to address[63:32].</p> <p>This field is valid when PCICONF9[19:16].PM ADDR TYPE = H'1 (64-bit address decode format).</p> <p>In this IP, this field has no effect on the hardware.</p> |

49.2.54 PCI Configuration Register 12 (PCICONF12)

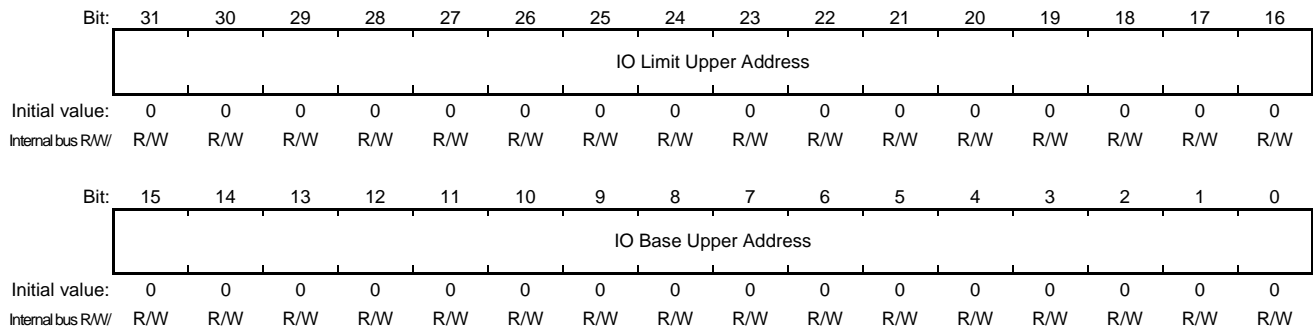
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00

| | | | | | | | | | | | | | | | | |
|------------------|-------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Expansion ROM BAR | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Expansion ROM BAR | | | | | | | | | | | | | | | EROME |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-------------------|---------------|---------|------------------|--|
| 31 to 11 | Expansion ROM BAR | All 0 | R | R | Expansion ROM BAR According to the PCI standard, indicates the value of expansion ROM base address register. This IP does not support expansion ROM. |
| 10 to 1 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | EROME | B'0 | R | R | Expansion ROM Enable According to the PCI standard, indicates that access to expansion ROM is enabled. This IP does not support expansion ROM. |

Header TYPE01



| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|----------|------------------------|---------------|------------------|--|
| 31 to 16 | IO Limit Upper Address | All 0 | R/W | <p>IO Limit Upper Address</p> <p>According to the PCI standard, sets the address[31:16] of the upper limit address of the IO transaction to be transferred to the primary bus.</p> <p>This field is valid when PCICONF7[11:8].IO Limit Type = 1 (32-bit address decode format).</p> <p>In this IP, this field has no effect on the hardware.</p> |
| 15 to 0 | IO Base Upper Address | All 0 | R/W | <p>IO Base Upper Address</p> <p>According to the PCI standard, sets the address[31:16] of the base address of the IO transaction to be transferred to the primary bus.</p> <p>This field is valid when PCICONF7[3:0].IO Base Type = 1 (32-bit address decode format).</p> <p>In this IP, this field has no effect on the hardware.</p> |

49.2.55 PCI Configuration Register 13 (PCICONF13)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----------------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | Capabilities Pointer | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------------------|---------------|---------|------------------|---|
| 31 to 8 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | Capabilities Pointer | H'40 | R | R/W | Capabilities Pointer Pointer to the extended capability list. This IP has the extended capability list and this field points to PCI PM capability pointer H'40. |

49.2.56 PCI Configuration Register 14 (PCICONF14)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------|---------------|---------|------------------|--|
| 31 to 0 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |

Header TYPE01

| | | | | | | | | | | | | | | | | |
|------------------|-------------------|----|----|----|----|----|----|----|----|----|----|----|----|-------|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Expansion ROM BAR | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Expansion ROM BAR | | | | | | | | | | | — | — | EROME | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|----------|-------------------|---------------|------------------|--|
| 31 to 11 | Expansion ROM BAR | All 0 | R | Expansion ROM BAR According to the PCI standard, indicates the value of expansion ROM base address register. This IP does not support Expansion ROM. |
| 10 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | EROME | B'0 | R | Expansion ROM Enable According to the PCI standard, indicates that access to expansion ROM is enabled. This IP does not support expansion ROM. |

49.2.57 PCI Configuration Register 15 (PCICONF15)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00

| | | | | | | | | | | | | | | | | |
|------------------|---------------|-----|-----|-----|-----|-----|-----|-----|----------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MAXLAT | | | | | | | | MINGNT | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Interrupt Pin | | | | | | | | Interrupt Line | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| PCI R/W | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Internal bus R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|----------------|---------------|---------|------------------|--|
| 31 to 24 | MAXLAT | All 0 | R | R | MAXLAT Fixed to 0. These bits are not used for PCI Express. |
| 23 to 16 | MINGNT | All 0 | R | R | MINGNT Fixed to 0. These bits are not used for PCI Express. |
| 15 to 8 | Interrupt Pin | All 0 | R | R/W | Interrupt Pin Specifies the legacy interrupt message used by the device at initialization. When no legacy interrupts are used, set 0. H'0: Legacy interrupt is not used. H'1: INTA is used. H'2: INTB is used. (Setting prohibited) H'3: INTC is used. (Setting prohibited) H'4: INTD is used. (Setting prohibited) This field must not be set to a value other than 0 and 1. |
| 7 to 0 | Interrupt Line | H'FF | R/W | R/W | Interrupt Line Indicates information on legacy interrupt routing. |

Header TYPE01

- Bridge control register

| | | | | | | | | | | | | | | | | |
|-------------------|---------------|-----|-----|-----|-------|-------|-------|------|----------------|-------------|-----------|--------------|------|------|-------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | DTSEE | DTSTS | SECDT | PRDT | FBBTC AP | SECBR ST | MABM D | VGA16 DEC | VGAE | ISAE | SERRE | PERS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W/ | R | R | R | R | R | R | R | R | R | R/W | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Interrupt Pin | | | | | | | | Interrupt Line | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Internal bus R/W/ | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|----------|----------|---------------|------------------|---|
| 31 to 28 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 | DTSEE | B'0 | R | Discard Timer SERR Enable This bit is always read as 0. The write value should always be 0. This bit is not used for PCI Express. |
| 26 | DTSTS | B'0 | R | Discard Timer Status This bit is always read as 0. The write value should always be 0. This bit is not used for PCI Express. |
| 25 | SECDT | B'0 | R | Secondary Discard Timer This bit is always read as 0. The write value should always be 0. This bit is not used for PCI Express. |
| 24 | PRDT | B'0 | R | Primary Discard Timer This bit is always read as 0. The write value should always be 0. This bit is not used for PCI Express. |
| 23 | FBBTCAP | B'0 | R | Fast Back to Back Transaction Capable This bit is always read as 0. The write value should always be 0. This bit is not used for PCI Express. |
| 22 | SECBRST | B'0 | R/W | Secondary Bus Reset Resets the secondary IF. Setting this bit to 1 causes the LTSSM to make a transition to the hot reset state. Clearing this bit cancels the hot reset state of LTSSM. |
| 21 | MABMD | B'0 | R | Master Abort Mode This bit is always read as 0. The write value should always be 0. This bit is not used for PCI Express. |
| 20 | VGA16DEC | B'0 | R/W | VGA 16-bit Decode According to the PCI standard, enables VGA IO decoding. In this IP, this field has no effect on the hardware. |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|---------|----------------|---------------|------------------|--|
| 19 | VGAE | B'0 | R/W | <p>VGA Enable</p> <p>According to the PCI standard, enables VGA address translation.</p> <p>In this IP, this field has no effect on the hardware.</p> |
| 18 | ISAE | B'0 | R/W | <p>ISA Enable</p> <p>According to the PCI standard, enables ISA IO address translation.</p> <p>In this IP, this field has no effect on the hardware.</p> |
| 17 | SERRE | B'0 | R/W | <p>SERR Enable</p> <p>Enables ERR_FATAL/ERR_NONFATAL/ERR_COR message transfer (notification) from the secondary IF to the primary IF.</p> <p>1: Enables error message transfer (notification). 0: Disables error message transfer (notification).</p> |
| 16 | PERS | B'0 | R/W | <p>Parity Error Response</p> <p>Enables setting of the PCICONF7[24].MDPE bit.</p> |
| 15 to 8 | Interrupt Pin | All 0 | R/W | <p>Interrupt Pin</p> <p>Specifies a legacy interrupt message to be used by the device. When no legacy interrupts are used, set 0.</p> |
| 7 to 0 | Interrupt Line | H'FF | R/W | <p>Interrupt Line</p> <p>According to the PCI standard, indicates information on legacy interrupt routing.</p> <p>According to this IP, the set values are defined as follows.</p> <p>H'00 - H'FE: INTx interrupt is reported to INTC. H'FF: INTx interrupt is not reported to INTC.</p> |

49.2.58 PCI Power Management Capability Register 0 (PMCAP0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

| | | | | | | | | | | | | | | | | |
|------------------|-------------------------|----------------|-------------|-------------|-------------|------|------|-------------|---------------|-----|-----|-----|------------|----------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | PMESP D3COLD | PMESP D3HOT | PMESP D2 | PMESP D1 | PMESP D0 | D2SP | D1SP | AUX Current | | | DSI | — | PMECL K | PCI PM Version | | |
| Initial value: | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Next Capability Pointer | | | | | | | | Capability ID | | | | | | | |
| Initial value: | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|-----|-------------|---------------|---------|------------------|---|
| 31 | PMESPD3COLD | B'1 | R | R/W | <p>PME Support D3COLD</p> <p>Indicates whether PME generation is supported or not in the D3Cold state.</p> <p>0: Does not support PME generation in the D3Cold state. 1: Supports PME generation in the D3Cold state.</p> <ul style="list-style-type: none"> • Root Port: Set this bit to 1 at initialization. • Endpoint: Set the device function at initialization. |
| 30 | PMESPD3HOT | B'1 | R | R/W | <p>PME Support D3HOT</p> <p>Indicates whether PME generation is supported or not in the D3Hot state.</p> <p>0: Does not support PME generation in the D3Hot state. 1: Supports PME generation in the D3Hot state.</p> <ul style="list-style-type: none"> • Root Port: Set this bit to 1 at initialization. • Endpoint: Set the device function at initialization. |
| 29 | PMESPD2 | B'0 | R | R/W | <p>PME Support D2</p> <p>Indicates whether PME generation is supported or not in the D2 state.</p> <p>0: Does not support PME generation in the D2 state. 1: Supports PME generation in the D2 state.</p> <p>Set the device function at initialization.</p> |
| 28 | PMESPD1 | B'0 | R | R/W | <p>PME Support D1</p> <p>Indicates whether PME generation is supported or not in the D1 state.</p> <p>0: Does not support PME generation in the D1 state. 1: Supports PME generation in the D1 state.</p> <p>Set the device function at initialization.</p> |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-------------|---------------|---------|------------------|---|
| 27 | PMESPD0 | B'1 | R | R/W | <p>PME Support D0</p> <p>Indicates whether PME generation is supported or not in the D0 state.</p> <p>0: Does not support PME generation in the D0 state. 1: Supports PME generation in the D0 state.</p> <ul style="list-style-type: none"> • Root Port: Set this bit to 1 at initialization. • Endpoint: Set the device function at initialization. |
| 26 | D2SP | B'0 | R | R/W | <p>D2 Support</p> <p>Indicates whether the D2 power management state is supported or not.</p> <p>0: Does not support the D2 state. 1: Supports the D2 state.</p> <p>Set the device function at initialization.</p> |
| 25 | D1SP | B'0 | R | R/W | <p>D1 Support</p> <p>Indicates whether the D1 power management state is supported or not.</p> <p>0: Does not support the D1 state. 1: Supports the D1 state.</p> <p>Set the device function at initialization.</p> |
| 24 to 22 | AUX Current | H'0 | R | R/W | <p>AUX Current</p> <p>Indicates the current of 3.3V auxiliary power supply. Set this field at initialization.</p> <p>H'0: 0 mA (without auxiliary power supply) H'1: 55 mA H'2: 100 mA H'3: 160 mA H'4: 220 mA H'5: 270 mA H'6: 320 mA H'7: 375 mA</p> <p>When PMCAP0[31].PMESPD3COLD = 0, use the device with this field set to the initial value (= 0).</p> |
| 21 | DSI | B'0 | R | R/W | <p>Device Specific Initialization</p> <p>Indicates that the device-specific initialization sequence is required after D0 initialization.</p> <p>0: The device-specific initialization sequence is not necessary. 1: The device-specific initialization sequence is required.</p> <p>Set this field at initialization.</p> |
| 20 | — | B'0 | R | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 19 | PMECLK | B'0 | R | R | <p>PME Clock</p> <p>This bit is always read as 0. The write value should always be 0.</p> <p>This bit is not used for PCI Express.</p> |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-------------------------|---------------|---------|------------------|---|
| 18 to 16 | PCI PM Version | H'3 | R | R/W | PCI PM Version Indicate the corresponding PCI PM Interface Specification version. This device corresponds to the PCI PM Interface Specification version 1.2. |
| 15 to 8 | Next Capability Pointer | H'50 | R | R/W | Next Capability Pointer Pointer to the extended capability list This field points to MSI capability pointer H'50. |
| 7 to 0 | Capability ID | H'01 | R | R/W | Capability ID Capability List ID Indicates the power management capability (H'01). |

49.2.59 PCI Power Management Capability Register 1 (PMCAP1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

| | | | | | | | | | | | | | | | | | |
|------------------|---------------|------------|-------------|----|----|----|------|-----------|-------|--------|----|---------|-----|-------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | DATA | | | | | | | | BPCCE | B2B3SP | — | — | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Internal bus R/W | R | R | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | PMEST | Data Scale | Data Select | | | | PMEE | — | — | — | — | NOS RST | — | Power State | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| PCI R/W | RW1C or RW1CS | R | R | R | R | R | R | RW or RWS | R | R | R | R | R | R | R | R/W | R/W |
| Internal bus R/W | RW1C | R | R | R | R | R | R | R/W | R | R | R | R | R/W | R | R/W | R/W | |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|----------|---------------|---------------|------------------|---|
| 31 to 24 | DATA | All 0 | R | R | DATA According to the PCI standard, indicates the value selected with PMCAP1[12:0].Data Select. This IP does not support this field. |
| 23 | BPCCE | B'0 | R | R/W | Bus Power Clock Control Enable Enables the PCI bus power/clock control function. 0: Disables the PCI bus power/clock control function. 1: Enables the PCI bus power/clock control function. Set this field to 0 at initialization. |
| 22 | B2B3SP | B'0 | R | R/W | B2 B3 Support Indicates the bus state in the D3Hot state. This field is valid when PMCAP1[23].BPCCE = 1. 0: Clock supply to the secondary IF does not stop in the D3Hot state. (B3) 1: Clock supply to the secondary IF stops in the D3Hot state. (B2) Set this field to 0 at initialization. |
| 21 to 16 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | PMEST | B'0 | RW1C or RW1CS | RW1C | PME Status This field is valid when the PME is supported. Indicates the PME generation status. 0: A PME message is not being transmitted. 1: A PME message is being transmitted. This bit is set to 1 when a PME message is transmitted. Note: When PME generation is enabled in the D3Cold state, this field is RW1CS. |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|-------------|---------------|------------|------------------|--|
| 14, 13 | Data Scale | All 0 | R | R | Data Scale According to the PCI standard, indicates the scale used to indicate the value selected with PMCAP1[12:9].Data Select in PMCAP1[31:24].DATA. This IP does not support this field. |
| 12 to 9 | Data Select | All 0 | R | R | Data Select According to the PCI standard, sets the value and unit to be indicated in PMCAP1[31:24].DATA and PMCAP1[14:13].Data Scale. This IP does not support this field. |
| 8 | PMEE | B'0 | R/W or RWS | R/W | PME Enable Enables PME message transmission. 0: Disables PME message transmission. 1: Enables PME message transmission. When the device does not support the PME message transmission function, this bit should always be 0. Note: When PME generation is enabled in the D3Cold state, this field is RWS. |
| 7 to 4 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 | NOSRST | B'0 | R | R/W | No Soft Reset Indicates whether internal reset is performed or not when the device makes a transition from the D3Hot state to the D0 state. 0: Internal reset is performed when the device makes a transition from D3Hot state to D0 state. 1: Internal reset is not performed when the device makes a transition from D3Hot state to D0 state. Set the device function at initialization. |
| 2 | — | B'0 | R | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 1, 0 | Power State | B'00 | R/W | R/W | Power State Specifies the power state. B'00: D0 B'01: D1 B'10: D2 B'11: D3Hot Note: The setting is ignored when 1 is set with D1 unsupported or 2 is set with D2 unsupported. |

49.2.60 MSI Capability Register 0 (MSICAP0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00 only

| | | | | | | | | | | | | | | | | |
|------------------|-------------------------|-----|-----|-----|-----|-----|-----|-------|---------------|-------|-----|-----|---------|-----|-----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | PVMSK | 64ADC AP | MMESE | | | MMESCAP | | | MSIE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R/W |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Next Capability Pointer | | | | | | | | Capability ID | | | | | | | |
| Initial value: | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

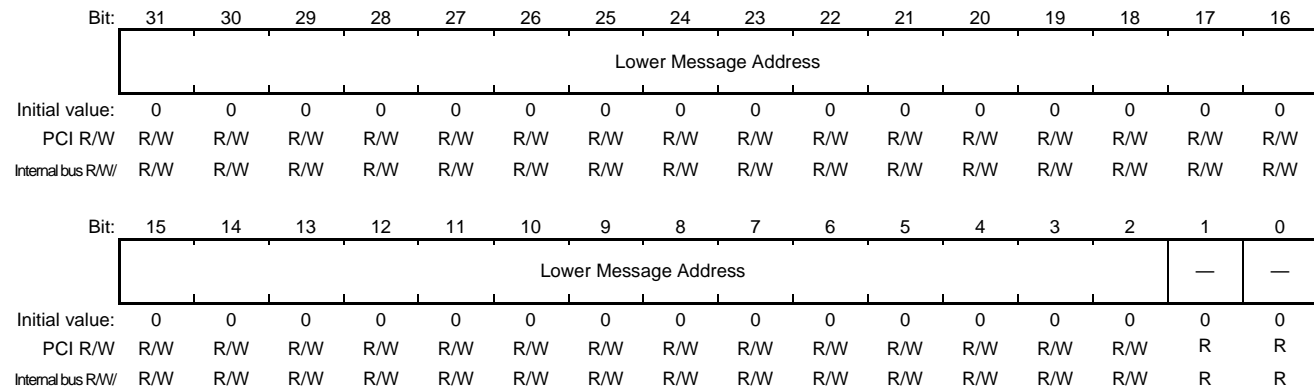
| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|----------|---------------|---------|------------------|--|
| 31 to 25 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | PVMSK | B'1 | R | R | Per Vector Masking Indicates whether the per vector masking function is supported or not. 0: The per vector masking function is not supported. 1: The per vector masking function is supported. This device supports the per vector masking function. |
| 23 | 64ADCAP | B'1 | R | R | 64-bit Address Capable Indicates whether transmission of a 64-bit address message is supported or not. 0: A 64-bit address message cannot be transmitted. 1: A 64-bit address message can be transmitted. This device supports transmission of a 64-bit address message. |
| 22 to 20 | MMESE | B'000 | R/W | R/W | Multiple Message Enable Sets the number of interrupt vectors that can be transmitted. B'000: 1 vector B'001: 2 vectors B'010: 4 vectors B'011: 8 vectors B'100: 16 vectors B'101: 32 vectors B'110: Reserved B'111: Reserved |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-------------------------|---------------|---------|------------------|--|
| 19 to 17 | MMESCAP | B'000 | R | R/W | <p>Multiple Message Capable</p> <p>Sets the number of interrupt vectors that a device can transmit.</p> <p>B'000: 1 vector B'001: 2 vectors B'010: 4 vectors B'011: 8 vectors B'100: 16 vectors B'101: 32 vectors B'110: Reserved B'111: Reserved</p> <p>Set the device function at initialization.</p> |
| 16 | MSIE | B'0 | R/W | R/W | <p>MSI Enable</p> <p>Enables the MSI function.</p> <p>0: Disables the MSI function. 1: Enables the MSI function.</p> |
| 15 to 8 | Next Capability Pointer | H'70 | R | R/W | <p>Next Capability Pointer</p> <p>Pointer to the extended capability list.</p> <p>Points to the PCI Express capability pointer H'70.</p> |
| 7 to 0 | Capability ID | H'05 | R | R/W | <p>Capability ID</p> <p>Capability list ID</p> <p>Indicates the MSI capability ID (H'05).</p> |

49.2.61 MSI Capability Register 1 (MSICAP1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00 only

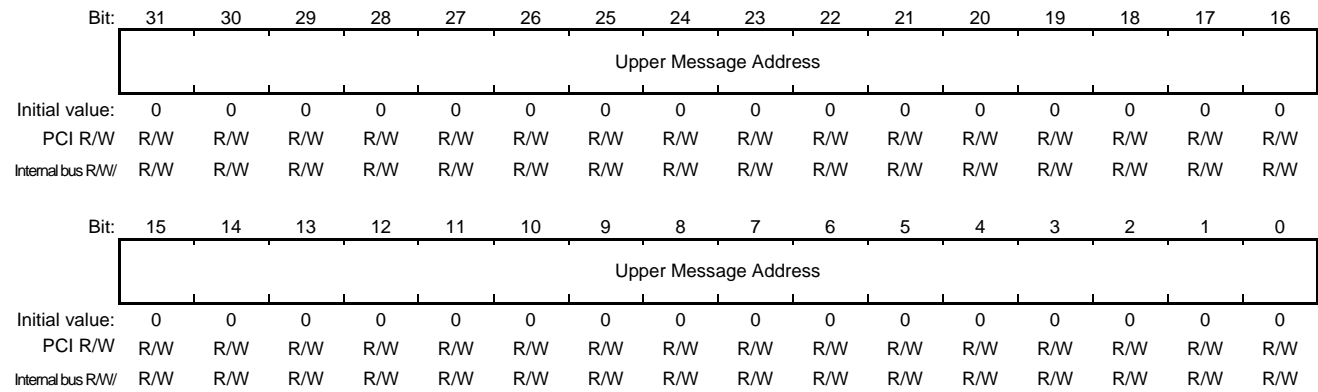


| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|-----------------------|---------------|---------|------------------|---|
| 31 to 2 | Lower Message Address | All 0 | R/W | R/W | Lower Message Address Specifies the address[31:2] of an MSI message during an MSI transmission. The value of this field is used for transmission of an MSI message. |
| 1, 0 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.62 MSI Capability Register 2 (MSICAP2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00 only



| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|-----------------------|---------------|---------|------------------|--|
| 31 to 0 | Upper Message Address | All 0 | R/W | R/W | Upper Message Address Specifies the address[63:32] of an MSI message during an MSI transmission. The value of this field is used for transmission of an MSI message. |

49.2.63 MSI Capability Register 3 (MSICAP3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00 only

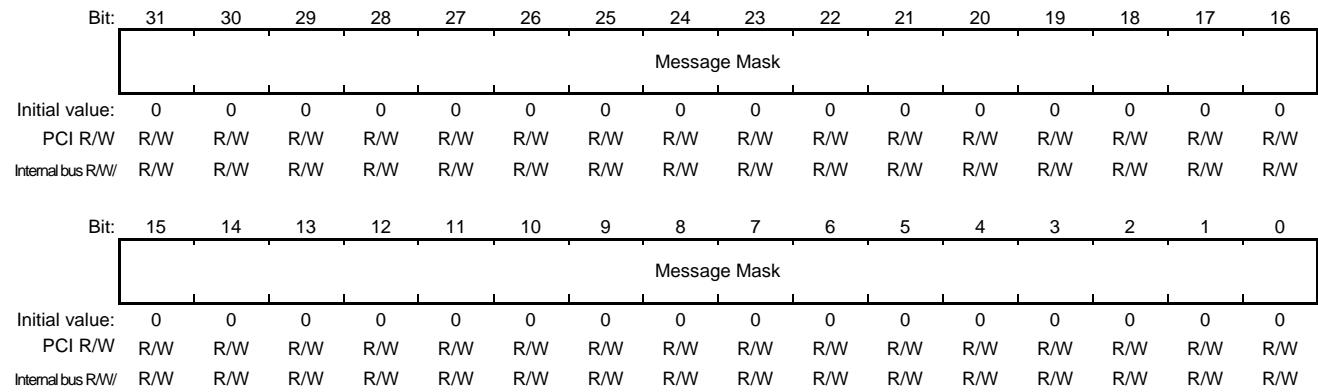
| | | | | | | | | | | | | | | | | |
|------------------|--------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Message Data | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Internal bus R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|--------------|---------------|---------|------------------|--|
| 31 to 16 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | Message Data | All 0 | R/W | R/W | Message Data Specifies the MSI message data during an MSI transmission. The value of this field is used for transmission of an MSI message. When more than one interrupt vector is assigned to this device through MSICAP0[22:20].MMESE, a quantity of transmit data equal to the assigned number of vectors can be changed. |

49.2.64 MSI Capability Register 4 (MSICAP4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00 only



| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|--------------|---------------|---------|------------------|--|
| 31 to 0 | Message Mask | All 0 | R/W | R/W | <p>Masks MSI transmission.</p> <p>Masks the MSI transmission of the interrupt vector assigned by MSICAP0[22:20].MMESE.</p> <p>Message Mask[N] = 1: Masks the MSI transmission of interrupt vector [N].</p> <p>Message Mask[N] = 0: Does not mask the MSI transmission of interrupt vector [N].</p> |

49.2.65 MSI Capability Register 5 (MSICAP5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00 only

| | | | | | | | | | | | | | | | | |
|------------------|-----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Message Pending | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Message Pending | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|-----------------|---------------|---------|------------------|---|
| 31 to 0 | Message Pending | All 0 | R | R | <p>Message Pending</p> <p>Indicates that MSI transmission is pending.</p> <p>If a transmission request of the interrupt vector corresponding to bit [N] is generated while it is masked by MSICAP4.Message Mask[N] = 1, MSICAP5.Message Pending[N] is set to 1 to indicate that transmission is pending.</p> <p>When MSICAP4.Message Mask[N] is updated to 0 upon reception of a configuration write request, MSICAP5.Message Pending[N] is updated to 0 to transmit the MSI message.</p> <p>This field is set to 1 when an interrupt transmission request is detected while MSICAP4.Message Mask[N] = 1.</p> <p>When MSICAP4.Message Mask[N] is cleared to 0 while MSICAP5.Message Pending[N] = 1, MSICAP5.Message Pending[N] is cleared to 0, and at the same time an MSI message is transmitted.</p> |

49.2.66 PCI Express Capability Register 0 (EXPCAP0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

| | | | | | | | | | | | | | | | | |
|------------------|-------------------------|-----|--------------------------|-----|-----|-----|--------|------------------|---------------|-----|-----|--------------------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | Interrupt Message Number | | | | SLTIMP | Device Port Type | | | | Capability Version | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Next Capability Pointer | | | | | | | | Capability ID | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|--------------------------|---------------|---------|------------------|---|
| 31, 30 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 to 25 | Interrupt Message Number | All 0 | R | R/W | Interrupt Message Number According to the PCI standard, indicates the offset of the interrupt vectors used for HotPlug MSI or PME MSI. This IP does not support this field. The write value should always be 0. |
| 24 | SLTIMP | B'0 | R | R | Slot Implemented Indicates whether the device link is connected to the PCI Express slot. 0: Not connected to the slot. 1: Connected to the slot. <ul style="list-style-type: none"> Root Port: This IP does not support the slot function. In this IP, this field has no effect on the hardware. Endpoint: Use the initial value. |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-------------------------|---------------|---------|------------------|--|
| 23 to 20 | Device Port Type | H'4 | R | R/W | <p>Device Port Type</p> <p>Indicates the device type.</p> <p>H'0: PCI Express Endpoint</p> <p>H'1: (Legacy PCI Express Endpoint) Setting prohibited</p> <p>H'4: Root Port of PCI Express Root Complex (initial value)</p> <p>H'5: (Upstream Port of PCI Express Switch) Setting prohibited</p> <p>H'6: (Downstream Port of PCI Express Switch) Setting prohibited</p> <p>H'7: (PCI Express to PCI/PCI-X Bridge) Setting prohibited</p> <p>H'8: (PCI/PCI-X to PCI Express Bridge) Setting prohibited</p> <p>H'9: (Root Complex Integrated Endpoint Device) Setting prohibited</p> <p>H'A: (Root Complex Event Collector) Setting prohibited</p> <ul style="list-style-type: none"> • Root Port: Set H'4 at initialization. • Endpoint: Set H'0 at initialization. |
| 19 to 16 | Capability Version | H'2 | R | R | <p>Capability Version</p> <p>Indicates the version of this capability.</p> |
| 15 to 8 | NEXT Capability Pointer | H'00 | R | R/W | <p>NEXT Capability Pointer</p> <p>Pointer to the extension capability list</p> <p>Indicates H'00, which is End Of List.</p> |
| 7 to 0 | Capability ID | H'10 | R | R/W | <p>Capability ID</p> <p>Capability List ID</p> <p>Indicates the PCI Express capability ID (H'10).</p> |

49.2.67 PCI Express Capability Register 1 (EXPCAP1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

- Device capability register

| | | | | | | | | | | | | | | | | | |
|------------------|------|----|----|------------|-----------|-----|-----|---------------------------------|-----|-----|-------|-----|----|------|-----|-----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | FLRCA P | CAPSLPLSC | | | Captured Slot Power Limit Value | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Internal bus R/W | R | R | R | R/W | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | RBER | — | — | — | EL1ACLAT | | | EL0ACLAT | | | ETAGS | PFS | | MPSS | | | |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Internal bus R/W | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|---------------------------------|---------------|---------|------------------|---|
| 31 to 29 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 | FLRCAP | B'0 | R | R/W | Function Level Reset Capability Indicates that the function level reset function is supported. <ul style="list-style-type: none"> • Root Port: Not need to be set. • Endpoint: This IP does not support the function level reset function. This bit should be fixed to 0. |
| 27, 26 | CAPSLPLSC | B'00 | R | R | Captured Slot Power Limit Scale Indicates the slot power limit value scale. B'00: 1.0x B'01: 0.1x B'10: 0.01x B'11: 0.001x <ul style="list-style-type: none"> • Root Port: Not need to be set. • Endpoint: This field is updated on reception of the Set_Slot_Power_Limit message. |
| 25 to 18 | Captured Slot Power Limit Value | H'00 | R | R | Captured Slot Power Limit Value Indicates the limit of power supplied from the slot (in Watts) in combination with EXPCAP1[27:26].CAPSLPLSC. <ul style="list-style-type: none"> • Root Port: Not need to be set. • Endpoint: This field is updated on reception of the Set_Slot_Power_Limit message. |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|----------|---------------|---------|------------------|--|
| 17, 16 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | RBER | B'1 | R | R | Role-Based Error Reporting The value is fixed to 1 according to the PCI Express version 1.1 or later. |
| 14 to 12 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 9 | EL1ACLAT | B'000 | R | R/W | Endpoint L1 Acceptable Latency Indicates the maximum L1 to L0 transition latency that the device can accept. B'000: L0 recovery must be completed within 1 μ s. B'001: L0 recovery must be completed within 2 μ s. B'010: L0 recovery must be completed within 4 μ s. B'011: L0 recovery must be completed within 8 μ s. B'100: L0 recovery must be completed within 16 μ s. B'101: L0 recovery must be completed within 32 μ s. B'110: L0 recovery must be completed within 64 μ s. B'111: No time limit Set the device support status at initialization. |
| 8 to 6 | EL0ACLAT | B'000 | R | R/W | Endpoint L0s Acceptable Latency Indicates the maximum L0s to L0 transition latency that the device can accept. B'000: L0 recovery must be completed within 64 ns. B'001: L0 recovery must be completed within 128 ns. B'010: L0 recovery must be completed within 256 ns. B'011: L0 recovery must be completed within 512 ns. B'100: L0 recovery must be completed within 1 μ s. B'101: L0 recovery must be completed within 2 μ s. B'110: L0 recovery must be completed within 4 μ s. B'111: No time limit Set the device support status at initialization. |
| 5 | ETAGS | B'1 | R | R | Extended Tag Field Supported Indicates the tag ID size supported as the requester ID. 0: 5-bit tag 1: 8-bit extension tag This IP supports the extension 8-bit tag ID function. |
| 4, 3 | PFS | B'00 | R | R | Phantom Function Supported Indicates the phantom function support status for transaction ID extension. B'00: The phantom function is not supported. B'01: The MSB of requester ID function number can be used for the phantom function. B'10: The upper two bits of requester ID function number can be used for the phantom function. B'11: All the three bits of requester ID function number can be used for the phantom function. This IP does not support the phantom function. Fixed to 0. |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|--------|----------|---------------|---------|------------------|---|
| 2 to 0 | MPSS | B'000 | R | R/W | Max Payload Size Supported Indicates the maximum payload size supported by the device. B'000: 128 bytes B'001: Reserved (setting prohibited) B'010: Reserved (setting prohibited) B'011: Reserved (setting prohibited) B'100: Reserved (setting prohibited) B'101: Reserved (setting prohibited) B'110: Reserved (setting prohibited) B'111: Reserved (setting prohibited) |

49.2.68 PCI Express Capability Register 2 (EXPCAP2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

- Device status register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|------|--------------|------------|------------|-------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | TRPD | AUXPD TCD | URDTC D | FEDTC D | NFEDT CD | CEDTC D |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | RW1C | RW1C | RW1C | RW1C |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | RW1C | RW1C | RW1C | RW1C |

- Device control register

| | | | | | | | | | | | | | | | | |
|------------------|----|-----------------------|-----|-----|-------|-------------|-----|-------|------------------|-----|-----|-------|-------|-------|------------|-------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | Max Read Request Size | | | ENSNP | AUXPP ME | PFE | ETAGE | Max Payload Size | | | ERLOD | URRPE | FERPE | NFERP E | CERPE |
| Initial value: | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R/W | R/W | R/W | R/W | RWS | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Internal bus R/W | R | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-----------|---------------|---------|------------------|--|
| 31 to 22 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 | TRPD | B'0 | R | R | Transaction Pending Indicates that there is a non-posted request not yet completed. 0: Non-posted request processing is not pending. 1: Non-posted request processing is pending. Non-posted requests issued are managed and the applicable value is reflected here. |
| 20 | AUXPDTC D | B'0 | R | R | AUX Power Detected This field is fixed to 0 with this IP. |
| 19 | URDTC D | B'0 | RW1C | RW1C | Unsupported Request Detected Indicates the unsupported request detection status. 0: An unsupported request has not been detected. 1: An unsupported request has been detected. This field is set to 1 when an unsupported request is detected. |
| 18 | FEDTC D | B'0 | RW1C | RW1C | Fatal Error Detected Indicates the fatal error detection status. 0: A fatal error has not been detected. 1: A fatal error has been detected. This field is set to 1 when a fatal error is detected. |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-----------------------|---------------|---------|------------------|--|
| 17 | NFEDTCD | B'0 | RW1C | RW1C | <p>Non-fatal Error Detected</p> <p>Indicates the non-fatal error detection status.</p> <p>0: A non-fatal error has not been detected.</p> <p>1: A non-fatal error has been detected.</p> <p>This field is set to 1 when a non-fatal error is detected.</p> |
| 16 | CEDTCD | B'0 | RW1C | RW1C | <p>Correctable Error Detected</p> <p>Indicates the correctable error detection status.</p> <p>0: A correctable error has not been detected.</p> <p>1: A correctable error has been detected.</p> <p>This field is set to 1 when a correctable error is detected.</p> |
| 15 | Reserved | B'0 | R | R | This bit is always read as 0. The write value should always be 0. |
| 14 to 12 | Max Read Request Size | B'010 | R/W | R/W | <p>Max Read Request Size</p> <p>Indicates the maximum size of the read request that can be issued as the requester.</p> <p>B'000: 128 bytes max.</p> <p>B'001: 256 bytes max.</p> <p>B'010: 512 bytes max.</p> <p>B'011: 1024 bytes max.</p> <p>B'100: 2048 bytes max.</p> <p>B'101: 4096 bytes max.</p> <p>B'110: Reserved</p> <p>B'111: Reserved</p> |
| 11 | ENSNP | B'1 | R/W | R/W | <p>Enable No Snoop</p> <p>Enables the device to issue the no-snoop transaction.</p> <p>0: The no-snoop transaction cannot be issued.</p> <p>1: The no-snoop transaction can be issued.</p> |
| 10 | AUXPPME | B'0 | RWS | R/W | <p>AUX Power PM Enable</p> <p>Enables AUX power power management.</p> <p>0: AUX power power management is disabled.</p> <p>1: AUX power power management is enabled.</p> <p>Note: When the AUX power function is not implemented (when PMCAP0[24:22].AUX Current = 0), this bit is fixed to 0.</p> |
| 9 | PFE | B'0 | R | R | <p>Phantom Function Enable</p> <p>Enables the phantom function.</p> <p>0: The phantom function is disabled.</p> <p>1: The phantom function is enabled.</p> <p>This IP does not support the phantom function.</p> <p>Fixed to 0.</p> |
| 8 | ETAGE | B'0 | R/W | R/W | <p>Extended Tag Enable</p> <p>Enables the extension 8-bit tag ID function.</p> <p>0: The extension 8-bit tag ID function is disabled.</p> <p>1: The extension 8-bit tag ID function is enabled.</p> <p>This ID supports the extension 8-bit tag ID function.</p> |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|--------|------------------|---------------|---------|------------------|--|
| 7 to 5 | Max Payload Size | B'000 | R/W | R/W | <p>Max Payload Size</p> <p>Indicates the maximum payload size for the device.</p> <p>B'000: 128 bytes max. B'001: 256 bytes max. B'010: 512 bytes max. B'011: 1024 bytes max. B'100: 2048 bytes max. B'101: 4096 bytes max. B'110: Reserved B'111: Reserved</p> |
| 4 | ERLOD | B'1 | R/W | R/W | <p>Enabled Relax Ordering</p> <p>Enables the device to issue the relaxed ordering transaction.</p> <p>0: The relaxed ordering transaction cannot be issued. 1: The relaxed ordering transaction can be issued.</p> |
| 3 | URRPE | B'0 | R/W | R/W | <p>Unsupported Request Reporting Enable</p> <p>Enables the error message report on detection of an unsupported request.</p> <p>0: The FATAL_ERR or NONFATAL_ERR message is not transmitted on detection of an unsupported request. 1: The FATAL_ERR or NONFATAL_ERR message is transmitted on detection of an unsupported request.</p> |
| 2 | FERPE | B'0 | R/W | R/W | <p>Fatal Error Reporting Enable</p> <p>Enables the error message report on detection of a FATAL_ERR.</p> <p>0: The ERR_FATAL message is not transmitted on detection of a FATAL_ERR. 1: The ERR_FATAL message is transmitted on detection of a FATAL_ERR.</p> |
| 1 | NFERPE | B'0 | R/W | R/W | <p>Non-fatal Error Reporting Enable</p> <p>Enables the error message report on detection of a NON_FATAL_ERR.</p> <p>0: The ERR_NONFATAL message is not transmitted on detection of a NON_FATAL_ERR. 1: The ERR_NONFATAL message is transmitted on detection of a NON_FATAL_ERR.</p> |
| 0 | CERPE | B'0 | R/W | R/W | <p>Correctable Error Reporting Enable</p> <p>Enables the error message report on detection of a correctable error.</p> <p>0: The ERR_COR message is not transmitted on detection of a correctable error. 1: The ERR_COR message is transmitted on detection of a correctable error.</p> |

49.2.69 PCI Express Capability Register 3 (EXPCAP3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

- Link capability register

| | | | | | | | | | | | | | | | | |
|------------------|-------------|------------------|-----|-----|----------------|-----|--------------------|-----|-----|-----|-----------------|-----------------------|---------------|-------|--------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Port Number | | | | | | | | — | — | LKBWN OTFCAP | DLLACT RPCAP | SDNERP CAP | CLKPM | L1ELAT | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | L1ELAT | L0s Exit Latency | | | ASPM Supported | | Maximum Link Width | | | | | Supported Link Speeds | | | | |
| Initial value: | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-------------|---------------|---------|------------------|---|
| 31 to 24 | Port Number | All 0 | R | R/W | Port Number Indicates the port number of this PCI Express port. Specify the port number at initialization. |
| 23, 22 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 | LKBWNOTFCAP | B'0 | R | R/W | Link Bandwidth Notification Capability Indicates whether or not the link bandwidth change report function is supported by EXPCAP4[30].LKBWMNGSTS or EXPCAP4[31].LKAUTOBWSTS. 0: The link bandwidth change report function is not supported. 1: The link bandwidth change report function is supported. <ul style="list-style-type: none"> Root Port: To support this function, set 1 at initialization. Endpoint: Use the initial value. Not need to be changed. |
| 20 | DLLACTRPCAP | B'0 | R | R/W | Data Link Layer Active Reporting Capable Indicates whether or not the Data Link Layer Active State can be indicated by EXPCAP4[29].DLLACT. 0: The Data Link Layer Active State cannot be indicated. 1: The Data Link Layer Active State can be indicated. <ul style="list-style-type: none"> Root Port: To support this function, set 1 at initialization. Endpoint: Use the initial value. Not need to be changed. |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-----------|---------------|---------|------------------|---|
| 19 | SDNERPCAP | B'0 | R | R | <p>Surprise Down Error Reporting Capable</p> <p>Indicates that the Surprise Down error detection function is supported.</p> <p>0: The Surprise Down error detection function is not supported.</p> <p>1: The Surprise Down error detection function is supported.</p> <p>This device does not support the Surprise Down error detection function.</p> <p>Fixed to 0.</p> |
| 18 | CLKPM | B'0 | R | R/W | <p>Clock Power Management</p> <p>Indicates that the clock can be stopped when the link is in the L1, L2Ready, or L3Ready state.</p> <p>0: Clock cannot be stopped.</p> <p>1: Clock can be stopped.</p> <p>Set the device function at initialization.</p> <p>Note: This field can be set with the device that conforms to the form factor supporting Clock Request Capability.</p> |
| 17 to 15 | L1ELAT | B'111 | R | R/W | <p>L1 Exit Latency</p> <p>Indicates the L1 to L0 transition latency of the device.</p> <p>B'000: L0 recovery is completed in less than 1 μs.</p> <p>B'001: L0 recovery is completed in 1 μs or more, but less than 2 μs.</p> <p>B'010: L0 recovery is completed in 2 μs or more, but less than 4 μs.</p> <p>B'011: L0 recovery is completed in 4 μs or more, but less than 8 μs.</p> <p>B'100: L0 recovery is completed in 8 μs or more, but less than 16 μs.</p> <p>B'101: L0 recovery is completed in 16 μs or more, but less than 32 μs.</p> <p>B'110: L0 recovery is completed in 32 μs or more, but 64 μs or less.</p> <p>B'111: L0 recovery is completed in more than 64 μs.</p> <p>Set the device condition at initialization.</p> |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-----------------------|---------------|---------|------------------|---|
| 14 to 12 | L0s Exit Latency | B'111 | R | R/W | <p>L0s Exit Latency</p> <p>Indicates the L0s to L0 transition latency of the device.</p> <p>B'000: L0 recovery is completed in less than 64 ns.</p> <p>B'001: L0 recovery is completed in 64 ns or more, but less than 128 ns.</p> <p>B'010: L0 recovery is completed in 128 ns or more, but less than 256 ns.</p> <p>B'011: L0 recovery is completed in 256 ns or more, but less than 512 ns.</p> <p>B'100: L0 recovery is completed in 512 ns or more, but less than 1 μs.</p> <p>B'101: L0 recovery is completed in 1 μs or more, but less than 2 μs.</p> <p>B'110: L0 recovery is completed in 2 μs or more, but 4 μs or less.</p> <p>B'111: L0 recovery is completed in more than 4 μs.</p> <p>Set the device condition at initialization.</p> |
| 11, 10 | ASPM Supported | B'01 | R | R | <p>ASPM Supported</p> <p>Indicates the ASPM support status.</p> <p>B'00: Reserved</p> <p>B'01: Supports L0s transition.</p> <p>B'10: Reserved</p> <p>B'11: Supports L0s and L1 transition (not supported with this IP).</p> <p>This IP only supports L0s transition.</p> |
| 9 to 4 | Maximum Link Width | H'01 | R | R/W | <p>Maximum Link Width</p> <p>Indicates the maximum link width.</p> <p>H'00: Reserved</p> <p>H'01: x1 link</p> <p>H'02: x2 link (not supported)</p> <p>H'04: x4 link (not supported)</p> <p>H'08: x8 link (not supported)</p> <p>H'0C: x12 link (not supported)</p> <p>H'10: x16 link (not supported)</p> <p>H'20: x32 link (not supported)</p> |
| 3 to 0 | Supported Link Speeds | H'2 | R | R/W | <p>Supported Link Speeds</p> <p>Indicates the maximum link speed.</p> <p>H'1: 2.5 GT/s</p> <p>H'2: 5.0 GT/s and 2.5 GT/s link speeds supported</p> <p>Other than above: Reserved</p> |

49.2.70 PCI Express Capability Register 4 (EXPCAP4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

- Link status register

| | | | | | | | | | | | | | | | | |
|------------------|-----------------|----------------|------------|--------------|------|----|-----------------------|----|----|----|----|--------------------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | LKAUTO BWSTS | LKBWM NGSTS | DLLAC T | SLCLK CFG | LKTR | — | Negotiated Link Width | | | | | Current Link Speed | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| PCI R/W | *1 | *1 | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | *2 | *2 | R | R/W | R | R | R | R | R | R | R | R | R | R | R | R |

- Link control register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|-----------------|-----------------|----------------|------------|-------------|-------------|------------|-------|-----|---|--------------|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | LKAUTOB WINT | LKBWMN GINTE | HWAUTO WDIS | ECLKP M | EXTSY NC | CCLKC FG | RTRNL K | LKDIS | RCB | — | ASPM Control | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | *1 | *1 | R | R/W | R/W | R/W | R | R | R/W | R | R/W | R/W |
| Internal bus R/W | R | R | R | R | *3 | *3 | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W |

- Notes:
- R/W changes by the state. Root Port:—, Endpoint:R
 - R/W changes by the state. Root Port:RW1C, Endpoint:R
 - R/W changes by the state. Root Port:R/W, Endpoint:R

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|-----|-------------|---------------|---------|------------------|--|
| 31 | LKAUTOBWSTS | B'0 | — | RW1C | Link Autonomous Bandwidth Status <ul style="list-style-type: none"> Root Port: <p>When a change of link bandwidth due to intentional factor is detected, 1 is set. This bit is cleared to 0 when 1 is written.</p> <p>Note: This field is only valid when EXCAP3[21].LKBWNOTFCAP is set to 1. If set to 0, this bit is fixed to 0.</p> |
| | | | R | R | Link Autonomous Bandwidth Status <ul style="list-style-type: none"> Endpoint: <p>Use the initial value. Not need to be changed.</p> |
| 30 | LKBWMNGSTS | B'0 | — | RW1C | Link Bandwidth Management Status <ul style="list-style-type: none"> Root Port: <p>When a change of link bandwidth due to reliability-related factor is detected, 1 is set. This bit is cleared to 0 when 1 is written.</p> <p>Note: This field is only valid when EXCAP3[21].LKBWNOTFCAP is set to 1. If set to 0, this bit is fixed to 0.</p> |
| | | | R | R | Link Bandwidth Management Status <ul style="list-style-type: none"> Endpoint: <p>Use the initial value. Not need to be changed.</p> |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-----------------------|---------------|---------|------------------|--|
| 29 | DLLACT | B'0 | R | R | Data Link Layer Active Indicates that DLCMSM is in the DL_ACTIVE state. 0: Not in the DL_ACTIVE state 1: In the DL_ACTIVE state Note: This bit is only valid when EXCAP3[20].DLLACTRPCAP is set to 1. If set to 0, this bit is fixed to 0. |
| 28 | SLCLKCFG | B'0 | R | R/W | Slot Clock Configuration Indicates that the same reference clock as that supplied to the connector by the platform is used. 0: The same reference clock is not used. 1: The same reference clock is used. |
| 27 | LKTR | B'0 | R | R | Link Training Indicates that link training by MAC LTSSM is in progress. <ul style="list-style-type: none"> Root Port: Set to 1 when link training is in progress. Endpoint: Fixed to 0. |
| 26 | — | B'0 | R | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 25 to 20 | Negotiated Link Width | H'04 | R | R | Negotiated Link Width Indicates the link width determined by negotiation between links. H'01: x1 link width H'02: x2 link width H'04: x4 link width H'08: x8 link width H'0C: x12 link width H'10: x16 link width H'20: x32 link width This field value is undefined while a link is not established. |
| 19 to 16 | Current Link Speed | H'1 | R | R | Current Link Speed Indicates the link speed determined by negotiation between links. H'1: 2.5 GT/s H'2: 5.0 GT/s |
| 15 to 12 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|-----|--------------|---------------|-----------------|------------------|---|
| 11 | LKAUTOBWINTE | B'0 | Root Port: — | Root Port: R/W | <p>Link Autonomous Bandwidth Interrupt Enable</p> <p>Enables generation of the interrupt by EXPCAP4[31].LKAUTOBWSTS.</p> <ul style="list-style-type: none"> Root Port: An interrupt is generated when LKAUTOBWSTS is 1. <p>Note: This bit is only valid when EXCAP3[21].LKBMNOTFCAP is set to 1. If set to 0, this bit is fixed to 0.</p> <hr/> <ul style="list-style-type: none"> Endpoint: Use the initial value. Not need to be changed. |
| 10 | LKBWMNGINTE | B'0 | Root Port: — | Root Port: R/W | <p>Link Bandwidth Management Interrupt Enable</p> <p>Enables generation of the interrupt by EXPCAP4[30].LKBWMNGSTS.</p> <ul style="list-style-type: none"> Root Port: Writing 1 to this bit sets it to 1, and enables generation of the interrupt when LKBWMNGSTS is 1. <p>Note: This bit is only valid when EXCAP3[21].LKBMNOTFCAP is set to 1. If set to 0, this bit is fixed to 0.</p> <hr/> <ul style="list-style-type: none"> Endpoint: Use the initial value. Not need to be changed. |
| 9 | HWAUTOWDIS | B'0 | R | R | <p>Hardware Autonomous Width Disable</p> <p>Disables autonomous change of link width by hardware except for the cases of poor link reliability.</p> <p>0: Autonomous change of link width by hardware is enabled.</p> <p>1: Autonomous change of link width by hardware is disabled.</p> <p>According to the standard, fixing this bit to 0 is allowed if hardware does not provide the autonomous link width changing mechanism.</p> <p>Since this IP does not provide the autonomous link width changing mechanism, this bit is fixed to 0.</p> |
| 8 | ECLKPM | B'0 | R/W | R/W | <p>Enable Clock Power Management</p> <p>Enables the clock power management function.</p> <p>Note: This bit is only valid when EXCAP3[18].CLKPM is set to 1.</p> <p>If set to 0, this bit is fixed to 0.</p> |
| 7 | EXTSYNC | B'0 | R/W | R/W | <p>Extended SYNC</p> <p>Transmits the ordered set when the MAC LTSSM makes a transition from L0s or in the recovery state.</p> |
| 6 | CCLKCFG | B'0 | R/W | R/W | <p>Common Clock Configuration</p> <p>Indicates that the same clock is used by both ends of the link.</p> |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|------|--------------|---------------|---------|-------------------------------------|--|
| 5 | RTRNLK | B'0 | R | R/W | Retrain Link Directs link retraining. Note: The read value is always 0. <ul style="list-style-type: none"> • Root Port: Write 1 to allow link retraining. • Endpoint: This bit is reserved. The write value should always be 0. |
| 4 | LKDIS | B'0 | R | R/W | Link Disable Disables the link. <ul style="list-style-type: none"> • Root Port: Write 1 to disable the link. • Endpoint: This bit is reserved. The write value should always be 0. |
| 3 | RCB | B'0 | R/W | Root Port: R/W Endpoint: R/W | Read Completion Boundary Indicates the read completion boundary. 0: 64-byte boundary 1: 128-byte boundary This field does not effect on the hardware. This IP divides a read completion only at 128-byte boundaries. |
| 2 | Reserved | B'0 | R | R | This bit is always read as 0. The write value should always be 0. |
| 1, 0 | ASPM Control | B'00 | R/W | R/W | ASPM Control Sets the ASPM support level. B'00: Disabled B'01: L0s transition enabled B'10: L1 transition enabled (setting prohibited) B'11: L0s/L1 transition enabled (setting prohibited) |

49.2.71 PCI Express Capability Register 5 (EXPCAP5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------|---------------|---------|------------------|--|
| 31 to 0 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |

Header TYPE01

- Slot capabilities register

| | | | | | | | | | | | | | | | | | |
|------------------|----------------------|------------------------|----|----|----|----|----|----|----|----|---------------|---------------|--------------|--------------|---------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | Physical Slot Number | | | | | | | | | | | | | NOCCS P | EMLKP RS | SLPLS C | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | SLPLS C | Slot Power Limit Value | | | | | | | | | HOTPL GCAP | HOTPL GSPR | PWIND PRS | ATIND PRS | MRLSE NPRS | PWCO NPRS | ATBTN PRS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|----------|----------------------|---------------|------------------|---|
| 31 to 19 | Physical Slot Number | All 0 | R | Physical Slot Number According to the PCI standard, indicates the physical slot number connected to the device. This field is invalid with this IP. |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|---------|------------------------|---------------|------------------|--|
| 18 | NOCCSP | B'0 | R | <p>No Command Complete Support</p> <p>According to the PCI standard, indicates that completion of a Hot Plug command is not to be reported.</p> <p>0: Upon completion of a Hot Plug command, it is reported.</p> <p>1: Upon completion of a Hot Plug command, it is not reported.</p> <p>This field is invalid with this IP.</p> |
| 17 | EMLKPRS | B'0 | R | <p>Electro Mechanical Interlock Present</p> <p>According to the PCI standard, indicates that Electromechanical Interlock is provided for the slot.</p> <p>This field is invalid with this IP.</p> |
| 16, 15 | SLPLSC | B'00 | R | <p>Slot Power Limit Scale</p> <p>According to the PCI standard, indicates the scale for slot power limit value.</p> <p>B'00: 1.0x</p> <p>B'01: 0.1x</p> <p>B'10: 0.01x</p> <p>B'11: 0.001x</p> <p>This field is invalid with this IP.</p> |
| 14 to 7 | Slot Power Limit Value | All 0 | R | <p>Slot Power Limit Value</p> <p>According to the PCI standard, indicates the upper limit of slot power supplied by the slot power supply in Watts.</p> <p>This field is invalid with this IP.</p> |
| 6 | HOTPLGCAP | B'0 | R | <p>Hot Plug Capable</p> <p>According to the PCI standard, indicates the Hot Plug support status in this slot.</p> <p>This field is invalid with this IP.</p> |
| 5 | HOTPLGSPR | B'0 | R | <p>Hot Plug Surprise</p> <p>According to the PCI standard, indicates that the adapter inserted to this slot can be unplugged without advance notice.</p> <p>This field is invalid with this IP.</p> |
| 4 | PWINDPRS | B'0 | R | <p>Power Indicator Present</p> <p>According to the PCI standard, indicates that the power indicator is provided.</p> <p>This field is invalid with this IP.</p> |
| 3 | ATINDPRS | B'0 | R | <p>Attention Indicator Present</p> <p>According to the PCI standard, indicates that the attention indicator is provided.</p> <p>This field is invalid with this IP.</p> |
| 2 | MRLSENPRS | B'0 | R | <p>MRL Sensor Present</p> <p>According to the PCI standard, indicates that the MRL sensor is provided.</p> <p>This field is invalid with this IP.</p> |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|------------|-----------------|----------------------|-------------------------|--|
| 1 | PWCONPRS | B'0 | R | Power Controller Present According to the PCI standard, indicates that the power controller is provided. This field is invalid with this IP. |
| 0 | ATBTNPRS | B'0 | R | Attention Button Present According to the PCI standard, indicates that the attention button is provided. This field is invalid with this IP. |

49.2.72 PCI Express Capability Register 6 (EXPCAP6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------|---------------|---------|------------------|--|
| 31 to 0 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |

Header TYPE01

- Slot status register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|-----------|----------|----------|-----------|---------|-----------|------------|-----------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | DLLST CHG | EMLKS TS | PRDET ST | MRLSE NST | COMC PL | PRDET CHG | MRLSE NCHG | PWFAL DET | ATBTN PRES |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

- Slot control register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|------------|----------|-----------|----------|---|----------|-------------|-------------|------------|-------------|------------|------------|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | DLLSTC HGE | EMLKC ON | PWCON CON | PWINDCON | — | ATINDCON | HOTPLG INTE | COMCP LINTE | PRDETC HGE | MRLSEN CHGE | PWFALD ETE | ATBTN RESE | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|----------|----------|---------------|------------------|--|
| 31 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | DLLSTCHG | B'0 | R | Data Link Layer ST Changed According to the PCI standard, indicates that the EXPCAP4[29].DLLACT has changed. This field is invalid with this IP. |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|----------|-----------|---------------|------------------|---|
| 23 | EMLKSTS | B'0 | R | Electro Mechanical Interlock Status According to the PCI standard, indicates the Electromechanical Interlock state. 0: Not locked by Electromechanical Interlock. 1: Locked by Electromechanical Interlock This field is invalid with this IP. |
| 22 | PRDETST | B'1 | R | Presence Detect State According to the PCI standard, indicates that the adapter is inserted in the slot. 0: The slot is empty. 1: The adapter is inserted in the slot. This field is invalid with this IP. |
| 21 | MRLSENST | B'0 | R | MRL Sensor State According to the PCI standard, indicates the MRL sensor state. 0: MRL is open. 1: MRL is closed. This field is invalid with this IP. |
| 20 | COMCPL | B'0 | R | Command Completed According to the PCI standard, indicates that the Hot Plug command has been completed when EXPCAP5[18].NOCCSP = 0. 0: The Hot Plug command is being executed. 1: The Hot Plug command execution has been completed. This field is invalid with this IP. |
| 19 | PRDETCHG | B'0 | R | Presence Detect Changed According to the PCI standard, indicates that the EXPCAP6[22].PRDETST has changed. This field is invalid with this IP. |
| 18 | MRLSENCHG | B'0 | R | MRL Sensor Changed According to the PCI standard, indicates that the EXPCAP6[21].MRLSENST has changed. This field is invalid with this IP. |
| 17 | PWFALDET | B'0 | R | Power Fault Detect According to the PCI standard, indicates that the power fault has been detected. This field is invalid with this IP. |
| 16 | ATBTNPRES | B'0 | R | Attention Button Pressed According to the PCI standard, indicates that the attention button has been pressed. This field is invalid with this IP. |
| 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | DLLSTCHGE | B'0 | R | Data Link Layer State Change Enable According to the PCI standard, enables the setting of PCIEXP6[24].DLLSTCHG. This field is invalid with this IP. |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|------|------------|---------------|------------------|--|
| 11 | EMLKCON | B'0 | R | Electro Mechanical Interlock Control According to the PCI standard, toggles the state of Electromechanical Interlock. This field is invalid with this IP. |
| 10 | PWCONCON | B'0 | R | Power Controller Control According to the PCI standard, controls the power controller. 0: Power Off 1: Power On This field is invalid with this IP. |
| 9, 8 | PWINDCON | B'00 | R | Power Indicator Control According to the PCI standard, controls the power indicator. B'00: Reserved B'01: On B'10: Blink B'11: Off This field is invalid with this IP. |
| 7, 6 | ATINDCON | B'00 | R | Attention Indicator Control According to the PCI standard, controls the attention indicator. B'00: Reserved B'01: On B'10: Blink B'11: Off This field is invalid with this IP. |
| 5 | HOTPLGINTE | B'0 | R | Hot Plug Interrupt Enable According to the PCI standard, enables MSI generation upon Hot Plug. |
| 4 | COMCPLINTE | B'0 | R | Command Completed Interrupt Enable According to the PCI standard, enables MSI generation upon Command Complete. This field is invalid with this IP. |
| 3 | PRDETCHGE | B'0 | R | Presence Detect Changed Enable According to the PCI standard, enables MSI generation by PRDETCHG. This field is invalid with this IP. |
| 2 | MRLSENGHGE | B'0 | R | MRL Sensor Changed Enable According to the PCI standard, enables MSI generation by MRLSENGHG. This field is invalid with this IP. |
| 1 | PWFALDETE | B'0 | R | Power Fault Detect Enable According to the PCI standard, enables MSI generation by PWFALDET. This field is invalid with this IP. |
| 0 | ATBTNPRES | B'0 | R | Attention Button Pressed Enable According to the PCI standard, enables MSI generation by ATBTNPRES. This field is invalid with this IP. |

49.2.73 PCI Express Capability Register 7 (EXPCAP7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------|---------------|---------|------------------|--|
| 31 to 0 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |

Header TYPE01

- Root capabilities register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CRSVIS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

- Root control register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|---|---|---|---|---|---------|---------|---------|----------|---------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | CRSVISE | PMEINTE | SERRFEE | SERRNFEE | SERRCEE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|----------|----------|---------------|------------------|--|
| 31 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|---------|----------|---------------|------------------|---|
| 16 | CRSVIS | B'0 | R/W | <p>CRS Software Visibility</p> <p>Indicates whether Root Port has the function of returning Configuration Retry Status to the upper software.</p> <p>Set the device function at initialization.</p> |
| 15 to 5 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 4 | CRSVISE | B'0 | R/W | <p>CRS Software Visibility Enable</p> <p>Enables Root Port to return Configuration Retry Status to the upper software.</p> <p>Before setting this bit to 1, the CRSVIS bit should be set. If these bits are set at the same time, CRSVISE is not set to 1. Setting this bit to 1 changes Configuration access behavior.</p> <p>For details, see section 49.3.2, Configuration Cycle.</p> |
| 3 | PMEINTE | B'0 | R/W | <p>PME Interrupt Enable</p> <p>Enables Root Port to generate the interrupt to the upper software upon reception of a PME message.</p> <p>0: If a PM_PME has been received when another PM_PME is received, EXPCAP8[15:0].PME_RequesterID is overwritten.</p> <p>1: If a PM_PME has been received when another PM_PME is received, EXPCAP8[15:0].PM_RequesterID is not overwritten. PM_RequesterID of the PM_PME received later is placed in the pending state.</p> <p>At this time, all the receive operations are stopped. Thus, always clear EXPCAP8[16].PMEST.</p> |
| 2 | SERRFEE | B'0 | R/W | <p>System Error on Fatal Error Enable</p> <p>Enables Root Port to generate the interrupt to the upper software upon reception of an ERR_FATAL message or occurrence of an error reported by Root Port itself.</p> |
| 1 | SERRNFEE | B'0 | R/W | <p>System Error on Non-fatal Error Enable</p> <p>Enables Root Port to generate the interrupt to the upper software upon reception of an ERR_NONFATAL message or occurrence of an error reported by Root Port itself.</p> |
| 0 | SERRCEE | B'0 | R/W | <p>System Error on Correctable Error Enable</p> <p>Enables Root Port to generate the interrupt to the upper software upon reception of an ERR_COR message or occurrence of an error reported by Root Port itself.</p> |

49.2.74 PCI Express Capability Register 8 (EXPCAP8)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Header TYPE00

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------|---------------|---------|------------------|--|
| 31 to 0 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |

Header TYPE01

- Root status register

| | | | | | | | | | | | | | | | | |
|------------------|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | PMEPD | PMEST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | RW1C |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PME Requester ID | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|----------|----------|---------------|------------------|---|
| 31 to 18 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 17 | PMEPD | B'0 | R | PME Pending Indicates that another PME is pending when this bit is 1. If the PME status is cleared by the upper software with $PMEPD = 1$ and $PMEST = 1$, $PMEST$ is set to 1 again and the pending requester ID is set in the PME Requester ID bits. |

| Bit | Bit Name | Initial Value | Internal Bus R/W | Description |
|---------|------------------|---------------|------------------|--|
| 16 | PMEST | B'0 | RW1C | PME Status Indicates that PME has been received from the requester indicated by the PME Requester ID bits. This bit is set to 1 when the PME is received and cleared to 0 when the upper software writes 1 to this bit. |
| 15 to 0 | PME Requester ID | All 0 | R | PME Requester ID Indicates the requester ID having issued the PME. This field sets the requester ID on reception of PME. |

49.2.75 PCI Express Capability Register 9 (EXPCAP9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

- Device capabilities 2 register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|------------|--------------|------------------------------------|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | ARIFWD SUP | CPLTOD ISSUP | Completion Timeout Range Supported | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|-------------|---------------|---------|------------------|--|
| 31 to 6 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 | ARIFWDSUP | B'0 | R | R | ARI Forwarding Support Indicates the ARI Forwarding support status. <ul style="list-style-type: none"> • Root Port: This IP does not support the ARI Forwarding. Fixed to 0. • Endpoint: Fixed to 0. |
| 4 | CPLTODISSUP | B'1 | R | R/W | Completion Timeout Disable Supported Indicates the Completion Timeout Disable function support status. 0: The Completion Timeout Disable function is not supported. 1: The Completion Timeout Disable function is supported. This IP supports the Completion Timeout Disable function. |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|--------|------------------------------------|---------------|---------|------------------|--|
| 3 to 0 | Completion Timeout Range Supported | H'0 | R | R | <p>Completion Timeout Range Supported</p> <p>Indicates the Completion Timeout Range support status.</p> <p>Timeout range specified by the standard:</p> <p>Range A: 50 μs to 10 ms</p> <p>Range B: 10 ms to 250 ms</p> <p>Range C: 250 ms to 4 s</p> <p>Range D: 4 s to 64 s</p> <p>Supported ranges:</p> <p>H'0: Setting the timeout value is prohibited (fixed).</p> <p>H'1: Range A</p> <p>H'2: Range B</p> <p>H'3: Range A and B</p> <p>H'6: Range B and C</p> <p>H'7: Range A, B and C</p> <p>H'E: Range B, C and D</p> <p>H'F: Range A, B, C and D</p> <p>Other than above: Reserved</p> <p>This IP does not support the timeout value setting.</p> |

49.2.76 PCI Express Capability Register 10 (EXPCAP10)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

- Device status 2 register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

- Device control 2 register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|---|---|---|---|-------------|--------------|--------------------------|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | ARIFW DE | CPLTO DIS | Completion Timeout Value | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------|---------------|---------|------------------|---|
| 31 to 6 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 | ARIFWDE | B'0 | R | R | ARI Forwarding Enable Enables the ARI Forwarding. <ul style="list-style-type: none"> • Root Port: This IP does not support the ARI Forwarding. Fixed to 0. • Endpoint: Fixed to 0. |
| 4 | CPLTODIS | B'0 | R/W | R/W | Completion Timeout Disable Sets the Completion Timeout Disable function. 0: The Completion Timeout Disable function is disabled. 1: The Completion Timeout Disable function is enabled. This bit is fixed to 0 when EXPCAP9[4].Completion Timeout Disable Supported is 0. |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|--------|--------------------------|---------------|---------|------------------|---|
| 3 to 0 | Completion Timeout Value | H'0 | R | R | <p>Completion Timeout Value</p> <p>Sets the Completion Timeout value.</p> <p>H'0: Default Range 50 μs to 50 ms (fixed)</p> <p>Value of TLCTLR.Default Completion Timeout Time</p> <p>H'1: 50 μs to 100 μs (not supported)</p> <p>H'2: 1 ms to 10 ms (not supported)</p> <p>H'5: 16 ms to 55 ms (not supported)</p> <p>H'6: 65 ms to 210 ms (not supported)</p> <p>H'9: 260 ms to 900 ms (not supported)</p> <p>H'A: 1 s to 3.5 s (not supported)</p> <p>H'C: 4 s to 13 s (not supported)</p> <p>H'D: 17 s to 64 s (not supported)</p> <p>Other than above: Reserved</p> <p>This IP does not support the timeout value setting.</p> |

49.2.77 PCI Express Capability Register 11 (EXPCAP11)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

- Link capabilities 2 register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------|---------------|---------|------------------|--|
| 31 to 0 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.78 PCI Express Capability Register 12 (EXPCAP12)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

- Link status 2 register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CURDEEMLEV |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

- Link control 2 register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|---------|--------|----------|-----------------|-----|-----|---------|-----------|--------|-------------------|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | CMPDEEM | CMPSOS | ETMDFCMP | Transmit Margin | | | SELDEEM | HWATSPDIS | ENTCMP | Target Link Speed | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| PCI R/W | R | R | R | RWS | RWS | RWS | RWS | RWS | RWS | R | R | RWS | RWS | RWS | RWS | RWS |
| Internal bus R/W | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|------------|---------------|---------|------------------|---|
| 31 to 17 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | CURDEEMLEV | B'1 | R | R | Current De-emphasis Level Indicates the current de-emphasis level during 5-GT/s link operation. 0: -6 dB 1: -3.5 dB The value is undefined during 2.5-GT/s operation. |
| 15 to 13 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | CMPDEEM | B'0 | RWS | R/W | Compliance De-emphasis Specifies the de-emphasis level when setting EXPCAP12[4].ENTCMP bit to 1 causes LTSSM to be placed in Polling.Compliance mode. 0: -6 dB 1: -3.5 dB The value is undefined during 2.5-GT/s operation. |
| 11 | CMPSOS | B'0 | RWS | R/W | Compliance SOS When this bit is 1, LTSSM periodically inserts the SKP ordered set between compliance patterns. 0: The SKP ordered set is not periodically inserted. 1: The SKP ordered set is periodically inserted. |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|--------|-----------------|---------------|---------|------------------|--|
| 10 | ETMDFCMP | B'0 | RWS | R/W | <p>Enter Modified Compliance</p> <p>When this bit is 1, LTSSM transmits modified compliance patterns on transition to the Polling.Compliance state.</p> <p>0: Normal compliance patterns are used.</p> <p>1: Modified compliance patterns are used.</p> |
| 9 to 7 | Transmit Margin | B'000 | RWS | R/W | <p>Transmit Margin</p> <p>Sets the signal level without de-emphasis.</p> <p>B'000: Normal signal level</p> <p>B'001: 800 to 1200 mV for full swing, 400 to 700 mV for half swing</p> <p>B'010-(n-1): n is specified by PHY ($3 \leq n \leq 7$).</p> <p>n-111: Reserved</p> <p>Determine n according to the PHY specification.</p> <p>Do not set the value exceeding the PHY specification.</p> |
| 6 | SELDEEM | B'0 | R | R/W | <p>Selectable De-emphasis</p> <p>Indicates the de-emphasis level during 5-GT/s upstream component operation.</p> <ul style="list-style-type: none"> • Root Port: Set a value appropriate for the PHY at initialization. 0: -6 dB 1: -3.5 dB • Endpoint: Fixed to 0. |
| 5 | HWATSPDIS | B'0 | R | R | <p>Hardware Autonomous Speed Disable</p> <p>Disables autonomous change of link width by hardware except for the cases of poor reliability.</p> <p>0: Autonomous change of link width by hardware is enabled.</p> <p>1: Autonomous change of link width by hardware is disabled.</p> <p>According to the standard, fixing this bit to 0 is allowed if hardware does not provide the autonomous link width changing mechanism.</p> <p>Since this IP does not provide the autonomous link width changing mechanism, this bit is fixed to 0. The write value should always be 0.</p> |
| 4 | ENTCMP | B'0 | RWS | R/W | <p>Enter Compliance</p> <p>Setting this bit of both ends of the link forcibly places a link into compliance mode at the specified speed.</p> <p>Used to set compliance mode.</p> |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|--------|-------------------|---------------|---------|------------------|--|
| 3 to 0 | Target Link Speed | B'0010 | RWS | R/W | <p>Target Link Speed</p> <ul style="list-style-type: none">• Root Port: Specifies the upper limit speed that is conveyed to the communication peer during link training. Also, specifies the speed in compliance mode. B'0001: 2.5 GT/s B'0010: 5.0 GT/s• Endpoint: Specifies the speed in compliance mode. B'0001: 2.5 GT/s B'0010: 5.0 GT/s |

49.2.79 PCI Express Capability Register 13 (EXPCAP13)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

- Slot capabilities 2 register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|----------|---------------|---------|------------------|--|
| 31 to 0 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.80 PCI Express Capability Register 14 (EXPCAP14)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Common to Header TYPE00/01

- Slot status 2 register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

- Slot control 2 register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|----------|---------------|---------|------------------|--|
| 31 to 16 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.81 VC Capability Register 0 (VCCAP0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

- Virtual channel enhanced capability header

| | | | | | | | | | | | | | | | | |
|------------------|------------------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|--------------------|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Next Capability Offset | | | | | | | | | | | | Capability Version | | | |
| Initial value: | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PCI Express Extended Capability ID | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|------------------------------------|---------------|---------|------------------|--|
| 31 to 20 | Next Capability Offset | H'1B0 | R | R/W | Next Capability Offset Pointer to the capability list Points to the pointer to Device Serial Number Capability, H'1B0. |
| 19 to 16 | Capability Version | H'1 | R | R | Capability Version Indicates the version of this capability. |
| 15 to 0 | PCI Express Extended Capability ID | H'0002 | R | R | PCI Express Extended Capability ID Indicates the ID of this capability. |

49.2.82 VC Capability Register 1 (VCCAP1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

- Port VC capability register 1

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----------------|--------|----|----|----|---------|----|----|----|-----|--------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | PT ARBTBL SIZE | REFCLK | — | — | — | LPEXTVC | — | — | — | — | EXT VC | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|----------------|---------------|---------|------------------|---|
| 31 to 12 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11, 10 | PT ARBTBL SIZE | B'00 | R | R | Port Arbitration Table entry Size Indicates the size of port arbitration table entry. This IP does not support port arbitration. |
| 9, 8 | REFCLK | B'00 | R | R | Reference Clock Indicates the reference clock for Time Base WRR port arbitration. This IP does not support port arbitration. |
| 7 | — | B'0 | R | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 to 4 | LPEXTVC | B'000 | R | R | Low Priority Extended VC Count Indicates the low priority VC count except the default VC0. With this IP, only VC0 is a low priority VC. |
| 3 | — | B'0 | R | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 2 to 0 | EXT VC | B'000 | R | R/W | Extended VC Count Indicates the VC count except the default VC0. The write value should always be 0. |

49.2.83 VC Capability Register 2 (VCCAP2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

- Port VC capability register 2

| | | | | | | | | | | | | | | | | |
|------------------|-----------------------------|----|----|----|----|----|----|----|----|----|----|----|-----------------|----------------|----------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | VC Arbitration Table Offset | | | | | | | | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | VCARB WRR128 | VCARB WRR64 | VCARB WRR32 | VCARB FIX |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-----------------------------|---------------|---------|------------------|---|
| 31 to 24 | VC Arbitration Table Offset | All 0 | R | R | VC Arbitration Table Offset Indicates the position of VC arbitration. This IP does not provide VC arbitration based on the table. |
| 23 to 4 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 | VCARBWRR128 | B'0 | R | R | VC Arbitration WRR 128 Phase Indicates that Weighted Round Robin 128 Phase is applied to VC arbitration of Low Priority Group VC. This IP does not support this function. |
| 2 | VCARB WRR64 | B'0 | R | R | VC Arbitration WRR 64 Phase Indicates that Weighted Round Robin 64 Phase is applied to VC arbitration of Low Priority Group VC. This IP does not support this function. |
| 1 | VCARB WRR32 | B'0 | R | R | VC Arbitration WRR 32 Phase Indicates that Weighted Round Robin 32 Phase is applied to VC arbitration of Low Priority Group VC. This IP does not support this function. |
| 0 | VCARB FIX | B'0 | R | R | VC Arbitration Hardware Fixed Indicates that HardWareFix scheme is applied to VC arbitration of Low Priority Group VC. This IP does not support this function. |

49.2.84 VC Capability Register 3 (VCCAP3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

- Port VC status register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | VCARB TBLST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

- Port VC control register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|---|---|---|---|---|---|-----------------------|---|----------------|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | VC Arbitration Select | | LDVCA RBTBL | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-----------------------|---------------|---------|------------------|--|
| 31 to 17 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | VCARB TBLST | B'0 | R | R | VC Arbitration Table Status Indicates the VC arbitration table status. This IP does not provide VC arbitration based on the table. |
| 15 to 4 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 1 | VC Arbitration Select | B'000 | R | R | VC Arbitration Select Selects the VC arbitration method for Low Priority Group VC. This IP does not provide VC arbitration. |
| 0 | LDVCA RBTBL | B'0 | R | R | Load VC Arbitration Table Loads the VC arbitration table. This IP does not provide VC arbitration. |

49.2.85 VC Capability Register 4 (VCCAP4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register is used to make settings for the VC0.

- VC0 resource capability register

| | | | | | | | | | | | | | | | | |
|------------------|-------------------------------|----|----|----|----|----|----|----|----|-------------------|-----------------|-------------------|-----------------|----------------|----------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Port Arbitration Table Offset | | | | | | | | — | Maximum Time Slot | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RJCTS NPTR | — | — | — | — | — | — | — | — | — | PTARB WRR256 | PTARBTB WRR128 | PTARB WRR128 | PTARB WRR64 | PTARB WRR32 | PTARB FIX |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-------------------------------|---------------|---------|------------------|--|
| 31 to 24 | Port Arbitration Table Offset | All 0 | R | R | Port Arbitration Table Offset Indicates that the offset address to the port arbitration table. VC0 does not have the port arbitration table. |
| 23 | — | B'0 | R | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 22 to 16 | Maximum Time Slot | All 0 | R | R | Maximum Time Slot Indicates the maximum time slot in port arbitration according to Time Based WRR. VC0 does not provide the WRR port arbitration. |
| 15 | RJCTSNPTR | B'0 | R | R/W | Reject Snoop Transactions Indicates that snoop transactions are rejected. <ul style="list-style-type: none"> Root Port: Set the VC condition at initialization. Endpoint: The write value should always be 0. |
| 14 to 6 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 | PTARBWRR256 | B'0 | R | R | Port Arbitration WRR256 Indicates that Weighted Round Robin 256 Phase is applied to port arbitration. This IP does not support port arbitration. |
| 4 | PTARBTBWRR128 | B'0 | R | R | Port Arbitration Time Base WRR128 Indicates that Time Base Weighted Round Robin 128 Phase is applied to port arbitration. This IP does not support port arbitration. |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|-----|-------------|---------------|---------|------------------|--|
| 3 | PTARBWRR128 | B'0 | R | R | Port Arbitration WRR128 Indicates that Weighted Round Robin 128 Phase is applied to port arbitration. This IP does not support port arbitration. |
| 2 | PTARBWRR64 | B'0 | R | R | Port Arbitration WRR64 Indicates that Weighted Round Robin 64 Phase is applied to port arbitration. This IP does not support port arbitration. |
| 1 | PTARBWRR32 | B'0 | R | R | Port Arbitration WRR32 Indicates that Weighted Round Robin 32 Phase is applied to port arbitration. This IP does not support port arbitration. |
| 0 | PTARBFIX | B'0 | R | R | Port Arbitration Hardware Fixed Indicates that HardWareFix scheme is applied to port arbitration. This IP does not support port arbitration. |

49.2.86 VC Capability Register 5 (VCCAP5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register is used to make settings for the VC0.

- VC0 resource control register

| | | | | | | | | | | | | | | | | |
|------------------|-----------|----|----|----|----|------|----|----|-----|-----|-----|-----|-------------------------|-----|-----|----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | VC Enable | — | — | — | — | VCID | | | — | — | — | — | Port Arbitration Select | | | LDPTA RBTBL |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | TC7 | TC6 | TC5 | TC4 | TC3 | TC2 | TC1 | TC0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| PCI R/W | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|-------------------------|---------------|---------|------------------|---|
| 31 | VC Enable | B'1 | R | R | VC Enable Indicates that VC is enabled. 0: VC0 is disabled. 1: VC0 is enabled. VC0 is always enabled. |
| 30 to 27 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 26 to 24 | VCID | All 0 | R | R | VCID Indicates the VC ID. VC0 is always 0. |
| 23 to 20 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 to 17 | Port Arbitration Select | All 0 | R | R | Port Arbitration Select Selects the port arbitration method. This IP does not support port arbitration. |
| 16 | LDPTARBTL | B'0 | R | R | Load Port Arbitration Table Loads the port arbitration table. This IP does not support port arbitration. |
| 15 to 8 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | TC7 | B'1 | R/W | R/W | TC7 Maps TC7 into VC0. All the TCs are mapped into VC0 by default. |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|-----|----------|---------------|---------|------------------|--|
| 6 | TC6 | B'1 | R/W | R/W | TC6 Maps TC6 into VC0. All the TCs are mapped into VC0 by default. |
| 5 | TC5 | B'1 | R/W | R/W | TC5 Maps TC5 into VC0. All the TCs are mapped into VC0 by default. |
| 4 | TC4 | B'1 | R/W | R/W | TC4 Maps TC4 into VC0. All the TCs are mapped into VC0 by default. |
| 3 | TC3 | B'1 | R/W | R/W | TC3 Maps TC3 into VC0. All the TCs are mapped into VC0 by default. |
| 2 | TC2 | B'1 | R/W | R/W | TC2 Maps TC2 into VC0. All the TCs are mapped into VC0 by default. |
| 1 | TC1 | B'1 | R/W | R/W | TC1 Maps TC1 into VC0. All the TCs are mapped into VC0 by default. |
| 0 | TC0 | B'1 | R | R | TC0 Maps TC0 into VC0. TC0 is always mapped into VC0. |

49.2.87 VC Capability Register 6 (VCCAP6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register is used to make settings for the VC0.

- VC0 resource status register

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | VCNGPD | PTARBTBLST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| | | | | | | | | | | | | | | | | |
|------------------|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|------------|---------------|---------|------------------|--|
| 31 to 18 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 17 | VCNGPD | B'1 | R | R | VC Negotiation Pending Indicates that VC0 is in the negotiation processing. 0: The VC0 negotiation processing is not in progress. 1: The VC0 negotiation processing is in progress. |
| 16 | PTARBTBLST | B'0 | R | R | Port Arbitration Table Indicates the port arbitration table status. This IP does not provide port arbitration based on the table. |
| 15 to 0 | — | All 0 | R | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.88 Device Serial Number Capability Register 0 (SERNUMCAP0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

- Device serial number enhanced capability header

| | | | | | | | | | | | | | | | | |
|------------------|------------------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|--------------------|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Next Capability Offset | | | | | | | | | | | | Capability Version | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PCI Express Extended Capability ID | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|----------|------------------------------------|---------------|---------|------------------|---|
| 31 to 20 | Next Capability Offset | H'000 | R | R/W | Next Capability Offset Pointer to the capability list. Indicates H'000, which is End Of List. |
| 19 to 16 | Capability Version | H'1 | R | R | Capability Version Indicates the version of this capability. |
| 15 to 0 | PCI Express Extended Capability ID | H'0003 | R | R | PCI Express Extended Capability ID Indicates the ID of this capability. |

49.2.89 Device Serial Number Capability Register 1 (SERNUMCAP1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|------------------|--------------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Device Serial Number 1st | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Device Serial Number 1st | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|--------------------------|---------------|---------|------------------|---|
| 31 to 0 | Device Serial Number 1st | All 0 | R | R | Device Serial Number 1st Indicates the serial number of the device. This field reflects the value of the DSERSETR0 field. |

49.2.90 Device Serial Number Capability Register 2 (SERNUMCAP2)

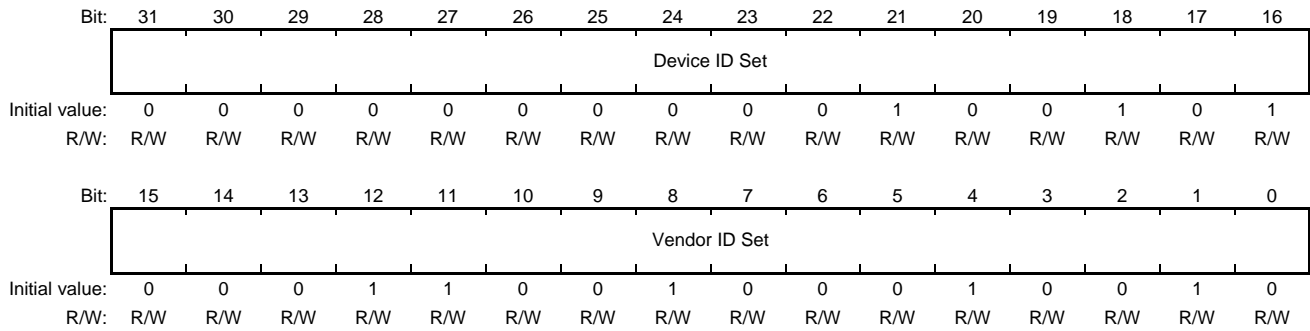
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|------------------|--------------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Device Serial Number 2nd | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Device Serial Number 2nd | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PCI R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Internal bus R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | PCI R/W | Internal Bus R/W | Description |
|---------|--------------------------|---------------|---------|------------------|---|
| 31 to 0 | Device Serial Number 2nd | All 0 | R | R | Device Serial Number 2nd Indicates the serial number of the device. This field reflects the value of the DSERSETR1 field. |

49.2.91 ID Setting Register 0 (IDSETR0)

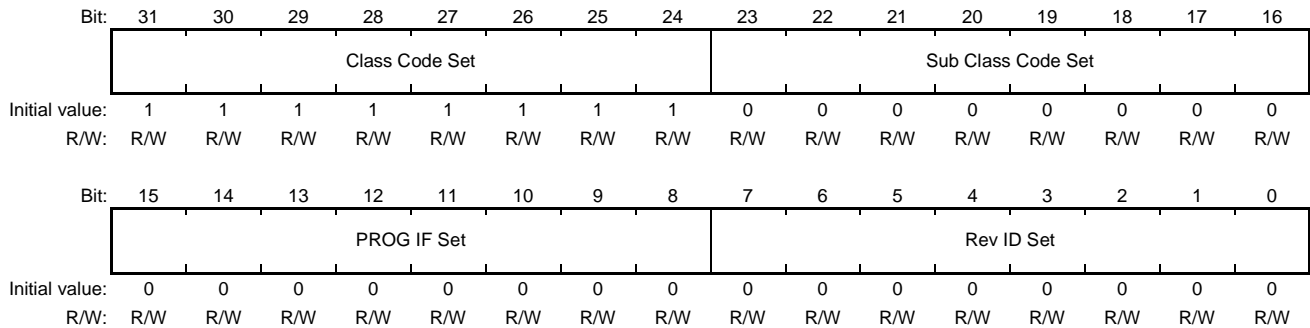
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 to 16 | Device ID Set | H'0025 | R/W | Device ID Set Indicates the device ID. The value of this field is reflected to PCICONF0[31:16].Device ID. |
| 15 to 0 | Vendor ID Set | H'1912 | R/W | Vendor ID Set Indicates the PCI device vendor ID. The value of this field is reflected to PCICONF0[15:0].Vendor ID. |

49.2.92 ID Setting Register 1 (IDSETR1)

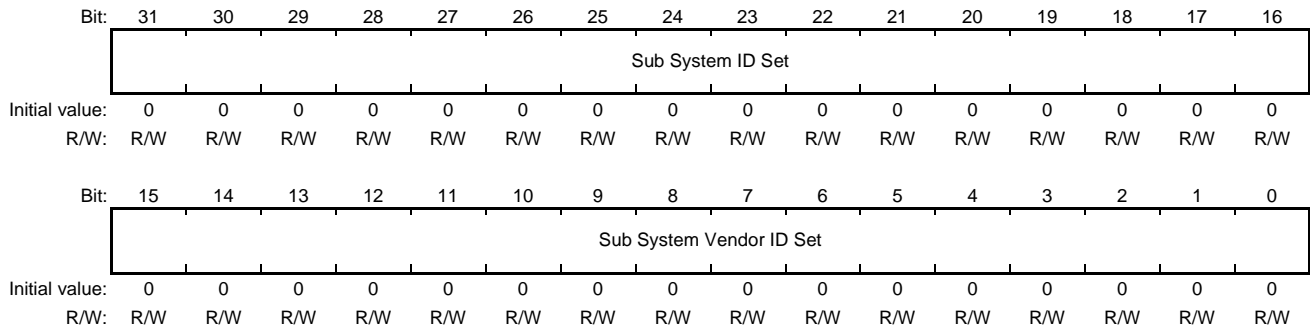
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------|---------------|-----|--|
| 31 to 24 | Class Code Set | H'FF | R/W | Class Code Set Set the class code at initialization. The value of this field is reflected to PCICONF2[31:24].Class Code. |
| 23 to 16 | Sub Class Code Set | H'00 | R/W | Sub Class Code Set Set the sub-class code at initialization. The value of this field is reflected to PCICONF2[23:16].Sub Class Code. |
| 15 to 8 | PROG IF Set | H'00 | R/W | PROG IF Set Set the programming IF code at initialization. The value of this field is reflected to PCICONF2[15:8].PROG IF. |
| 7 to 0 | Rev ID Set | H'00 | R/W | Rev ID Set Set the revision ID at initialization. The value of this field is reflected to PCICONF2[7:0].Revision ID. |

49.2.93 SUBID Setting Register (SUBIDSETR)

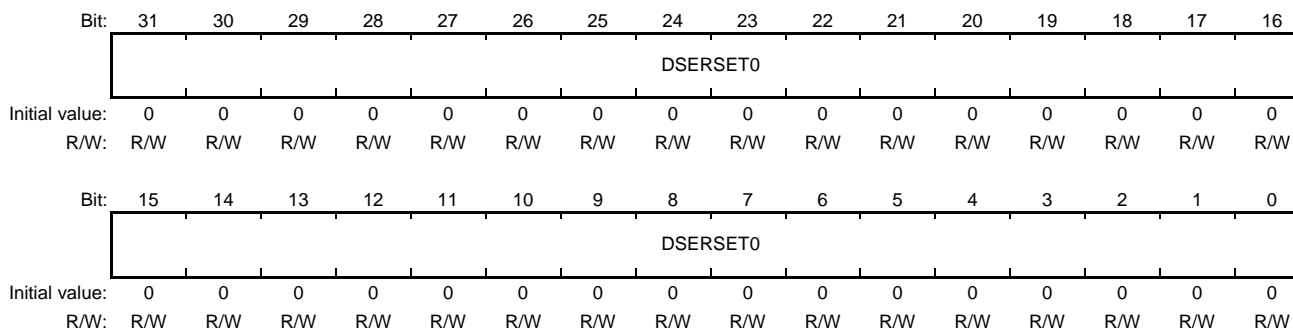
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------------|---------------|-----|--|
| 31 to 16 | Sub System ID Set | H'0000 | R/W | <p>Sub System ID Set</p> <p>Sets the sub-system ID.</p> <p>Set the sub-system device ID allocated to the PCI device vendor at initialization.</p> <p>The value of this field is reflected to PCICONF11[31:16].Sub System ID.</p> |
| 15 to 0 | Sub System Vendor ID Set | H'0000 | R/W | <p>Sub System Vendor ID Set</p> <p>Sets the sub-system vendor ID.</p> <p>Set the sub-system vendor ID allocated to the PCI device vendor at initialization.</p> <p>The value of this field is reflected to PCICONF11[15:0].Sub System Vendor ID.</p> |

49.2.94 Device Serial Number Setting Register 0 (DSERSETR0)

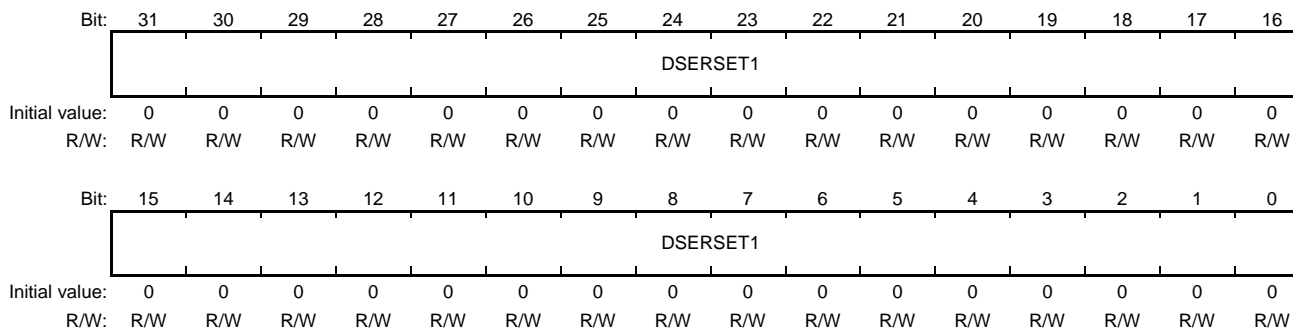
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | DSERSET0 | All 0 | R/W | DSERSET0 Set 4 bytes of 1stDW of the device serial number at initialization. The value of this field is reflected to SERNUMCAP1.Device Serial Number 1st. |

49.2.95 Device Serial Number Setting Register 1 (DSERSETR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | DSERSET1 | All 0 | R/W | DSERSET1 Set 4 bytes of 2ndDW of the device serial number at initialization. The value of this field is reflected to SERNUMCAP2.Device Serial Number 2nd. |

49.2.96 TL Control Register (TLCTLR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----|---------------------------------|-----|-----|-----|-----|-----|---------------|-----|-----|-----|-----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Bus Number | | | | | | | | Device Number | | | | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | Default Completion Timeout Time | | | | | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------------------|---------------|-----|---|
| 31 to 24 | Bus Number | H'00 | R/W | <p>Bus Number</p> <p>The value of this field is used as the bus number at issuance of a request.</p> <p>The value of this field is used for checking the transaction ID bus number part within this IP on reception of Completion.</p> <p>Root Port:</p> <p>Set the device bus number.</p> <p>Endpoint:</p> <p>This field is always updated on reception of a Type0 Configuration Write request.</p> |
| 23 to 19 | Device Number | H'00 | R/W | <p>Device Number</p> <p>The value of this field is used as the device number at issuance of a request.</p> <p>The value of this field is used for checking the transaction ID device number part within this IP on reception of Completion.</p> <p>Root Port:</p> <p>Set the device number.</p> <p>Endpoint:</p> <p>This field is always updated on reception of a Type0 Configuration Write request.</p> |
| 18 to 14 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 13 to 8 | Default Completion Timeout Time | H'00 | R/W | <p>Default Completion Timeout Time</p> <p>Sets the Completion Timer timeout time for default setting (unit: ms). If a Completion cannot be received within the time set by this field, it is handled as a Completion timeout. Set a value from 10 to 50 ms.</p> |
| 7 to 0 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

49.2.97 MAC Status Register (MACSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|------|-----------|-----------|-------|-----------|------------|----|-----------|------------|-------|------------|-------|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | LKTR | LPBAK 1ST | LPBAK 2ST | DISST | HOTRS TST | Link Width | | | | | Link Speed | | | | |
| Initial value: | — | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | SPCHG SUC | SPCHG FAIL | SPCHG | SPCHG FIN | L0ENT | — | — | — |
| Initial value: | — | — | — | 0 | — | — | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — |
| R/W: | R | R | R | R | R | R | R | R | RW1C | RW1C | RW1C | RW1C | RW1C | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 31 | — | — | R | Reserved The read value is undefined. The write value should always be 0. |
| 30 | LKTR | B'0 | R | Link Training Indicates that MAC LTSSM is in the recovery or configuration state and that link training is in progress. 0: Link training is not in progress. 1: Link training is in progress. The value of this field is reflected to EXPCAP4[27].LKTR. |
| 29 | LPBAK1ST | B'0 | R | Loop Back1 State Indicates that the MAC LTSSM is in Per Lane Loop BAK mode. Loopback test can be executed as a loopback master. Note: LTSSM has made a transition to the loopback state without executing Configuration. 0: Not in the Loop Back1 state 1: In the Loop Back1 state |
| 28 | LPBAK2ST | B'0 | R | Loop Back2 State Indicates that the MAC LTSSM is in Configured Link Loop BAK mode. Loopback test can be executed as a loopback master. Note: LTSSM has made a transition to the loopback state after executing Configuration. 0: Not in the Loop Back2 state 1: In the Loop Back2 state |
| 27 | DISST | B'0 | R | Disabled State Indicates that the MAC LTSSM is in the disabled state. Note: This bit is set only when this IP itself makes a request for link disable (sets 1 to EXPCAP4[4].LKDIS). Therefore, in Endpoint mode, 0 is always read. 0: Not in the disabled state 1: In the disabled state |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|------|---|
| 26 | HOTRSTST | B'0 | R | Hot Reset State Indicates that the MAC LTSSM is in the Hot Reset state. 0: Not in the Hot Reset state 1: In the Hot Reset state |
| 25 to 20 | Link Width | B'00_0100 | R | Link Width Indicates the width of the link established by MAC configuration. This field is valid when PCIETSTR.DLLACT = 1. B'00_0001: x1 link width B'00_0010: x2 link width B'00_0100: x4 link width Indicates the same value as the value of EXPCAP4.Negotiated Link Width. |
| 19 to 16 | Link Speed | H'1 | R | Link Speed Indicates the speed of the link established by the MAC configuration. This field is valid when PCIETSTR.DLLACT = 1. This IP supports the following link speeds: H'1: 2.5 GT/s H'2: 5.0 GT/s Indicates the same value as EXPCAP4.Current Link Speed. When read by firmware, the same value is read from the both fields. |
| 15, 14 | — | — | R | Reserved The read value is undefined. The write value should always be 0. |
| 13 | — | — | R | Reserved Ignore the value of this bit. The read value is undefined. The write value should always be 0. |
| 12 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 11, 10 | — | — | R | Reserved The read value is undefined. The write value should always be 0. |
| 9, 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | SPCHGSUC | B'0 | RW1C | Speed Change Success Indicates that speed negotiation started by this module has succeeded. Clear this bit by writing 1. Writing 0 is invalid. |
| 6 | SPCHGFAIL | B'0 | RW1C | Speed Change Fail Indicates that speed negotiation started by this module has failed. Clear this bit by writing 1. Writing 0 is invalid. |
| 5 | SPCHG | B'0 | RW1C | Speed Change Indicates that speed has been changed. When this bit is set to 1, write 1 to clear the bit. Writing 0 is invalid. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|------|---|
| 4 | SPCHGFIN | B'0 | RW1C | <p>Speed Change Finish</p> <p>Indicates that speed change processing requested by this IP has been completed.</p> <p>This bit is set to 1 at the same time that MACCTLR.SPCHG is cleared to 0.</p> <p>This bit is not updated if speed change processing is caused by the communication peer. For the usage, see section 49.3.9 (1), Changing Speed.</p> <p>When this bit is set to 1, write 1 to clear the bit.</p> <p>Writing 0 is invalid.</p> |
| 3 | LOENT | B'0 | RW1C | <p>L0 Enter</p> <p>This bit is set to 1 when a transition to L0 is made. This bit is cleared by hardware when the MACCSR.SPCHGFIN interrupt is generated.</p> <p>For the usage, see section 49.3.9 (1), Changing Speed.</p> <p>When this bit is set to 1, write 1 to clear the bit. However, the bit cannot be cleared in the L0 state.</p> <p>Writing 0 is invalid.</p> |
| 2, 1 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 0 | — | — | R | <p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p> |

49.2.98 MAC Control Register (MACCTLR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|----|-----------|----|------------|------------|----|-----------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | LTSM IS | — | LPB AK | — | SCR DIS | FRC CMP | — | SPC HG | — | — | — | — | — | — | — | — |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W | R | R/W | R | R/W | R/W | R | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 31 | LTSM DIS | B'1 | R/W | <p>LTSSM Disable</p> <p>Disables MAC LTSSM operation.</p> <p>LTSSM does not operate while this bit is 1.</p> <p>0: MAC LTSSM is not disabled and operation is performed.</p> <p>1: MAC LTSSM is disabled.</p> <p>This bit is automatically cleared when PCIETCTLR[0].CFINIT is set.</p> |
| 30 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 29 | LPBAK | B'0 | R/W | <p>Loop Back</p> <p>Requests MAC LTSSM to perform loopback transfer.</p> <p>When this bit is set, MAC LTSSM will be a loopback master after the Configuration or Recovery state, and makes a transition to the loopback state.</p> <p>Read MACSR.LPBAK1ST or LPBAK2ST to confirm that loopback state transition has been completed.</p> <p>0: MAC LTSSM will not be a loopback master.</p> <p>1: MAC LTSSM will be a loopback master.</p> |
| 28 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 27 | SCRDIS | B'0 | R/W | <p>Scramble Disable</p> <p>Disables the MAC data scramble function.</p> <p>Note: Set this bit when MACCTLR.LTSM DIS is 1. If this bit is changed with MACCTLR.LTSM DIS being 0, operation is undefined.</p> <p>0: The data scramble function is enabled.</p> <p>1: The data scramble function is disabled.</p> |
| 26 | FRCCMP | B'0 | R/W | <p>Force Compliance</p> <p>Directs a forcible transition to compliance mode.</p> <p>If set to 1 before a transition to the Polling.Active state is made, Polling.Compliance state transition is made after a timeout occurs in the Polling.Active state. If cleared to 0 during or after a transition, a transition to the Detect state is made.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 25 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 24 | SPCHG | B'0 | R/W | Speed Change Directs speed change. If set to 1 when LTSSM is in the L0, L0s, or L1 state, LTSSM makes a transition to the recovery state thus allowing speed negotiation. Only writing 1 is valid and writing 0 is invalid. This bit is read as 1 during a transition. |
| 23 to 16 | — | H'FF | R | Reserved Modifying these bits is prohibited. The write value should be H'FF. |
| 15 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | — | B'1 | R | Reserved The write value should always be 0. Note: Write 0 at initialization. |

49.2.99 PM Status Register (PMSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------|----|----|----|----|----|----|----|---------|----|----|----|----|---------|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | L1FAEG | — | — | — | — | — | — | — | PMEL1RX | — | — | — | — | PMSTATE | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | RW1C | R | R | R | R | R | R | R | RW1C | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|------|---|
| 31 | L1FAEG | B'0 | RW1C | <p>L1 Fall Edge</p> <p>This bit is set to 1 when L1 activation sequence started by writing 1 to PMCTLR.L1IATN is stopped or completed. Writing 1 clears this bit. Writing 0 is invalid.</p> <p>0: L1 activation sequence has not been completed or stopped.</p> <p>1: L1 activation sequence has been completed or stopped.</p> |
| 30 to 24 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 23 | PMEL1RX | B'0 | RW1C | <p>PM Enter L1RX</p> <p>This bit is set to 1 when PM_ENTER_L1 DLLP is received. Writing 1 clears this bit. Writing 0 is invalid.</p> <p>0: PM_ENTER_L1 DLLP has not been received.</p> <p>1: PM_ENTER_L1 DLLP has been received.</p> |
| 22 to 19 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 18 to 16 | PMSTATE | B'000 | R | <p>PMSTATE</p> <p>Indicates the state of the power management state machine. These bits cannot be written to.</p> <p>B'000: LDn state</p> <p>B'001: L0 state</p> <p>B'011: L1 state</p> <p>B'100: L0s state</p> <p>Other than above: Reserved</p> |
| 15 to 0 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

49.2.100 PM Control Register (PMCTRLR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | L1IATN | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 | L1IATN | B'0 | R/W | <p>L1 Initiation</p> <p>Writing 1 starts a transition to L1. Writing 0 is invalid.</p> <p>This bit is read as 1 during a transition to L1.</p> <p>This bit is 0 when the PM state makes a transition from L0 to another state.</p> <p>0: L1 transition sequence has not been started.</p> <p>1: Transition processing to the L1 state is in progress.</p> |
| 30 to 0 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

49.2.101 MAC Interrupt Enable Register (MACINTENR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|-------|----|----|----|----|----|----|---------------|----------------|------------|---------------|------------|-----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | LKTRE | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | SPCHG SUCE | SPCHG FAILE | SPCHG E | SPCHG FINE | LOENT E | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 30 | LKTRE | B'0 | R/W | Link Training Enable Enables generation of the interrupt by link training. When this bit is set to 1, the interrupt is generated. 0: The interrupt by link training is disabled. 1: The interrupt by link training is enabled. |
| 29 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | SPCHGSUCE | B'0 | R/W | Speed Change Success Enable Enables generation of the interrupt by Speed Change Success. When this bit is set to 1, the interrupt is generated. 0: The interrupt by Speed Change Success is disabled. 1: The interrupt by Speed Change Success interrupt is enabled. |
| 6 | SPCHGFAILE | B'0 | R/W | Speed Change Fail Enable Enables generation of the interrupt by Speed Change Fail. When this bit is set to 1, the interrupt is generated. 0: The interrupt by Speed Change Fail is disabled. 1: The interrupt by Speed Change Fail is enabled. |
| 5 | SPCHGE | B'0 | R/W | Speed Change Enable Enables generation of the interrupt by speed change. When this bit is set to 1, the interrupt is generated. 0: The interrupt by speed change is disabled. 1: The interrupt by speed change is enabled. |
| 4 | SPCHGFINE | B'0 | R/W | Speed Change Finish Enable Enables generation of the interrupt by completion of speed change processing caused by this IP. When this bit is set to 1, the interrupt is generated. 0: The interrupt by completion of speed change processing caused by this IP is disabled. 1: The interrupt by completion of speed change processing caused by this IP is enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 3 | L0ENTE | B'0 | R/W | L0 Enter Enable Enables generation of the interrupt by L0 transition. When this bit is set to 1, the interrupt is generated. 0: The interrupt by L0 transition is disabled. 1: The interrupt by L0 transition is enabled. |
| 2 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.102 PM Interrupt Enable Register (PMINTENR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------|----|----|----|----|----|----|----|--------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | L1FAE GE | — | — | — | — | — | — | — | PMEL1 RXE | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | L1FAEGE | B'0 | R/W | <p>L1 Fall Edge Enable</p> <p>Enables generation of the interrupt by L1FAEG. When this bit is set to 1, the interrupt is generated.</p> <p>0: The interrupt by L1FAIEdGe is disabled.</p> <p>1: The interrupt by L1FAIEdGe is enabled.</p> |
| 30 to 24 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 23 | PMEL1RXE | B'0 | R/W | <p>PM_Enter_L1_RX Enable</p> <p>Enables generation of the interrupt by PMEnterL1RX. When this bit is set to 1, the interrupt is generated.</p> <p>0: The interrupt by PMEnterL1RX is disabled.</p> <p>1: The interrupt by PMEnterL1RX is enabled.</p> |
| 22 to 0 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

49.2.103 MAC Status Register 2 (MACS2R)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|-------------------|----|----|--------------------|----|----|----|----|----|----------------------------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | Compliance Status | — | — | Initial Link Width | | | | | | Current Support Link Speed | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | Max Support Link Speed | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------------|---------------|-----|---|
| 31, 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29, 28 | Compliance Status | B'00 | R | Compliance Status Indicates the compliance pattern transmission state. B'00: Compliance patterns are not being transmitted. B'01: Normal compliance patterns are being transmitted. B'10: Modified compliance patterns are being transmitted. |
| 27, 26 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25 to 20 | Initial Link Width | B'00_0100 | R | Initial Link Width Indicates the width of the initially established link. These bits are only valid when PCIETSTR.DLLACT = 1. B'00_0001: x1 link width B'00_0010: x2 link width B'00_0100: x4 link width |
| 19 to 16 | Current Support Link Speed | H'1 | R | Current Support Link Speed Indicates the maximum link speed currently supported by the peer. These bits are only valid when PCIETSTR.DLLACT = 1. H'1: 2.5 GT/s H'2: 5.0 GT/s |
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | Max Support Link Speed | H'1 | R | Max Support Link Speed Indicates the maximum link speed supported by the peer after the link establishment. These bits are only valid when PCIETSTR.DLLACT = 1. H'1: 2.5 GT/s H'2: 5.0 GT/s |

49.2.104 MAC Control Register 2 (MACCTL2R)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|---------------|--------------|----|----|-------------|----|----|----|------------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | EXITLP BAK | FRCLP BAK | — | — | EXIT CMP | — | — | — | Force Link Speed | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R | R | R/W | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|--|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 | EXITLPBAK | B'0 | R/W | EXIT Loop Back If set to 1 during loopback slave operation in the Loopback.Active state, a transition to the Detect state is made. Only writing 1 is valid and writing 0 is invalid. This bit is read as 1 during a transition. |
| 10 | FRCLPBAK | B'0 | R/W | Force Loop Back Sets the mode when MACCTL1R.LPBAK is set to 1. When this bit is 1, a transition to the Loopback.Active state is made after a specified time elapsed even if TSOS has not been received from the peer in the Loopback.Entry state. |
| 9, 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | EXIT CMP | B'0 | R/W | EXIT Compliance When reception of TS1 (Compliance) causes transition to Polling.Compliance state, setting this bit to 1 causes transition to the Detect state. Only writing 1 is valid and writing 0 is invalid. This bit is read as 1 during a transition. |
| 6 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | Force Link Speed | H'2 | R/W | Force Link Speed Specifies the upper limit of supported rate and loopback compliance rate to be conveyed to the peer when executing a loopback master operation or forcible compliance. H'1: 2.5 GT/s H'2: 5.0 GT/s |

49.2.105 MAC Speed Change Setting Register (MACCGSPSETR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------------|----|----|----|----|----|----|----|----|----|----|----|-------------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SPCNG RSN | — | — | — | — | — | — | — | — | — | — | — | Target Link Speed | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------|---------------|-----|--|
| 31 | SPCNGRSN | B'0 | R/W | Speed Change Reason Indicates the factor of link speed change. 0: An intentional factor 1: A link reliability problem |
| 30 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 to 16 | Target Link Speed | H'2 | R/W | Target Link Speed Specifies the speed of the link to be established using the link speed change in Endpoint mode. (In Root Port mode, EXPCAP12[3:0].Target Link Speed is used.) H'1: 2.5 GT/s H'2: 5.0 GT/s |
| 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

49.2.106 AXI Cache Control Register (AXI_CACHE_REG)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | — | √ |

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----------|-----|-----|-----|----|----|----|----|----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | MARCACHE | | | | — | — | — | — | MAWCACHE | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | — | B'1 | R | Reserved This bit is always read as 1. The write value should always be 1. |
| 30 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 8 | MARCACHE | H'0 | R/W | B'0000 Device Non-bufferable B'0001 Device Bufferable B'0010 Normal Non-cacheable Non-bufferable B'0011 Normal Non-cacheable Bufferable B'0110 Write-through No-allocate B'1110 (0110) Write-through Read-allocate B'1010 Write-through Write-allocate B'1110 Write-through Read and Write-allocate B'1111 (0111) Write-back No-allocate B'1111 Write-back Write-allocate B'1111 Write-back Read and Write-allocate |
| 7 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 3 to 0 | MAWCACHE | H'0 | R/W | B'0000 Device Non-bufferable B'0001 Device Bufferable B'0010 Normal Non-cacheable Non-bufferable B'0011 Normal Non-cacheable Bufferable B'0110 Write-through No-allocate B'1110 (0110) Write-through Read-allocate B'1010 Write-through Write-allocate B'1110 Write-through Read and Write-allocate B'1111 (0111) Write-back No-allocate B'1111 Write-back Write-allocate B'1111 Write-back Read and Write-allocate |

[RZ/G2E]

| | | | | | | | | | | | | | | | | |
|----------------|------|----|----|----|----------|-----|-----|-----|----|----|----|----|----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | RATE | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | MARCACHE | | | | — | — | — | — | MAWCACHE | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | RATE | B'1 | R/W | Aurora-mode 0: 2.5GT/s 1: 5.0GT/s |
| 30 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 8 | MARCACHE | H'0 | R/W | B'0000 Device Non-bufferable B'0001 Device Bufferable B'0010 Normal Non-cacheable Non-bufferable B'0011 Normal Non-cacheable Bufferable B'0110 Write-through No-allocate B'1110 (0110) Write-through Read-allocate B'1010 Write-through Write-allocate B'1110 Write-through Read and Write-allocate B'1111 (0111) Write-back No-allocate B'1111 Write-back Write-allocate B'1111 Write-back Read and Write-allocate |
| 7 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 3 to 0 | MAWCACHE | H'0 | R/W | B'0000 Device Non-bufferable B'0001 Device Bufferable B'0010 Normal Non-cacheable Non-bufferable B'0011 Normal Non-cacheable Bufferable B'0110 Write-through No-allocate B'1110 (0110) Write-through Read-allocate B'1010 Write-through Write-allocate B'1110 Write-through Read and Write-allocate B'1111 (0111) Write-back No-allocate B'1111 Write-back Write-allocate B'1111 Write-back Read and Write-allocate |

49.2.107 PHY clock Register (PHY_CLK)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----------------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | PHY_REF_USE_PAD | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | PHY_REF_USE_PAD | B'1 | R/W | Select reference clock source 0: Internal reference clock 1: External reference clock |
| 7 to 0 | — | H'19 | R | Reserved These bits are always read as H'19. The write value should always be H'19. |

For more detail, refer to section 49.3.14.

49.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The following describes the operation of this module.

Note: The following operation flowcharts are each divided in two parts and described. One part covers the H/W, hardware operation of this module and the other covers the S/W, software operation to control this module. The arrows originating from the outside of the figure frame indicate that the event arises from the user program.

In the explanation of the operation flows, interrupt sources are described in the H/W part. The S/W should first verify a given interrupt flag and then clear it.

If generation of interrupt sources is expected, enable each of the interrupt sources referring to Figure 49.31 and 49.32.

49.3.1 Initialization

(1) Standby State Cancellation

- After canceling a reset, cancel the module standby state. For the module standby state, refer to section “12. Module Standby and Software Reset”. This module is in the module standby state after a power-on reset is canceled.

(2) Initial Setting of PCI Express

- Figure 49.2 shows the PCIEC operation in initial setting.
- After start-up processing for the physical layer is completed, the initial setting is executed to start operation as a PCI Express. Until the initial setting is completed, this module cannot work correctly as a PCI Express device. Therefore, a link is not established immediately after a reset is canceled. When the initial setting is completed, enable the PCIETCTLR.CFINIT to start operation.
- Set the bits in the configuration registers as directed in the related register description, if there is a direction on initialization.
- Set H'000 in the Next Capability Offset [31:20] bits in VCCAP0.
- Be sure to write the initial value (= H'80FF 0000) to MACCTLR before enabling PCIETCTLR.CFINIT.
- When a register is used with the default setting, setting the register is not required.

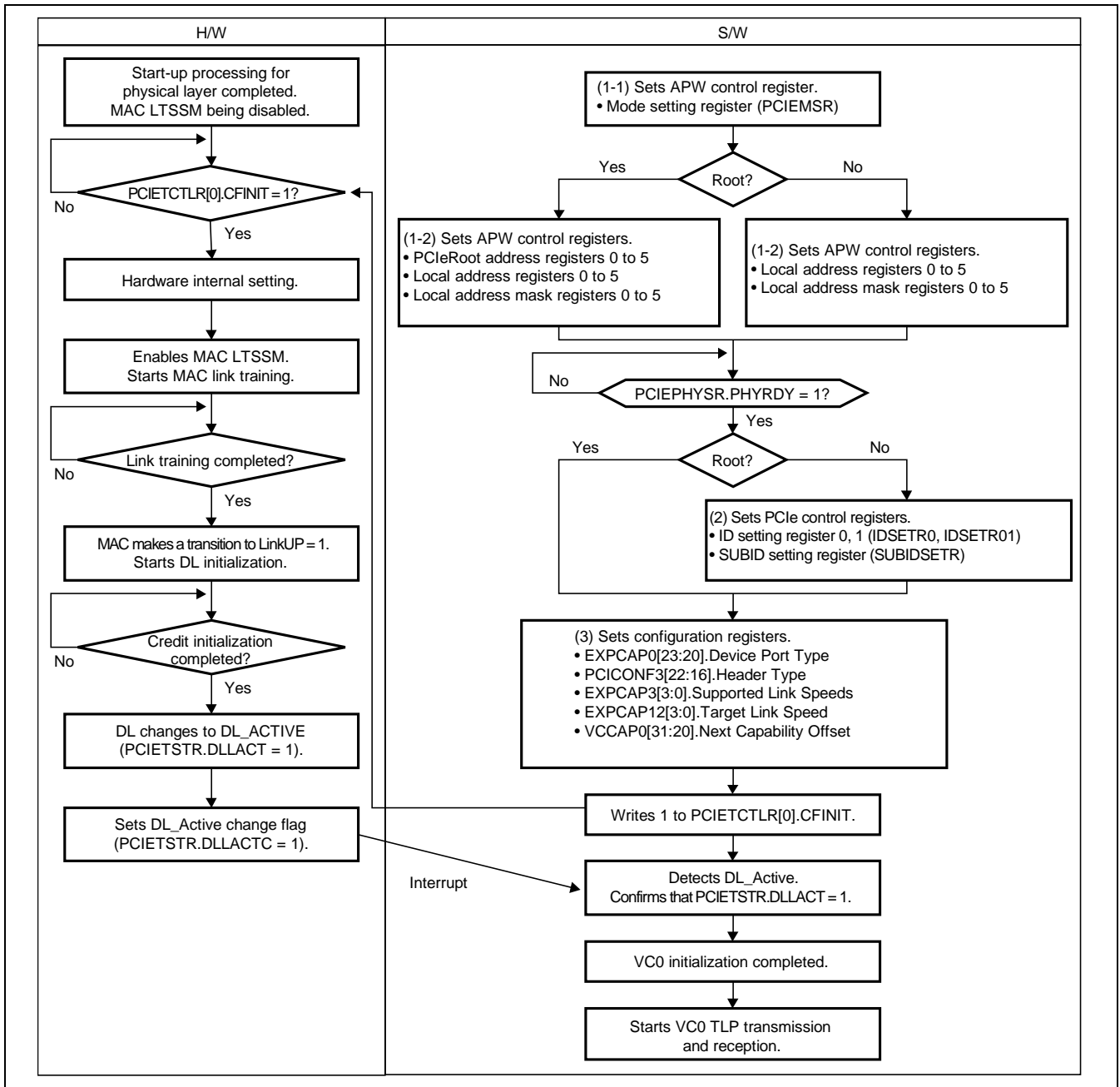


Figure 49.2 Initial Setting of PCIEC

(3) DL_Down Operation

Figure 49.3 shows the operation when a link goes down (LinkUP = 0) in the state where a link is established, thus causing DL_Down. When DL_Down is caused, PCIETCTLR.DLDOWN is set to 1. Since the current data is invalidated, all the resources of the data transmitter/receiver of this module are initialized.

- Upon DL_Down, a write to the transmission buffer is disabled thus stopping transmission.
- Upon DL_Down, the receive buffer is forcibly cleared.
- If a link goes down, transmission and reception are suspended; therefore, data before and after DL_Down becomes illegal. Since the specific process here differs depending on the application, the operation is simply indicated as “Processes errors.” in the flowchart shown in Figure 49.3.

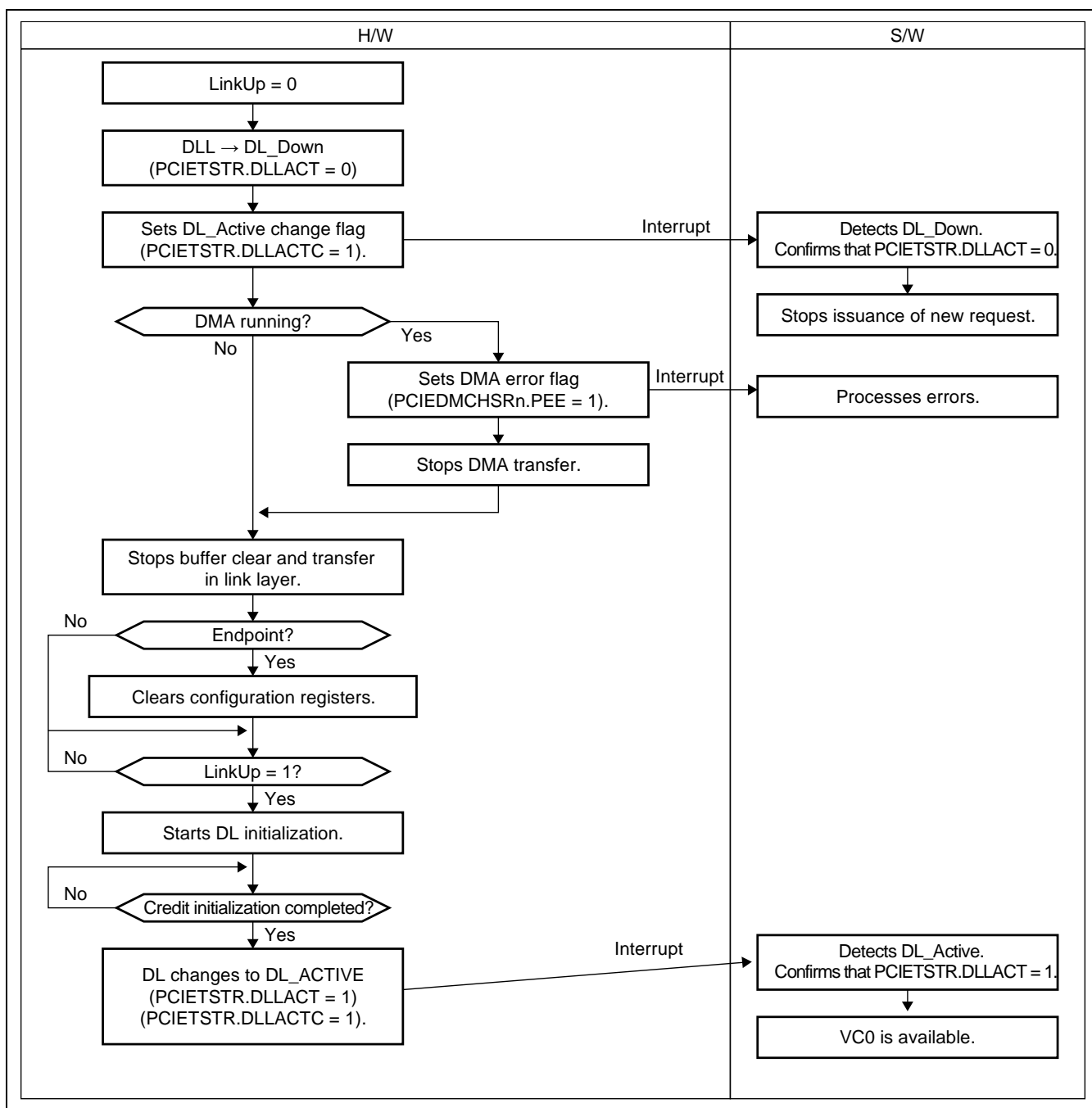


Figure 49.3 DL_Down Operation

(4) Hot Reset Operation

(a) Hot Reset Transmission

In Root Port mode, to send a hot reset to a downstream port, write 1 to the secondary bus reset bit in the configuration register. Before issuing a hot reset, confirm that this module has completed transfer.

(b) Hot Reset Reception

In Endpoint mode, when a hot reset is received from an upstream port, the LTSSM on the link layer makes a transition to the hot reset state. This causes LinkUp = 0 and then DL_Down = 1. After this, the same operation is performed as the operation upon the DL_Down.

49.3.2 Configuration Cycle

When using this module as a Root Port, a configuration cycle must be generated to configure the connection-target device. A configuration cycle refers to the process of checking the status of the configuration register of the connection-target Endpoint, switch, or bridge through the use of configuration access, and assigning values to the configuration registers of the Root Port itself and the Endpoint according to the result of the checking. When the Root Port accesses its own configuration register, such access is normally made through the internal bus.

This section describes the transmission (write) and reception (read) operations of the configuration access, and the items to be specified during a configuration cycle.

When this module is used as an Endpoint, it receives a configuration access from a Root Port, and accepts the initialization processing.

(1) Configuration Read Operation

Figure 49.4 shows a flow of configuration read operation.

With EXPCAP7[4].CRSVISE = 0, an error response is returned when the CRS status is received.

With EXPCAP7[4].CRSVISE = 1, response data is H'FFFF_0001 when the CRS status is received. Therefore, the first access to the peer device from which the CRS status may possibly be returned must be configuration read from register 0. Here, judgment is possible only by the VID that has read that no device is existent (including an unsupported request response) or that a CRS response is received, thus simplifying the procedure for setting the bus number of the PCI bus topology.

Automatic hardware response according to the CRS status is not supported.

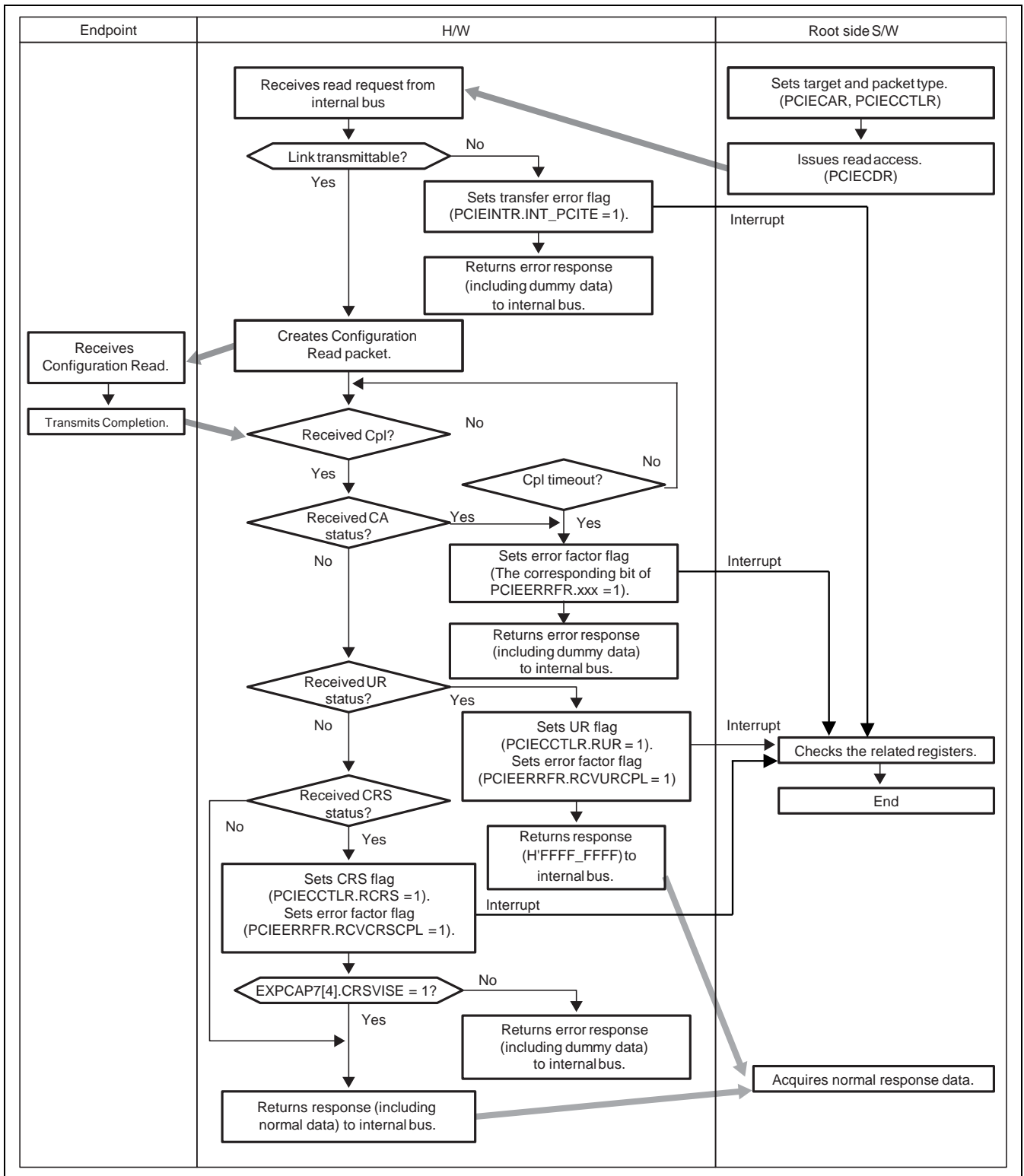


Figure 49.4 Configuration Read Operation

(2) Configuration Write Operation

Figure 49.5 shows a flow of configuration write operation.

The peer device should be the function for which configuration read has been succeeded (= the function from which CRS may never be returned).

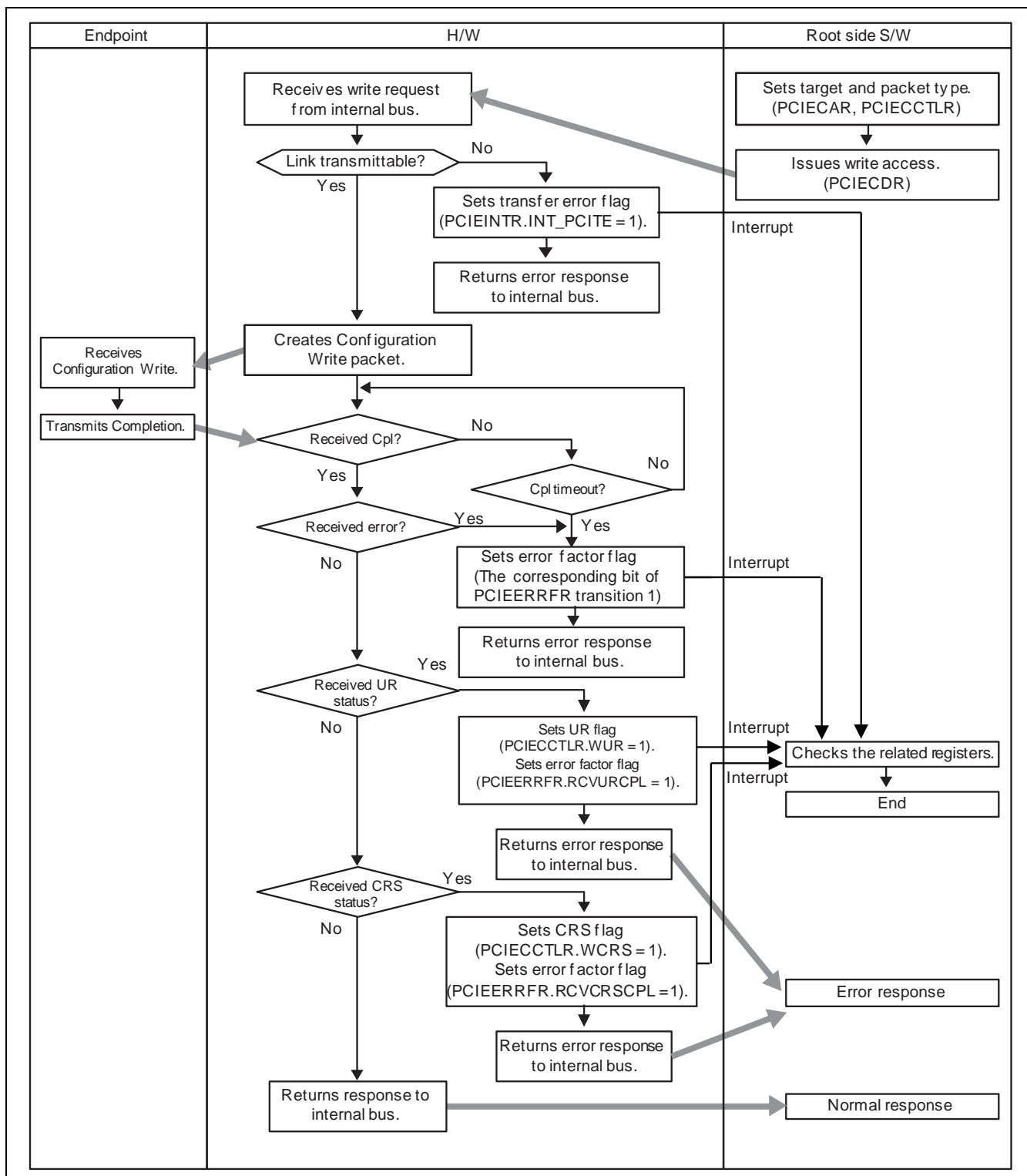


Figure 49.5 Configuration Write Operation

(3) Initialization of PCIEC in Root Port Mode

In Root Port mode, set all the Endpoints (indicated as system in the flow chart) connected to the PCIEC after link establishment. Figure 49.6 shows a sample setting flowchart.

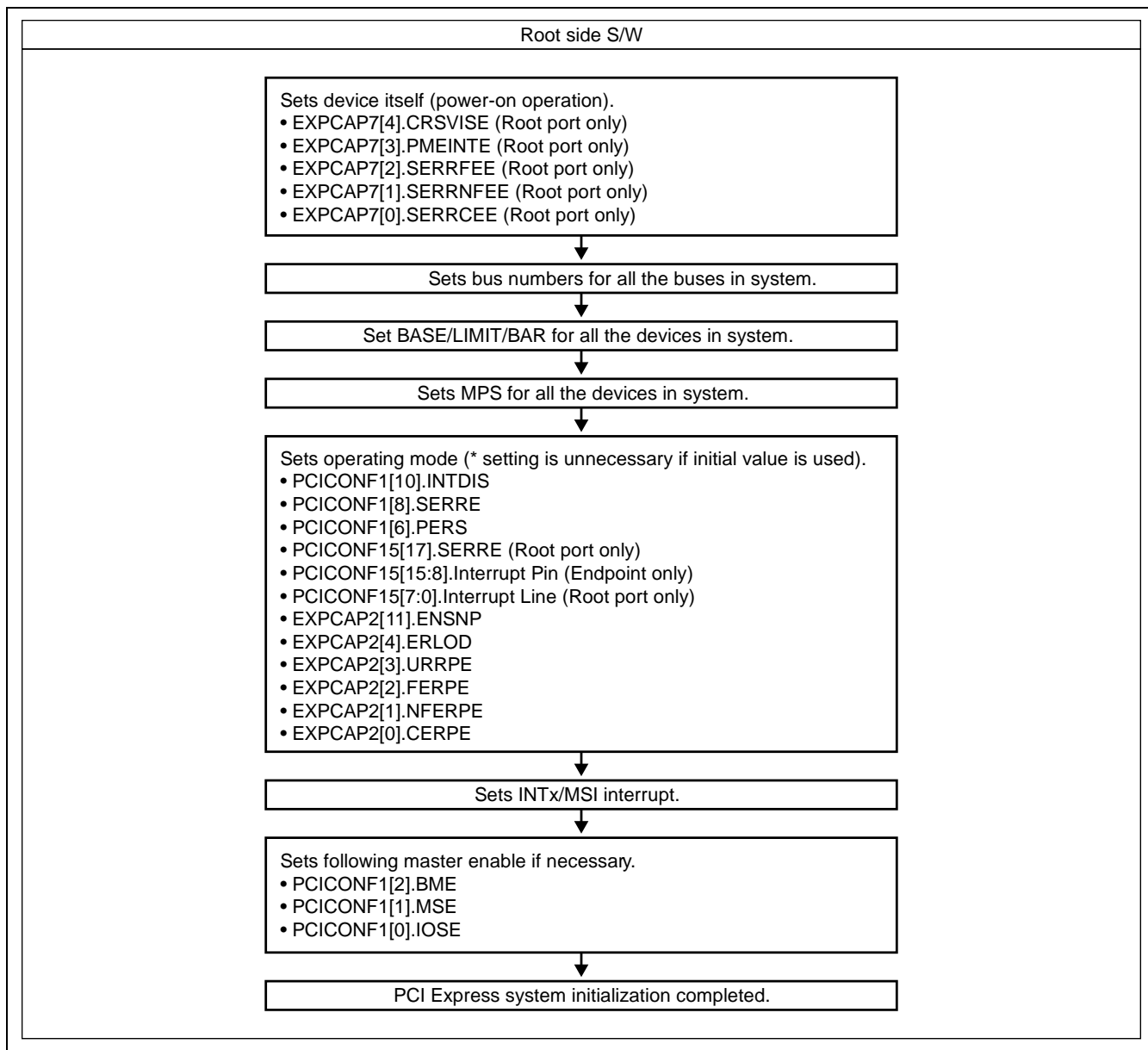


Figure 49.6 PCIEC Initialization in Root Port Mode

49.3.3 Messages and MSI

This module supports transmission and reception of the messages and MSI (Message Signaled Interrupt) defined in the PCI Express standard. The messages allow interrupt signaling, power management, error information notification, and unlocking. The MSI is used instead of INTx to signal interrupts. The following sections describe transmission and reception of the messages and MSI.

(1) Overview of Messages

This module automatically transmits and receives messages that indicate the error state of a link through hardware (Figure 49.7 (a) and 49.8 (a)). The other types of messages can be transmitted and received only by writing and reading the registers of this module (Figure 49.7 (b) and 49.8 (b)). The PM-related messages and INTx messages have separate 1-bit fields corresponding to the messages and simply writing to the pertinent bit allows this module to create and transmit a message. In message reception, reading the status of the pertinent bit completes reception.

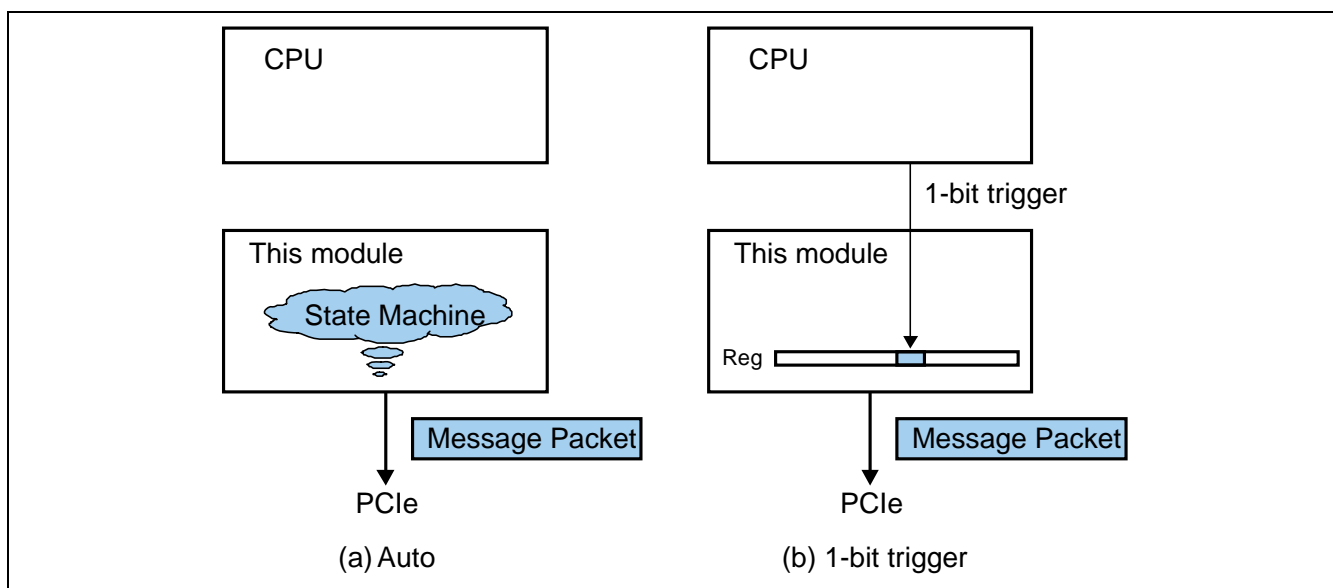


Figure 49.7 Message Transmission Methods

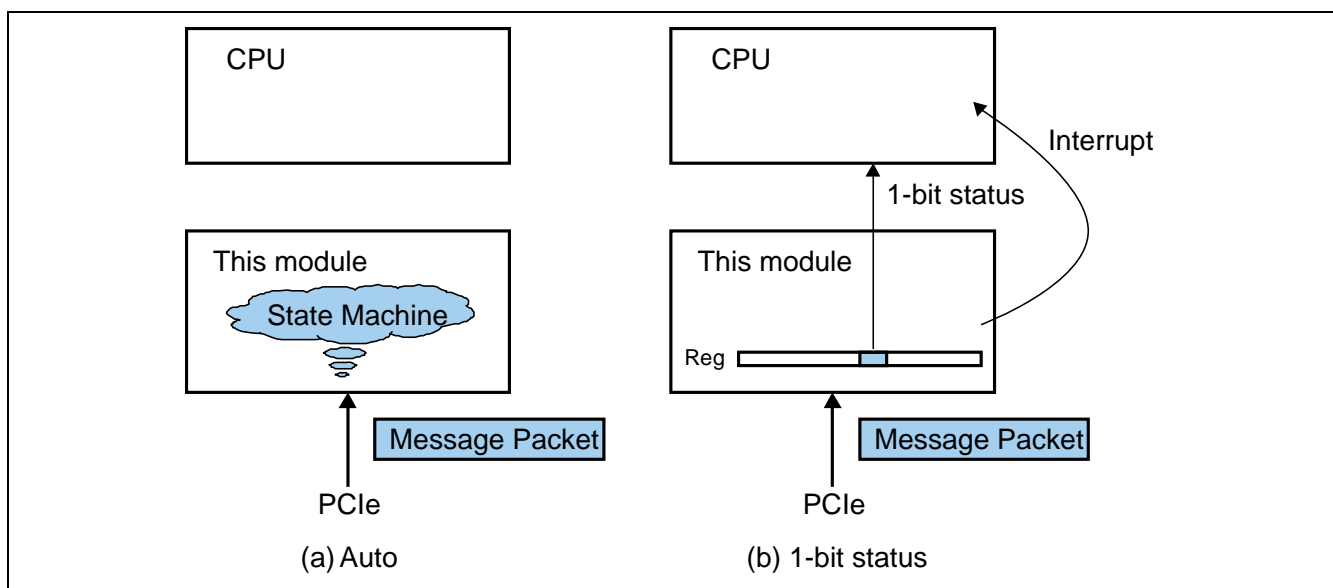


Figure 49.8 Message Reception Methods

(2) Message Transmission

Messages can be transmitted by either of the following two methods.

(a) Automatic Message Transmission by Hardware

This module automatically sends a necessary message when an error occurs. The following messages are automatically sent.

- **ERR_COR:** When transmission of the correctable error message is enabled, this message is automatically sent if a correctable error occurs.
- **ERR_NONFATAL:** When transmission of the non-fatal error message is enabled, this message is automatically sent if a non-fatal error occurs.
- **ERR_FATAL:** When transmission of the fatal error message is enabled, this message is automatically sent if a fatal error occurs.

(b) Initiation of Message Transmission by 1-bit Writing through Software

This module sends unlock or INTx interrupt messages under control by software. To send these messages, access the following registers.

- **Unlocking (unlock message transmission):** Writing 1 to PCIEUNLOCKCR[0].ASTUNLOCK allows transmission of the unlock message and unlocking. The unlock messages can be transmitted only by the Root Port.
- **Power management message transmission (PME_* message transmission):** Writing 1 to the bits corresponding to the messages in the power management message transmission control register allows transmission of the messages given below: PME_TO_ACK, PME_TOFF, and PM_PME.
- **INTx interrupt generation and cancellation (Assert_INTx/Deassert_INTx message transmission: x = A, B, C, or D):** Writing 1 to PCIEINTXR[16].ASTINTX allows transmission of the Assert_INTx message corresponding to the INTx specified with PCICONF15[15:8].Interrupt Pin thus allowing generation of the INTx interrupt. Writing 0 to PCIEINTXR[16].ASTINTX allows transmission of the Deassert_INTx message corresponding to the INTx specified with PCICONF15[15:8].Interrupt Pin thus allowing cancellation of the INTx interrupt. The INTx interrupts can be generated and canceled only by the Endpoint.

(3) Message Reception

In this module, reception of some messages is automatically handled by hardware. For the message reception processes not performed through hardware, refer to the status and process reception through software. Note that the messages except Vendor_Defined_Message are required by the PCI Express standard to be transmitted and received with TC0. If such a message is received with the other TC, the PCIEC considers it as a malformed TLP (fatal error) thus performing error processing automatically.

(a) Automatic Message Reception by Hardware

- **Assert_INTx (x is A to D):** When this message is received in Root Port mode, PCIEINTXR[0, 1, 2, 3].INTx are set. The INTx interrupt is generated here when PCICONF15[7:0].Interrupt Line is not H'FF. When received in Endpoint mode, this message is considered as a malformed TLP and error processing is performed.
- **Deassert_INTx (x is A to D):** When this message is received in Root Port mode, PCIEINTXR[0, 1, 2, 3].INTx are cleared. The INTx interrupt is canceled here (only in the Root Port) when PCICONF15[7:0].Interrupt Line is not H'FF. When received in Endpoint mode, this message is considered as a malformed TLP and error processing is performed.
- **ERR_COR:** When this message is received, the appropriate value is set to the configuration register to indicate reception of a correctable error and the INT_PCICERR interrupt is generated.
- **ERR_NONFATAL:** When this message is received, the appropriate value is set to the configuration register to indicate reception of a non-fatal error and the INT_PCINFERR interrupt is generated.
- **ERR_FATAL:** When this message is received, the appropriate value is set to the configuration register to indicate reception of a fatal error and the INT_PCIFERR interrupt is generated.

- **Unlock:** The unlock message is discarded since neither the PCI Express Endpoint nor Root Port accepts a lock request.
- **Set_Slot_Power_Limit:** (When this message is received, the power information specified with the message is automatically reflected to the configuration register (EXPCAP1[27:18]) through hardware.)

(b) Reception of Messages Requiring Software Processing

The module hardware detects reception of the following messages but does not automatically process them. Reception of these messages must be checked and processed by software as described below.

- PM_PME

A message reception can be checked via the following registers:

- PCIERMSGR
- PCIERMSGIER
- EXPCAP8

In the PCIERMSGR, bits are defined for each specified message, and 1 is set to a bit corresponding to the received message when a message is received. In this situation, if the corresponding bit in the PCIERMSGIER is set to 1 and if the PCIEINTER[5].INT_PCIMESE is set to 1, a PCIMES interrupt is generated.

Table 49.7 shows the messages, reception of which is detected by hardware and which are processed by software.

Table 49.7 Messages Whose Reception can be Detected by the PCIEC

| Register | Bit No. | Register | Bit No. | Receive Message |
|----------|---------|-----------|---------|----------------------|
| PCIEINTR | 5 | PCIERMSGR | 12 | Set_Slot_Power_Limit |
| | 30 | EXPCAP8 | 17 | PM_PME |

(4) Overview of MSI

This section describes the MSI interrupts.

This module supports MSI interrupts. Whether the PCI Express system should use INTx or MSI interrupts is determined by the Root Port during a configuration cycle. When this module is to be used as a Root Port, whether INTx or MSI interrupt is used should be determined during a configuration cycle, and the results of the determination must be set in the configuration register.

When this module is used as a Root Port, a maximum of 32 interrupt vectors can be used as MSI interrupts. In addition, this module supports both Per Vector Masking and the transmission of 64-bit address messages.

(5) MSI Transmission

MSI can be transmitted only in Endpoint mode. Figure 49.9 shows an MSI transmission flow.

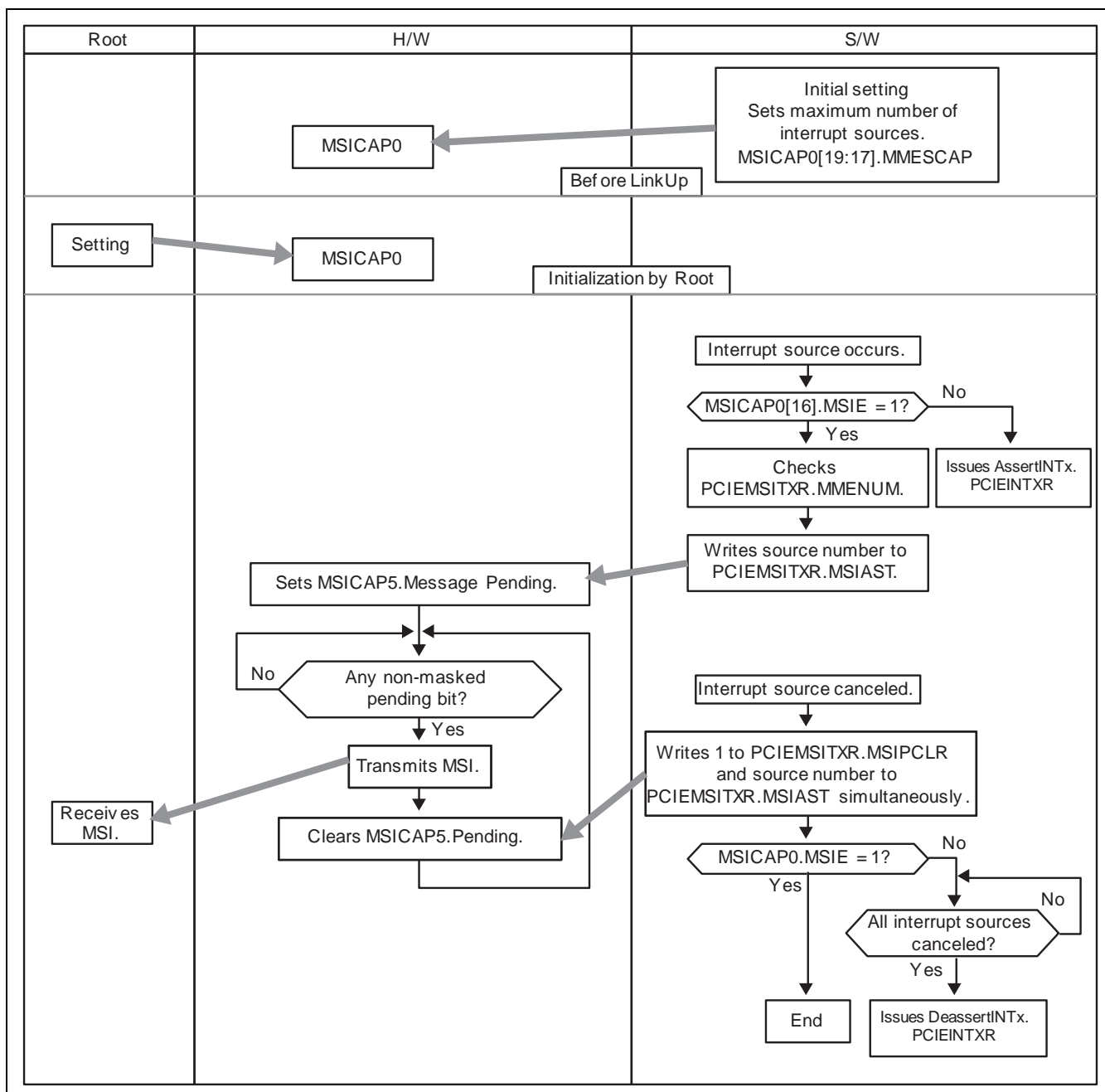


Figure 49.9 MSI Transmission

(6) MSI Reception

MSI can be received only in Root Port mode. The following describes a MSI reception flow.

(a) MSI Interrupt Setting (Root Port)

When this module is used as a Root Port, whether INTx or MSI interrupts are used should be determined during a configuration cycle. If MSI interrupts are selected, the following settings should be performed.

(1) Assigning Interrupt Vectors to Endpoints

Reference the MSICAP0[19:17].MMESCAP of the Endpoint device, and check the number of MSI interrupt vectors required by devices. Based on the result of the checking, the number of vectors to be assigned to the Endpoints is determined so that the number of vectors to be assigned to all devices is 32 or less, the number of vectors assigned to each Endpoint is 1 or greater and less than or equal to the requested number of assignments.

The number of vectors assigned to each Endpoint should be set in the MSICAP0[22:20].MMESE.

(2) Setting Receive Message Address

Specify the address to which the MSI is received to PCIEMSIAR and PCIEMSIAR and also enable MSI address decoding.

- Notes:
1. When there is a switch or bridge between the Endpoint and this module, set the address so that the memory write request to the address specified by the switch or bridge should be routed to the Root Port, or change the address specified by the switch or bridge and set the address so that the memory write request to the address specified by this module should be routed to the Root Port instead.
 2. When the specified address falls within the target area, 1DW memory write request to the specified address is not handled as a target transfer, but as an MSI message.

(3) Setting Endpoint Message Address

Set the transmission destination address of the MSI message to MSICAP1[31:2].Lower Message Address and MSICAP2[31:0].Upper Message Address of the Endpoint device.

The transmission destination address of MSI message should be the address set to PCIEMSIAR and PCIEMSIAR.

(4) Setting Message Data

Set the message data for transmission to the MSICAP3[15:0].Message Data of the Endpoint device.

In the 32-bit MSI vectors supported by this module, bits [12:8] in write data must specify the starting vector number of the MSI vectors assigned to the Endpoint, and all other bits must be cleared to 0. Figure 49.10 shows an example of an MSI message data to be set in an Endpoint.

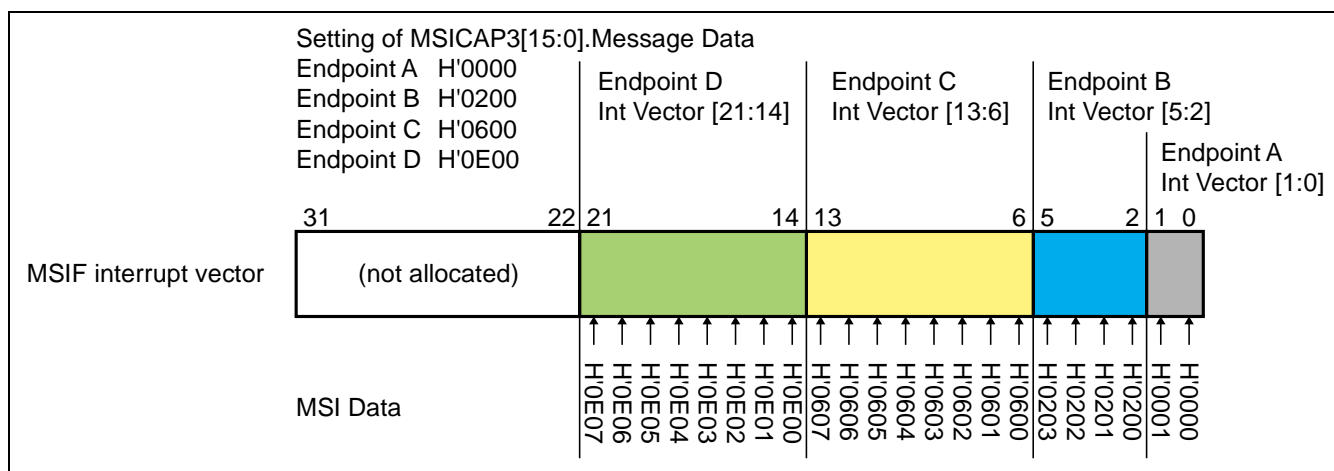


Figure 49.10 MSI Message Data to be Set in Endpoint
 (an example of setting where Vectors 2, 4, 8 and 8 are allocated to Endpoints A, B, C and D, respectively)

(5) Enabling MSI Interrupts of the Peer Device

Set MSICAP0[16].MSIE of all the devices to 1.

(6) Enabling MSI Interrupts

Set the MSIIE bits.

49.3.4 Overview of Transmission and Reception

PCI Express transfers are classified into two types of requests and one type of completion.

The following three methods are provided for PCI Express transactions. For details on each transfer, refer to the following sections.

1. PIO transfer: A request is issued from the internal bus side to a device on the PCI Express.
2. Target transfer: A request is received by the internal bus side from the PCI Express.
3. DMA transfer: A great quantity of data is automatically written and read between the PCI Express and the internal bus without intervention of the CPU.

Table 49.8 shows the relationship between two types of requests and PCI Express transactions and also shows the transaction issuance methods with this module.

Table 49.8 Relationship between Posted/Non-Posted Requests and PCI Express Transactions

| Transaction Type | Transfer Type | Completion | Device Type that can Issue Transaction | Transaction Issuance Method with This Module | Operation |
|---------------------|---------------|--------------|--|---|---|
| Memory Write | Posted | Not provided | Root Port & Endpoint | PIO, DMA, (MSI Endpoint only) | Data is written to the memory of the communication peer. 1 byte to 4 Kbytes of data not exceeding MPS* ¹ can be written. |
| Memory Read | Non-Posted | Provided | Root Port & Endpoint | PIO, DMA | A quantity of data not exceeding MRRS* ² is read from the memory of the communication peer. The read data is returned with a completion separately from this request; no data is included in this request. |
| I/O Write | Non-Posted | Provided | Root Port only | PIO | Data is written to I/O. Data size is 1 DW. A completion is necessary that indicates whether this request has succeeded. |
| I/O Read | Non-Posted | Provided | Root Port only | PIO | Data is read from I/O. No data is included. The read data and a completion that indicates whether this request has succeeded are necessary. |
| Configuration Write | Non-Posted | Provided | Root Port only | PCI Express transfer control register | Configuration write is performed. Data size is 1 DW. A completion is necessary that indicates whether this request has succeeded. |
| Configuration Read | Non-Posted | Provided | Root Port only | PCI Express transfer control register | Configuration read is performed. No data is included. The read data and a completion that indicates whether this request has succeeded are necessary. |
| Message | Posted | Not provided | Root Port & Endpoint | Issued by this IP, message transmit register, message transmit module | A message request is transmitted. This is used for power management and interrupt control. |

Notes: 1. MPS: Max Payload Size
2. MRRS: Max Read Request Size

In Table 49.8, the transactions for which “Provided” is indicated in the “Completion” column, make a response with a completion. The data returned with a completion differs depending on the transaction. Table 49.9 shows the particulars of the completion contents returned. Specifically, the table shows whether completion data for each type of transactions is included or not, the devices that can issue the transaction, and some other related information.

Table 49.9 Relationship between Completion and PCI Express Transactions

| Transaction Type | Completion Data | Device Type that can Issue Transaction | Transaction Issuance Method with This Module | Operation |
|---------------------|-----------------|--|--|---|
| Memory Read | Included | Root Port & Endpoint | Automatically issued by this module. | Data requested by memory read request and status are returned. |
| I/O Write | Not included | Root Port only | Automatically issued by this module. | No data is included. Status in response to I/O write is returned. |
| I/O Read | Included | Root Port only | Automatically issued by this module. | Data size is 1 DW. The read data and a completion that indicates whether this request has succeeded are returned. |
| Configuration Write | Not included | Root Port only | Automatically issued by this module. | No data is included. Status in response to configuration write is returned. |
| Configuration Read | Included | Root Port only | Automatically issued by this module. | Data size is 1 DW. The read data and a completion that indicates whether this request has succeeded are returned. |

49.3.5 PIO Transfer (Data Transfer from Internal Bus → PCIEC → External Device)

This section describes PIO transfer. PIO transfer here refers to the transfer method to create a PCI Express packet by accessing the memory space assigned to the PCI Express via the internal bus.

(1) Overview of PIO Transfer

PIO transfer is the transfer method with which the device on the internal bus transmits a packet to the PCI Express by writing to (or reading from) the space assigned to the PCI Express. When a write access is made, the written data is stored in the data payload of the PCI Express packet. The maximum data size that can be transmitted equals to the maximum data size of an internal bus packet. The type of the PCI Express packet to be transmitted can be specified using the appropriate setting register and either memory read/write or IO read/write can be selected.

Before starting transfer, it is necessary to set the internal bus space that is transferred to the PCIEC and to set the particulars of the packet to be transmitted. Section (2) describes these settings and section (3) describes specific transfer operations.

(2) Initial Setting of PIO Transfer

The internal bus address cannot unequivocally define one transfer destination since the address space of the PCI Express, which is the transfer destination, is represented with 64 bits. Therefore, the upper address of the PCI Express space is specified using the PCIEC address upper register n ($0 \leq n \leq 3$) and PCIEC address lower register n ($0 \leq n \leq 3$). Set the PCI conversion control register n (PCIEPTCTRL n ; $0 \leq n \leq 3$) [31].PARE to indicate that such area setting is valid. Set the PCIEC address upper and lower registers at initial setting.

Table 49.10 shows the PIO transfer control registers. The PCIEC memory areas 0 to 3 are mapped to the PCI memory space or I/O space according to the setting of these registers.

Table 49.10 PIO Transfer Control Registers

| | |
|--|--|
| PCIEC address upper registers 0 to 3 (PCIEPAUR0 to PCIEPAUR3) | Address (upper 32 bits) of the PCI address space to which the PCIE memory areas 0 to 3 are to be mapped |
| PCIEC address lower registers 0 to 3 (PCIEPALR0 to PCIEPALR3) | Address (lower 32 bits) of the PCI address space to which the PCIE memory areas 0 to 3 are to be mapped |
| PCIEC address mask registers 0 to 3 (PCIEPAMR0 to PCIEPAMR3) | Specify the size in the PCIE memory areas 0 to 3 to be mapped to the PCI address space. |
| PCIEC conversion control registers 0 to 3 (PCIEPTCLR0 to PCIEPTCLR3) | Specify whether the PCI space is valid or invalid. Specify the transfer destination space (PCI memory space or PCI IO space). Specify attributes (TC, LOCK, ZLR, and ATTR) for conversion. |

Table 49.11 shows the PCI Express memory area addresses on the AXI bus space.

Table 49.11 AXI Bus Space Address Map

| Memory Area | Address | Physical Address Size |
|----------------|----------------------------------|-----------------------|
| PCIE0 memory 0 | H'00_FE10_0000 to H'00_FE1F_FFFF | 1 Mbyte |
| PCIE0 memory 1 | H'00_FE20_0000 to H'00_FE3F_FFFF | 2 Mbytes |
| PCIE0 memory 2 | H'00_3000_0000 to H'00_37FF_FFFF | 128 Mbytes |
| PCIE0 memory 3 | H'00_3800_0000 to H'00_3FFF_FFFF | 128 Mbytes |
| PCIE1 memory 0 | H'00_EE90_0000 to H'00_EE9F_FFFF | 1Mbyte |
| PCIE1 memory 1 | H'00_EEA0_0000 to H'00_EEBF_FFFF | 2Mbytes |
| PCIE1 memory 2 | H'00_C000_0000 to H'00_C7FF_FFFF | 128 Mbytes |
| PCIE1 memory 3 | H'00_C800_0000 to H'00_CFFF_FFFF | 128 Mbytes |

The following shows the example of mapping the internal bus space to the PCIEC address space.

Example of mapping the internal bus space to the PCIEC address space:

```

PCIEC address lower register 0[31:0] = H'0000_0000
PCIEC address lower register 1[31:0] = H'5640_0000
PCIEC address lower register 2[31:0] = H'2000_0000
PCIEC address lower register 3[31:0] = H'8000_0000
PCIEC address upper register 0[31:0] = H'0000_0000
PCIEC address upper register 1[31:0] = H'0000_0008
PCIEC address upper register 2[31:0] = H'0000_0000
PCIEC address upper register 3[31:0] = H'F000_0000
PCIEC address mask register 0[31:0] = H'0000_FF80
PCIEC address mask register 1[31:0] = H'001F_FF80
PCIEC address mask register 2[31:0] = H'1FFF_FF80
PCIEC address mask register 3[31:0] = H'1FFF_FF80
PCIEC conversion control register 0[31:0] = H'8000_0100
PCIEC conversion control register 1[31:0] = H'8000_0000
PCIEC conversion control register 2[31:0] = H'8000_0000
PCIEC conversion control register 3[31:0] = H'8000_0000

```

Figure 49.11 shows an example of mapping the internal bus space to the PCI Express space with the above setting.

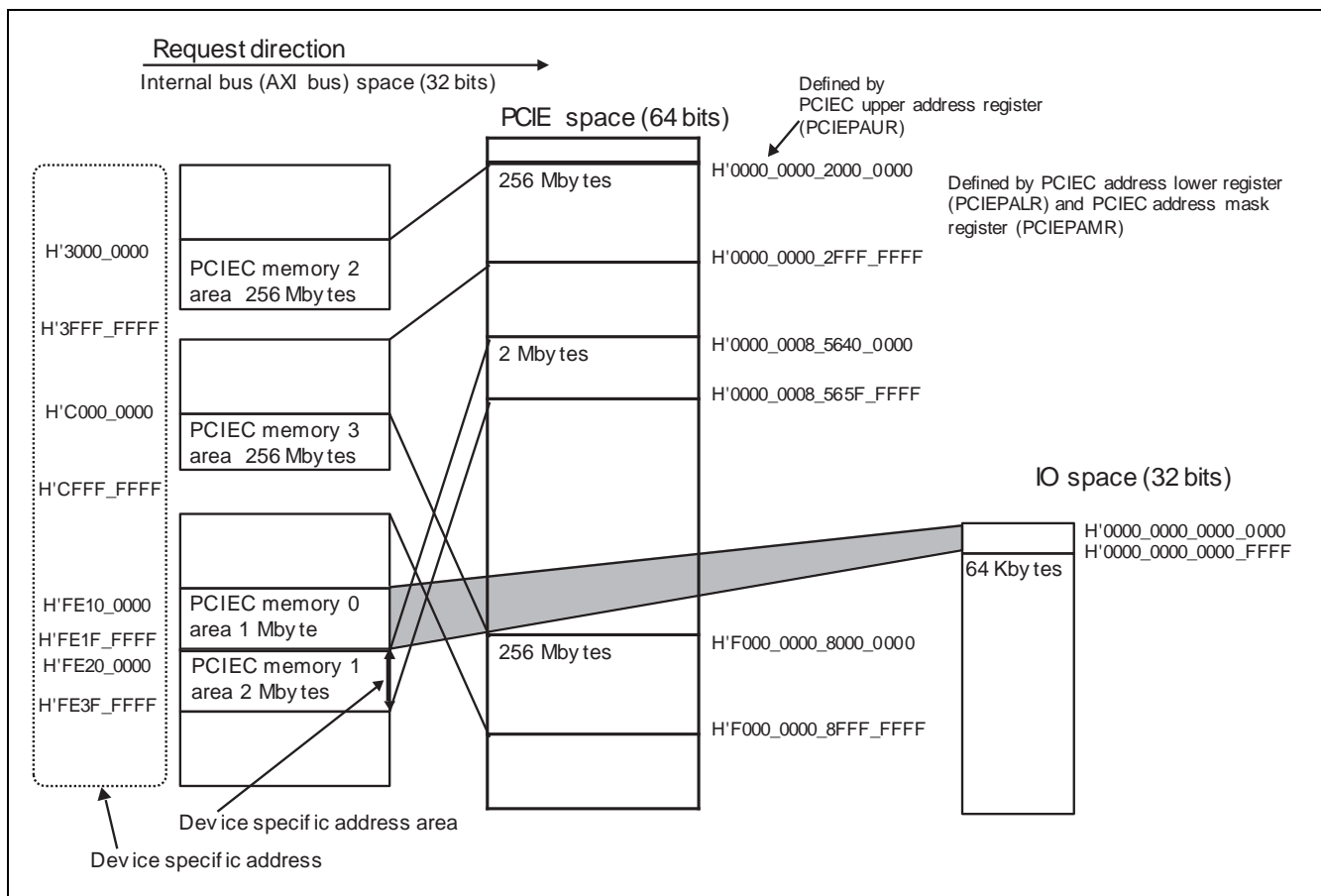


Figure 49.11 Example of Address Mapping for PIO Transfer

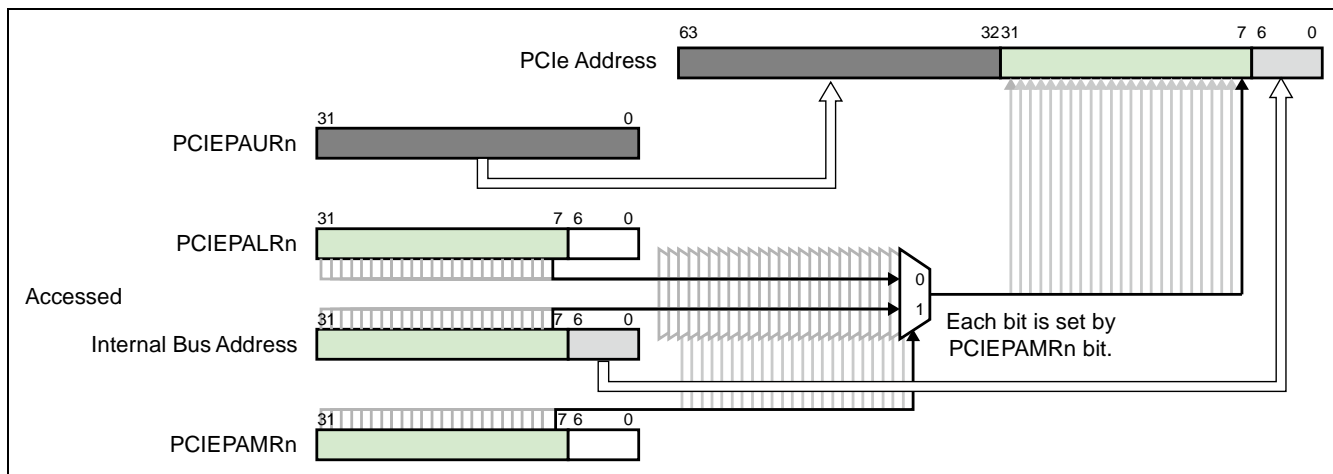


Figure 49.12 Address Translation for PIO Transfer

(3) PIO Transfer Operation

PIO transfer performs memory write/read to the corresponding memory/IO space on the PCI Express from the internal bus side. In reading, a read request is issued to the PCI Express, a completion from the device to be read is converted to data (response), and it is written to the area in the internal bus. Figures 49.13 and 49.14 show write and read operations of PIO transfer, respectively. In the figures, the processes for which “Check the related registers.” is indicated are software-dependent; perform appropriate processing for the application.

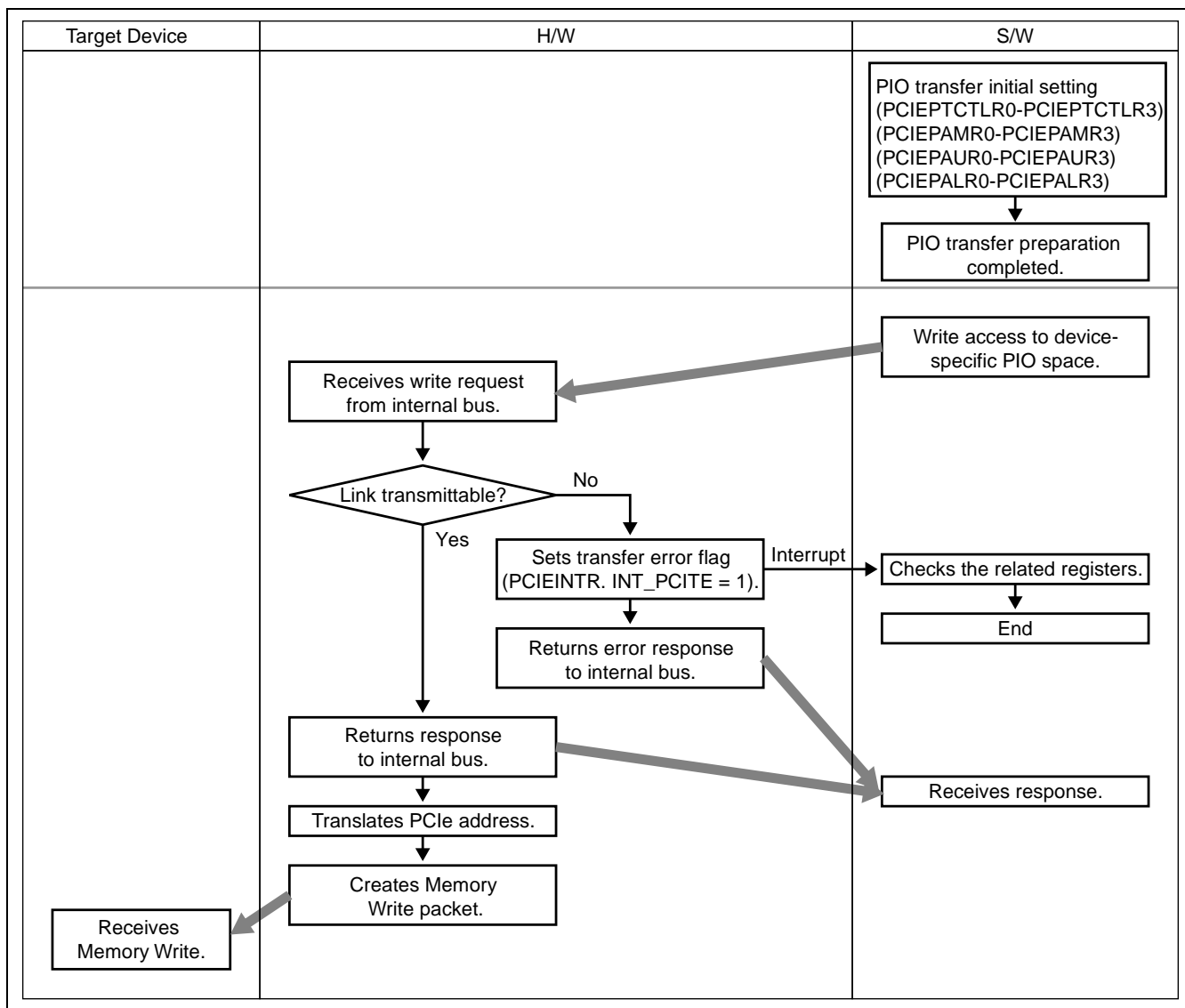


Figure 49.13 PIO Transfer Operation (memory write request)

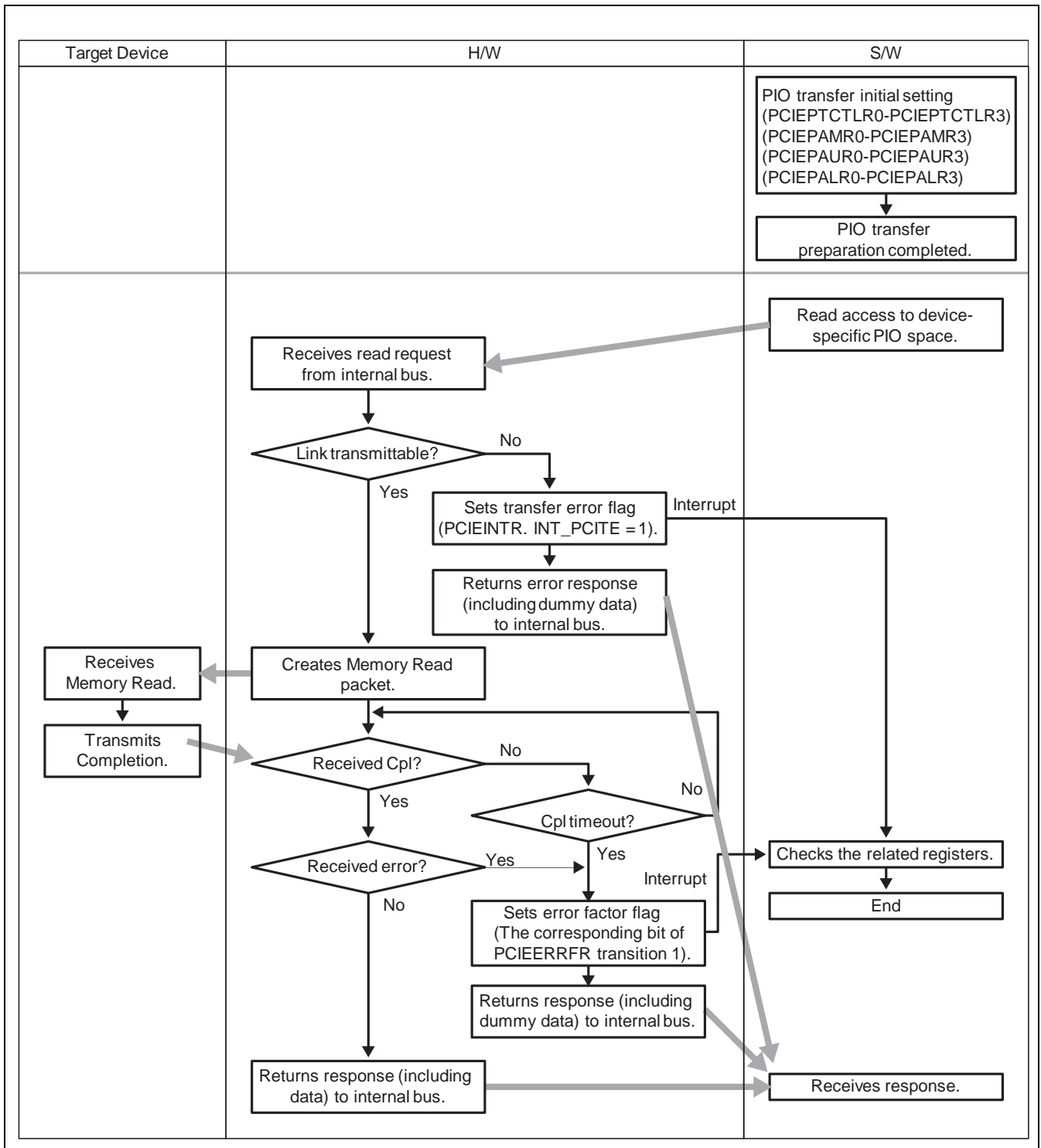


Figure 49.14 PIO Transfer Operation (memory read request)

49.3.6 Target Transfer (Data Transfer from External Device → PCIEC → Internal Bus)

With target transfer, the device on the PCI Express accesses the device on the internal bus of this module through a PCI Express request packet.

(1) Overview of Target Transfer

Using target transfer, the device on the PCI Express can send packets for memory read/write and I/O read/write to read from and write to the area in the internal bus.

In a target transfer, this module can receive packets of any data length less than or equal to the size specified in the MPS (Max Payload Size).

(2) Initial Setting of Target Transfer

The assignment of addresses in the PCI Express space is dynamically determined by the Root Port during a configuration cycle, based on the register settings at initialization. At initialization, the registers are set to specify the parameters such as the sizes of various areas and types of the areas (memory space, I/O space, or the other). When CFINIT is set to 1 and initialization is completed, the initial setting is reflected to BAR_n (Base Address Register n) of the configuration registers. In Root Port mode, PCIEPRAR_n is set through software. Here, BAR indicates the area to be released in response to the request from the downstream. In Endpoint mode, the BAR_n bits of configuration registers are set in the subsequent configuration cycle.

As an area in which memory space is to be allocated, this module supports either 64-bit PCI address space or 32-bit PCI address space (the first 4G area of the 64-bit space). When allocating an area in 32-bit address space, one BAR_n is used; when allocating an area in 64-bit address space, two contiguous BAR_n registers (BAR_{n+1}/BAR_n) are used. For this reason, a maximum of three 64-bit address space areas can be allocated.

In I/O space, areas are allocated using one PCIEPRAR or BAR register.

In Root Port mode, set the internal bus side space to be correlated to each of the PCIEPRAR registers in the PCI Express space to the local (internal bus) address registers 0 to 5 (PCIELAR0 to PCIELAR5). Also set the size to be allocated and the space type to the local (internal bus) address mask registers 0 to 5 (PCIELAMR0 to PCIELAMR5). Set these registers before a link is established (before setting CFINIT to 1). Although PCIEPRAR can be modified after a link is established, turn off the Master Enable of the communication peer or take other appropriate measures to prevent issuance of a request to the corresponding area during PCIEPRAR modification.

In Endpoint mode, set the internal bus side space to be correlated to each of the BAR registers in the PCI Express space to the local (internal bus) address registers 0 to 5 (PCIELAR0 to PCIELAR5). Also set the size to be allocated and the space type to the local (internal bus) address mask registers 0 to 5 (PCIELAMR0 to PCIELAMR5). Set these registers before a link is established (before setting CFINIT to 1).

The target transfer area must not be allocated to the internal bus space for the registers of this module or to the internal bus space for PIO transfer.

The following shows the example of mapping the PCI Express address space to the internal bus space.

Example of mapping the PCI Express address space to the internal bus space (Root Port):

Figure 49.15 shows an example of mapping the PCI Express space to the internal bus space (Root Port).

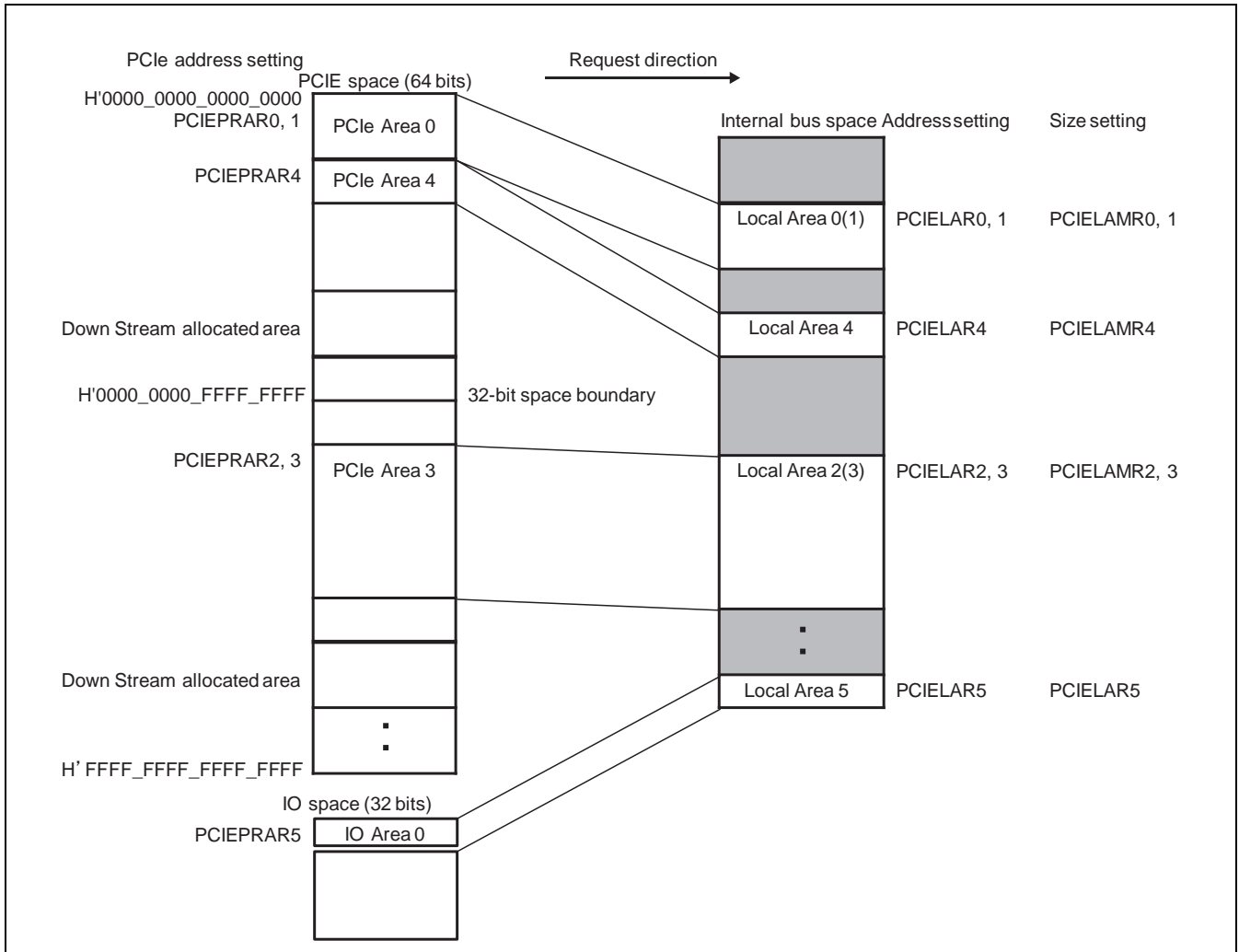


Figure 49.15 Example of Mapping PCI Express Address Space to Internal Bus Space (Root Port)

Example of mapping the PCI Express address space to the internal bus space (Endpoint):

Figure 49.16 shows an example of mapping the PCI Express space to the internal bus space (Endpoint)

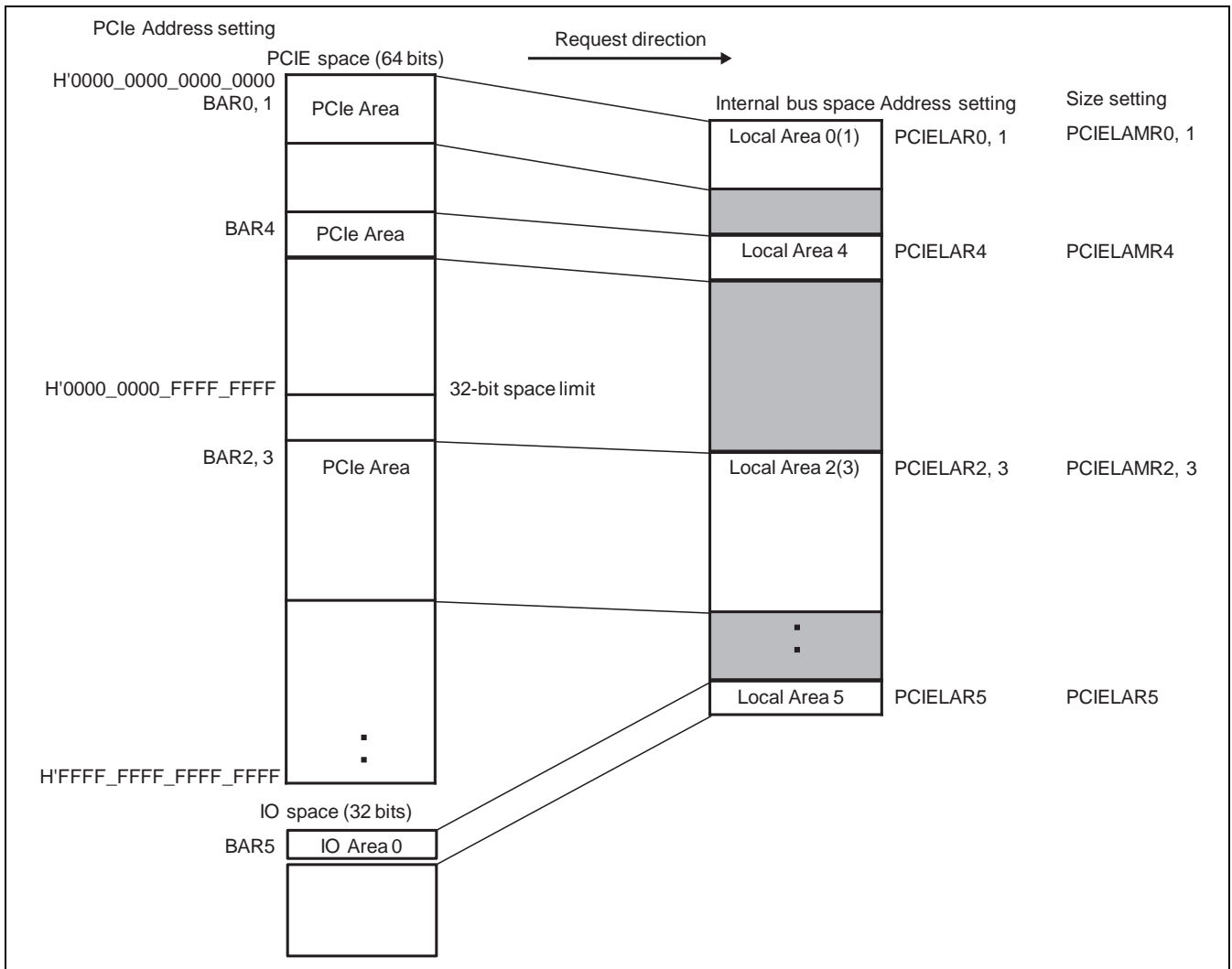


Figure 49.16 Example of Mapping PCI Express Address Space to Internal Bus Space (Endpoint)

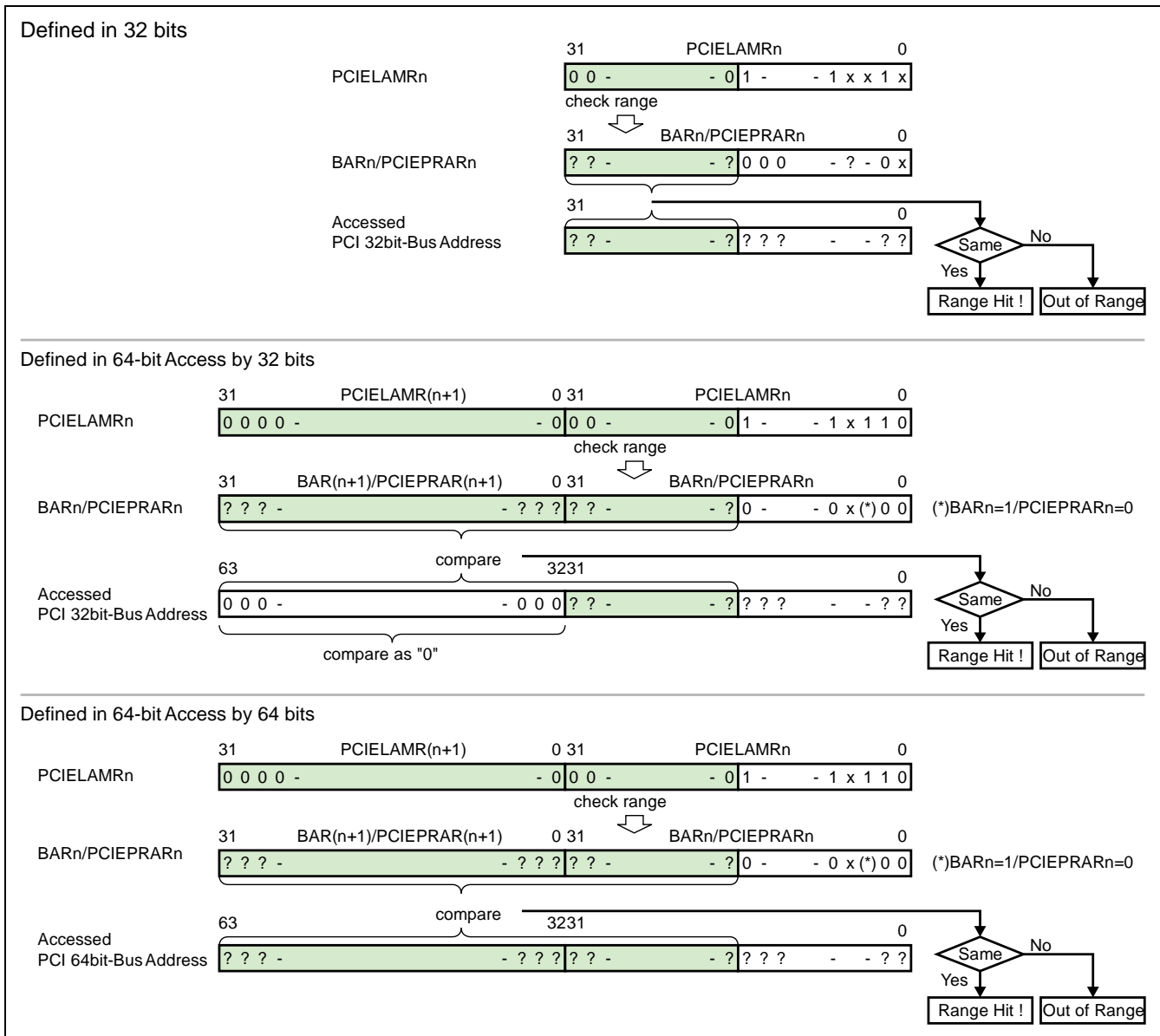


Figure 49.17 Address Comparison for Target Transfer

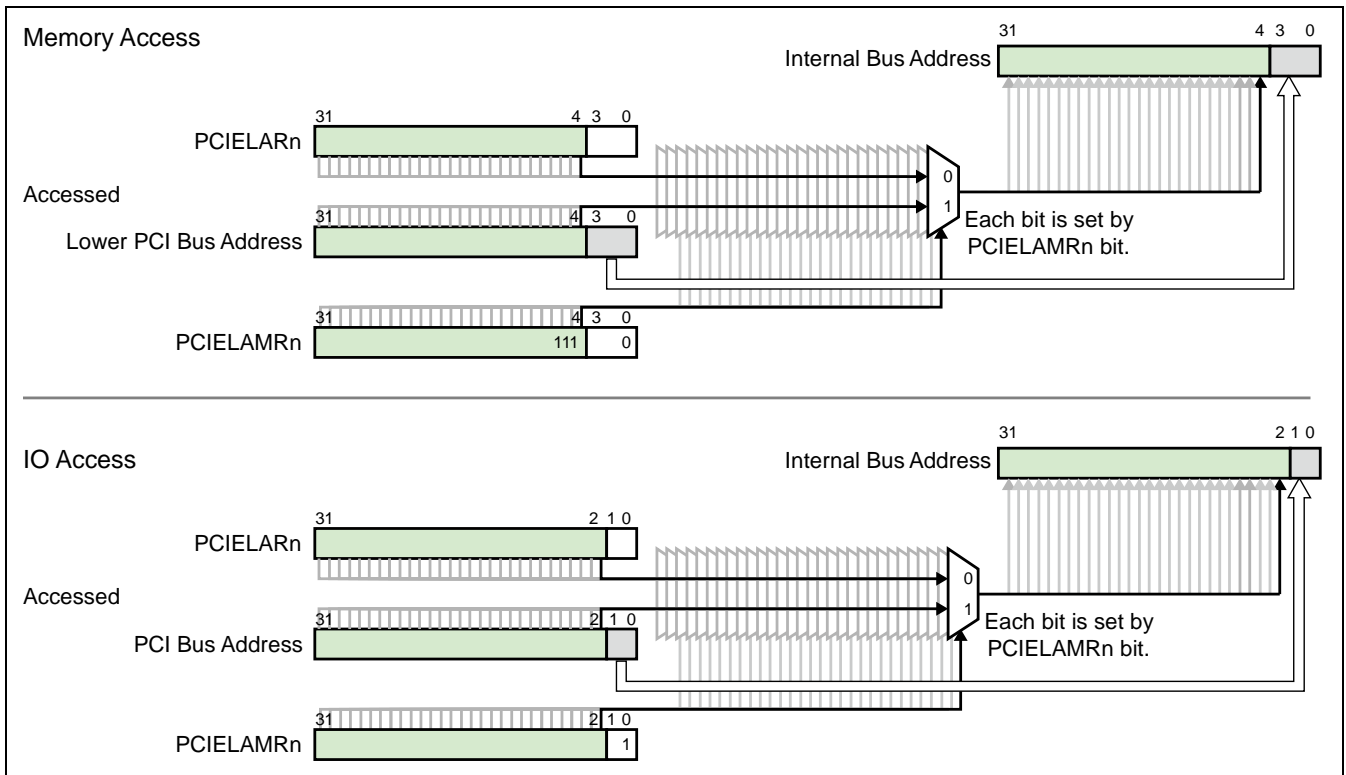


Figure 49.18 Address Translation for Target Transfer

(3) Target Transfer Operation

Figures 49.19 and 49.20 show the target transfer operations to receive a memory write request and the other request, respectively. In receiving a memory write request, the received data is written to the corresponding internal bus address. In receiving the other request, the internal bus is accessed, a completion packet is created and returned (note that when the internal bus is read, all the byte lanes are accessed irrespective of the PCIe space setting (prefetchable or non-prefetchable type) since the internal bus does not have a read strobe).

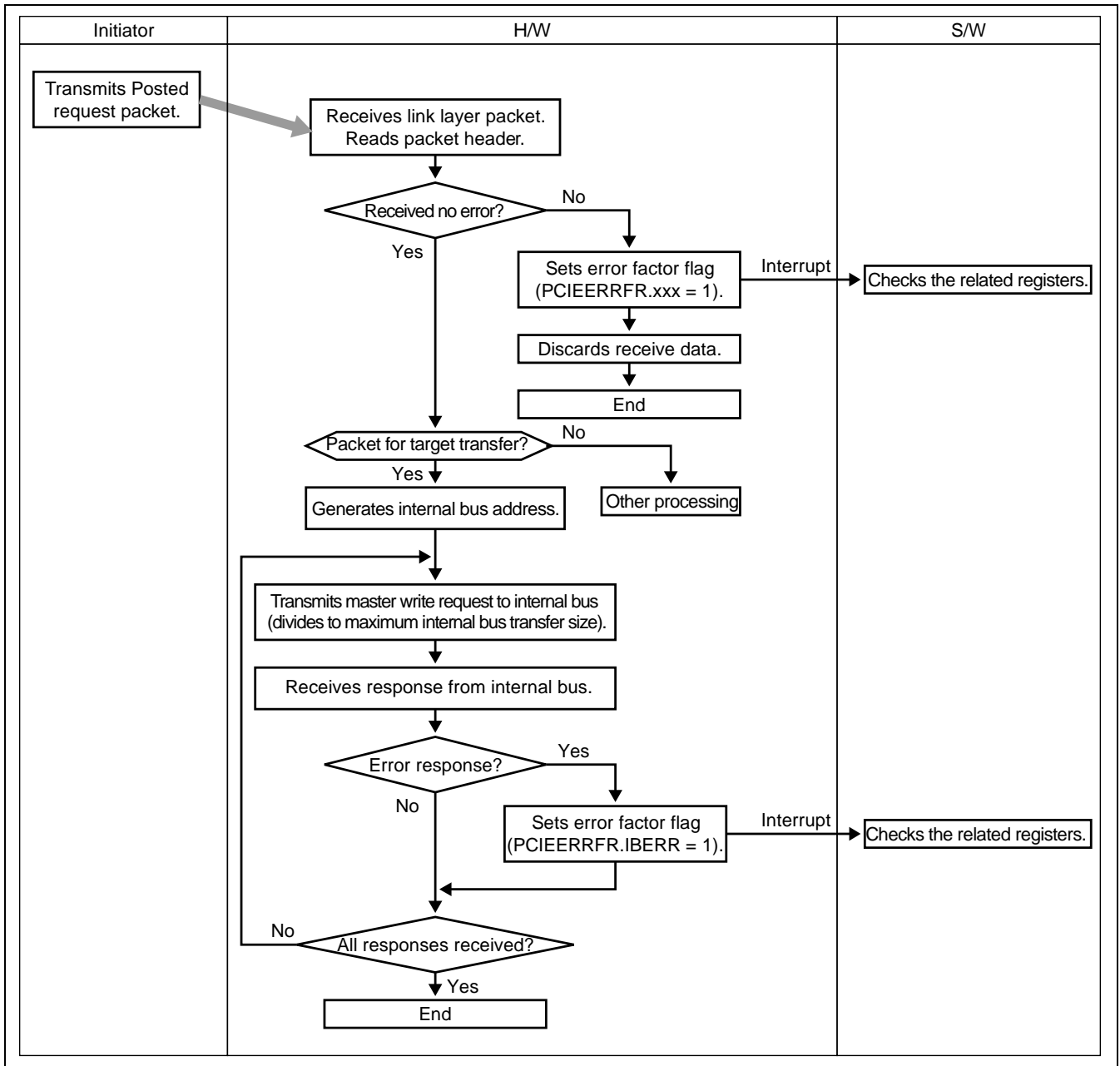


Figure 49.19 Target Transfer (Reception of a Memory Write Request)

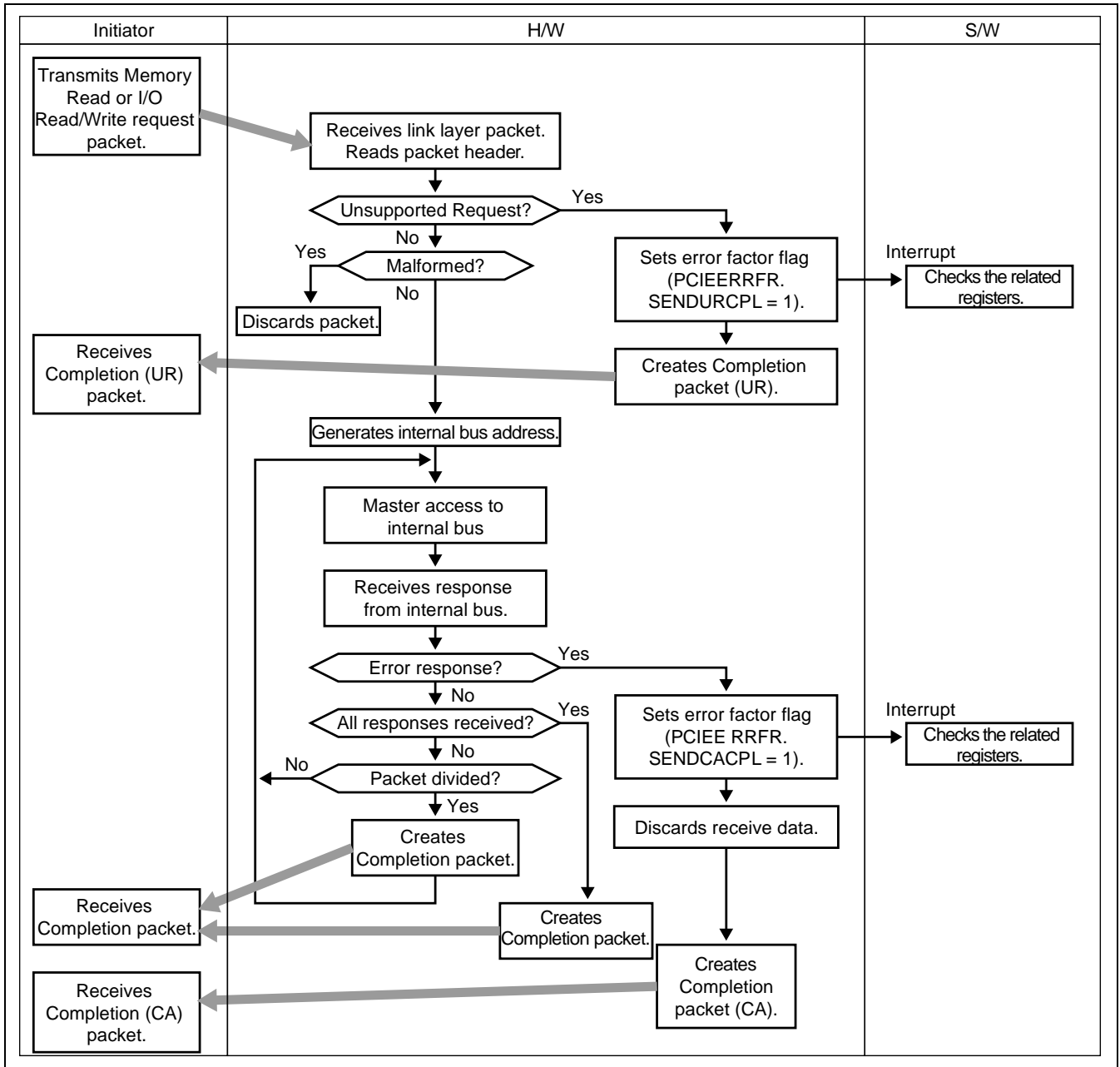


Figure 49.20 Target Transfer (Reception of a Request Other than a Memory Write Request)

49.3.7 DMA Transfer

This section describes the DMA transfer using the DMAC (PCIEC-DMAC) incorporated in this module.

(1) Overview of DMA Transfer

The PCIEC-DMAC enables effective data transfer between the PCI Express and the internal bus area. The PCIEC-DMAC can perform a maximum of 2^{29} count of transfer without intervention of the CPU.

The PCIEC-DMAC supports the command chain function, which allows consecutive execution of multiple transfer commands. With the command chain function, the DMAC-related setting information, such as the addresses of the transfer destination and source and transfer size, is considered as a command; by reading and executing these commands from the memory one by one, consecutive execution of multiple transfer units is provided without intervention of the CPU.

The PCIEC-DMAC has the following features.

- Number of channels: 8 channels
- Address space: PCI Express = 32/64 bits, internal bus = 32 bits
- Transfer data length: PCI Express = 8 to 128 bytes, internal bus = 8 to 128 bytes
- Maximum transfer count: 536,870,912 (2^{29})
- Address mode: Dual address mode
- Transfer request: Started by register control
- Data transfer: Normal mode (continuous transfer), command chain
- Priority: Either channel priority fixed mode or round-robin mode selectable
- Interrupt request: Interrupts can be requested to the INTC when a data transfer has completed or an error has occurred.

(2) Initial Setting of DMA Transfer

Set the PCI Express and internal bus addresses, byte count, transfer end interrupt. When the command chain is not used, set the PCIEDMCHCRn.CCE to 0. In the DMA transfer (internal bus side), access to the internal bus space for the registers and access to the internal bus space for PIO transfer are prohibited.

(3) DMA Transfer Operation

DMA transfer can be initiated by simultaneously setting the transfer direction and enabling the channels with PCIEDMCHCRn. Transfer end is detected by confirming PCIEDMCHSRn[0].TE = 1 or detecting a transfer end interrupt. After transfer is ended, transfer is completed by setting PCIEDMCHCRn[31].CHE to 0. Simultaneously, PCIEDMCHSRn[0].TE is set to 1 to clear this bit.

Figure 49.21 and 49.22 show DMA transfer from the internal bus to the PCI Express space and from the PCI Express space to the internal bus, respectively.

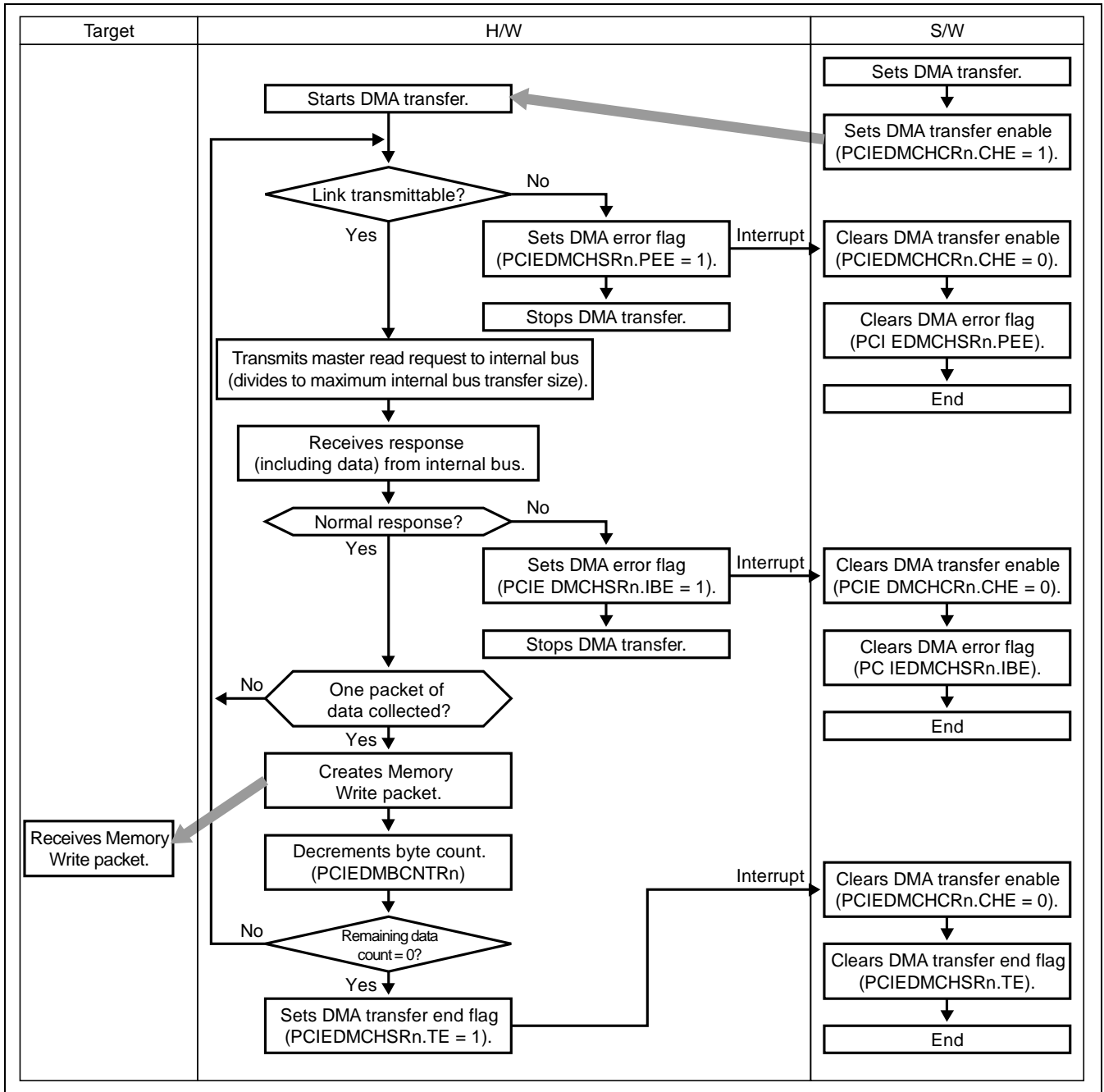


Figure 49.21 DMA Transfer (Internal Bus → PCI Express)

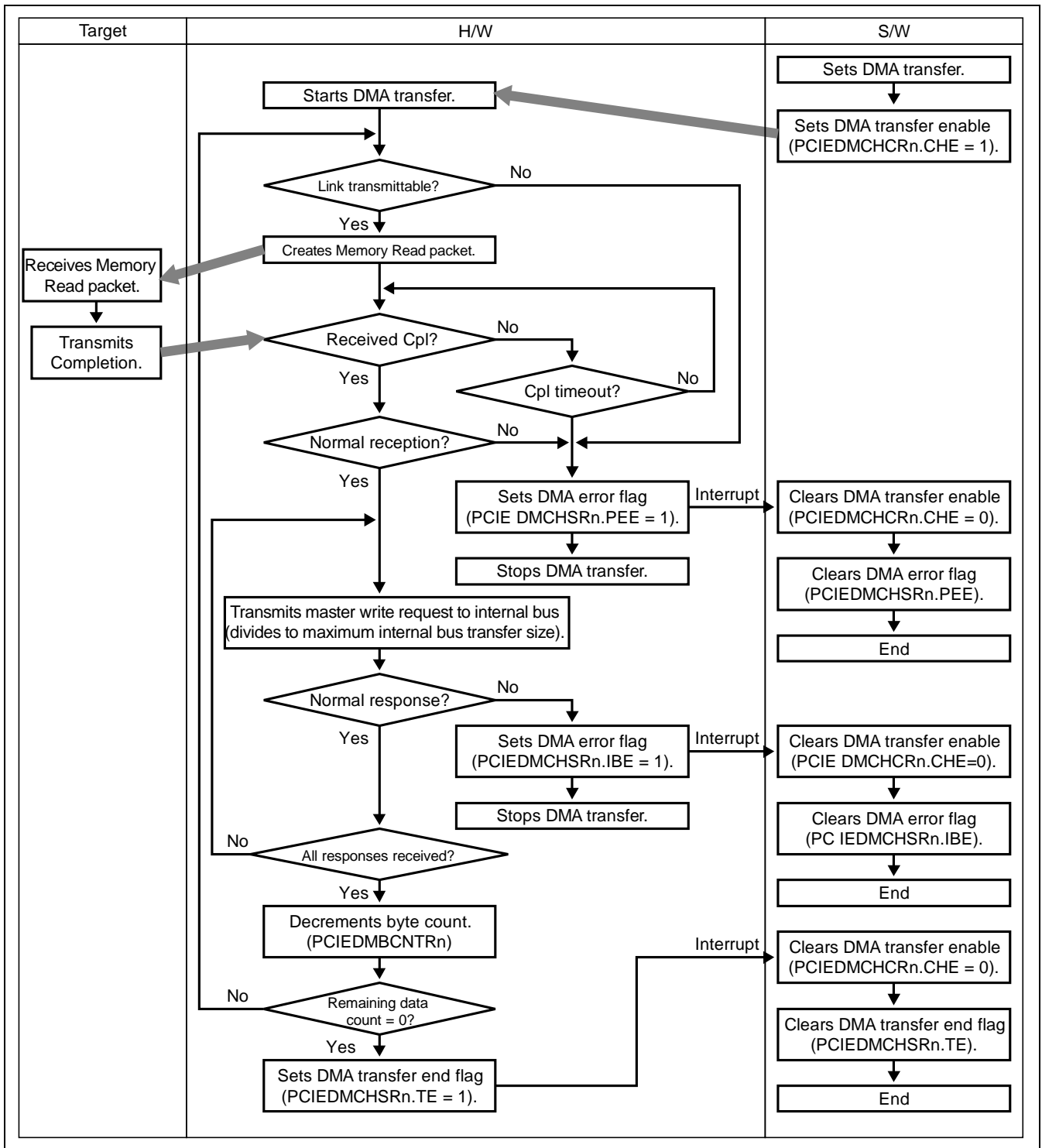


Figure 49.22 DMA Transfer (PCI Express → Internal Bus)

(4) DMA Channel Priority

When more than one DMA channel is used simultaneously, they are actually used according to the priority described in this section. The priority can be set with the PCIEDMAOR register. Set the relevant bit at the same time as setting PCIEDMAOR.DMAE to 1.

The channel priority is initialized to CH0 > CH1 > CH2 > CH3 > CH4 > CH5 > CH6 > CH7 while PCIEDMAOR.DMAE = 0.

(a) Arbitration Timing between DMA Channels

Arbitration is performed every transfer unit (128 bytes max.) of the DMA channel currently transferring data.

(b) Fixed Mode

When PCIEDMAOR.ABT is 0, the priority is fixed to the default setting.

(c) Round Robin Mode

When PCIEDMAOR.ABT is 1, round robin scheduling is applied to all the channels to determine the priority. When a certain channel completes transfer, the channel is given a bottom priority and the channels that have been given a lower priority than that channel are given a higher priority as shown in Figure 49.23.

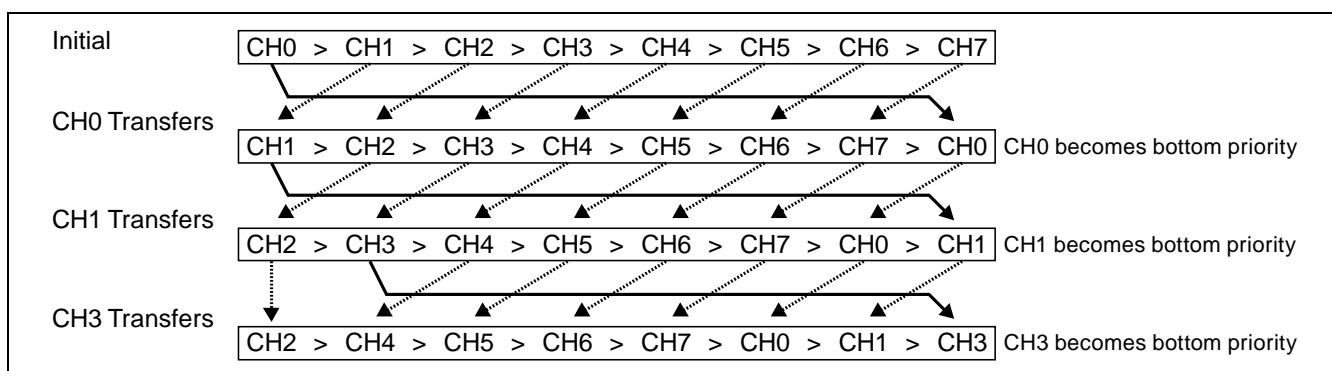


Figure 49.23 Round Robin Scheduling among All the Channels

(5) Dividing Packet at Address Boundary

If the PCI Express side address or internal bus address of a packet crosses the 4-Kbyte boundary, the packet is divided.

(6) Command Chain

(a) Overview of Command Chain Operation

A command chain allows this module to continuously execute multiple DMAC commands. A DMAC command refers to a set of information related to the PCIEC-DMAC transfer; that is, the information specified by PCIEDMPALRn, PCIEDMPAURn, PCIEDMIARn, PCIEDMBCNTRn, PCIEDMCCARn, and PCIEDMCHCRn. As well as in the PCIEC-DMA control register, the information can also be set in the memory in the format shown in Figure 49.24. With the command chain, each time a DMAC command execution is completed, the next DMAC command is read from memory and the content of the DMAC command is written to the relevant PCIEC-DMA control register, which allows execution of the DMAC command.

By specifying the next DMAC command in the DMAC command to be read, a command chain is created and thus data can be transferred consecutively.

During command chain operation, a DMAC command is read from the memory address indicated by PCIEDMCCARn and the content of the read command is written to the register of the relevant channel to execute the command. If the CCE bit in the read DMAC command is 1, the next command is read from memory again and executed upon completion of the current command. If the CCE bit in the read DMAC command is 0, completion of the command completes the command chain operation.

(b) Initial Setting of Command Chain

To start a command chain on a channel, enable the channel while PCIEDMCHCRn.CCE is set to 1. Before starting a command chain, store a chain of DMAC commands in a memory location accessible from the internal bus, and set the address of the first DMAC command in PCIEDMCCARn.

Note: As the setting (information) of the first transfer of the command chain, the information in the memory specified with PCIEDMCCARn is used; the setting in the PCI Express-DMA control registers is not used (here, any setting other than PCIEDMCCARn is ignored).

(c) DMAC Command Format

Figure 49.24 shows the format of DMAC commands to be stored in memory. Set each field of the commands as described below and store such commands.

- CHE field: Always set 1.
- CCE field: Set 0 for the command to be executed last; set 1 for the other commands.
- CCA field: Set 0 in all the bits for the command to be executed last; set the address of the next command for the other commands.
- Reserved field: Always set 0.

| Offset | Registers | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|-----------|-------|------|-----|----|----|----|----|------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 00 | DMPALR | PADRL | | | | | | | | | | | | | | | | R | | | | | | | | | | | | | | | |
| 04 | DMPAUR | PADRU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 08 | DMIAR | IADR | | | | | | | | | | | | | | | | R | | | | | | | | | | | | | | | |
| 0C | DMBCNTR | R | BCNT | | | | | | | | | | | | | | | R | | | | | | | | | | | | | | | |
| 10 | - | R | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 14 | - | R | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 18 | DMCCAR | CCA | | | | | | | | | | | | | | | | R | | | | | | | | | | | | | | | |
| 1C | DMCHCR | CHE | DIR | CCE | R | | | | ATTR | R | | | | TC | R | | | | | | | | | | | | | | | | | | |

R = Reserved

Figure 49.24 Formats of DMAC Commands

(d) Channel Priority for Command Read

While the command chain operation is in progress on multiple DMA channels, DMAC command read requests are issued from the multiple DMA channels; here, round robin scheduling is applied to all the channels to determine the priority (refer to section 49.3.7 (4)(c), Round Robin Mode).

Note: The channel priority here is different from the priority of the DMA transfer operation specified with PCIEDMAOR.

The channel priority is initialized to CH0 > CH1 > CH2 > CH3 > CH4 > CH5 > CH6 > CH7 while PCIEDMAOR.DMAE = 0.

(7) Notes on DMA Transfer

(a) Handling upon DL_Down / DL_Active

If a DL_Down occurs during DMA transfer, it is handled as a transfer error on the PCI Express side and thus transfer is ended after PCIEDMCHSRn.PEE is set. However, if a request is being made to the internal bus when a DL_Down occurs, a response to the request is first received and then flag is set and transfer is ended. Therefore, not all the operations are stopped immediately after a DL_Down.

When returned from a DL_Down to DL_Active, first read PCIEDMAOR.DMAACT = 0 to confirm that all the channels that have been operating before DL_Down have stopped operating, and then re-set DMA and perform transfer again.

49.3.8 Transfer Priority

This section describes the transfer priority.

(1) Arbitration (Transmission to PCI Express)

Accesses to transmit packets to the PCI Express include: DMA transfer, response processing to target transfer, PIO transfer, and Message transfer. The DMAC and the modules controlling PIO transfer output posted and non-posted requests, respectively.

Arbitration between these accesses is performed according to the priority shown below.

(High priority)

- Completion of target transfer (configuration)
- Completion of target transfer (memory, IO)
- PIO transfer (configuration, memory read, IO)
- PIO transfer (memory write)
- Various messages
- DMA transfer (memory read)
- DMA transfer (memory write)

(Low priority)

(2) Arbitration (Reception from PCI Express)

The completion receiver and request receiver receive packets from the PCI Express and make reception access requests.

Arbitration between these accesses is performed according to the priority shown below.

(High priority)

- PCIEC completion receiver
- PCIEC request receiver

(Low priority)

The PCIEC link of this module, which processes the PCIe protocol, processes packets received from the PCI Express one by one in order of reception. After that, the processed packets are sequentially distributed to the target receiver, DMA receiver, message receiver, and PIO receiver according to the type of the received packet and then processed one by one there. If the receiver corresponding to the processing target is waiting for a certain process to be completed (for example, waiting for a request to the internal bus or a response), the PCIe link starts the next reception process after completion of the process. Until completion, the next packet is suspended from being processed.

(3) PCI Express Transmit/Receive Ordering

The following describes the transaction ordering rules of this module.

Table 49.12 shows the transmit transaction ordering rules of this module.

Table 49.12 Transmit Transaction Ordering Rules of This Module

| Row | Pass Column | 1st Posted | 1st Non-Posted | 1st Completion |
|----------------|-------------|------------|----------------|----------------|
| 2nd posted | | No | Yes | Yes |
| 2nd non-posted | | | No | Yes |
| 2nd completion | | | Yes | No |

Note: Yes: The second transaction can pass the first transaction.

No: The second transaction cannot pass the first transaction.

Table 49.13 shows the receive transaction ordering rules of this module.

Table 49.13 Receive Transaction Ordering Rules of This Module

| Row | Pass Column | | 1st Transaction | | | | | Completion | Memory Read (DMA) |
|-----------------|-------------------|--------------------------|-----------------|------------|--------------------------|------------|-----|------------|-------------------|
| | | | Request | | | | | | |
| | | | Posted | Non-Posted | | | | | |
| | | Memory Write/Message | Memory/IO Read | IO Write | Configuration Write/Read | Read/Write | | | |
| 2nd transaction | Posted | Memory write/message | No* | Yes | Yes | Yes | Yes | Yes | |
| | Non-posted | Memory/IO read | No | Yes | No | Yes | Yes | Yes | |
| | | IO write request | No* | Yes | No | Yes | Yes | Yes | |
| | | Configuration write/read | No | Yes | Yes | No | Yes | Yes | |
| Completion | Read/write | Yes | Yes | Yes | Yes | No | Yes | | |
| | Memory read (DMA) | Yes | Yes | Yes | Yes | Yes | No | | |

Notes: Yes: The second transaction can pass the first transaction.

No: The second transaction cannot pass the first transaction.

* For Memory/IO Write, the second request cannot pass the first request if they are issued to the same BAR space; the second request may pass the first request if issued to the different BAR spaces. For a combination of Memory Write and Message, the second request may pass the first request.

This module applies the ordering rules conforming to the standard to the transmission and reception sides. According to the rules, some packets cannot be passed by (to prevent an undesired influence on the order of sequential processing and such). Therefore, depending on the packet transfer order or combination, processing of a packet may be delayed until the previous packet processing is completed.

(4) Arbitration (Bus Master Request)

Accesses to output a master write request to the internal bus include: DMA transfer, target transfer, and message transfer.

Arbitration between these accesses is performed according to the priority shown below.

(High priority)

- Target transfer (reception of memory write and IO write)
- DMA transfer (reception of completion for memory read)

(Low priority)

Accesses to output a master read request to the internal bus include: DMA transfer, target transfer, and message transfer.

Arbitration between these accesses is performed according to the priority shown below.

(High priority)

- Target transfer (reception of memory read and IO read)
- DMA (command read)
- DMA (transmission of memory write)

(Low priority)

49.3.9 Link Function

(1) Changing Speed

The PCI Express 2.0 standard allows changing of the transfer speed intentionally or to handle the problem related to the link reliability. Figure 49.25 shows a flow of changing the transfer speed to 5.0 GT/s.

- Allow the factor of speed change to be generated while speed change is enabled. (According to the PCI Express 2.0 standard, if speed change is attempted by itself and ends in failure, starting of speed change by itself is prohibited for 200 ms.)
- If speed change is attempted while another speed change is in progress, it is impossible to know which speed change factor is used for judging the end of speed change. For reliable judgment of the end of speed change, do not make any attempt to change a speed during another speed change. (MACCTLR.SPCHG == 0 and MACSR.SPCHGFIN == 0 indicate that speed change has not been set or speed change processing is not in progress.)

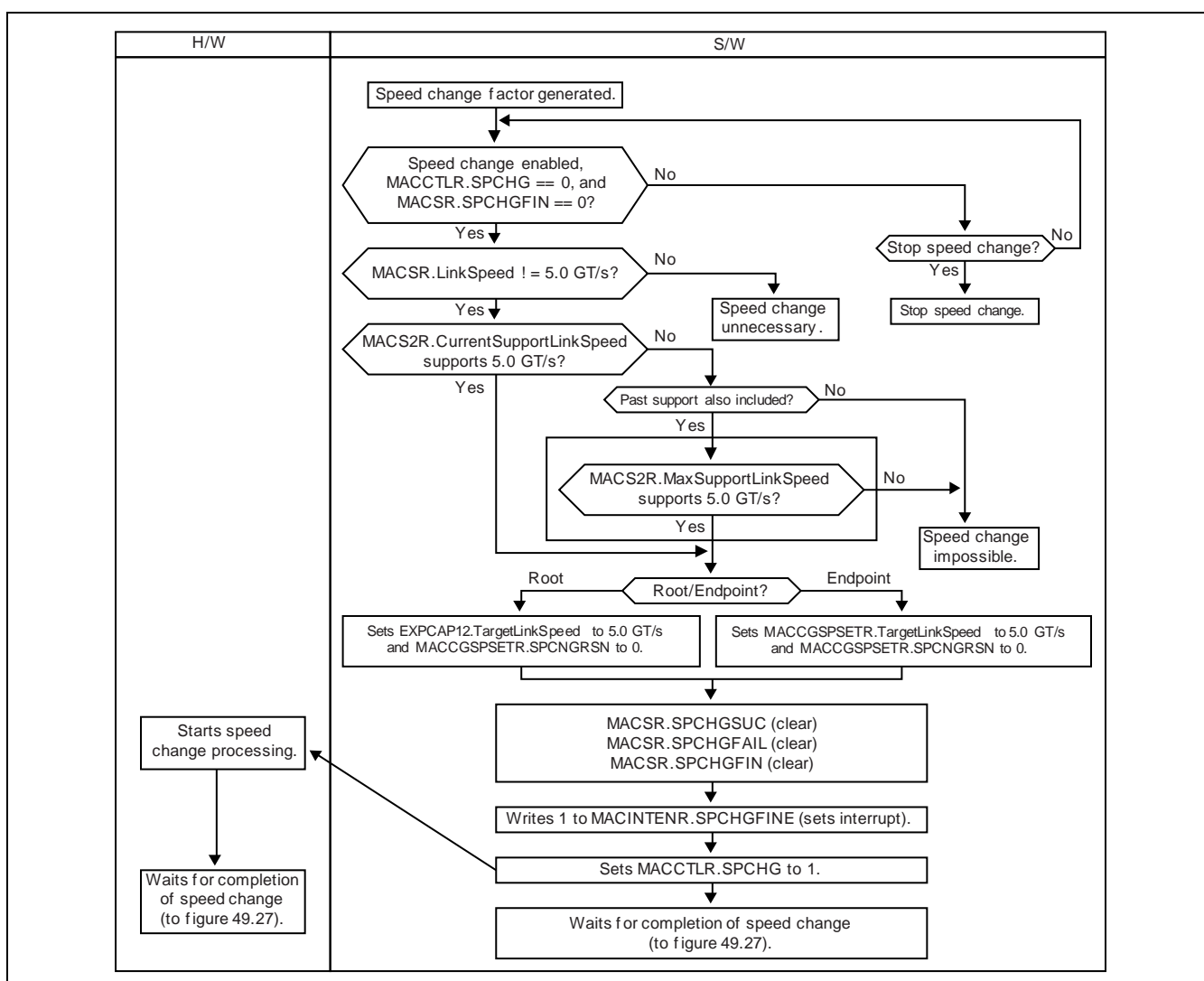


Figure 49.25 Changing Speed to 5.0 GT/s

Figure 49.26 shows a flow of changing the transfer speed to 2.5 GT/s.

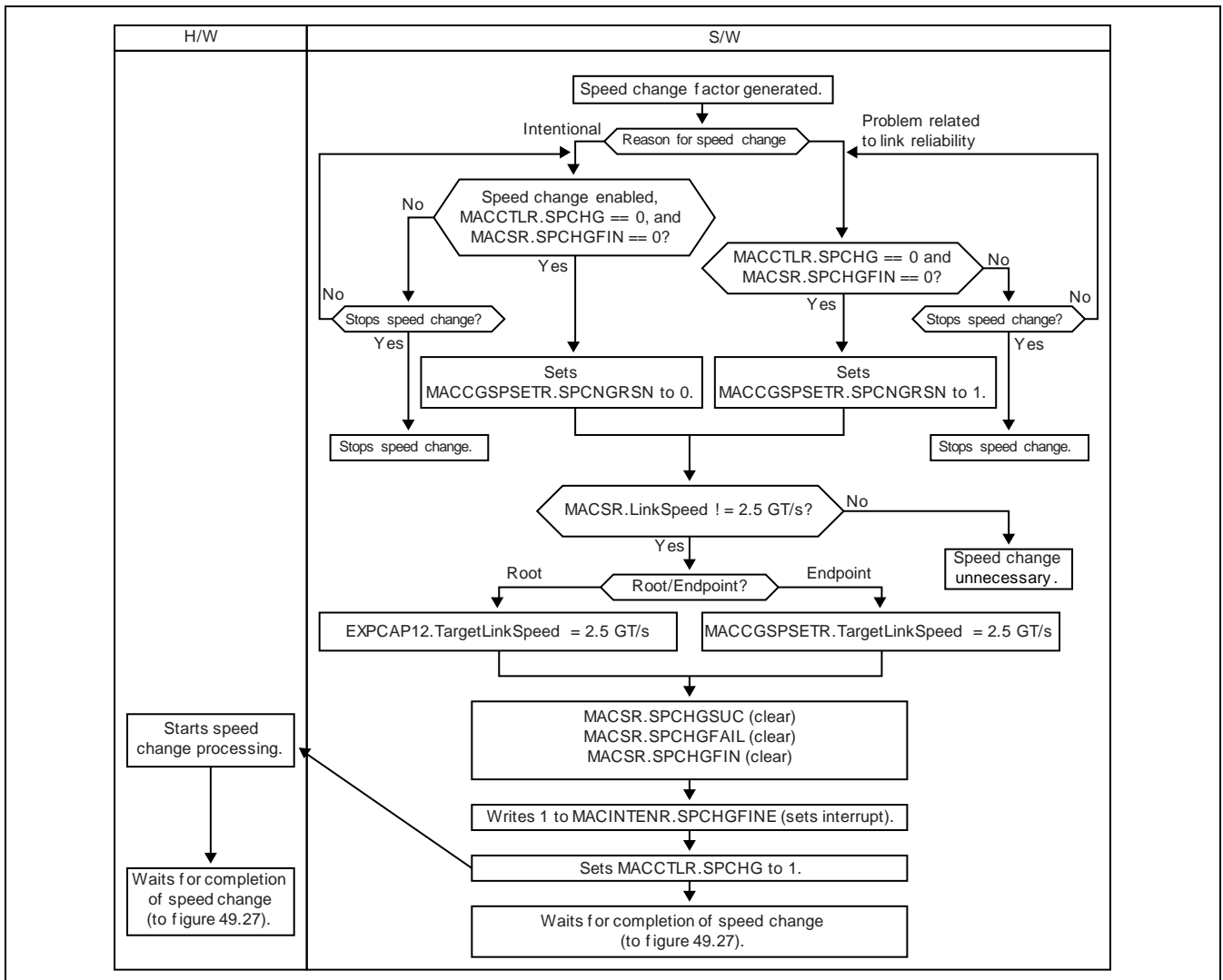


Figure 49.26 Changing Speed to 2.5 GT/s

Figure 49.27 shows a flow of the ending process for the transfer speed change.

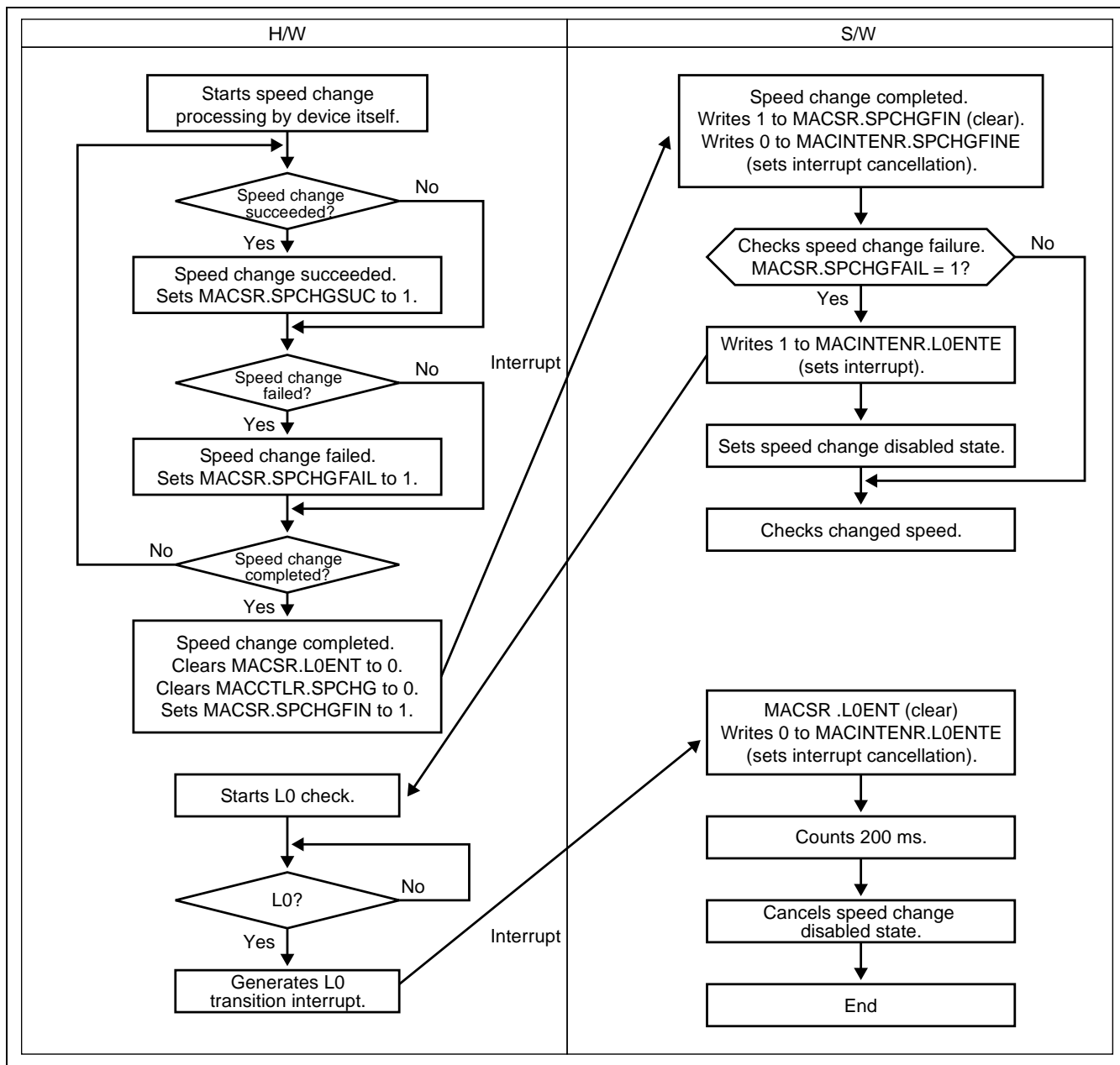


Figure 49.27 Ending Process of Speed Change

49.3.10 Power Management

This section describes power management of this module. This module has two power management modes below.

- PCI-PM
- ASPM (L0s)

PCI-PM is PCI-compatible and ASPM is specific to PCI Express. ASPM mode supports L0s only.

(1) PCI-PM

This section describes PCI-compatible power management mode. Table 49.14 shows power management operations in PCI-PM mode.

Table 49.14 PCI-PM Operations

| No. | Transition | Target Device |
|-----|--------------|---------------|
| 1 | L0 → L1 → L0 | Endpoint |
| 2 | | Root port |

(a) L0 → L1 → L0 (Endpoint)

After modification of the Power State field, initiate the L1 transition sequence through software according to the flowchart shown in Figure 49.28.

After initiated through software, the L1 transition sequence is performed through hardware according to the following procedure.

1. Prepares for L1 transition.
 2. Processes the handshake with the peer device for L1 transition.
 3. Causes the LTSSM of this module to make an L1 transition.
- Triggered by a write of PMCTLR.L1IATN = 1, hardware initiates the L1 transition sequence. As preparation for causing L1 transition, this module first allows hardware to automatically perform the following. (1) Waits for the minimum amount of credits to be accumulated necessary to transmit the largest TLP for which all the types of credits are configured. (2) Waits for the retry buffer to be empty. (3) Waits for all the TLP to be transmitted.
 - After completion of the above preparation, this module transmits PM_Enter_L1 DLLP and receives PM_Request_Ack DLLP to process the handshake with the peer device for L1 transition.
 - After completion of the handshake for L1 transition, this module disables TLP/DLLP transmission and directs the LTSSM to make a transition to L1. If directions are given to make a transition to the detect or recovery state before the LTSSM completes L1 transition, this module gives priority to a transition to the detect or recovery state.
 - Hardware clears the PMCTLR.L1IATN bit to 0 when the LTSSM makes a transition to the state other than L0, recovery, and configuration, that is, when the LTSSM makes a transition to L1 (PMSR.PMSTATE = L1) properly or LDn (PMSR.PMSTATE = LDn). Simultaneously, the PMSR.L1FAEG bit is set. Software can confirm that TLP write is again enabled and the transition destination by reading PMSR.L1FAEG = 1 and PMSR.PMSTATE. After confirming PMSR.L1FAEG = 1, write 1 to PMSR.L1FAEG to clear the bit.
 - After initiation of the L1 transition sequence, if transitions are made as L0 → recovery → L0 or L0 → recovery → configuration → L0 before a transition to the L1 state is caused, hardware automatically initiates the L1 transition sequence again. Therefore, software has nothing to process. PMSR.L1FAEG is not set.

The sequence for a recovery from L1 is performed under the following conditions.

1. Cases in which recovery is initiated by this module itself (caused by software)

Software gives directions to transmit the PM_PME message to prepare for communication again; or an error message is transmitted through software processing.

The message of the error caused by the PM_PME message is scheduled for transmission by software when the PM_PME message transmission is supported. Be sure to confirm that a transition to L1 has been properly made (PMSR.L1FAEG = 1 and PMSR.PMSTATE = L1) before scheduling.

2. Cases in which recovery is initiated by the peer

The receive lane becomes no longer electrically idle.

Typical example: The peer issues D0 configuration write for recovery and makes a transition to recovery itself.

3. Cases in which recovery is initiated by this module itself (caused by hardware)

An error message is transmitted through hardware processing.

Depending on the timing, an error message is transmitted after L1 transition and a recovery may be made.

A completion of the TLP received before L1 transition is transmitted.

After a recovery to L0, if the device is in the non-D0 state and there is no TLP to be transmitted, software should confirm that hardware is in the L0 state (PMSR.PMSTATE = L0) and initiate the L1 transition sequence again (write 1 to PMCTLR.L1IATN). In this case, the sequence is: L0 → L1 → L0 recovery → L1 again.

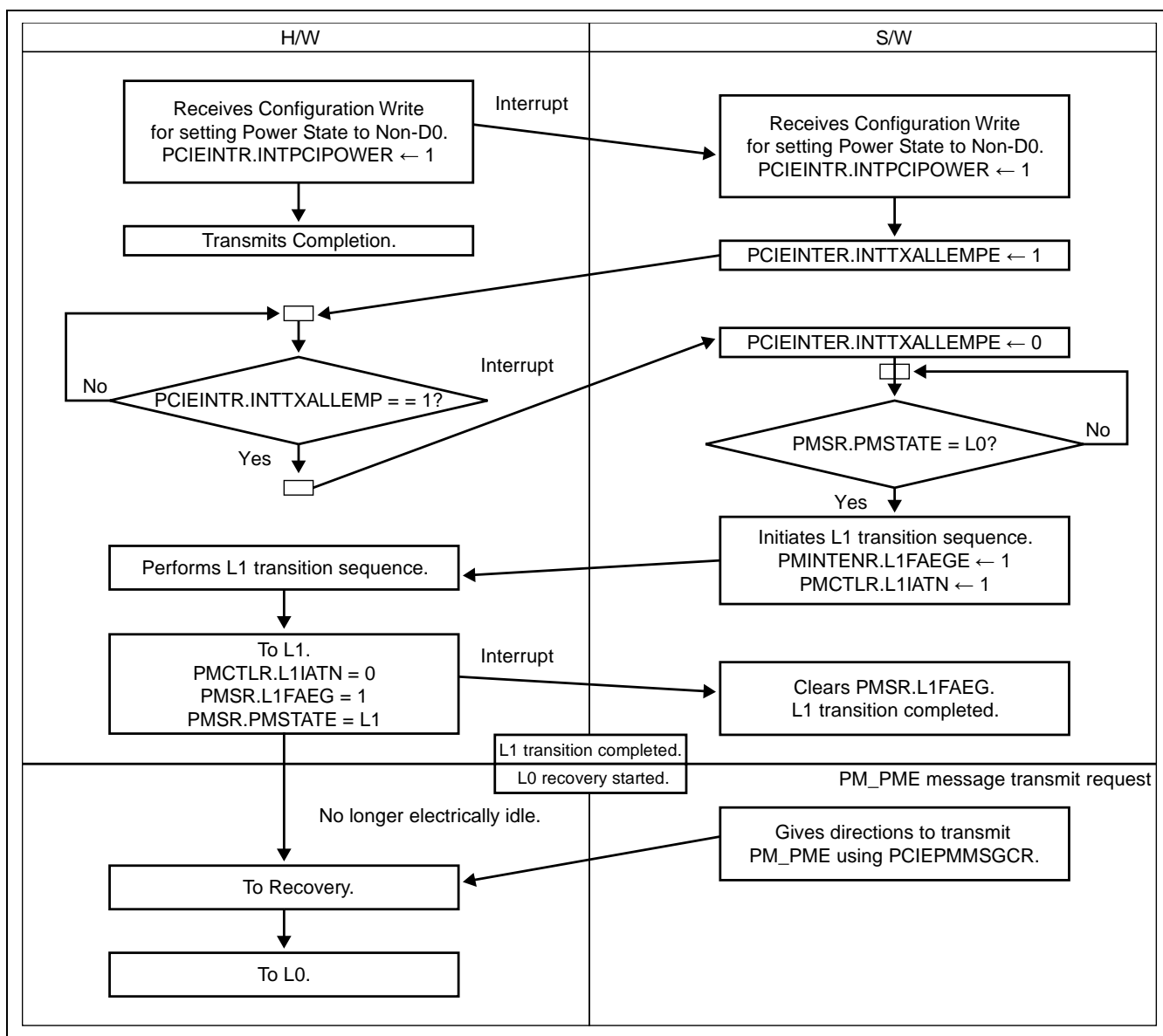


Figure 49.28 Transition from L0 to L1 to L0 (Endpoint)

(b) L0 → L1 → L0 (Root Port)

Initiate the L1 transition sequence through software according to the flowchart shown in Figure 49.29.

After initiated through software, the L1 transition sequence is performed through hardware according to the following procedure.

1. Prepares for L1 transition.
 2. Processes the handshake with the peer device for L1 transition.
 3. Causes the LTSSM of this module to make an L1 transition.
- Triggered by a write of PMCTLR.L1IATN = 1, hardware initiates the L1 transition sequence. As preparation for causing L1 transition, this module first allows hardware to automatically perform the following. (1) Waits for the retry buffer to be empty. (2) Waits for all the TLP to be transmitted.
 - After completion of the above preparation, this module transmits PM_Request_Ack DLLP and receives the electrical idle ordered set to process the handshake for L1 transition.
 - When having received the electrical idle ordered set on any lane, this module disables TLP/DLLP transmission. If directions are given to make a transition to the Detect or Recovery state before disabling TLP/DLLP transmission, this module gives priority to a transition to the Detect or Recovery state.
 - After completion of the handshake for L1 transition, this module directs the LTSSM to make a transition to L1 and transmits the electrical idle ordered set.
 - Hardware clears the PMCTLR.L1IATN bit to 0 when the LTSSM makes a transition to the state other than L0, Recovery, and Configuration, that is, when the LTSSM makes a transition to L1 (PMSR.PMSTATE = L1) properly or LDn (PMSR.PMSTATE = LDn). Simultaneously, the PMSR.L1FAEG bit is set. Software can confirm that TLP write is again enabled and the transition destination by reading PMSR.L1FAEG = 1 and PMSR.PMSTATE. After confirming PMSR.L1FAEG = 1, write 1 to PMSR.L1FAEG to clear the bit.
 - After initiation of the L1 transition sequence, if transitions are made as L0 → Recovery → L0 or L0 → Recovery → Configuration → L0 before a transition to the L1 state is caused, hardware automatically initiates the L1 transition sequence again. Therefore, software has nothing to process. PMSR.L1FAEG is not set.

The sequence for a recovery from L1 is performed under the following conditions.

1. Cases in which recovery is initiated by this module itself

Software performs configuration write to transmit TLP again.

The request for configuration write is scheduled for transmission by software. Be sure to confirm that a transition to L1 has been properly made (PMSR.L1FAEG = 1 and PMSR.PMSTATE = L1) before scheduling.
2. Cases in which recovery is initiated by the peer

The receive lane becomes no longer electrically idle.

Typical example: The Endpoint issues a PME message for requesting a recovery itself and the peer makes a transition to Recovery.

After a recovery to L0, if the device is in the Non-D0 state and PM_Enter_L1 DLLP is transmitted from the downstream device, software should confirm that hardware is in the L0 state (PMSR.PMSTATE = L0) and initiate the L1 transition sequence again (write 1 to PMCTLR.L1IATN). In this case, the sequence is: L0 → L1 → L0 recovery → L1 again.

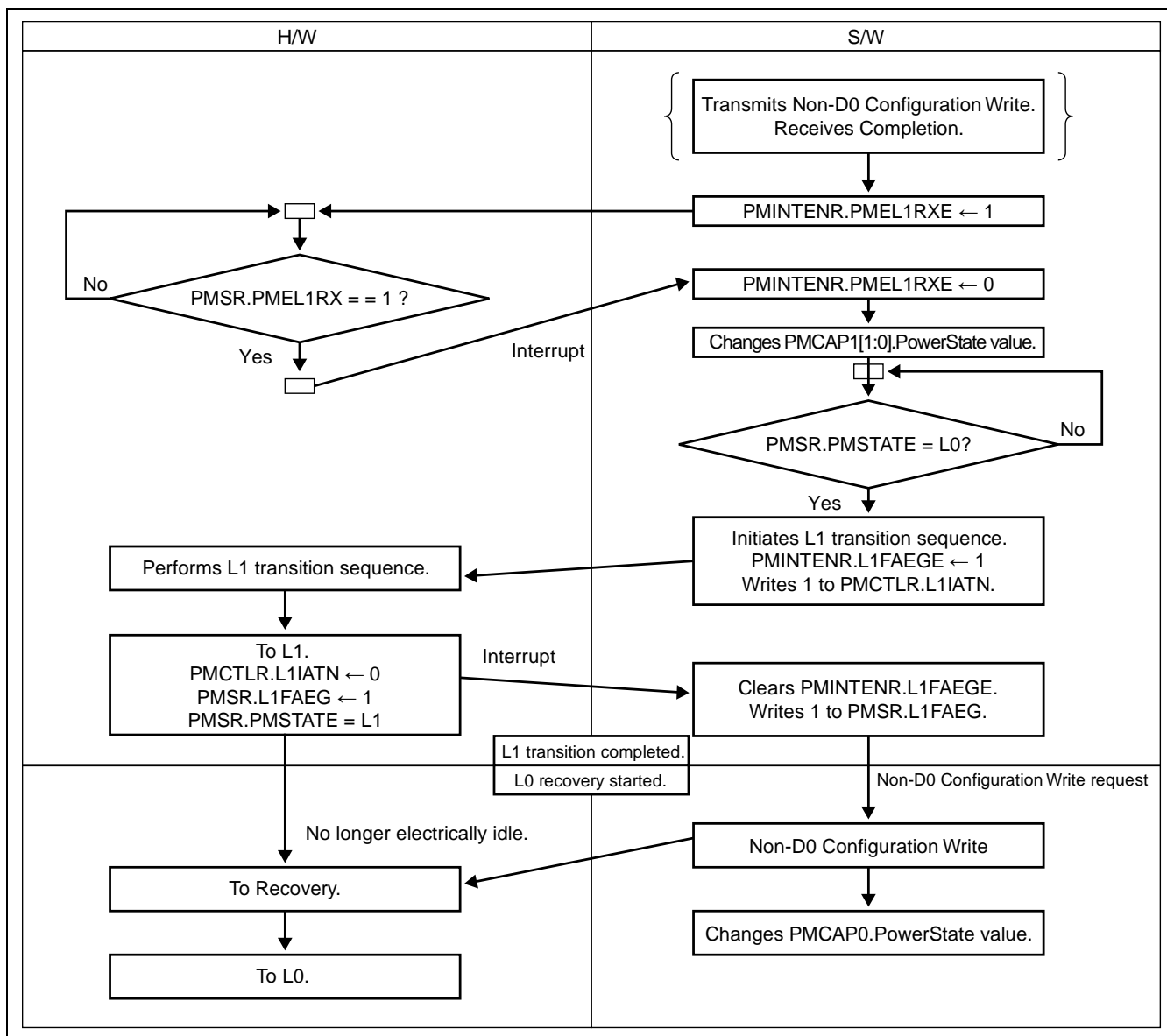


Figure 49.29 Transition from L0 to L1 to L0 (Root Port)

(2) Active State Power Management (ASPM)

This section describes PCI Express-specific power management mode. Table 49.15 shows power management operations in Active State Power Management (hereinafter referred to as ASPM) mode. ASPM is automatically controlled by hardware; software should take nothing particular into account.

Table 49.15 ASPM Operation

| No. | Transition | Target Device |
|-----|---------------|----------------------------------|
| 1 | L0 → L0s → L0 | Common to Endpoint and Root Port |

(a) L0 → L0s → L0

When a link is in the idle state for 4 μs, hardware automatically initiates the L0s transition sequence. This module determines a link to be in the idle state in any of the following cases.

- Cases in which No TLP is transmitted
 1. No TLP to be transmitted is scheduled in the VC buffer.
 2. TLP is scheduled but there are not enough credits to transmit the TLP in the VC buffer.
 3. TLP is scheduled and there are enough credits but Non-posted cannot be issued according to the limitations on the number of Outstanding by this module.
- Cases in which no DLLP is scheduled for transmission

Figure 49.30 shows an overall overview of L0 → L0s → L0 transition.

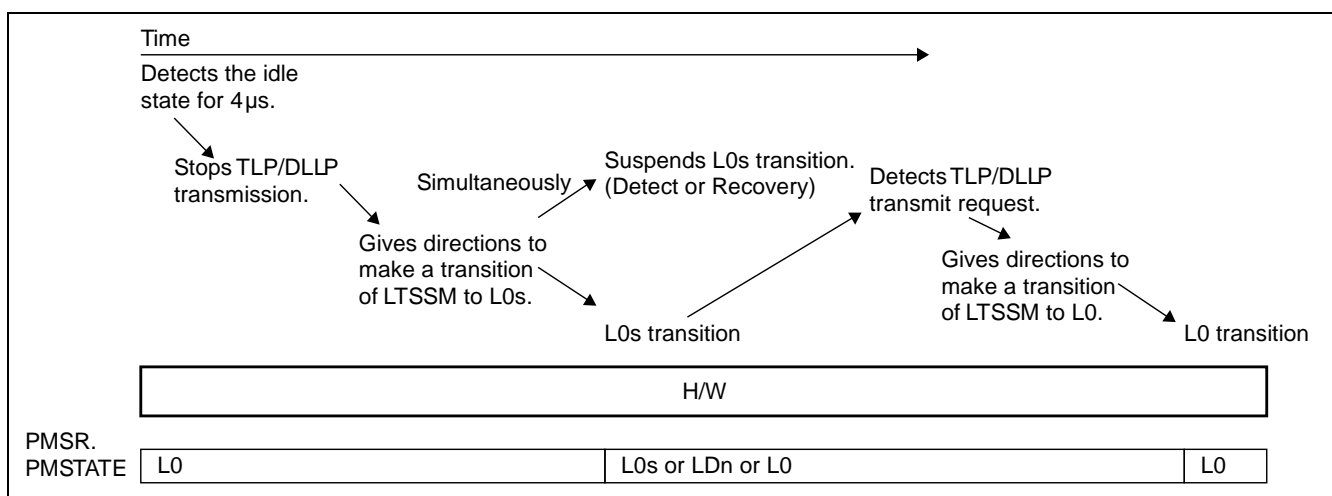


Figure 49.30 Overall Overview of L0 → L0s → L0 (Common to Endpoint/Root Port)

The L0s transition sequence is as follows.

1. Detects the idle state for 4 μs.
2. Stops TLP/DLLP transmission.
3. Gives directions to make a transition to L0s.

Note that if the directions to make a transition to L0s and the directions to make a transition to Detect or Recovery are given simultaneously in this module, the directions to make a transition to Detect or Recovery are given priority. If the directions are given to make a transition to L0s and then the directions are given to make a transition to the other state (such as Detect and Recovery), a transition to L0s is first made and then a transition to the designated state is made.

The sequence for a recovery from L0s is: when there is TLP/DLLP to be transmitted, recovery from L0s to L0 is automatically made. After recovery, TLP/DLLP is transmitted.

49.3.11 Error Handling

This section describes handling of the errors detected by this module.

(1) Error Types

This module classifies detected errors into four types: three types defined in the PCI Express standard (correctable errors, non-fatal errors, and fatal errors) and other errors (system errors). It reports each type of error through the corresponding interrupt.

(a) Correctable Error

Correctable errors are defined as the type of errors from which recovery is possible through the PCI Express protocol. Recovery from the errors is performed by hardware.

When the conditions described later are satisfied, occurrence of a correctable error is signaled by the INT_PCICERR interrupt.

(b) Non-Fatal Error

Non-fatal errors are defined as the type of errors from which recovery is impossible through the PCI Express protocol but which is not fatal. After this error, a single packet is discarded but the subsequent data communication can be continued. In most cases, error processing such as retransmission of the discarded packet is needed.

When the conditions described later are satisfied, occurrence of a non-fatal error is signaled by the INT_PCINFERR interrupt.

(c) Fatal Error

Fatal errors are defined as the type of errors from which recovery is impossible through the PCI Express protocol and which is fatal. Specifically, the errors that require resetting the communication path or other appropriate processing fall into this type. In most cases, this module must be initialized.

When the conditions described later are satisfied, occurrence of a fatal error is signaled by the INT_PCIFERR interrupt.

(d) Other Errors Detected by This Module (System Error)

The errors that do not fall into the above types are classified as system errors. The necessary processing depends on the cause of each error.

Occurrence of a system error is signaled by the INT_PCISERR interrupt.

(2) Priority of Errors

Upon detecting multiple errors during reception of a single packet, this module reports only one error that it determines as the most important according to the following priority among errors.

Receiver Overflow

Malformed TLP

Unsupported request (UR), or unexpected completion

Poisoned TLP received

(3) Correctable Error

Correctable errors are defined as the type of errors from which recovery is possible without loss of transfer data through hardware processing such as automatic retransmission request.

If a correctable error occurs, the EXPCAP2[16].CEDTCD bit is set to 1. If a correctable error occurs when the EXPCAP2[0].CERPE bit is 1 in Endpoint mode, the ERR_COR message is transmitted to the Root Port to notify occurrence of an error. In Root Port mode, the INT_PCICERR interrupt is generated if EXPCAP7[0].SERRCEE is 1.

In Root Port mode, if the ERR_COR message is received from an Endpoint when all of the PCICONF15[17].SERRE, EXPCAP2[0].CERPE, and EXPCAP7[0].SERRCEE bits are 1, the INT_PCICERR interrupt is generated.

The following errors are classified as correctable errors.

(a) Receiver error

8b/10b decode error

Disparity error

Elastic buffer overflow

Elastic buffer underflow

(b) BAD TLP

LCRC error

Sequence number error

(c) BAD DLLP

16-bit CRC error

(d) Replay timeout

Ack/Nak timeout

(e) Replay Num rollover

Replay Num rollover

(4) Non-Fatal Error

A non-fatal error cannot be corrected through hardware control and one packet of data will be lost. Therefore, recovery processing by software, such as retransmission processing, is needed.

If a non-fatal error occurs, the EXPCAP2[17].NFEDTCD bit is set to 1. If a non-fatal error occurs when PCICONF1[8].SERRE is 1, the PCICONF1[30].SSE bit is set to 1. If a non-fatal error occurs when the EXPCAP2[1].NFERPE bit or PCICONF1[8].SERRE bit is 1 in Endpoint mode, the ERR_NONFATAL message is transmitted to the Root Port to notify occurrence of an error. In Root Port mode, the INT_PCINFERR interrupt is generated if EXPCAP7[1].SERRNFEE is 1. However, if the error source is an unsupported request (described later), this module does not send the message or generate an interrupt when the conditions described later are satisfied.

In Root Port mode, if the ERR_NONFATAL message is received, PCICONF7[30].RSE is set to 1. If the ERR_NONFATAL message is received when the PCICONF15[17].SERRE and PCICONF1[8].SERRE bits are 1, the PCICONF1[30].SSE bit is set to 1. If the ERR_NONFATAL message is received from an Endpoint when the PCICONF15[17].SERRE bit is 1, the EXPCAP2[1].NFERPE or PCICONF1[8].SERRE bit is 1, and the EXPCAP7[1].SERRNFEE bit is 1, the INT_PCINFERR interrupt is generated. However, if the error source is an unsupported request (described later), this module does not generate an interrupt when the conditions described later are satisfied. In addition, if the non-fatal error comes under the advisory non-fatal error cases described later, separately defined processing is performed.

The following errors are classified as non-fatal errors.

(a) Poisoned TLP Reception

If a packet with the Poisoned attribute is written, it is handled as a non-fatal error.

This module does not transfer a write request of a packet from the PCI Express side to the internal bus side if the packet has the Poisoned attribute. If a request with the Poisoned attribute is received, the packet is discarded.

If the completion responding to the request issued by this module has the Poisoned attribute, the data in the completion is not transferred to the internal bus side.

If ConfigurationWrite Type0, I/O Write, Set_Slot_Power_Limit, or MSI has the Poisoned attribute, it is handled also as an unsupported request.

(b) Unsupported Request (UR) Reception

The following requests are handled as unsupported requests.

- Access to the area other than the PCI address area specified with BAR
- Reception of memory/IO request in Root Port mode when PCICONF1[2].BME = 0
- Access to memory space when PCICONF1[1].MSE = 0
- Access to IO space when PCICONF1[0].IOSE = 0
- Reception of a message with an undefined message code
- Reception of MsgD with message code Msg
- Reception of a message to be received only by an Endpoint in Root Port mode
- Reception of a message to be received only by a Root/Switch in Endpoint mode
- Reception of a Type0 Vendor Defined Message
- Reception of TLP other than configuration access and message in the Non-D0 state
- Access to a configuration register not implemented
- Reception of a Type1 Configuration Request
- Reception of a Configuration Request in Root Port mode
- Reception of a packet without the No Snoop attribute when acceptance of a snoop transaction is rejected by VCCAP4[15].RJCTSNPTR

- Reception of a MRdLk request

If an unsupported request is received, the EXPCAP2[19].URDTCD bit is set to 1.

If an unsupported request is received when both the EXPCAP2[3].URRPE and PCICONF1[8].SERRE bits are 0, neither transmission of a message nor generation of an interrupt is performed.

If an unsupported request is received, the packet is discarded.

(c) Completer Abort (CA)

If an error occurs on the internal bus when the packet is transferred from the PCI Express to the internal bus, this module considers it as a Completer Abort and performs error handling. When the packet received from the PCI Express is a Non-Posted request, this module returns a completion with the Completer Abort (CA)– Completion Status thus setting the PCICONF1[27].STA bit to 1.

If the packet received by this module from the PCI Express side violates the PCI Express standard or is considered to be damaged, it is handled as an unsupported request or malformed TLP, not as a Completer Abort.

(d) Unexpected Completion Reception

If the completion not corresponding to the request issued by this module is received, it is considered as an unexpected completion and error handling is performed.

Reception of an unexpected completion falls into the advisory non-fatal error cases described later.

(5) Fatal Error

Fatal errors are defined as the type of errors from which recovery is impossible through hardware and which usually require system-level error recovery processing such as initialization of the PCI Express.

If a fatal error occurs, the EXPCAP2[18].FEDTCD bit is set to 1. If a fatal error occurs when the PCICONF1[8].SERRE bit is 1, the PCICONF1[30].SSE bit is set to 1. If a fatal error occurs when the EXPCAP2[2].FERPE or PCICONF1[8].SERRE bit is 1, in Endpoint mode the ERR_FATAL message is transmitted to the Root Port to notify occurrence of an error. In Root Port mode, the INT_PCINFERR interrupt is generated if EXPCAP7[2].SERRFEE is 1.

In Root Port mode, if the ERR_FATAL message is received, PCICONF7[30].RSE is set to 1. If the ERR_FATAL message is received when both of the PCICONF15[17].SERRE and PCICONF1[8].SERRE bits are 1, the PCICONF1[30].SSE bit is set to 1. If the ERR_FATAL message is received from an Endpoint when the PCICONF15[17].SERRE bit is 1, the EXPCAP2[1].FERPE or PCICONF1[8].SERRE bit is 1, and the EXPCAP7[2].SERRFEE bit is 1, the INT_PCIFERR interrupt is generated.

The following errors are classified as fatal errors.

- Data Link Layer Protocol Error
- Reception of malformed TLP

The following packets are considered as malformed TLPs, which cause a fatal error when received. These packets are discarded when received.

- Reception of a packet with data size exceeding the Max Payload Size
- Reception of a packet with data of different length from the length specified in the size field
- Reception of a TLP with TD field in header = 0 and with the digest field
- Reception of a TLP with TD field in header = 1 and without the digest field
- Reception of an Assert_INTx/Deassert_INTx message of non TC0
- Reception of a power management message of non TC0
- Reception of an error signal message of non TC0

- Reception of an unlock message of non TC0
- Reception of a Set_Slot_Power_Limit message of non TC0
- Reception of a Set_Slot_Power_Limit message without data
- Reception of a completion with data of different size from the size specified in the Length field
- Reception of a completion with data size exceeding the Max Payload Size
- Reception of a TLP of undefined type or fmt
- Reception of a TLP of the unassigned TC
- Reception of an IO request of TC! = 0, Attr[1:0]! = 0, Length[9:0]! = 1, or Last DWBE[3:0]! = 0
- Reception of a configuration request of TC! = 0, Attr[1:0]! = 0, Length[9:0]! = 1, or Last DWBE[3:0]! = 0
- Reception of an Assert_INTx/Deassert_INTx message in Endpoint mode
- Reception of a memory request with illegal DWBE
- Reception of a memory request crossing a 4-Kbyte boundary

(6) System Error

System errors are defined as the errors that are not classified as any of the correctable, non-fatal, and fatal errors.

This module can signal the following events as the system errors by so setting PCIEERRFER.

- Internal bus error (reception of error response or disagreement of read data size)
- Posted/Non-Posted Buffer overflow
- CplLk reception
- Reception of an Unexpected Completion
- Disagreement of read data size requested and Completion data size
- Completion timeout
- Reception of a Completion with the CRS/CA/UR status
- Transmission of a Completion with the CA/UR status

If any system error occurs, the INT_PCISERR interrupt is generated.

(7) Advisory Non-Fatal Error Case

The PCI Express standard prescribes that some of non-fatal errors should not be processed when the applicable packet is received because the errors should be judged in terms of cause and significance by the packet transmission source, software, or system. These non-fatal errors are called advisory non-fatal error cases, which include the cases described in the following sections. If such a case is detected, general non-fatal error processing is not performed but processing prescribed individually is performed instead.

(a) Return of Completion with Unsupported Request/Completer Abort Status

If a completion with unsupported request/Completer Abort status is returned, this module does not perform non-fatal error processing defined in the PCI Express standard. This module records occurrence of an error in PCIEERRFR[0].SENDURCPL or PCIEERRFR[1].SENDACPL, and generates an INT_PCISERR interrupt if the corresponding bit in PCIEERRFER is set to 1.

(b) Completion Timeout

If a completion timeout occurs, this module does not perform non-fatal error processing defined in the PCI Express standard. This module records occurrence of an error in PCIEERRFR[8].CPLTIMEOUT, and generates an INT_PCISERR interrupt if the corresponding bit in PCIEERRFER is set to 1. When software determines that further retransmission is impossible, allow the software to perform error processing such as transmission of ERR_COR.

(c) Reception of Unexpected Completion

If this module receives an unexpected completion, it does not perform non-fatal error processing defined in the PCI Express standard. This module records occurrence of an error in PCIEERRFR[12].UNEXPECTED COMPLETION, and generates an INT_PCISERR interrupt if the corresponding bit in PCIEERRFER is set to 1.

(d) Reception of Unsupported Request/Completer Abort Completion

If this module receives a completion of unsupported request/Completer Abort, it does not perform non-fatal error processing defined in the PCI Express standard. This module records occurrence of an error in PCIEERRFR[4].RECEIVEURCPL or PCIEERRFR[5].RECEIVECACPL, and generates an INT_PCISERR interrupt if the corresponding bit in PCIEERRFER is set to 1.

(e) Reception of Poisoned TLP

According to the PCI Express standard, the case in which a Poisoned TLP is received and reception is continued is also defined as an advisory non-fatal error case; however, an advisory non-fatal error of such a case does not occur with this module since this module prohibits continuation of Poisoned data processing. Although the cases related to the intermediate receiver are also defined by the standard, the corresponding processing is not performed since this module does not operate as an intermediate receiver.

49.3.12 Interrupts

Table 49.16 shows the interrupt signal lines of this module, definition, and the conditions under which interrupts are generated. The interrupt signals are synchronized with the internal bus clock.

Table 49.16 List of Interrupt Signal Lines

| Name | Definition | Conditions of Generation | Related Registers |
|---------------|--|---|--|
| pci_int_serr | System error interrupt | <ul style="list-style-type: none"> • Occurrence of a system error • Completion with the UR status is transmitted. • Completion with the CA status is transmitted. • Completion with the UR status is received. • Completion with the CA status is received. • Completion with the CRS status is received. • Completion timeout occurs. • Size error occurs. • Unexpected Completion is received. • CplLk is received. • Posted buffer overflows. • Non-Posted buffer overflows. • Transfer error occurs on the internal bus. | <ul style="list-style-type: none"> • PCIEINTR[0].INT_PCISERR • PCIEERRFR[0].SENDURCPL • PCIEERRFR[1].SENDACPL • PCIEERRFR[4].RCVURCPL • PCIEERRFR[5].RCVCACPL • PCIEERRFR[6].RCVCRSCPL • PCIEERRFR[8].CPLTOUT • PCIEERRFR[9].RCVSZECPL • PCIEERRFR[12].UNEXPCPL • PCIEERRFR[13].RCVCPLLK • PCIEERRFR[20].NPOVF • PCIEERRFR[21].POVF • PCIEERRFR[28].IBERR |
| pci_int_ferr | PCI Express Fatal error interrupt | <ul style="list-style-type: none"> • Occurrence of a fatal error • Malformed TLP • Data Link Layer Protocol Error | <ul style="list-style-type: none"> • PCIEINTR[1].INT_PCIFERR • EXPCAP2[18].FEDTCD |
| pci_int_nferr | PCI Express Non Fatal Error interrupt | <ul style="list-style-type: none"> • Occurrence of a non-fatal error • Unsupported request • Poisoned TLP | <ul style="list-style-type: none"> • PCIEINTR[2].INT_PCINFERR • EXPCAP2[17].NFEDTCD |
| pci_int_cerr | PCI Express Correctable Error interrupt | <ul style="list-style-type: none"> • Occurrence of a correctable error • Receiver Error • Bad DLLP • Bad TLP • Replay Number Rollover • Replay Timeout | <ul style="list-style-type: none"> • PCIEINTR[3].INT_PCICERR • EXPCAP2[16].CEDTCD |
| pci_int_power | Power-down interrupt | PMCAP1[1:0] is set to the Non-D0 state by Configuration Write. | PCIEINTR[4].INT_PCIPOWER |
| pci_int_mes | PCI Express message reception or error interrupt | <ul style="list-style-type: none"> • Reception of a message or occurrence of an error • PM_PME is received. • Set_Slot_Power_Limit is received. | <ul style="list-style-type: none"> • PCIEINTR[30].INT_PM_PME_RCV • EXPCAP8[16].PMEST • PCIEINTR[5].INT_PCIMES • PCIERMSGR[12].SLOTPOWER |

| Name | Definition | Conditions of Generation | Related Registers |
|------------------|---|---|---|
| pci_int_bw | PCI Express link bandwidth change interrupt | Change of link bandwidth <ul style="list-style-type: none"> Change of bandwidth is caused by intentional factor. Change of bandwidth is caused by reliability. | PCIEINTR[28].INT_PCIBW <ul style="list-style-type: none"> EXPCAP4[31].LKAUTOBWSTS EXPCAP4[30].LKBWMNGSTS |
| pci_int_txallemp | PCI Express all transmission buffer empty interrupt | All the PCI Express transmit buffers are empty. | PCIEINTR[29].INT_TXALLEMP |
| pci_int_te | PCI Express transfer error interrupt | Occurrence of a PCI Express transfer error <ul style="list-style-type: none"> PCIECDR is accessed with PCIECCTLR[31].CCIE = 0. The pertinent PIO space is accessed with PCIEPTCTLRn[31].PARE = 0. Transfer is executed with TC being set that is not mapped to VC0. PIO transfer is executed in Endpoint mode with PCICONF1[2].BME = 0. Transfer is executed with the attribute being set that is not enabled with the configuration register. Transfer other than PME message transfer is executed in the Non D0 state. PM_PME message transfer is executed in Endpoint mode with PMCAP1[8].PMEE = 0. | PCIEINTR[31].INT_PCITE |
| pci_int_sc | PCI Express status change interrupt | Change of the following status <ul style="list-style-type: none"> PHYRDY INTx Disable (PCICONF1[10].INTDIS) DL_Active- MSI Enable (MSICAP0[16].MSIE) PM Status (PMCAP1[1:0].Power State) | PCIEPHYSR[16].PHYRDYC PCIETSTR[12].INTXDC PCIETSTR[16].DLLACTC PCIETSTR[28].MSIEC PCIEPMSR[24].PSTC |
| pci_int_pm | PCI Express-IP power management interrupt | Detection of PM status <ul style="list-style-type: none"> PM_Enter_L1 DLLP is received. L1 initiation sequence is completed/suspended. | PCIEINTR[12].INT_PM <ul style="list-style-type: none"> PMSR[23].PMEL1RX PMSR[31].L1FAEG |
| pci_int_mac | PCI Express IP MAC interrupt | Detection of MAC status <ul style="list-style-type: none"> Transition of LTSSM to L0 is made. Link speed change is completed. Link speed change occurs. Link speed change fails. Link speed change succeeds. Link training is in progress. | PCIEINTR[13].INT_MAC <ul style="list-style-type: none"> MACSR[3].L0ENT MACSR[4].SPCHGFIN MACSR[5].SPCHG MACSR[6].SPCHGFAIL MACSR[7].SPCHGSUC MACSR[30].LKTR |

| Name | Definition | Conditions of Generation | Related Registers |
|-------------------|----------------------------------|---|---|
| pci_int_dmace | PCI Express DMAC error interrupt | <p>Occurrence of a PCI Express error</p> <ul style="list-style-type: none"> DMA transfer is initiated with DMAPCIEDMAOR.DMAE = 0. DMA transfer is initiated while a link is not established. DMA transfer is initiated in Non D0. DMA transfer is initiated with TC being set that is not mapped to VC0. DMA transfer is initiated in Endpoint mode with PCICONF1[2].BME = 0. DMA transfer is initiated with the attribute being set that is not enabled with the configuration register. A completion other than SC is received when PCI -> internal bus. A completion timeout occurs when PCI -> internal bus. A malformed completion is received when PCI -> internal bus. A Poisoned Completion is received when PCI -> internal bus. <p>Occurrence of an internal bus error</p> <ul style="list-style-type: none"> An error response is received. A response with data size different from the requested size is received. <p>Forced termination</p> <ul style="list-style-type: none"> Forced termination of DMA is completed. | PCIEDMCHSR0-7[11].PEE PCIEDMCHSR0-7[9].IBE PCIEDMCHSR0-7[12].CHTC |
| pci_int_dmac0 | PCI Express DMAC0 interrupt | Completion of data transfer | PCIEDMCHSR0[0].TE |
| pci_int_dmac1 | PCI Express DMAC1 interrupt | Completion of data transfer | PCIEDMCHSR1[0].TE |
| pci_int_dmac2 | PCI Express DMAC2 interrupt | Completion of data transfer | PCIEDMCHSR2[0].TE |
| pci_int_dmac3 | PCI Express DMAC3 interrupt | Completion of data transfer | PCIEDMCHSR3[0].TE |
| pci_int_dmac4 | PCI Express DMAC4 interrupt | Completion of data transfer | PCIEDMCHSR4[0].TE |
| pci_int_dmac5 | PCI Express DMAC5 interrupt | Completion of data transfer | PCIEDMCHSR5[0].TE |
| pci_int_dmac6 | PCI Express DMAC6 interrupt | Completion of data transfer | PCIEDMCHSR6[0].TE |
| pci_int_dmac7 | PCI Express DMAC7 interrupt | Completion of data transfer | PCIEDMCHSR7[0].TE |
| pci_int_msi[31:0] | MSI interrupt | MSI reception | PCIEMSI[31:0] |
| pci_int_intx[3:0] | INTx interrupt | Assert_INTx message reception | PCIEINTXR[3:0] |

Figure 49.31 and 49.32 show the interrupts whose corresponding interrupt status and enable bits are located in the multiple registers.

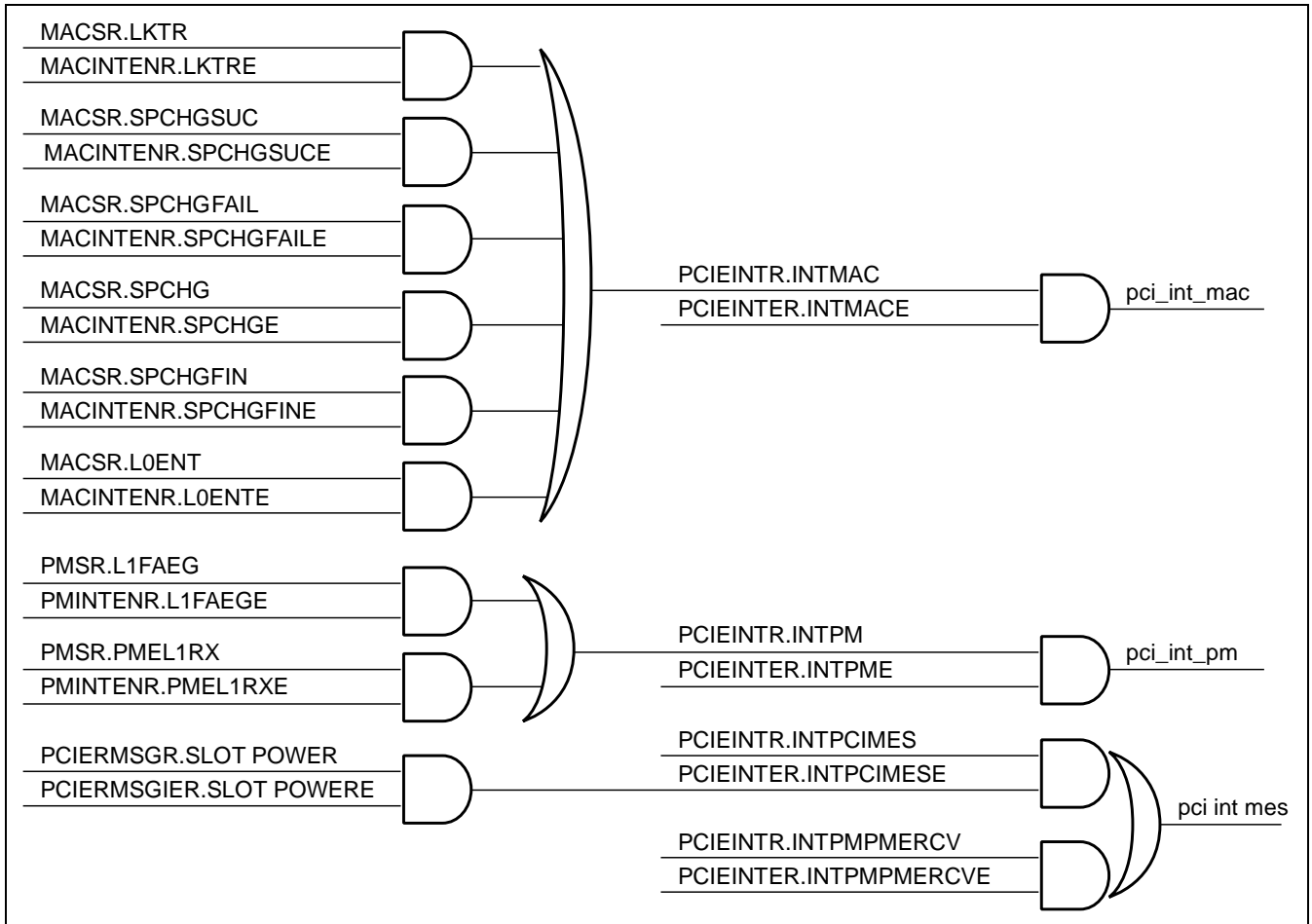


Figure 49.31 Interrupts Involving Multiple Registers (1)

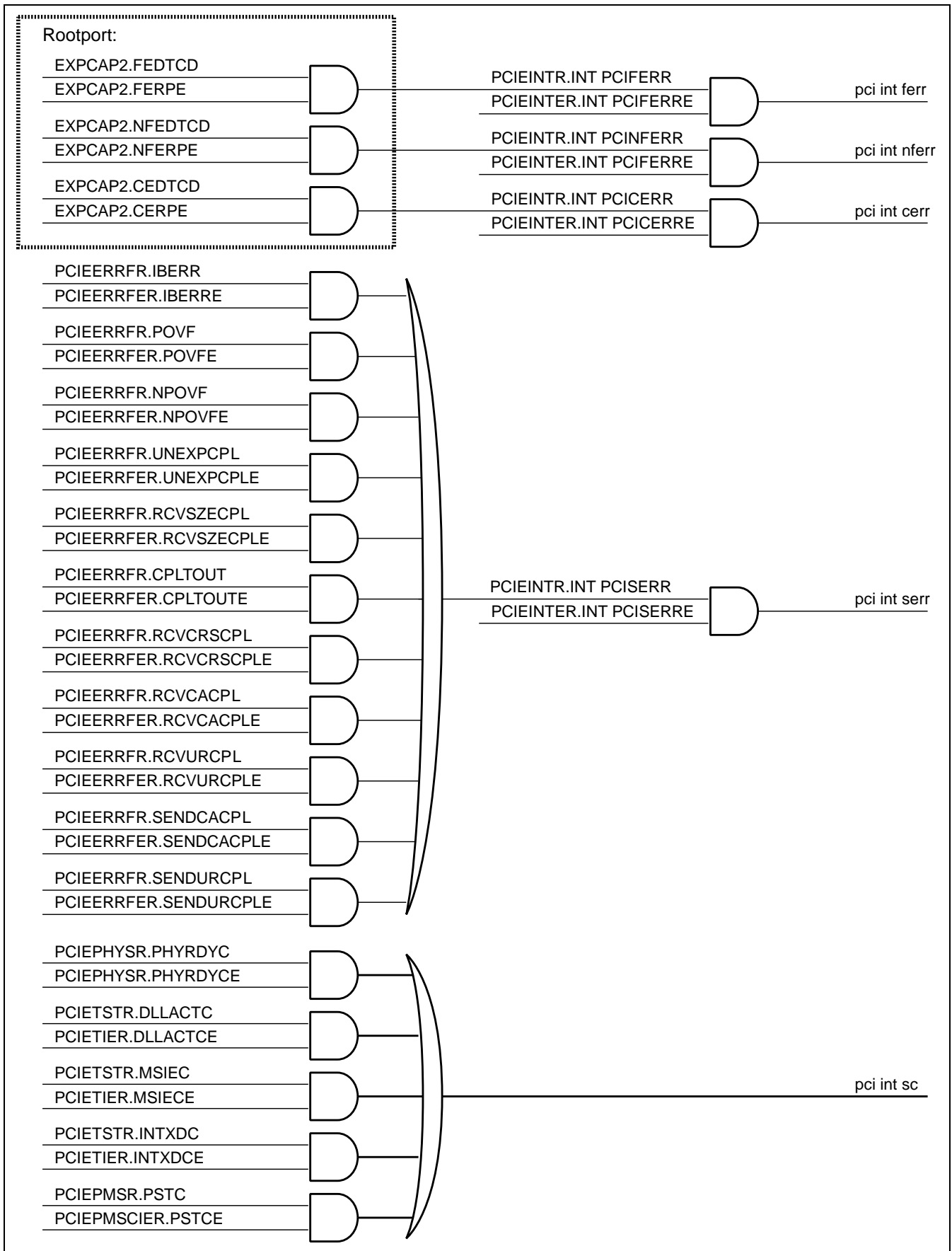


Figure 49.32 Interrupts Involving Multiple Registers (2)

49.3.13 Compliance Test

Test compliance according to the PCI Express standard.

49.3.14 Internal Reference Clock Supply

This section describes the internal reference clock supply. This specification applies from PCIE1 to PCIE0 and supports with RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, and RZ/G2N.

Reference clock supply sequence is as follows.

1. Cancel a reset and module standby for PCIE0. No need to cancel PCIE1.
2. Write 0 to PHY_REF_USE_PAD bit in PHY_CLK_RST register for PCIE0.

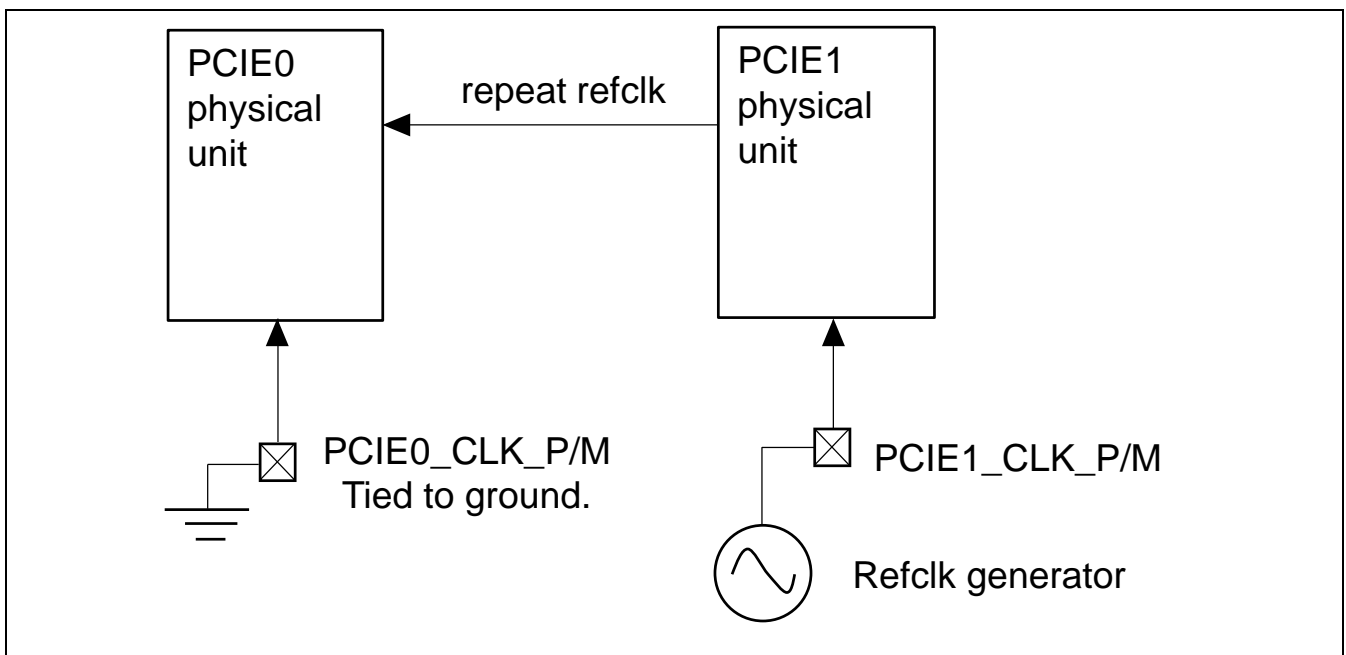


Figure 49.33 Internal Reference Clock Supply PCIE1 to PCIE0

49.4 Usage Notes

RZ/G2H

RZ/G2M V1.3

RZ/G2M V3.0

RZ/G2N

RZ/G2E

49.4.1 Write/Read Attributes of Configuration Registers

Although the registers in the configuration space have the register attributes defined by the PCI Express standard, these registers have different attributes when accessed by the internal bus. This section shows the attributes required for control. Software refers to the software running on the device (i.e., CPU) on the internal bus.

Table 49.17 Write/Read Attributes

| Attributes of PCI Express Registers | Description | Attributes of IP Registers | Example |
|-------------------------------------|---|--|---------------------------------|
| HWInit | Hardware initialized bit. Initialized by hardware or software. | R for fields initialized by hardware. | EXPCAP0[24].SLTI_MP |
| | | RW for fields initialized by software. | EXPCAP3[31:24].Port Number |
| RO | Read-only bit. Cannot be modified by software. | R for fields initialized by hardware. | PCICONF0[31:16].Device ID |
| | | RW for fields initialized by software. | EXPCAP0[23:20].Device Port Type |
| RW | Read/Write bit. Can be modified by software. | RW | PCICONF1[2].BME |
| RW1C | Read-only status bit. Writing 1 clears the bit. Writing 0 has no effect. | RW1C | PCICONF1[24].MDPE |
| ROS | Sticky-read-only bit. Cannot be modified by software. Not initialized by hot reset. | Applies to no IP registers. | — |
| RWS | Sticky-read write bit. Can be modified by software. Not initialized by hot reset. | RW | EXPCAP2[10].AUXPPME |
| RW1CS | Sticky-read-only status bit. Writing 1 clears the bit. Writing 0 has no effect. Not initialized by hot reset. | RW1C | PMCAP1[15].PMEST |
| RsvdP | Reserved bit. Read-only. May be used as a RW bit in the future. Reading the bit returns 0. | R | EXPCAP4[15:12] |
| RsvdZ | Reserved bit. Read-only. May be used as a RW1C bit in the future. Reading the bit returns 0. | R | EXPCAP2[31:22] |

The following shows which PCI Configuration Capability is supported by this IP.

| | | Offset | Capability | Spec Requirements | Root Port Type01 | Endpoint Type00 | |
|-------------------------|--|--------|---|--|------------------|-----------------|---|
| PCI Configuration Space | PCI3.0 Compatible Configuration Space | H'000 | PCI Compatible Configuration Registers | Required | √ | √ | |
| | | H'03F | | | | | |
| | | H'040 | PCI Power Management Capability Structure | Required | √ | √ | |
| | | | MSI or MSI-X Capability Structure | Required if an interrupt resource is required | — | √ | |
| | | | PCI Express Capability Structure | Required | √ | √ | |
| | | H'0FF | | | | | |
| | PCI Express Extended Configuration Space | H'0100 | | Virtual Channel Capability | Optional | √ | √ |
| | | | | Device Serial Number Capability | Optional | x | x |
| | | | | Advanced Error Reporting Capability | Optional | x | x |
| | | | | PCI Express Root Complex Link Declaration Capability | Optional | x | x |
| | | | | PCI Express Root Complex Internal Link Control Capability | Optional | x | x |
| | | | | Power Budgeting Capability | Optional | x | x |
| | | | | PCI Express Root Complex Event Collector Endpoint Association Capability | Optional | x | x |
| | | | | Multi-Function Virtual Channel Capability | Optional | x | x |
| | | | Vendor-Specific Capability | Optional | x | x | |
| | | | RCRB Header Capability | Optional | x | x | |
| | H'0FFF | | | | | | |

49.4.2 Notes on Accessing Configuration Registers

- In Endpoint mode, writing to the configuration registers of this IP is prohibited except at initialization.
- “PCI RW” in the column header of the table in the description section of each register indicates the attribute of configuration access from the PCI Express bus in Endpoint mode; it is invalid in Root Port mode. For the registers with the same attribute shared in Endpoint mode and Root Port, “—” is indicated.

49.4.3 Registers Which Affect Hardware Internal Operation in Root Port Mode

Since the Endpoint configuration space may be accessed by the peer Root Port, it should be set appropriately. The registers implemented according to the Root Port configuration space are not accessed by Endpoint. Therefore, in Root Port mode, the user should appropriately set the registers that have influence on hardware operation shown in Table 49.18 at initialization. For details on register setting, refer to description of each register.

Table 49.18 Root Port Configuration Registers Which Affect Hardware Internal Operation

| Register Name | Bit Name |
|---|--|
| PCICONF1 | System Error Enable |
| | Parity Error Response |
| | Bus Master Enable |
| | Memory Space Enable |
| | IO Space Enable |
| PCICONF3 | Header Type |
| PCICONF15 | Secondary Bus Reset |
| | SERR Enable |
| | Parity Error Response |
| | Interrupt Line |
| PMCAP1 | Power State |
| EXPCAP0 | Device Port Type |
| EXPCAP1 | Max Payload Size Supported |
| EXPCAP2 | Max Read Request Size |
| | Enable No Snoop |
| | Extended Tag Enable |
| | Max Payload Size |
| | Enabled Relax Ordering |
| | Unsupported Request Reporting Enable |
| | Fatal Error Reporting Enable |
| | Non Fatal Error Reporting Enable |
| | Correctable Error Reporting Enable |
| | EXPCAP3 |
| Data Link Layer Active Reporting Capability | |
| Maximum Link Width | |
| Supported Link Speeds | |
| EXPCAP4 | Link Autonomous Bandwidth Interrupt Enable |
| | Link Bandwidth Management Interrupt Enable |
| | Extended Sync |
| | Retrain Link |
| | Link Disable |
| | ASPM Control |
| EXPCAP7 | CRS Software Visibility |
| | CRS Software Visibility Enable |
| | PME Interrupt Enable |
| | System Error on Fatal Error Enable |
| | System Error on Non Fatal Error Enable |
| | System Error on Correctable Error Enable |
| EXPCAP10 | Completion Timeout Disable |

| Register Name | Bit Name |
|----------------------|--------------------------------|
| EXPCAP12 | Compliance De-emphasis |
| | Compliance SOS |
| | Enter Modified Compliance |
| | Transmit Margin |
| | Selectable De-emphasis |
| | Enter Compliance |
| | Target Link Speed |
| VCCAP1 | Extended VC Count |
| VCCAP4 | VC(0) Reject Snoop Transaction |
| VCCAP5 | TC/VC(0) MAP |

Table 49.19 shows the registers which reflect the hardware status.

Table 49.19 Root Port Configuration Registers Which Reflect Hardware Status

| Register Name | Bit Name |
|---------------|----------------------------------|
| PCICONF1 | Detected Parity Error |
| | Signaled System Error |
| | Received Master Abort |
| | Received Target Abort |
| | Signaled Target Abort |
| | Master Data Parity Error |
| PCICONF7 | Detected Parity Error |
| | Received System Error |
| | Received Master Abort |
| | Received Target Abort |
| | Signaled Target Abort |
| | Master Data Parity Error |
| EXPCAP2 | Transaction Pending |
| | Unsupported Request Detected |
| | Fatal Error Detected |
| | Non Fatal Error Detected |
| | Correctable Error Detected |
| EXPCAP4 | Link Autonomous Bandwidth Status |
| | Link Bandwidth Management Status |
| | Data Link Layer Active |
| | Link Training |
| | Negotiated Link Width |
| | Current Link Speed |
| EXPCAP8 | PME Pending |
| | PME Status |
| | PME Requester ID |
| EXPCAP12 | Current De-emphasis |
| VCCAP6 | VC(0) Negotiation Pending |

49.4.4 Power Management States

Do not stop inputting the reference clock signal in the power management states (L0s/L). Transition to the power management states (L0s/L1) is made from L0.

49.4.5 State Transition

Transition to the L2 state is prohibited since L2 is not supported.

50. Serial Communication Interface with FIFO (SCIF)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

50.1 Overview

This LSI has serial communication interfaces with FIFO buffers (SCIF) that handles asynchronous communication and clock synchronous serial communication. The SCIF has two 16-stage FIFO buffers separately for transmission and reception, which enables fast, efficient, and uninterrupted communication. The following table lists the functions of each interface.

The presence of the SCK_n, RTS_m#, and CTS_m# pins depends on the product or channel. The synchronous mode and mode control functions are supported in the interface channel which has the pins related to these functions.

n = 0 to 5

m = 0, 1, 3, 4

Internal clock (PCK):

- RZ/G2H: S3D4φ
- RZ/G2M V1.3: S3D4φ
- RZ/G2M V3.0: S3D4φ
- RZ/G2N: S3D4φ.
- RZ/G2E: S3D4Cφ.

Internal clock (SCKi):

- RZ/G2H: S3D1φ
- RZ/G2M V1.3: S3D1φ
- RZ/G2M V3.0: S3D1φ
- RZ/G2N: S3D1φ.
- RZ/G2E: S3D1Cφ.

Table 50.1 Channel Information

| Name | Function | Pin | Base Address | Remarks | Second Generation RZ/G series products | | | |
|------|---|--|--------------|---|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 0 | SCIF-0 Asynchronous mode (modem control is enabled) Clock synchronous mode | RX0, TX0, SCK0, RTS0#, and CTS0# | H'E6E6_0000 | Transmission/rec eption clock can be supplied externally from SCK0 pin. | √ | √ | √ | √ |
| | | Asynchronous mode Clock synchronous mode | | | RX0, TX0 and SCK0 | √ | √ | √ |
| 1 | SCIF-1 Asynchronous mode (modem control is enabled) Clock synchronous mode | RX1, TX1, SCK1, RTS1#, and CTS1# | H'E6E6_8000 | Transmission/rec eption clock can be supplied externally from SCK1 pin. | √ | √ | √ | √ |
| | | Asynchronous mode Clock synchronous mode | | | RX1, TX1 and SCK1 | √ | √ | √ |
| 2 | SCIF-2 Asynchronous mode Clock synchronous mode | RX2, TX2 and SCK2 | H'E6E8_8000 | Transmission/rec eption clock can be supplied externally from SCK2 pin. | √ | √ | √ | √ |
| 3 | SCIF-3 Asynchronous mode (modem control is enabled) Clock synchronous mode | RX3, TX3, SCK3, RTS3#, and CTS3# | H'E6C5_0000 | Transmission/rec eption clock can be supplied externally from SCK3 pin. | √ | √ | √ | √ |
| | | Asynchronous mode Clock synchronous mode | | | RX3, TX3 and SCK3 | √ | √ | √ |
| 4 | SCIF-4 Asynchronous mode (modem control is enabled) Clock synchronous mode | RX4, TX4, SCK4, RTS4#, and CTS4# | H'E6C4_0000 | Transmission/rec eption clock can be supplied externally from SCK4 pin. | √ | √ | √ | √ |
| | | Asynchronous mode Clock synchronous mode | | | RX4, TX4 and SCK4 | √ | √ | √ |
| 5 | SCIF-5 Asynchronous mode Clock synchronous mode | RX5, TX5 and SCK5 | H'E6F3_0000 | Transmission/rec eption clock can be supplied externally from SCK5 pin. | √ | √ | √ | √ |

50.1.1 Features

The SCIF has the following features.

- **Asynchronous serial communication mode**

The SCIF performs serial data communication based on a character-by-character asynchronous system. This feature enables serial data communication with standard asynchronous communication chips that support Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communication Interface Adapter (ACIA). There is a choice of eight serial data transfer formats.

 - Data length: 7 or 8 bits
 - Stop bit length: 1 or 2 bits
 - Parity: Even/odd/none
 - Receive error detection: Parity, framing, and overrun errors
 - Break detection:

A break is detected when a framing error lasts for more than 1 frame length at Space 0 (low level).

When a framing error occurs, a break can also be detected by reading the RX pin level directly from the serial port register (SCSPTR).
- **Clock synchronous serial communication mode**

The SCIF performs serial data communication synchronized with a clock. This feature enables serial data communication with other LSIs that support synchronous communication. There is a single serial data communication format for clock synchronous serial communication.

 - Data length: 8 bits
 - Receive error detection: Overrun errors
- **Full-duplex communication capability**

The SCIF has an independent transmitter and receiver that enable simultaneous transmission and reception. The transmitter and receiver both have a 16-stage FIFO buffer structure, enabling continuous serial data transmission and reception.
- **On-chip baud rate generator, enabling any bit rate to be selected**

The SCIF enables choice of a clock source for transmission/reception: a clock from the on-chip baud rate generator based on the internal clock or an external clock.
- **Eight interrupt sources**

The SCIF has eight types of interrupt sources – receive-data-ready, receive-FIFO-data-full, break, transmit-FIFO-data-empty, transmit-end, receive-error, overrun-error and time-out and enables any of them to be requested independently.
- **DMA data transfer**

When the transmit FIFO register is empty or the receive FIFO register has received data, issuing a DMA transfer request activates the DMA controller (DMAC) to execute a data transfer.
- **The amount of data in the transmit/receive FIFO registers and the number of receive errors in receive data in the receive FIFO register are available.**
- **In asynchronous mode, a receive data ready (DR) or a timeout error (TO) can be detected during reception.**

50.1.2 Block Diagram

Figure 50.1 shows the SCIF block diagram.

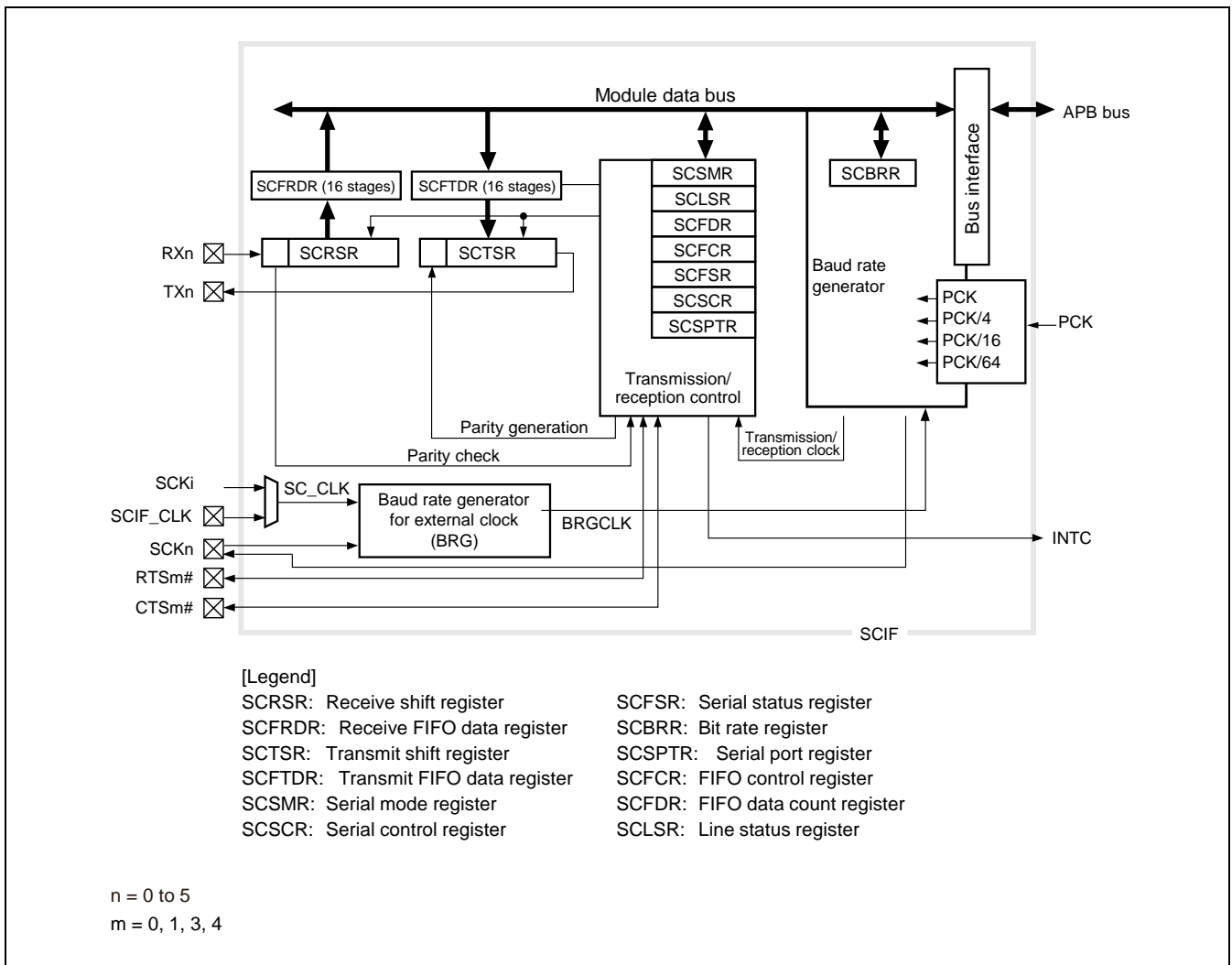


Figure 50.1 SCIF Block Diagram

50.1.3 External Pins

Table 50.2 shows the SCIF pin configuration. Pin functions can differ with the interface number. Other functions are also multiplexed on the same pins, so the multiplexed pin settings may restrict usage of the pins.

Table 50.2 Pin Configuration

| Name | Abbreviation | I/O | Function | Second Generation RZ/G Series Products | | | |
|--------------------------------|--------------|--------|---|---|----------------------------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Serial clock pin | SCK | I/O | Synchronous clock I/O | √ | √ | √ | √ |
| Receive data pin | RX | Input | Receive data input | √ | √ | √ | √ |
| Transmit data pin | TX | Output | Transmit data output | √ | √ | √ | √ |
| Modem control pin | CTS# | I/O | Transmission enabled | √ | √ | √ | √ |
| Modem control pin | RTS# | I/O | Transmission request | √ | √ | √ | √ |
| Baud rate generation clock pin | SCIF_CLK | Input | Clock for input to the baud rate generator for the external clock | √ | √ | √ | √ |

These pins are made to function as serial pins by setting up SCIF operation using bit C/A# in SCSMR, bits TE, RE, CKE[1:0] in SCSCR, and bit MCE in SCFCR. SCSPTR of the SCIF can be used to handle the transmission and detection of break states.

50.1.4 Register Configuration

Table 50.3 shows the registers in the SCIF.

Table 50.3 Register Configuration

| | | | | | | Second Generation RZ/G Series Products | | | |
|-----------------------------|--------------|-------------|-----------------------------|---------------|------|---|----------------------------|--------|--------|
| | Abbreviation | R/W | Offset from Base Address | Initial Value | Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Serial mode register | SCSMR | R/W | H'00 | H'0000 | 16 | √ | √ | √ | √ |
| Bit rate register | SCBRR | R/W | H'04 | H'FF | 8 | √ | √ | √ | √ |
| Serial control register | SCSCR | R/W | H'08 | H'0000 | 16 | √ | √ | √ | √ |
| Transmit FIFO data register | SCFTDR | —/WB | H'0C | Undefined | 8 | √ | √ | √ | √ |
| Serial status register | SCFSR | R/WC0 *1 | H'10 | H'0060 | 16 | √ | √ | √ | √ |
| Receive FIFO data register | SCFRDR | R | H'14 | Undefined | 8 | √ | √ | √ | √ |
| FIFO control register | SCFCR | R/W | H'18 | H'0000 | 16 | √ | √ | √ | √ |
| FIFO data count register | SCFDR | R | H'1C | H'0000 | 16 | √ | √ | √ | √ |
| Serial port register | SCSPTR | R/W | H'20 | H'00XX*3 | 16 | √ | √ | √ | √ |
| Line status register | SCLSR | R/WC0 *2 | H'24 | H'0000 | 16 | √ | √ | √ | √ |
| Frequency division register | DL | R/W | H'30 | H'0000 | 16 | √ | √ | √ | √ |
| Clock Select register | CKS | R/W | H'34 | H'0000 | 16 | √ | √ | √ | √ |

Notes: 1. Only 0 can be written to bits 7 to 4, 1, and 0 to clear the flags. Bits 15 to 8, 3, and 2 are read-only bits and cannot be modified.

2. Only 0 can be written to bits 2 and 0 to clear the flags. Bits 15 to 3, and 1 are read-only bits and cannot be modified.

3. The initial values of SCSPTR bits 6, 4, 2, and 0 are undefined.

The table below lists the registers used in asynchronous mode, asynchronous mode with modem control, and clock synchronous mode. When setting up the registers, set the bits related to unsupported modes in each interface to their initial values. Otherwise, the SCIF may malfunction. Do not write to any registers other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from registers other than listed below are undefined.

Table 50.4 Register Settings in Each Mode

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| SCFRDR | — | — | — | — | — | — | — | — | * | * | * | * | * | * | * | * |
| SCFTDR | — | — | — | — | — | — | — | — | * | * | * | * | * | * | * | * |
| SCSMR | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | C | A | A | A | A | 0 | * | * |
| SCSCR | 0 | 0 | 0 | 0 | * | 0 | 0 | 0 | * | * | * | * | * | A | * | * |
| SCFSR | A | A | A | A | A | A | A | A | A | * | * | A | A | A | * | A |
| SCBRR | — | — | — | — | — | — | — | — | * | * | * | * | * | * | * | * |
| SCFCR | 0 | 0 | 0 | 0 | 0 | M | M | M | * | * | * | * | M | * | * | M |
| SCFDR | 0 | 0 | 0 | * | * | * | * | * | 0 | 0 | 0 | * | * | * | * | * |
| SCSPTR | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | M | M | M | M | C | C | * | * |
| SCLSR | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | A | 0 | * |
| DL | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * |
| CKS | * | * | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

[Legend]

* : Used in any mode.

A : Used in asynchronous mode.

M : Used in asynchronous mode (modem control is enabled) (in addition to A).

C : Used in clock synchronous mode.

0 : Reserved (the write value should always be 0.)

— : Undefined

50.1.5 Connected Module

Table 50.5 shows the connected modules to the SCIF.

Table 50.5 Connected modules

| Module name | Connected module name | Function of connected module |
|-------------|-----------------------|------------------------------|
| SCIF | AP-System Core | Access the Register |
| | CPG | Output Clocks |
| | PFC | Select External pins |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |
| | INTC | Control to interrupt |
| | SYS-DMAC | Control Direct Memory Access |

50.2 Register Description

Explanation of abbreviation of register

Initial value: Value of the register after power-on reset

—: Undefined value

R/W: The bit or field is readable and writable.

R/WC0: The bit or field is readable and writable. Writing 0 to the bit initializes the bit. Writing 1 to the bit is ignored.

R: The bit or field is readable only. When writing to the register, write 0 to it.

—/WB: The bit or field is writable. Note that values read from write-only bits are not guaranteed.

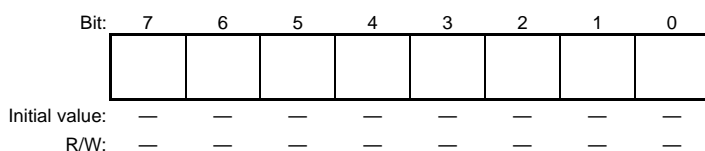
50.2.1 Receive Shift Register (SCRSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SCRSR is a register that receives serial data.

The SCIF sets serial data input to the SCRSR from the RX pin in the order received, starting with the LSB (bit 0), and converts it to parallel data. When one byte of data has been received, it is transferred to the receive FIFO data register SCFRDR, automatically.

SCRSR cannot be read from and written to by the CPU.



50.2.2 Receive FIFO Data Register (SCFRDR)

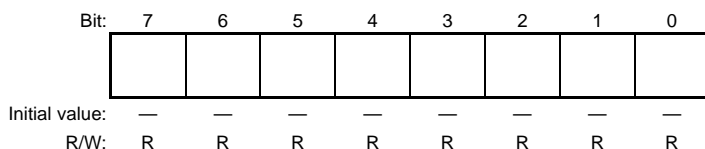
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SCFRDR is a 16-stage FIFO register that stores received serial data.

When the SCIF has received one byte of serial data, it transfers the received data from the receive shift register (SCRSR) to SCFRDR for storage, and reception is thus completed. SCRSR is then ready for reception, and is capable of receiving up to 16 consecutive bytes of data before SCFRDR is full.

SCFRDR is a read-only register and cannot be modified by the CPU. Note that the read value will be undefined while there is no receive data in SCFRDR. When SCFRDR is full of receive data, subsequent serial data is lost.

SCFRDR is read as an undefined value after a power-on reset.



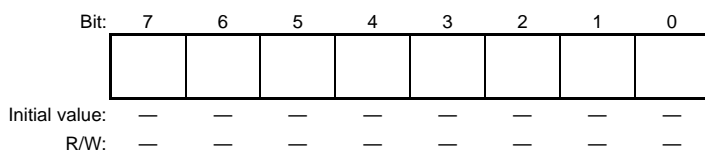
50.2.3 Transmit Shift Register (SCTSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SCTSR is a register that transmits serial data.

To perform serial data transmission, the SCIF first transfers transmit data from the transmit FIFO data register (SCFTDR) to SCTSR, then sends the data to the TX pin starting with the LSB (bit 0). When transmission of one byte is completed, the SCIF transfers the next transmit data from SCFTDR to SCTSR automatically, then starts transmission.

SCTSR cannot be read from and written to directly by the CPU.



50.2.4 Transmit FIFO Data Register (SCFTDR)

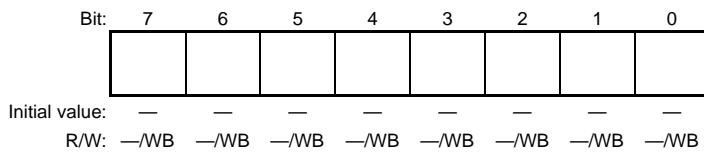
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SCFTDR is an 8-bit FIFO register of 16 stages that stores data for serial transmission.

If SCTSR is empty after transmit data has been written to SCFTDR, the SCIF transfers the data from SCFTDR to SCTSR and starts serial transmission.

SCFTDR is a write-only register and cannot be read from by the CPU. Writing further data to SCFTDR is no longer possible when it is full (contains 16 bytes of transmit data). Attempts at writing data to the register in this situation are ignored.

SCFTDR is read as an undefined value on a power-on reset.



50.2.5 Serial Mode Register (SCSMR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|------|-----|-----|------|------|-----|---|-----|----------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | C/A# | CHR | PE | O/E# | STOP | — | — | — | CKS[1:0] |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W |

SCSMR is a 16-bit register that sets the SCIF's serial transfer format and selects the baud rate generator clock source.

SCSMR can always be read from and written to by the CPU.

SCSMR is initialized to H'0000 by a power-on reset.

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | C/A# | B'0 | R/W | Communication Mode Selects asynchronous mode or clock synchronous mode for the SCIF operation. 0: Asynchronous mode 1: Clock synchronous mode Note: Whether clock synchronous mode can be selected or not depends on the interface. |
| 6 | CHR | B'0 | R/W | Character Length Selects 7 or 8 bits for asynchronous mode data length. In clock synchronous mode, the data length is fixed at 8 bits regardless of the CHR bit setting. 0: 8 bits 1: 7 bits* Note: * When the 7-bit data is selected, the MSB (bit 7) in the transmit FIFO data register (SCFTDR) is not transmitted. |
| 5 | PE | B'0 | R/W | Parity Enable Determines whether parity bit is added in transmission or not, and parity bit is checked in reception in asynchronous mode or not. In clock synchronous mode, the parity bit is not added and checked regardless of the PE bit setting. When bit PE is set to 1, the parity (even or odd) specified by bit O/E# is added to transmit data. In reception, the parity bit is checked for the parity (even or odd) specified by bit O/E#. 0: Disables parity bit addition and check. 1: Enables parity bit addition and check. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 4 | O/E# | B'0 | R/W | <p>Parity Mode</p> <p>Selects either even or odd parity to use in parity addition and check.</p> <p>In asynchronous mode, the O/E# bit setting is valid only when bit PE is set to 1, enabling parity bit addition and check. In clock synchronous mode or when parity addition or check is disabled in asynchronous mode, the O/E# bit setting is invalid.</p> <p>0: Even parity*1 1: Odd parity*2</p> <p>Notes: 1. When even parity is set, the parity bit is added in transmission so that the total number of 1-bits in the transmit character plus the parity bit is even. In reception, a check is performed to see if the total number of 1-bits in the receive character plus the parity bit is even.</p> <p>2. When odd parity is set, the parity bit is added in transmission so that the total number of 1-bits in the transmit character plus the parity bit is odd. In reception, a check is performed to see if the total number of 1-bits in the receive character plus the parity bit is odd.</p> |
| 3 | STOP | B'0 | R/W | <p>Stop Bit Length</p> <p>Selects 1 bit or 2 bits as the stop bit length in asynchronous mode.</p> <p>The stop bit setting is valid only in asynchronous mode. Since the stop bit is not added in clock synchronous mode, the STOP bit setting is invalid in this mode.</p> <p>In reception, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1, it is treated as a stop bit. If it is 0, it is treated as the start bit of the next transmit character.</p> <p>0: 1 stop bit*1 1: 2 stop bits*2</p> <p>Notes: 1. In transmission, high level of 1 bit (stop bit) is added to the end of a transmit character before it is sent.</p> <p>2. In transmission, high level of 2 bits (stop bits) are added to the end of a transmit character before it is sent.</p> |
| 2 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 1, 0 | CKS[1:0] | B'00 | R/W | <p>Clock Select 1 and 0</p> <p>These bits select the clock source for the on-chip baud rate generator.</p> <p>The clock source can be selected from PCK, PCK /4, PCK /16, and PCK /64, according to the setting of bits CKS1 and CKS0.</p> <p>B'00: PCK B'01: PCK /4 B'10: PCK /16 B'11: PCK /64</p> |

50.2.6 Serial Control Register (SCSCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SCSCR is a register that enables or disables transmission/reception by the SCIF, enables or disables interrupt requests, and selects transmission/reception clock source for the SCIF.

SCSCR can always be read from and written to by the CPU.

SCSCR is initialized to H'0000 by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|------|----|---|---|-----|-----|-----|-----|------|------|----------|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | TEIE | — | — | — | TIE | RIE | TE | RE | REIE | TOIE | CKE[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 15 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 | TEIE | B'0 | R/W | Transmit End Interrupt Enable When a transmit-end request is enabled by the TIE bit, the TEIE bit selects the source of the transmit end interrupt request from the following: <ul style="list-style-type: none"> Setting the TDFE flag in SCFSR Setting the TEND flag in SCFSR 0: The transmit FIFO data empty (TDFE) interrupt request is used. 1: The transmit end (TEND) interrupt request is used. |
| 10 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | TIE | B'0 | R/W | <p>Transmit Interrupt Enable</p> <p>Enables or disables a transmit-FIFO-data-empty interrupt (TDFE) request when the TEIE bit in SCSCR is pulled 0, if all of the following conditions are satisfied:</p> <ul style="list-style-type: none"> Serial transmit data has been transferred from SCFTDR to SCTSR. The number of data bytes in SCFTDR is equal to or less than the transmit trigger count. The TDFE flag in SCFSR is set to 1. <p>Enables or disables a transmit-end interrupt (TEND) request when the TEIE bit of SCSCR is set to 1, if both of the following conditions are satisfied:</p> <ul style="list-style-type: none"> Transmission was ended because there is no valid data in SCFTDR when the last bit of the transmit character in SCTSR was transmitted. The TEND flag of SCFSR is set to 1. <p>0: When the TEIE bit is 0, disables transmit-FIFO-data-empty interrupt (TDFE) request. When the TEIE bit is 1, disables transmit-end interrupt (TEND) request.</p> <p>1: When the TEIE bit is 0, enables transmit-FIFO-data-empty interrupt (TDFE) request. When the TEIE bit is 1, enables transmit-end interrupt (TEND) request.</p> |
| 6 | RIE | B'0 | R/W | <p>Receive Interrupt Enable</p> <p>Enables or disables a receive-FIFO-data-full interrupt request when the RDF flag in SCFSR is set to 1, a receive-data-ready interrupt request when the DR flag in SCFSR is set to 1, a receive-error interrupt request when the ER flag in SCFSR is set to 1, a break interrupt request when the BRK flag in SCFSR is set to 1, and an overrun error interrupt request when the ORER flag in SCLSR is set to 1.</p> <p>0: Disables receive-FIFO-data-full interrupt (RDF) requests, receive-data-ready interrupt (DR) requests, receive-error interrupt (ER) requests, break interrupt (BRK) requests, and overrun error interrupt (ORER) requests.</p> <p>1: Enables receive-FIFO-data-full interrupt (RDF) requests, receive-data-ready interrupt (DR) requests, receive-error interrupt (ER) requests, break interrupt (BRK) requests, and overrun error interrupt (ORER) requests.</p> |
| 5 | TE | B'0 | R/W | <p>Transmit Enable</p> <p>Enables or disables the start of SCIF serial transmission. The SCIF starts serial transmission when transmit data is written to the SCFTDR while TE is 1. Before setting TE to 1, set SCSMR and SCFCR to specify the transmission format and reset the transmit FIFO.</p> <p>0: Disables transmission.</p> <p>1: Enables transmission.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|--|
| 4 | RE | B'0 | R/W | <p>Receive Enable</p> <p>Enables or disables the start of SCIF serial reception. When RE is 1, the SCIF starts serial reception by detecting a start bit in asynchronous mode or by detecting a synchronization clock input in clock synchronous mode. Before setting RE to 1, set SCSMR and SCFCR to specify the reception format and reset the receive FIFO.</p> <p>0: Disables reception.* 1: Enables reception.</p> <p>Note: Even if RE is cleared to 0, the DR, ER, BRK, RDF, FER, PER, TO and ORER flags are not affected, and retain their states.</p> |
| 3 | REIE | B'0 | R/W | <p>Receive Error Interrupt Enable</p> <p>Enables or Disables generation of receive-error interrupt (ER) requests, break interrupt (BRK) requests and overrun-error interrupt (ORER) requests.</p> <p>0: Disables receive-error interrupt (ER) requests, break interrupt (BRK) requests and overrun-error interrupt (ORER) requests.* 1: Enables receive-error interrupt (ER) requests, break interrupt (BRK) requests and overrun-error interrupt (ORER) requests.</p> <p>Note: * When REIE is 1, ER, BRK or ORER interrupt requests will occur even if RIE is cleared to 0. This setting is used to notify the interrupt controller of ER, BRK, and ORER interrupt requests during DMAC transfer.</p> |
| 2 | TOIE | B'0 | R/W | <p>Timeout Interrupt Enable</p> <p>Enables or disables generation of timeout interrupt (TO) requests when the TO flag in SCLSR is set to 1.</p> <p>0: Disables timeout interrupts (TO). 1: Enables timeout interrupts (TO).</p> |
| 1, 0 | CKE[1:0] | B'00 | R/W | <p>Clock Enable 1 and 0</p> <p>These bits select the SCIF clock source and enables or disables the clock output from the SCK pin.</p> <p>Whether the SCK pin functions as a serial clock output pin or a serial clock input pin is determined by combination of the CKE [1:0] bits settings. To specify synchronization clock output in clock synchronous mode, set C/A# to 1 in SCSMR then set CKE [1:0].</p> <p>See table 50.5 for the bit settings.</p> |

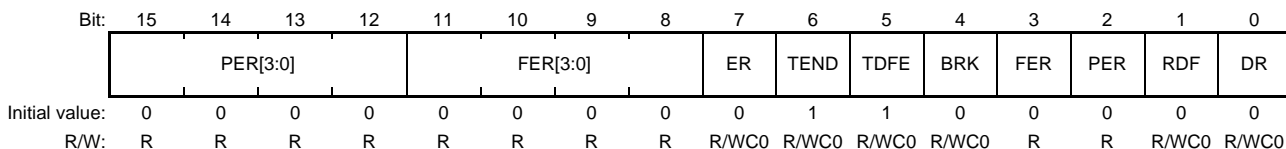
Table 50.6 Clock Selection

| CKE[1:0] | Mode | Clock Source | SCK Pin |
|-----------------|------------------------|--|--|
| B'00 | Asynchronous mode | Internal clock (PCK, PCK /4, PCK/16, PCK/64) | The SCK pin is not used. The SCK pin functions as an input pin (Input signals are ignored). (Initial value) |
| | Clock synchronous mode | | The SCK pin outputs the synchronization clock. (Initial value) |
| B'01 | Asynchronous mode | | The SCK pin outputs the clock (with a frequency 16 times the bit rate). |
| | Clock synchronous mode | | The SCK pin outputs the synchronization clock. |
| B'10 | Asynchronous mode | Baud rate generator output for external clock or SCK | When SC_CLK is selected: The SCK pin is an input pin (Input signals are ignored). Set the SC_CLK frequency so that the frequency of BRGCLK is 16 times the bit rate. |
| | Clock synchronous mode | | When SCK is selected: The SCK pin inputs the clock (with a frequency 16 times the bit rate). |
| | Clock synchronous mode | SCK* | The SCK pin inputs the synchronization clock. |
| B'11 | Prohibited | — | — |

Note: * It is not allowed to set synchronous communication using SC_CLK for input.

50.2.7 Serial Status Register (SCFSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



SCFSR is a 16-bit register. The lower 8 bits are a status flag that indicates the operating status of the SCIF, and the upper 8 bits indicate the number of reception errors of data in the receive FIFO register.

SCFSR can always be read from and written to by the CPU. However, the flags ER, TEND, TDFE, BRK, RDF, and DR cannot be written by 1. The FER and PER flags are read-only flags and cannot be modified.

SCFSR is initialized to H'0060 by a power-on reset.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 15 to 12 | PER[3:0] | All 0 | R | Parity Error Count These bits indicate the number of parity errors of receive data stored in the receive FIFO data register (SCFRDR). After the ER is set in SCFSR, the value represented by bits 15 to 12 indicates the number of parity errors. If a parity error occurs in all 16-byte receive data in SCFRDR, PER3 to PER0 are cleared to 0. |
| 11 to 8 | FER[3:0] | All 0 | R | Framing Error Count These bits indicate the number of framing errors of receive data stored in the receive FIFO data register (SCFRDR). After the ER bit is set in SCFSR, the value represented by bits 11 to 8 indicates the number of framing errors. If a framing error occurs in all 16-byte receive data in SCFRDR, FER3 to FER0 are cleared to 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 7 | ER | B'0 | R/WC0 | <p>Receive Error</p> <p>Indicates that a framing error or a parity error has occurred in reception.*¹ The ER flag is not affected by an error and retains its previous state when the RE bit is 0 in SCSCR.</p> <p>If a receive error occurs, receive data will be transferred to SCFRDR and reception operation will be continued. Whether there is a receive error in data read from SCFRDR can be determined by the FER and PER bits in SCFSR.</p> <p>0: Indicates that no framing or parity error has occurred in reception.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> A power-on reset is executed. 0 is written to ER. <p>1: Indicates that a framing error or a parity error has occurred in reception.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> The SCIF checks whether the stop bit at the end of receive data is 1, but the stop bit is 0.*² The number of 1-bits in receive data plus the parity bit does not match the parity setting (even or odd) specified by the O/E# bit in SCSMR during reception. <p>Note: In the 2-stop-bit mode, only the first stop bit is checked that the value is 1; the second stop bit is not checked.</p> |
| 6 | TEND | B'1 | R/WC0 | <p>Indicates that transmission has been ended * because there was no valid data in SCFTDR when the last bit of the transmit character was transmitted.</p> <p>0: Indicates that transmission is in progress.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> Transmit data is written to SCFTDR, and 0 is written to TEND. Data is written to SCFTDR by the DMAC. <p>1: Indicates that transmission has been ended.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> A power-on reset is executed. The TE bit in SCSCR is 0. There is no transmit data in SCFTDR when the last bit of a 1-byte serial transmission character is transmitted. <p>Note: When transmit data is written to SCFTDR by the DMAC in clock synchronous mode, the TEND flag may not be cleared. Therefore, if the DMAC is used for transmission in clock synchronous mode, the TEND flag should be read using the following procedure.</p> <ol style="list-style-type: none"> Check that data transfer is completed in the DMAC. Read the TEND flag. If TEND = 1, clear the TEND flag to 0. Read the TEND flag again. Use the second read TEND flag. |

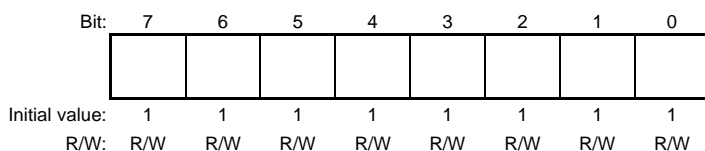
| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 5 | TDFE | B'1 | R/WC0 | <p>Transmit FIFO Data Empty</p> <p>Indicates that the SCIF has transferred data from SCFTDR to SCTSR, the number of data bytes in SCFTDR becomes equal to or less than the transmit trigger count specified by the TTRG1 and TTRG0 bits in SCFCR, and SCFTDR is ready to be written by new transmit data.</p> <p>SCFTDR is a 16-byte FIFO register. The maximum number of bytes that can be written to when TDFE = 1 is "16 – [the transmit trigger count]". If data exceeding this value is attempted to be written, the data will be ignored. The number of data bytes in SCFTDR is indicated by the upper bits of SCFDR.</p> <p>If the number of data written in SCFTDR is equal to or less than the transmit trigger count, this bit will be set to 1 even if it is cleared to 0 after it is read as 1.</p> <p>0: Indicates that the number of transmit data written to SCFTDR exceeds the transmit trigger count.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • Transmit data exceeding the specified transmit trigger count have been written to SCFTDR, and 0 is written to TDFE. • Transmit data exceeding the specified transmit trigger count have been written to SCFTDR by the DMAC. <p>1: Indicates that the number of transmit data in SCFTDR is equal to or less than the transmit trigger count.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> • A power-on reset is executed. • The number of transmit data in SCFTDR is equal to or less than the transmit trigger count after transmission. |
| 4 | BRK | B'0 | R/WC0 | <p>Break Detect</p> <p>Indicates that a receive data break signal has been detected. If a break signal is detected, receive data (H'00) transfer to SCFRDR is stopped. After the break is canceled and the receive signal returns to 1, the receive data transfer resumes.</p> <p>0: Indicates that no break signal has been received.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • A power-on reset is executed. • 0 is written to BRK. <p>1: Indicates that a break signal has been received.</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> • Data with a framing error is received, followed by the space "0" (low level) for at least one frame length. |
| 3 | FER | B'0 | R | <p>Framing Error</p> <p>Indicates that a framing error has been found in the data that is to be read next from SCFRDR in asynchronous mode.</p> <p>0: Indicates that there is no framing error in the receive data that is to be read from SCFRDR.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • A power-on reset is executed. • There is no framing error in the data that is to be read next from SCFRDR. <p>1: Indicates that there is a framing error in the receive data that is to be read from SCFRDR.</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> • There is a framing error in the data that is to be read next from SCFRDR. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 2 | PER | B'0 | R | <p>Parity Error</p> <p>This bit indicates that a parity error has been found in the data that is to be read next from SCFRDR in asynchronous mode.</p> <p>0: Indicates that there is no parity error in the receive data that is to be read from SCFRDR.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • A power-on reset is executed. • There is no parity error in the data that is to be read next from SCFRDR. <p>1: Indicates that there is a parity error in the receive data that is to be read from SCFRDR.</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> • There is a parity error in the data that is to be read next from SCFRDR. |
| 1 | RDF | B'0 | R/WC0 | <p>Receive FIFO Data Full</p> <p>Indicates that the received data has been transferred from SCRSR to SCFRDR, and the number of receive data bytes in SCFRDR becomes equal to or more than the receive trigger count specified by the RTRG1 and RTRG0 bits in SCFCR.</p> <p>SCFRDR is a 16-byte FIFO register. When RDF = 1, data equal to or more than the number of receive trigger data bytes can be read. When SCFRDR is empty, SCFRDR is read as an undefined value. The number of receive data bytes in SCFRDR is indicated by the lower bits of SCFDR.</p> <p>If the number of data in SCFRDR is equal to or more than the trigger count, this bit will be set to 1 even if it is cleared to 0. At this time, read receive data until the number of data in SCFRDR is less than the trigger count, read RDF as 1, and then clear RDF.</p> <p>0: Indicates that the number of receive data bytes in SCFRDR is less than the specified receive trigger count.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • A power-on reset is executed. • SCFRDR is read until the number of receive data bytes in SCFRDR is less than the receive trigger count, and 0 is written to RDF. • SCFRDR is read by the DMAC until the number of receive data bytes in SCFRDR is less than the receive trigger count. <p>1: Indicates that the number of receive data bytes in SCFRDR is equal to or more than the specified receive trigger count.</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> • Receive data more than the receive trigger count have been stored in SDFRDR. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 0 | DR | B'0 | R/WC0 | <p>Receive Data Ready</p> <p>Indicates that the number of data bytes in SCFRDR is less than the receive trigger count and that no further data has been received for at least 15 etu after the stop bit of the data received last in asynchronous mode. This bit is not set in clock synchronous mode.</p> <p>0: Indicates that data is being received or has been successfully received, and there is no receive data in SCFRDR.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • A power-on reset is executed. • All the receive data in SCFRDR has been read, and 0 is written to DR. • All the receive data in SCFRDR has been read by the DMAC. <p>1: Indicates that no further receive data has been received.</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> • The receive FIFO data register (SCFRDR) has the number of data byte that is below the receive trigger number of data bytes and no further data has arrived for at least 15 etu after the stop bit of the data received last. <p>Note: etu: Elementary Time Unit (time for transfer of 1 bit) Equivalent to 1.5 frames with an 8-bit, 1-stop-bit format.</p> |

50.2.8 Bit Rate Register (SCBRR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



SCBRR is an 8-bit register that sets the serial transmission/reception bit rate in accordance with the baud rate generator operating clock selected by the CKS1 and CKS0 bits in SCSMR. This baud rate generator is intended for PCK, PCK/4, PCK/16, and PCK/64. For details on the baud rate generator for external clock, see section 50.3.3, Baud Rate Generator for External Clock (BRG).

SCBRR can always be read from and written to by the CPU except for during transfer.

SCBRR is initialized to H'FF by a power-on reset.

The SCBRR setting is determined by the following equation:

[Asynchronous mode]

$$N = \frac{PCK}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$

[Clock synchronous mode]

$$N = \frac{PCK}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B: Bit rate (bit/s)

N: SCBRR setting for the baud rate generator ($0 \leq N \leq 255$) (which satisfies the electrical characteristics)

PCK: Peripheral module operating frequency (MHz)

n: Baud rate generator input clock ($n = 0, 1, 2, 3$)
(See the table 50.6 for the relation between n and the clock.)

Table 50.7 SCSMR Settings

| n | Clock | SCSMR.CKS[1:0] Setting |
|---|--------|------------------------|
| 0 | PCK | B'00 |
| 1 | PCK/4 | B'01 |
| 2 | PCK/16 | B'10 |
| 3 | PCK/64 | B'11 |

The bit rate error in asynchronous mode is determined by the following equation:

$$\text{error (\%)} = \left\{ \frac{PCK \times 10^6}{(N+1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100$$

50.2.9 FIFO Control Register (SCFCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SCFCR is a register that resets data counts and sets the number of trigger data bytes for the transmit and receive FIFO registers. It also has a loopback test enable bit.

SCFCR can always be read from and written to by the CPU except for during transfer.

SCFCR is initialized to H'0000 by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|------------|-----|-----------|-----|-----------|-----|-----|-------|--------|------|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | RSTRG[2:0] | | RTRG[1:0] | | TTRG[1:0] | | MCE | TFRST | RF RST | LOOP | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|--------------------------|-------------------------------|--|
| 15 to 11 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 to 8 | RSTRG[2:0] | B'000 | R/W | RTS Output Active Trigger The RTS# signal is high when the number of receive data bytes stored in SCFRDR is equal to or more than the specified trigger number shown below: B'000: 15 B'001: 1 B'010: 4 B'011: 6 B'100: 8 B'101: 10 B'110: 12 B'111: 14 |
| 7, 6 | RTRG[1:0] | B'00 | R/W | Receive FIFO Data Count Trigger These bits specify the number of receive data bytes that makes the RDF (receive data full) flag to be set in SCFSR. The RDF flag is set when the number of receive data bytes in SCFRDR is equal to or more than the specified trigger number shown below: |
| | RTRG[1:0] | Asynchronous Mode | Clock Synchronous Mode | |
| | B'00 | 1 | 1 | |
| | B'01 | 4 | 2 | |
| | B'10 | 8 | 8 | |
| | B'11 | 14 | 14 | |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|-----------|---------------|-----|---|
| 5, 4 | TTRG[1:0] | B'00 | R/W | <p>Transmit FIFO Data Count Trigger</p> <p>These bits specify the number of remaining transmit data bytes that makes the transmit FIFO data register empty (TDFE) flag to be set in SCFSR.</p> <p>The TDFE flag is set when the number of transmit data bytes in SCFTDR is equal to or less than the specified trigger count shown below, after transmission:</p> <p>B'00: 8 (8) B'01: 4 (12) B'10: 2 (14) B'11: 0 (16)</p> <p>Note: Values in parentheses indicate the empty space in SCFTDR in bytes.</p> |
| 3 | MCE | B'0 | R/W | <p>Modem Control Enable</p> <p>Enables or disables modem control signals CTS# and RTS#.</p> <p>This bit should always be set to 0 in clock synchronous mode. Whether modem control can be selected or not depends on the interface.</p> <p>0: Disables modem signals.* 1: Enables modem signals.</p> <p>Note: CTS# and RTS# control ports.</p> |
| 2 | TFRST | B'0 | R/W | <p>Transmit FIFO Data Register Reset</p> <p>Enables or disables a transmit FIFO data register reset that empties the register.</p> <p>0: Disables the reset.* 1: Enables the reset.</p> <p>Note: The register is reset by a power-on reset.</p> |
| 1 | RFRST | B'0 | R/W | <p>Receive FIFO Data Register Reset</p> <p>Enables or disables a receive FIFO data register reset that empties the register.</p> <p>0: Disables the reset.* 1: Enables the reset.</p> <p>Note: The register is reset by a power-on reset.</p> |
| 0 | LOOP | B'0 | R/W | <p>Loopback Test</p> <p>Enables or disables the loopback test by internally connecting the transmit output pin (TX) and receive input pin (RX), and the RTS# pin and CTS# pin. If the loopback test is specified, the signal on the RTS# pin is internally conveyed to CTS# pin, and data transmission is automatically suspended when RTS# is asserted.</p> <p>0: Disables the loopback test. 1: Enables the loopback test.</p> <p>Even if loopback test is specified by this bit, SCSPTR.RTS DT and SCSPTR.CTSDT bits does not reflect signal on loopback path, because those registers shows signals on RTS#/CTS# pins.</p> |

50.2.10 FIFO Data Count Register (SCFDR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SCFDR is a 16-bit register that indicates the number of data bytes stored in SCFTDR and that in SCFRDR.

The upper 8 bits indicate the number of transmit data bytes in SCFTDR, and the lower 8 bits indicates the number of receive data bytes in SCFRDR.

SCFDR can always be read by the CPU.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|--------|----|----|---|---|---|---|--------|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | T[4:0] | | | | — | — | — | R[4:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 to 8 | T[4:0] | H'00 | R | These bits show the number of data bytes untransmitted and still stored in SCFTDR. H'00 indicates that there is no transmit data in SCFTDR. H'10 indicates that SCFTDR is full of transmit data. |
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | R[4:0] | H'00 | R | These bits show the number of receive data stored in SCFRDR in bytes. H'00 indicates that there is no receive data in SCFRDR. H'10 indicates that SCFRDR is full of receive data. |

50.2.11 Serial Port Register (SCSPTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SCSPTR controls multiplexed input/output and data on the serial communication interface (SCIF) ports. Bits 1 and 0 control breaks in serial transmission/reception by reading input data from the RX pin and writing output data to the TX pin. Bits 3 and 2 read input data from and write output data to the SCK pin. Bits 5 and 4 read input data from and write output data to the CTS# pin. Bits 7 and 6 read input data from and write output data to the RTS# pin.

SCSPTR is a 16-bit register that can always be read from and written to by the CPU.

All SCSPTR bits except bits 6, 4, 2, and 0 are initialized to 0 by a power-on reset. The values of bits 6, 4, 2, and 0 are undefined.

Note: Whether modem control can be selected or not depends on the interface.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|-------|-------|-------|-------|-------|-------|------------|------------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | RTSIO | RTSDT | CTSIO | CTSdT | SCKIO | SCKdT | SPB2I O | SPB2D T |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | 0 | — | 0 | — | 0 | — |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | RTSIO | B'0 | R/W | Serial Port RTS# Pin Input/output Specifies input or output for the serial port RTS# pin. To actually set the RTS# pin as a port output pin to output the value set by the RTSdT bit, the MCE bit in SCFCR should be cleared to 0. 0: Indicates that this bit does not output the value of the RTSdT bit to the RTS# pin. 1: Indicates that this bit outputs the value of the RTSdT bit to the RTS# pin. |
| 6 | RTSDT | — | R/W | Serial Port RTS# Pin Data Specifies the input/output data level of the serial port RTS# pin. Whether the pin is set for input or output is determined by the RTSIO bit. When the pin is set for output, the value of the RTSdT bit is output to the RTS# pin. Regardless of the value of the RTSIO bit, the value of the RTS# pin is read from the RTSdT bit. The initial value of this bit is undefined after a power-on reset. 0: Indicates that the input/output data is low level. 1: Indicates that the input/output data is high level. |
| 5 | CTSIO | B'0 | R/W | Serial Port CTS# Pin Input/output Specifies input or output for the serial port CTS# pin. To actually set the CTS# pin as a port output pin to output the value set by the CTSdT bit, the MCE bit in SCFCR should be cleared to 0. 0: Indicates that the CTSdT bit value is not output to the CTS# pin. 1: Indicates that the CTSdT bit value is output to the CTS# pin. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 4 | CTSDT | — | R/W | <p>Serial Port CTS# Pin Data</p> <p>Specifies the input/output data level of the serial port CTS# pin. Whether the pin is set for input or output is determined by the CTSIO bit. When the pin is set for output, the value of the CTSDT bit is output to the CTS# pin. Regardless of the value of the CTSIO bit, the value of the CTS# pin is read from the CTSDT bit.</p> <p>The initial value of this bit is undefined after a power-on.</p> <p>0: Indicates that the input/output data is low level.</p> <p>1: Indicates that the input/output data is high level.</p> |
| 3 | SCKIO | B'0 | R/W | <p>Serial Port Clock Pin Input/output</p> <p>Specifies input or output for the serial port SCK pin. To actually set the SCK pin as a port output pin to output the value set by the SCKDT bit, the CKE [1:0] bits in SCSCR should be cleared to 0.</p> <p>0: Indicates that the SCKDT bit value is not output to the SCK pin.</p> <p>1: Indicates that the SCKDT bit value is output to the SCK pin.</p> |
| 2 | SCKDT | — | R/W | <p>Serial Port Clock Pin Data</p> <p>Specifies the input/output data level of the serial port SCK pin. Whether the pin is set for input or output is determined by the SCKIO bit. When the pin is set for output, the value of the SCKDT bit is output to the SCK pin. Regardless of the value of the SCKIO bit, the value of the SCK pin is read from the SCKDT bit.</p> <p>The initial value of this bit is undefined after a power-on reset.</p> <p>0: Indicates that the input/output data is low level.</p> <p>1: Indicates that the input/output data is high level.</p> |
| 1 | SPB2IO | B'0 | R/W | <p>Serial Port Break Input/Output</p> <p>Specifies the output condition of the serial port TX pin. To actually set the TX pin as a port output pin to output the value set by the SPB2DT bit, the TE bit in SCSCR should be cleared to 0.</p> <p>0: Indicates that the SPB2DT bit value is not output to the TX pin.</p> <p>1: Indicates that the SPB2DT bit value is output to the TX pin.</p> |
| 0 | SPB2DT | — | R/W | <p>Serial Port Break Data</p> <p>Specifies the input level of the serial port RX pin and the output level of the TX pin. The TX pin output conditions are determined by the SPB2IO bit. When the TX pin is set for output, the value of the SPB2DT bit is output to the TX pin. Regardless of the value of the SPB2IO bit, the value of the RX pin is read from the SPB2DT bit.</p> <p>The initial value of this bit is undefined after a power-on reset.</p> <p>0: Indicates that the input/output data is low level.</p> <p>1: Indicates that the input/output data is high level.</p> |

Figures 50.2 to 50.6 show the block diagrams of the SCIF I/O ports.

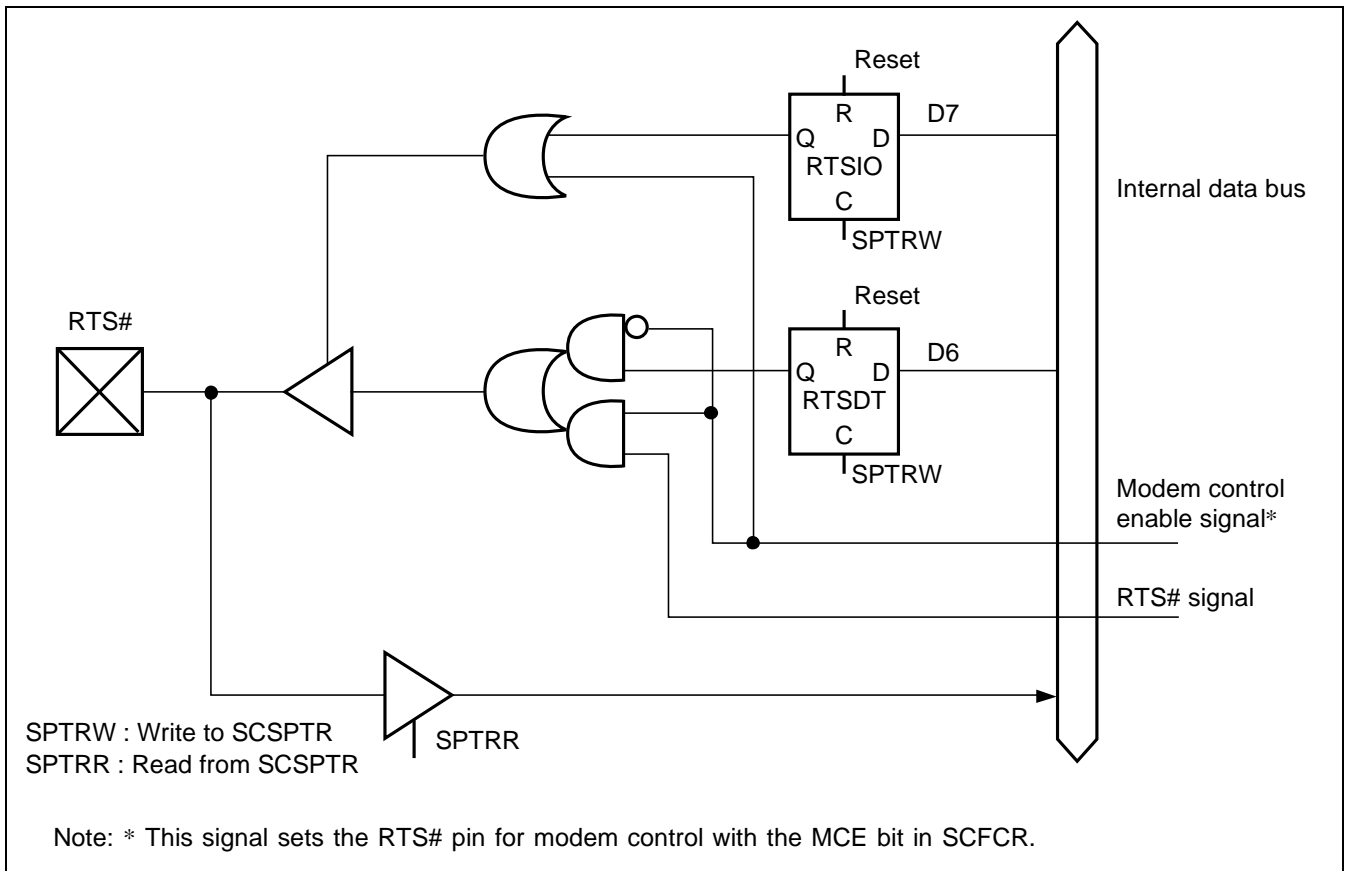


Figure 50.2 RTS# Pin

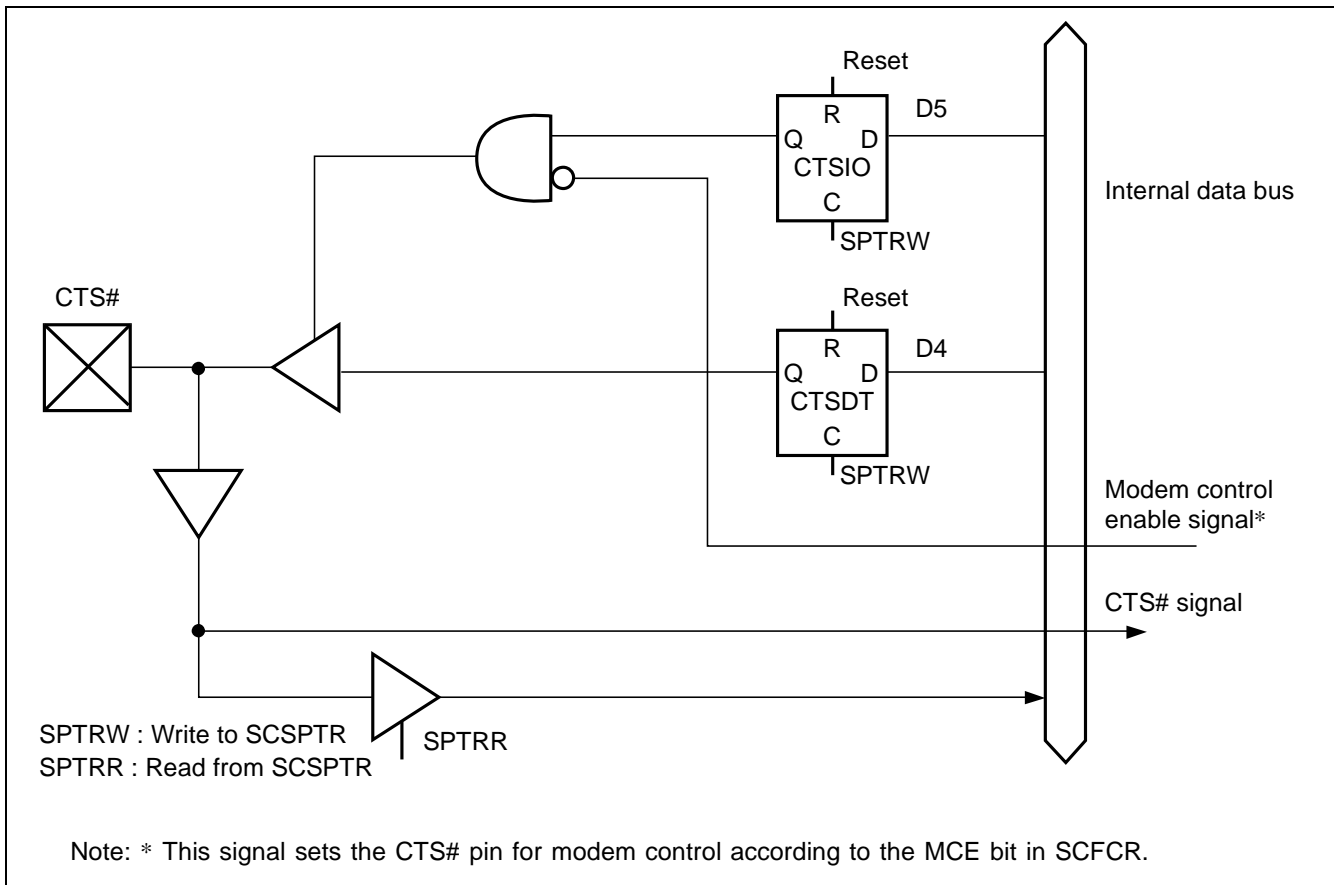


Figure 50.3 CTS# Pin

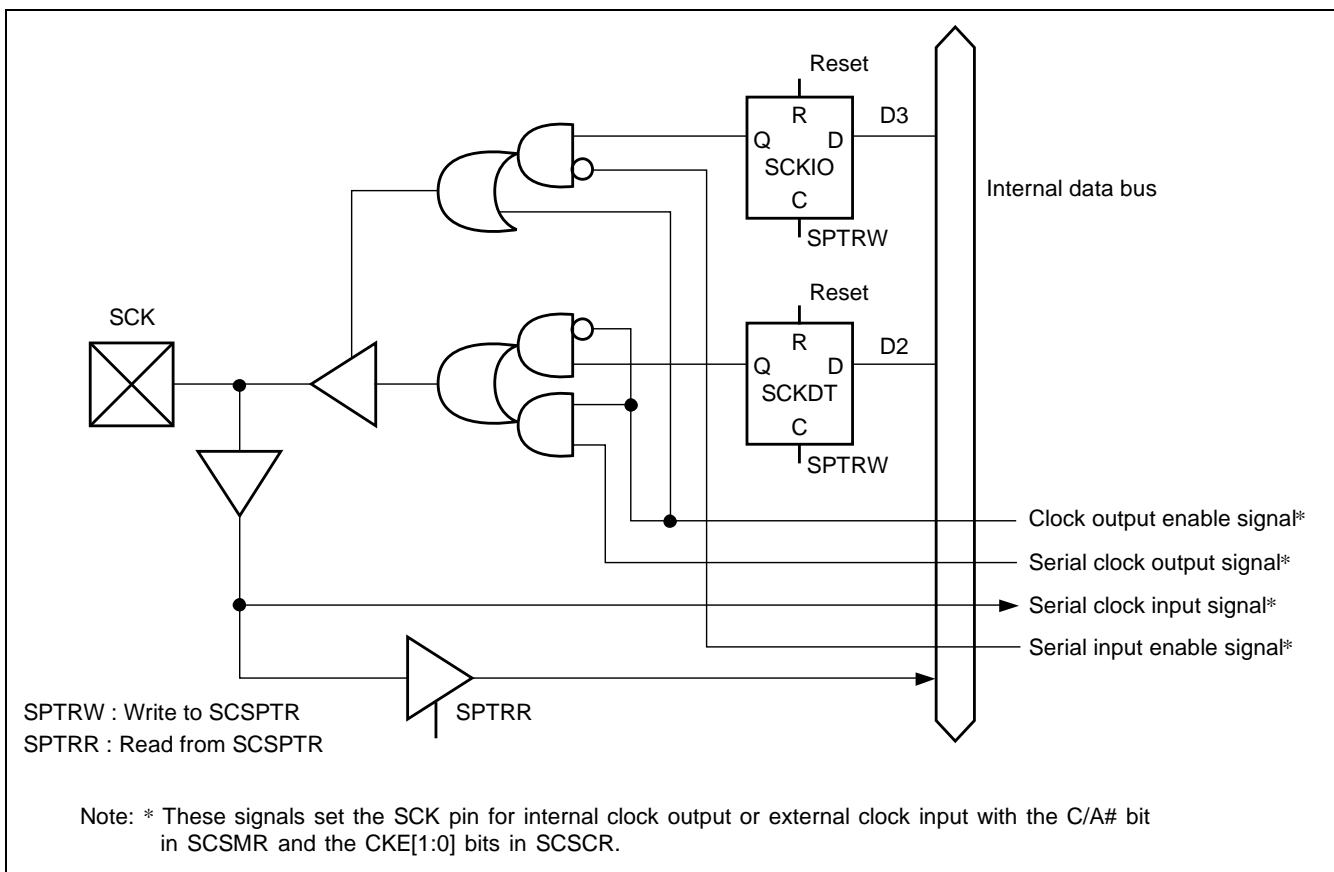


Figure 50.4 SCK Pin

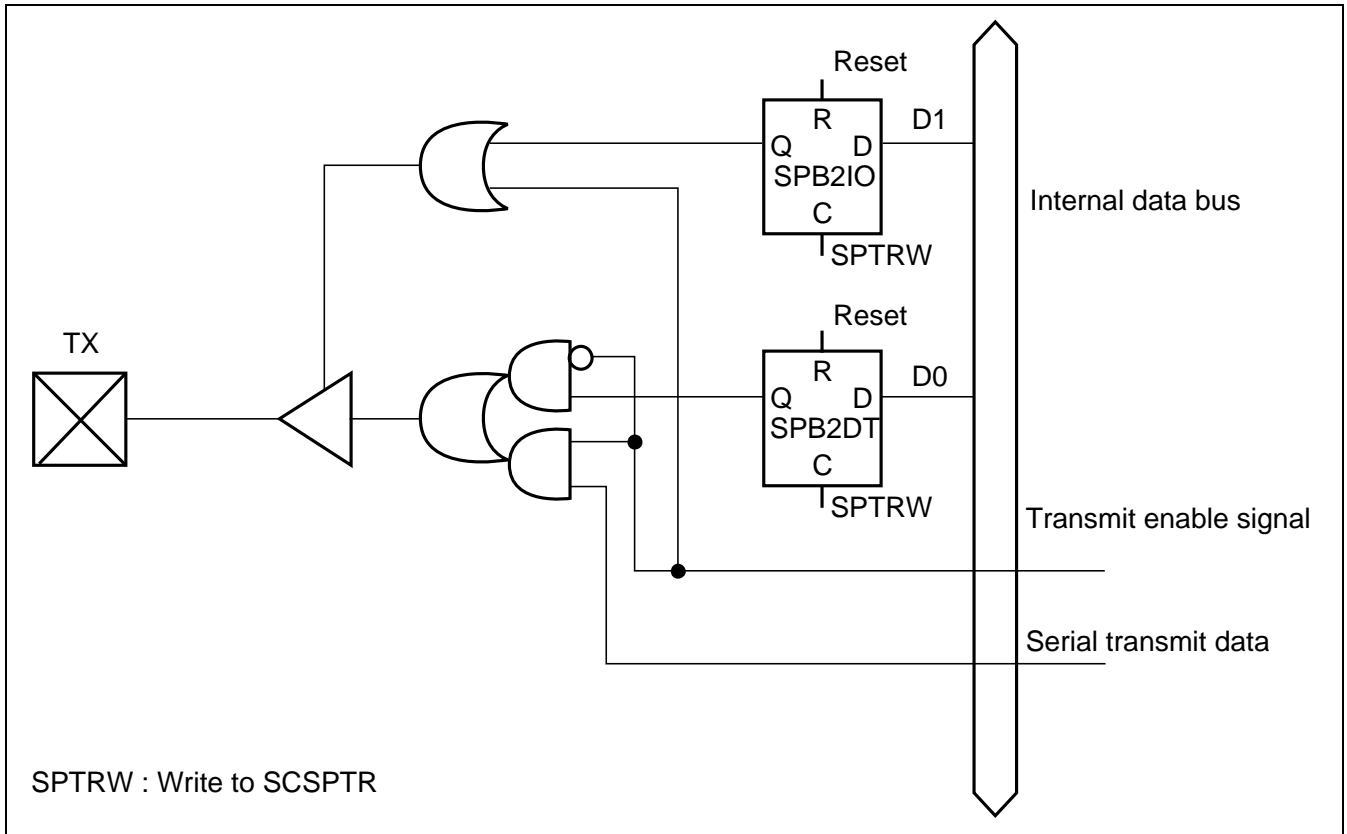


Figure 50.5 TX Pin

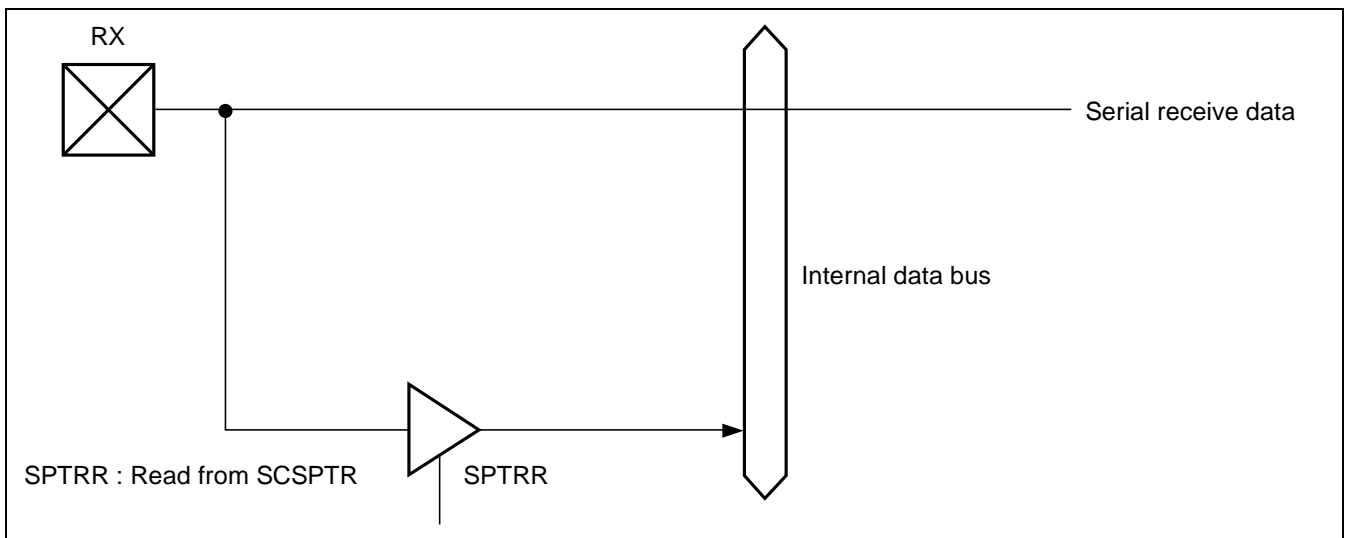


Figure 50.6 RX Pin

50.2.12 Line Status Register (SCLSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|---|---|---|-------|---|-------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | TO | — | ORER |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/WC0 | R | R/WC0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-------|--|
| 15 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | TO | B'0 | R/WC0 | <p>Timeout</p> <p>Indicates that the number of data bytes in SCFRDR is less than the receive trigger count, and that no further data has been received for at least 15 etu after the stop bit of the last receive data in asynchronous mode. This bit is not set in clock synchronous mode.</p> <p>0: Indicates that data is being received or has been successfully received and that there is no receive data in SCFRDR.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> A power-on reset is executed. All the receive data in the SCFRDR has been read, and 0 is written to TO. <p>1: Indicates that no further receive data has been received (receive timeout).</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> The number of data bytes in SCFRDR is less than the receive trigger count and no further data has been received for at least 15 etu after the stop bit of the last receive data.* <p>Note: etu: Elementary Time Unit (time for transfer of 1 bit) Equivalent to 1.5 frames with an 8-bit, 1-stop-bit format.</p> |
| 1 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 0 | ORER | B'0 | R/WC0 | <p>Overrun Error</p> <p>Indicates that an overrun error has occurred in reception and abnormal termination is caused.</p> <p>If an overrun error occurs, the receive data prior to the overrun error is retained in SCFRDR and the data received subsequently is discarded.</p> <p>Any subsequent serial reception is disabled while the ORER flag is 1.</p> <p>To resume data reception after clearing the ORER flag, be sure to first read (or clear) data in the receive FIFO and handle the error, then clear the ORER flag.</p> <p>0: Indicates that data is being received or has been successfully received.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • A power-on reset is executed. • 0 is written to ORER. <p>1: Indicates that an overrun error has occurred in reception.</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> • The next serial reception has been completed while SCFRDR is full of 16-byte data. <p>Note: When bit RE in SCSCR is cleared to 0, the ORER flag is not affected and its previous state is retained.</p> |

50.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

50.3.1 Operation in Asynchronous Mode

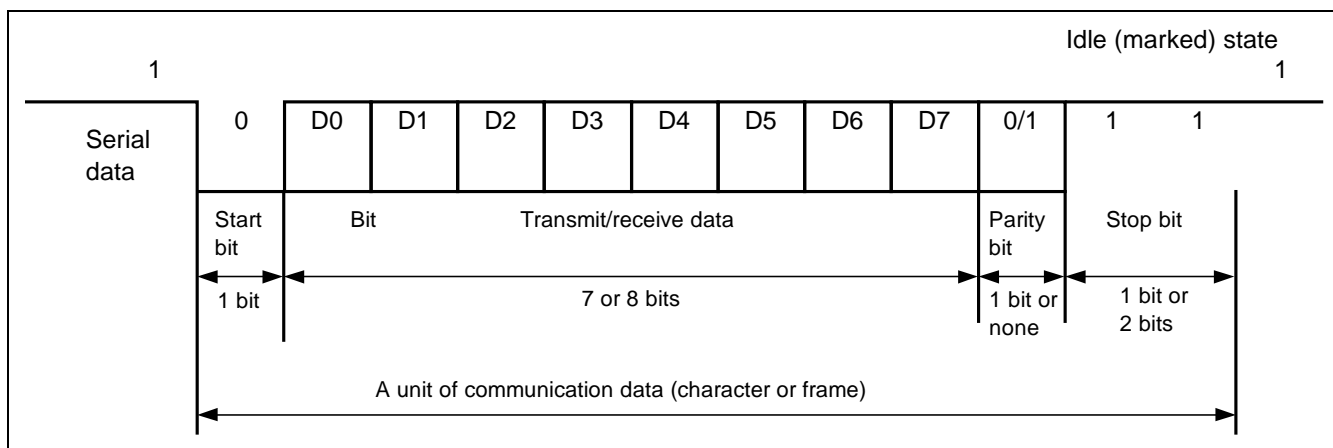
In asynchronous mode, the SCIF performs serial communication, in which data is transmitted/received in character units using the attached start bit indicating the start of communication and stop bit indicating the end of communication.

Figure 50.7 shows the general format for asynchronous serial communication.

In asynchronous serial communication, the transmission line is usually kept marked (high level). The SCIF monitors the transmission line, and when it finds a space (low level), it regards the space as a start bit and starts serial communication.

One character in serial communication consists of a start bit (low level), followed by transmit/receive data (LSB-first; from the lowest bit), a parity bit (high or low level), and a stop bit (high level).

During reception in asynchronous mode, the SCIF performs synchronization at the falling edge of the start bit. Communication data is acquired at the center of each bit because the SCIF samples data at the eighth pulse of the clock which has a frequency of 16 times the bit rate.



**Figure 50.7 Data Format in Asynchronous Mode
(Example of 8-Bit Data with Parity and Two Stop Bits)**

(1) Transmission/Reception Format

Table 50.8 shows available data transfer formats. The SCIF supports 8 transfer formats, which can be specified by SCSMR.

Table 50.8 Serial Transmission/Reception Formats (Asynchronous Mode)

| SCSMR Settings | | | Serial Transmission/Reception Format and Frame Length | | | | | | | | | | | |
|----------------|----|------|---|---|---|---|---|---|---|------|------|------|------|----|
| CHR | PE | STOP | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 0 | 0 | 0 | S 8-bit data | | | | | | | | STOP | | | |
| | | 1 | S 8-bit data | | | | | | | | STOP | STOP | | |
| | 1 | 0 | S 8-bit data | | | | | | | | P | STOP | | |
| | | 1 | S 8-bit data | | | | | | | | P | STOP | STOP | |
| 1 | 0 | 0 | S 7-bit data | | | | | | | STOP | | | | |
| | | 1 | S 7-bit data | | | | | | | STOP | STOP | | | |
| | 1 | 0 | S 7-bit data | | | | | | | P | STOP | | | |
| | | 1 | S 7-bit data | | | | | | | P | STOP | STOP | | |

[Legend]

- S: Start bit
- STOP: Stop bit
- P: Parity bit

(2) Clock

The transfer clock can be selected from the following two clocks using the CKE [1:0] bits in SCSCR:

- Internal clock generated by the on-chip baud rate generator
- External clock generated by an external clock baud rate generator

(3) Data Transmission/Reception

(a) Initialization of SCIF (Asynchronous Mode)

Before transmitting/receiving data or changing the operating mode or communication format, the SCIF should be initialized using the sample flowchart for SCIF initialization shown in Figure 50.8.

[Notes]

Clearing the TE bit to 0 initializes SCTSR. However, SCFSR, SCFTDR, and SCFRDR contents are retained even if the TE and RE bits are cleared to 0.

The TE bit should be cleared to 0 after all transmit data has been sent and the TEND flag has been set in SCFSR. The TE bit can be cleared to 0 during transmission, but the data being transmitted will enter the marked state after clearing. In addition, before setting the TE bit to 1 to restart the transmission, set the TFRST bit to 1 in SCFCR to reset SCFTDR.

When an external clock is used, do not stop the clock during operation or initialization. If stopped, the operation will be unreliable. Furthermore, when the baud rate generator for external clock is also to be used, be sure to make settings for it before starting initialization of the SCIF.

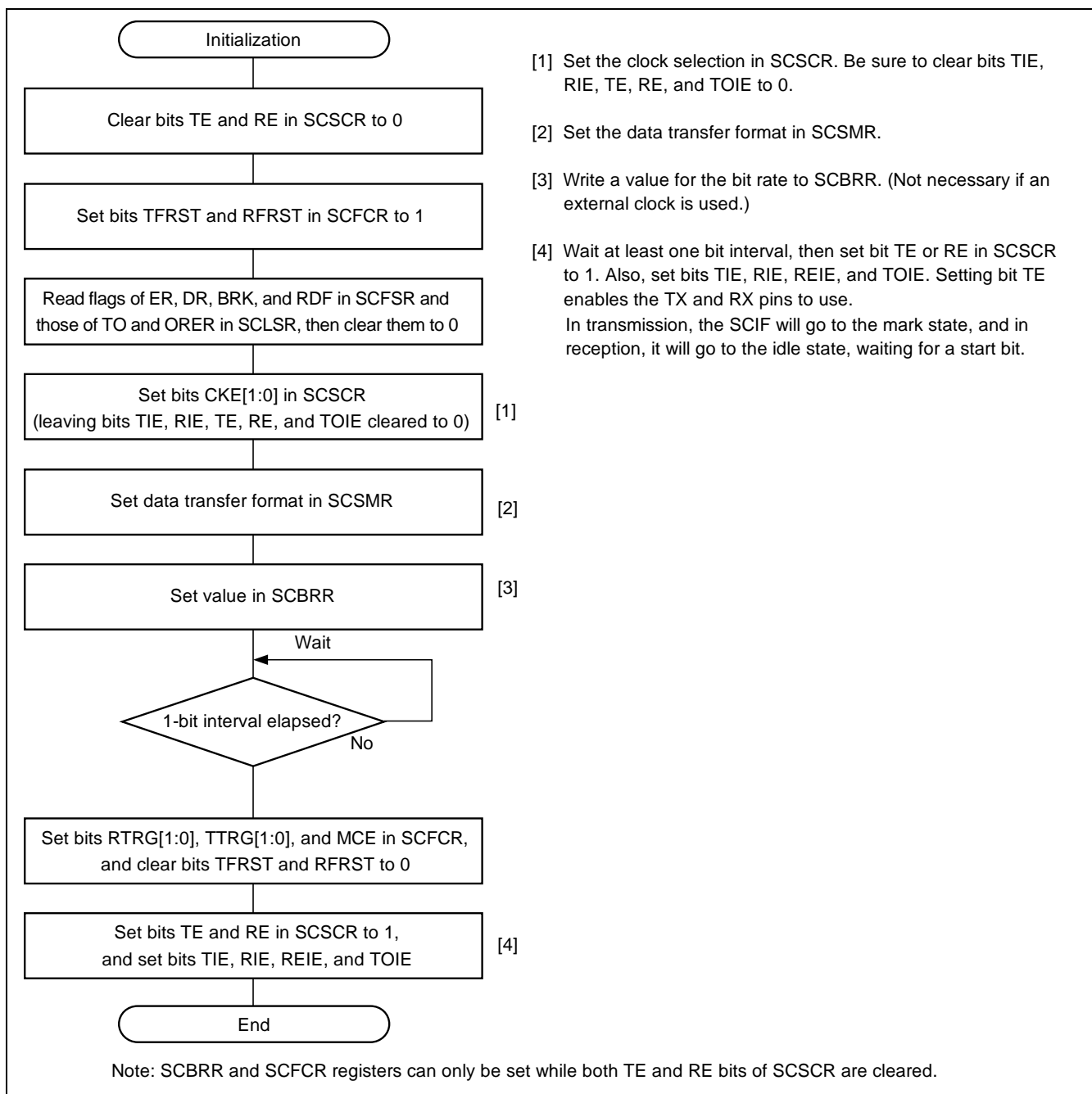


Figure 50.8 Sample Flowchart for Initializing the SCIF

(b) Serial Data Transmission (Asynchronous Mode)

Figure 50.9 shows a sample flowchart for serial transmission.

After the SCIF transmission operation is enabled, serial data transmission can be performed using the following procedure:

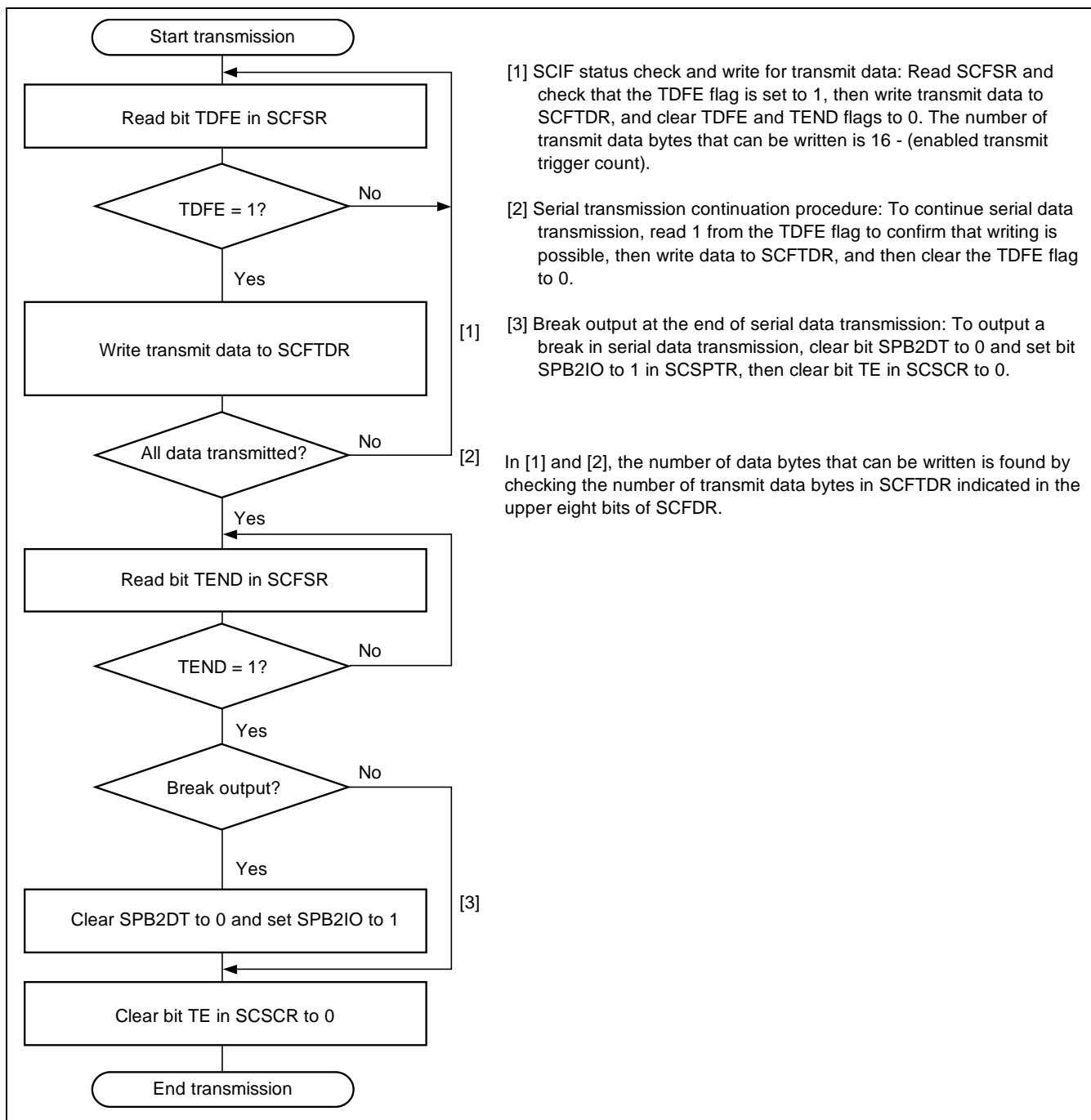


Figure 50.9 Sample Flowchart for Serial Transmission

In serial transmission, the SCIF operates as follows:

1. When data is written to SCFTDR, the SCIF transfers the data from SCFTDR to SCTSR and starts transmitting. Confirm that the TDFE flag in SCFSR is set to 1 before writing transmit data to SCFTDR. The number of data bytes that can be written is at least (16- (transmit trigger count)).
2. When data is transferred from SCFTDR to SCTSR and the SCIF starts transmission, consecutive transmission is performed until there is no transmit data left in SCFTDR. When the number of transmit data bytes in SCFTDR is equal to or less than the transmit trigger count specified in SCFCR, the TDFE flag is set. If the TIE and TEIE bits in SCSCR are set to 1 and 0, respectively at this time, a transmit-FIFO-data-empty interrupt (TDFE) request occurs. The serial transmit data is sent from the TX pin in the following order:
 - A. Start bit: One 0-bit is output.
 - B. Transmit data: 8-bit or 7-bit data is output in LSB-first order.
 - C. Parity bit: One parity bit (even or odd parity) is output.
A format that does not output a parity bit can also be selected.
 - D. Stop bit(s): One or two stop bits are output.
 - E. Marked state: 1 is output continuously until the start bit that starts the next transmission is sent.
3. The SCIF checks transmit data in SCFTDR when sending the stop bit. If there is data in it, the SCIF transfers the data from SCFTDR to SCTSR, sends the stop bit, and then starts serial transmission of the next frame. If there is no transmit data, the SCIF sets the TEND flag to 1 in SCFSR and sends out the stop bit, and then the marked state is entered to output 1 continuously. At this time, if the TIE and TEIE bits in SCSCR are set to 1, a transmit-end interrupt (TEND) request occurs.

Figure 50.10 shows an example of transmission in asynchronous mode.

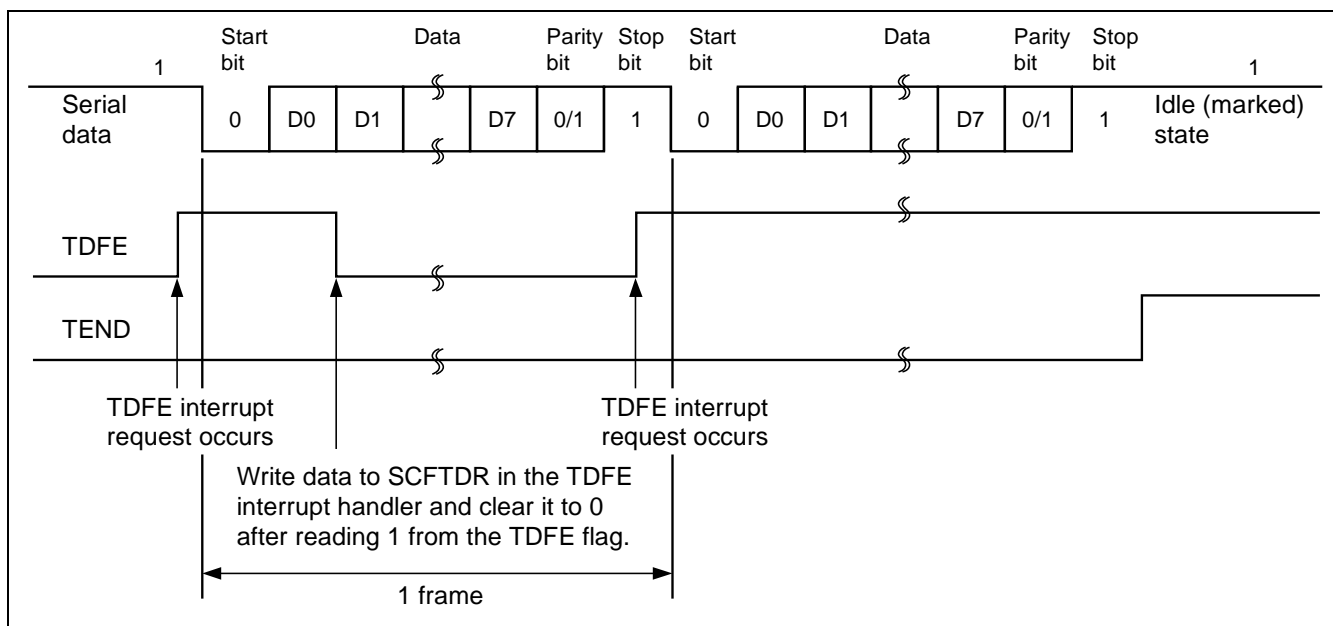


Figure 50.10 Sample SCIF Transmission Operation (Example of 8-Bit Data with Parity and One Stop Bit)

4. When modem control is enabled, transmission can be stopped or resumed in accordance with the CTS# input value. When CTS# is set to 1 during transmission, the marked state is entered after one frame of data transmission is ended. Setting CTS# to 0 restarts outputting the next transmit data from the start bit. Figure 50.11 shows an example of the operation with modem control enabled.

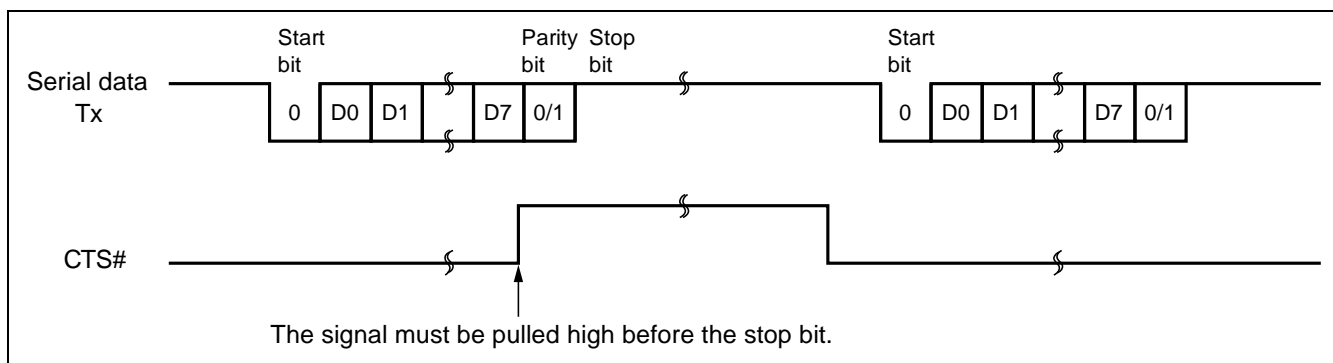


Figure 50.11 Sample Operation with Modem Control Enabled (CTS#)

(c) Serial Data Reception (Asynchronous Mode)

Figures 50.12 and 50.13 show sample flowcharts for serial reception.

After the SCIF reception operation is enabled, serial data reception can be performed using the following procedure:

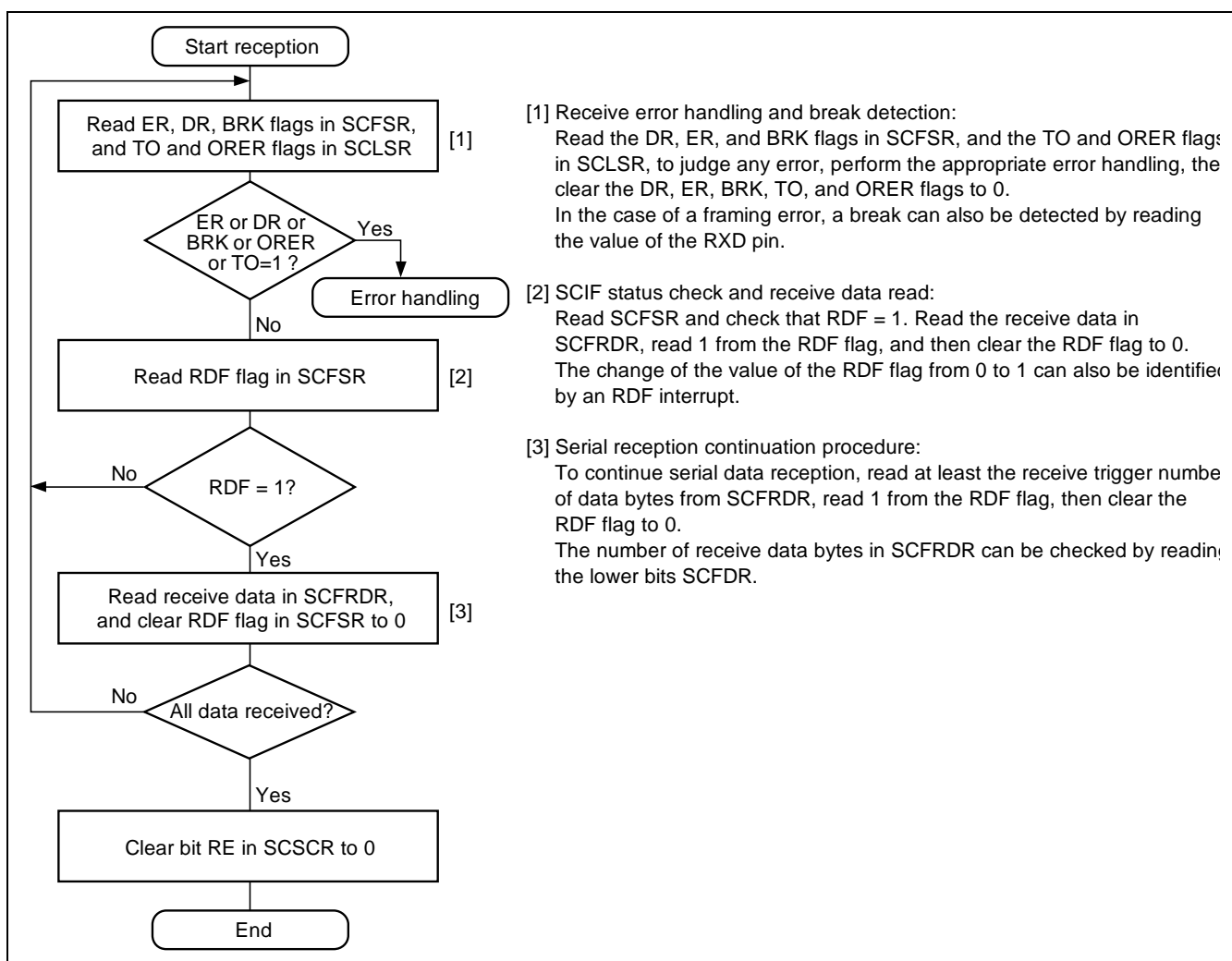
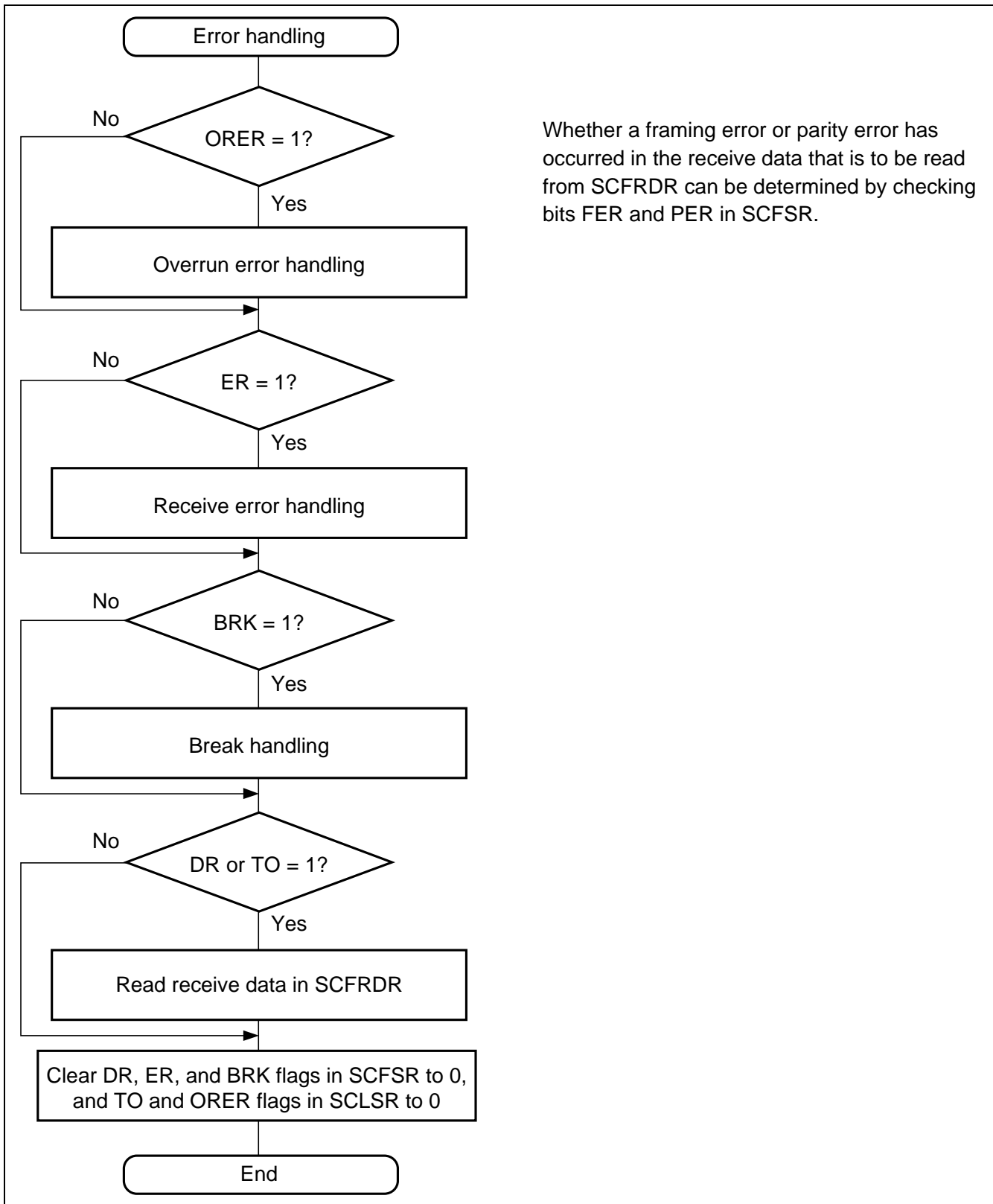


Figure 50.12 Sample Flowchart for Serial Reception (1)



Whether a framing error or parity error has occurred in the receive data that is to be read from SCFRDR can be determined by checking bits FER and PER in SCFSR.

Figure 50.13 Sample Flowchart for Serial Reception (2)

In serial reception, the SCIF operates as follows:

1. The SCIF monitors the transmission line, and when detecting the start bit 0, it performs internal synchronization and starts reception.
2. The SCIF stores the received data in SCRSR in LSB-to-MSB order.
3. The SCIF receives the parity bit and stop bit.

After receiving these bits, the SCIF performs the following checks:

- A. Stop bit: The SCIF checks whether the stop bit is 1.
If there are two stop bits, it checks only the first stop bit.
- B. Receive data: The SCIF checks that receive data can be transferred from the receive shift register (SCRSR) to SCFRDR.
- C. Overrun error: The SCIF checks that the ORER flag is 0, indicating that no overrun error has occurred.
- D. Break state: The SCIF checks that the BRK flag is 0, indicating that the break state is not set.

If the SCIF can confirm the conditions of (b), (c), and (d), it stores the receive data in SCFRDR.

Note: The SCIF continues to receive data even when a parity error or a framing error occurs.

4. If the RDF flag changes to 1 while the RIE bit in SCSCR is 1, a receive-FIFO-data-full interrupt (RDF) request occurs.

If the DR flag changes to 1 while the RIE bit in SCSCR is 1, a receive-data-ready interrupt (DR) request occurs.

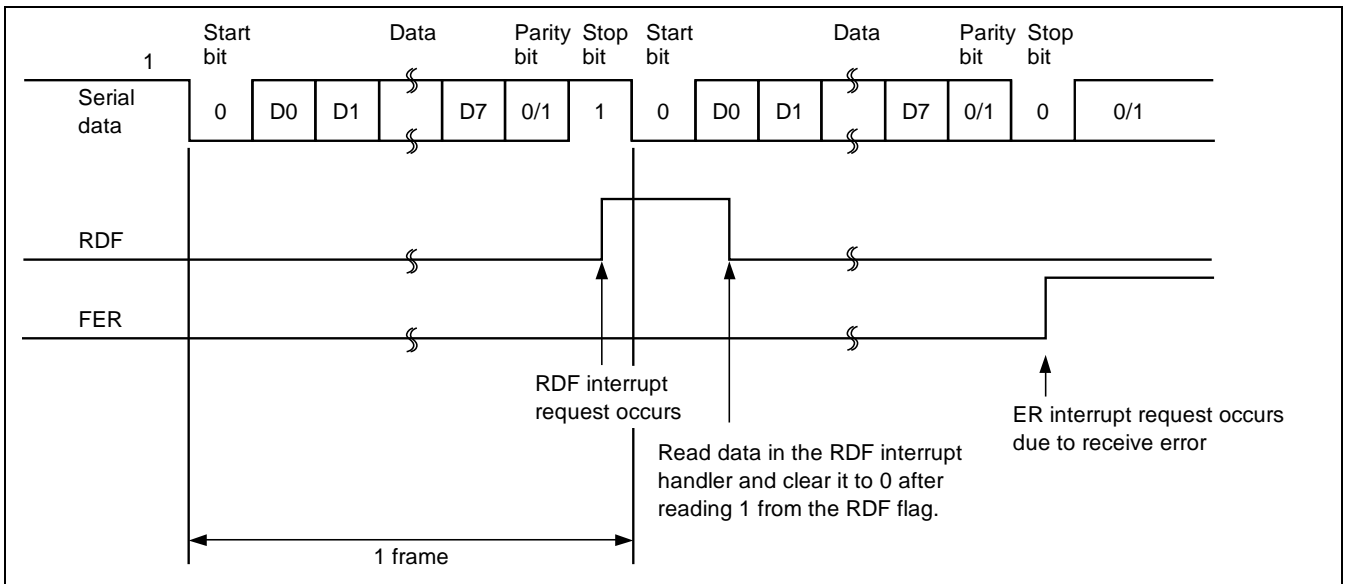
If the TO flag changes to 1 while the TOIE bit in SCSCR is 1, a timeout interrupt (TO) request occurs.

If the ER flag changes to 1 while the RIE or REIE bit in SCSCR is 1, a receive-error interrupt (ER) request occurs.

If the BRK flag changes to 1 while the RIE or REIE bit in SCSCR is 1, a break interrupt (BRK) request occurs.

If the ORER flag changes to 1 while the RIE or REIE bit in SCSCR is 1, an overrun-error interrupt (ORER) request occurs.

Figure 50.14 shows an example of reception in asynchronous mode.



**Figure 50.14 Sample SCIF Receive Operation
(Example of 8-Bit Data with Parity and One Stop Bit)**

5. When modem control is enabled, the RTS# signal is output when SCFRDR is empty. When RTS# is 0, data can be received. When RTS# is set to 1, the number of data bytes in SCFRDR is equal to or more than the RTS# output active trigger count.

Figure 50.15 shows an example of the operation with modem control enabled.

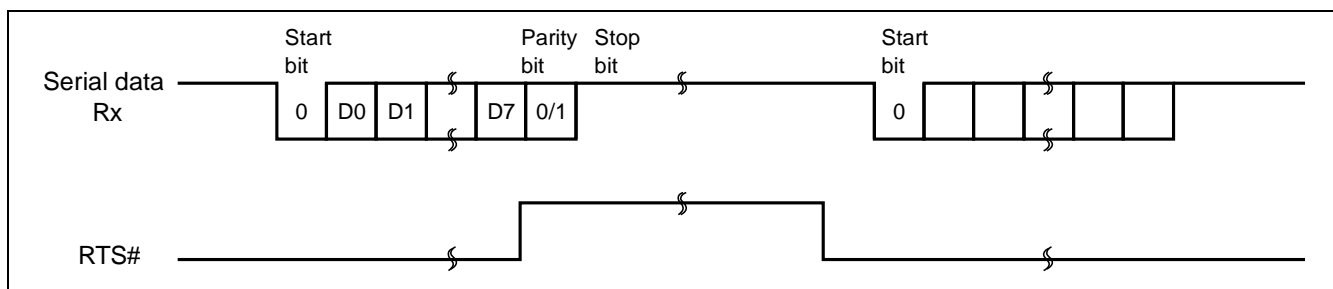


Figure 50.15 Example of the Operation with Modem Control Enabled (RTS#)

50.3.2 Operation in Clock Synchronous Mode

Clock synchronous mode, in which data is transmitted and received in synchronization with clock pulses, is suitable for fast serial communication.

Figure 50.16 shows the general format for clock synchronous serial communication. In the clock synchronous serial communication, data on the communication line is output between one falling edge of the synchronization clock and the next falling edge. The data decision is done at the rising edge of the clock.

In serial communication, each character is output starting with the LSB and ending with the MSB. After the MSB is output, the communication line retains the state of the last data.

In clock synchronous mode, the SCIF receives data in synchronization with a rising edge of the synchronization clock.

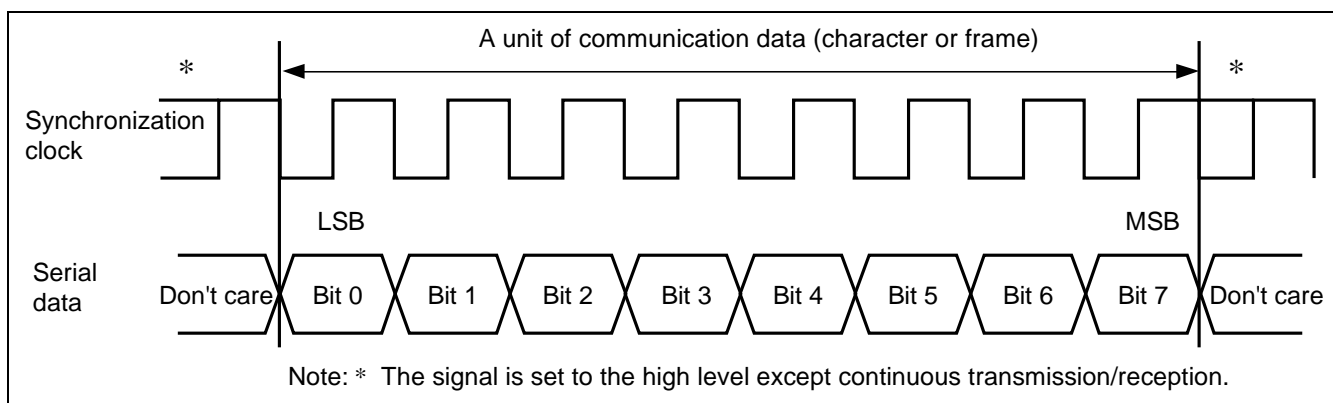


Figure 50.16 Data Format for Clock Synchronous Communication

(1) Data Transfer Format

The data transfer format is fixed to 8 bits. No parity bit can be added.

(2) Clock

The clock can be selected from the following two clocks using the C/A# bit in SCSMR and the CKE [1:0] bits in SCSCR:

- Internal clock generated by the on-chip baud rate generator
- External synchronization clock input at the SCK pin

When the SCIF is operated on an internal clock, the synchronization clock is output from the SCK pin. Eight synchronization clock pulses are output in one-character transfer, and when no transfer is performed the clock is fixed high. When only reception operation is performed, selecting an internal clock outputs clock pulses until the number of data bytes in the receive FIFO reaches the specified receive trigger count, while the RE bit in SCSCR is 1.

(3) Data Transmission/Reception

(a) Initialization of SCIF (Synchronous Mode)

Before transmitting/receiving data or changing the operating mode or communication format, the TE and RE bits in SCSCR to 0 and then the SCIF should be initialized using the sample flowchart for SCIF initialization shown in Figure 50.17.

[Note]

Clearing the TE bit to 0 initializes SCTSR. However, clearing the RE bit to 0 does not affect the RDF, PER, FER, and ORER flags and SCFRDR contents.

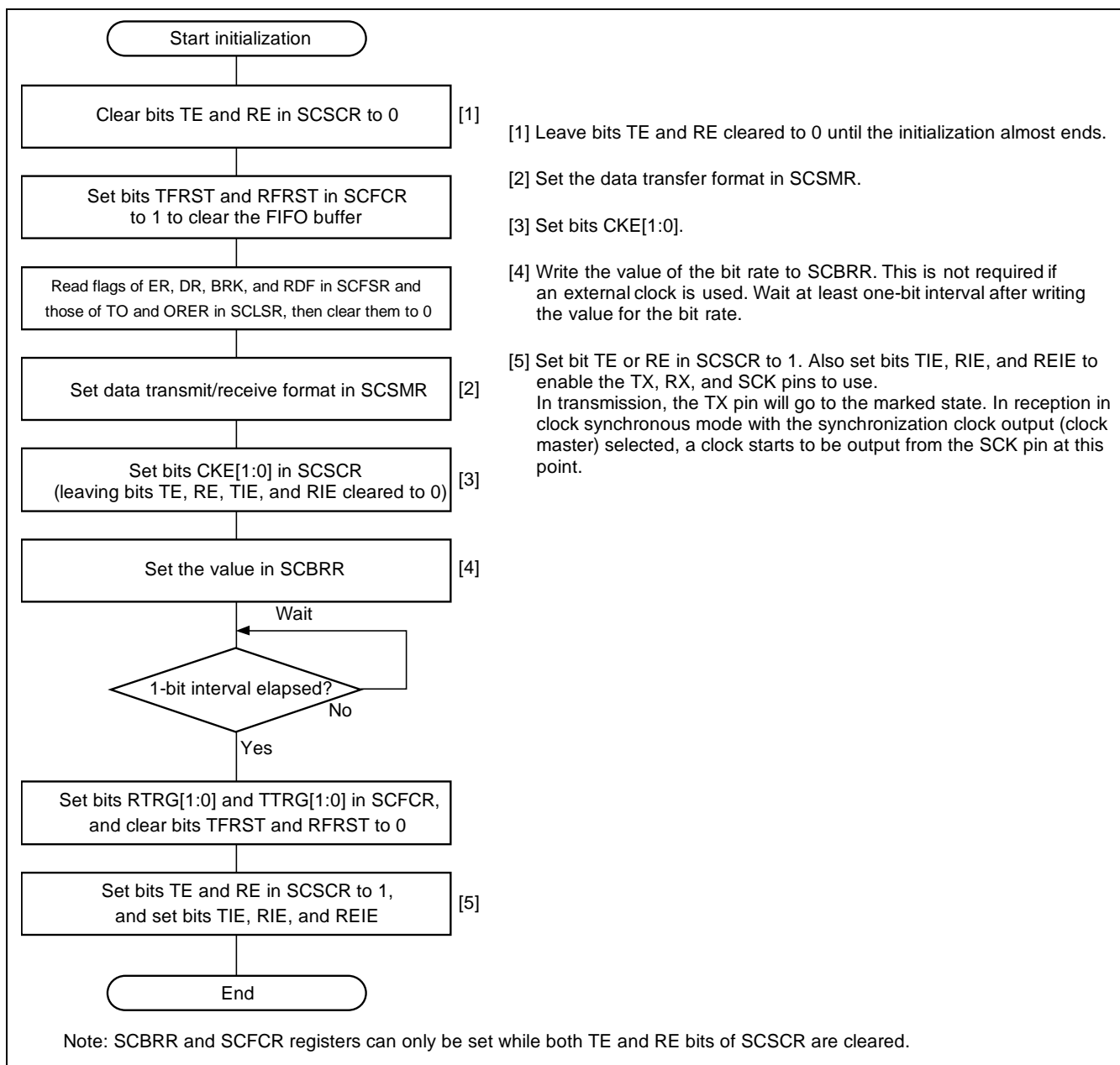


Figure 50.17 Sample Flowchart for SCIF Initialization

(b) Serial Data Transmission (Clock Synchronous mode)

Figure 50.18 shows a sample flowchart for serial transmission.

After the SCIF transmission operation is enabled, serial data transmission can be performed using the following procedure:

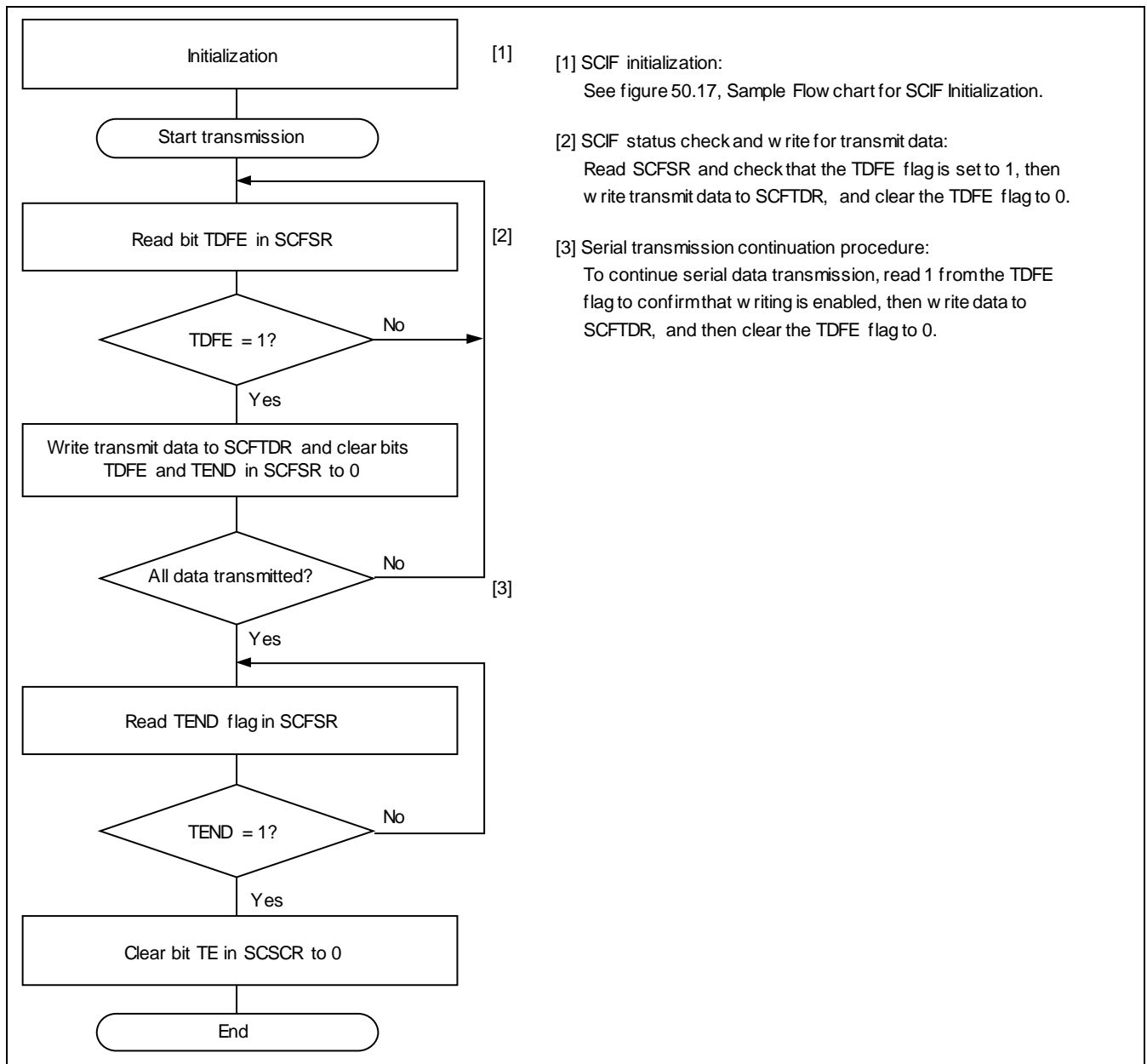


Figure 50.18 Sample Flowchart for Serial Transmission

In serial data transmission, the SCIF operates as described below:

1. When data is written to SCFTDR, the SCIF transfers the data from SCFTDR to SCTSR to start transmission. Confirm that the TDFE flag in SCFSR is set to 1 before writing transmit data to SCFTDR. The number of data bytes that can be written is at least (16- (the transmit trigger count)).
2. After the SCIF transfers data from SCFTDR to SCTSR and starts data serial transmission, the SCIF consecutively transmits it until there is no transmit data left in SCFTDR. When the number of transmit data bytes in SCFTDR is equal to or less than the transmit trigger count set in SCFCR, the TDFE flag is set. If the TIE and TEIE bits in SCSCR are set to 1 and 0, respectively at this time, a transmit-FIFO-data-empty interrupt (TDFE) request occurs. When the external clock is selected, data is output in synchronization with the input clock. The serial transmit data is output at the TX pin in the LSB-first order.
3. The SCIF checks transmit data in SCFTDR when sending the last bit. If there is transmit data, the SCIF transfers the data to SCTSR, and starts serial transmission of the next frame. If there is no transmit data left, the TEND flag in SCFSR is set to 1, and the transmit data pin (TX pin) retains its state after the last bit is sent. At this time, if the TIE and TEIE bits in SCSCR are set to 1, a transmit-end interrupt (TEND) request occurs.
4. After serial transmission ends, the SCK pin is fixed high.

Note: In clock synchronous mode, when transmit data is written to SCFTDR by the DMAC, the TEND flag may not be cleared. Therefore, if the DMAC is used for transmission in clock synchronous mode, read the TEND flag in the following method.

1. Confirm that data transfer is completed in the DMAC.
2. Read the TEND flag.
3. Clear the TEND flag to 0 if TEND = 1.
4. Read the TEND flag again.
5. Use the second-read TEND flag.

Figure 50.19 shows an example of SCIF serial transmission in clock synchronous mode.

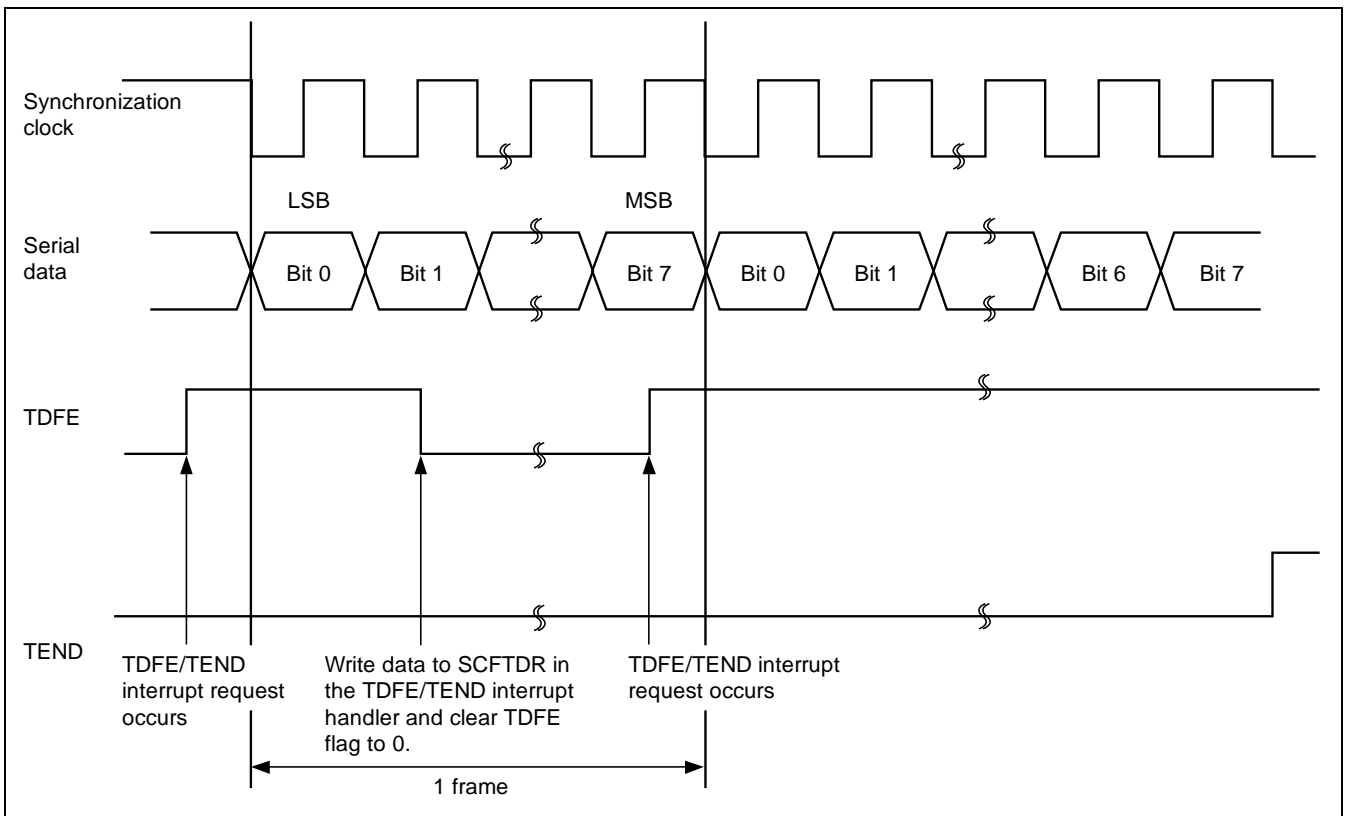


Figure 50.19 Example of SCIF Serial Transmission in Clock Synchronous Mode

(c) Serial Data Reception (Clock Synchronous Mode)

Figures 50.20 and 50.21 show sample flowcharts for serial reception.

The SCIF reception should be enabled before taking the following steps for serial data reception.

When switching operating mode from asynchronous mode to clock synchronous mode without initializing the SCIF, check that the ORER, PER3 to PER0, and FER3 to FER0 flags are cleared to 0.

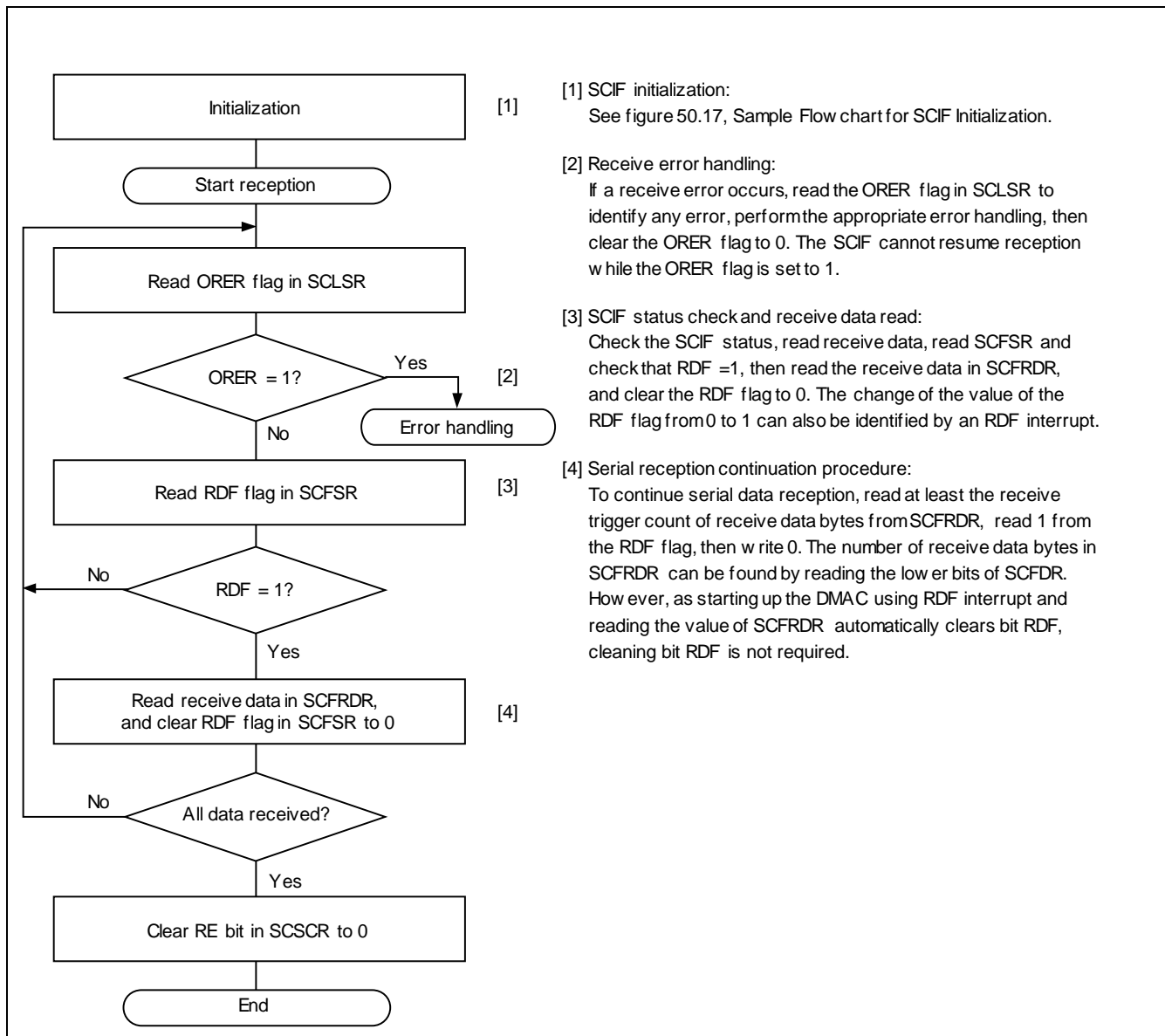


Figure 50.20 Sample Flowchart for Serial Data Reception

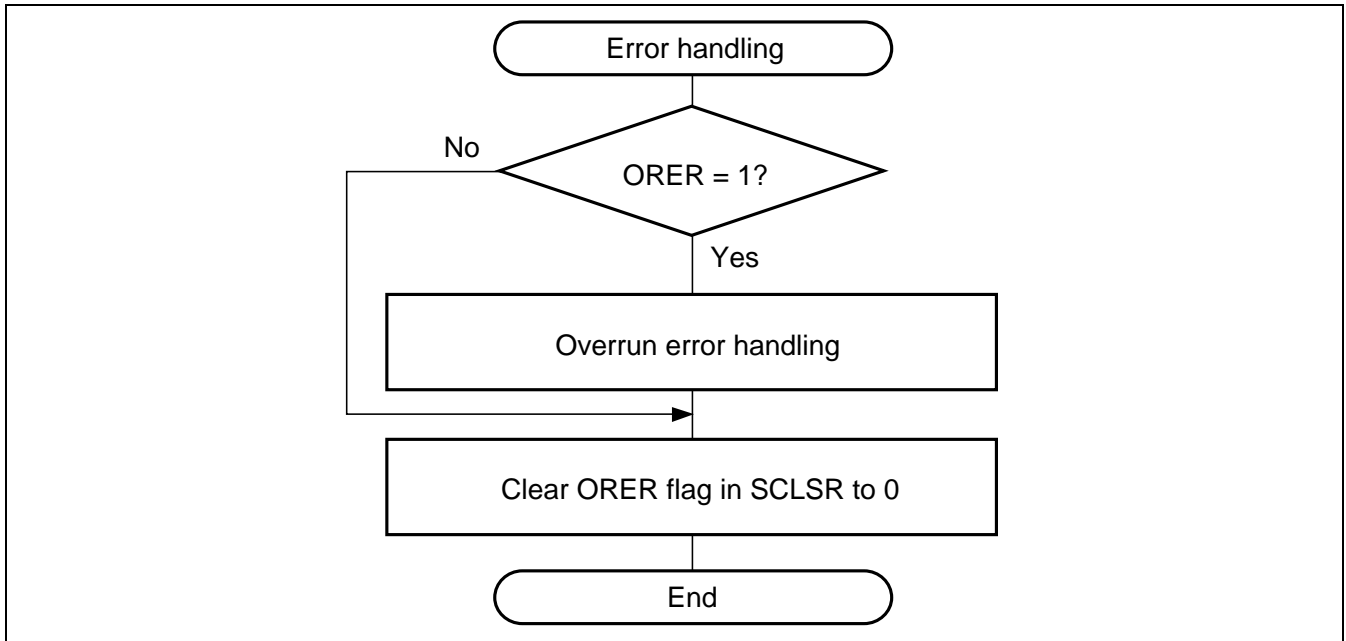


Figure 50.21 Sample Flowchart for Serial Data Reception

In serial data reception, the SCIF operates as described below:

1. The SCIF starts receiving data in synchronization with synchronization clock input or output.
2. The SCIF stores receive data in SCRSR in LSB-to-MSB order.
After receiving the data, the SCIF first checks if the receive data can be transferred from SCRSR to SCFRDR, then starts storing the receive data in SCFRDR.
If the SCIF detects an overrun error, the SCIF cannot receive subsequent data.
3. If the RDF flag changes to 1 while the RIE bit in SCSCR is 1, a receive-FIFO-data-full interrupt (RDF) request occurs. If the ORER flag changes to 1 while the RIE or REIE bit in SCSCR is 1, a break interrupt (BRI) request occurs.

Figure 50.22 shows an example of SCIF reception in clock synchronous mode.

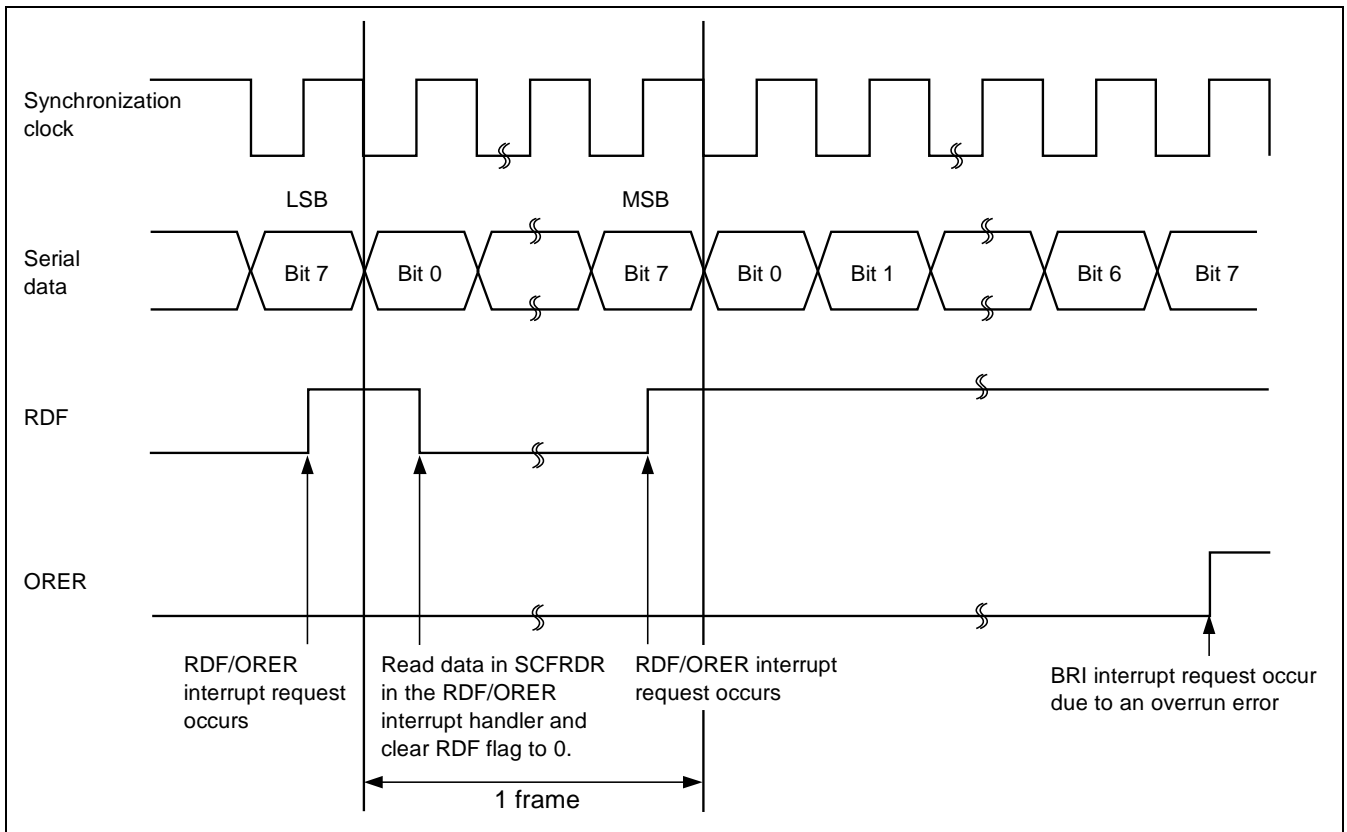


Figure 50.22 Example of SCIF Reception in Clock Synchronous Mode

(d) Simultaneous Serial Data Transmission/Reception (Clock Synchronous Mode)

Figure 50.23 shows a sample flowchart for simultaneous serial data transmission/reception.

The SCIF transmission and reception should be enabled before taking the following steps for simultaneous serial data transmission/reception.

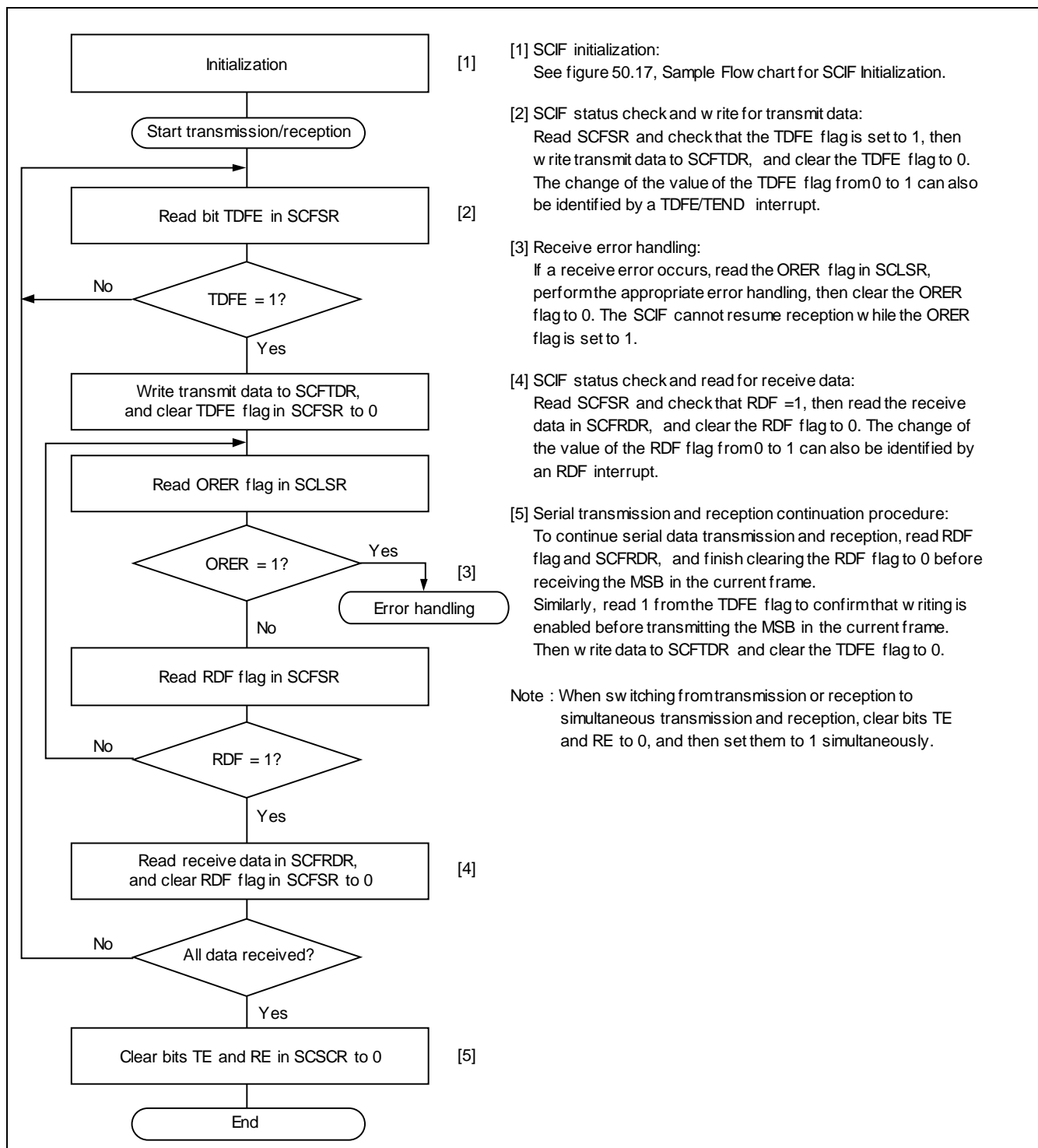


Figure 50.23 Sample Flowchart for Simultaneous Serial Data Transmission/Reception

50.3.3 Baud Rate Generator for External Clock (BRG)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

(1) Overview

The SCIF incorporates a baud rate generator for external clock (abbreviated as BRG, hereafter). The BRG supplies a sampling clock (BRGCLK) to the SCIF core by dividing the external clock SC_CLK (selectable between SCIF_CLK and SCKi) by 1 to $2^{16} - 1$. In addition, the BRG switches the output between the external clock SCK and divided clock.

Figure 50.24 shows a block diagram of the BRG.

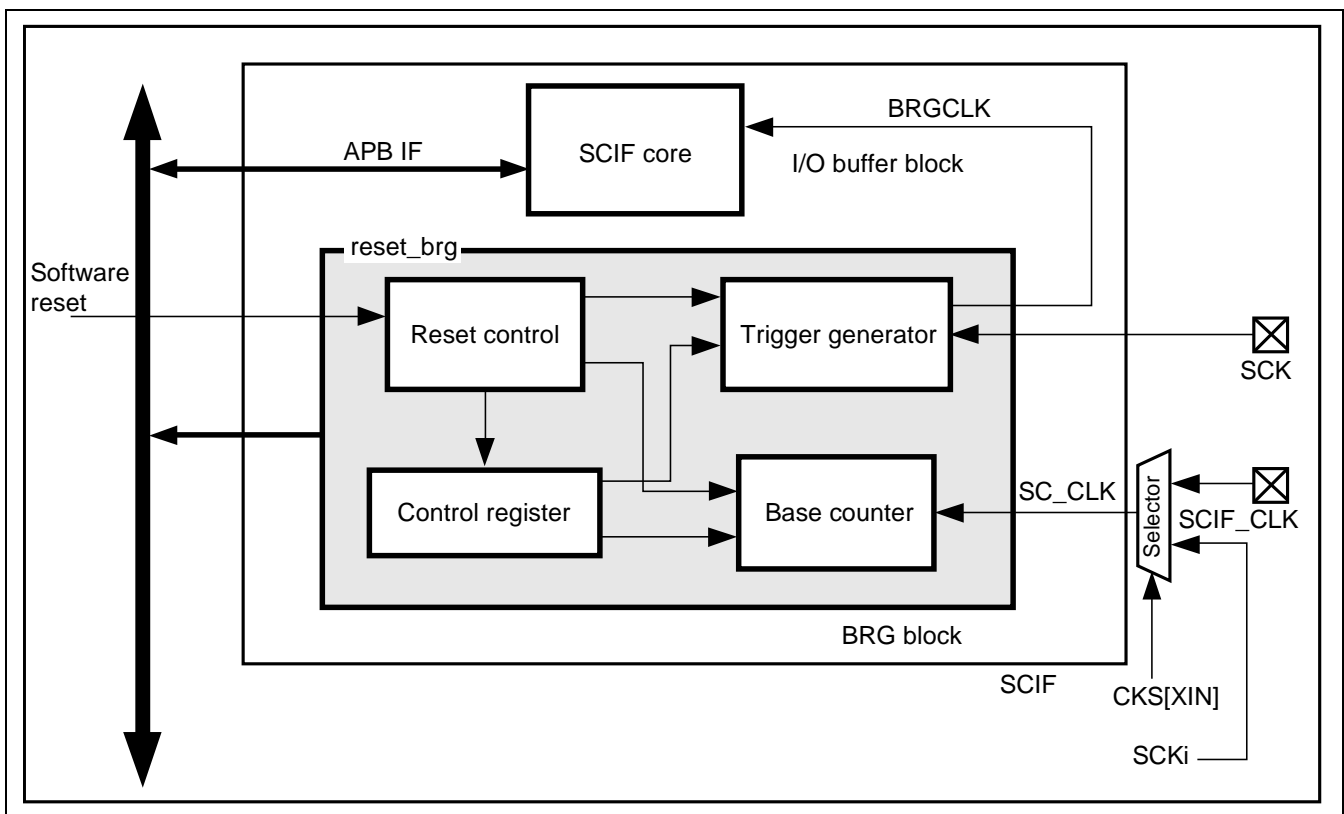


Figure 50.24 BRG Block Diagram

(a) Reset Controller:

The reset controller handles resetting the control register, base counter, and trigger generator.

(b) Control Register:

The control register has the frequency division register and clock select register.

(c) Base Counter:

The base counter is a 16-bit CLK synchronization counter that is used to determine the timing for generating a frequency divided clock.

(d) Trigger Generator:

The trigger generator generates rising-edge/falling-edge triggers for a frequency divided clock with the timing according to values of the frequency division register and base counter. The triggers are used to generate the frequency divided clock. In addition, the trigger generator switches the output between the SCK (external clock input) and frequency divided clock.

(2) Register Configuration

Table 50.9 shows the registers in the BRG block.

Table 50.9 List of Registers

| Name | Abbreviation | R/W | Initial Value | Relative Address | Access Size |
|-----------------------------|--------------|-----|---------------|------------------|-------------|
| Frequency division register | DL | R/W | H'0000 | H'30 | 16 |
| Clock select register | CKS | R/W | H'0000 | H'34 | 16 |

(3) Frequency Division Register (DL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register specifies the value of frequency division for the frequency divided clock generated by the BRG.

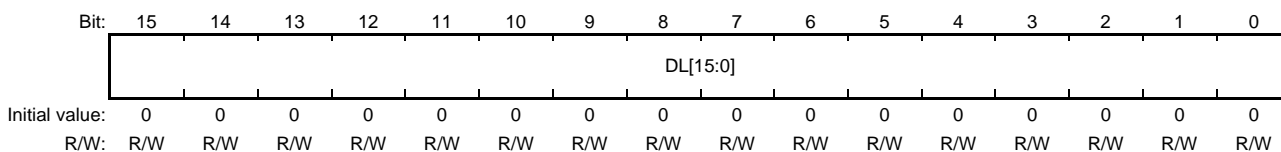
This register supports a 16-bit binary format that allows specifying a value in the range of 1 to 65535.

Setting all 0s in this register makes the BRG output the frequency divided clock at the low level.

The value of frequency division is given by the following formula:

$$\text{The value of frequency division} = (\text{clock input frequency}) / (\text{required baud rate} \times 16)$$

Table 50.10 shows how to use the baud rate generator with a 3.6864-MHz crystal resonator.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 15 to 0 | DL[15:0] | All 0 | R/W | Specifies a division value of frequency clock generated in BRG. The value settings enabled in the range of 1 to 65535. |

Table 50.10 Baud Rate (3.6864-MHz clock)

| Baud Rate | Value of Frequency Division | Error Rate (*) |
|-----------|-----------------------------|----------------|
| 50 | 4608 | — |
| 75 | 3072 | — |
| 110 | 2095 | -0.022 |
| 134.5 | 1713 | 0.001 |
| 150 | 1536 | — |
| 300 | 768 | — |
| 600 | 384 | — |
| 1200 | 192 | — |
| 1800 | 128 | — |
| 2000 | 115 | 0.174 |
| 2400 | 96 | — |
| 3600 | 64 | — |
| 4800 | 48 | — |
| 7200 | 32 | — |
| 9600 | 24 | — |
| 14400 | 16 | — |
| 19200 | 12 | — |
| 38400 | 6 | — |
| 76800 | 3 | — |
| 115200 | 2 | — |

Note: —: Indicates that the error rate is 0.

(4) Clock Select Register (CKS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register switches the output between the frequency divided clock and specifies a source clock for the external baud rate.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CKS | XIN | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 15 | CKS | B'0 | R/W | This bit switches the output between the frequency divided clock (SC_CLK) and external clock (SCK). 0: Selects the frequency divided clock. 1: Selects the external clock. |
| 14 | XIN | B'0 | R/W | Selects the baud rate generator clock source for the external clock between SCIF_CLK and SCKi. 0: Selects the external clock (SCIF_CLK). 1: Selects the internal clock (SCKi). |
| 13 to 0 | — | All 0 | R | Reserved These bits are always read as 0. Only 0 should be written to these bits. |

(5) Restrictions in BRG

Notes on Setting Frequency Division Register

- For the initial setting of the register after a reset, at least one bit of waiting period is required to secure the clock stabilization time.

(Example) One bit period when DL = 2

$$3.68 \text{ (MHz)} \times 1/2 \times 1/16 = 0.115 \text{ (MHz)} \rightarrow 8695 \text{ (ns)}$$

- For modifying the register value after the setting stated in <1> above, at least one bit of waiting period at the maximum bit rate (DL = '65535') is required.

The SCIF registers and BRG registers should be set as the following table:

- Asynchronous mode (SC_CLK external input)

| SCIF | | BRG | |
|----------------|---------------|---------------|------------------|
| Register Name | Setting Value | Register Name | Setting Value |
| SCSMR.C/A# | B'0 | CKS | H'0000 |
| SCSCR.CKE[1:0] | B'10 | DL | H'0001 to H'FFFF |

- Asynchronous mode (SCK external input)

| SCIF | | BRG | |
|----------------|---------------|---------------|---------------|
| Register Name | Setting Value | Register Name | Setting Value |
| SCSMR.C/A# | B'0 | CKS | H'8000 |
| SCSCR.CKE[1:0] | B'10 | DL | Don't care |

- Clock synchronous mode (external input)

| SCIF | | BRG | |
|----------------|---------------|---------------|---------------|
| Register Name | Setting Value | Register Name | Setting Value |
| SCSMR.C/A# | B'1 | CKS | H'8000 |
| SCSCR.CKE[1:0] | B'10 | DL | Don't care |

- Asynchronous mode (SC_CLK internal input)

| SCIF | | BRG | |
|----------------|---------------|---------------|------------------|
| Register Name | Setting Value | Register Name | Setting Value |
| SCSMR.C/A# | B'0 | CKS | H'4000 |
| SCSCR.CKE[1:0] | B'10 | DL | H'0001 to H'FFFF |

- The register settings for the baud rate generator for external clock should be made before starting initialization of the SCIF.

50.3.4 SCIF Interrupt Sources and the DMAC

If the DMAC is used for transmission/reception, set and enable the DMAC before setting the SCIF.

Transmission Interrupts and DMA Transfer:

If the TDFE/TEND flag in SCFSR is set to 1 when the TDFE/TEND interrupt is enabled by the TIE bit, a TDFE/TEND interrupt request and a transmit-FIFO-data-empty DMA transfer request will occur. If the TDFE/TEND flag is set to 1 when TDFE/TEND interrupt is disabled by the TIE bit, only the transmit-FIFO-data-empty DMA transfer request will occur. (A transmit-FIFO-data-empty DMA transfer request is generated when the TDFE flag is set while TEIE is 0, or when the TEND flag is set while TEIE is 1. DMA transfer requests are not affected by the TEIE bit.)

When TDFE/TEND interrupt requests are enabled, the interrupt requests are cleared by the DMAC regardless of the interrupt handling program.

Reception Interrupts and DMA Transfer:

If the RDF/DR flag in SCFSR is set to 1 when RDF/DR interrupt is enabled by the RIE bit, an RDF/DR interrupt request occurs. If the RDF/DR flag is set to 1, a receive-FIFO-data-full DMA transfer request occurs. If the RDF/DR flag is set to 1 when RDF/DR interrupt is disabled by the RIE bit, and only a receive-FIFO-data-full DMA transfer request occurs and DMAC can be activated to perform data transfer.

Setting the RIE bit in SCSCR to 0 and the REIE bit to 1 generates the ER/BRK/ORER interrupt requests without generating RDF/DR interrupt requests. When the BRK flag in SCFSR or the ORER flag in SCLSR is set to 1, BRK/ORER interrupt requests occur.

If the TO flag is set to 1 in SCLSR when TO interrupts are enabled by the TOIE bit, TO interrupt requests occur.

DR/TO interrupt requests generated by setting the DR or TO flag to 1 or receive-FIFO-data-full DMA transfer requests occur only in asynchronous mode.

When DR/TO interrupt requests are enabled to be issued, interrupt requests generated by the DR flag are cleared by the DMAC regardless of the interrupt handling program, however, those generated by the TO flag are not cleared by the DMAC. Therefore, the TO flag interrupt requests need to be cleared with the interrupt handling program. (The DR and TO flags are set at the same time, but cleared separately.)

Table 50.11 SCIF Interrupt Sources

| Interrupt Source | DMAC Activation | Priority on Reset Release |
|---|-----------------|---------------------------|
| Interrupts generated by receive error flag (ER) | Disabled | High |
| Interrupts generated by receive-FIFO-data-full (RDF), receive-data-ready (DR) or timeout (TO) * | Enabled | ↑ ↓ |
| Interrupts generated by break (BRK) or overrun error (ORER) | Disabled | |
| Interrupts generated by transmit FIFO data empty (TDFE) | Enabled | Low |

Note: * RXI interrupts by means of the DR or TO flag are available only in asynchronous mode.

50.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Note the following on use of the SCIF.

50.4.1 Break Detection and Operation

Break signals can also be detected by reading the RX pin value directly when a framing error (FER) is detected. In the break state, the input values from the RX pin are all 0s. So, the parity error flag (PER) may be set after the FER flag is set to 1.

Although the SCIF stops receive data transfer to SCFRDR after detecting a break, it continues data reception.

50.4.2 Sending a Break Signal

The input/output condition and level of the TX pin are determined by the SPB2IO and SPB2DT bits in SCSPTR. This enables to send a break signal.

The pin does not function as the TX pin from the initialization of the serial transmitter to setting of the TE bit (enabling transmission). In this period, the marked state is substituted by the value of the SPB2DT bit. Therefore, the SPB2IO and SPB2DT bits should be set to 1 (designating output and high level) beforehand.

To send a break signal during serial transmission, clear the SPB2DT bit to 0 (low level), and then clear the TE bit to 0 (halting transmission). When the TE bit is cleared, the transmitter is initialized regardless of the current transmission state, and the TX pin outputs 0.

50.4.3 Data Sampling Timing and Reception Margin in Asynchronous Mode

In asynchronous mode, the SCIF operates on the base clock with a frequency 16 times the bit rate.

In reception, the SCIF performs the internal synchronization by sampling the fall edge of the start bit using the base clock. In addition, the SCIF takes receive data at the rising edge of the eighth pulse on the base clock.

Figure 50.25 shows the timing of this operation.

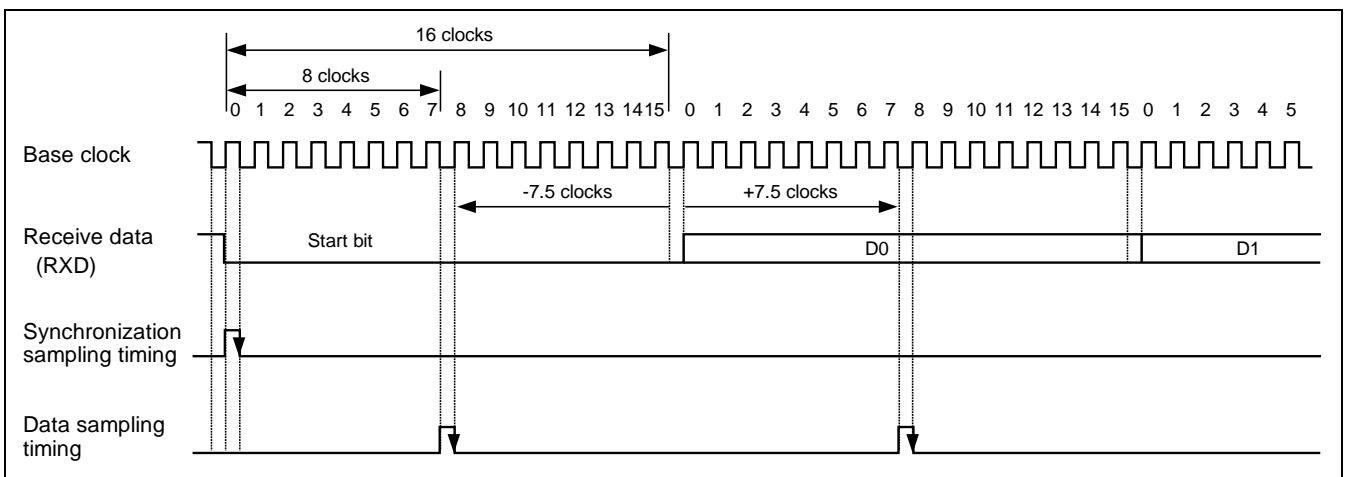


Figure 50.25 Timing Chart of Receive Data Sampling

The reception margin in asynchronous mode is given by formula (1).

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5) F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100\% \quad \text{Expression (1)}$$

M: Receive margin (%)

N: Ratio of bit rate to clock (N = 16)

D: Clock duty (D = 0 to 1.0)

L: Frame length (L = 9 to 12)

F: Absolute value of clock frequency deviation

Assuming F = 0 and D = 0.5 for expression (1), the reception margin obtained with expression (2) is 46.875% as shown below:

Assuming D = 0.5 and F = 0

$$M = (0.5 - 1 / (2 \times 16)) \times 100\% \\ = 46.875\% \quad \text{.....expression (2)}$$

This is a theoretical value. A reasonable margin allowed in system designs is 20% to 30%.

50.4.4 Reception Margin and Baud Rate Error

The value of 46.875% obtained by the above expression (2) indicates the reception margin when the baud rate error is 0 (F = 0). If there is no error in the reception and transmission baud rates, reception is possible even with misalignment of approx. 1/2 bit. If there is an error in the reception and transmission baud rates, the errors are accumulated up to the stop bit reception, which reduces the reception margin. The allowable baud rate error can be obtained by modifying the F in expression (1). When D = 0.5:

$$F = \{(15/32 - M) / (L - 0.5)\} \times 100 (\%) \quad \text{.....expression (3)}$$

By using expression (3), the relationship between the allowable error and reception margin with the frame length = 12 can be summarized as follows:

| Allowed Error (%) | Reception Margin (%) |
|-------------------|----------------------|
| 4.07 | 0 |
| 3.64 | 5 |
| 3.20 | 10 |
| 2.33 | 20 |
| 1.46 | 30 |

51. High Speed Serial Communication Interface with FIFO (HSCIF)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

51.1 Overview

The Second generation RZ/G series products have a high speed serial communication interface with built-in FIFO buffers (high-speed serial communication interface with FIFO: HSCIF) that handles asynchronous communication. The HSCIF has two 128-stage FIFO buffers separately for transmission and reception, which enables fast, efficient, and uninterrupted communication. RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E have 5 channels of HSCIF. All functions of each channel are same.

Internal clock:

- RZ/G2H: S3D1 ϕ
- RZ/G2M V1.3: S3D1 ϕ
- RZ/G2M V3.0: S3D1 ϕ
- RZ/G2N: S3D1 ϕ .
- RZ/G2E: S3D1C ϕ .

51.1.1 Features

The HSCIF has the following features.

- Asynchronous serial communication mode

The HSCIF performs serial data communication based on a character-by-character asynchronous system. This feature enables serial data communication with standard asynchronous communication chips that support universal asynchronous receiver/transmitter (UART) or asynchronous communication interface adapter (ACIA). There is a choice of eight serial data transfer formats.

 - Data length: 7 or 8 bits
 - Stop bit length: 1 or 2 bits
 - Parity: Even/odd/none
 - Receive error detection: Parity, framing, and overrun errors
 - Break detection:

A break is detected when a framing error lasts for more than 1 frame length at space 0 (low level).

When a framing error occurs, a break can also be detected by reading the HRX pin level directly from the serial port register (HSSPTR).
 - Sampling rate: Variable (integer number from 8 to 32)
- Capable of full-duplex communication

The HSCIF has an independent transmitter and receiver that enable simultaneous transmission and reception. The transmitter and receiver both have a 128-stage FIFO buffer structure, enabling continuous serial data transmission and reception.
- On-chip baud rate generator, enabling any bit rate to be selected

The HSCIF enables choice of a clock source for transmission/reception: a clock from the on-chip baud rate generator based on the internal clock or an external clock.
- Eight interrupt sources

The HSCIF has eight types of interrupt sources: receive-data-ready, receive-FIFO-data-full, break detection, transmit-FIFO-data-empty, transmit-end, receive-error, overrun-error and time-out, and enables any of them to be requested independently.

- DMA data transfer

When the transmit FIFO register is empty or the receive FIFO register has received data, issuing a DMA transfer request activates the DMA controller (DMAC) to execute a data transfer.

- Modem control functions (HRTS# and HCTS#) are stored.
- The amount of data in the transmit/receive FIFO registers and the number of receive errors in receive data in the receive FIFO register are available.
- A receive data ready (DR) or a timeout error (TO) can be detected during reception.

51.1.2 Block Diagram

Figure 51.1 shows the HSCIF block diagram.

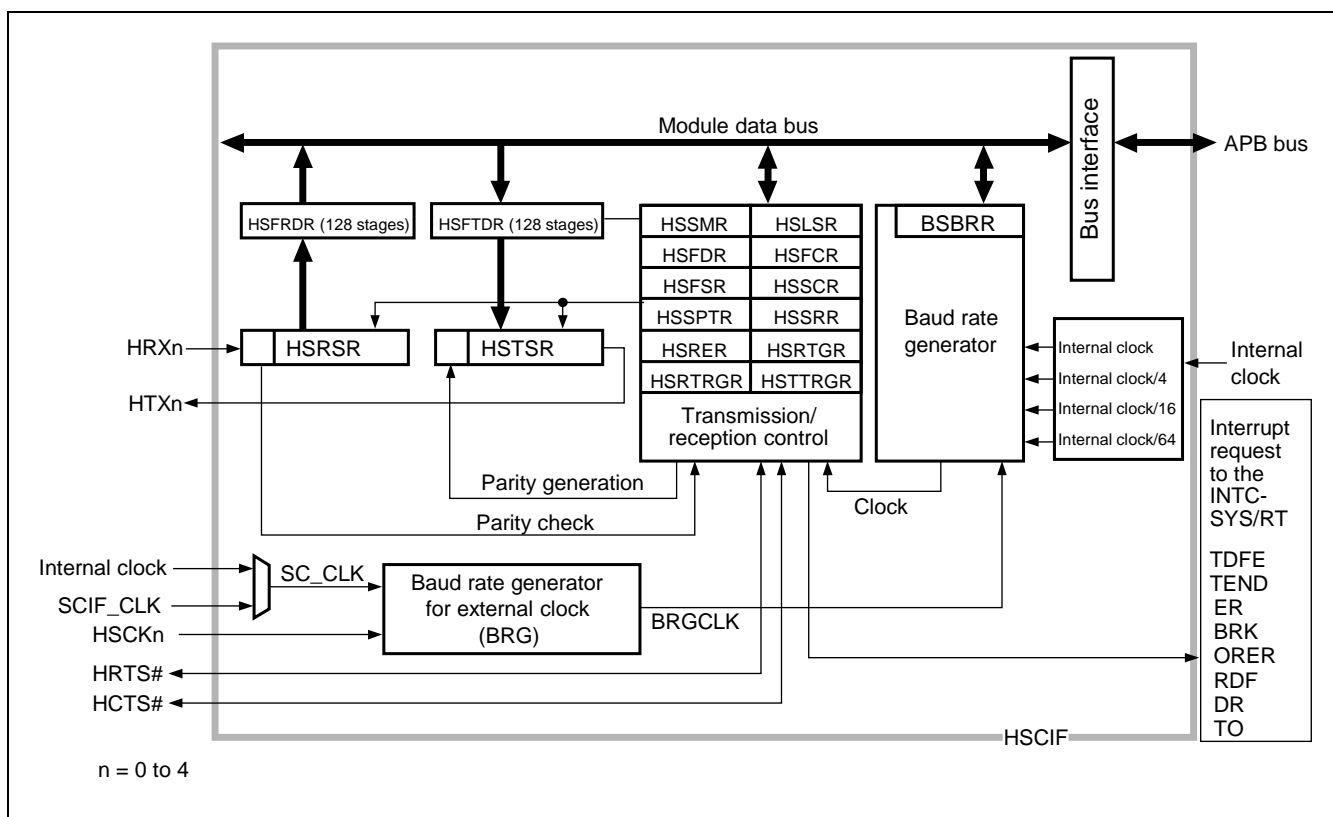


Figure 51.1 HSCIF Block Diagram

[Legend]

- | | |
|-------------------------------------|--|
| HSRSR: Receive shift register | HSFCR: FIFO control register |
| HSFRDR: Receive FIFO data register | HSFDR: FIFO data count register |
| HSTSR: Transmit shift register | HLSLR: Line status register |
| HSFTDR: Transmit FIFO data register | HSSRR: Sampling rate register |
| HSSMR: Serial mode register | HSRER: Serial error register |
| HSSCR: Serial control register | HSRTGR: RTS output active trigger register |
| HSFSR: Serial status register | HSRTRGR: Receive FIFO data count trigger register |
| HSBRR: Bit rate register | HSTTRGR: Transmit FIFO data count trigger register |
| HSSPTR: Serial port register | |

51.1.3 External Pins

Table 51.1 shows the HSCIF pin configuration. These pins are multiplexed in other functions, so that the usage of the pins are may be restricted depending on the multiplexed pin settings.

Table 51.1 Pin Configuration

| Name | Abbreviation | Function | I/O | Descriptions | Second Generation RZ/G Series Products | | | |
|----------|--------------|--------------------------------|--------|---|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| HSCK0 | HSCK | Serial clock pin | Input | Clock Input | √ | √ | √ | √ |
| HSCK1 | | Serial clock pin | Input | Clock Input | √ | √ | √ | √ |
| HSCK2 | | Serial clock pin | Input | Clock Input | √ | √ | √ | √ |
| HSCK3 | | Serial clock pin | Input | Clock Input | √ | √ | √ | √ |
| HSCK4 | | Serial clock pin | Input | Clock Input | √ | √ | √ | √ |
| HRX0 | HRX | Receive data pin | Input | Receive data input | √ | √ | √ | √ |
| HRX1 | | Receive data pin | Input | Receive data input | √ | √ | √ | √ |
| HRX2 | | Receive data pin | Input | Receive data input | √ | √ | √ | √ |
| HRX3 | | Receive data pin | Input | Receive data input | √ | √ | √ | √ |
| HRX4 | | Receive data pin | Input | Receive data input | √ | √ | √ | √ |
| HTX0 | HTX | Transmit data pin | Output | Transmit data output | √ | √ | √ | √ |
| HTX1 | | Transmit data pin | Output | Transmit data output | √ | √ | √ | √ |
| HTX2 | | Transmit data pin | Output | Transmit data output | √ | √ | √ | √ |
| HTX3 | | Transmit data pin | Output | Transmit data output | √ | √ | √ | √ |
| HTX4 | | Transmit data pin | Output | Transmit data output | √ | √ | √ | √ |
| HCTS0# | HCTS# | Modem control pin | I/O | Transmission enabled | √ | √ | √ | √ |
| HCTS1# | | Modem control pin | I/O | Transmission enabled | √ | √ | √ | √ |
| HCTS2# | | Modem control pin | I/O | Transmission enabled | √ | √ | √ | √ |
| HCTS3# | | Modem control pin | I/O | Transmission enabled | √ | √ | √ | √ |
| HCTS4# | | Modem control pin | I/O | Transmission enabled | √ | √ | √ | √ |
| HRTS0# | HRTS# | Modem control pin | I/O | Transmission request | √ | √ | √ | √ |
| HRTS1# | | Modem control pin | I/O | Transmission request | √ | √ | √ | √ |
| HRTS2# | | Modem control pin | I/O | Transmission request | √ | √ | √ | √ |
| HRTS3# | | Modem control pin | I/O | Transmission request | √ | √ | √ | √ |
| HRTS4# | | Modem control pin | I/O | Transmission request | √ | √ | √ | √ |
| SCIF_CLK | SCIF_CLK | Baud rate generation clock pin | Input | Clock for input to the baud rate generator for the external clock | √ | √ | √ | √ |

Note: These pins are made to function as serial pins by setting up HSCIF operation using bits TE, RE, CKE[1:0] in HSSCR, and bit MCE in HSFCCR. HSSPTR of the HSCIF can be used to handle the transmission and detection of break states.

51.1.4 Register Configuration

Table 51.2 shows the registers in the HSCIF. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined.

Table 51.2 Register Configuration

| Register Name | Abbreviation | R/W | Value after Reset | Offset From Base Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|--------------|---------|-------------------|--------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Serial mode register | HSSMR | R/W | H'0000 | H'00 | 16 | √ | √ | √ | √ |
| Bit rate register | HSBRR | R/W | H'FF | H'04 | 8 | √ | √ | √ | √ |
| Serial control register | HSSCR | R/W | H'0000 | H'08 | 16 | √ | √ | √ | √ |
| Transmit FIFO data register | HSFTDR | —/WB | Undefined | H'0C | 8 | √ | √ | √ | √ |
| Serial status register | HSFSR | R/WC0*1 | H'0060 | H'10 | 16 | √ | √ | √ | √ |
| Receive FIFO data register | HSFRDR | R | Undefined | H'14 | 8 | √ | √ | √ | √ |
| FIFO control register | HSFCR | R/W | H'0000 | H'18 | 16 | √ | √ | √ | √ |
| FIFO data count register | HSFDR | R | H'0000 | H'1C | 16 | √ | √ | √ | √ |
| Serial port register | HSSPTR | R/W | H'00XX*3 | H'20 | 16 | √ | √ | √ | √ |
| Line status register | HLSLR | R/WC0*2 | H'0000 | H'24 | 16 | √ | √ | √ | √ |
| Sampling rate register | HSSRR | R/W | H'000F | H'40 | 16 | √ | √ | √ | √ |
| Serial error register | HSRER | R | H'0000 | H'44 | 16 | √ | √ | √ | √ |
| RTS output active trigger register | HSRTGR | R/W | H'000F | H'50 | 16 | √ | √ | √ | √ |
| Receive FIFO data count trigger register | HSRTRGR | R/W | H'0001 | H'54 | 16 | √ | √ | √ | √ |
| Transmit FIFO data count trigger register | HSTTRGR | R/W | H'0008 | H'58 | 16 | √ | √ | √ | √ |

Notes: 1. Only 0 can be written to clear the flags. Bits 15 to 8, 3, and 2 are read-only bits and cannot be modified.
 2. Only 0 can be written to clear the flags. Bits 15 to 3, and 1 are read-only bits and cannot be modified.
 3. The initial values of HSSPTR bits 6, 4, 2, and 0 are undefined.

| Channel No. | Base Address | Second Generation RZ/G Series Products | | | |
|-------------|--------------|--|----------------------------|--------|--------|
| | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 0 | H'E654_0000 | √ | √ | √ | √ |
| 1 | H'E655_0000 | √ | √ | √ | √ |
| 2 | H'E656_0000 | √ | √ | √ | √ |
| 3 | H'E66A_0000 | √ | √ | √ | √ |
| 4 | H'E66B_0000 | √ | √ | √ | √ |

51.2 Register Description

Registers in the HSCIF are allocated to and arranged in the address space of the internal bus.

[Legend for Register Description]

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

R: Read-only. When the bit is reserved, the write value should always be 0. Writing 1 to these bits can cause a malfunction of HSCIF.

—/WB: Write-only. The read value is undefined.

51.2.1 Receive Shift Register (HSRSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

HSRSR is a register that receives serial data.

The HSCIF sets serial data input to the HSRSR from the HRX pin in the order received, starting with the LSB (bit 0), and converts it to parallel data. When one byte of data has been received, it is transferred to the receive FIFO register HSFRDR, automatically.

HSRSR cannot be read from and written to by the CPU.



51.2.2 Receive FIFO Data Register (HSFRDR)

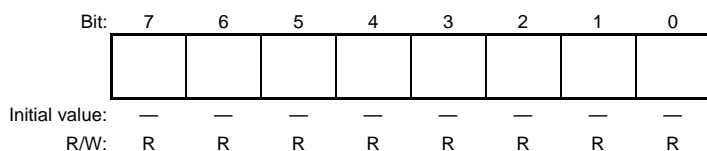
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

HSFRDR is a 128-stage FIFO register that stores received serial data.

When HSCIF has received one byte of serial data, it transfers the received data from the receive shift register (HSRSR) to HSFRDR for storage, and reception is thus completed. HSRSR is then ready for reception, and is capable of receiving up to 128 consecutive bytes of data before HSFRDR is full.

HSFRDR is a read-only register and cannot be modified by the CPU. Note that the read value will be undefined while there is no receive data in HSFRDR. When HSFRDR is full of receive data, subsequent serial data is lost.

HSFRDR is read as an undefined value after a power-on reset.



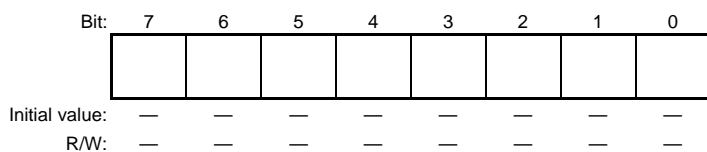
51.2.3 Transmit Shift Register (HSTSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

HSTSR is a register that transmits serial data.

To perform serial data transmission, the HSCIF first transfers transmit data from the transmit fifo data register (HSFTDR) to HSTSR, and then sends the data to the HTX pin starting with the LSB (bit 0). When transmission of one byte is completed, the HSCIF transfers the next transmit data from HSFTDR to HSTSR automatically, then starts transmission.

HSTSR cannot be read from and written to directly by the CPU.



51.2.4 Transmit FIFO Data Register (HSFTDR)

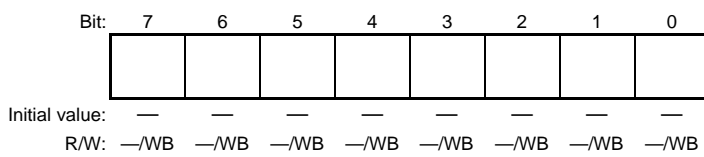
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

HSFTDR is an 8-bit FIFO register of 128 stages that stores data for serial transmission.

If HSTSR is empty after transmit data has been written to HSFTDR, the HSCIF transfers the data from HSFTDR to HSTSR and starts serial transmission.

HSFTDR is a write-only register and cannot be read from by the CPU. Writing further data to HSFTDR is no longer possible when it is full. Attempts at writing data to the register in this situation are ignored.

HSFTDR is read as an undefined value on a power-on reset.



51.2.5 Serial Mode Register (HSSMR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|-----|-----|------|------|---|-----|----------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | CHR | PE | O/E# | STOP | — | — | CKS[1:0] |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R | R/W | R/W |

HSSMR is a 16-bit register that sets the HSCIF's serial transfer format and selects the baud rate generator clock source.

HSSMR can always be read from and written to by the CPU.

HSSMR is initialized to H'0000 by a power-on reset.

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 15 to 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 | CHR | B'0 | R/W | Character Length Selects 7 or 8 bits for data length. When the 7-bit data is selected, the MSB (bit 7) in the transmit FIFO data register (HSFTDR) is not transmitted. 0: 8 bits 1: 7 bits |
| 5 | PE | B'0 | R/W | Parity Enable Determines whether parity bit is added in transmission or not, and parity bit is checked in reception or not. When bit PE is set to 1, the parity (even or odd) specified by bit O/E# is added to transmit data. In reception, the parity bit is checked for the parity (even or odd) specified by bit O/E#. 0: Disables parity bit addition and check. 1: Enables parity bit addition and check. |
| 4 | O/E# | B'0 | R/W | Parity Mode Selects either even or odd parity to use in parity addition and check. The O/E# bit setting is valid only when bit PE is set to 1, enabling parity bit addition and check. 0: Even parity 1: Odd parity When even parity is set, the parity bit is added in transmission so that the total number of 1-bit in the transmit character plus the parity bit is even. In reception, a check is performed to see if the total number of 1-bit in the receive character plus the parity bit is even. When odd parity is set, the parity bit is added in transmission so that the total number of 1-bit in the transmit character plus the parity bit is odd. In reception, a check is performed to see if the total number of 1-bit in the receive character plus the parity bit is odd. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 3 | STOP | B'0 | R/W | <p>Stop Bit Length</p> <p>Selects 1 bit or 2 bits as the stop bit length.</p> <p>In reception, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1, it is treated as a stop bit. If it is 0, it is treated as the start bit of the next transmits character.</p> <p>0: 1 stop bit*¹ 1: 2 stop bits*²</p> <p>Notes: 1. In transmission, high level of 1 bit (stop bit) is added to the end of a transmit character before it is sent. 2. In transmission, high level of 2 bits (stop bits) are added to the end of a transmit character before it is sent.</p> |
| 2 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 1, 0 | CKS[1:0] | B'00 | R/W | <p>Clock Select 1 and 0</p> <p>These bits select the clock source for the on-chip baud rate generator.</p> <p>The clock source can be selected from Internal clock, Internal clock/4, Internal clock/16, and Internal clock/64, according to the setting of bits CKS [1:0].</p> <p>B'00: Internal clock B'01: Internal clock/4 B'10: Internal clock/16 B'11: Internal clock/64</p> |

51.2.6 Serial Control Register (HSSCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

HSSCR is a register that enables or disables transmission/reception by the HSCIF, enables or disables interrupt requests, and selects transmission/reception clock source for the HSCIF.

HSSCR can always be read from and written to by the CPU.

HSSCR is initialized to H'0000 by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----------|-----|----|------|-----|----|---|-----|-----|-----|-----|------|------|----------|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TOT[1:0] | — | — | TEIE | — | — | — | TIE | RIE | TE | RE | REIE | TOIE | CKE[1:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 15, 14 | TOT[1:0] | B'00 | R/W | Set the time for a data ready (DR) or a timeout (TO) to be set in asynchronous mode. B'00: 15 etu* B'01: 31 etu B'10: 47 etu B'11: 63 etu Note: * ETU: Elementary Time Unit (time for transfer of one bit) Equivalent to 1.5 frames with an 8-bit, 1-stop-bit format. |
| 13, 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 | TEIE | B'0 | R/W | Transmit End Interrupt Enable When a transmit-end request is enabled by the TIE bit, the TEIE bit selects the source of the transmit end interrupt request from the following: <ul style="list-style-type: none"> Setting the TDFE flag in HSFSR Setting the TEND flag in HSFSR 0: The transmit FIFO data empty (TDFE) interrupt request is used. 1: The transmit end (TEND) interrupt request is used. |
| 10 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | TIE | B'0 | R/W | <p>Transmit Interrupt Enable</p> <p>Enables or disables a transmit-FIFO-data-empty interrupt (TDFE) request when the TEIE bit in HSSCR is pulled 0, if all of the following conditions are satisfied:</p> <ul style="list-style-type: none"> Serial transmit data has been transferred from HSFTDR to HSTSR. The number of data bytes in HSFTDR is equal to or less than the transmit trigger count. The TDFE flag in HSFSR is set to 1. <p>Enables or disables a transmit-end interrupt (TEND) request when the TEIE bit of HSSCR is set to 1, if both of the following conditions are satisfied:</p> <ul style="list-style-type: none"> Transmission was ended because there is no valid data in HSFTDR when the last bit of the transmit character in HSTSR was transmitted. The TEND flag of HSFSR is set to 1. <p>0: When the TEIE bit is 0, disables transmit-FIFO-data-empty interrupt (TDFE) request. When the TEIE bit is 1, disables transmit-end interrupt (TEND) request.</p> <p>1: When the TEIE bit is 0, enables transmit-FIFO-data-empty interrupt (TDFE) request. When the TEIE bit is 1, enables transmit-end interrupt (TEND) request.</p> |
| 6 | RIE | B'0 | R/W | <p>Receive Interrupt Enable</p> <p>Enables or disables a receive-FIFO-data-full interrupt request when the RDF flag in HSFSR is set to 1, a receive-data-ready interrupt request when the DR flag in HSFSR is set to 1, a receive-error interrupt request when the ER flag in HSFSR is set to 1, a break interrupt request when the BRK flag in HSFSR is set to 1, and an overrun error interrupt request when the ORER flag in HSLSR is set to 1.</p> <p>0: Disables receive-FIFO-data-full interrupts (RDF) requests, receive-data-ready interrupt (DR) requests, receive-error interrupt (ER) requests, break interrupt (BRK) requests, and overrun error interrupt (ORER) requests.</p> <p>1: Enables receive-FIFO-data-full interrupt (RDF) requests, receive-data-ready interrupt (DR) requests, receive-error interrupt (ER) requests, break interrupt (BRK) requests, and overrun error interrupt (ORER) requests.</p> |
| 5 | TE | B'0 | R/W | <p>Transmit Enable</p> <p>Enables or disables the start of HSCIF serial transmission. The HSCIF starts serial transmission when transmit data is written to the HSFTDR while TE is 1. Before setting TE to 1, set HSSMR and HSFCR to specify the transmission format and reset the transmit FIFO.</p> <p>0: Disables transmission.</p> <p>1: Enables transmission.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 4 | RE | B'0 | R/W | <p>Receive Enable</p> <p>Enables or disables the start of HSCIF serial reception. When RE is 1, the HSCIF starts serial reception by detecting a start bit. Before setting RE to 1, set HSSMR and HSFCR to specify the reception format and reset the receive FIFO.</p> <p>0: Disables reception.* 1: Enables reception.</p> <p>Note: * Even if RE is cleared to 0, the DR, ER, BRK, RDF, FER, PER, TO and ORER flags are not affected, and retain their states.</p> |
| 3 | REIE | B'0 | R/W | <p>Receive Error Interrupt Enable</p> <p>Enables or Disables generation of receive-error interrupt (ER) requests, break interrupt (BRK) requests and overrun-error interrupt (ORER) requests.</p> <p>0: Disables receive-error interrupt (ER) requests, break interrupt (BRK) requests and overrun-error interrupt (ORER) requests.* 1: Enables receive-error interrupt (ER) requests, break interrupt (BRK) requests and overrun-error interrupt (ORER) requests.</p> <p>Note: * When REIE is 1, ER, BRK or ORER interrupt requests will occur even if REIE is cleared to 0. This setting is used to notify the interrupt controller of ER, BRK, and ORER interrupt requests during DMAC transfer.</p> |
| 2 | TOIE | B'0 | R/W | <p>Timeout Interrupt Enable</p> <p>Enables or disables generation of timeout interrupt (TO) requests when the TO flag in HSLSR is set to 1.</p> <p>0: Disables timeout interrupts (TO). 1: Enables timeout interrupts (TO).</p> |
| 1, 0 | CKE[1:0] | B'00 | R/W | <p>Clock Enable 1 and 0</p> <p>See Table 51.3 for the bit settings.</p> |

Table 51.3 Clock Selection

| CKE[1] | CKE[0] | Clock Source | HSCK Pin |
|--------|--------|--|---|
| B'0 | B'0 | Internal clock (Internal clock, Internal clock/4, Internal clock/16, and Internal clock/64) | The HSCK pin is not used. The HSCK pin functions as an input pin (input signals are ignored). (Initial value) |
| B'0 | B'1 | Prohibited | — |
| B'1 | B'0 | Baud rate generator output for external clock or HSCK | <p>When SC_CLK is selected: The HSCK pin is an input pin (input signals are ignored). Set the SC_CLK frequency so that the frequency of BRGCLK is multiplied by the sampling rate.</p> <p>When HSCK is selected: The HSCK pin inputs the clock (with the bit rate multiplied by the sampling rate).</p> |
| | B'1 | Prohibited | |

Note: It is not allowed to set synchronous communication using SC_CLK for input.

51.2.7 Serial Status Register (HSFSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

HSFSR is a 16-bit register. The lower 8 bits are status flags that indicate the operating status of the HSCIF, and the upper 8 bits are all reserved.

HSFSR can always be read from and written to by the CPU. However, the flags ER, TEND, TDFE, BRK, RDF, and DR cannot be written by 1. The FER and PER flags are read-only flags and cannot be modified.

HSFSR is initialized to H'0060 by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|-------|-------|-------|-------|-----|-----|-------|-------|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | ER | TEND | TDFE | BRK | FER | PER | RDF | DR | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R/WC0 | R/WC0 | R/WC0 | R/WC0 | R | R | R/WC0 | R/WC0 | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-------|--|
| 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | ER | B'0 | R/WC0 | <p>Receive Error</p> <p>Indicates that a framing error or a parity error has occurred in reception. The ER flag is not affected by an error and retains its previous state when the RE bit is 0 in HSSCR.</p> <p>If a receive error occurs, receive data will be transferred to HSFRDR and reception operation will be continued. Whether there is a receive error in data read from HSFRDR can be determined by the FER and PER bits in HSFSR.</p> <p>0: Indicates that no framing or parity error has occurred in reception.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> A power-on reset is executed. 0 is written to ER. <p>1: Indicates that a framing error or a parity error has occurred in reception.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> The HSCIF checks whether the stop bit at the end of receive data is 1, but the stop bit is 0.* The number of 1-bit in receive data plus the parity bit does not match the parity setting (even or odd) specified by the O/E# bit in HSSMR during reception. <p>Note: * In the 2-stop-bit mode, only the first stop bit is checked that the value is 1; the second stop bit is not checked.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 6 | TEND | B'1 | R/WC0 | <p>Transmit End</p> <p>Indicates that transmission has been ended because there was no valid data in HSFTDR when the last bit of the transmit character was transmitted.</p> <p>0: Indicates that transmission is in progress.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> Transmit data is written to HSFTDR, and 0 is written to TEND. Data is written to HSFTDR by the DMAC. <p>1: Indicates that transmission has been ended.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> A power-on reset is executed. The TE bit in HSSCR is 0. There is no transmission data in HSFTDR when the last bit of a 1-byte serial transmission character is transmitted. |
| 5 | TDFE | B'1 | R/WC0 | <p>Transmit FIFO Data Empty</p> <p>Indicates that the HSCIF has transferred data from HSFTDR to HSTSR, the number of data bytes in HSFTDR becomes equal to or less than the transmit trigger count specified by the HSTTRGR, and HSFTDR is ready to be written by new transmit data.</p> <p>HSFTDR is a 128-byte FIFO register. The maximum number of bytes that can be written to when TDFE = 1 is "128 – [the transmit trigger count]". If data exceeding this value is attempted to be written, the data will be ignored. The number of data bytes in HSFTDR is indicated by the upper bits of HSFDR.</p> <p>If the number of data written in HSFTDR is equal to or less than the transmit trigger count, this bit will be set to 1 even if it is cleared to 0 after it is read as 1.</p> <p>0: Indicates that the number of transmit data written to HSFTDR exceeds the transmit trigger count.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> Transmit data exceeding the specified transmit trigger count have been written to HSFTDR, and 0 is written to TDFE. Transmit data exceeding the specified transmit trigger count have been written to HSFTDR by the DMAC. <p>1: Indicates that the number of transmit data in HSFTDR is equal to or less than the transmit trigger count.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> A power-on reset is executed. The number of transmit data in HSFTDR is equal to or less than the transmit trigger count after transmission. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 4 | BRK | B'0 | R/WC0 | <p>Break Detect</p> <p>Indicates that a receive data break signal has been detected. If a break signal is detected, receive data (H'00) transfer to HSFRDR is stopped. After the break is canceled and the receive signal returns to 1, the receive data transfer resumes.</p> <p>0: Indicates that no break signal has been received.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> A power-on reset is executed. 0 is written to BRK. <p>1: Indicates that a break signal has been received.</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> Data with a framing error is received, followed by the space "0" (low level) for at least one frame length. |
| 3 | FER | B'0 | R | <p>Framing Error</p> <p>Indicates that a framing error has been found in the data that is to be read next from HSFRDR.</p> <p>0: Indicates that there is no framing error in the receive data that is to be read from HSFRDR.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> A power-on reset is executed. There is no framing error in the data that is to be read next from HSFRDR. <p>1: Indicates that there is a framing error in the receive data that is to be read from HSFRDR.</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> There is a framing error in the data that is to be read next from HSFRDR. |
| 2 | PER | B'0 | R | <p>Parity Error</p> <p>This bit indicates that a parity error has been found in the data that is to be read next from HSFRDR.</p> <p>0: Indicates that there is no parity error in the receive data that is to be read from HSFRDR.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> A power-on reset is executed. There is no parity error in the data that is to be read next from HSFRDR. <p>1: Indicates that there is a parity error in the receive data that is to be read from HSFRDR.</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> There is a parity error in the data that is to be read next from HSFRDR. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 1 | RDF | B'0 | R/WC0 | <p>Receive FIFO Data Full</p> <p>Indicates that the received data has been transferred from HSRSR to HSFRDR, and the number of receive data bytes in HSFRDR becomes equal to or more than the receive trigger count specified by the HSRTRGR.</p> <p>HSFRDR is a 128-byte FIFO register. When RDF = 1, data equal to or more than the number of receive trigger data bytes can be read. When HSFRDR is empty, HSFRDR is read as an undefined value. The number of receive data bytes in HSFRDR is indicated by the lower bits of HSFRDR.</p> <p>If the number of data in HSFRDR is equal to or more than the trigger count, this bit will be set to 1 even if it is cleared to 0. At this time, read receive data until the number of data in HSFRDR is less than the trigger count, read RDF as 1, and then clear RDF.</p> <p>0: Indicates that the number of receive data bytes in HSFRDR is less than the specified receive trigger count.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • A power-on reset is executed. • HSFRDR is read until the number of receive data bytes in HSFRDR is less than the receive trigger count, and 0 is written to RDF. • HSFRDR is read by the DMAC until the number of receive data bytes in HSFRDR is less than the receive trigger count. <p>1: Indicates that the number of receive data bytes in HSFRDR is equal to or more than the specified receive trigger count.</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> • Receive data more than the receive trigger count have been stored in HSFRDR. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 0 | DR | B'0 | R/WC0 | <p>Receive Data Ready</p> <p>Indicates that the receive FIFO data register (HSFRDR) contains fewer bytes than the trigger number for reception and no further data have arrived over at least the time corresponding to the setting of HSSCR[15:14]* since the stop bit for the byte to have been received.</p> <p>0: Indicates that data is being received or has been successfully received, and there is no receive data in HSFRDR.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • A power-on reset is executed. • All the receive data in HSFRDR has been read, and 0 is written to DR. • All the receive data in HSFRDR has been read by the DMAC. <p>1: Indicates that no further receive data has been received.</p> <p>[Setting condition]</p> <p>The receive FIFO data register (HSFRDR) contains fewer bytes than the trigger number for reception and no further data have arrived over at least the time corresponding to the setting of HSSCR[15:14]* since the stop bit for the byte to have been received.</p> <p>Note: * When the setting is 00, the time is 15 etu. This is equivalent to 1.5 frames in an 8-bit, 1-stop-bit format. etu: Elementary Time Unit (time for transfer of one bit)</p> |

51.2.8 Bit Rate Register (HSBRR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

HSBRR is an 8-bit register that sets the serial transmission/reception bit rate in accordance with the baud rate generator operating clock selected by the CKS [1:0] bits in HSSMR. This baud rate generator is intended for Internal clock, Internal clock/4, Internal clock/16, and Internal clock/64. For details on the baud rate generator for external clock, see section 51.5, Baud Rate Generator for External Clock (BRG).

HSBRR can always be read from and written to by the CPU except for during transfer.

HSBRR is initialized to H'FF by a power-on reset.

| | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

The HSBRR setting is determined by the following equation:

[Asynchronous mode]

$$N = \frac{\text{Internal clock}}{\text{Sr} \times 2^{2n+1} \times B} \times 10^6 - 1$$

B: Bit rate (bit/s)

N: HSBRR setting for the baud rate generator ($0 \leq N \leq 255$) (which satisfies the electrical characteristics)

Internal clock: AXI bus clock operating frequency (MHz)

n: Baud rate generator input clock ($n = 0, 1, 2, 3$)
(See the Table 51.4 for the relation between n and the clock.)

Sr: Sampling rate (8 to 32)

Table 51.4 HSSMR Settings

| n | Baud Rate Generator Input Clock | HSSMR Setting | |
|---|------------------------------------|---------------|--------|
| | | CKS[1] | CKS[0] |
| 0 | Internal clock | B'0 | B'0 |
| 1 | Internal clock/4 | B'0 | B'1 |
| 2 | Internal clock/16 | B'1 | B'0 |
| 3 | Internal clock/64 | B'1 | B'1 |

The bit rate error in asynchronous mode is determined by the following equation:

$$\text{error (\%)} = \left\{ \frac{\text{Internal clock} \times 10^6}{(N + 1) \times B \times \text{Sr} \times 2^{2n+1}} - 1 \right\} \times 100$$

51.2.9 FIFO Control Register (HSFCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

HSFCR is a register that resets data counts for transmit and receive FIFO registers. It also has a modem control and a loopback test enable bit.

HSFCR can always be read from and written to by the CPU except for during transfer.

HSFCR is initialized to H'0000 by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|---|---|-----|-------|-------|------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | MCE | TFRST | RFRST | LOOP |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 | MCE | B'0 | R/W | Modem Control Enable Enables or disables modem control signals HCTS# and HRTS#. 0: Disables modem signals.* 1: Enables modem signals. Note: * HCTS# and HRTS# control ports. |
| 2 | TFRST | B'0 | R/W | Transmit FIFO Data Register Reset Enables or disables a transmit FIFO data register reset that empties the register. 0: Disables the reset.* 1: Enables the reset. Note: * The register is reset by a power-on reset. |
| 1 | RFRST | B'0 | R/W | Receive FIFO Data Register Reset Enables or disables a receive FIFO data register (HSFRDR) reset that empties the register. 0: Disables the reset.* 1: Enables the reset. Note: * The register is reset by a power-on reset. |
| 0 | LOOP | B'0 | R/W | Loopback Test Enables or disables the loopback test by internally connecting the transmit output pin (HTX) and receive input pin (HRX), and the HRTS# pin and HCTS# pin. 0: Disables the loopback test. 1: Enables the loopback test. |

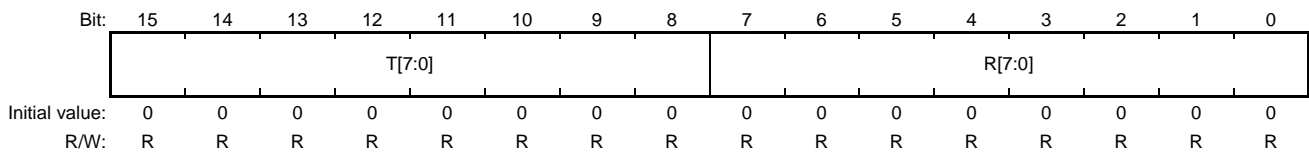
51.2.10 FIFO Data Count Register (HSFDR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

HSFDR is a 16-bit register that indicates the number of data bytes stored in HSFTDR and that in HSFRDR.

The upper 8 bits indicate the number of transmit data bytes in HSFTDR, and the lower 8 bits indicates the number of receive data bytes in HSFRDR.

HSFDR can always be read by the CPU.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 15 to 8 | T[7:0] | H'00 | R | These bits indicate the number of data bytes un-transmitted and still stored in HSFTDR. H'00 indicates that there is no transmit data in HSFTDR, and H'80 indicates that HSFTDR is full of transmit data. |
| 7 to 0 | R[7:0] | H'00 | R | These bits indicate the number of receive data stored in HSFRDR. H'00 indicates that there is no receive data in HSFRDR, and H'80 indicates that HSFRDR is full of receive data. |

51.2.11 Serial Port Register (HSSPTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

HSSPTR controls multiplexed input/output and data on the high speed serial communication interface (HSCIF) ports. Bits 1 and 0 control breaks in serial transmission/reception by reading input data from the HRX pin and writing output data to the HTX pin. Bits 3 and 2 read input data from and write output data to the HSCK pin. Bits 5 and 4 read input data from and write output data to the HCTS# pin. Bits 7 and 6 read input data from and write output data to the HRTS# pin.

HSSPTR is a 16-bit register that can always be read from and written to by the CPU.

All HSSPTR bits except bits 6, 4, 2, and 0 are initialized to 0 by a power-on reset. The values of bits 6, 4, 2, and 0 are undefined.

Note: Whether modem control can be selected or not depends on the channel.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|-------|-------|-------|-------|-------|-------|---------|---------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | RTSIO | RTSDT | CTSIO | CTSDT | SCKIO | SCKDT | SPB 2IO | SPB 2DT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | 0 | — | 0 | — | 0 | — |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|------|---|
| 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | RTSIO | B'0 | R/W | Serial Port – RTS Port Input/output Specifies input or output for the serial port HRTS# pin. To actually set the HRTS# pin as a port output pin to output the value set by the RTSDT bit, the MCE bit in HSFCR should be cleared to 0. 0: Indicates that this bit does not output the value of the RTSDT bit to the HRTS# pin. 1: Indicates that this bit outputs the value of the RTSDT bit to the HRTS# pin. |
| 6 | RTSDT | — | R/W* | Serial Port – RTS Port Data Specifies the input/output data level of the serial port HRTS# pin. Whether the pin is set for input or output is determined by the RTSIO bit. When the pin is set for output, the value of the RTSDT bit is output to the HRTS# pin. Regardless of the value of the RTSIO bit, the value of the HRTS# pin is read from the RTSDT bit. The initial value of this bit is undefined after a power-on reset. 0: Indicates that the input/output data is low level. 1: Indicates that the input/output data is high level. |
| 5 | CTSIO | B'0 | R/W | Serial Port – CTS Port Input/output Specifies input or output for the serial port HCTS# pin. To actually set the HCTS# pin as a port output pin to output the value set by the CTSDT bit, the MCE bit in HSFCR should be cleared to 0. 0: Indicates that the CTSDT bit value is not output to the HCTS# pin. 1: Indicates that the CTSDT bit value is output to the HCTS# pin. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 4 | CTS DT | — | R/W* | <p>Serial Port – CTS Port Data</p> <p>Specifies the input/output data level of the serial port HCTS# pin. Whether the pin is set for input or output is determined by the CTSIO bit. When the pin is set for output, the value of the CTS DT bit is output to the HCTS# pin. Regardless of the value of the CTSIO bit, the value of the HCTS# pin is read from the CTS DT bit.</p> <p>The initial value of this bit is undefined after a power-on reset.</p> <p>0: Indicates that the input/output data is low level.</p> <p>1: Indicates that the input/output data is high level.</p> |
| 3 | SCKIO | B'0 | R | Serial Port – Clock Port Input |
| 2 | SCKDT | — | R | <p>Serial Port – Clock Port Data</p> <p>The initial value of this bit is undefined after a power-on reset.</p> |
| 1 | SPB2IO | B'0 | R/W | <p>Serial Port – Break Input/output</p> <p>Specifies the output condition of the serial port HTX pin. To actually set the HTX pin as a port output pin to output the value set by the SPB2DT bit, the TE bit in HSSCR should be cleared to 0.</p> <p>0: Indicates that the SPB2DT bit value is not output to the HTX pin.</p> <p>1: Indicates that the SPB2DT bit value is output to the HTX pin.</p> |
| 0 | SPB2DT | — | R/W* | <p>Serial Port – Break Data</p> <p>Specifies the input level of the serial port HRX pin and the output level of the HTX pin. The HTX pin output conditions are determined by the SPB2IO bit. When the HTX pin is set for output, the value of the SPB2DT bit is output to the HTX pin. Regardless of the value of the SPB2IO bit, the value of the HRX pin is read from the SPB2DT bit.</p> <p>The initial value of this bit is undefined after a power-on reset.</p> <p>0: Indicates that the input/output data is low level.</p> <p>1: Indicates that the input/output data is high level.</p> |

Note: * The value written to these bits can't be read because the value of the pin is read.

51.2.12 Line Status Register (HLSLR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|---|---|---|-------|---|-------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | TO | — | ORER |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/WC0 | R | R/WC0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-------|--|
| 15 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | TO | B'0 | R/WC0 | <p>Timeout</p> <p>Indicates that the receive FIFO data register (HSFRDR) contains fewer bytes than the trigger number for reception and no further data have arrived over at least the time corresponding to the setting of HSSCR[15:14]* since the stop bit for the byte to have been received.</p> <p>0: Indicates that data is being received or has been successfully received and that there is no receive data in HSFRDR.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> A power-on reset is executed. All the receive data in the HSFRDR has been read, and 0 is written to TO. <p>1: Indicates that no further receive data has been received (receive timeout).</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> The receive FIFO data register (HSFRDR) contains fewer bytes than the trigger number for reception and no further data have arrived over at least the time corresponding to the setting of HSSCR[15:14]* since the stop bit for the byte to have been received.* <p>Note: * When the setting is 00, the time is 15 etu. This is equivalent to 1.5 frames in an 8-bit, 1-stop-bit format. etu: Elementary Time Unit (time for transfer of one bit)</p> |
| 1 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 0 | ORER | B'0 | R/WC0 | <p>Overrun Error</p> <p>Indicates that an overrun error has occurred in reception and abnormal termination is caused.</p> <p>If an overrun error occurs, the receive data prior to the overrun error is retained in HSFRDR and the data received subsequently is discarded.</p> <p>Any subsequent serial reception is disabled while the ORER flag is 1.</p> <p>To resume data reception after clearing the ORER flag, be sure to first read (or clear) data in the receive FIFO and handle the error, then clear the ORER flag.</p> <p>0: Indicates that data is being received or has been successfully received.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • A power-on reset is executed. • 0 is written to ORER. <p>1: Indicates that an overrun error has occurred in reception.</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> • The next serial reception has been completed while HSFRDR is full of 128-byte data. <p>Note: When bit RE in HSSCR is cleared to 0, the ORER flag is not affected and its previous state is retained.</p> |

51.2.13 Sampling Rate Register (HSSRR)

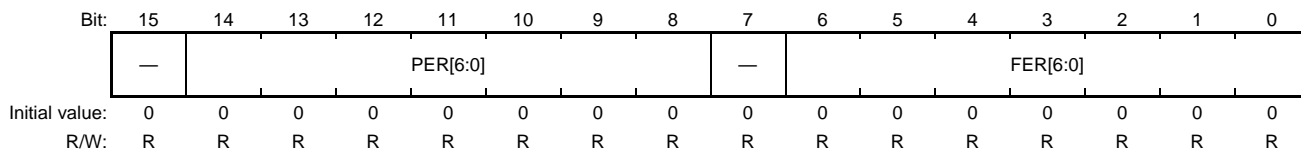
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|------|----|----|-----------|-----|-----|-----|---|---|---|------------|-----|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SRE | SRDE | — | — | SRHP[3:0] | | | | — | — | — | SRCYC[4:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| R/W: | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 15 | SRE | B'0 | R/W | Sampling Rate Register Enable (SRE) 0: Set the SRCYC [4: 0] bits to 15 (initial value). 1: Validates the setting of the SRCYC [4:0] bits. |
| 14 | SRDE | B'0 | R/W | Sampling Point Register Enable (SRDE) 0: Invalidates the setting of the SRHP [3:0] bits and the sampling point will be (S+1)/2 for an odd sampling rate (S) and S/2 for an even sampling rate. 1: Validates the setting of the SRHP [3: 0] bits. |
| 13, 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 to 8 | SRHP[3:0] | B'0000 | R/W | Sampling Point Register (SRHP) The sampling point can be moved by setting the SDRE bit to 1 and setting a value in these bits. Normally, the sampling point is the point of S/2 or (S+1)/2 for a sampling rate of S. By setting a signed 4-bit integer in these bits, the sampling point can be shifted by the amount of the specified sampling clock cycles. This will improve the receive margin. When setting a value in these bits, take notice that the sampling point does not become a negative value or it does not exceed the sampling rate. The shifted sampling point must satisfy the setup margin and hold margin. |
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | SRCYC[4:0] | B'0_1111 | R/W | Bits 4 to 0: Sampling Rate Register (SRCYC) Data transfer at a desired sampling rate can be enabled by setting the SRE bit to 1 and setting a value in these bits. Set a value of "S - 1" in these bits for a sampling rate of S. Note that the sampling rate must be from 8 to 32 (a value from 7 to 31 can be set in these bits). Set these bits to 15 (initial value) when the SRE bit is set to 0. |

51.2.14 Serial Error Register (HSRER)

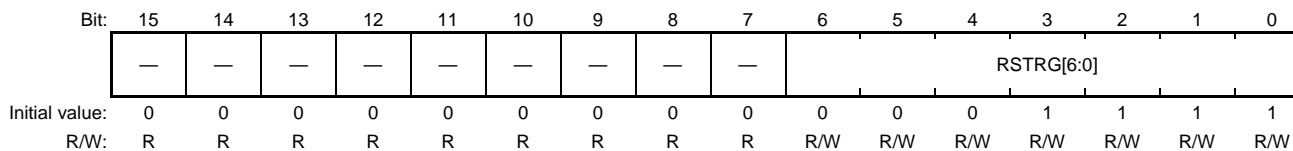
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 15 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 14 to 8 | PER[6:0] | H'00 | R | Parity Error Count These bits indicate the number of data items in which a parity error occurred in the receive data stored in the receive FIFO data register (HSFRDR). After the ER bit in HSFSR is set, the value in bits 14 to 8 will be the number of data items in which a parity error occurred. If all 128 bytes of receive data in HSFRDR have parity errors, bits PER [6:0] will have the value 0. |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 to 0 | FER[6:0] | H'00 | R | Framing Error Count These bits indicate the number of data items in which a framing error occurred in the receive data stored in the receive FIFO data register (HSFRDR). After the ER bit in HSFSR is set, the value in bits 6 to 0 will be the number of data items in which a framing error occurred. If all 128 bytes of receive data in HSFRDR have framing errors, bits FER [6:0] will have the value 0. |

51.2.15 RTS Output Active Trigger Register (HSRTGR)

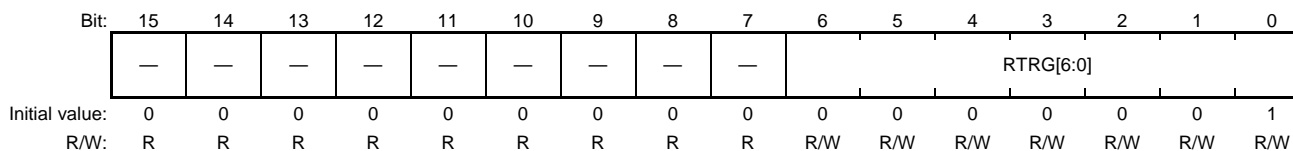
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 15 to 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 to 0 | RSTRG[6:0] | H'0F | R/W | RTS Output Active Trigger Count The HRTS# signal goes high when the number of receive data items stored in the receive FIFO data register (HSFRDR) exceeds the value set in these bits. The initial value is 15. |

51.2.16 Receive FIFO Data Count Trigger Register (HSRTRGR)

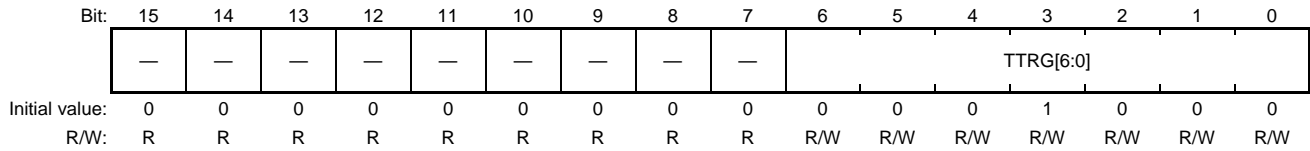
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 15 to 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 to 0 | RTRG[6:0] | H'01 | R/W | Receive FIFO Data Count Trigger These bits set the receive data item count at which the receive data full (RDF) flag in the serial status register (HSFSR) is set. The RDF flag is set when the number of receive data items stored in the receive FIFO data register (HSFRDR) equals or exceeds the value set in these bits. The initial value is 1. |

51.2.17 Transmit FIFO Data Count Trigger Register (HSTTRGR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 15 to 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 to 0 | TTRG[6:0] | H'08 | R/W | Transmit FIFO Data Count Trigger These bits set the un-transmitted data item count at which the transmit FIFO data register empty (TDFE) flag in the serial status register (HSFSR) is set. The TDFE flag is set when the number of transmit data items in the transmit FIFO data register (HSFTDR) falls under the value set in these bits due to transmit operations. The initial value is 8. |

51.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

51.3.1 Operation in Asynchronous Serial Communication Mode

In asynchronous serial communication mode, the HSCIF performs serial communication, in which data is transmitted or received in character units using the attached start bit indicating the start of communication and stop bit indicating the end of communication.

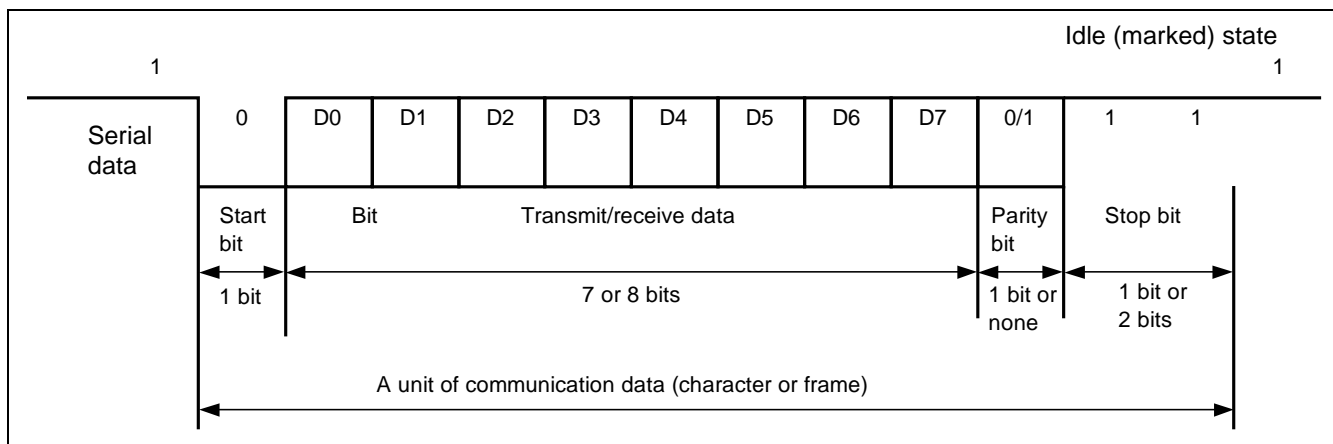
Figure 51.2 shows the general format for asynchronous serial communication.

In asynchronous serial communication, the transmission line is usually kept marked (high level). The HSCIF monitors the transmission line, and when it finds a space (low level), it regards the space as a start bit and starts serial communication.

One character in serial communication consists of a start bit (low level), followed by transmit/receive data (LSB-first; from the lowest bit), a parity bit (high or low level), and a stop bit (high level).

During reception in asynchronous mode, the HSCIF performs synchronization at the falling edge of the start bit. Transferred data is acquired at the center of each bit because the HSCIF samples on $S/2$ th cycles of the clock which has a frequency of S times the bit rate, when the sampling rate as set in the sampling rate register (HSSRR) is S (if the sampling-rate setting is an odd number, the data is sampled on every $(S+1)/2$ th pulse).

In addition, when the setting of the SRDE bit makes the setting of the sampling point bits effective, the point where the bits are latched can be intentionally moved from the centers of each bit.



**Figure 51.2 Data Format in Asynchronous Mode
(Example of 8-Bit Data with Parity and Two Stop Bits)**

(1) Transmission/reception format

Table 51.5 shows available data transfer formats. The HSCIF supports 8 transfer formats, which can be specified by HSSMR.

Table 51.5 Serial Transmission/Reception Formats (Asynchronous Mode)

| SCSMR settings | | | Serial transmission/reception format and frame length | | | | | | | | | | | |
|----------------|----|------|---|---|---|---|---|---|---|---|---|----|----|----|
| CHR | PE | STOP | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 0 | 0 | 0 | S [8-bit data] STOP | | | | | | | | | | | |
| 0 | 0 | 1 | S [8-bit data] STOP STOP | | | | | | | | | | | |
| 0 | 1 | 0 | S [8-bit data] P STOP | | | | | | | | | | | |
| 0 | 1 | 1 | S [8-bit data] P STOP STOP | | | | | | | | | | | |
| 1 | 0 | 0 | S [7-bit data] STOP | | | | | | | | | | | |
| 1 | 0 | 1 | S [7-bit data] STOP STOP | | | | | | | | | | | |
| 1 | 1 | 0 | S [7-bit data] P STOP | | | | | | | | | | | |
| 1 | 1 | 1 | S [7-bit data] P STOP STOP | | | | | | | | | | | |

[Legend]

- S : Start bit
- STOP : Stop bit
- P : Parity bit

(2) Clock

The transfer clock can be selected from the following two clocks using the $CKE[1:0]$ bits in HSSCR:

- Internal clock generated by the on-chip baud rate generator
- External clock generated by an external clock baud rate generator

(3) Data transmission/reception

(a) Initialization of HSCIF (asynchronous mode)

Before transmitting/receiving data or changing the operating mode or communication format, the HSCIF should be initialized using the sample flowchart for HSCIF initialization shown in Figure 51.3.

[Notes]

Clearing the TE bit to 0 initializes HSTSR. However, HSFSR, HSFTDR, and HSFRDR contents are retained even if the TE and RE bits are cleared to 0.

The TE bit should be cleared to 0 after all transmit data has been sent and the TEND flag has been set in HSFSR. The TE bit can be cleared to 0 during transmission, but the data being transmitted will enter the marked state after clearing. In addition, before setting the TE bit to 1 to restart the transmission, set the TFRST bit to 1 in HSFCR to reset HSFTDR.

When an external clock is used, do not stop the clock during operation or initialization. If stopped, the operation will be unreliable. Furthermore, when the baud rate generator for external clock is also to be used, be sure to make settings for it before starting initialization of the HSCIF.

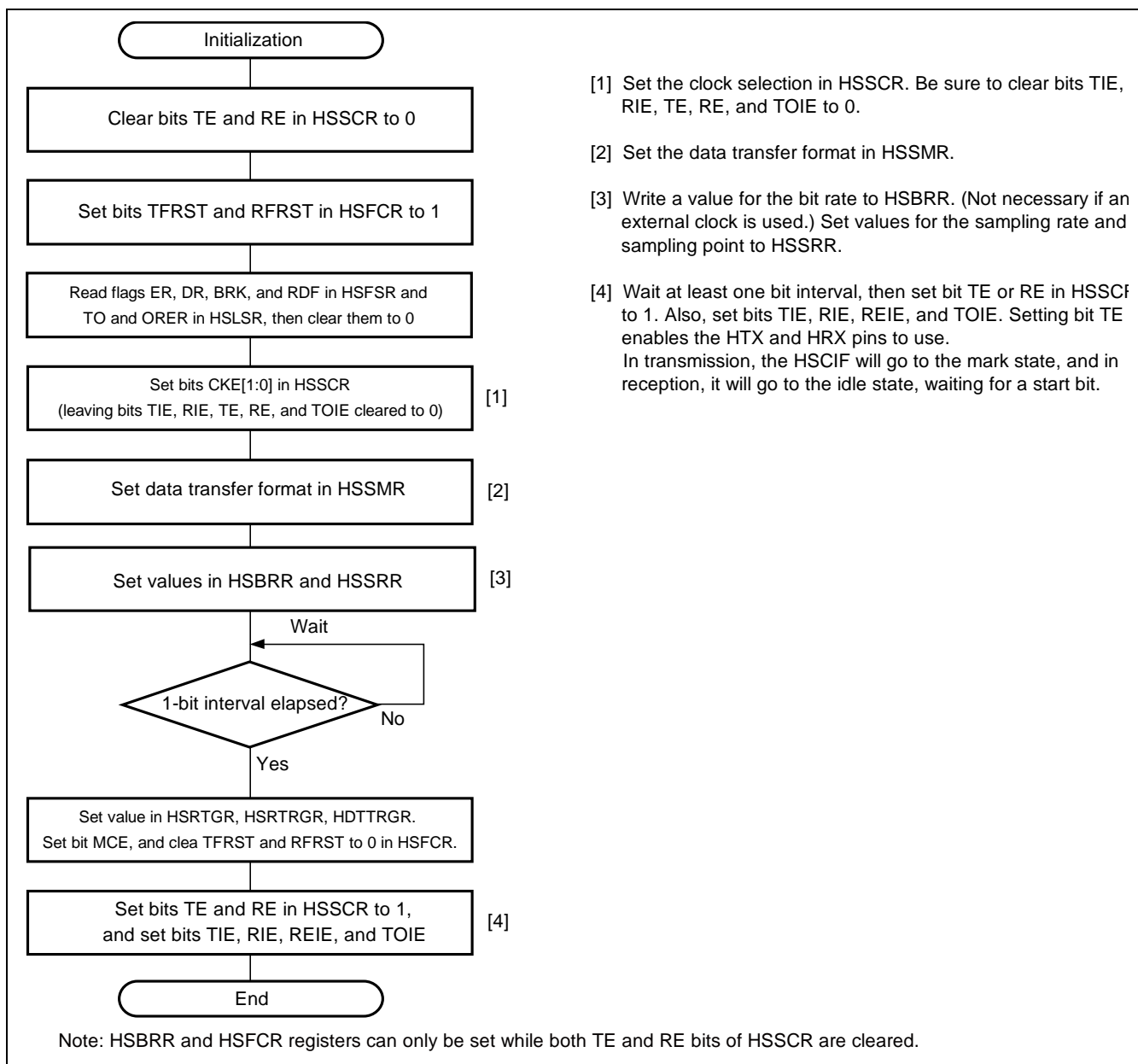


Figure 51.3 Sample Flowcharts for Initializing the HSCIF

(b) Serial data transmission (asynchronous mode)

Figure 51.4 shows a sample flowchart for serial transmission.

After the HSCIF transmission operation is enabled, serial data transmission can be performed using the following procedure:

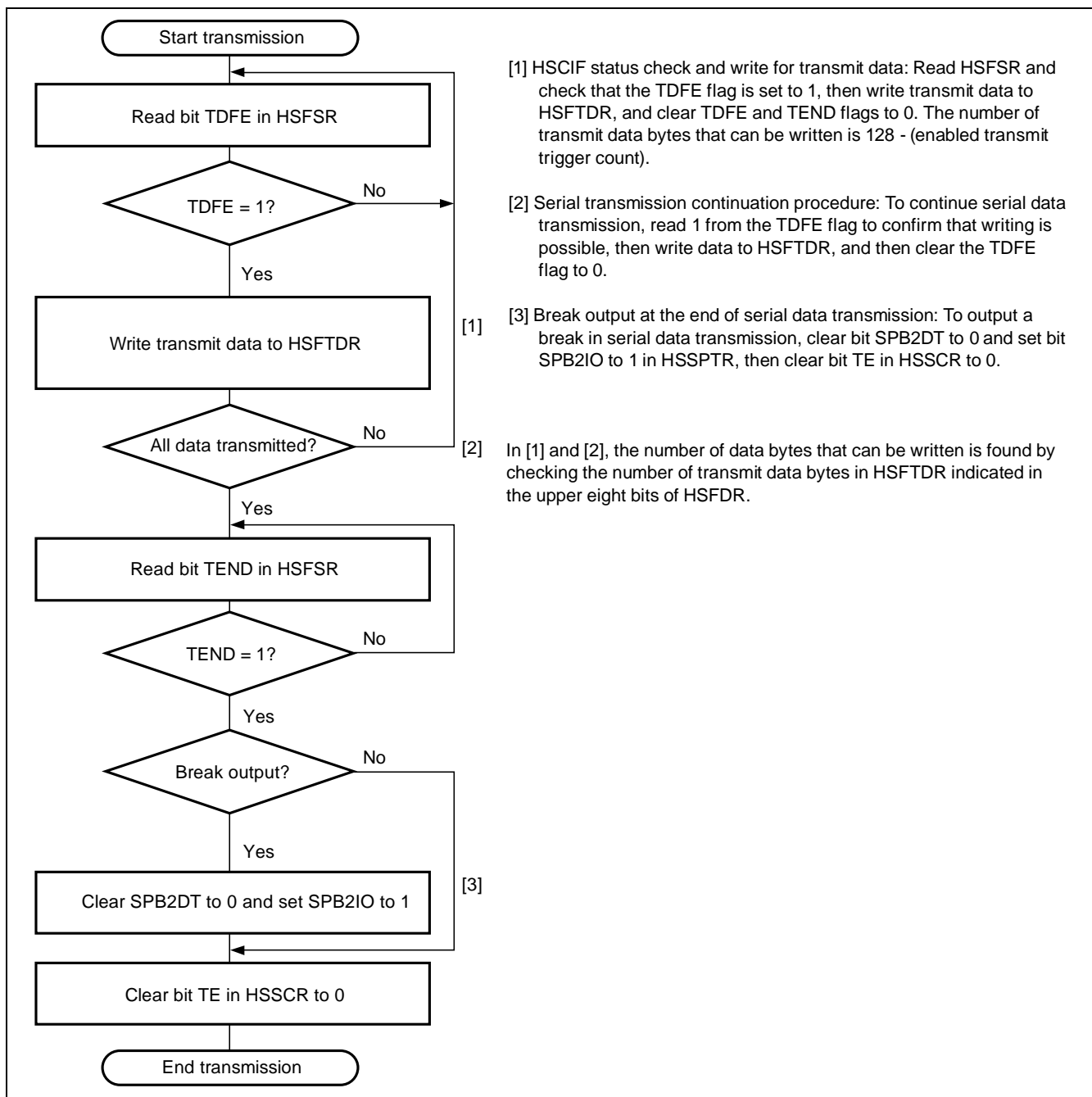


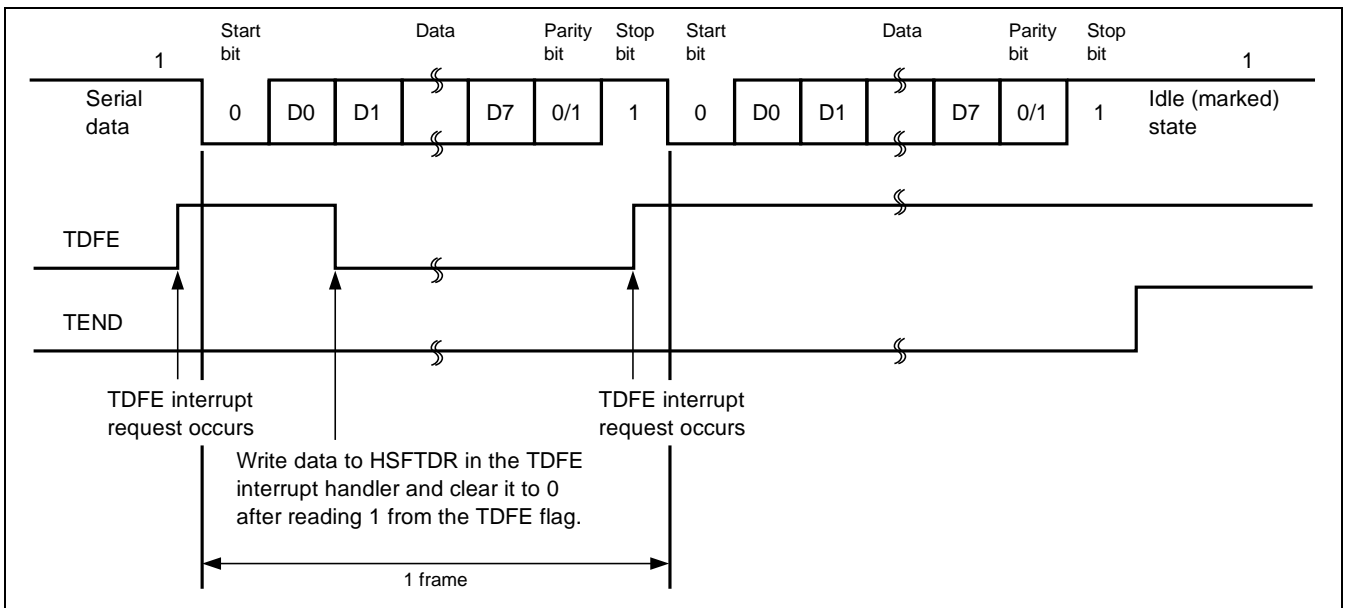
Figure 51.4 Sample Flowcharts for Serial Transmission

In serial transmission, the HSCIF operates as follows:

1. When data is written to HSFTDR, the HSCIF transfers the data from HSFTDR to HSTSR and starts transmitting. Confirm that the TDFE flag in HSFSR is set to 1 before writing transmit data to HSFTDR. The number of data bytes that can be written is at least 128 - (transmit trigger count).

2. When data is transferred from HSFTDR to HSTSR and the HSCIF starts transmission, consecutive transmission is performed until there is no transmit data left in HSFTDR. When the number of transmit data bytes in HSFTDR is equal to or less than the transmit trigger count specified in HSTTRGR, the TDFE flag is set. If the TIE and TEIE bits in HSSCR are set to 1 and 0, respectively at this time, a transmit-FIFO-data-empty interrupt (TDFE) request occurs. The serial transmit data is sent from the HTX pin in the following order:
 - A. Start bit: One 0-bit is output.
 - B. Transmit data: 8-bit or 7-bit data is output in LSB-first order.
 - C. Parity bit: One parity bit (even or odd parity) is output.
A format that does not output a parity bit can also be selected.
 - D. Stop bit(s): One or two stop bits are output.
 - E. Marked state: 1 is output continuously until the start bit that starts the next transmission is sent.
3. The HSCIF checks transmit data in HSFTDR when sending the stop bit. If there is data in it, the HSCIF transfers the data from HSFTDR to HSTSR, sends the stop bit, and then starts serial transmission of the next frame. If there is no transmit data, the HSCIF sets the TEND flag to 1 in HSFSR and sends out the stop bit, and then the marked state is entered to output 1 continuously. At this time, if the TIE and TEIE bits in HSSCR are set to 1, a transmit-end interrupt (TEND) request occurs.

Figure 51.5 shows an example of transmission in asynchronous mode.



**Figure 51.5 Sample HSCIF Transmission Operation
(Example of 8-Bit Data with Parity and One Stop Bit)**

4. When modem control is enabled, transmission can be stopped or resumed in accordance with the HCTS# input value. When HCTS# is set to 1 during transmission, the marked state is entered after one frame of data transmission is ended. Setting HCTS# to 0 restarts outputting the next transmit data from the start bit. Figure 51.6 shows an example of the operation with modem control enabled.

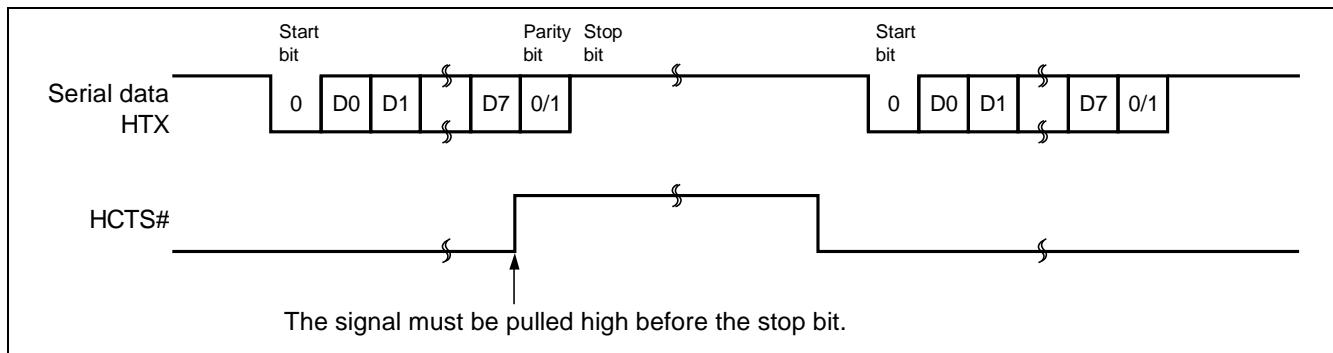


Figure 51.6 Sample Operation with Modem Control Enabled (HCTS#)

(c) Serial data reception (asynchronous mode)

Figures 51.7 and 51.8 show sample flowcharts for serial reception.

After the HSCIF reception operation is enabled, serial data reception can be performed using the following procedure:

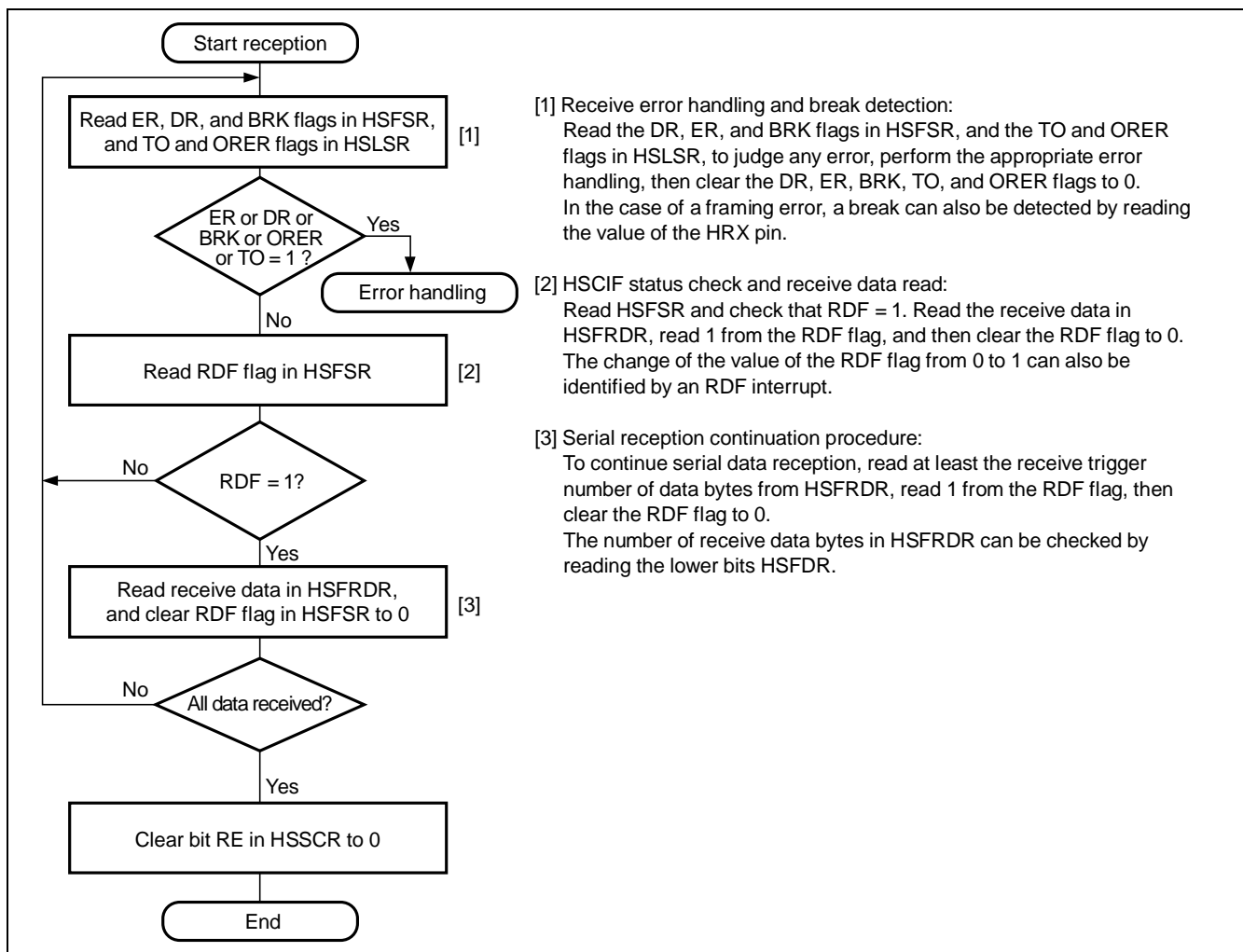


Figure 51.7 Sample Flowcharts for Serial Reception (1)

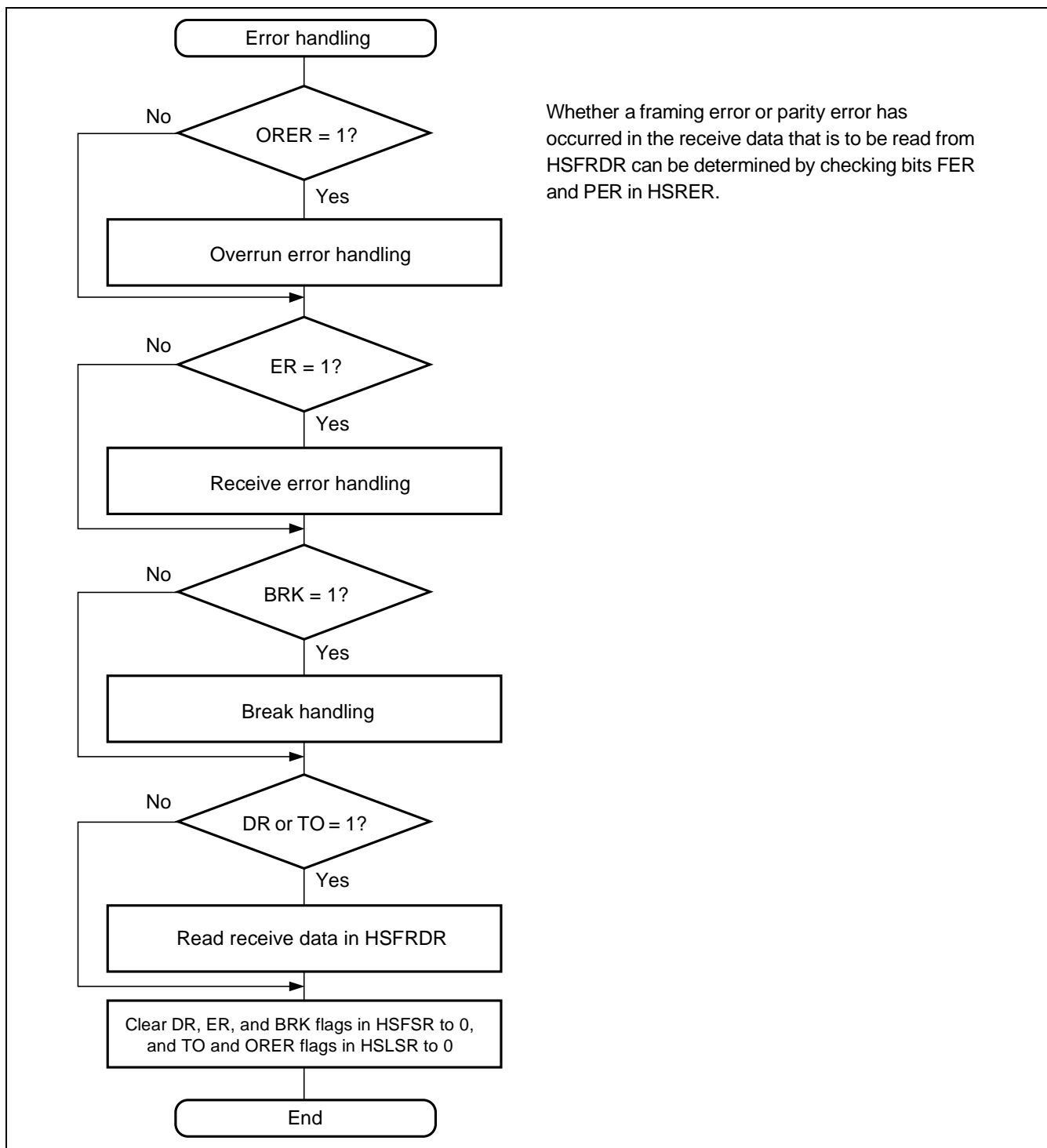


Figure 51.8 Sample Flowcharts for Serial Reception (2)

In serial reception, the HSCIF operates as follows:

1. The HSCIF monitors the transmission line, and when detecting the start bit 0, it performs internal synchronization and starts reception.
2. The HSCIF stores the received data in HSRSR in LSB-to-MSB order.
3. The HSCIF receives the parity bit and stop bit.

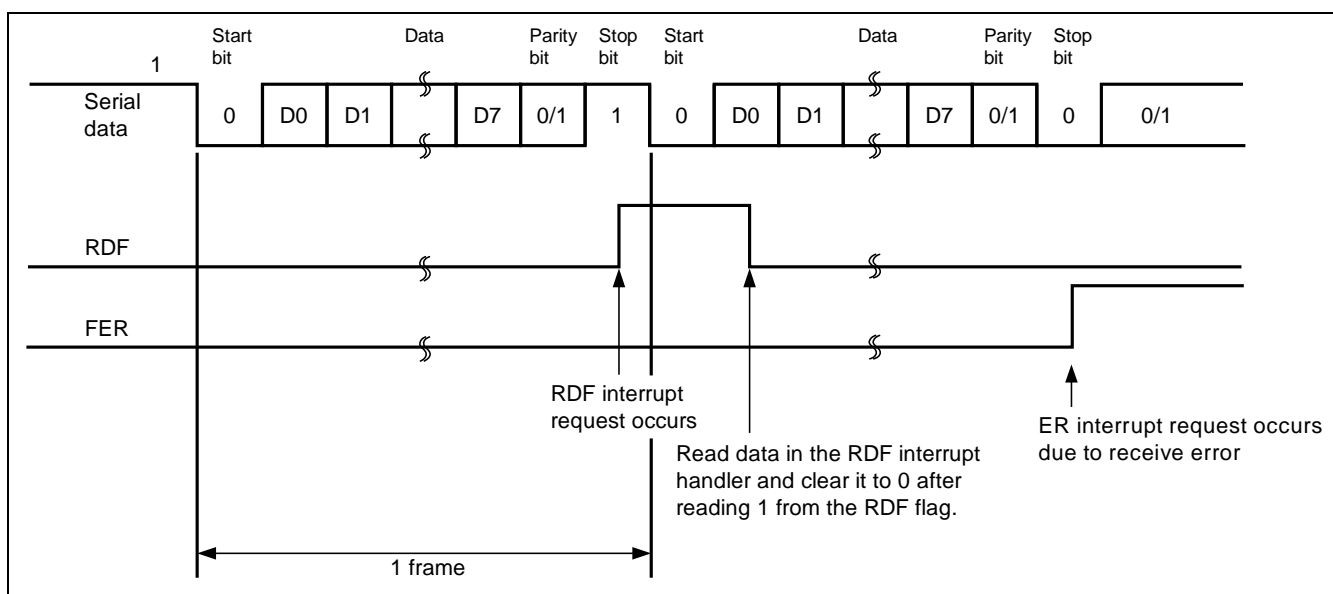
After receiving these bits, the HSCIF performs the following checks:

- A. Stop bit: The HSCIF checks whether the stop bit is 1.
If there are two stop bits, it checks only the first stop bit.
 - B. Receive data: The HSCIF checks that receive data can be transferred from the receive shift register (HSRSR) to HSFRDR.
 - C. Overrun error: The HSCIF checks that the ORER flag is 0, indicating that no overrun error has occurred.
 - D. Break state: The HSCIF checks that the BRK flag is 0, indicating that the break state is not set.
- If the HSCIF can confirm the conditions of (b), (c), and (d), it stores the receive data in HSFRDR.

Note: The HSCIF continues to receive data even when a parity error or a framing error occurs.

- 4. If the RDF flag changes to 1 while the RIE bit in HSSCR is 1, a receive-FIFO-data-full interrupt (RDF) request occurs.
If the DR flag changes to 1 while the RIE bit in HSSCR is 1, a receive-data-ready interrupt (DR) request occurs.
If the TO flag changes to 1 while the TOIE bit in HSSCR is 1, a timeout interrupt (TO) request occurs.
If the ER flag changes to 1 while the RIE or REIE bit in HSSCR is 1, a receive-error interrupt (ER) request occurs.
If the BRK flag changes to 1 while the RIE or REIE bit in HSSCR is 1, a break interrupt (BRK) request occurs.
If the ORER flag changes to 1 while the RIE or REIE bit in HSSCR is 1, an overrun-error interrupt (ORER) request occurs.

Figure 51.9 shows an example of reception in asynchronous mode.



**Figure 51.9 Sample HSCIF Receive Operation
(Example of 8-Bit Data with Parity and One Stop Bit)**

- 5. When modem control is enabled, the HRTS# signal is output when HSFRDR is empty. When HRTS# is 0, data can be received. When HRTS# is set to 1, the number of data bytes in HSFRDR is equal to or more than the RTS output active trigger count (HSRTGR.RSTRG).

Figure 51.10 shows an example of the operation with modem control enabled.

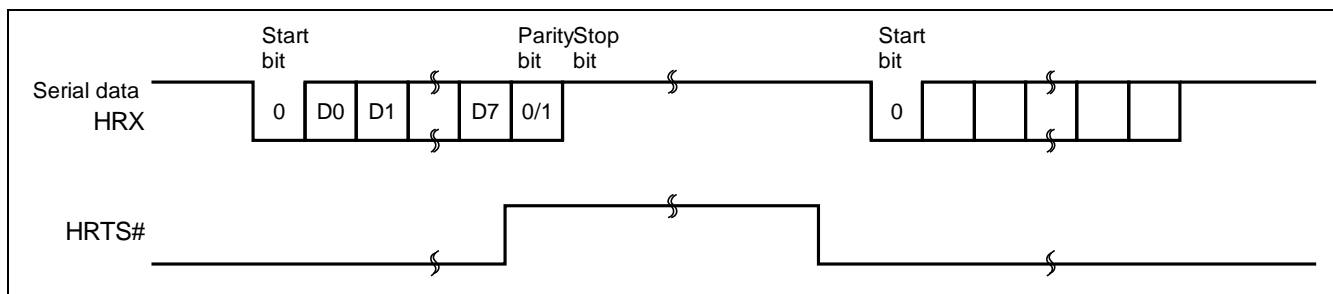


Figure 51.10 Example of the Operation with Modem Control Enabled (HRTS#)

51.3.2 HSCIF Interrupt Sources and the DMAC

If the DMAC is used for transmission/reception, set and enable the DMAC before setting the HSCIF.

Transmission interrupts and DMA transfer:

If the TDFE/TEND flag in HSFSR is set to 1 when the TDFE/TEND interrupt is enabled by the TIE bit, a TDFE/TEND interrupt request and a transmit-FIFO-data-empty DMA transfer request will occur. If the TDFE/TEND flag is set to 1 when TDFE/TEND interrupt is disabled by the TIE bit, only the transmit-FIFO-data-empty DMA transfer request will occur. (A transmit-FIFO-data-empty DMA transfer request is generated when the TDFE flag is set while TEIE is 0, or when the TEND flag is set while TEIE is 1. DMA transfer requests are not affected by the TEIE bit.)

When TDFE/TEND interrupt requests are enabled, the interrupt requests are cleared by the DMAC regardless of the interrupt handling program.

Reception interrupts and DMA transfer:

If the RDF/DR flag in HSFSR is set to 1 when RDF/DR interrupt is enabled by the RIE bit, an RDF/DR interrupt request occurs. If the RDF/DR flag is set to 1, a receive-FIFO-data-full DMA transfer request occurs. If the RDF/DR flag is set to 1 when RDF/DR interrupt is disabled by the RIE bit, and only a receive-FIFO-data-full DMA transfer request occurs and DMAC can be activated to perform data transfer.

Setting the RIE bit in HSSCR to 0 and the REIE bit to 1 generates the ER/BRK/ORER interrupt requests without generating RDF/DR interrupt requests. When the BRK flag in HSFSR or the ORER flag in HSLSR is set to 1, BRK/ORER interrupt requests occur.

If the TO flag is set to 1 in HSLSR when TO interrupts are enabled by the TOIE bit, TO interrupt requests occur.

When DR/TO interrupt requests are enabled to be issued, interrupt requests generated by the DR flag are cleared by the DMAC regardless of the interrupt handling program, however, those generated by the TO flag are not cleared by the DMAC. Therefore, the TO flag interrupt requests need to be cleared with the interrupt handling program. (The DR and TO flags are set at the same time, but cleared separately.)

Table 51.6 HSCIF Interrupt Sources

| Interrupt Source | DMAC Activation |
|---|-----------------|
| Interrupts generated by receive error flag (ER) | Disabled |
| Interrupts generated by receive-FIFO-data-full (RDF), receive-data-ready (DR) or timeout (TO) | Enabled |
| Interrupts generated by break (BRK) or overrun error (ORER) | Disabled |
| Interrupts generated by transmit FIFO data empty (TDFE) | Enabled |

51.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Note the following on use of the HSCIF.

(1) Break detection and operation

Break signals can also be detected by reading the HRX pin value directly when a framing error (FER) is detected. In the break state, the input values from the HRX pin are all 0s. So, the parity error flag (PER) may be set after the FER flag is set to 1.

Although the HSCIF stops receive data transfer to HSFRDR after detecting a break, it continues data reception.

(2) Sending a break signal

The input/output condition and level of the HTX pin are determined by the SPB2IO and SPB2DT bits in HSSPTR. This enables to send a break signal.

The pin does not function as the HTX pin from the initialization of the serial transmitter to setting of the TE bit (enabling transmission). In this period, the marked state is substituted by the value of the SPB2DT bit. Therefore, the SPB2IO and SPB2DT bits should be set to 1 (designating output and high level) beforehand.

To send a break signal during serial transmission, clear the SPB2DT bit to 0 (low level), and then clear the TE bit to 0 (halting transmission). When the TE bit is cleared, the transmitter is initialized regardless of the current transmission state, and the HTX pin outputs 0.

(3) Data sampling timing and reception margin in asynchronous mode

The HSCIF operates on the base clock with a frequency multiplied by the number which is set as a sampling rate for the bit rate.

In reception, the HSCIF performs the internal synchronization by sampling the falling edge of the start bit using the base clock. In addition, the HSCIF takes receive data at the rising edge of the S/2 pulse (when S is an even number) or (S+1)/2 pulse (when S is an odd number) on the base clock when sampling rate is S.

Figure 51.11 shows the timing of this operation when S = 16.

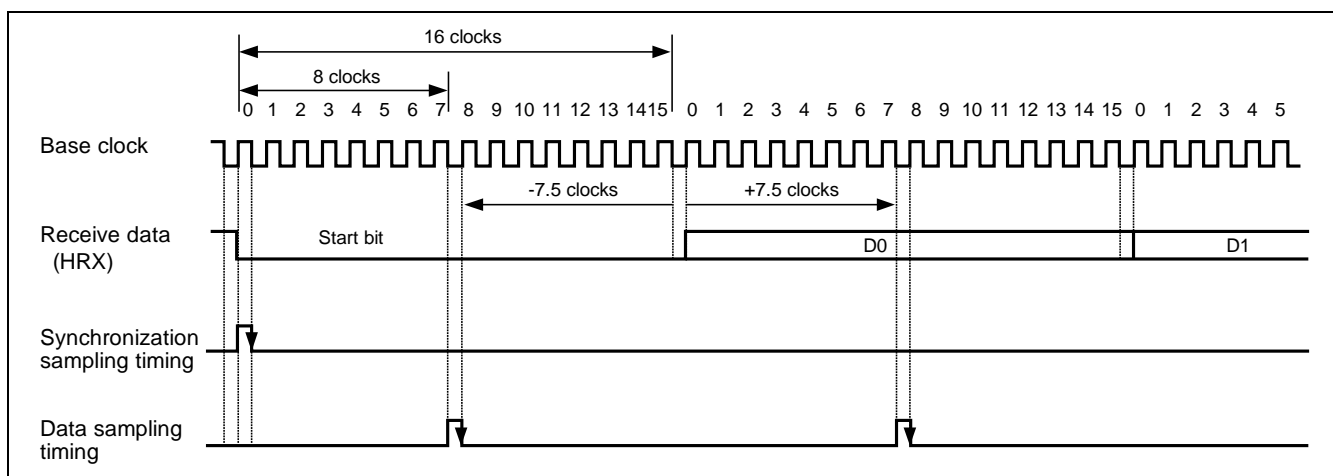


Figure 51.11 Timing Chart of Receive Data Sampling

The reception margin in asynchronous mode is given by formula (1).

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5) F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100\% \quad \text{Expression (1)}$$

M: Receive margin (%)

N: Ratio of bit rate to clock (N = sampling rate)

D: Clock duty (D = 0 to 1.0)

L: Frame length (L = 9 to 12)

F: Absolute value of clock frequency deviation

Assuming F = 0, D = 0.5 and sampling rate = 16 for expression (1), the reception margin obtained with expression (2) is 46.875% as shown below:

Assuming D = 0.5 and F = 0

$$\begin{aligned} M &= (0.5 - 1/(2 \times 16)) \times 100\% \\ &= 46.875\% \dots\dots \text{expression (2)} \end{aligned}$$

This is a theoretical value. A reasonable margin allowed in system designs is 20% to 30%.

(4) Reception margin and baud rate error

The value of 46.875% obtained by the above expression (2) indicates the reception margin when the baud rate error is 0 (F = 0). If there is no error in the reception and transmission baud rates, reception is possible even with misalignment of approx. 1/2 bit. If there is an error in the reception and transmission baud rates, the errors are accumulated up to the stop bit reception, which reduces the reception margin. The allowable baud rate error can be obtained by modifying the F in expression (1). When D = 0.5:

$$F = \{(15/32 - M)/(L - 0.5)\} \times 100 (\%) \dots\dots \text{expression (3)}$$

By using expression (3), the relationship between the allowable error and reception margin with the frame length = 12 can be summarized as follows:

| Allowed Error (%) | Reception Margin (%) |
|-------------------|----------------------|
| 4.07 | 0 |
| 3.64 | 5 |
| 3.20 | 10 |
| 2.33 | 20 |
| 1.46 | 30 |

(5) Usage method of SRHP bits

The method for using the SRHP bits in the sampling rate register is described below.

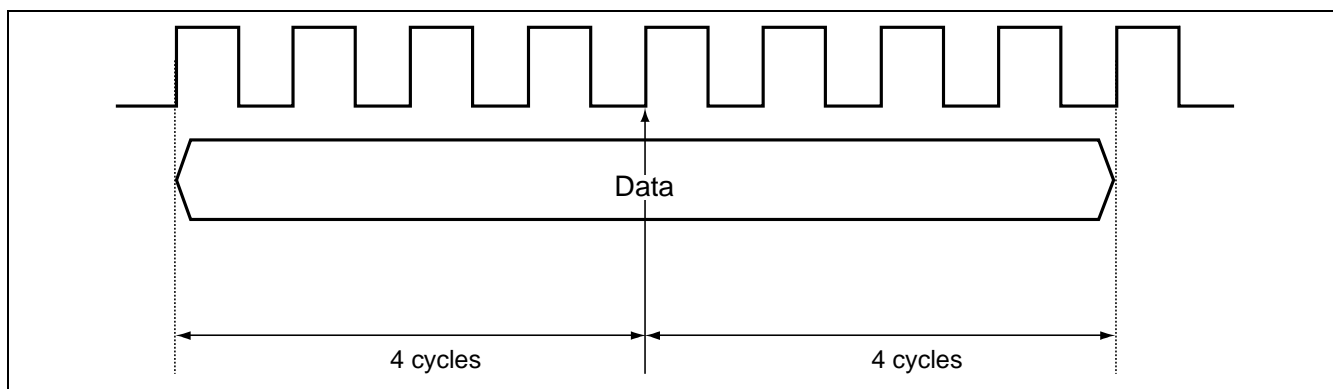


Figure 51.12 Sampling with Invalid SRHP Bits

Figure 51.12 shows a sampling example in which the SRHP bits are invalid (sampling rate = 8). In this case, the HSCIF samples data at half of the sampling rate, which is at the rising edge of the fourth pulse of the clock. This allows a setup margin and hold margin of 50% each to be provided.

However, if the ratio between the baud rate and sampling clock is not 1:1, either the setup margin or hold margin is omitted in data reception of a single frame. If the setup margin is omitted, the hold margin increases. On the other hand, if the hold margin is omitted, the setup margin increases. Figure 51.13 and 51.14 show examples.

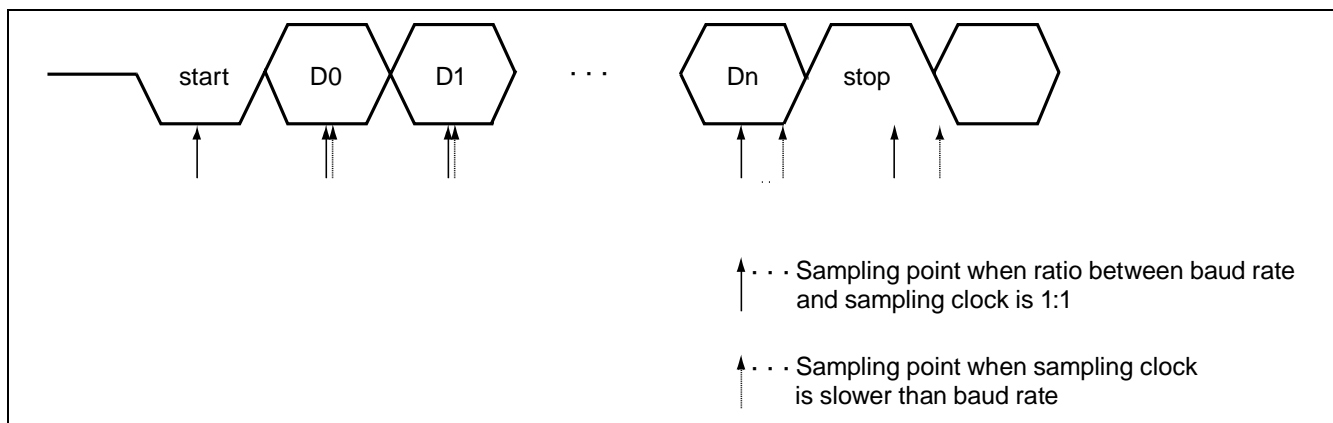


Figure 51.13 Sampling Point with Sampling Clock Slower Than Baud Rate

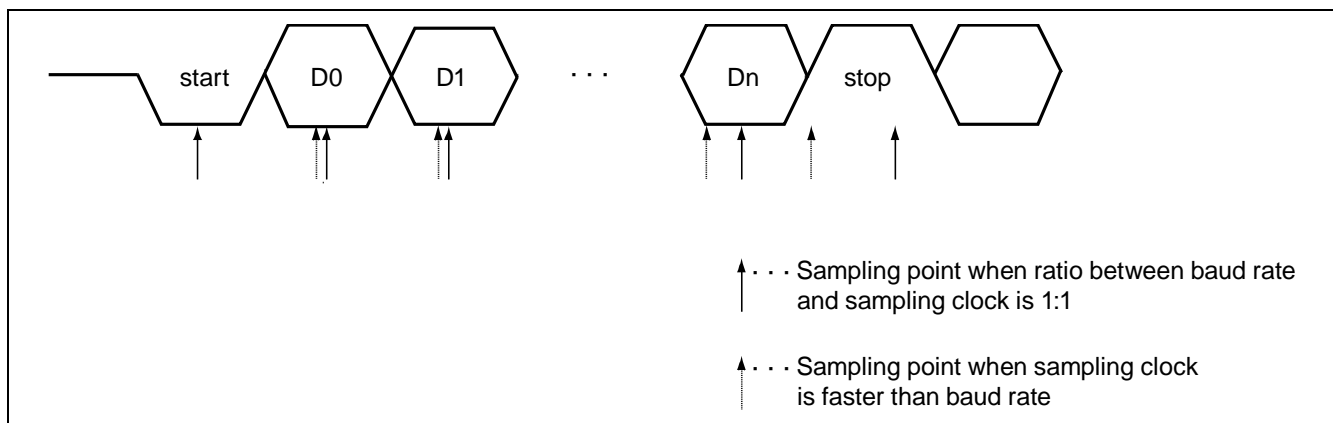


Figure 51.14 Sampling Point with Sampling Clock Faster Than Baud Rate

The ratio between the baud rate and sampling clock can be used to determine which margin among the setup margin and hold margin is to be omitted and which is to be increased. Based on this, the margin within a single frame can be increased by intentionally increasing the margin to be omitted in advance. The sampling point can be moved by setting a value in the SRHP bits. Figure 51.15 shows an example.

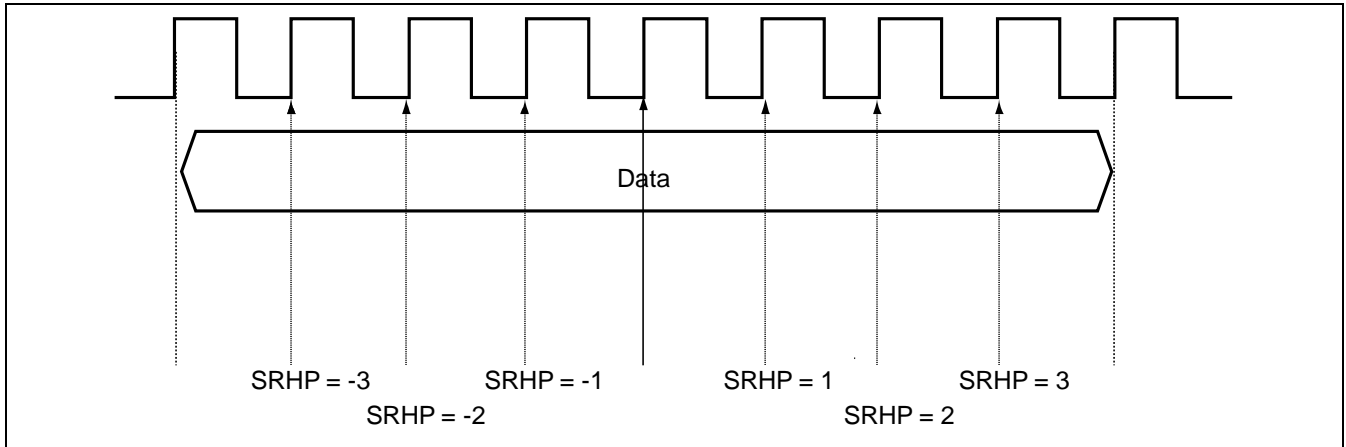


Figure 51.15 Sampling Point Moved by SRHP Bits

(6) Sampling rate and bit rate settings and respective margin

To set the baud rate in the HSCIF, in addition to setting the clock division ratio, the two registers, bit rate register (HSBRR) and sampling rate register (HSSRR), need to be set. These registers must be set so that the margin calculated by expression (1) in section 51.4 (3) will be a sufficient value.

The bit rate error is small when the value obtained by dividing the divided clock frequency used in the HSCIF with the sampling rate is near the desired baud rate. If there are many combinations of bit rate and sampling rate with the same bit rate error, choosing a combination with a large sampling rate provides a large margin. This is because when the bit rate error is constant in expression (1) in section 51.4 (3), that is, when the "absolute value of clock frequency deviation (F)" is constant, the margin (M) increases in accordance with the sampling rate (N).

If the sampling rate is made larger in a combination of bit rate and sampling rate with different bit rate errors, the bit rate error is increased. Accordingly, the "absolute value of clock frequency deviation (F)" on the right side in expression (1) in section 51.4 (3) gets larger and the margin (M) smaller.

Refer to the following procedure when selecting the sampling rate and bit rate settings.

1. Obtain the bit rate with the smallest bit rate error for sampling rates from 8 to 32.
2. Using expression (1) in section 51.4 (3), calculate the margin for each combination of sampling rate and bit rate which was obtained in step 1.
3. Select the combination with the largest margin among the combinations of sampling rate and bit rate.

51.5 Baud Rate Generator for External Clock (BRG)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

51.5.1 Overview

The HSCIF incorporates a baud rate generator for external clock (abbreviated as BRG, hereafter). The BRG supplies a sampling clock (BRGCLK) to the HSCIF core by dividing the external clock SC_CLK (selectable between SCIF_CLK and Internal clock) by 1 to $2^{16} - 1$. In addition, the BRG switches the output between the external clock HSCK and divided clock.

51.5.2 Block Diagram

Figure 51.16 shows a block diagram of the BRG.

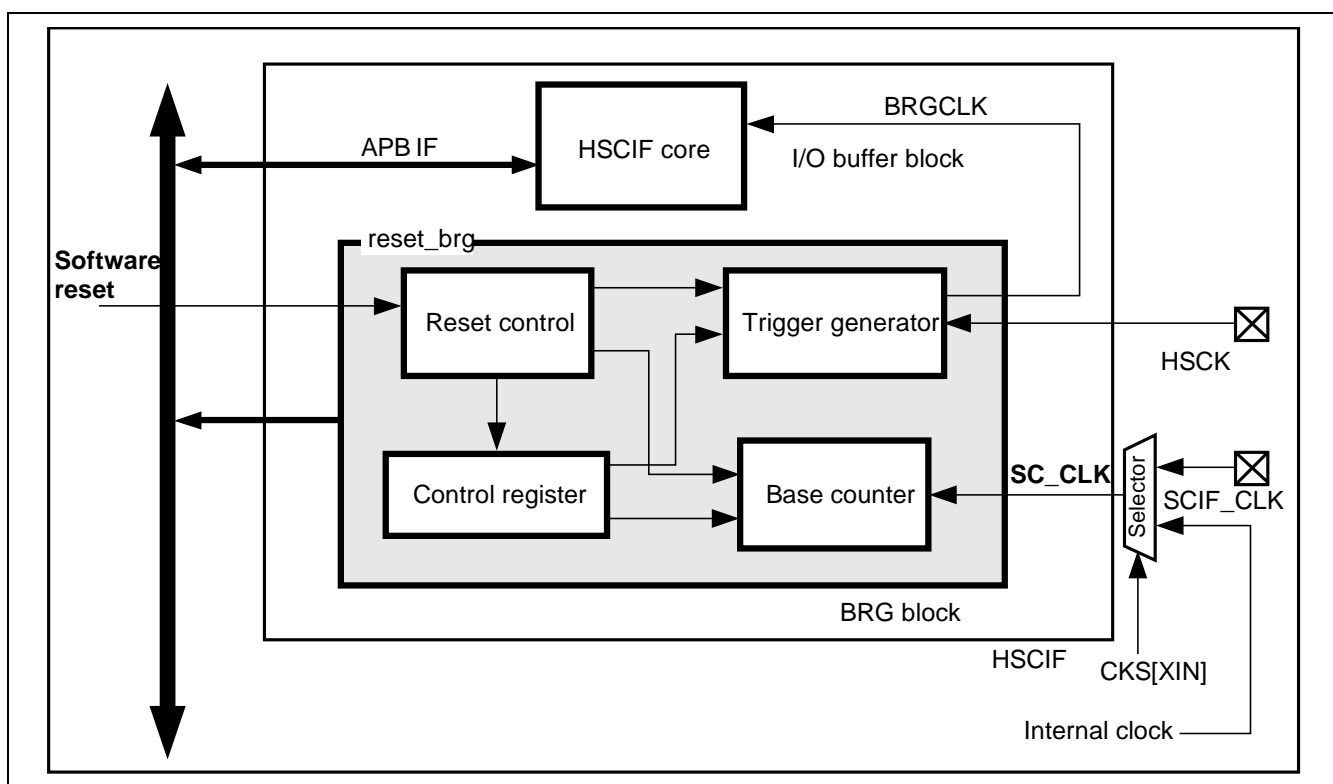


Figure 51.16 BRG Block Diagram

(1) Reset controller:

The reset controller handles resetting the control register, base counter, and trigger generator.

(2) Control register:

The control register has the frequency division register and clock select register.

(3) Base counter:

The base counter is a 16-bit CLK synchronization counter that is used to determine the timing for generating a frequency divided clock.

(4) Trigger generator:

The trigger generator generates rising-edge/falling-edge triggers for a frequency divided clock with the timing according to values of the frequency division register and base counter. The triggers are used to generate the frequency divided clock. In addition, the trigger generator switches the output between the HSCK (external clock input) and frequency divided clock.

51.5.3 BRG Register Configuration

Table 51.7 shows the registers in the BRG block.

Table 51.7 List of Registers

| Name | Code | R/W | Initial Value | Offset From Base Address | Access Size | Second Generation RZ/G Series Products | | | |
|-----------------------------|------|-----|---------------|--------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Frequency division register | DL | R/W | H'00 | H'30 | 16 | √ | √ | √ | √ |
| Clock select register | CKS | R/W | H'00 | H'34 | 16 | √ | √ | √ | √ |

51.5.3.1 Frequency Division Register (DL)

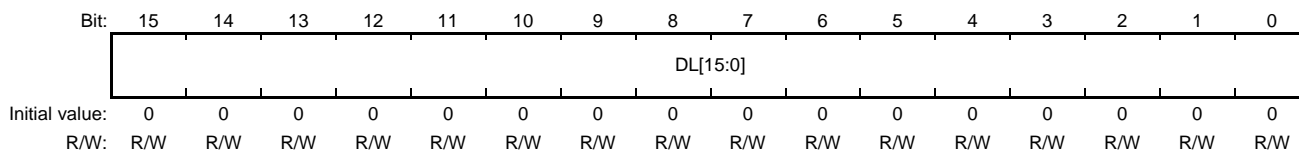
This register specifies the value of frequency division for the frequency divided clock generated by the BRG.

This register supports a 16-bit binary format that allows specifying a value in the range of 1 to 65535.

Setting all 0s in this register makes the BRG output the frequency divided clock at the low level.

The value of frequency division is given by the following formula:

$$\text{The value of frequency division} = (\text{clock input frequency}) / (\text{required baud rate} \times \text{sampling rate})$$



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 15 to 0 | DL[15:0] | All 0 | R/W | Specifies a division value of frequency clock generated in BRG. The setting value is enabled in the range of 1 to 65535. |

Table 51.8 and Table 51.9 show how to use the baud rate generator with a 3.6864-MHz crystal resonator, and a 26-MHz crystal resonator.

Table 51.8 Baud Rate (3.6864-MHz clock)

| Baud Rate | Value of Frequency Division | Sampling Rate | Error Rate* |
|-----------|-----------------------------|---------------|-------------|
| 50 | 4608 | 16 | — |
| 75 | 3072 | 16 | — |
| 110 | 4189 | 8 | -0.002 |
| 134.5 | 1713 | 16 | 0.001 |
| 150 | 1536 | 16 | — |
| 300 | 768 | 16 | — |
| 600 | 384 | 16 | — |
| 1200 | 192 | 16 | — |

| Baud Rate | Value of Frequency Division | Sampling Rate | Error Rate* |
|-----------|-----------------------------|---------------|-------------|
| 1800 | 128 | 16 | — |
| 2000 | 123 | 15 | 0.098 |
| 2400 | 96 | 16 | — |
| 3600 | 64 | 16 | — |
| 4800 | 48 | 16 | — |
| 7200 | 32 | 16 | — |
| 9600 | 24 | 16 | — |
| 14400 | 16 | 16 | — |
| 19200 | 12 | 16 | — |
| 38400 | 6 | 16 | — |
| 76800 | 3 | 16 | — |
| 115200 | 2 | 16 | — |

Note: * "—" Indicates that the error rate is 0.

Table 51.9 Baud Rate (26-MHz clock)

| Baud Rate | Value of Frequency Division | Sampling Rate | Error Rate* |
|-----------|-----------------------------|---------------|-------------|
| 9600 | 129 | 21 | 0.025 |
| 19200 | 52 | 26 | -0.160 |
| 38400 | 26 | 26 | -0.160 |
| 57600 | 15 | 30 | -0.309 |
| 115200 | 9 | 25 | -0.309 |
| 230400 | 4 | 28 | -0.756 |
| 460800 | 2 | 28 | -0.756 |
| 921600 | 1 | 28 | -0.756 |
| 1843200 | 1 | 14 | -0.756 |
| 3250000 | 1 | 8 | — |

Note: * "—" Indicates that the error rate is 0.

51.5.3.2 Clock Select Register (CKS)

This register switches the output between the frequency divided clocks and specifies a source clock for the external baud rate.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CKS | XIN | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 15 | CKS | B'0 | R/W | This bit switches the output between the frequency divided clock (SC_CLK) and external clock (HSCK). 0: Selects the frequency divided clock. 1: Selects the external clock. |
| 14 | XIN | B'0 | R/W | Selects the baud rate generator clock source for the external clock between SCIF_CLK and Internal clock. 0: Selects the external clock (SCIF_CLK). 1: Selects the internal clock (Internal clock). |
| 13 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

51.5.4 Restrictions in BRG

(1) Notes on setting frequency division register

1. For the initial setting of the register after a reset, at least one bit of waiting period is required to secure the clock stabilization time.

(Example) One bit period when DL = 2

$$3.68 \text{ (MHz)} \times 1/2 \times 1/16 = 0.115 \text{ (MHz)} \rightarrow 8695 \text{ (ns)}$$

2. For modifying the register value after the setting stated in <1> above, at least one bit of waiting period at the maximum bit rate (DL = '65535') is required.

The HSCIF registers and BRG registers should be set as the following table:

- Asynchronous mode (SC_CLK external input)

| HSCIF | | BRG | |
|----------------|---------------|---------------|---------------|
| Register Name | Setting Value | Register Name | Setting Value |
| HSSCR.CKE[1:0] | B'10 | CKS | H'0000 |
| | | DL | H'1 to H'FFFF |

- Asynchronous mode (HSCK external input)

| HSCIF | | BRG | |
|----------------|---------------|---------------|---------------|
| Register Name | Setting Value | Register Name | Setting Value |
| HSSCR.CKE[1:0] | B'10 | CKS | H'8000 |
| | | DL | Don't care |

- Asynchronous mode (SC_CLK internal input)

| HSCIF | | BRG | |
|----------------|---------------|---------------|---------------|
| Register Name | Setting Value | Register Name | Setting Value |
| HSSCR.CKE[1:0] | B'10 | CKS | H'4000 |
| | | DL | H'1 to H'FFFF |

3. The register settings for the baud rate generator for external clock should be made before starting initialization of the HSCIF.

52. I2C Bus Interface (I2C)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

52.1 Overview

52.1.1 Features

Second generation RZ/G series products have up to seven I2C bus interfaces conformant with the Philips Semiconductors (now NXP Semiconductors) I2C bus (Inter-IC bus) specification. However, the configuration of registers that control the I2C bus differs partly from the NXP Semiconductors register configuration. The I2C bus interface has the following features. Note that "module clock (MOD_CLK)" refers to the peripheral module clock in this section.

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]: module clock is S3D2 ϕ for channels I2C0/1/2 and S0D6 ϕ for channels I2C3/4/5/6. S3D2 ϕ and S0D6 ϕ have the same frequency (133.33 MHz).

[RZ/G2E]: module clock is S3D2 ϕ (133.33 MHz) for all channels.

- The I2C interfaces support the I2C bus specification from NXP Semiconductors
- Multi-master compatible
- Seven- or ten-bit address compatible master
- Seven-bit slave address
- Fast mode compatible
- Variable clock frequencies
- DMA transfer support for Master/Slave Transmission Master/Slave Reception.

Figure 52.1 shows a block diagram of the I2C bus interface.

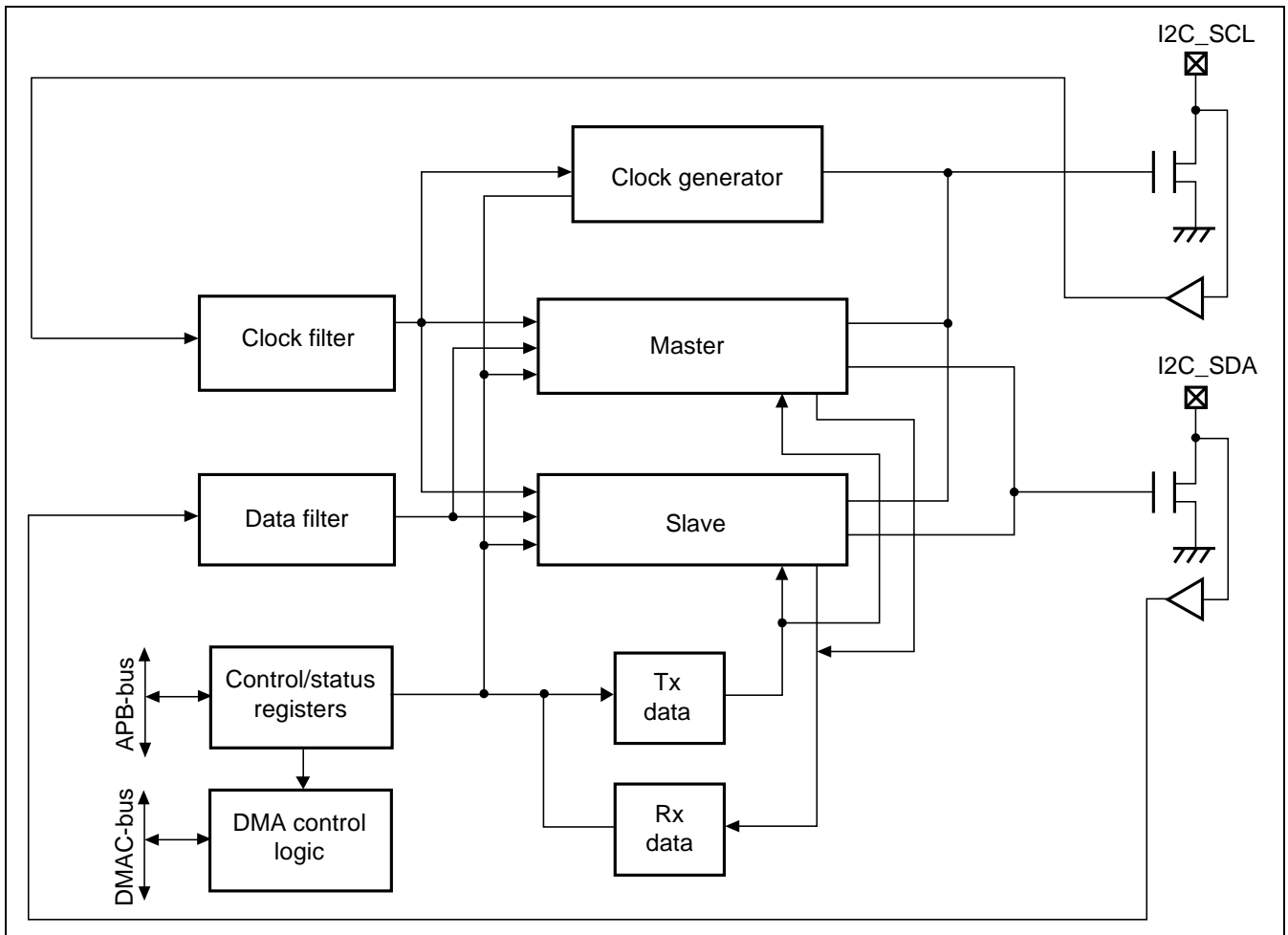


Figure 52.1 Block Diagram of I2C Bus Interface

52.1.2 External Pins

Table 52.1 lists the pins used in the I2C bus interface.

Table 52.1 I2C Bus Interface

| Channel Number | Pin Name *1 | I/O | Function | Second Generation RZ/G Series Products | | | |
|----------------|----------------|-----|--|--|----------------------------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| | | | | Output Buffer Type*2 | | | |
| 0 | SCL | I/O | SCL: I2C serial clock input/output pin | OD | OD | OD | OD |
| | SDA | | | | | | |
| 1 | SCL | | SDA: I2C serial data input/output pin | LVTTL | LVTTL | LVTTL | LVTTL |
| | SDA | | | | | | |
| 2 | SCL | | | LVTTL | LVTTL | LVTTL | LVTTL |
| | SDA | | | | | | |
| 3 | SCL | | | OD | OD | OD | OD |
| | SDA | | | | | | |
| 4 | SCL | | | OD | OD | OD | LVTTL |
| | SDA | | | | | | |
| 5 | SCL | | | OD | OD | OD | LVTTL |
| | SDA | | | | | | |
| 6 | SCL | | | LVTTL | LVTTL | LVTTL | LVTTL |
| | SDA | | | | | | |
| 7 | SCL | | | — | — | — | LVTTL |
| | SDA | | | | | | |

- Notes: 1. The actual pin names are I2Cn_SCL and I2Cn_SDA (n = 0 to 6 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, n = 0 to 7 for RZ/G2E). These are referred to as I2C_SCL and I2C_SDA in this section.
2. Output buffer type: "OD" is open drain buffer and "LVTTL" is low level drive only LVTTL buffer. OD assigned channel pins are 1.8 V I/O and input/output 3.3 V tolerant; LVTTL assigned channel pins are 3.3 V I/O.

The I2C buffer in this LSI is not compliant with the 5V-input. When turning off the I/O power source (3.3 V) of this LSI, turn off the power source of the pull-up resistors connected to the I2C pins.

Note: When using 1.8V I2C pins (OD assigned channel) as 3.3 V tolerant, all the external pull-up voltage power supply to the I2C of this LSI must keep the same power on/off sequence as the VCCQ (3.3 V) of this LSI.

52.1.3 Register Configuration

The base address for each channel of registers of the I2C bus interface is as follows:

- I2C0: H'E650_0000
- I2C1: H'E650_8000
- I2C2: H'E651_0000
- I2C3: H'E66D_0000
- I2C4: H'E66D_8000
- I2C5: H'E66E_0000
- I2C6: H'E66E_8000
- I2C7: H'E669_0000 (RZ/G2E)

Note: Number of I2C channels:

7 channels for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N (channels 0 to 6)

8 channels for RZ/G2E (channels 0 to 7)

Each channel of the I2C bus interface has the registers below.

Table 52.2 Register Configuration

| Register Name | Abbreviation | R/W | Offset Address from Base Address | Initial Value | Size | Second Generation RZ/G Series Products | | | |
|----------------------------------|--------------|---------|----------------------------------|---------------|------|--|-------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Slave control register | ICSCR | R/W | H'00 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Master control register | ICMCR | R/W | H'04 | H'0000_0000*3 | 32 | √ | √ | √ | √ |
| Slave status register | ICSSR | R/(W)*1 | H'08 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Master status register | ICMSR | R/(W)*2 | H'0C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Slave interrupt enable register | ICSIER | R/W | H'10 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Master interrupt enable register | ICMIER | R/W | H'14 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Clock control register | ICCCR | R/W | H'18 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Slave address register | ICSAR | R/W | H'1C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Master address register | ICMAR | R/W | H'20 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Receive data register | ICRXD | R | H'24 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Transmit data register | ICTXD | W | H'24 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Clock control register 2 | ICCCR2 | R/W | H'28 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SCL mask control register | ICMPR | R/W | H'2C | H'0000_0000 | 32 | √ | √ | √ | √ |
| SCL high control register | ICHPR | R/W | H'30 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SCL low control register | ICLPR | R/W | H'34 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DMA enable register | ICDMAER | R/W | H'3C | H'0000_0000 | 32 | √ | √ | √ | √ |
| First bit setup cycle register | ICFBSCR | R/W | H'38 | H'0000_0004 | 32 | √ | √ | √ | √ |

Notes: Do not write to any addresses other than listed above. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed above are undefined.

Although the actual register name and its abbreviation have channel number as a suffix respectively, it is omitted in this section (the number "2" of ICCCR2 is not channel number).

Operation cannot be guaranteed if the register is not accessed as a longword(32bit)

1. Only 0 can be written to bits 4 to 0 to clear the flags.
2. Only 0 can be written to bits 6 to 0 to clear the flags.
3. Bits 30, 29, 22, 21, 14, 13, 6, and 5 are undefined.

52.2 Register Description

Registers in the I2C are allocated to and arranged in the address space of the internal bus.

Legend for Register Description

Initial value: Register value after a reset

- : Undefined value
- R/W: Readable/writable. The written value can be read.
- R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.
- R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.
- R: Read-only. When the bit is reserved, the write value should always be 0. Writing 1 to these bits can cause a malfunction of I2C.
- /W: Write-only. The read value is undefined.

52.2.1 Slave Control Register (ICSCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|------|-----|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | SDBS | SIE | GCAE | FNA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved When these bits are read,, bits 31 to 24, 23 to 16, and 15 to 8 reflect the values of bits 7 to 0. The write value should always be 0. |
| 7 to 4 | — | All 0 | R | Reserved The write value should always be 0. |
| 3 | SDBS | B'0 | R/W | Slave Data Buffer Select This bit is used to select the data buffer. Select the single-buffer mode supported in this module. When this bit is set to 1, the single-buffer mode is selected. SCL will be held low from the timing when the receive data register acquires the data packet until the SDR flag is cleared. 0: Setting prohibited 1: Single-buffer mode |
| 2 | SIE | B'0 | R/W | Slave Interface Enable This bit must be set for the slave operation. If this bit is low, the slave interface is reset. |
| 1 | GCAE | B'0 | R/W | General Call Acknowledgement Enable When slave devices are to issue an acknowledgement in response to a general call address sent from a master, this bit must be set to 1. |
| 0 | FNA | B'0 | R/W | Forced Non Acknowledgement In the slave receive mode, the level of this bit is sent to the transmitting device as the acknowledge signal. This bit is set to 0 during the period that the data packet is being received, and set to 1 on completion of data reception. Forced non acknowledgement is returned to the master during slave reception. When the slave has received the last byte of data in a data packet, the slave communicates with the master by sending a NACK, meaning that the acknowledgement is not driven. The master issues a stop on the bus after receiving a NACK. The setting of this bit does not affect the acknowledgement of the slave address. |

52.2.2 Slave Status Register (ICSSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Bits 0 to 4 among the status bits in the slave status register are cleared by writing 0 to the respective status bit positions. The individual bits are held 1 until 0 is written to (other than the GCAR and STM bits).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|------|-----|------|------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | GCAR | STM | SSR | SDE | SDT | SDR | SAR |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W* | R/W* | R/W* | R/W* | R/W* |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|------|---|
| 31 to 8 | — | All 0 | R | Reserved When these bits are read, bits 31 to 24, 23 to 16, and 15 to 8 reflect the values of bits 7 to 0. The write value should always be 0. |
| 7 | — | B'0 | R | Reserved The write value should always be 0. |
| 6 | GCAR | B'0 | R | General Call Address Received Indicates that the address received from the bus is a general call address (H'00). This status bit does not cause an interrupt. This bit is automatically cleared by hardware when the SIE bit (bit 2 in the slave control register) is set to 0 or when the SSR bit (bit 4 in this register) is set to 1. |
| 5 | STM | B'0 | R | Slave Transmit Mode Indicates whether the current slave transmit mode is read or write. When this bit is set to 0, the mode is write (slave reception). When this bit is set to 1, the mode is read (slave transmission). This status bit does not cause an interrupt. This bit is automatically cleared by hardware when the SIE bit (bit 2 in the slave control register) is set to 0 or when the SSR bit (bit 4 in the slave status register) is set to 1. |
| 4 | SSR | B'0 | R/W* | Slave Stop Received A stop condition has been output on the bus. This status bit becomes active after the rising edge of SDA during reception of a stop bit. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 3 | SDE | B'0 | R/W* | <p>Slave Data Empty</p> <p>Indicates that data to be transmitted has been loaded into the shift register. At the start of byte data transmission, the contents of the ICTXD register are loaded into a shift register ready for outputting data on the bus. This status bit indicates that data has been loaded and the ICTXD register is again ready for further data. This status bit becomes active on the falling edge of SCL before reception of the first data bit.</p> <ul style="list-style-type: none"> - When bit SDE = 1 & bit TSDMAE = 0, the interrupt SDE is issued to CPU, so the transmit data register is to accept further data. - When bit SDE = 1 & bit TSDMAE = 1, enable DMA transfer request to DMAC. <p>During the single-buffer mode, this bit must be reset every time new data has been written to ICTXD. If not, the slave will hold SCL low after the current data transfer has finished to stop the bus.</p> <p>During DMA transmit mode, the SDE bit is also set after data is loaded into the shift register. In this case, this bit is cleared by DMA logic control. This mechanism to assure the continuous data transfer without software control. For the last byte data transmitted, this bit is cleared by software.</p> |
| 2 | SDT | B'0 | R/W* | <p>Slave Data Transmitted</p> <p>A byte of data has been transmitted to the bus. This bit becomes active after the falling edge of SCL during reception of the last data bit.</p> |
| 1 | SDR | B'0 | R/W* | <p>Slave Data Received</p> <p>A byte of data has been received from the bus and is ready for reading from the receive data register. This bit becomes active after the falling edge of SCL during reception of the last data bit. During the single-buffer mode, this bit must be reset after data has been read from the ICRXD register.</p> <ul style="list-style-type: none"> - When bit SDR = 1 & bit RSDMAE = 0, the interrupt SDR is issued to CPU, so the transmit data register is to accept further data. - When bit SDR = 1 & bit RSDMAE = 1, enable DMA transfer request to DMAC. <p>When SDBS is set to 1, SCL will be held low from the timing when the receive data register acquires the data packet until the SDR flag is cleared.</p> <p>During DMA receive mode, the SDR bit is also set after data is received into RXD register. In this case, this bit is cleared by DMA logic control. This mechanism to assure the continuous data transfer without software control.</p> |
| 0 | SAR | B'0 | R/W* | <p>Slave Address Received</p> <p>Indicates that the slave has recognized its own address on the bus (defined by the contents of the slave address register). If the general call acknowledgement enable bit is enabled in the slave control register, then this status bit is also set to 1 even if the address on the bus is a general call address. In this case, the GCAR bit in this register is used to determine whether or not the address is a general call address. The STM bit indicates whether the access is read (high) or write (low). This status becomes active after the falling edge of SCL during reception of the last address bit. The slave holds the SCL signal low from the start of the ACK phase until the software resets this status bit.</p> |

Note: * This bit can be read from or written to. Writing 0 clears this bit to 0 and writing 1 is ignored.

52.2.3 Slave Interrupt Enable Register (ICSIER)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----------|----------|----------|----------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | SSR E | SDE E | SDT E | SDR E | SAR E |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved When these bits are read, bits 31 to 24, 23 to 16, and 15 to 8 reflect the values of bits 7 to 0. The write value should always be 0. |
| 7 to 5 | — | All 0 | R | Reserved The write value should always be 0. |
| 4 | SSRE | B'0 | R/W | Slave Stop Received Interrupt Enable 0: Disables the SSR interrupt. 1: Enables the SSR interrupt. |
| 3 | SDEE | B'0 | R/W | Slave Data Empty Interrupt Enable 0: Disables the SDE interrupt. 1: Enables the SDE interrupt. |
| 2 | SDTE | B'0 | R/W | Slave Data Transmitted Interrupt Enable 0: Disables the SDT interrupt. 1: Enables the SDT interrupt. |
| 1 | SDRE | B'0 | R/W | Slave Data Received Interrupt Enable 0: Disables the SDR interrupt. 1: Enables the SDR interrupt. |
| 0 | SARE | B'0 | R/W | Slave Address Received Interrupt Enable 0: Disables the SAR interrupt. 1: Enables the SAR interrupt. |

52.2.4 Slave Address Register (ICSAR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | SADD0 _6 | SADD0 _5 | SADD0 _4 | SADD0 _3 | SADD0 _2 | SADD0 _1 | SADD0 _0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved When these bits are read, bits 31 to 24, 23 to 16, and 15 to 8 reflect the values of bits 7 to 0. The write value should always be 0. |
| 7 | — | B'0 | R | Reserved The write value should always be 0. |
| 6 to 0 | SADD0_6 to SADD0_0 | All 0 | R/W | Slave Address This is the unique 7-bit address allocated to the slave on the I2C bus. The slave interface compares this address with the first seven bits transmitted as the slave address, at the beginning of a data packet transmission. |

52.2.5 Master Control Register (ICMCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----------|----------|----------|----------|-----|----------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | — | — | 0 | 0 | 0 | 0 | 0 | 0 | — | — | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | MDB S | FSC L | FSD A | OBP C | MIE | TSB E | FSB | ESG |
| Initial value: | 0 | — | — | 0 | 0 | 0 | 0 | 0 | 0 | — | — | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---|-----|--|
| 31 to 8 | — | All 0, except bits 30, 29, 22, 21, 14, and 13, in which case, initial value is undefined. | R | Reserved When these bits are read, bits 31 to 24, 23 to 16, and 15 to 8 reflect the values of bits 7 to 0. The write value should always be 0. |
| 7 | MDBS | B'0 | R/W | Master Data Buffer Select This bit is used to select the data buffer mode. Select the single-buffer mode supported in this module. When this bit is set to 1, the single-buffer mode is selected. SCL will be held low from the timing when the receive data register acquires the data packet until the MDR flag is cleared. 0: Setting prohibited 1: Single-buffer mode |
| 6 | FSCL | — | R/W | Forced SCL This bit controls the status of the I2C_SCL pin (reading reflects the current status of the I2C_SCL pin). When the OBPC bit is set, this bit directly controls the SCL line on the bus. During a read cycle, the level on this bit (which includes the reset level) will change depending on the level on I2C_SCL since it directly reflects the level on the I2C_SCL. |
| 5 | FSDA | — | R/W | Forced SDA This bit controls the status of the I2C_SDA pin (reading reflects the busy status level on the I2C bus). When the OBPC bit is set, then this bit directly controls the SDA line on the bus. During a read cycle, the level on this bit (which includes the reset level) will show the busy status of the I2C bus (1 for busy; 0 for not busy). |
| 4 | OBPC | B'0 | R/W | Override Bus Pin Control When this bit is set to 1, the FSDA and FSCL bits in this register control SDA and SCL directly. |
| 3 | MIE | B'0 | R/W | Master Interface Enable When this bit is set to 1, the master interface is enabled. The master interface is disabled when this interface loses in arbitration of bus mastership even MIE bit indicates 1. If 0 is not subsequently written to the bit, the master interface is restored on detection of a STOP condition on the I2C bus. Regarding the state following a loss in arbitration of bus mastership, see the description of the MAL bit in ICMSR. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 2 | TSBE | B'0 | R/W | Start Byte Transmission Enable The write value for this bit should always be 0. |
| 1 | FSB | B'0 | R/W | Forced Stop onto the Bus When this bit is set to 1, the master transmits a STOP condition on the bus at the end of the current transfer. If ESG is also set, the master immediately transmits a START condition and begins transmitting a new data packet. If ESG is not set, the master enters the idle state. |
| 0 | ESG | B'0 | R/W | Enable Start Generation When this bit is set to 1, the master starts transmission of a data packet. If the bus is idle when ESG is set, the master transmits a START condition on the bus and then transmits the slave address. If the master is transferring data when ESG is set, at the end of that data byte transfer, the master transmits a repeated START condition before transmitting the slave address. When transmitting a data packet, the software must reset this bit when the slave address has been transmitted, otherwise a repeated START condition is transmitted after every transmission is completed. |

52.2.6 Master Status Register (ICMSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The status bits (bits 0 to 6) in the master status register are cleared by writing 0 to the respective status bit positions. The individual status bits are held 1 until a reset by writing 0 to the appropriate bit position.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|------|------|------|------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | MNR | MAL | MST | MDE | MDT | MDR | MAT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|------|--|
| 31 to 8 | — | All 0 | R | Reserved When these bits are read, bits 31 to 24, 23 to 16, and 15 to 8 reflect the values of bits 7 to 0. The write value should always be 0. |
| 7 | — | B'0 | R | Reserved The write value should always be 0 |
| 6 | MNR | B'0 | R/W* | Master NACK Received When this bit is set to 1, this bit indicates that the master has received a NACK response (the SDA line is high during the acknowledge cycle on the bus) to either an address or data transmission. To restart the address transmission from idle state on Master, this bit must be cleared. |
| 5 | MAL | B'0 | R/W* | Master Arbitration Lost In a multi-master system, when this bit is set to 1, it indicates that the master has lost arbitration to other masters on the bus. At this point, the master interface is disabled then Master enters the idle state again. To restart the address transmission from idle state on Master, this bit must be cleared. [except RZ/G2M V1.3, RZ/G2M V3.0] |
| 4 | MST | B'0 | R/W* | Master Stop Transmitted When this bit is set to 1, it indicates that the master has sent a STOP condition on the bus. A STOP condition can be sent either as a result of the setting of the forced stop bit in the control register, or from a NACK being received from a slave during a slave receive data packet. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|---|
| 3 | MDE | B'0 | R/W* | <p>Master Data Empty</p> <p>At the start of a byte data transmission, the contents of the transmit data register are loaded into a shift register ready for transmitting on the bus. This status bit becomes active:</p> <ul style="list-style-type: none"> - When bit MDE = 1, MAT = 0 & bit TMDMAE = 0, the interrupt MDE is issued to CPU, so the transmit data register is to accept further data. - When bit MDE = 1, MAT = 0 & bit TMDMAE = 1, enable DMA transfer request to DMAC. <p>During master transmission, the MDE bit is also set at the same time as the MAT bit is set after transmission of the slave address. In this case, you need to clear the MDE and MAT bits after the ICMCR's ESG bit is cleared. The clearing will restart the data transmission.</p> <p>During DMA transmit mode, the MDE bit is also set after data is loaded into the shift register. In this case, this bit is cleared by DMA logic control. This mechanism to assure the continuous data transfer without software control. For the last byte data transmitted, this bit cleared by software.</p> |
| 2 | MDT | B'0 | R/W* | <p>Master Data Transmitted</p> <p>Byte data has been sent to the slave on the bus. This status bit becomes active after the falling edge of SCL during reception of the last data bit.</p> |
| 1 | MDR | B'0 | R/W* | <p>Master Data Received</p> <p>Byte data has been received from the bus and is in the receive data register. This status bit becomes active after the falling edge of SCL during reception of the last data bit. During single-buffer mode, this status bit must be reset after data has been read from the receive data register.</p> <ul style="list-style-type: none"> - When bit MDR = 1, MAT = 0 & bit RMDMAE = 0, the interrupt MDR is issued to CPU, so the receive data register is to accept further data. - When bit MDR = 1, MAT = 0 & bit RMDMAE = 1, enable DMA transfer request to DMAC. <p>When MDBS is set to 1, SCL will be held low from the timing when the receive data register acquires the data packet until the MDR flag is cleared.</p> <p>During master reception mode, the MDR bit is also set at the same time as the MAT bit is set after transmission of the slave address. In this case, you must clear the MDR and MAT bits after the ICMCR's ESG bit is cleared. Clearing will start the data reception.</p> <p>During DMA receive mode, the MDR bit is also set after data is received into RXD register. In this case, this bit is cleared by DMA logic control. This mechanism to assure the continuous data transfer without software control.</p> |
| 0 | MAT | B'0 | R/W* | <p>Master Address Transmitted</p> <p>The master has been transmitted the slave address byte of a data packet. This bit becomes active after the falling edge of SCL during reception of the ACK bit following transmission of the address.</p> |

Note: * This bit can be read from or written to. Writing 0 clears this bit to 0 and writing 1 is ignored.

52.2.7 Master Interrupt Enable Register (ICMIER)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----------|----------|----------|----------|----------|----------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | MNR E | MAL E | MST E | MDE E | MDT E | MDR E | MAT E |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved When these bits are read, bits 31 to 24, 23 to 16, and 15 to 8 reflect the values of bits 7 to 0. The write value should always be 0. |
| 7 | — | B'0 | R | Reserved The write value should always be 0. |
| 6 | MNRE | B'0 | R/W | Master NACK Received Interrupt Enable 0: Disables the MNR interrupt. 1: Enables the MNR interrupt. |
| 5 | MALE | B'0 | R/W | Master Arbitration Lost Interrupt Enable 0: Disables the MAL interrupt. 1: Enables the MAL interrupt. |
| 4 | MSTE | B'0 | R/W | Master Stop Transmitted Interrupt Enable 0: Disables the MST interrupt. 1: Enables the MST interrupt. |
| 3 | MDEE | B'0 | R/W | Master Data Empty Interrupt Enable 0: Disables the MDE interrupt. 1: Enables the MDE interrupt. |
| 2 | MDTE | B'0 | R/W | Master Data Transmitted Interrupt Enable 0: Disables the MDT interrupt. 1: Enables the MDT interrupt. |
| 1 | MDRE | B'0 | R/W | Master Data Received Interrupt Enable 0: Disables the MDR interrupt. 1: Enables the MDR interrupt. |
| 0 | MATE | B'0 | R/W | Master Address Transmitted Interrupt Enable 0: Disables the MAT interrupt. 1: Enables the MAT interrupt. |

52.2.8 Master Address Register (ICMAR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | SADD1 _6 | SADD1 _5 | SADD1 _4 | SADD1 _3 | SADD1 _2 | SADD1 _1 | SADD1 _0 | STM1 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved When these bits are read, bits 31 to 24, 23 to 16, and 15 to 8 reflect the values of bits 7 to 0. The write value should always be 0. |
| 7 to 1 | SADD1_6 to SADD1_0 | All 0 | R/W | Slave Address These bits are the address of the slave which the master communicates with. |
| 0 | STM1 | B'0 | R/W | Slave Transfer Mode This bit specifies the mode in which the slave operates. Bit STM1 sets the operating mode (transmit or receive mode) of the slave, which is an external slave device whose address matches the slave address (SADD1) sent from the master. The slave device is automatically set to transmit/receive mode by hardware on reception of the STM1 signal. When this bit is set to 1, it indicates a read operation. When this bit is cleared to 0, it indicates a write operation. |

52.2.9 Clock Control Register (ICCCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|------|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | SCGD | | | | | CDF | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 9 | — | All 0 | R | Reserved The read value is always 0. The write value should always be 0. |
| 8 to 3 | SCGD | All 0 | R/W | <p>SCL Clock Generation Divider</p> <p>When operation is in master mode, the SCL clock is generated from the internal clock by using SCGD as the ratio. This is also the operating clock in slave-mode operation while SCL is being held low to stop the bus after an overflow has occurred. SCGD must be specified in both master and slave modes. The formula expressing the relationship is:</p> <p>Equation 2: SCL rate calculation</p> $SCLfreq = I2Cck / (20 + SCGD \times 8 + F[(tICF + tr + IntDelay) \times I2Cck])$ <p>I2Cck: I2C internal clock frequency tICF: I2C SCL falling time (depending on external load) tr: I2C SCL rising time (depending on external load) IntDelay: LSI internal delay corresponds to output buffer type. Open drain buffer: 50 ns (typ.), 110 ns (max.) LVTTTL (low drive only) buffer: 5 ns (typ.), 6 ns (max.) F[n]: n rounded down to an integer</p> <p>Suggested settings for CDF and SCGD for CPU speeds and the two I2C bus speeds are given in Table 52.3.</p> |
| 2 to 0 | CDF | All 0 | R/W | <p>Clock Division Factor</p> <p>The internal clock used in most blocks in the I2C module is a divided module clock. The internal I2C clock is generated from the module clock using the CDF as the division ratio:</p> <p>Equation 1: I2C internal clock frequency calculation</p> $I2Cck = MOD_CLK / (1 + CDF)$ <p>MOD_CLK: module clock frequency</p> <p>Ensure that minimum values for setup and hold times of the SDA signal relative to the SCL signal on the bus are satisfied.</p> <p>The clock frequency is to ensure that the glitch filtering will operate with glitches of up to 50 ns as described in the fast mode I2C specification.</p> <p>Note: CDF must be set so that the clock frequency (I2Cck) is lower than 20 MHz.</p> |

Table 52.3 Recommended Settings for CDF and SCGD*1

- Recommended register values for open drain buffer

| Module Clock Frequency | 100 kHz | | 400 kHz | |
|------------------------|---------|----------|---------|----------|
| | CDF(*2) | SCGD(*3) | CDF(*2) | SCGD(*3) |
| 133.33 MHz | 6 | 21 | 6 | 3 |
| Error | | -0.80% | | -0.80% |

- Recommended register values for LVTTTL buffer (low drive only)

| Module Clock Frequency | 100 kHz | | 400 kHz | |
|------------------------|---------|----------|---------|----------|
| | CDF(*2) | SCGD(*3) | CDF(*2) | SCGD(*3) |
| 133.33 MHz | 6 | 21 | 6 | 3 |
| Error | | -0.28% | | +1.31% |

Notes: 1. These recommended values are for the SCL rate with an external load that produces $t_{ICF} = 3$ ns and $t_r = 150$ ns.

Recalculate to obtain the actual values after measuring t_{ICF} and t_r under actual conditions of usage.

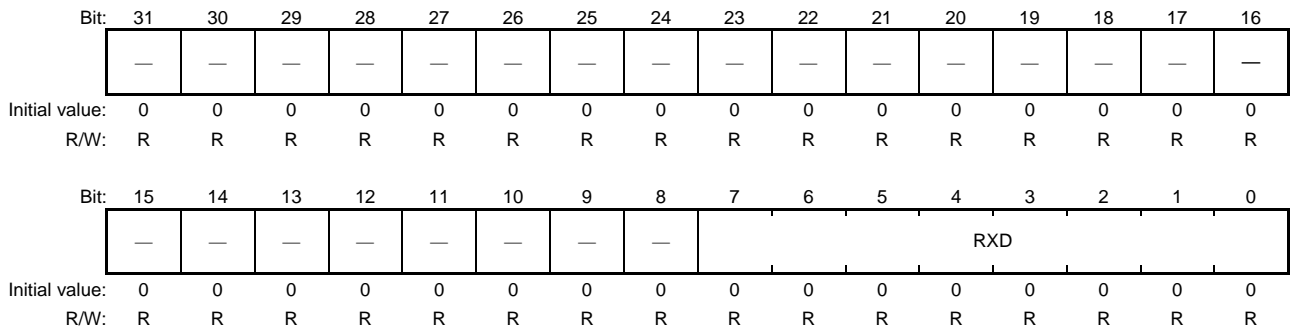
- Recommended value for CDF bit of ICCCR.
- Recommended value for SCGD bit of ICCCR.

52.2.10 Receive and Transmit Data Registers (ICRXD and ICTXD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

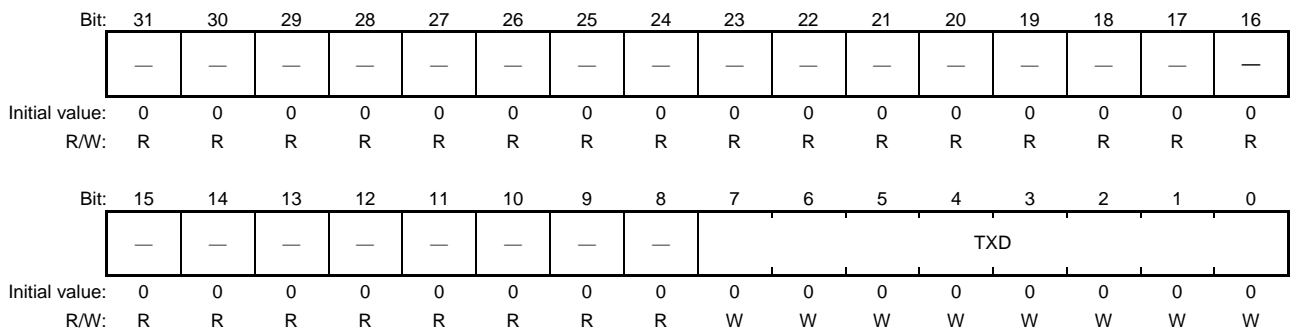
Reading from or writing to these registers access different physical internal registers. When data is to be transmitted, the contents of the shift register are loaded via TXD. After data has been received into the shift register from the I2C bus, it is then loaded into RXD.

(1) Receive data register (ICRXD)



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved When these bits are read, bits 31 to 24, 23 to 16, and 15 to 8 reflect the values of bits 7 to 0. The write value should always be 0. |
| 7 to 0 | RXD | All 0 | R | Read Receive Data Data received by master or slave. |

(2) Transmit data register (ICTXD)



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved The write value should always be 0. |
| 7 to 0 | TXD | All 0 | W | Write Transmit Data Data to be transmitted by master or slave. |

52.2.11 Clock Control Register 2 (ICCCR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----------|----------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | CDF D | HLS E | SME |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved When these bits are read, the value in bits 7 to 0 is set to bits 31 to 24, bits 23 to 16, and bits 15 to 8. The write value should always be 0. |
| 7 to 3 | — | All 0 | R | Reserved The write value should always be 0. |
| 2 | CDFD | B'0 | R/W | CDF Disable When this bit is set to 1, the module clock is used for internal clocks except the clock filter. For SCL rate computation, see equation 3 in Note 2. The clock filter always operates at a clock frequency divided by the CDF value. |
| 1 | HLSE | B'0 | R/W | HIGH/LOW Separate Control Enable When this bit is set to 1, the SCGD setting is ignored and SCL is generated with the division ratio set by ICHPR during a high period and with the division ratio set by ICLPR during a low period. For SCL rate computation, see equation 3 in Note 2. The program should be prepared so that ICHPR and ICLPR should be written to before setting this bit. |
| 0 | SME | B'0 | R/W | SCL Mask Enable When this bit is set to 1, a change of internal SCL is ignored after an SCL external input pin changes during the time equal to the number of internal clock cycles specified by ICMR. For SCL rate computation, see equation 3 in Note 2. The program should be prepared so that ICMR should be written to before setting this bit. |

Notes: 1. The control bits should be set to "CDFD = 0, HLSE = 0, and SME = 0" or "CDFD = 1, HLSE = 1, and SME = 1". Otherwise, operation cannot be guaranteed. ICCCR2 should be written to prior to writing ICCCR.

2. Equation 3: SCL Rate Computation (CDFD = 1, HLSE = 1, and SME = 1)

$$\text{SCL freq} = \text{MOD_CLK} / (8 + 2 \times \text{SMD} + \text{SCLD} + \text{SCHD} + F[(\text{tICF} + \text{tr} + \text{IntDelay}) \times \text{MOD_CLK}])$$

MOD_CLK: Module Clock Frequency

SMD: SCL masked period (set by the SCL mask control register)

SCLD: I2C SCL low clock period (set by the SCL HIGH control register)

SCHD: I2C SCL high clock period (set by the SCL LOW control register)

tICF: I2C SCL falling time (depending on external load)

tr: I2C SCL rising time (depending on external load)

IntDelay: LSI internal delay corresponds to output buffer type.

Open drain buffer: 50 ns (typ.), 110 ns (max.)

LVTTL (low drive only) buffer: 5 ns (typ.), 6 ns (max.).

F[n]: n rounded down to an integer

Table 52.4 shows recommended register values for two types of I2C bus speeds.

Table 52.4 Recommended Register Values when CDFD = 1, HLSE = 1, and SME = 1

- Recommended Register Values for open drain buffer

| Module Clock Frequency | 100 kHz | | | | | 400 kHz | | | | |
|---------------------------|-------------|--------------|-------------|--------------|--------------|-------------|--------------|-------------|--------------|--------------|
| | CDF (*1) | SCGD (*2) | SMD (*3) | SCHD (*4) | SCLD (*5) | CDF (*1) | SCGD (*2) | SMD (*3) | SCHD (*4) | SCLD (*5) |
| 133.33 MHz | 6 (*6) | 0 | 20 | 632 | 653 | 6 (*6) | 0 | 21 | 133 | 150 |
| Error | | | -1.53% | | | | | -5.84% | | |

- Recommended Register Values for LVTTTL buffer (low drive only)

| Module Clock Frequency | 100 kHz | | | | | 400 kHz | | | | |
|---------------------------|-------------|--------------|-------------|--------------|--------------|-------------|--------------|-------------|--------------|--------------|
| | CDF (*1) | SCGD (*2) | SMD (*3) | SCHD (*4) | SCLD (*5) | CDF (*1) | SCGD (*2) | SMD (*3) | SCHD (*4) | SCLD (*5) |
| 133.33 MHz | 6 (*6) | 0 | 14 | 639 | 658 | 6 (*6) | 0 | 16 | 138 | 155 |
| Error | | | -0.43% | | | | | -1.67% | | |

Note: $t_{ICF} = 20\text{ns}$, $t_r = 35\text{ns}$, and $\text{IntDelay} = 110\text{ns}$ (open drain buffer)/ 6ns (LVTTTL) are used to calculate Error value.

Notes: 1. Recommended value for CDF bit of ICCCR.

2. Recommended value for SCGD bit of ICCCR.

3. Recommended value for SMD bit of ICMPR.

4. Recommended value for SCHD bit of ICHPR.

5. Recommended value for SCLD bit of ICLPR.

6. In case of connecting to device, whose $t_{HD, DAT}$ is smaller than 100ns, specifying 0 as CDF is recommended.

52.2.12 SCL Mask Control Register (ICMPR)

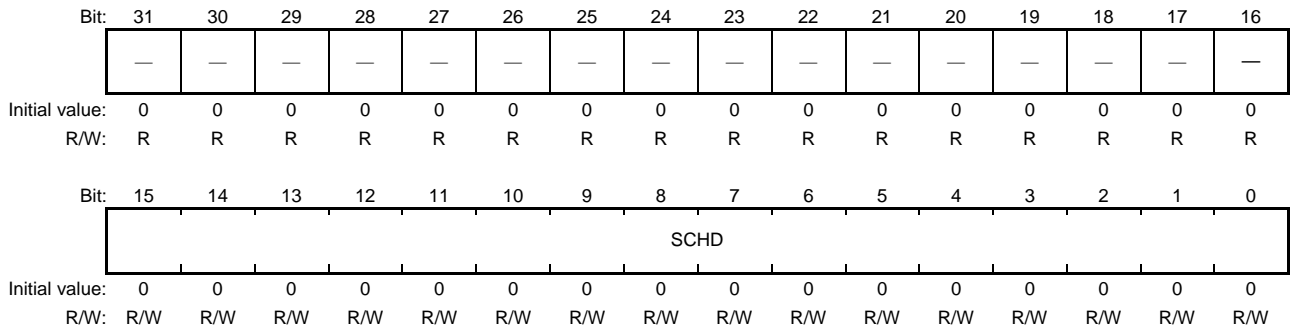
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | SMD | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved When these bits are read, the value in bits 7 to 0 is set to bits 31 to 24, bits 23 to 16, and bits 15 to 8. The write value should always be 0. |
| 7 to 0 | SMD | All 0 | R/W | SCL Mask Division When SME = 1, a change of internal SCL is ignored after the external SCL changes during the time equal to the number of internal clock cycles specified by SMD. The value set in SMD should be smaller than SCLD and SCLD. |

52.2.13 SCL HIGH Control Register (ICHPR)

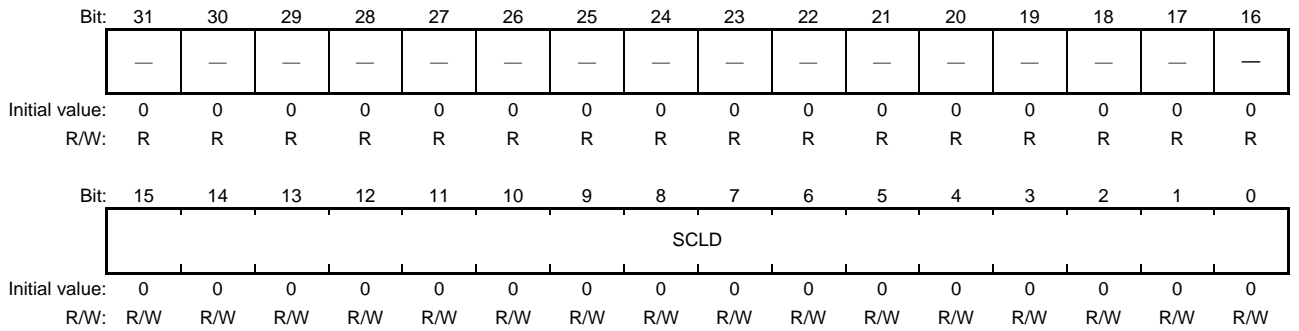
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved The write value should always be 0. |
| 15 to 0 | SCHD | All 0 | R/W | SCL HIGH Clock Division When HLSE is 1 and internal SCL is driven high, the clock generated using the SCHD internal clock division ratio is output. The period over the SCL signal is driven high can be calculated by the formula below. Period over SCL is driven high = 1/MOD_CLK × (4 + SMD + SCHD) |

52.2.14 SCL LOW Control Register (ICLPR)

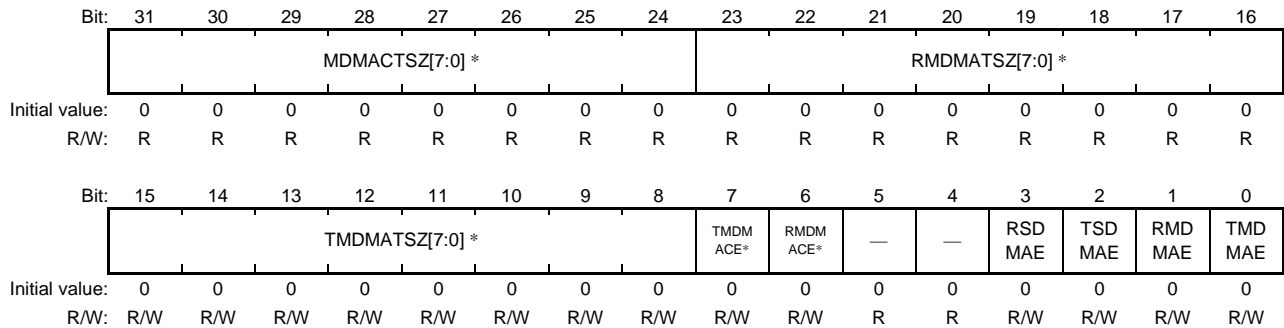
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved The write value should always be 0. |
| 15 to 0 | SCLD | All 0 | R/W | SCL LOW Clock Division When HLSE is 1 and internal SCL is driven low, the clock generated using the SCLD internal clock division ratio is output. The period over the SCL signal is driven low can be calculated by the formula below. Period over SCL is driven low = 1/MOD_CLK × (4 + SMD + SCLD) |

52.2.15 DMA Enable Register (ICDMAER)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



Note: * RZ/G2E is assigned R/W, other products are read only.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|---|--|
| 31 to 24 | MDMACTSZ | All 0 | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Reserved When these bits are read, bits 31 to 24, 23 to 16, and 15 to 8 reflect the values of bits 7 to 0. The write value should always be 0. |
| | | | [RZ/G2E] R/W | [RZ/G2E] DMA Master Continuous mode Transfer Size Specify the number of transfer counts. Transfer counts equals the number of DMAC descriptor number for this transfer. This register should be set same value of DMACHCRB.DCNT[7:0] for this DMA transfer. Minimum number is H'00. (Use 1 descriptor only) Maximum number is H'FF. (Use 256 descriptor) |
| 23 to 16 | RMDMATSZ | All 0 | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Reserved When these bits are read, bits 31 to 24, 23 to 16, and 15 to 8 reflect the values of bits 7 to 0. The write value should always be 0. |
| | | | [RZ/G2E] R/W | [RZ/G2E] DMA Master Continuous Received mode 1unit Transfer Size register. Specify byte number of 1 unit transfer size. For example, Start/End bit is send every 4byte, specify H'04. Minimum size setting is "H'01". (Transfer of 1 unit is Once) Maximum size setting is "H'00". (Transfer of 1 unit is 256 times) |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|---|--|
| 15 to 8 | TMDMATSZ | All 0 | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Reserved When these bits are read, bits 31 to 24, 23 to 16, and 15 to 8 reflect the values of bits 7 to 0. The write value should always be 0. |
| | | | [RZ/G2E] R/W | [RZ/G2E] DMA Master Continuous Transmitted mode 1unit Transfer Size register. Specify byte number of 1 unit transfer size. For example, Start/End bit is send every 4byte, specify H'04. Minimum size setting is "H'01". (Transfer of 1 unit is Once) Maximum size setting is "H'00". (Transfer of 1 unit is 256 times) |
| 7 | TMDMACE | B'0 | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Reserved The write value should always be 0. |
| | | | [RZ/G2E] R/W | [RZ/G2E] DMA Master Continuous Transmitted Enable 0: Disables DMA Continuous Transmitted mode 1: Enables DMA Continuous Transmitted mode |
| 6 | RMDMACE | B'0 | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Reserved The write value should always be 0. |
| | | | [RZ/G2E] R/W | [RZ/G2E] DMA Master Continuous Received Enable 0: Disables DMA Continuous Received mode 1: Enables DMA Continuous Received mode |
| 5 to 4 | — | All 0 | R | Reserved The write value should always be 0. |
| 3 | RSDMAE | B'0 | R/W | DMA Slave Received Enable 0: Disables DMA Slave Received Mode. 1: Enables DMA Slave Received Mode. |
| 2 | TSDMAE | B'0 | R/W | DMA Slave Transmitted Enable 0: Disables DMA Slave Transmitted Mode. 1: Enables DMA Slave Transmitted Mode. |
| 1 | RMDMAE | B'0 | R/W | DMA Master Received Enable 0: Disables DMA Master Received Mode. 1: Enables DMA Master Received Mode. |
| 0 | TMDMAE | B'0 | R/W | DMA Master Transmitted Enable 0: Disables DMA Master Transmitted Mode. 1: Enables DMA Master Transmitted Mode. |

52.2.16 First Bit Setup Cycle Register (ICFBSCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|------------|------------|------------|------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | FBS C_4 | FBS C_3 | FBS C_2 | FBS C_1 | FBS C_0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved When these bits are read, bits 31 to 24, 23 to 16, and 15 to 8 reflect the values of bits 7 to 0. The write value should always be 0. |
| 7 to 5 | — | All 0 | R | Reserved The write value should always be 0. |
| 4 to 0 | FBSC_4 to FBSC_0 | B'0_0100 | R/W | Setting the delay time of the 1 st data bit between SDA and SCL |

Note: The delay time of the 1st data bit between SDA and SCL was calculated as below table:

Table 52.5 First Bit Setup Cycle Value

| 1 st Bit setup Cycle | FBSC's value | 1 st Bit setup Cycle | FBSC's value |
|---------------------------------|--------------|---------------------------------|--------------|
| 02 × Tcyc | B'0_0000 | 18 × Tcyc | B'1_0000 |
| 03 × Tcyc | B'0_0001 | 19 × Tcyc | B'1_0001 |
| 04 × Tcyc | B'0_0010 | 20 × Tcyc | B'1_0010 |
| 05 × Tcyc | B'0_0011 | 21 × Tcyc | B'1_0011 |
| 06 × Tcyc | B'0_0100 | 22 × Tcyc | B'1_0100 |
| 07 × Tcyc | B'0_0101 | 23 × Tcyc | B'1_0101 |
| 08 × Tcyc | B'0_0110 | 24 × Tcyc | B'1_0110 |
| 09 × Tcyc | B'0_0111 | 25 × Tcyc | B'1_0111 |
| 10 × Tcyc | B'0_1000 | 26 × Tcyc | B'1_1000 |
| 11 × Tcyc | B'0_1001 | 27 × Tcyc | B'1_1001 |
| 12 × Tcyc | B'0_1010 | 28 × Tcyc | B'1_1010 |
| 13 × Tcyc | B'0_1011 | 29 × Tcyc | B'1_1011 |
| 14 × Tcyc | B'0_1100 | 30 × Tcyc | B'1_1100 |
| 15 × Tcyc | B'0_1101 | 31 × Tcyc | B'1_1101 |
| 16 × Tcyc | B'0_1110 | 32 × Tcyc | B'1_1110 |
| 17 × Tcyc | B'0_1111 | 33 × Tcyc | B'1_1111 |

Note: In Normal mode, Tcyc is period of I2Cck.
In Duty variable mode, Tcyc is period of MOD_CLK.

52.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

52.3.1 Data and Clock Filters

These blocks filter out glitches on signals coming from the I2C bus. Glitches up to one internal clock period in width are rejected (for details on the internal clock frequency, see section 52.2.9, Clock Control Register (ICCCR)). This is for the faster I2C bus rate (400 kHz) but does not violate the slower I2C bus rate specification.

These blocks also resynchronize bus signals with the internal clock.

52.3.2 Clock Generator

The clock generator has two functions. Firstly, it generates the SCL (I2C bus clock) according to commands of the master or slave interface. Secondly, it controls the internal clock rate, used by filtering blocks and the master and slave interfaces. This clock functions as a clock enable signal of the registers in these blocks.

52.3.3 Master/Slave Interfaces

These two interfaces run independently and in parallel. The master interface controls the transmission of address and data on the I2C bus. The slave interface monitors the I2C bus and takes part in transmissions if its programmed address is seen on the bus. The interfaces communicate with the control/status registers independently. There is only one interrupt line output from the I2C module. The interrupt source is either the master or the slave.

52.3.4 Software Status Interlocking

To make the software interface with the I2C module as robust and thus simple as possible, various interlocking status mechanisms have been included in the operation of the master and slave interfaces. The status bits involved are:

(1) MDR and SDR

MDR or SDR is set to 1 when data is received. Clear the status after reading the receive data register. If data is received while MDR or SDR is set, the hardware recognizes that unread data remains in the receive data register and automatically holds SCL at the low level to suspend data transmission. In this case, transmission can be resumed by clearing the status after reading the receive data.

Consequently, when receiving data continuously, be sure to clear the status bit (MDR or SDR) after reading the receive data register.

(2) MDE and SDE

If the MDE or SDE status bit is still set when data in the transmit data register is to be transmitted on the I2C bus by the slave or master, the SCL line must be held low until the MDE or SDE status bit is reset. The MDE or SDE status bit being set indicates that the data currently held in the transmit data register has already been transmitted on the I2C bus.

The software must clear this status bit when it writes to the transmit data register after transmission of the next data byte has become possible. This is not required for the first byte of data to be transmitted on the bus.

(3) MAL

When the master loses arbitration, the MAL bit (of the master status register) is set and the master interface is disabled. At this point, master mode is invalid and the I2C bus interface enters the slave mode. When master interface is restored, MAL bit must be cleared before the address re-transmission. [except RZ/G2M V1.3, RZ/G2M V3.0].

(4) SAR

The SAR status bit is set when the slave identifies its address on the I2C bus. At this point the slave interface forces the SCL line low until the SAR status bit is reset.

This is particularly important when a slave transmit is about to take place on the bus, and the slave will transmit the data from the transmit data register. The software responds to the SAR status by writing the required data into the transmit data register and then resetting the SAR status bit. This allows the slave interface to continue the access.

Unless the SAR bit is in use, when the slave is about to receive data, the software might still be reading out data that was loaded in previous access from the receive data register. In this case the valid data still held in the receive data register is overwritten. However, this is avoided using the SAR status bit. After the software has read data in the receive data register, reset the SAR bit (if it is set). Then overwriting the receive data register is avoided.

52.3.5 I2C Bus Data Format

Figure 52.2 shows a timing chart for the I2C bus interface. Table 52.6 describes the meaning of each symbol in Figure 52.2.

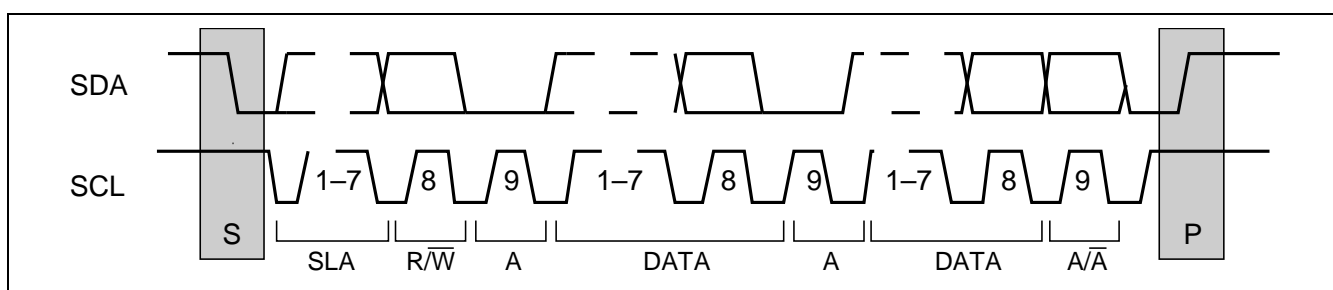


Figure 52.2 I2C Bus Timing

Table 52.6 Description on Symbols of I2C Bus Data Format

| Symbol | Description |
|--------|---|
| S | Indicates a start condition. A master device changes SDA from high to low while SCL is high level. |
| SLA | Indicates a slave address. A slave address is used when a master device selects a slave device. |
| R/W | Indicates the direction of data transmission. If the R/W bit is 1, the data flows from the slave to the master device. If the bit is 0, the data flows from the master to the slave device. |
| A | Indicates data acknowledge. Data receiving device makes SDA low level (the slave device returns a data acknowledge signal in master transmission mode, and vice versa). |
| DATA | Indicates transmit or receive data. The data length is eight bits, which are transferred in the MSB first. |
| P | Indicates a stop condition. A master device changes SDA from low to high while SCL is high. |

52.3.6 7-Bit Address Format

Figure 52.3 shows the format of data transfer from a master to a slave device (master data transmit format). Figure 52.4 shows the data transfer format (master data receive format) when a master device reads the second and the following byte data from a slave device.

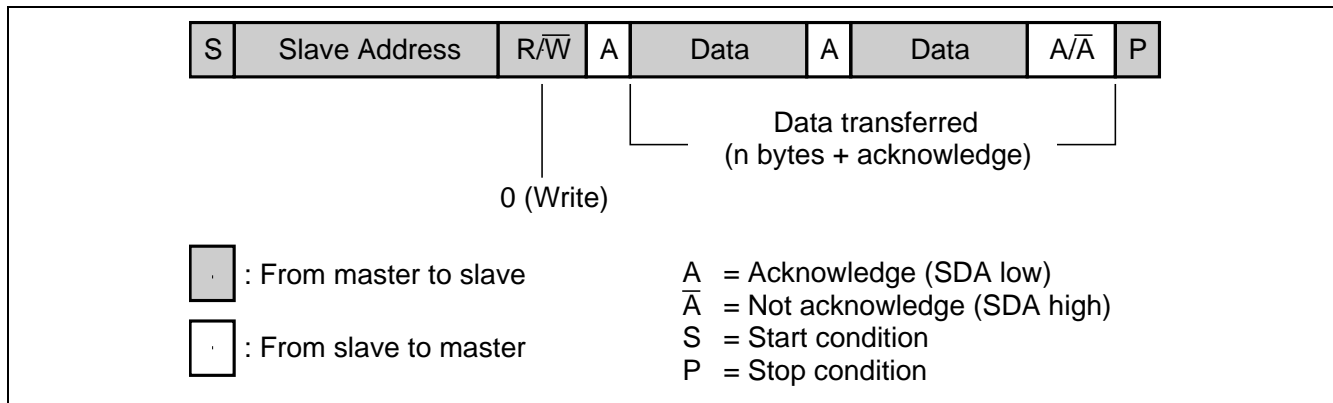


Figure 52.3 Master Data Transmit Format

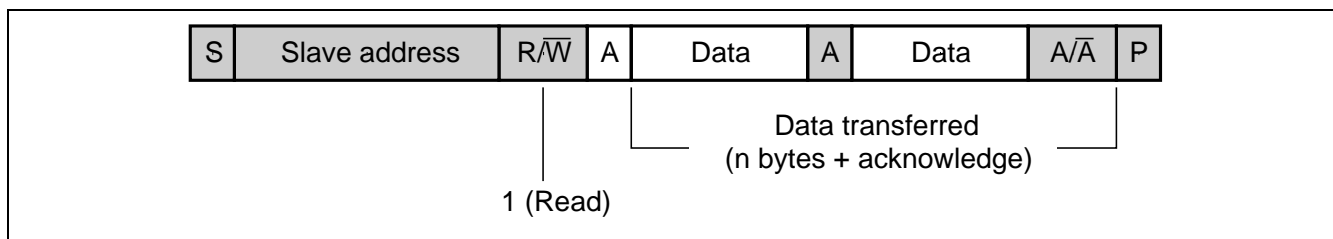


Figure 52.4 Master Data Receive Format

Figure 52.5 shows the combined format when the data transfer direction changes during one transfer. When changing the direction after the first transfer, the repeated START condition (Sr), slave address and R/W bits are transmitted. In this case, the R/W bit is set to the direction opposite to the first transfer direction. The repeated START condition is issued by the master at the end of a transmit or receive cycle if the enable start generation bit in the master control register has been set.

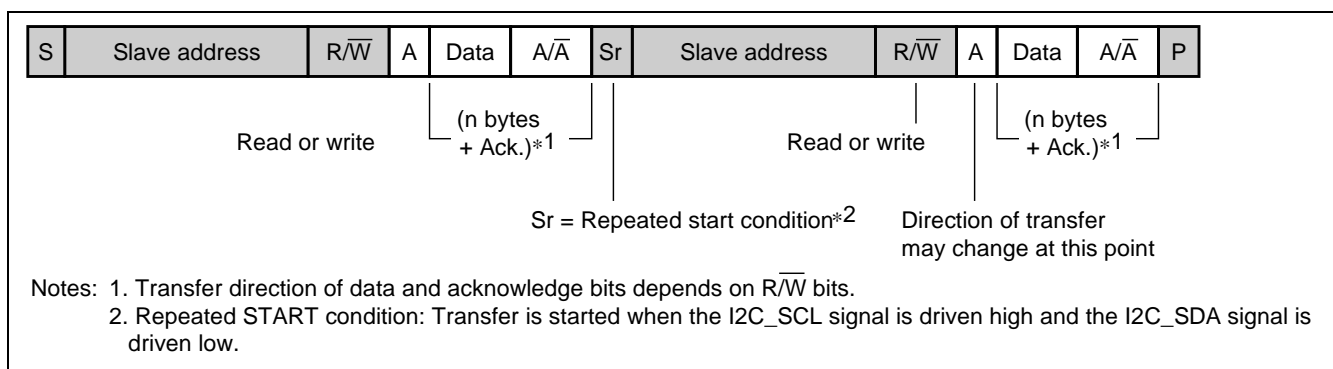


Figure 52.5 Combination Transfer Format of Master Transfer

52.3.7 10-Bit Address Format

Description is given below on the 10-bit address transfer format supported in master mode. This format has three transfer methods as the 7-bit address transfer format.

Figure 52.6 shows the data transmit format. The set value in the master address register is output in one byte following the first START condition (S). The value set in the transmit data register (TXD) is transmitted as a slave address in the second byte. Data on and after the third byte is transferred in the same way as the 7-bit address data.

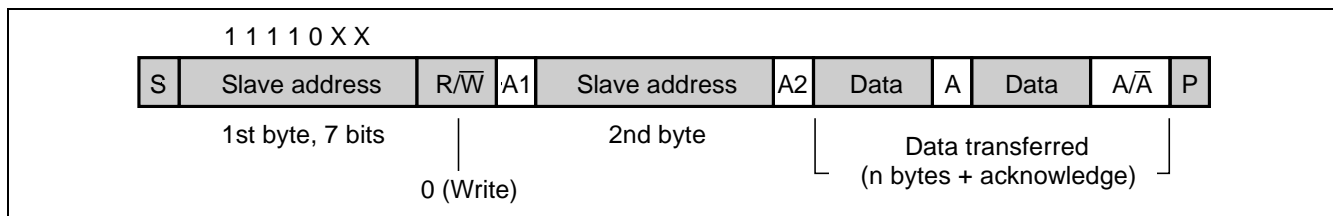


Figure 52.6 10-Bit Address Data Transmit Format

Figure 52.7 shows the data receive format. Two bytes of an address is transmitted in the same way as in the data transmit format. Then, repeated START condition (Sr) is transmitted and the value set in the address register is output. At this time, STM1 must be set to 1 (receive mode). Data is transferred in the same way as in the 7-bit address data receive format.

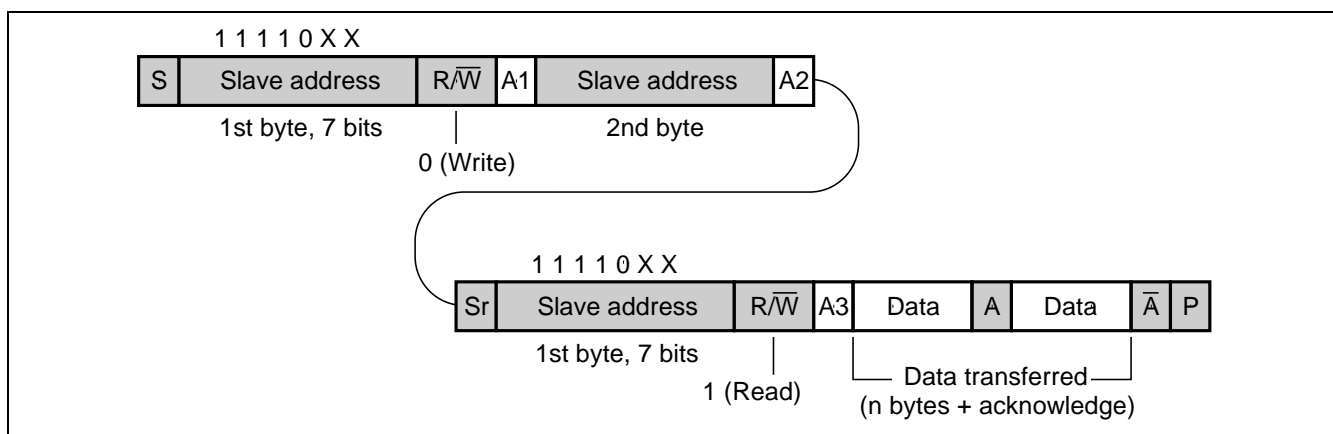


Figure 52.7 10-Bit Address Data Receive Format

Figure 52.8 shows the data transmit/receive combined format.

In the data transmit/receive combined format, data is transmitted after an address is transmitted with the first two bytes. Then, the repeated START condition (Sr) is transmitted instead of STOP condition (P). After Sr is transmitted, the procedure is the same as that in the data receive format.

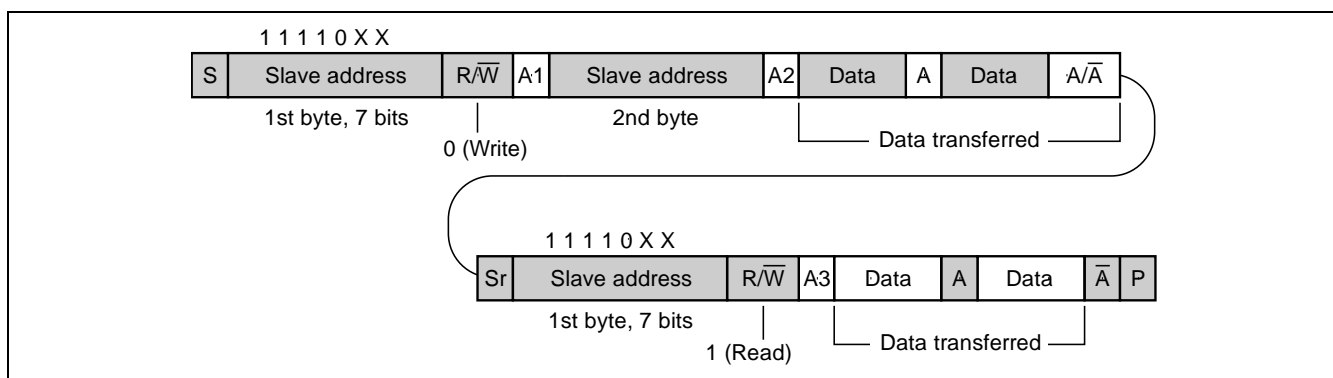


Figure 52.8 10-Bit Address Transmit/Receive Combined Format

52.3.8 Master Transmit Operation

The procedure and operations for transmission in master transmit mode are described below. Figure 52.9 shows the timing chart in master transmit mode. Setting the ICMCR.MDBS bit allows the I2C to operate in single-buffer mode.

1. Set up the initial state by setting the ICMAR, ICTXD, ICCCR, and ICMIER. Note that the setting of the clock control register must be in accord with the rate of transmission.
2. Monitor the ICMCR.FSDA. Confirm that this bit is low, meaning that other I2C devices are not using the bus. After confirmation, set the ICMCR.MIE (bit 3) and ICMCR.ESG (bit 0) bits to 1 to start master transmission.
3. After the transmit START condition, slave address, and data transfer direction bits are transmitted, an interrupt due to the ICMSR.MAT and ICMSR.MDE bits is generated at the timing of (1) in Figure 52.9. At this time, clear the ICMCR.ESG bit to 0. To suspend the data transmission, the master device will hold SCL low until the ICMSR.MDE bit is cleared.
4. Data is transmitted in units of nine bits: 8-bit data and 1-bit ACK. An interrupt of MDE (bit 3) is generated at the ninth clock before data transfer (at the timing of (2) in Figure 52.9). An interrupt of ICMSR.MDT (bit 2) is generated at the eighth clock after 1-byte data transfer (at the timing of (3) in Figure 52.9, Figure 52.10, Figure 52.11). Clear ICMSR.MDE to 0 after setting transmit data.
5. When this I2C receives NACK from the slave device, ICMSR.MNR is set to 1. After that, this I2C makes STOP condition and ICMSR.MST is set to 1.
6. To end of transmission, need to set ICMCR.FSB = 1.
ICMCR.FSB must be set by MDE interruption at final byte data transmission.
ICMSR.MST (master stop transmitted, at the timing of (4) in Figure 52.9) is checked by interruption or polling.

Signal level changes of (1) to (6) in Figure 52.9 are generated after the falling edge of the clock.

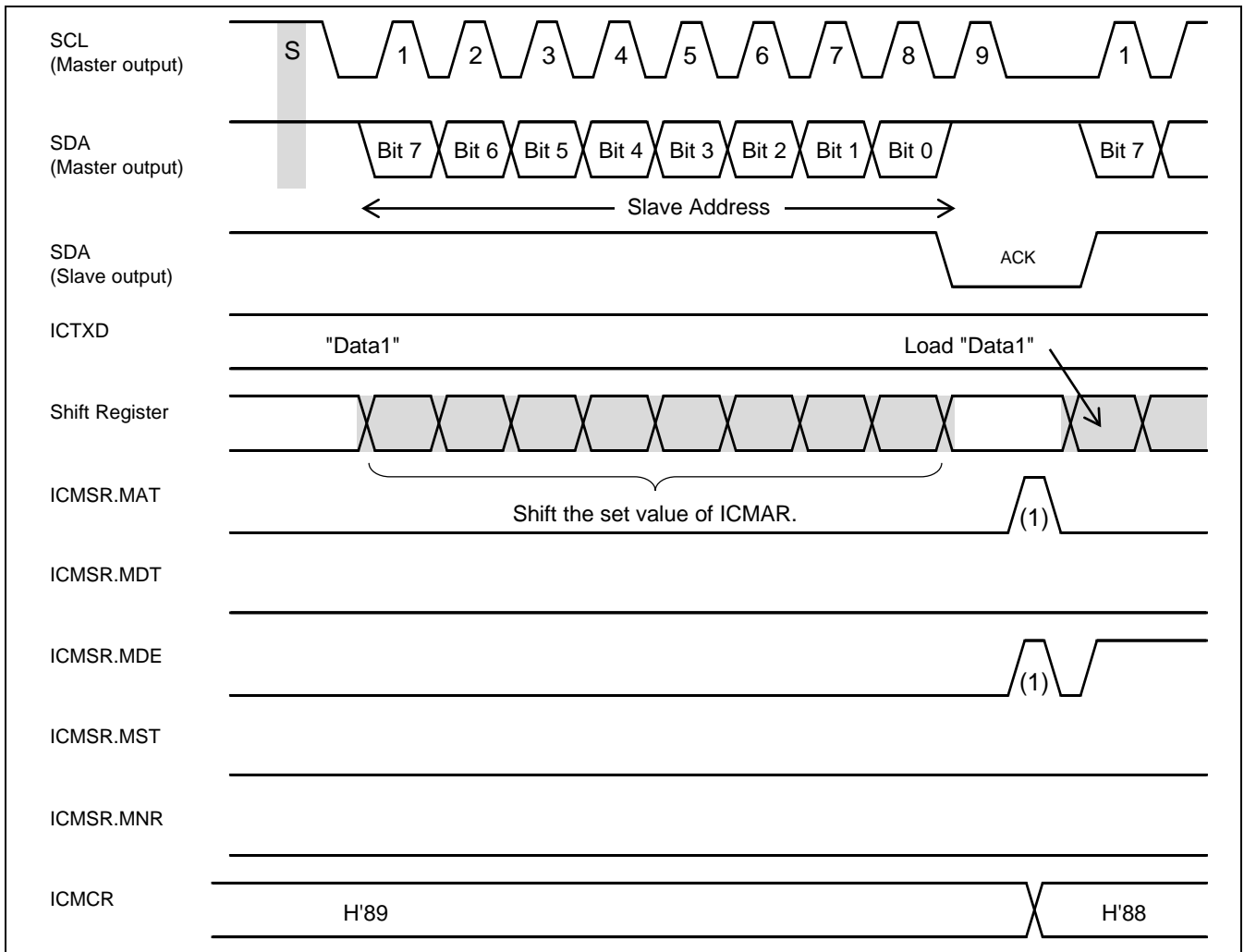


Figure 52.9 Data Transmit Mode Operation Timing

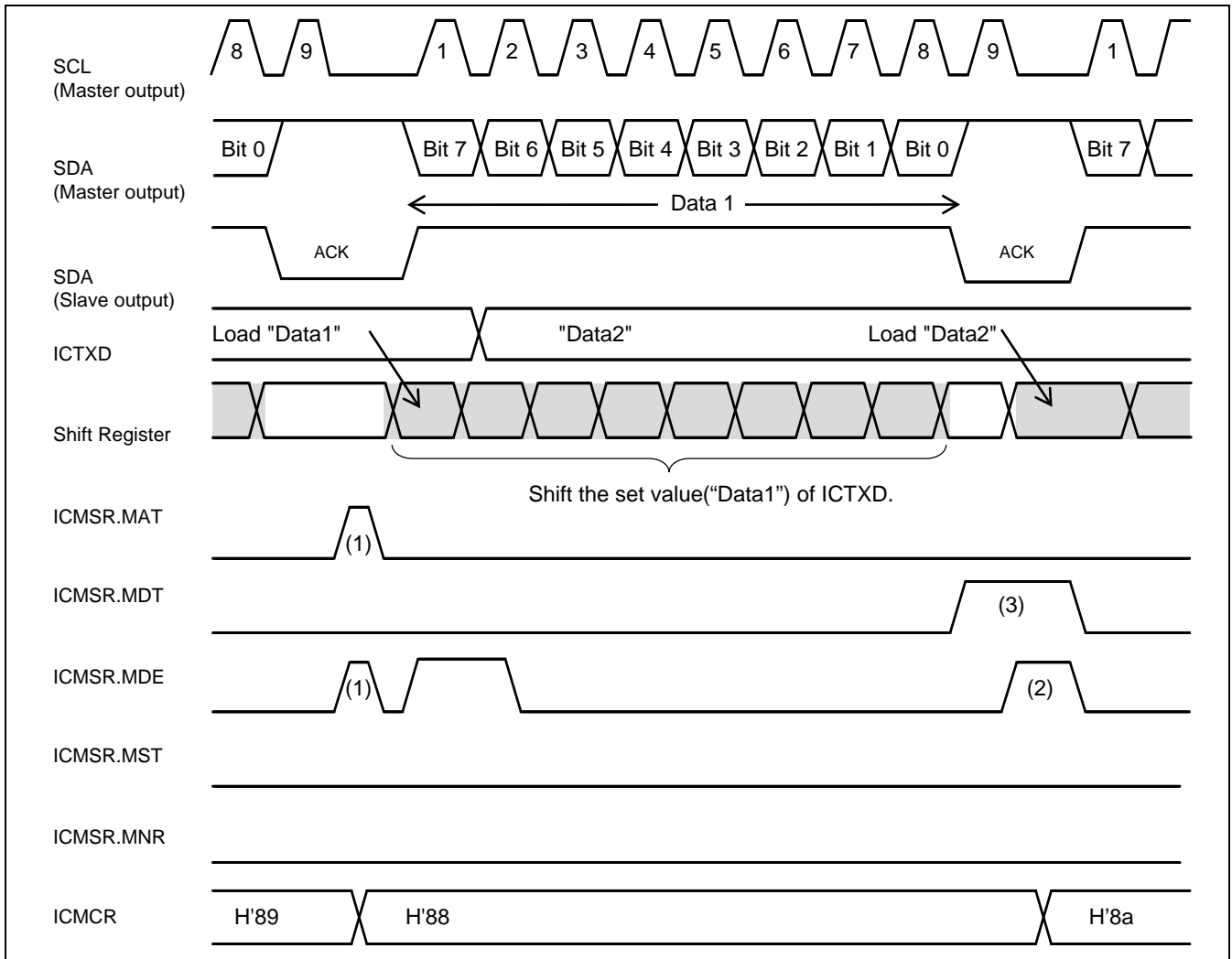


Figure 52.10 Data Transmit Mode Operation Timing

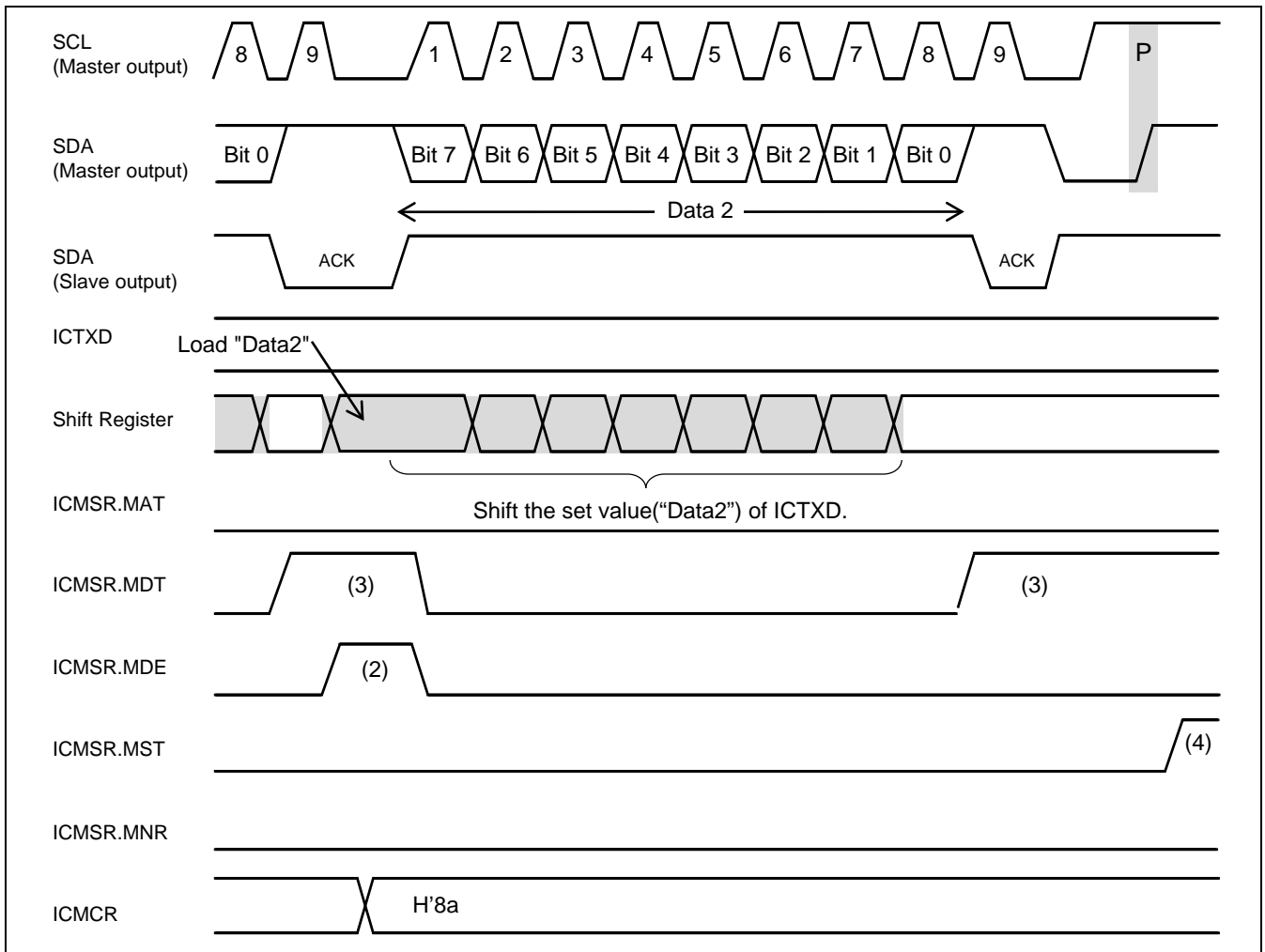


Figure 52.11 Data Transmit Mode Operation Timing

52.3.9 Master Receive Operation

The data receive procedure and operation in master receive mode are described below. Figure 52.12 shows the timing chart in master receive mode. Setting the ICMCR.MDBS bit allows the I2C to operate in single-buffer mode.

1. In master receive mode, as to transmit of a slave address and a 1-bit signal indicating the data transfer direction, operation is the same as that in master transmit mode. At this time, set the data transfer direction to 1 (reception).
2. The slave device automatically enters the data transmit mode according to the signal that indicates the data transfer direction, and transmits 1-byte data in synchronization with the SCL clock output from the master device. The master device generates an interrupt of ICMSR.MDR (bit 1) at the eighth clock (at the timing of (1) in Figure 52.12). Clear the ICMSR.MDR bit after the master device reads ICRXD. If this processing is delayed, the master device extends the SCL period to suspend data transmission (at the timing of (2) in Figure 52.12).
3. To end data transfer, set ICMCR.FSB (bit 1) of the master device and output STOP condition. In order to stop the communication after predetermined number of byte data is transferred, ICMCR.FSB bit needs to be set before the last byte data transfer is started. After confirming reception of the last byte, even when the master receiver has completed the reception transaction.

Signal level changes of (1) to (3) in Figure 52.12 and Figure 52.13 are generated after the falling edge of the clock.

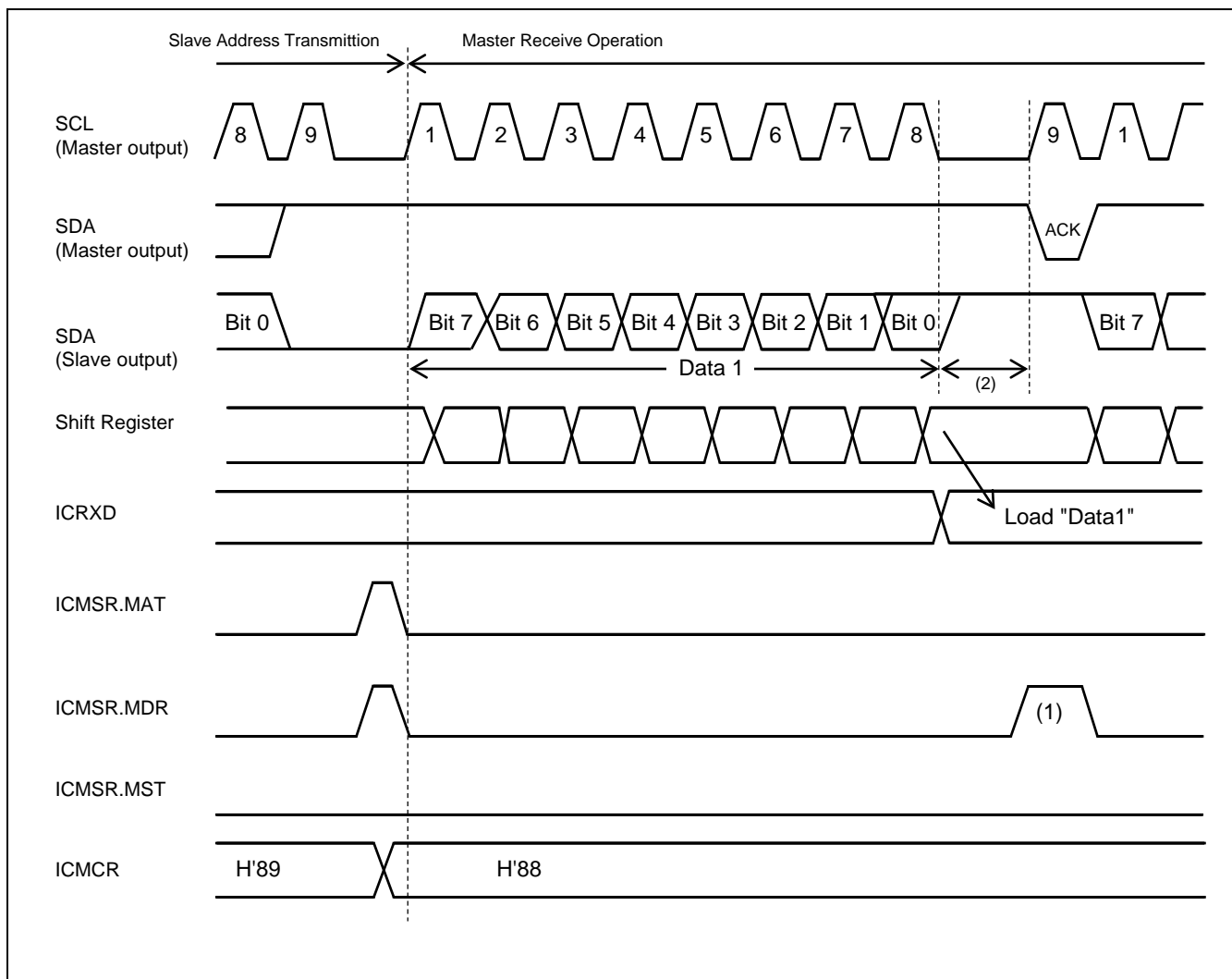


Figure 52.12 Data Receive Mode Operation Timing

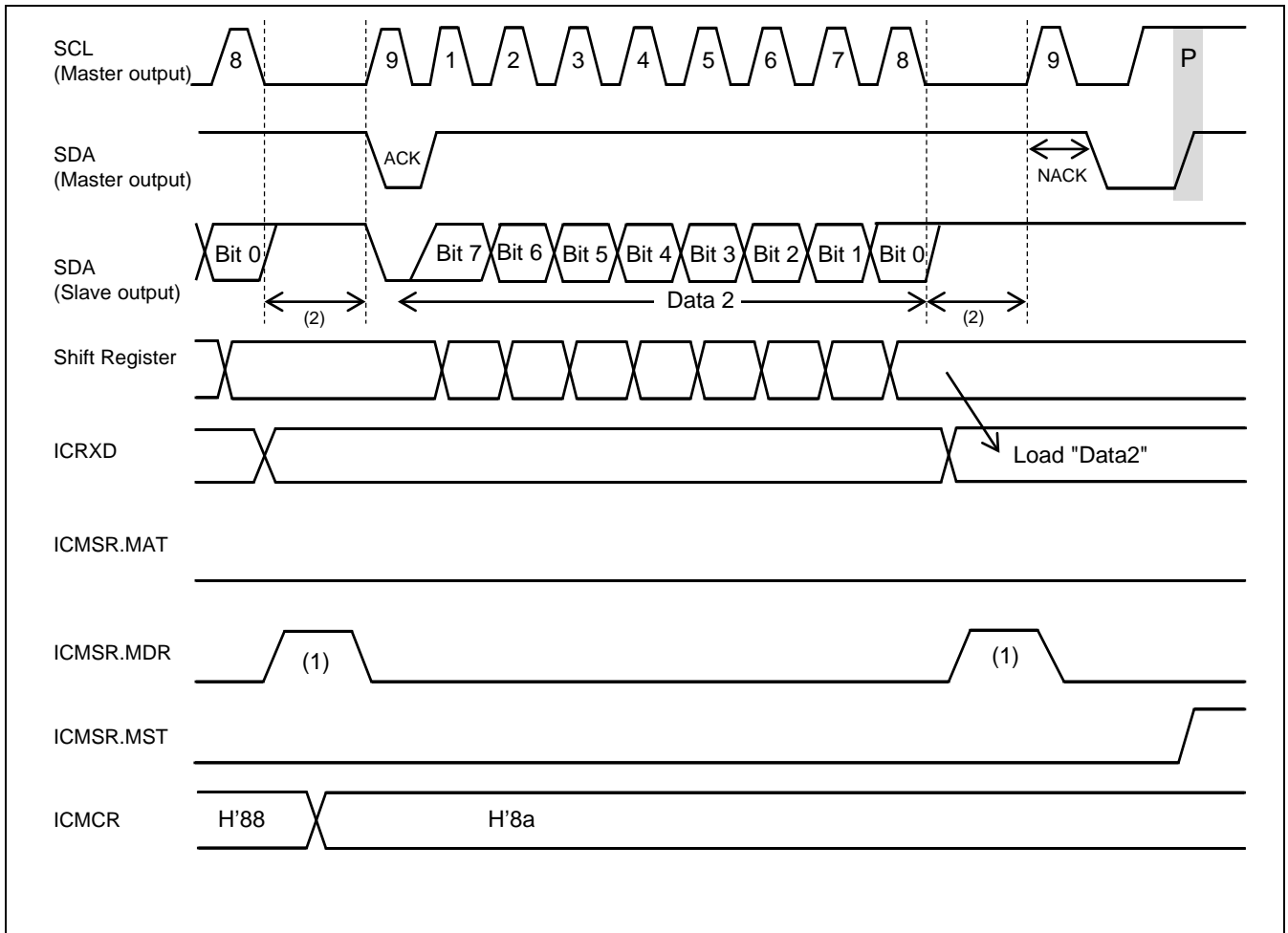


Figure 52.13 Data Receive Mode Operation Timing

52.4 Programming Examples

52.4.1 Master Transmitter

In order to set up the master interface to transmit a data packet on the I2C bus, follow the procedure below (an example for I2C0):

(1) Set value for the clock control register

1. Reset I2C with SRCR and SRSTCLR.
2. Set the Clock Control register 2 (ICCCR2) = H'07 (CDFD = 1, HLSE = 1, SME = 1).
3. Set the Clock Control register (ICCCR) = H'06 (SCGD = 0, CDF = 6, I2C internal clock I2Cck: 19 MHz).
– Setting ICCCR, ICCCR2 for using Duty ratio fixed or Duty ratio variable. The case for using Variable Duty ratio, have to setting ICCCR2 too. The case for using Fixed Duty ratio, have to setting ICCCR only.
4. Set the SCL Mask Control register (ICMPR) = H'1C (Variable Duty ratio only).
5. Set the SCL High Control register (ICHPR) = H'73 (Variable Duty ratio only).
6. Set the SCL Low Control register (ICLPR) = H'85 (Variable Duty ratio only).
7. Set the First Bit Setup Cycle register (ICFBSCR) = H'0F (1st bit setup cycle = 17 × Tcyc).

(2) Set value for the master control register, first data byte, and address

1. Clear the Master Status register (ICMSR) = H'00 (all 0 clear).
2. Set the Master Interrupt Enable register (ICMIER) = H'09 (MDEE = 1, MATE = 1).
3. Set the Master Address register (ICMAR) = Slave device address + H'00 (Write mode).
4. Set the Transmit Data register (ICTXD) = 1st transmission data.
5. Set the Master Control register (ICMCR) = H'89 (MDBS = 1, MIE = 1, ESG = 1).

(3) Wait for outputting address

1. Wait for master event (an interrupt of the ICMSR.MAT and ICMSR.MDE).
2. Set ICMCR to H'88 (to suspend the data transmission, the master device will hold the SCL low until the MDE bit is cleared).
— If only one byte of data is transmitted, set the ICMCR to H'8A, meaning that the stop generation is enabled. This generates a stop on the bus as soon as one byte has been transmitted.
3. Clear the ICMSR.MAT & ICMSR.MDE bits.

(4) Monitor transmission of data

1. Wait for master event, ICMSR.MDE.
2. ICTXD = subsequent data.
3. Clear the ICMSR.MDE bit.

Clear ICMSR.MDE after setting the last byte to be transmitted. After the last byte data is loaded into the shift register, ICMSR.MDE is generated. Before clearing ICMSR.MDE, you must set ICMSR to H'8A (set the force stop control bit).

(5) Wait for end of transmission

1. Wait for the master event, ICMSR.MST.
2. Clear the ICMSR.MST bit.

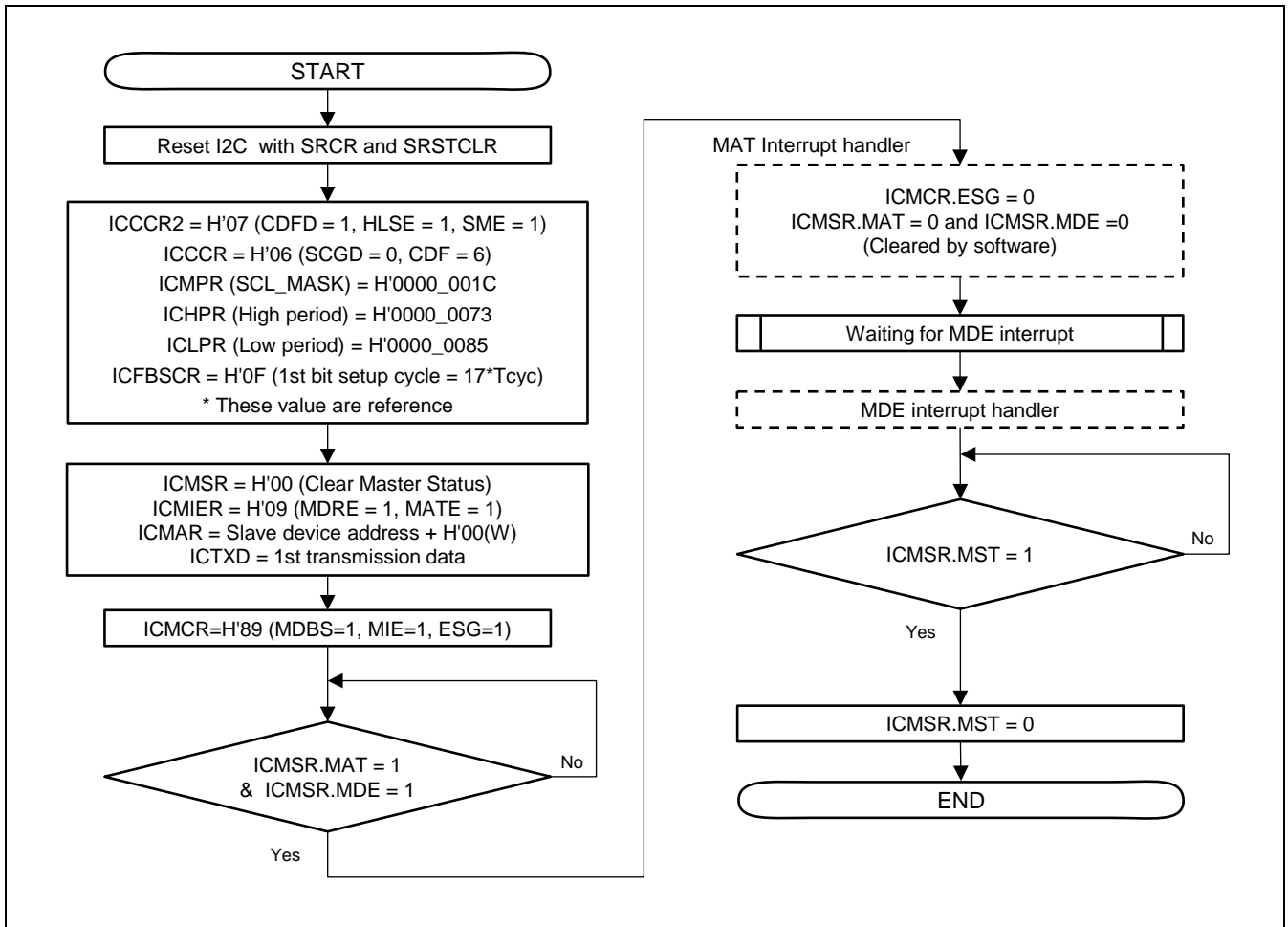


Figure 52.14 Master Transmitter Operation Setting

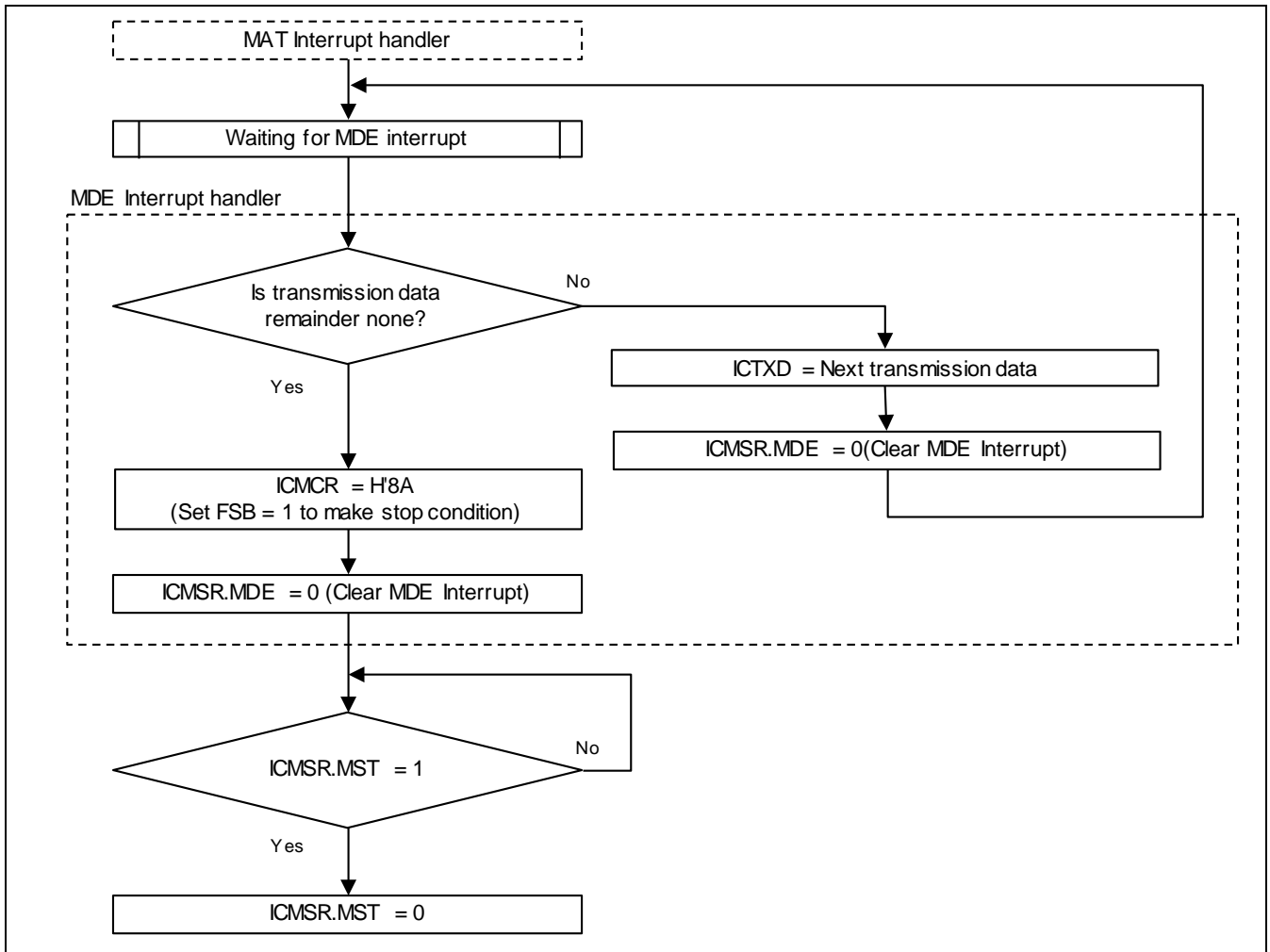


Figure 52.15 MDE Interrupt Handler for Master Transmitter Operation Setting

52.4.2 Master Receiver

In order to set up the master interface to receive a data packet on the I2C bus, follow the procedure below:

(1) Set value for the clock control register

The same procedure as the master transmitter is applied. See the program example for the master transmitter.

(2) Set value for the master control register and address

1. Clear the Master Status register (ICMSR) = H'00 (all 0 clear)
2. Set the Master Interrupt Enable register (ICMIER) = H'03 (MDRE = 1, MATE = 1)
3. Set the Master Address Register (ICMAR) = Slave device address + H'01 (Read mode)
4. Set the Master Control Register (ICMCR) = H'89 (MDBS = 1, MIE = 1, ESG = 1)

(3) Wait for outputting address

1. Wait for master event (an interrupt of the ICMSR.MAT and ICMSR.MDR).
2. Set the ICMCR to H'88 (to suspend the data transmission, the master device will hold the SCL low until the MDR bit is cleared).
 - If only one byte of data is transmitted, set the ICMCR to H'8A, meaning that the stop generation is enabled. This generates a stop on the bus as soon as one byte has been received.
3. Clear the ICMSR.MAT & ICMSR.MDR bits.

(4) Monitor reception of data

1. Wait for master event, ICMSR.MDR.
2. Clear the MDR bit after reading data from the ICRXD.
3. Set the ICMCR to H'8A (set the ICMCR.FSB) before the last byte data transfer is started.

(5) Wait for end of reception

1. Handle the receive interrupt (ICMSR.MDR) in the last byte: that is, read the data and clear the ICMSR.MDR.
2. Wait for master event, ICMSR.MST.
3. Clear the ICMSR.MST bit.

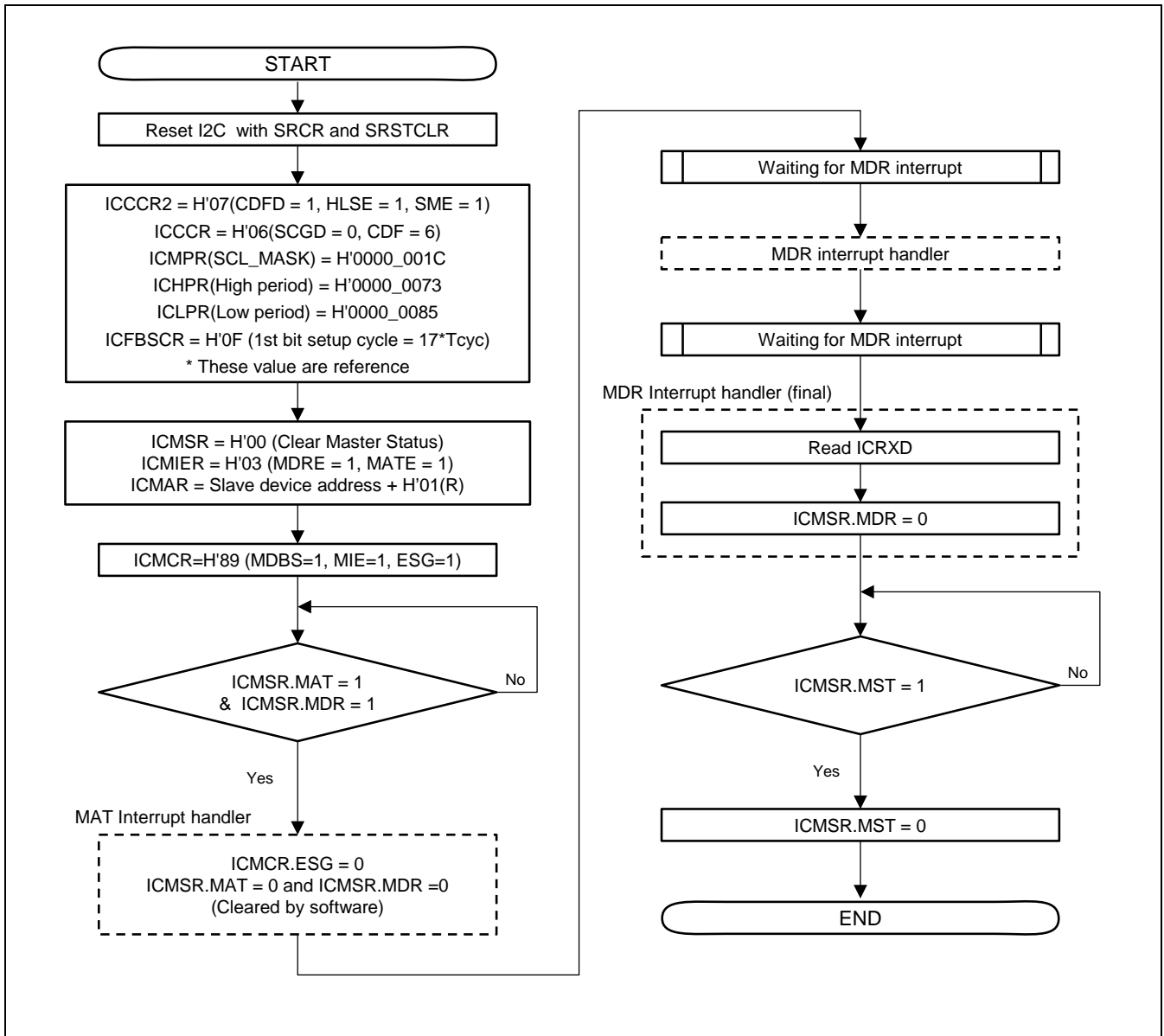


Figure 52.16 Master Receiver Operation Setting

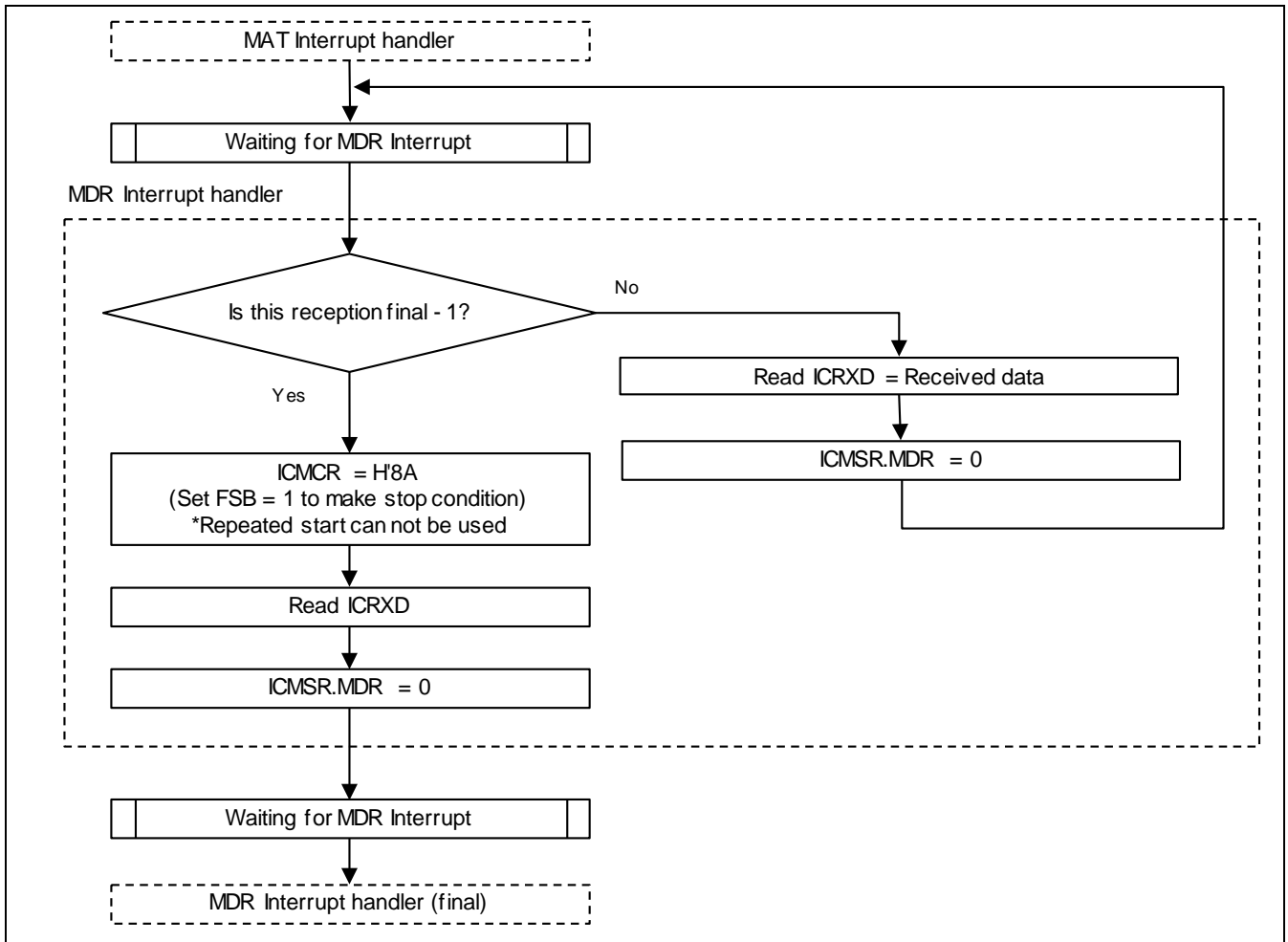


Figure 52.17 MDR Interrupt Handler for Master Receiver Operation Setting

52.4.3 Master Transmitter—Repeated START—Master Receiver

In order to set up the master interface to transmit a data packet on the I2C bus, issue a repeated START condition, then read byte data back from the slave, follow the procedure below:

(1) Set value for the clock control register

The same procedure as the master transmitter is applied. See the program example for the master transmitter.

(2) Set value for the master control register and address

1. Clear the Master Status register (ICMSR) = H'00 (all 0 clear).
2. Set the Master Interrupt Enable register (ICMIER) = H'0B (MDEE = 1, MDRE = 1, MATE = 1).
3. Set the Master Address register (ICMAR) = Slave device address + H'00 (Write mode).
4. Set the Transmit Data register (ICTXD) = 1st transmission data.
5. Set the Master Control register (ICMCR) = H'89 (MDBS = 1, MIE = 1, ESG = 1).

(3) Wait for outputting slave-address for master transmission

1. Wait for master event (an interrupt of the ICMSR.MAT and ICMSR.MDE bits).
2. Clear ICMCR.ESG bit.
3. Clear the ICMSR.MAT and ICMSR.MDE bits.

(4) Monitor transmission of data

1. Wait for master event (an interrupt of the ICMSR.MDE bit).
2. Set ICMAR = Slave device address + H'01 (Read mode).
3. Set ICMCR = H'89 (MDBS = 1, MIE = 1, ESG = 1).
4. Clear the ICMSR.MDE bit.

(5) Wait for outputting slave-address for master reception

1. Wait for master event (an interrupt of ICMSR.MAT and ICMSR.MDR bits).
2. Clear ICMCR ESG bit. (To suspend stop the data transmission, the master device will hold the SCL low until the ICMSR.MDR bit is cleared.)
3. Clear the ICMSR.MAT and ICMSR.MDR bits.

(6) Monitor reception of data

1. Wait for master event (the ICMSR.MDR bit).
2. Clear the ICMSR.MDR bit after reading data from the receive data register.
3. Set ICMCR.FSB to 1 (set the force stop control bit) before the last byte data transfer is started.

(7) Wait for end of reception

1. Handle the receive interrupt (ICMSR.MDR) in the last byte: that is, read the data and clear the ICMSR.MDR.
2. Wait for master event, ICMSR.MST.
3. Clear the ICMSR.MST bit.

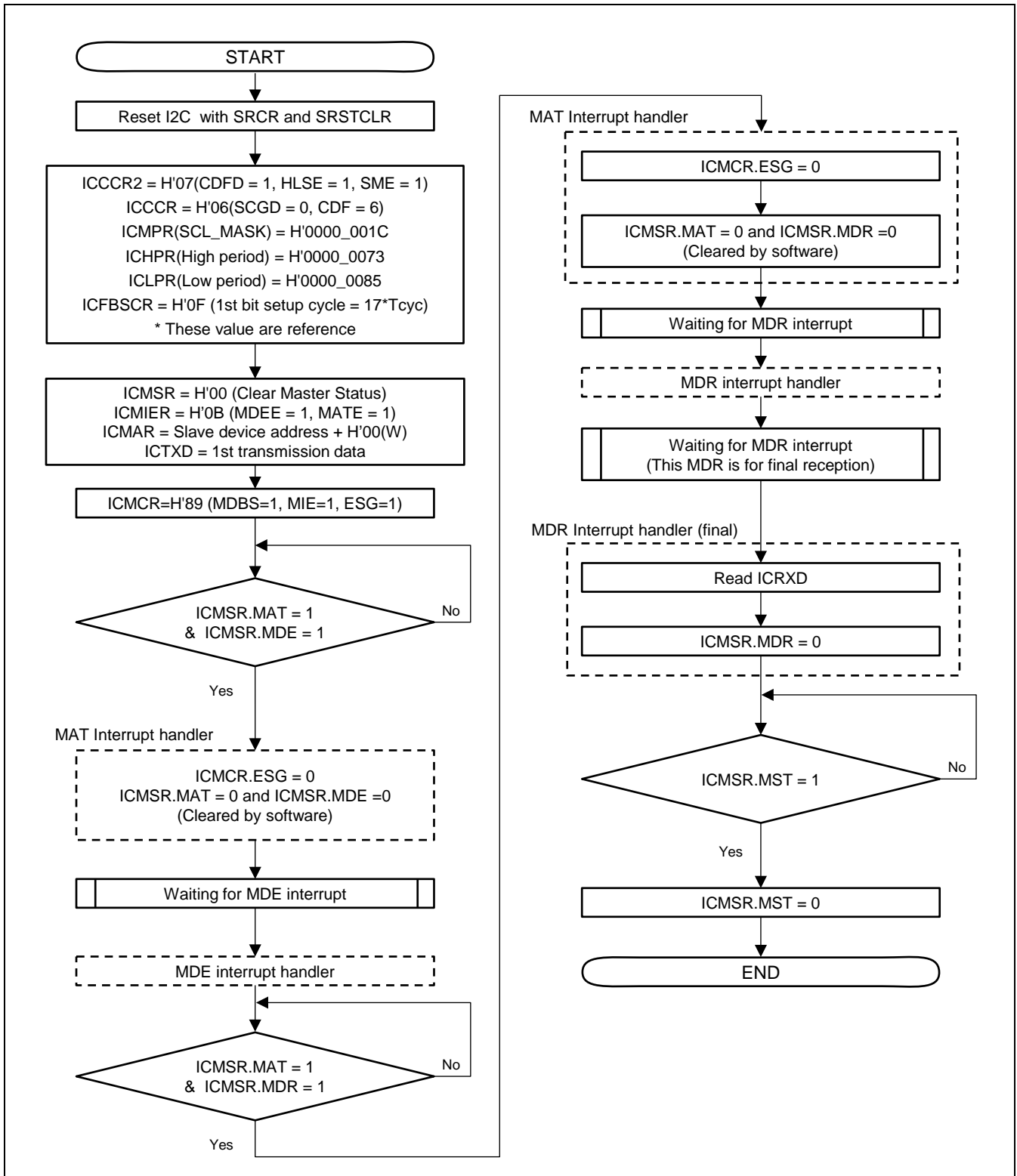


Figure 52.18 Master Transmitter – Repeated START – Receiver Operation Setting

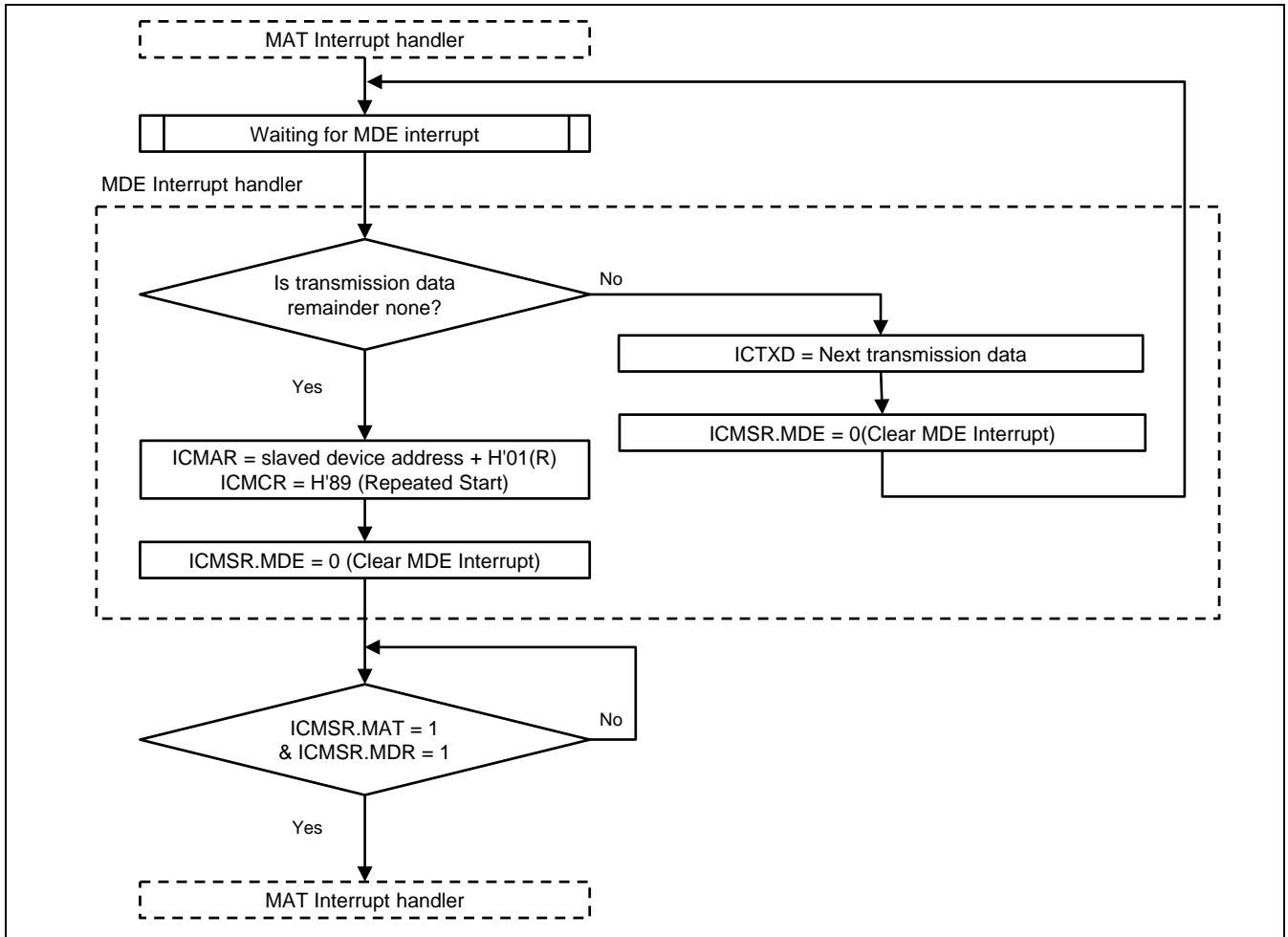


Figure 52.19 MDE Interrupt Handler for Master Transmitter – Repeated START - Receiver Operation Setting

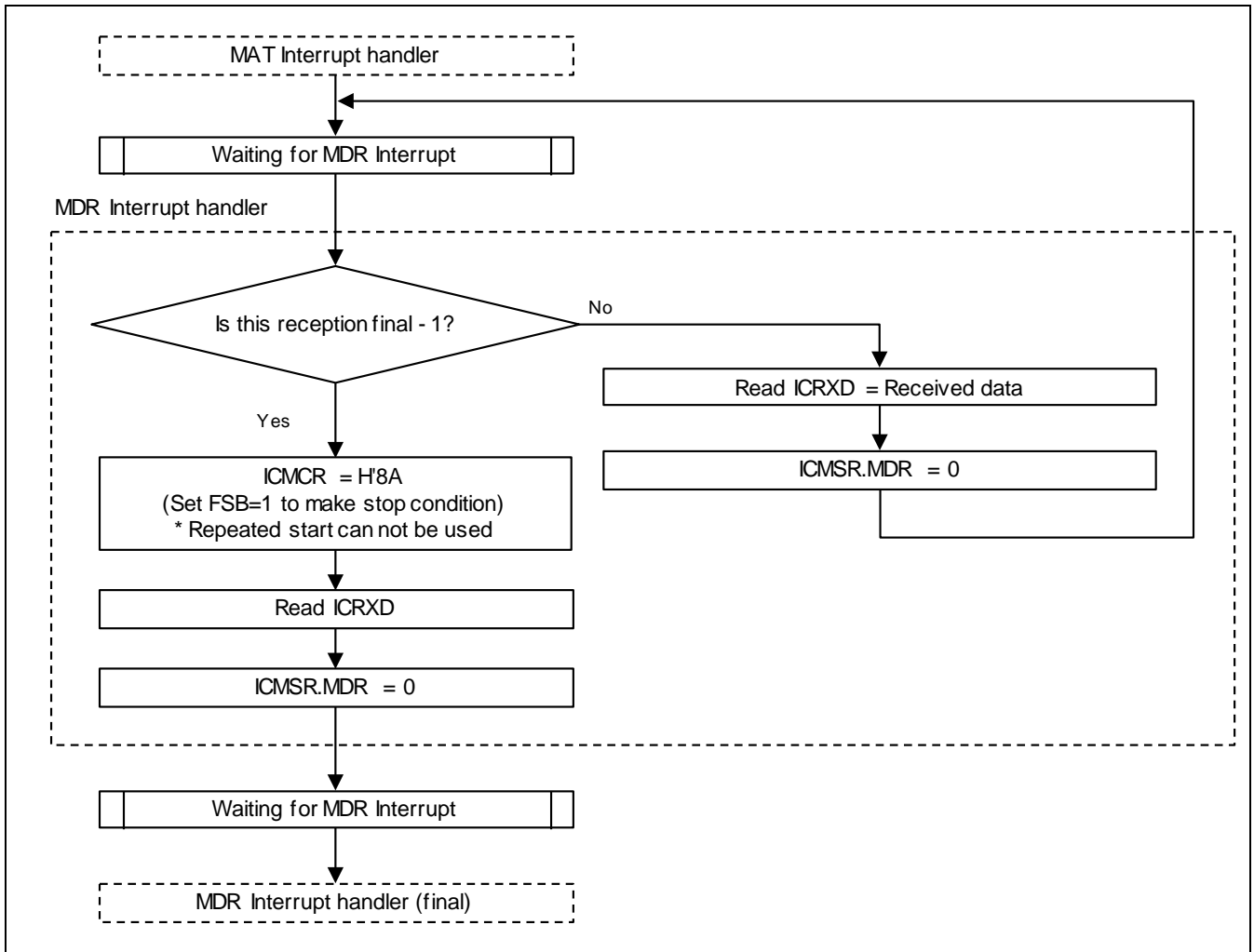


Figure 52.20 MDR Interrupt Handler for Master Transmitter – Repeated START – Receiver Operation Setting

52.4.4 Slave Transmitter

In order to set up the master interface to transmit a data packet on the I2C bus, follow the procedure below (an example for I2C0):

(1) Set value for the slave control register, first data byte and address

1. Reset I2C with SRCR and SRSTCLR.
2. Clear the Slave Status register (ICSSR) = H'00 (all 0 clear).
3. Set the slave Interrupt register (ICSIER) = H'09 (SDEE = 1, SARE = 1).
4. Set Slave Address register (ICSAR) = Slave device address (unique value from other slave devices).
5. Set the Transmit Data register (ICTXD) = 1st transmission data.
6. Set the Slave control register (ICSCR) = H'0E (SDBS = 1, SIE = 1, GCAE = 1).

(2) Wait for receiving address

1. Wait for slave event (an interrupt of the ICSSR.SAR).
2. Clear the ICSSR.SAR bit.

(3) Monitor transmission of data

1. Wait for master event, ICSSR.SDE.
2. ICTXD = subsequent data.
3. Clear the ICSSR.SDE bit.

Clear ICSSR.SDE after setting the last byte to be transmitted. After the last byte data is loaded into the shift register, ICSSR.SDE is generated. So, skip writing to ICTXD at ICSSR.SDE after last byte data is loaded into the shift register.

(4) Wait for end of transmission

1. Wait for slave event, ICSSR.SSR.
2. Clear the ICSSR.SSR bit.

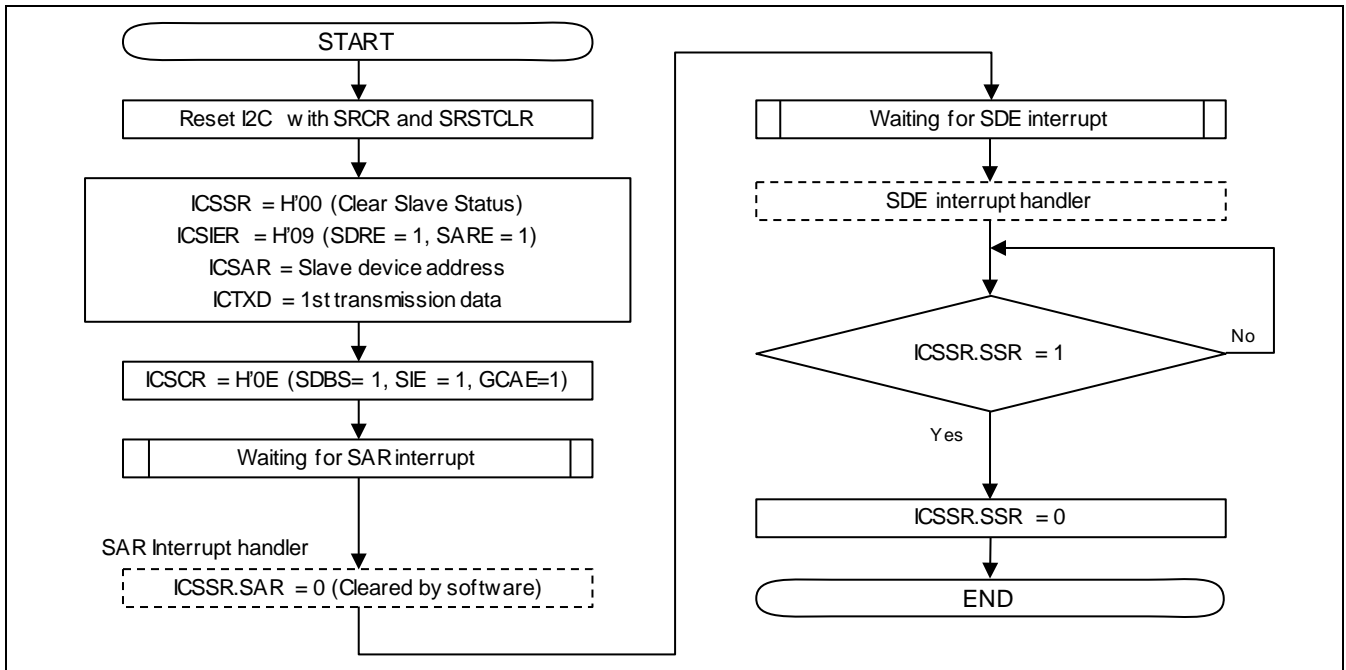


Figure 52.21 Slave Transmitter Operation Setting

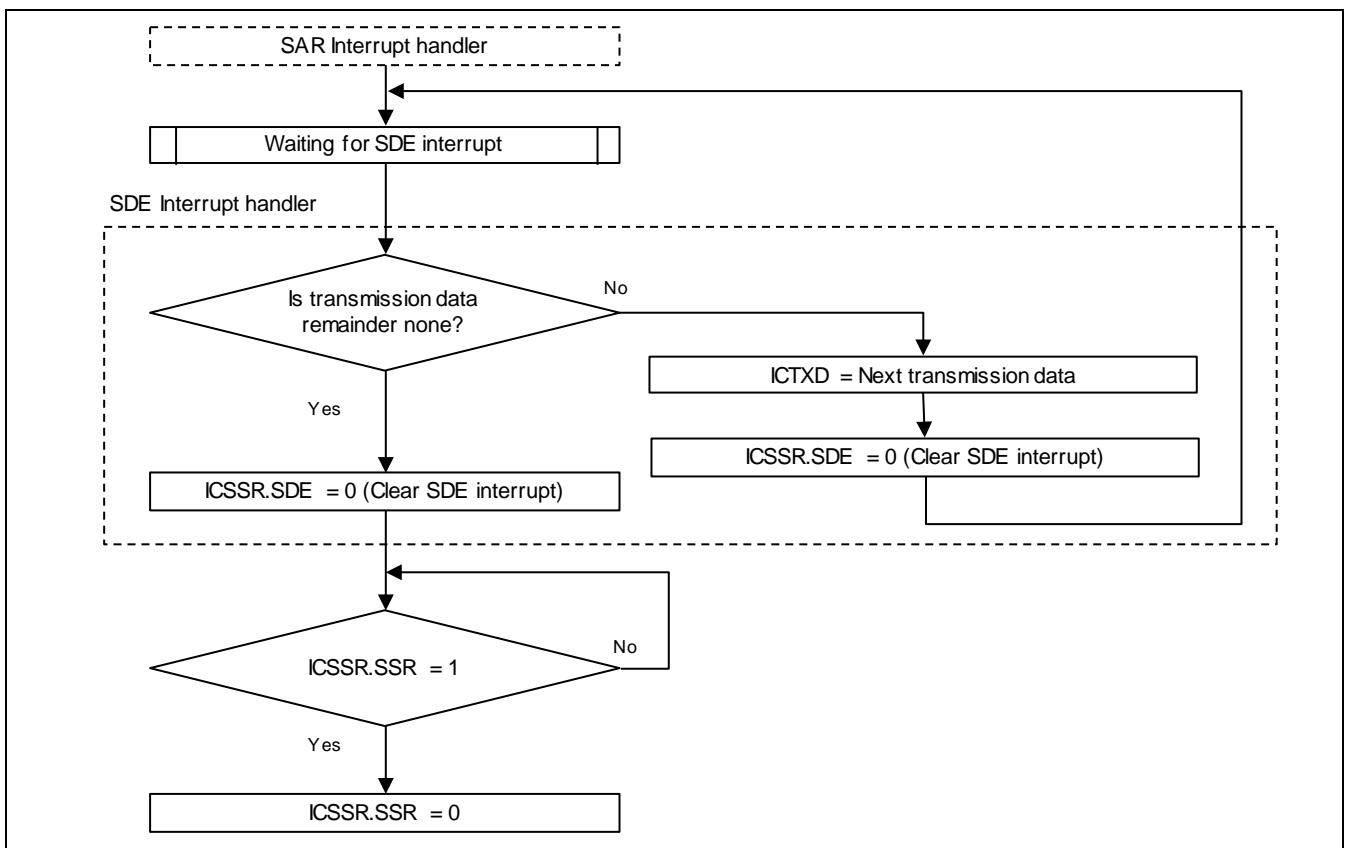


Figure 52.22 SDE Interrupt Handler for Slave transmitter Operation Setting

52.4.5 Slave Receiver

In order to set up the master interface to receive a data packet on the I2C bus, follow the procedure below:

(1) Set value for the master control register and address

1. Reset I2C with SRCR and SRSTCLR.
2. Clear the Slave Status register (ICSSR) = H'00 (all 0 clear).
3. Set the slave Interrupt register (ICSIER) = H'03 (SDRE = 1, SARE = 1).
4. Set Slave Address register (ICSAR) = Slave device address (unique value from other slave devices).
5. Set the Slave control register (ICSCR) = H'0E (SDBS = 1, SIE = 1, GCAE = 1).

(2) Wait for receiving address

1. Wait for slave event (an interrupt of the ICSSR.SAR).
2. Clear the ICSSR.SAR bit.

(3) Monitor reception of data

1. Wait for slave event, ICSSR.SDR bit.
2. Clear the ICSSR.SDR bit after reading data from ICRXD register.

(4) Wait for end of reception

1. Handle the ICSSR.SDR in the last byte: that is, read the data and clear the ICSSR.SDR.
2. Wait for master event, ICSSR.SSR.
3. Clear the ICSSR.SSR bit.

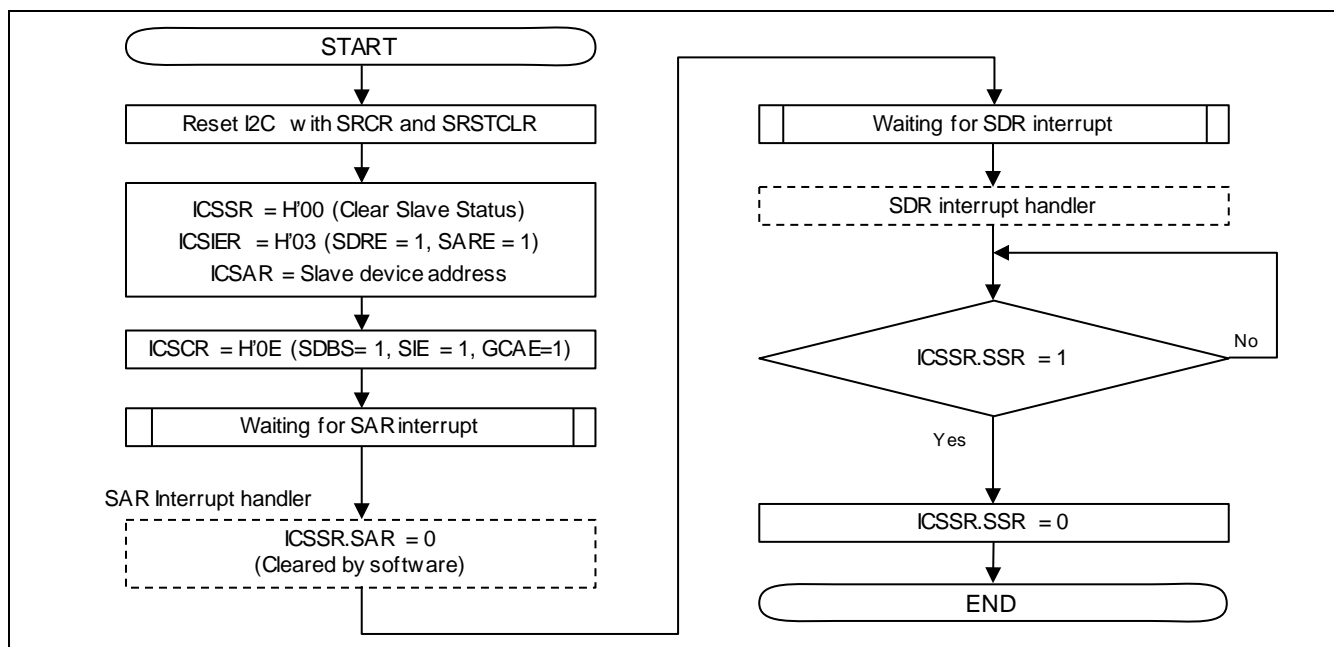


Figure 52.23 Slave Receiver Operation Setting

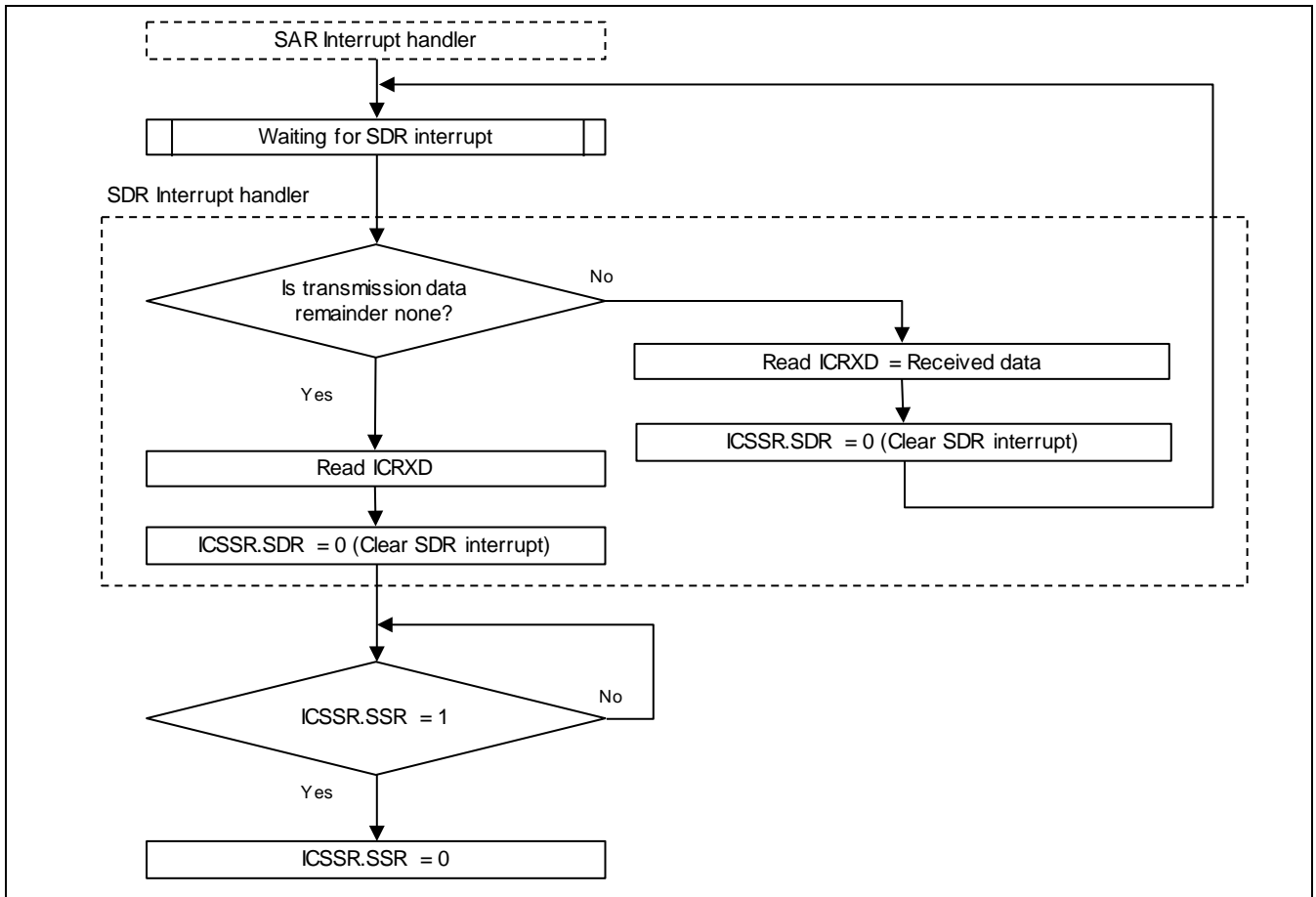


Figure 52.24 SDR Interrupt Handler for Slave Receiver Operation Setting

52.4.6 DMA Mode - Master Transmit Operation

The 1st byte is transferred by PIO, after that enter to DMA mode to start transfer the following bytes.

In DMA transmit mode, I2C module asserts a DMA request to the DMAC. After receiving the request, DMAC write data into I2C data register, and return DMARACK to I2C module. For instance, in order to transfer N0, N1, ... , N10 byte. Byte N0 will be transferred by PIO, bytes N1 to N10 will be transferred by using DMAC.

The transmission procedure and operations are described as below:

(1) Set value for the clock control register

1. Reset I2C with SRCR and SRSTCLR.
2. Set the Clock Control register 2 (ICCCR2) = H'07 (CDFD = 1, HLSE = 1, SME = 1).
3. Set the Clock Control register (ICCCR) = H'06 (SCGD = 0, CDF = 6, I2C internal clock I2Cck: 19 MHz).
— Setting ICCCR, ICCCR2 for using Duty ratio fixed or Duty ratio variable. The case for using Variable Duty ratio, have to setting ICCCR2 too. The case for using Fixed Duty ratio, have to setting ICCCR only.
4. Set the SCL Mask Control register (ICMPR) = H'1C (Variable Duty ratio only).
5. Set the SCL High Control register (ICHPR) = H'73 (Variable Duty ratio only).
6. Set the SCL Low Control register (ICLPR) = H'85 (Variable Duty ratio only).
7. Set the First Bit Setup Cycle register (ICFBSCR) = H'0F (1st bit setup cycle = 17 × Tcyc).

(2) Setting of DMAC

1. Set the number of DMA times (“Total transmit times – 1”).
- In this phase, DMACHCR.DE = 0.

(3) Set value for the master control register, first data byte, and address

1. Clear the Master Status register (ICMSR) = H'00 (all 0 clear).
2. Set the Master Interrupt Enable register (ICMIER) = H'09 (MDEE = 1, MATE = 1).
3. Set the Master Address register (ICMAR) = Slave device address + H'00 (Write mode).
4. Set the Transmit Data register (ICTXD) = 1st transmission data.
5. Set the Master Control register (ICMCR) = H'89 (MDBS = 1, MIE = 1, ESG = 1).

(4) Wait for outputting address

1. Wait for master event, ICMSR.MAT and ICMSR.MDE.
2. Set ICMCR to H'88 (to suspend the data transmission, the master device will hold the SCL low until the ICMSR.MDE bit is cleared).
- If only one byte of data is transmitted, set the ICMCR to H'8A, meaning that the stop generation is enabled.
This generates a stop on the bus as soon as one byte has been transmitted.
3. Clear the ICMSR.MAT & ICMSR.MDE bits.

(5) Enable DMA

1. Wait for master event, ICMSR.MDE.
2. Set the DMA Enable register (ICDMAER) = H'01 (TMDMAE = 1).
3. Set the DMACHCR.DE = 1 to enable DMA.

(6) DMA interrupt handler

1. Wait for DEI interrupt.
2. Set ICDMAER = H'00 (TMDMAE = 0) to disable DMA transmission.

(7) MDE interrupt handler

1. Wait for MDE interrupt.
2. Set ICMCR = H'8A (FSB = 1) to make stop condition.
3. Clear MDE interrupt (ICMSR.MDE = 0).

(8) Wait for end of transmission

1. Wait for ICMSR.MST = 1.
2. Clear ICMSR.MST = 0.

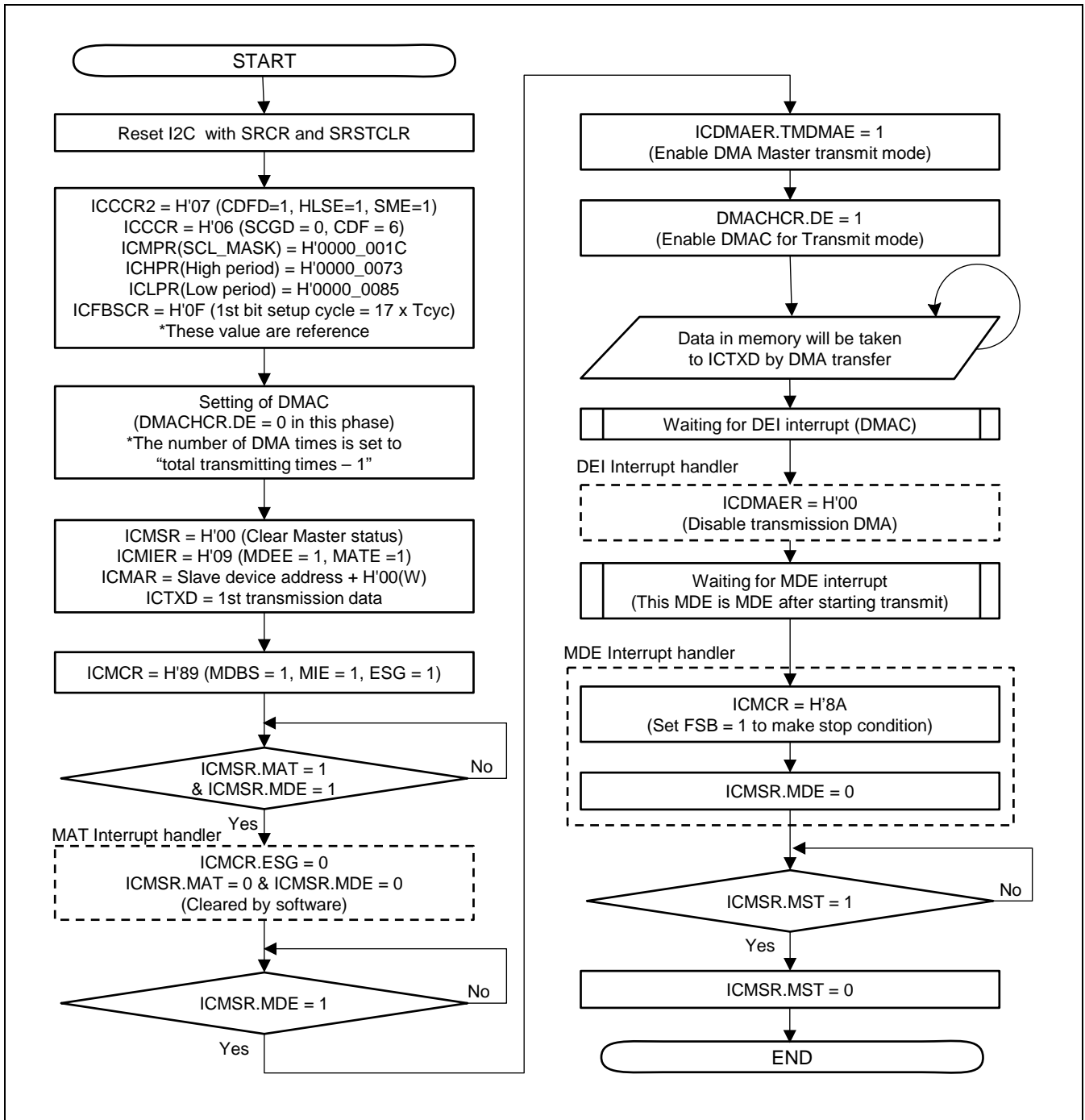


Figure 52.25 Example DMA mode – Master Transmitted Operation Setting

52.4.7 DMA Mode – Master Receive Operation

In DMA receive mode, I2C module asserts a DMA request to the DMAC. After receiving the request, DMAC reads the received data from I2C data register and returns DMARACK to I2C module. All received data bytes will be transferred by using DMAC.

The reception procedure and operations are described as below:

(1) Set value for the clock control register

The same procedure as the master transmitter is applied. See the program example for the master transmitter.

(2) Set value for the master control register, first data byte, and address

1. Clear the Master Status register (ICMSR) = H'00 (all 0 clear).
2. Set the Master Interrupt Enable register (ICMIER) = H'0B (MDEE = 1, MDRE = 1, MATE = 1).
3. Set the Master Address register (ICMAR) = Slave device address + H'00 (Write mode).
4. Set the Transmit Data register (ICTXD) = 1st transmission data.
5. Set the Master Control register (ICMCR) = H'89 (MDBS = 1, MIE = 1, ESG = 1).

(3) Wait for outputting address

1. Wait for master event (an interrupt of the ICMSR.MAT and ICMSR.MDE).
2. Set the ICMCR to H'88 (to suspend the data transmission, the master device will hold the SCL low until the MDE bit is cleared).
— This generates a stop on the bus as soon as one byte has been transmitted.
3. Clear the ICMSR.MAT & ICMSR.MDE bits.

(4) MDE interrupt handler

1. Wait for master event, ICMSR.MDE.
2. Set the ICMAR = slave device address + H'01(Read mode).
3. Set the ICMCR = H'89 (MDBS = 1, MIE = 1, ESG = 1).
4. Clear the ICMSR.MDE = 0.

(5) Enable DMA of I2C

1. Wait for master event, ICMSR.MAT and ICMSR.MDR.
2. Clear the ICMCR.ESG = 0.
3. Set the DMA Enable register (ICDMAER) = H'02 (RMDMAE = 1).
4. Clear the ICMSR.MAT and ICMSR.MDR bits.

(6) Setting of DMAC

1. Set the number of DMA times ("Total receiving times - 2").
– In this phase, DMACHCR.DE = 0.

(7) Enable DMA of DMAC

1. Set the DMACHCR.DE = 1 to enable DMA.

(8) DMA interrupt handler

1. Wait for DEI interrupt.
2. Set ICDMAER = H'00 (RMDMAE = 0) to disable DMA transmission.

(9) MDR interrupt handler

1. Wait for master event, ICMSR.MDR.
2. Set the ICMCR = H'8A (MDBS = 1, MIE = 1, FSB = 1).
3. Read ICRXD.
4. Clear the ICMSR.MDR = 0.

(10) Final MDR interrupt handler

1. Wait for master event, ICMSR.MDR.
2. Read ICRXD.
3. Clear the ICMSR.MDR = 0.

(11) Wait for end of transmission

1. Wait for the master event, ICMSR.MST.
2. Clear the ICMSR.MST = 0.

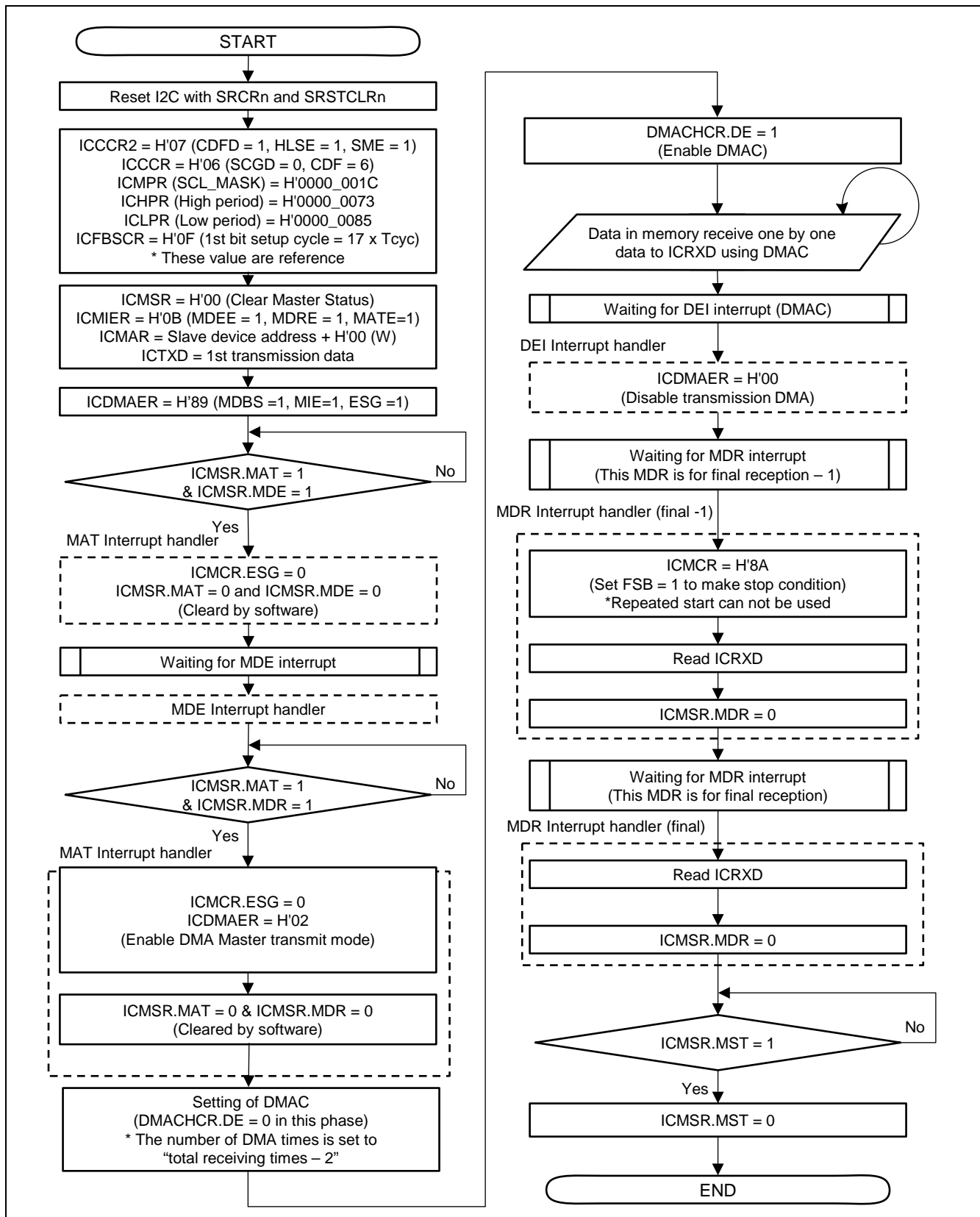


Figure 52.26 Example DMA mode – Master Received Operation Setting

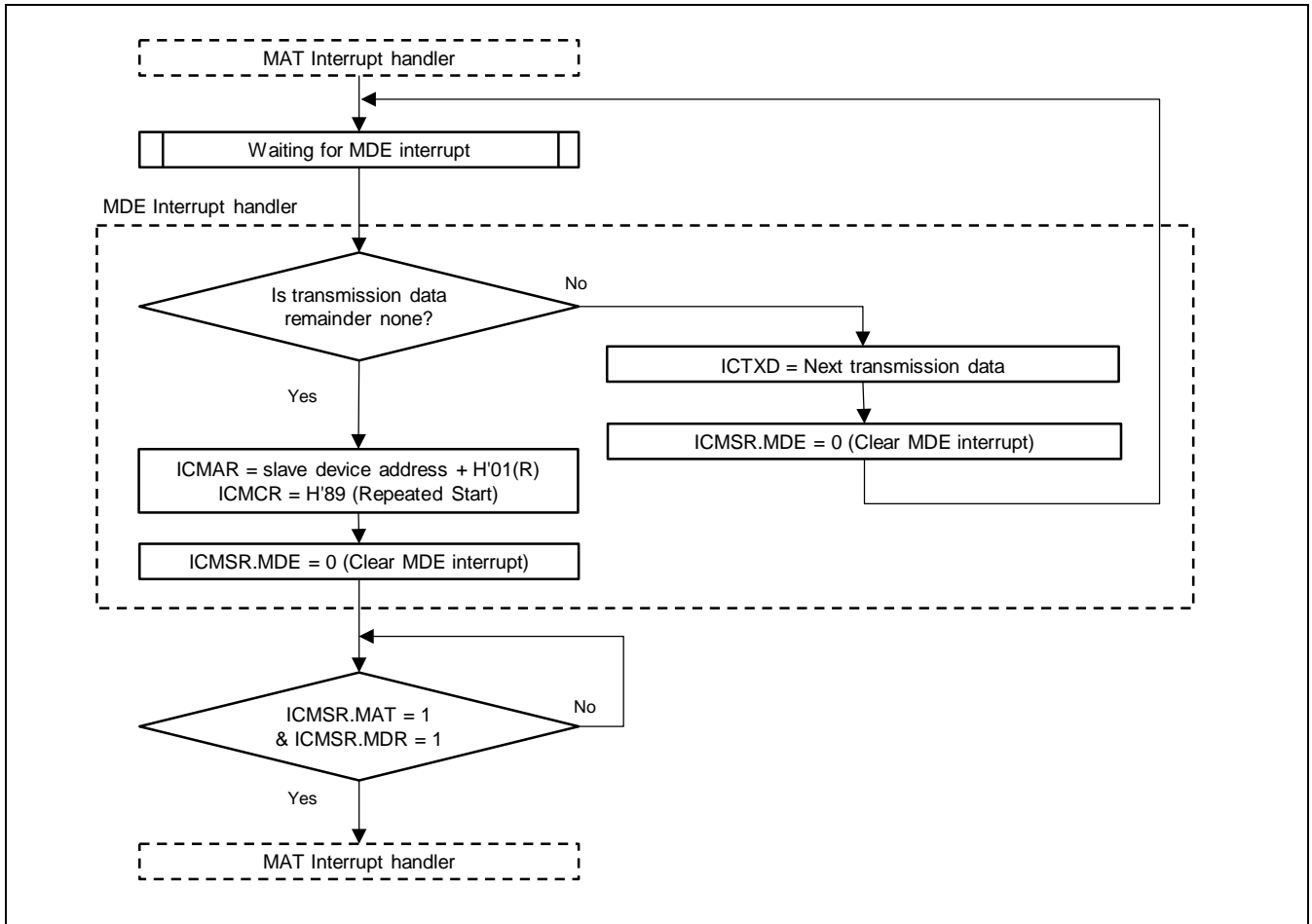


Figure 52.27 MDE Interrupt Handler for Example DMA mode – Master Received Operation Setting

52.4.8 DMA Mode – Slave Transmit Operation

The 1st byte is transferred by PIO, after that enter to DMA mode to start transfer the following bytes.

In DMA transmit mode, I2C module asserts a DMA request to the DMAC. After receiving the request, DMAC write data into I2C data register, and return DMARACK to I2C module. For instance, in order to transfer N0, N1, ... , N10 byte. Byte N0 will be transferred by PIO, bytes N1 to N10 will be transferred by using DMAC.

The transmission procedure and operations are described as below:

(1) Setting of DMAC

1. Set the number of DMA times (“Total transmit times – 1”).
— In this phase, DMACHCR.DE = 0.

(2) Set value for the slave control register, first data byte, and address

1. Reset I2C with SRCR and SRSTCLR.
2. Clear the Slave Status register (ICSSR) = H'00 (all 0 clear).
3. Set the Slave Interrupt Enable register (ICSIER) = H'09 (SDEE = 1, SARE = 1).
4. Set the Slave Address register (ICSAR) = Slave device address.
5. Set the Transmit Data register (ICTXD) = 1st transmission data.
6. Set the DMA Enable register (ICDMAER) = H'04 (TSDMAE = 1).
7. Set the Slave Control register (ICSCR) = H'0E (SDBS = 1, SIE = 1, GCAE = 1).

(3) Wait for receiving address

1. Wait for slave event, ICSSR.SAR.
2. Clear the ICSSR.SAR = 0

(4) Enable DMA

1. Wait for slave event, ICSSR.SDE.
2. Set the DMACHCR.DE = 1 to enable DMA.

(5) DMA interrupt handler

1. Wait for DEI interrupt.
2. Set ICDMAER = H'00 (TSDMAE = 0) to disable DMA transmission.

(6) SDE interrupt handler

1. Wait for slave event, ICSSR.SDE.
2. Clear SDE interrupt (ICSSR.SDE = 0).

(7) Wait for end of transmission

1. Wait for ICSSR.SSR = 1.
2. Set ICSSR.SSR = 0.

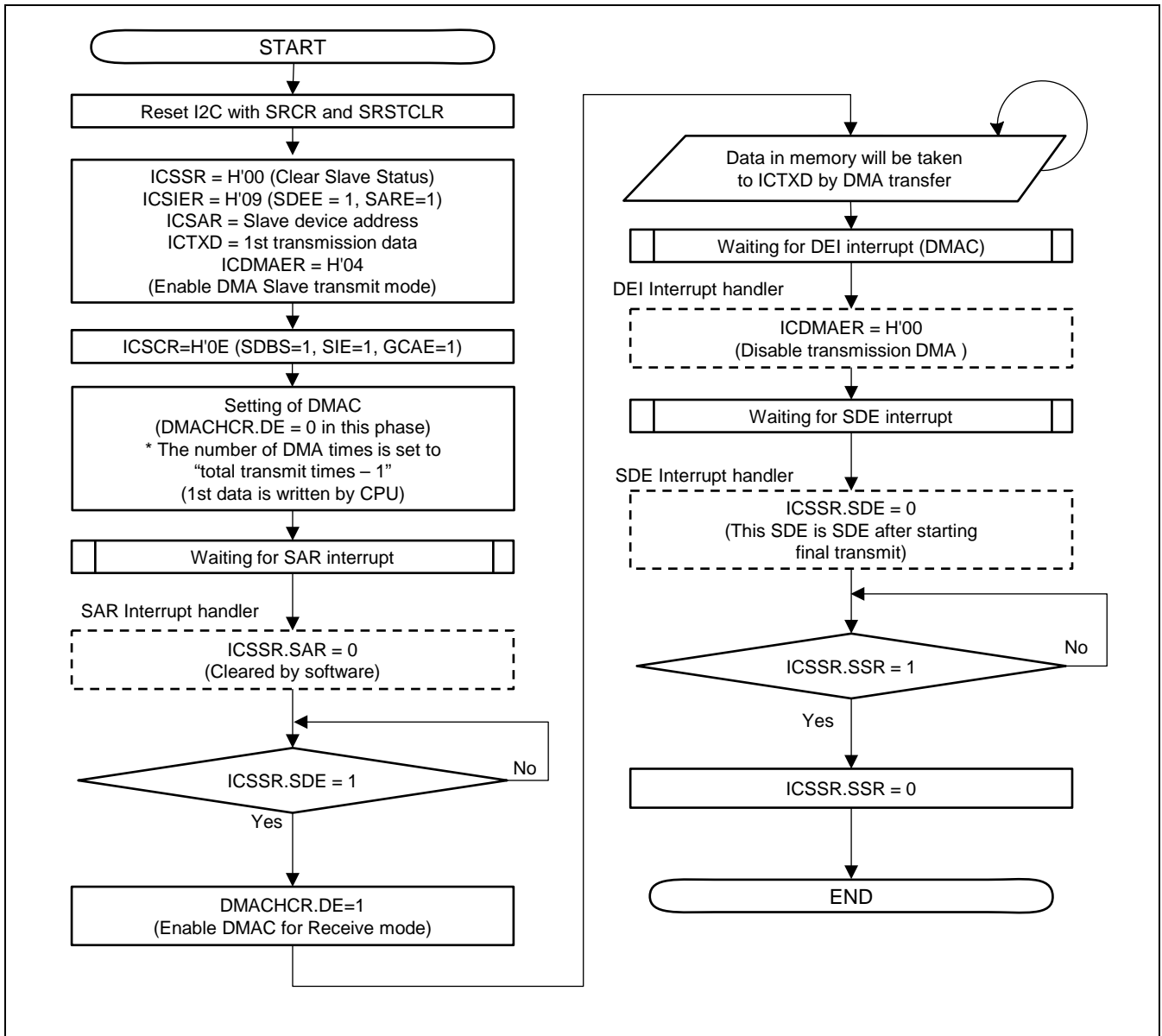


Figure 52.28 Example DMA mode – Slave Transmitted Operation Setting

52.4.9 DMA Mode – Slave Receive Operation

In DMA receive mode, I2C module asserts a DMA request to the DMAC. After receiving the request, DMAC reads the received data from I2C data register and returns DMARACK to I2C module. All received data bytes will be transferred by using DMAC.

The reception procedure and operations are described as below:

(1) Setting of DMAC

1. Set the number of DMA times (“Total receiving times”).
— In this phase, DMACHCR.DE = 0.

(2) Set value for the slave control register, first data byte, and address

1. Reset I2C with SRCR and SRSTCLR.
2. Clear the Slave Status register (ICSSR) = H'00 (all 0 clear).
3. Set the Slave Interrupt Enable register (ICSIER) = H'03 (SDRE = 1, SARE = 1).
4. Set the Slave Address register (ICSAR) = Slave device address.
5. Set the DMA Enable register (ICDMAER) = H'08 (RSDMAE = 1).
6. Set the Slave Control register (ICSCR) = H'0E (SDBS = 1, SIE = 1, GCAE = 1).

(3) Wait for receiving address

1. Wait for slave event, ICSSR.SAR.
2. Clear the ICSSR.SAR = 0.

(4) Enable DMA

1. Wait for slave event, ICSSR.SDR.
2. Set the DMACHCR.DE = 1 to enable DMA.

(5) DMA interrupt handler

1. Wait for DEI interrupt.
2. Set ICDMAER = H'00 (RSDMAE = 0) to disable DMA transmission.

(6) Wait for end of transmission

1. Wait for ICSSR.SSR = 1.
2. Set ICSSR.SSR = 0.

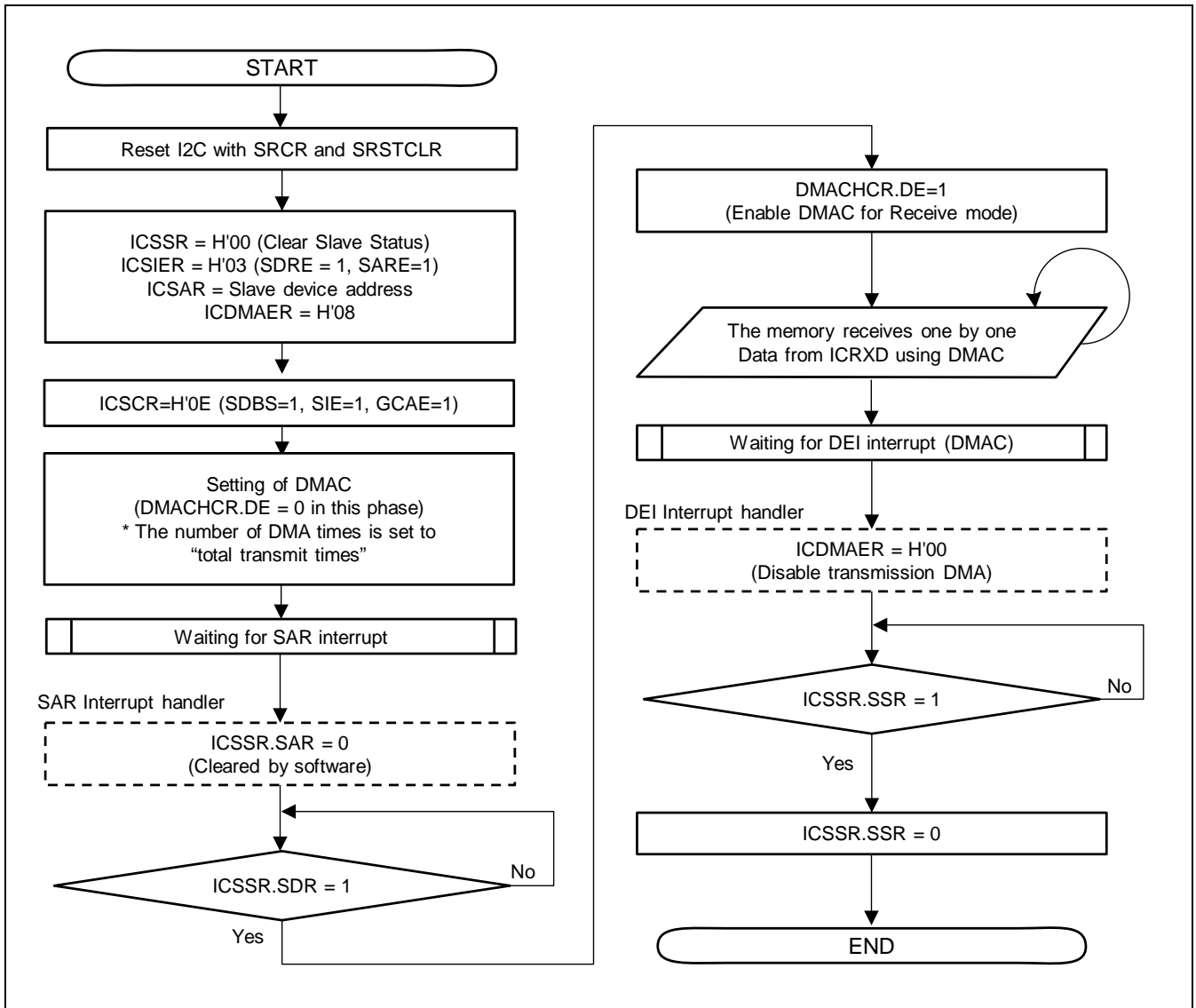


Figure 52.29 Example DMA mode – Slave Received Operation Setting

52.5 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

52.5.1 Usage note for Multi-Master

Do not stop clock distribution to this module, even if this module is not directly used for data transfer, while this module is used in multi-master environment. This module is monitoring SCL and SDA bus line for detecting START and STOP condition for arbitration. If clock is stopped, arbitration problem may occur.

52.5.2 Usage note for SCL frequency calculation equation

In section 52.2.9, Clock Control Register (ICCCR), in the description of SCGD and CDF, following equation is described.

$$\text{SCLfreq} = \text{I2Cck} / (20 + \text{SCGD} \times 8 + \text{F}[(\text{tICF} + \text{tr} + \text{IntDelay}) \times \text{I2Cck}])$$

I2Cck: I2C internal clock frequency

tICF: I2C SCL falling time (depending on external load)

tr: I2C SCL rising time (depending on external load)

IntDelay: LSI internal delay corresponds to output buffer type.

Open drain buffer: 50 ns (typ.), 110 ns (max.)

LVTTL (low drive only) buffer: 5 ns (typ.), 6 ns (max.).

F[n]: n rounded down to an integer

$$\text{I2Cck} = \text{MOD_CLK} / (1 + \text{CDF})$$

MOD_CLK: module clock frequency

Based on parameter described above, this equation approximately estimates the SCL frequency, but includes some error in the resulting frequency. Before finalizing register setting to be used, measure actual SCL frequency on the board, and confirm that observed frequency satisfy the constraint. The reason why calculated result includes some error is parameters used in this equation varies depending on the characteristics of each sample. For example, tICF and tr depend on logical threshold of IO-cell, and intDelay depends on variation of fabrication process.

For adjustment of SCL frequency, modifying the setting of ICCCR [2:0]. CDF and ICCCR [8:3]. SCGD will be effective. Increasing the value of CDF or SCGD value will make SCL frequency lower. However, how they decrease SCL frequency is different. So, by increasing one register value and decreasing the other, SCL frequency can be adjusted.

52.5.3 Usage note for Pull-up resistance

Do not use internal pull up of PUEN and PUD registers in Pin Function Controller (PFC) section for I2C function. External pull up resistance on board is mandatory for SCL and SDA pins.

52.5.4 Usage note for the transmission and reception procedure

Be sure to reset I2C with Software Reset Register SRCRn (n = 9 or 10*) at the beginning of transmission and reception procedure.

A software reset can be set by controlling the Software Reset Register SRCRn (n = 9 or 10*) and Software Reset Clearing Register SRSTCLRn (n = 9 or 10*) of the “8A. Module Standby, Software Reset”

*: SRCR10 and SRSTCLR10 are only for RZ/G2E

Reset Sequence:

Set the target I2C bit of SRCRn to 1. (**)
 → Wait 1μsec → Set the target I2C bit of SRSTCLRn to 1. (**)
 → Wait for the target bit of SRCRn to 0. → End sequence

** : Other bit should be written 0.

52.5.5 Usage note for DMA mode of Receive Operation

If use DMA mode of Receive Operation more than once with repeated START, issue "STOP condition" and start from beginning of transmission and reception procedure instead of repeated START.

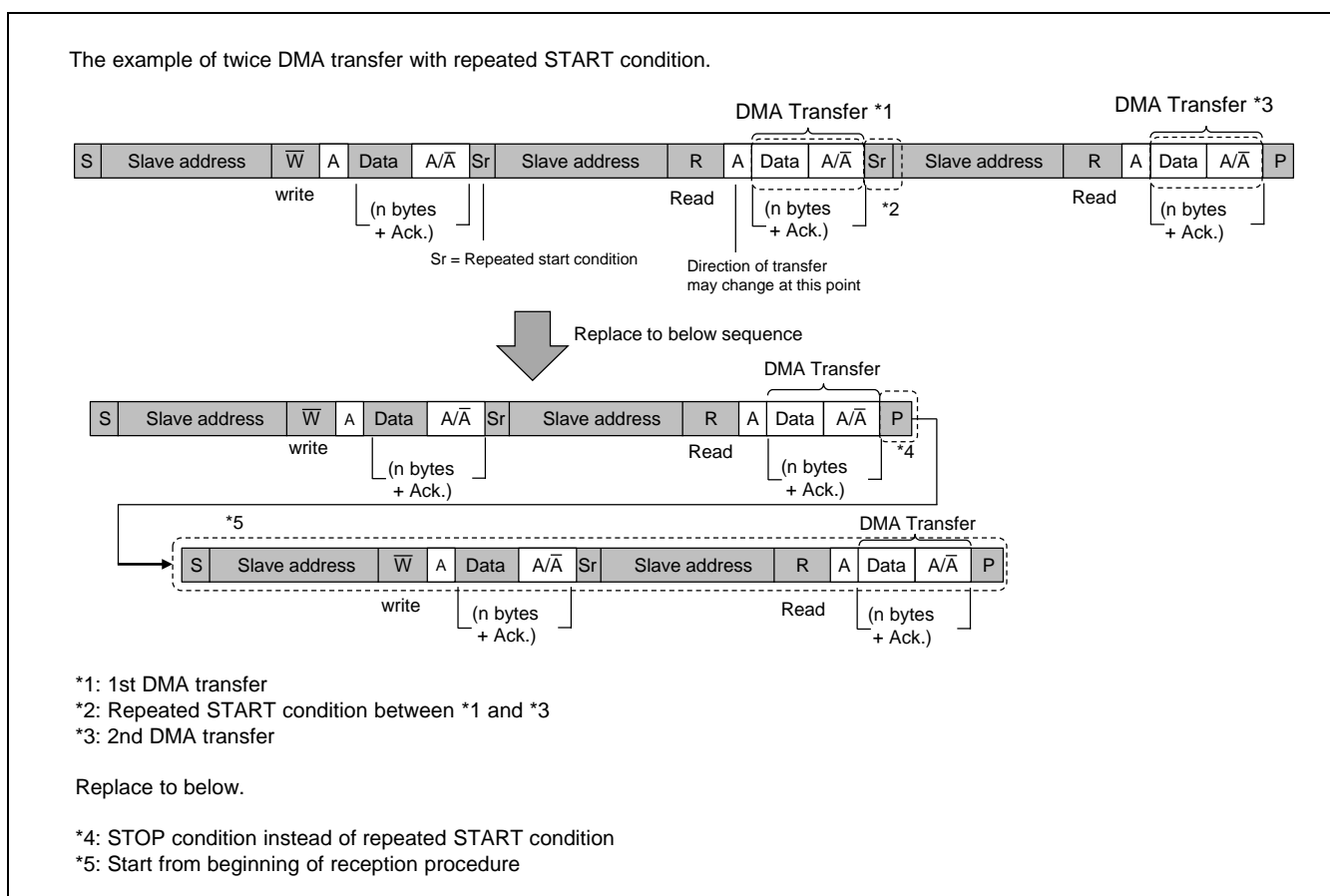


Figure 52.30 Example of Usage Note for DMA mode of Receive Operation

53. IIC Bus Interface for PMIC (IIC for PMIC)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

53.1 Overview

The IIC bus interface uses only one data line (SDA) and one clock line (SCL) to transfer data, saving board and connector space.

Note: "IIC" mentioned in this chapter means the same as "I2C".

53.1.1 Features

The second generation RZ/G series products have one IIC bus interface module (IIC-PMIC).

The module clock for the IIC-PMIC is $CP\phi$.

- Data transfer based on I2C bus format from Philips Semiconductor (now NXP Semiconductors)
- Automatic generation of START and STOP conditions
- Control of acknowledge level after receiving data
- Automatic checking of acknowledge bit after transmitting data
- Wait function
 - A wait can be inserted by driving the SCL pin low after data transfer, excluding acknowledgement.
 - The wait can be canceled by clearing the interrupt flag.
- This module supports only master mode.
 - If arbitration is lost, data transfer is stopped and I2C bus is released.
- Four interrupt sources
 - Data transfer enable
 - Wait state
 - Non-acknowledge detection
 - Arbitration lost
- Data transfer speed
 - Supports Standard-mode (100 kHz) and Fast-mode (400 kHz)
 - SCL clock frequency can be controlled by register setting.
- Support clock synchronization of SCL line
 - A hazard (transient spike) in the SCL high-count period is detected as a loss of arbitration.
- DMA transfer
 - Support DMA transfer (transmission and reception).

Figure 53.1 shows a block diagram of the IIC bus interface module.

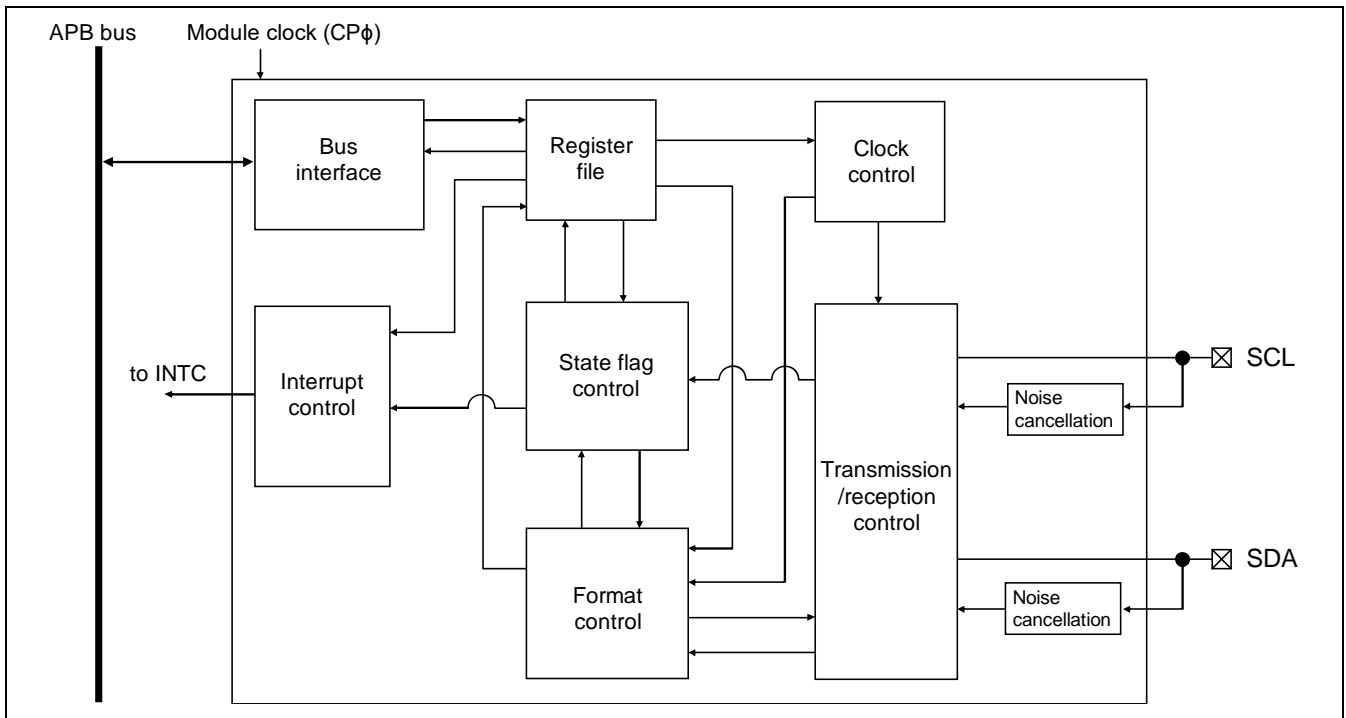


Figure 53.1 Block Diagram of IIC Bus Interface Module

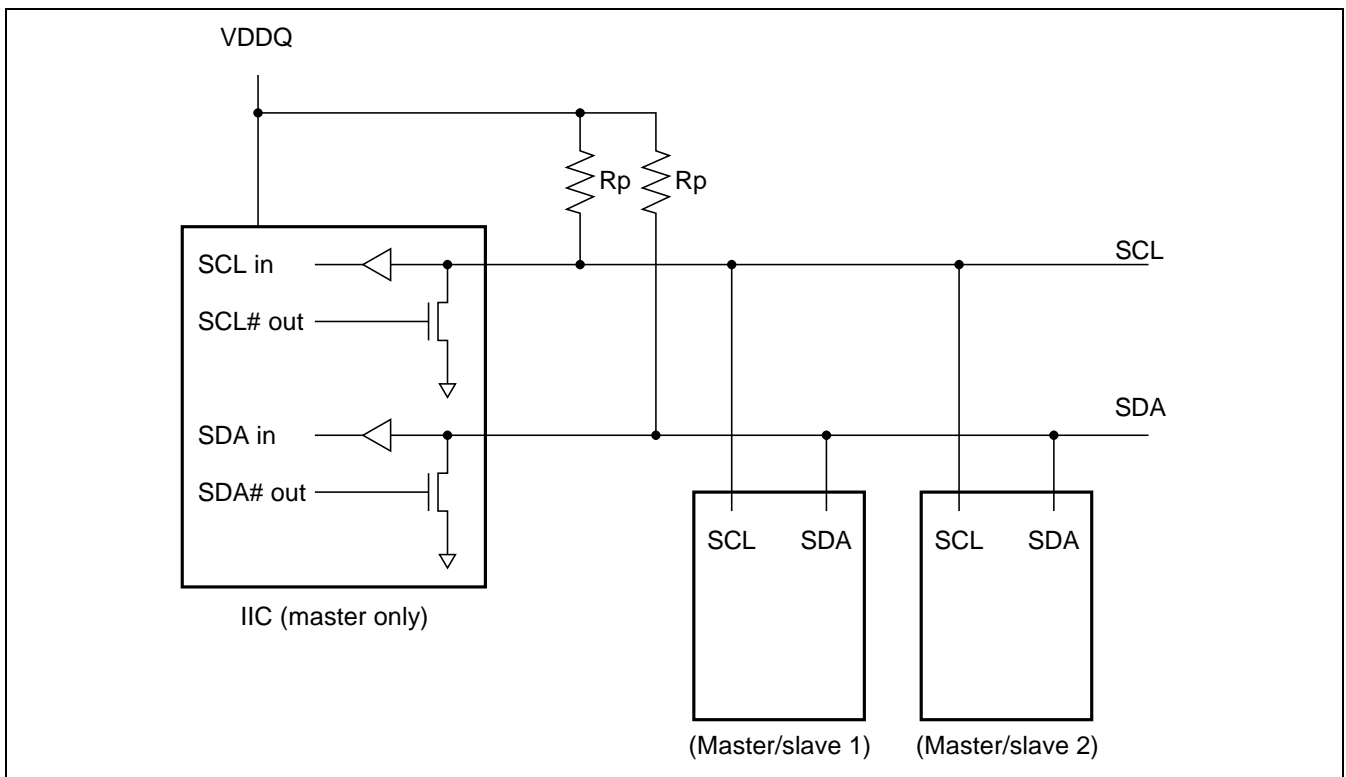


Figure 53.2 Connection Example of IIC Bus Interface Module

53.1.2 External Pins

Table 53.1 summarizes the input/output pins used by the IIC bus interface.

Table 53.1 IIC Bus Interface Pins

| Channel Number | Pin Name*1 | I/O | Function | Second Generation RZ/G Series Products | | | |
|----------------|------------|-----|---|--|----------------------------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| | | | | Output Buffer Type*2 | | | |
| 0 | SCL | I/O | SCL: IIC serial clock input/output pin | OD | OD | OD | OD |
| | SDA | | SDA: IIC serial data input/output pin | | | | |

- Notes: 1. The actual pin names are SCL_PMIC, SDA_PMIC.
 2. Output buffer type: "OD" is open drain buffer. OD assigned channel pins are 1.8 V I/O and input/output 3.3 V tolerant.

The IIC buffer in this LSI is not compliant with the 5V-input. When turning off the I/O power source (3.3 V) of this LSI, turn off the power source of the pull-up resistors connected to the IIC pins.

Note: When using 1.8V IIC pins (OD assigned channel) as 3.3 V tolerant, all the external pull-up voltage power supply to the IIC of this LSI must keep the same power on/off sequence as the VDDQ (3.3 V) of this LSI.

53.1.3 Register Configuration

Table 53.2 shows the register configuration of the IIC bus interface. Table 53.3 shows the register state in each processing mode.

The base address for each channel of the IIC bus interface registers is as follows:

- IIC-PMIC: H'E60B_0000

IIC bus interface module has the registers below.

Table 53.2 Register Configuration of Bus Interface

| Register Name | Abbreviation | R/W | Offset Address | Size | Second Generation RZ/G Series Products | | | |
|---|--------------|-------|----------------|------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| IIC bus data register | ICDR | R/W | H'000 | 8 | √ | √ | √ | √ |
| IIC bus control register | ICCR | R/W | H'004 | 8 | √ | √ | √ | √ |
| IIC bus status register | ICSR | R/WC0 | H'008 | 8 | √ | √ | √ | √ |
| IIC interrupt control register | ICIC | R/W | H'00C | 8 | √ | √ | √ | √ |
| IIC clock control register low | ICCL | R/W | H'010 | 8 | √ | √ | √ | √ |
| IIC clock control register high | ICCH | R/W | H'014 | 8 | √ | √ | √ | √ |
| IIC automatic transmission interrupt mask register | ICIMSK | R/W | H'050 | 8 | √ | √ | √ | — |
| IIC automatic transmission interrupt flag register | ICINT | R | H'054 | 8 | √ | √ | √ | — |
| IIC automatic transmission processing enabling register | ICACE | R/W | H'058 | 8 | √ | √ | √ | — |
| IIC automatic transmission timer control register 1 | ICTMC1 | R/W | H'060 | 8 | √ | √ | √ | — |
| IIC automatic transmission timer control register 2 | ICTMC2 | R/W | H'064 | 8 | √ | √ | √ | — |
| IIC automatic transmission wait control register | ICTMCW | R/W | H'068 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit start register | ICSTART | R/W | H'070 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit control register | ICATFR | R/W | H'080 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit time register 1 | ICATSET1 | R/W | H'084 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit time register 2 | ICATSET2 | R/W | H'088 | 8 | √ | √ | √ | — |
| IIC automatic transmission receive time register 1 | ICARSET1 | R/W | H'08C | 8 | √ | √ | √ | — |
| IIC automatic transmission receive time register 2 | ICARSET2 | R/W | H'090 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 00 | ICATD00 | R/W | H'100 | 8 | √ | √ | √ | — |

| Register Name | Abbreviation | R/W | Offset Address | Size | Second Generation RZ/G Series Products | | | |
|---|--------------|-----|----------------|------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| IIC automatic transmission transmit data 01 | ICATD01 | R/W | H'104 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 02 | ICATD02 | R/W | H'108 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 03 | ICATD03 | R/W | H'10C | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 04 | ICATD04 | R/W | H'110 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 05 | ICATD05 | R/W | H'114 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 06 | ICATD06 | R/W | H'118 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 07 | ICATD07 | R/W | H'11C | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 08 | ICATD08 | R/W | H'120 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 09 | ICATD09 | R/W | H'124 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 10 | ICATD10 | R/W | H'200 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 11 | ICATD11 | R/W | H'204 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 12 | ICATD12 | R/W | H'208 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 13 | ICATD13 | R/W | H'20C | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 14 | ICATD14 | R/W | H'210 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 15 | ICATD15 | R/W | H'214 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 16 | ICATD16 | R/W | H'218 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 17 | ICATD17 | R/W | H'21C | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 18 | ICATD18 | R/W | H'220 | 8 | √ | √ | √ | — |
| IIC automatic transmission transmit data 19 | ICATD19 | R/W | H'224 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 00 | ICARD00 | R | H'300 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 01 | ICARD01 | R | H'304 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 02 | ICARD02 | R | H'308 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 03 | ICARD03 | R | H'30C | 8 | √ | √ | √ | — |

| Register Name | Abbreviation | R/W | Offset Address | Size | Second Generation RZ/G Series Products | | | |
|--|--------------|-----|----------------|------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| IIC automatic transmission receipt data 04 | ICARD04 | R | H'310 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 05 | ICARD05 | R | H'314 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 06 | ICARD06 | R | H'318 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 07 | ICARD07 | R | H'31C | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 08 | ICARD08 | R | H'320 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 09 | ICARD09 | R | H'324 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 10 | ICARD10 | R | H'400 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 11 | ICARD11 | R | H'404 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 12 | ICARD12 | R | H'408 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 13 | ICARD13 | R | H'40C | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 14 | ICARD14 | R | H'410 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 15 | ICARD15 | R | H'414 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 16 | ICARD16 | R | H'418 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 17 | ICARD17 | R | H'41C | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 18 | ICARD18 | R | H'420 | 8 | √ | √ | √ | — |
| IIC automatic transmission receipt data 19 | ICARD19 | R | H'424 | 8 | √ | √ | √ | — |

Table 53.3 Register State in Each Processing Mode

| Register name | Power-on Reset/Software Reset | Module Standby |
|---------------|-------------------------------|----------------|
| ICDR | Initialized | Retained |
| ICCR | Initialized | Retained |
| ICSR | Initialized | Retained |
| ICIC | Initialized | Retained |
| ICCL | Initialized | Retained |
| ICCH | Initialized | Retained |
| ICIMSK | Initialized | Retained |
| ICINT | Initialized | Retained |
| ICACE | Initialized | Retained |
| ICTMC1 | Initialized | Retained |
| ICTMC2 | Initialized | Retained |
| ICTMCW | Initialized | Retained |
| ICSTART | Initialized | Retained |
| ICATFR | Initialized | Retained |
| ICATSET1 | Initialized | Retained |
| ICATSET2 | Initialized | Retained |
| ICARSET1 | Initialized | Retained |
| ICARSET2 | Initialized | Retained |
| ICATD00 | Initialized | Retained |
| ICATD01 | Initialized | Retained |
| ICATD02 | Initialized | Retained |
| ICATD03 | Initialized | Retained |
| ICATD04 | Initialized | Retained |
| ICATD05 | Initialized | Retained |
| ICATD06 | Initialized | Retained |
| ICATD07 | Initialized | Retained |
| ICATD08 | Initialized | Retained |
| ICATD09 | Initialized | Retained |
| ICATD10 | Initialized | Retained |
| ICATD11 | Initialized | Retained |
| ICATD12 | Initialized | Retained |
| ICATD13 | Initialized | Retained |
| ICATD14 | Initialized | Retained |
| ICATD15 | Initialized | Retained |
| ICATD16 | Initialized | Retained |
| ICATD17 | Initialized | Retained |
| ICATD18 | Initialized | Retained |
| ICATD19 | Initialized | Retained |
| ICARD00 | Initialized | Retained |
| ICARD01 | Initialized | Retained |
| ICARD02 | Initialized | Retained |
| ICARD03 | Initialized | Retained |

| Register name | Power-on Reset/Software Reset | Module Standby |
|----------------------|--------------------------------------|-----------------------|
| ICARD04 | Initialized | Retained |
| ICARD05 | Initialized | Retained |
| ICARD06 | Initialized | Retained |
| ICARD07 | Initialized | Retained |
| ICARD08 | Initialized | Retained |
| ICARD09 | Initialized | Retained |
| ICARD10 | Initialized | Retained |
| ICARD11 | Initialized | Retained |
| ICARD12 | Initialized | Retained |
| ICARD13 | Initialized | Retained |
| ICARD14 | Initialized | Retained |
| ICARD15 | Initialized | Retained |
| ICARD16 | Initialized | Retained |
| ICARD17 | Initialized | Retained |
| ICARD18 | Initialized | Retained |
| ICARD19 | Initialized | Retained |

53.2 Register Description

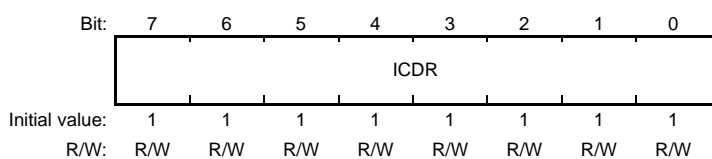
53.2.1 IIC Bus Data Register (ICDR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

ICDR is an 8-bit readable/writable register that is used as a transmit data register when transmitting and a receive data register when receiving.

In transmission mode, write transmit data to this register when the DTE bit in the IIC bus status register (ICSR) is 1.

In receive mode, read receive data from this register when the DTE bit in the IIC bus status register (ICSR) is 1.



53.2.2 IIC Bus Control Register (ICCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

ICCR is an 8-bit readable/writable register that enables or disables the IIC bus interface, specifies the transmission/receive mode, issues START/STOP conditions, and confirms the IIC bus interface bus status.

Table 53.4 ICCR Settings

| Operation | Write Data to ICCR (Transmission Mode) | Write Data to ICCR (Receive Mode) |
|----------------------------------|---|---|
| START condition | H'94 | — |
| Repeated START condition | H'94 | H'D4 (Change from receive mode to transmission mode) |
| transmission/receive mode change | H'81 (Change from transmission mode to receive mode) | — |
| STOP condition | H'90 | H'C0 |

| | | | | | | | | |
|----------------|-----|------|---|-----|---|------|---|--------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | ICE | RACK | — | TRS | — | BBSY | — | SCP |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R/W | R/W | R | R/W | R | R/W | R | (R/W)* |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | ICE | B'0 | R/W | <p>IIC Bus Interface Enable</p> <p>Enables or disables the IIC bus interface.</p> <p>When this bit is set to 1, transfer operation is enabled. When this bit is cleared to 0 during data transmission, the I2C bus is released and all registers in this module are initialized. (See section 53.4, Usage Notes.) When this bit is cleared to 0 after data transmission, all registers in this module are initialized.</p> <p>When the ICCR is updated while this module is operating, this bit must be set to 1.</p> <p>0: IIC bus interface module is disabled. (All internal registers are initialized.)</p> <p>1: IIC bus interface module is enabled.</p> |
| 6 | RACK | B'0 | R/W | <p>Receive Acknowledge</p> <p>Specifies the acknowledge level in receive mode.</p> <p>In reception, this bit provides the acknowledge level for output on the SDA line after the IIC module receives data.</p> <p>0: In receive mode, 0 is output to SDA at acknowledge output timing.</p> <p>1: In receive mode, 1 is output to SDA at acknowledge output timing.</p> |
| 5 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|--------|---|
| 4 | TRS | B'0 | R/W | <p>Transmit/Receive Select</p> <p>Specifies either transmission mode or receive mode.</p> <p>The setting of this bit can be rewritten even during data transfer, but actual operating mode is changed after completion of ongoing transfer including acknowledge bit.</p> <p>0: Master reception mode.</p> <p>1: Master transmission mode.</p> <p>When writing 1 to this bit, also set BBSY and SCP bits at the same time.</p> |
| 3 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 2 | BBSY | B'0 | R/W | <p>Bus Busy</p> <p>This bit is used to issue START and STOP conditions.</p> <p>To issue a START condition, write 1 to this bit and 0 to the SCP bit. A repeated START condition is issued in the same way. To issue a STOP condition, write 0 to this bit and 0 to the SCP bit.</p> <p>0: A STOP condition is issued (the SCP bit is also used).</p> <p>1: A START or repeated START condition is issued (the SCP bit is also used).</p> |
| 1 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 0 | SCP | B'1 | (R/W)* | <p>START Condition/STOP Condition Prohibit</p> <p>Controls issuance of START and STOP conditions. To issue a START condition, write 1 to the BBSY bit and 0 to this bit. A repeated START condition is issued in the same way. To issue a STOP condition, write 0 to the BBSY bit and 0 to this bit. This bit is always read as 1. Writing 1 to this bit is ignored.</p> <p>0: Writing 0 to this bit issues a START or STOP condition, in combination with the BBSY flag.</p> <p>1: Writing 1 to this bit is ignored.</p> |

Note: * For the SCP bit, only writing 0 is valid. Always read as 1.

53.2.3 IIC Bus Status Register (ICSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

ICSR is an 8-bit readable/writable register that performs interrupt request.

| | | | | | | | | |
|----------------|------|------|---|------|-------|-------|-------|-----|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SCLM | SDAM | — | BUSY | AL | TACK | WAIT | DTE |
| Initial value: | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/WC0 | R/WC0 | R/WC0 | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | SCLM | B'1 | R | <p>SCL Monitor</p> <p>Indicates the SCL state sampled by the module clock.</p> <p>0: Indicates that SCL is 0.</p> <p>1: Indicates that SCL is 1.</p> |
| 6 | SDAM | B'1 | R | <p>SDA Monitor</p> <p>Indicates the SDA state sampled by the module clock.</p> <p>0: Indicates that SDA is 0.</p> <p>1: Indicates that SDA is 1.</p> |
| 5 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 4 | BUSY | B'0 | R | <p>IIC Transmit State Bit</p> <p>Indicates the IIC state. After the START condition is generated, the IIC enters the transfer state and this bit is set to 1. After the STOP condition is generated, the IIC enters the non-transfer state and this bit is cleared to 0.</p> <p>When a loss of arbitration is detected, the SDA line is immediately released, but output of the clock signal on SCL continues until frame acknowledgement is completed. This bit is cleared to 0 after end of frame transfer.</p> <p>0: The IIC module is not busy (transfer by the IIC module is not in progress).</p> <p>1: The IIC module is busy (transfer by the IIC module is in progress).</p> <p>Setting condition:</p> <p style="padding-left: 20px;">The IIC module issues the START condition.</p> <p>Clearing condition:</p> <p style="padding-left: 20px;">The IIC module issues the STOP condition.</p> <p>A loss of arbitration is detected (cleared after SCL clock output is stopped).</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|--|
| 3 | AL | B'0 | R/WC0 | <p>Arbitration Lost</p> <p>Indicates a loss of arbitration.</p> <p>The IIC bus interface monitors the SDA. If the IIC bus interface detects data different from the data it sent, it sets this bit to 1 to indicate that the bus has been conflicted. This bit is reset by writing 0 to this bit or performing write access to ICDR in transmit mode or read access to ICDR in receive mode when the DTE bit is 1.</p> <p>When a loss of arbitration is detected during data transfer, the SDA line is immediately released. Output of the clock line on SCL continues until frame acknowledgement is completed. This bit does not depend on the ALE bit in ICIC and is always updated. When the ALE bit in ICIC is 0, an arbitration lost interrupt request is not issued to the interrupt controller. When other masters issue the START condition, a loss of arbitration is detected and operation is stopped.</p> <p>0: Bus arbitration won</p> <ul style="list-style-type: none"> • When data are written to ICDR (transmit mode) or read from ICDR (receive mode) • When 0 is written to this bit after reading AL = 1 <p>1: Arbitration lost</p> <ul style="list-style-type: none"> • When the level of the data being received changes during counting of the period at high level for the SCL clock in master receive mode • When the level on the SCL line becomes low during counting of the period at high level for the SCL clock • When a START condition is detected before this module issues its own START condition • When the monitored external SDA and data being output from this module not being equal |
| 2 | TACK | B'0 | R/WC0 | <p>Transmit Acknowledge Bit</p> <p>Indicates the level of the SDA line during acknowledge cycle in transmit mode. This bit is cleared by writing 0. This bit is always 0 when the TACK bit in ICIC is 0.</p> <p>0: Indicates that acknowledge (low level on SDA line) have been detected during acknowledge cycle.</p> <p>1: Indicates that no acknowledge (high level on SDA line) have been detected during acknowledge cycle.</p> |
| 1 | WAIT | B'0 | R/WC0 | <p>Wait</p> <p>Indicates that this module have entered the wait state after data transmission other than the acknowledge bit.</p> <p>When the value of the WAITE bit in ICIC is 1, this module enters the wait state after data transfer (except for acknowledge bit), with driving the SCL to the low level. At this time, this bit is automatically set to 1 and a WAIT interrupt is generated. By writing 0 to this bit, this module recovers from the wait state.</p> <p>This bit is always 0 when WAITE in ICIC is 0.</p> <p>0: This module is in the normal state.</p> <p>1: This module is in the wait state.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | DTE | B'0 | R | <p>Data Transmit Enable</p> <p>Indicates the transmit state between ICDR and internal transmit/receive buffer. This bit is read only and automatically set or cleared.</p> <p>0:</p> <ul style="list-style-type: none">• When reset is performed• When data is written to ICDR in transmit mode (TRS = 1).• When data is read from ICDR in receive mode (TRS = 0).• When the TRS bit is changed.• When the repeated START condition/STOP conditions are written to the BBSY and SCP bits in ICCR. <p>1:</p> <ul style="list-style-type: none">• When the START/repeated START conditions are generated.• When transmit data is sent to the transmit buffer from ICDR in transmit mode (TRS = 1).• When receive data is sent to ICDR in receive mode (TRS = 0). |

53.2.4 IIC Interrupt Control Register (ICIC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

ICIC is an 8-bit register that enables or disables an interrupt request.

| | | | | | | | | |
|----------------|--------|--------|-------|-------|-----|-------|-------|------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | ICCLB8 | ICCHB8 | TDMAE | RDMAE | ALE | TACKE | WAITE | DTEE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | ICCLB8 | B'0 | R/W | <p>IIC Clock Control Low Bit 8</p> <p>Sets the low-level period of SCL when the period setting for the module in {ICIC[7], ICCL} is H'100 to H'1FF.</p> <p>Table 53.5 shows transfer rate settings.</p> |
| 6 | ICCHB8 | B'0 | R/W | <p>IIC Clock Control High Bit 8</p> <p>Sets the high-level period of SCL when the period setting for the module in {ICIC[6], ICCH} is H'100 to H'1FF.</p> <p>Table 53.5 shows transfer rate settings.</p> |
| 5 | TDMAE | B'0 | R/W | <p>Transmit Data DMA Transfer Request Enable</p> <p>When DTEE is 1, this bit specifies whether data transmit interrupt in transmission mode (TRS = 1) is used as an interrupt to the CPU or a DMA transfer request.</p> <p>When DMA is used for automatic transmission, setup of this bit is unnecessary.</p> <p>0: Uses TDMAE as an interrupt to the CPU.</p> <p>1: Uses TDMAE as a DMA transfer request to DMAC.</p> |
| 4 | RDMAE | B'0 | R/W | <p>Receive Data DMA Transfer Request Enable</p> <p>When the DTEE bit is 1, this bit specifies whether data receive interrupt in receive mode (TRS = 0) is used as an interrupt to the CPU or a DMA transfer request.</p> <p>When DMA is used for automatic transmission, setup of this bit is unnecessary.</p> <p>0: Uses RDMAE as an interrupt to the CPU.</p> <p>1: Uses RDMAE as a DMA transfer request to DMAC.</p> |
| 3 | ALE | B'0 | R/W | <p>Arbitration Lost Interrupt Enable and Function Enable</p> <p>Enables or disables arbitration lost interrupt request.</p> <p>An arbitration lost interrupt is generated if a loss of arbitration occurs when this bit is 1. At this time, the AL bit in ICSR is set to 1. Do not clear this bit to 0 with the AL bit set to 1.</p> <p>0: Disables an arbitration lost Interrupt.</p> <p>1: Enables an arbitration lost interrupt.</p> |

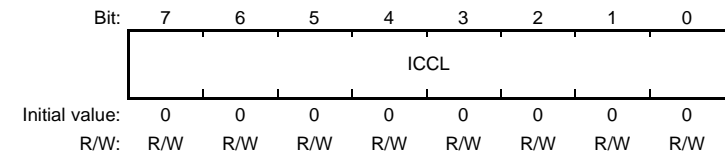
| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | TACKE | B'0 | R/W | <p>Non-acknowledge Detection Interrupt Enable and Function Enable</p> <p>Enables or disables a non-acknowledge detection interrupt request. A non-acknowledge interrupt is generated if non-acknowledge is received when this bit is 1. At this time, the TACK bit in ICSR is set to 1. Do not clear this bit to 0 with the TACK bit set to 1.</p> <p>0: Disables non-acknowledge detection Interrupt. 1: Enables non-acknowledge detection interrupt.</p> |
| 1 | WAITE | B'0 | R/W | <p>Wait Interrupt Enable and Function Enable</p> <p>Enables or disables a wait interrupt request.</p> <p>When WAITE is 1, a wait state in which SCL is driven to the low level follows the transmission of data other than the acknowledge bit.</p> <p>At this time, a WAIT interrupt is generated and the WAIT bit in ICSR is set to 1. Do not clear this bit to 0 with the WAIT bit in ICSR set to 1.</p> <p>0: Disables an interrupt. 1: A wait interrupt is generated.</p> <p>Note: In Automatic transfer mode, set 1 in a WAITE bit.</p> |
| 0 | DTEE | B'0 | R/W | <p>Data Transmit Enable Interrupt</p> <p>Enables or disables a data transmit enable interrupt request.</p> <p>An interrupt request is issued if the DTE bit in ICSR is set to 1 when this bit is 1.</p> <p>0: Disables an interrupt. 1: A data transmit enable interrupt is generated.</p> |

53.2.5 IIC Clock Control Register Low (ICCL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

ICCL is an 8-bit readable/writable register that specifies the low-level period of SCL.

ICCL is initialized to H'00 by a reset.



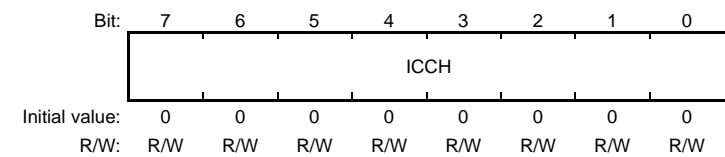
Note: Table 53.5 shows transfer rate settings.

53.2.6 IIC Clock Control Register High (ICCH)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

ICCH is an 8-bit readable/writable register that specifies the high-level period of SCL.

ICCH is initialized to H'00 by a reset.



Note: Table 53.5 shows transfer rate settings.

53.2.7 IIC Automatic Transmission interrupt Mask Register (ICIMSK)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

ICIMSK is an 8-bit register that enables or disables the interrupt request output of automatic transmission.

An interrupt is output to AL interrupt event code. The generated interrupt can be checked by ICINT.

| | | | | | | | | |
|----------------|-------|-------|-------|--------|-----|-------|-----|-----|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | AENDM | TMC2M | TMC1M | TMOUTM | ALM | TACKM | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 7 | AENDM | B'0 | R/W | Automatic Transmission End Interrupt Mask 0: Interrupt is not output to INTC- AP at the time of the end of automatic transmission. 1: Interrupt is output to INTC- AP at the time of the end of automatic transmission. |
| 6 | TMC2M | B'0 | R/W | Timer Count 2 End Interrupt Mask 0: Interrupt is not output to INTC- AP at the time of the end of timer count 2. 1: Interrupt is output to INTC- AP at the time of the end of timer count 2. |
| 5 | TMC1M | B'0 | R/W | Timer Count 1 End Interrupt Mask 0: Interrupt is not output to INTC- AP at the time of the end of timer count 1. 1: Interrupt is output to INTC- AP at the time of the end of timer count 1. |
| 4 | TMOUTM | B'0 | R/W | Timeout Error Interrupt Mask 0: Interrupt is not output to INTC- AP at the time of a timeout error. 1: Interrupt is output to INTC- AP at the time of a timeout error. |
| 3 | ALM | B'0 | R/W | AL Interrupt Mask 0: Interrupt is not output to INTC- AP at the time of the automatic transmission AL. 1: Interrupt is output to INTC- AP at the time of the automatic transmission AL. |
| 2 | TACKM | B'0 | R/W | TACK Interrupt Mask 0: Interrupt is not output to INTC- AP at the time of the automatic transmission TACK. 1: Interrupt is output to INTC- AP at the time of the automatic transmission TACK. |
| 1, 0 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |

53.2.8 IIC Automatic Transmission Interrupt Flag Register (ICINT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

ICINT is an 8-bit register that indicates the interrupt state during automatic transmission.

The interrupt bit can be cleared by writing 1 to ICSTART[7] or clearing ICCR[7] to 0 at the start of automatic operation.

| | | | | | | | | |
|----------------|------|------|------|------|-----|-----|-------|------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | AEND | TMC2 | TMC1 | TMOU | AAL | ATA | AWAIT | ADTE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | AEND | B'0 | R | Automatic Transmission End Flag 0: Under automatic transmission 1: End of automatic transmission |
| 6 | TMC2 | B'0 | R | Timer Count 2 End Flag 0: Timer count 2 is not ended. 1: End of timer count 2 |
| 5 | TMC1 | B'0 | R | Timer Count 1 End Flag 0: Timer count 1 is not ended. 1: End of timer count 1 |
| 4 | TMOU | B'0 | R | Timeout Error Flag 0: No timeout error 1: Timeout error is detected. |
| 3 | AAL | B'0 | R | Automatic Transmission AL Flag 0: No AL 1: AL is detected. |
| 2 | ATA | B'0 | R | Automatic Transmission TACK Flag 0: No TACK 1: TACK is detected. Note: After having finished all the processing appointed in Format, the detection of NACK shows that there was NACK reply during transfer. |
| 1 | AWAIT | B'0 | R | Automatic Transmission WAIT Flag 0: No WAIT 1: WAIT is detected. |
| 0 | ADTE | B'0 | R | Automatic Transmission DTE Flag 0: No DTE 1: DTE is detected. |

53.2.9 IIC Automatic Transmission Processing Enabling Register (ICACE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

ICACE is an 8-bit register that specifies the usage of each function during automatic transmission.

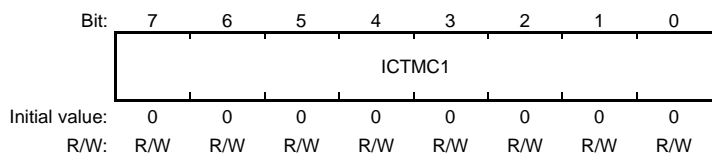
| | | | | | | | | |
|----------------|-------|-------|-------|--------|-----|-------|-------|-----|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | AENDE | TMC2E | TMC1E | TMOUTE | ALE | TACKE | WAITE | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | AENDE | B'0 | R/W | Automatic Transmission End Detection Enable 0: End of automatic transmission is not detected. 1: Detects the end of automatic transmission and outputs it to ICINT. |
| 6 | TMC2E | B'0 | R/W | Timer Count 1 Enable, Timer Count 2 Enable |
| 5 | TMC1E | B'0 | R/W | Specify whether wait by TMC1 or TMC2 timer is inserted before assertion of an automatic transmission interrupt and a timeout error detection interrupt. Refer to Figure 53.3 for the wait timing of TMC1 and TMC2. ICACE[6:5] = B'00: TMC1 and TMC2 are not used. ICACE[6:5] = B'01: Interrupt is asserted after counting by TMC1 is finished. ICACE[6:5] = B'11: Interrupt is asserted after counting by TMC1 and TMC2 is finished. ICACE[6:5] = B'10: Setting prohibited |
| 4 | TMOUTE | B'0 | R/W | Timeout Error Enable When SDA is driven to low even after the STOP condition is generated and counting by TMC1 and TMC2 is finished, a timeout error is detected. 0: Timeout error is not detected. 1: Timeout error is detected and recorded in ICINT. |
| 3 | ALE | B'0 | R/W | Automatic Transmission AL Error-Handling Enable Specifies whether handling for AL errors detected during automatic transmission is performed or not. 0: No automatic AL error handling (AL is not detected) 1: Automatic AL error handling is performed (to release the SDA bus, continue output of SCL until the time corresponding to 1 byte + ACK, and perform retry or stop processing). |
| 2 | TACKE | B'0 | R/W | Automatic Transmission ACK Error-Handling Enable Specifies whether handling for ACK error is performed or not (when error handling is selected, either retry or stop processing is performed). 0: No automatic ACK error handling 1: Automatic ACK error handling is performed (retry or stop processing is performed). |
| 1 | WAITE | B'0 | R/W | WAIT Processing Enable Specifies whether wait processing is inserted during automatic transmission. 0: No automatic WAIT processing. 1: Automatic WAIT processing is performed (waiting for the period specified by the ICTMCW register). Note: In Automatic transfer mode by DMA, set 1 in a WAITE bit. |
| 0 | — | B'0 | R/W | Reserved The write value should always be 0. |

53.2.10 IIC Automatic Transmission Timer Control Register 1 (ICTMC1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

ICTMC1 is an 8-bit register that specifies the delay time for the timing of detection of automatic transmission being completed and timeout errors.



The delay time is specified by ICTMC1 by following formula.

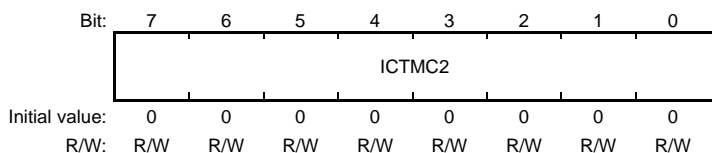
$$\text{Delay time} = (\text{ICTMC1})_{10} \times 200 \times t_{\text{Module clock}}$$

(When the frequency of the module clock is 16.66 MHz, the delay time can be specified within the range from 0 μs up to 3060 μs in 12- μs increments.)

53.2.11 IIC Automatic Transmission Timer Control Register 2 (ICTMC2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

ICTMC2 is an 8-bit register that specifies the delay time for the timing of detection of automatic transmission being completed and timeout errors.



The delay time is specified by ICTMC2 by the following formula.

$$\text{Delay time} = (\text{ICTMC2}) 10 \times 200 \times t_{\text{Module clock}}$$

(When the frequency of the module clock is 16.66 MHz, the delay time can be specified within the range from 0 μs up to 3060 μs in 12-μs increments.)

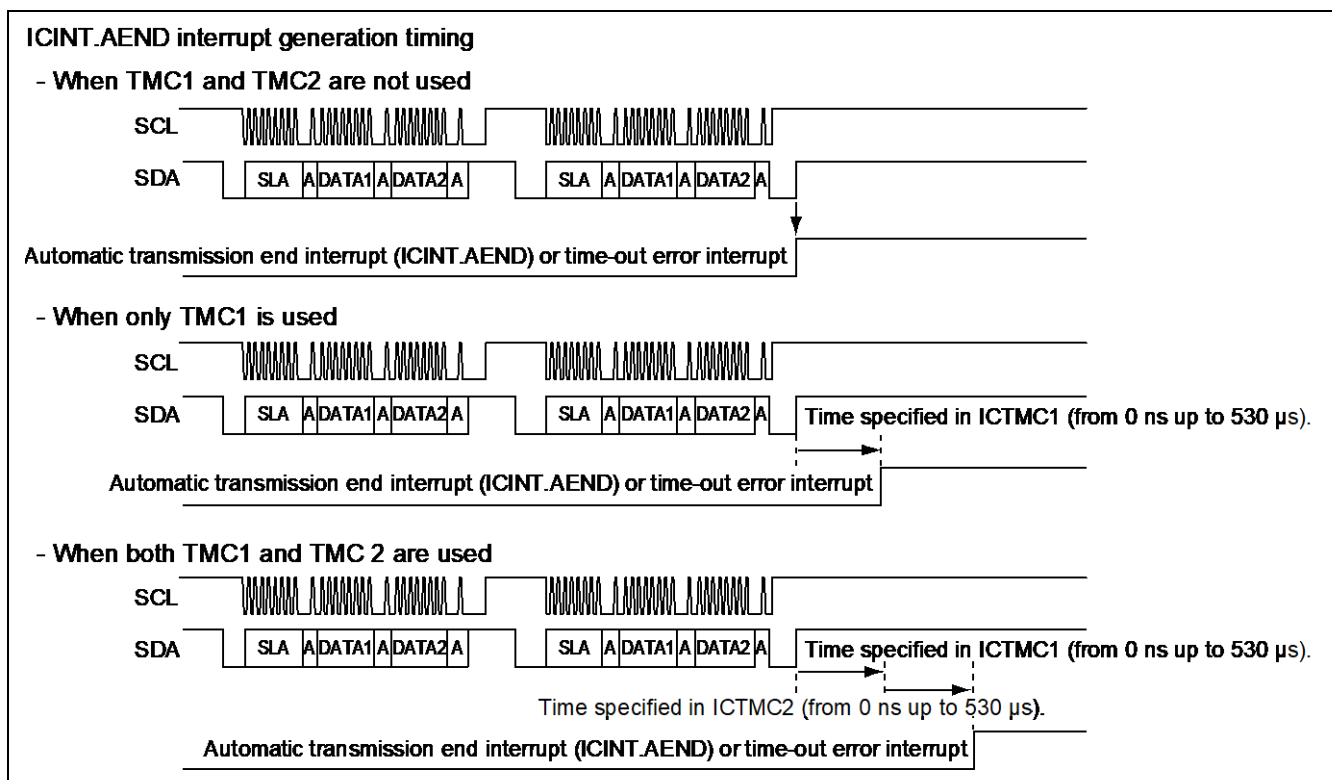
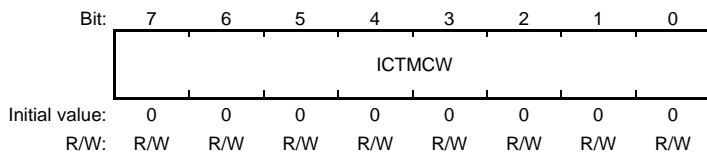


Figure 53.3 Wait Timing Chart of TMC1 and TMC2

53.2.12 IIC Automatic Transmission Wait Control Register (ICTMCW)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

ICTMCW is an 8-bit register that specifies the wait time during automatic transmission. When ICACE.WAITE is set to 1 for activating the wait processing function, set a value other than H'00 to this register.



The wait time is specified by ICTMCW by following formula.

$$\text{Delay time} = (\text{ICTMCW})_{10} \times 20 \times t_{\text{Module clock by ICTMCW}}$$

(When the frequency of the module clock is 16.66 MHz, the delay time can be specified within the range from 0 μs up to 306 μs in 1.2- μs increments.)

Although a wait time can be specified by this register, the period of waiting may be different from the specified value because a transition of the actual SCL bus may be delayed by a clock cycle. Also, in the case of automatic transmission, 1 cycle of waiting is always inserted automatically before the acknowledge cycle.

53.2.13 IIC Automatic Transmission Transmit Start Register (ICSTART)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

ICSTART is an 8-bit register that specifies start, reset, and abort of automatic transmission.

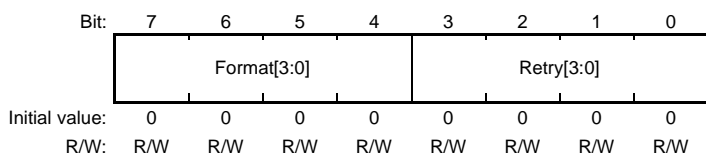
| | | | | | | | | |
|----------------|------------|-----------------|-----------|-----|-----|-----|-------|-------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Auto Start | AutoTrans Reset | Auto Stop | — | — | — | ATDMA | ARDMA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------|---------------|-----|--|
| 7 | AutoStart | B'0 | R/W | Automatic Transmission Start Automatic transmission starts by writing 1 to this bit. Do not clear this bit to 0 during automatic transmission. |
| 6 | AutoTransReset | B'0 | R/W | Automatic Transmission Reset The internal state of automatic transmission is reset by writing 1 to this bit. After reset, write 0 to this bit. Writing 1 to this bit does not initialize automatic transmission-related registers. For register initialization, use reset function by writing 0 to ICCR.ICE. |
| 5 | AutoStop | B'0 | R/W | Automatic Transmission Stop By writing 1 to this bit, the STOP condition is generated at the boundary of a 1byte + ACK-cycle unit and automatic transmission stops. Writing 1 to this bit may result in stopping transmission such that the transfer format is not as expected. 0: Normal operation 1: STOP condition is generated and automatic transmission stops. |
| 4 to 2 | — | All 0 | R/W | Reserved The write value should always be 0. |
| 1 | ATDMA | B'0 | R/W | Automatic Transmission DMA Enable 0: DMA is not used for data transmission by automatic transmission. 1: DMA is used for all data transmission by automatic transmission. |
| 0 | ARDMA | B'0 | R/W | Automatic Reception DMA Enable 0: DMA is not used for data reception by automatic transmission. 1: DMA is used for all data reception by automatic transmission. |

53.2.14 IIC Automatic Transmission Transmit Control Register (ICATFR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

ICATFR is an 8-bit register that specifies an automatic transmission format and the number of retry processing.



| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-------------|---------------|-----|--|
| 7 to 4 | Format[3:0] | H'0 | R/W | <p>IIC Automatic Transmission Format Selection</p> <p>Specify the format of automatic transmission. Refer to Figure 53.5 for a communication format.</p> <p>S = START condition, Sr = Repeated START condition, P = STOP condition</p> <p>H'0: S → Transmission 1 → P</p> <p>H'1: S → Transmission 1 → Reception 1 → P</p> <p>H'2: S → Transmission 1 → Sr → Transmission 2 → P</p> <p>H'3: S → Transmission 1 → Sr → Transmission 2 → Receive 1 → P</p> <p>H'4: S → Transmission 1 → Reception 1 → Sr → Transmission 2 → P</p> <p>H'5: S → Transmission 1 → Reception 1 → Sr → Transmission 2 → Reception 2 → P</p> <p>Others: Setting prohibited</p> |
| 3 to 0 | Retry[3:0] | H'0 | R/W | <p>Retry Number for Automatic Transmission *</p> <p>Specify the maximum number of retry when an AL error or non-acknowledge error occurs. When either of errors has reached the specified number, AL or TACK error is reported.</p> <p>H'0: No retry</p> <p>H'1: Retry once.</p> <p>H'2: Retry up to twice.</p> <p>-</p> <p>H'E: Retry up to 14 times.</p> <p>H'F: Setting prohibited.</p> |

Note: * After having finished all the processing appointed in Format, the error is reflected. When there is an error during transfer, IIC transfers it again from a beginning of Format. Because DMAC does not support this re-try, the RETRY processing using DMAC is not possible.

53.2.15 IIC Automatic Transmission Transmit Time Register 1 (ICATSET1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

ICATSET1 is an 8-bit register that specifies the number of data of the transmission 1 in the format specified by ICATFR[7:4].

The maximum value which can be specified in this register is 10 when using the ICATD00 to ICATD09 registers, and 256 when using a DMA transfer.

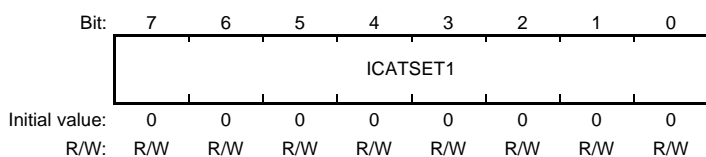
A slave address is also contained in the transmission data.

Specify the values as follows when ICATD00 to ICATD09 are in use.

- 1-byte transmission ICATSET1[7:0] = H'00
- 2-byte transmission ICATSET1[7:0] = H'01
- 10-byte transmission ICATSET1[7:0] = H'09

Specify the size of transmission for a DMA transfer. In the case of DMA, transfer the data to only ICATD00.

- 256-byte transmission ICATSET1[7:0] = H'FF (maximum)



53.2.16 IIC Automatic Transmission Transmit Time Register 2 (ICATSET2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

ICATSET2 is an 8-bit register that specifies the number of data of the transmission 2 in the format specified by ICATFR[7:4].

The maximum number which can be specified in this register is 10 when using the ICATD10 to ICATD19 registers, and 256 when using a DMA transfer.

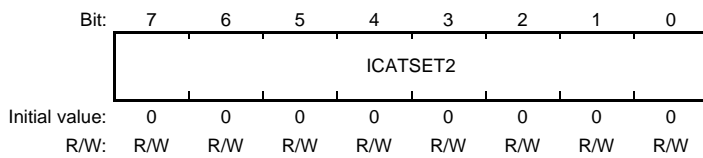
A slave address is also contained in the transmission data.

Specify the values as follows when ICATD10 to ICATD19 are in use.

- 1-byte transmission ICATSET2[7:0] = H'00
- 2-byte transmission ICATSET2[7:0] = H'01
- 10-byte transmission ICATSET2[7:0] = H'09

Specify the size of transmission for a DMA transfer. In the case of DMA, transfer the data to only ICATD10.

256-byte transmission ICATSET2[7:0] = H'FF (maximum)



53.2.17 IIC Automatic Transmission Receive Time Register 1 (ICARSET1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

ICARSET1 is an 8-bit register that specifies the number of data of the reception 1 in the format specified by ICATFR[7:4].

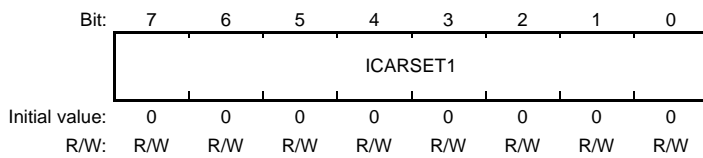
The maximum number which can be specified in this register is 10 when using the ICARD00 to ICARD09 registers, and 256 when using a DMA transfer.

Specify the values as follows when ICARD00 to ICARD09 are in use.

- 1-byte transmission ICARSET1[7:0] = H'00
- 2-byte transmission ICARSET1[7:0] = H'01
- 10-byte transmission ICARSET1[7:0] = H'09

Specify the size of transmission for a DMA transfer. In the case of DMA, read the data only from ICARD00.

256-byte transmission ICARSET1[7:0] = H'FF (maximum)



53.2.18 IIC Automatic Transmission Receive Time Register 2 (ICARSET2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

ICARSET2 is an 8-bit register that specifies the number of data of the reception 1 in the format specified by ICATFR[7:4].

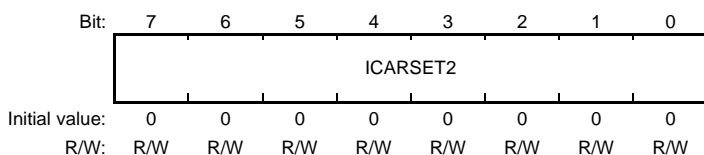
The maximum number which can be specified in this register is 10 when using the ICARD10 to ICARD19 registers, and 256 when using a DMA transfer.

Specify the values as follows when ICARD10 to ICARD19 are in use.

- 1-byte transmission ICARSET2[7:0] = H'00
- 2-byte transmission ICARSET2[7:0] = H'01
- 10-byte transmission ICARSET2[7:0] = H'09

Specify the size of transmission for a DMA transfer. In the case of DMA, read the data only from ICARD10.

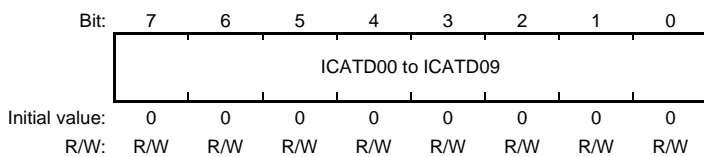
- 256-byte transmission ICARSET2[7:0] = H'FF (maximum)



53.2.19 IIC Automatic Transmission Transmit Data 00 to 09 (ICATD00 to ICATD09)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

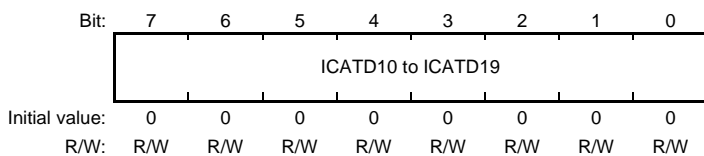
ICATD00 to ICATD09 are 8-bit registers that are used to store the transmit data of transmission 1.



53.2.20 IIC Automatic Transmission Transmit Data 10 to 19 (ICATD10 to ICATD19)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

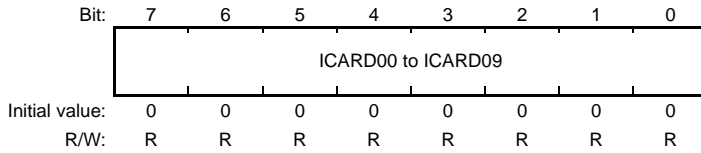
ICATD10 to ICATD19 are 8-bit registers that are used to store the transmit data of transmission 2.



53.2.21 IIC Automatic Transmission Receipt Data 00 to 09 (ICARD00 to ICARD09)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

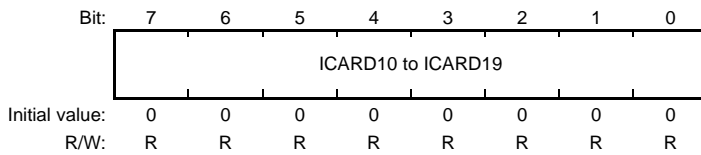
ICARD00 to ICARD09 are 8-bit registers that are used to store the received data of reception 1.



53.2.22 IIC Automatic Transmission Receipt Data 10 to 19 (ICARD10 to ICARD19)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

ICARD10 to ICARD19 are 8-bit registers that are used to store the received data of reception 2.



53.2.23 Transfer Rate

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The IIC transfer rate can be specified by the settings in ICCL and ICCH is based on the module clock.

Table 53.5 lists transfer rate settings, and Figure 53.4 shows SCL waveforms.

Table 53.5 Transfer Rate Maximum Settings Example

| | Module clock | ICIC[7] + ICCL | ICIC[6] + ICCH | Transfer Rate* |
|-----------|--------------|----------------|----------------|----------------|
| Standard | 8.33 MHz | H'017 | H'010 | 99.21 kHz |
| | 10 MHz | H'01B | H'014 | 100.00 kHz |
| | 12.5 MHz | H'022 | H'01A | 99.21 kHz |
| | 16.66 MHz | H'02E | H'023 | 99.21 kHz |
| Fast-mode | 8.33 MHz | H'005 | H'003 | 378.79 kHz |
| | 10 MHz | H'006 | H'004 | 384.62 kHz |
| | 12.5 MHz | H'008 | H'005 | 390.63 kHz |
| | 16.66 MHz | H'00B | H'007 | 396.83 kHz |

Note: * The transfer rates in above settings are theoretical values. The actual transfer rates are generally smaller than those theoretical values due to the pull-up resistor (Rp) and the capacitance (Cp). However, there is no harm in communication because the actual transfer rates satisfy the I2C bus specifications. (Smaller ICCL and ICCH can increase the actual transfer rates. In this case, though, set ICCL and ICCH to ensure that t_{LOW} and t_{HIGH} are within the specifications.)

Calculation example of ICCL and ICCH:

- Conditions

Module clock: 16.66 MHz

IIC transfer rate: 100 kHz

L/H sample ratio in SCL: L/H = 5/4

- Result

$$ICIC[7] + ICCL = ((Module\ clock / IIC\ transfer\ rate) \times (L / (L + H)) - 1) / 2$$

$$= (((16.66 \times 10^6) / (100 \times 10^3)) \times (5 / (5 + 4)) - 1) / 2 = 45 = H'2D$$

$$ICIC[6] + ICCH = ((Module\ clock / IIC\ transfer\ rate) \times (H / (L + H)) - 5) / 2$$

$$= (((16.66 \times 10^6) / (100 \times 10^3)) \times (4 / (5 + 4)) - 5) / 2 = 34 = H'22$$

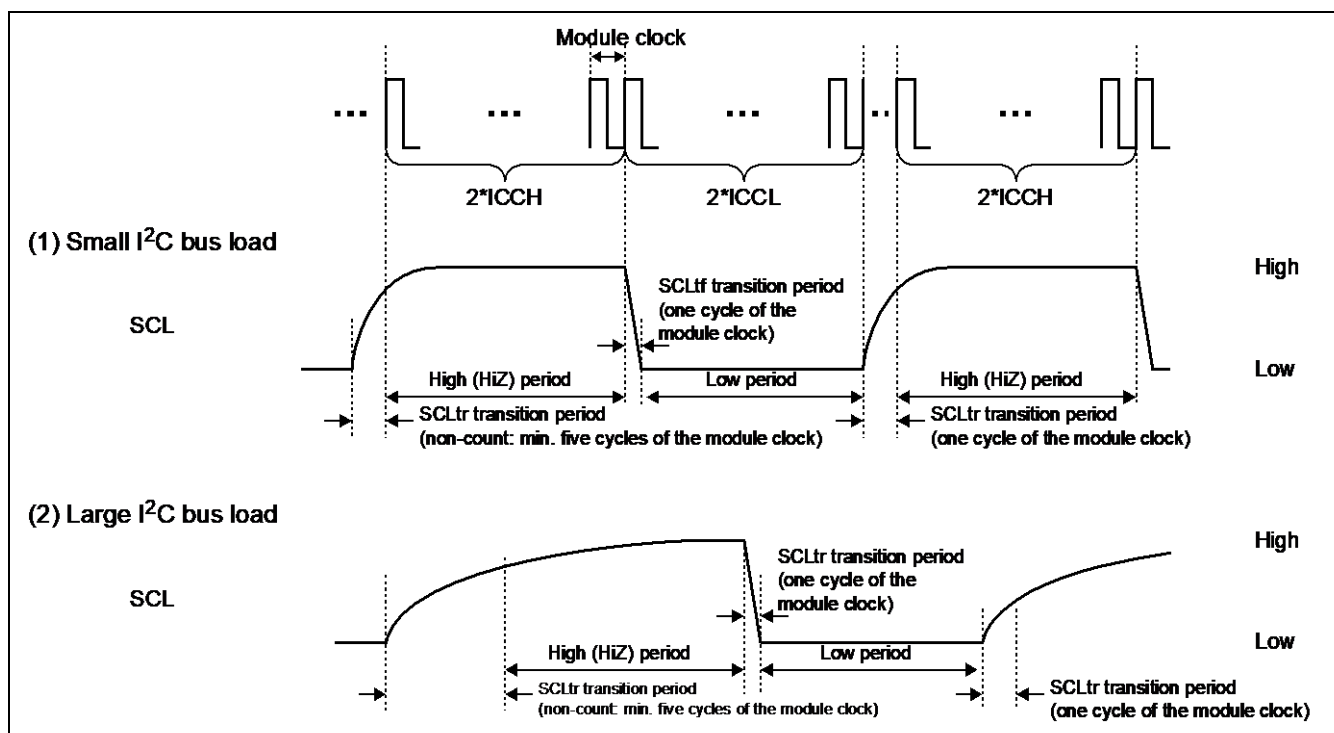


Figure 53.4 SCL Waveforms

53.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

53.3.1 I2C Bus Data Format

The I2C bus is based on protocol, which contains the slave address, data, and an acknowledge bit. These are shown in Figure 53.5. The first frame following a START condition always consists of 8 bits.

The symbols used in Figure 53.5 are explained in Table 53.6.

Table 53.6 I2C Bus Data Format Symbols

| Symbol | Description |
|--------|---|
| S | START condition. The master device drives SDA from high to low while SCL is high. |
| Sr | Repeated START condition. The master device drives SDA from high to low while SCL is high. |
| SLA | Slave address. The master device selects a slave device. |
| R/W | Indicates the direction of data transfer: from the slave device to the master device when R/W is 1, or from the master device to the slave device when R/W is 0. |
| A | Acknowledge. The receiving device (the slave device in master transmit mode, or the master device in master receive mode) drives SDA low to acknowledge a transfer. |
| DATA | Transmitted or received data. The bit length is 8 bits. |
| P | STOP condition. The master device drives SDA from low to high while SCL is high. |

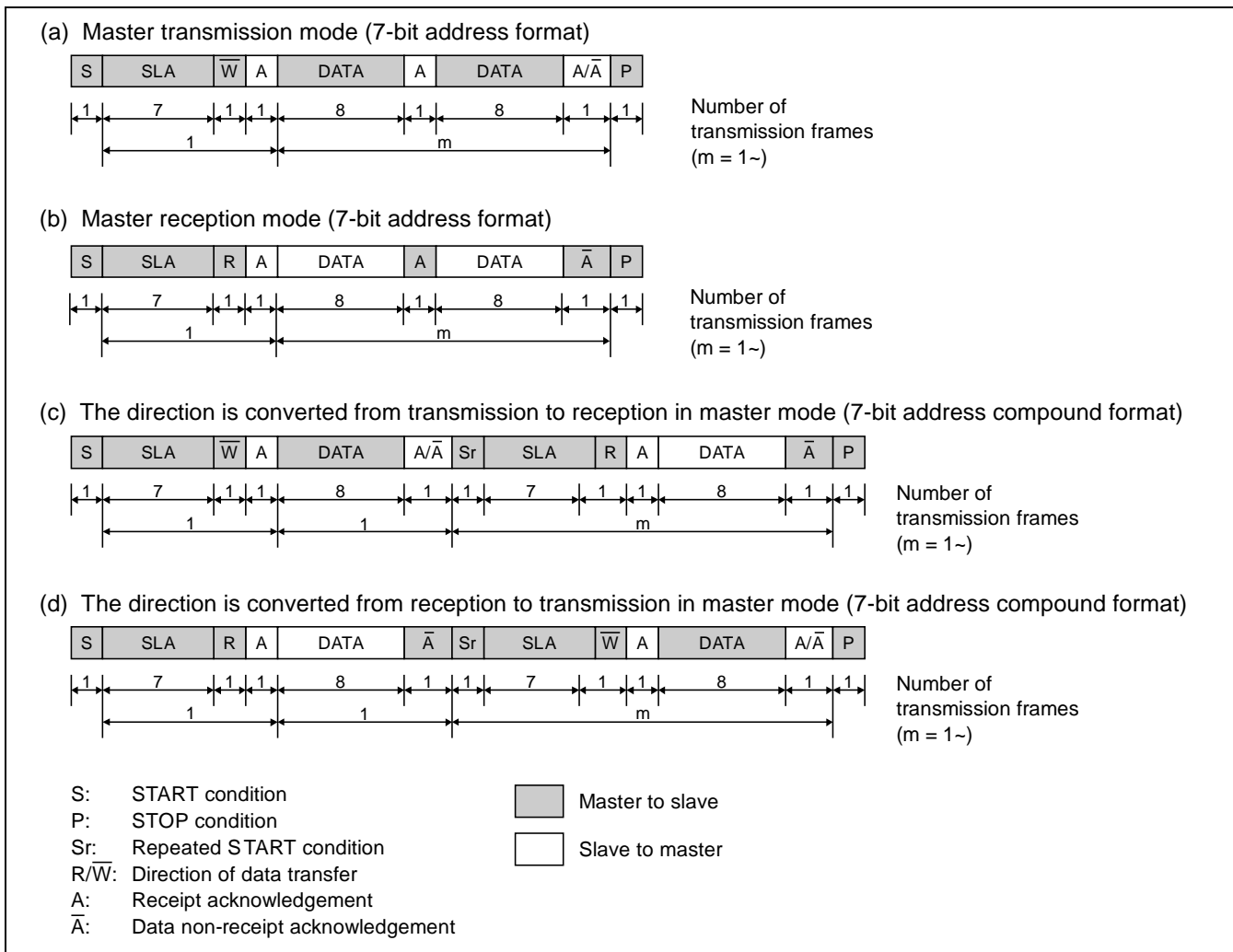


Figure 53.5 I2C Bus Data Formats (7-Bit Address Format)

53.3.2 Master Transmit Operation (Using WAIT Function)

In master transmit mode, the master device outputs the transmission clock and transmission data, and receives the acknowledge signal after receiving data. The transmission procedure and operations are described below.

1. Write ICCR.ICE to 0 to reset the IIC module. (Write ICCR = H'01)
2. Set ICCR.ICE to activate the IIC module. (Write H'81 to ICCR)
3. Set ICIC[7]+ICCL, ICIC[6]+ICCH, and ICIC[3:0], according to the operating mode.
4. Write H'94 to ICCR to issue the START condition. The DTE bit in ICSR is then set to 1.
5. Wait for a DTE interrupt or until ICSR.DTE becomes 1.
6. Clear ICIC.DTEE to disable a DTE interrupt when the ICSR.DTE bit becomes 1.
And write transmission data (slave address and write bit) to ICDR. The ICSR.DTE bit is then automatically cleared.
Data transmission starts after data is sent to the internal transmit buffer.
7. Wait for a WAIT interrupt or until ICSR.WAIT becomes 1.
8. If more transmission data remains, write transmission data (ex. register address or data) to ICDR. And clear ICSR.WAIT to exit from the WAIT status. After exiting from the WAIT state, go back to step 7.
9. If no more transmission data remains, write H'90 to ICCR to issue the STOP condition.
And clear ICSR.WAIT to exit from the WAIT status. After exiting, go to step 10.
10. Wait until ICSR.BUSY is cleared.
11. Clear ICCR.ICE to 0 to reset the IIC module. (Write ICIC = H'01)

Figure 53.6 shows the flowchart of the master transmission procedure.

By repeating steps 7 and 8, data can be transmitted consequently.

A non-acknowledge interrupt occurs if the received acknowledge signal is 1 (TACK = 1). If the received acknowledge bit is 1, it is recommended to STOP transmission after the STOP condition is generated. Refer to section 53.3.11, Non-Acknowledge Operation, for the stop method.

To change the transmission mode to the receive mode, write H'81 to ICCR, instead of generating the STOP condition in step 9.

To generate a repeated START condition, write H'94 to ICCR, instead of generating the STOP condition in step 9.

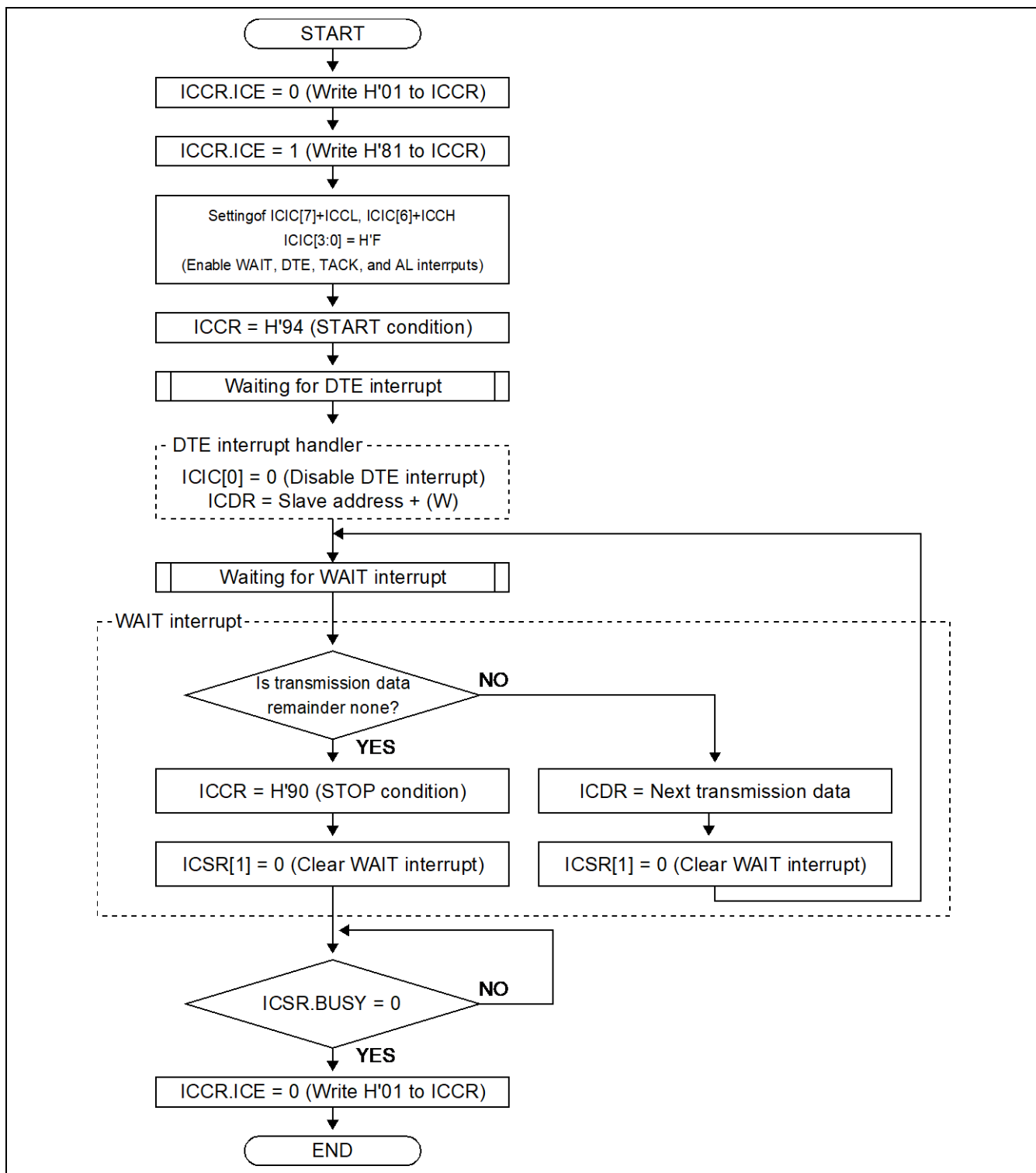


Figure 53.6 Example Flow for Transmission (a) Master Transmission Mode (Using WAIT Function)

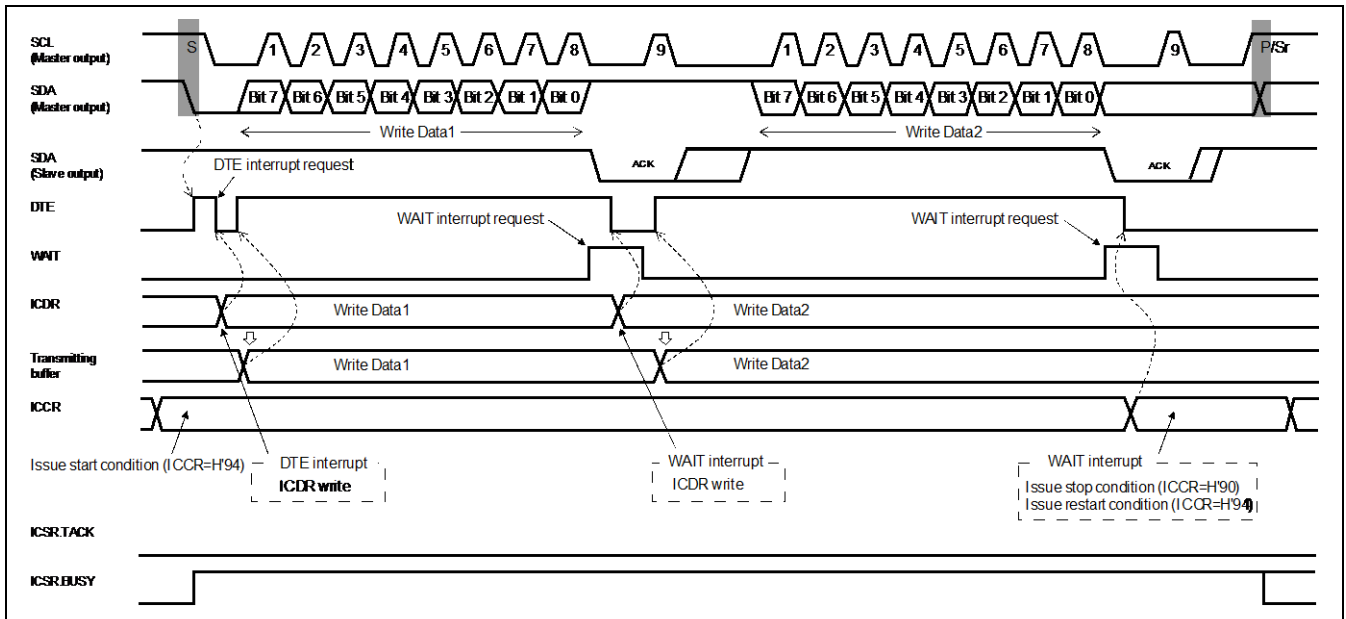


Figure 53.7 Example for Transmission (a) Master Transmission Mode (Using WAIT Function)

53.3.3 Master Receive Operation (Using WAIT Function)

In master receive mode, the master device outputs the receive clock, and returns an acknowledge signal after the slave device transmits data. The reception procedure and operations in master receive mode are described below.

(1) When two or more bytes of data are received continuously

1. Write ICCR.ICE to 0 to reset the IIC module. (Write ICCR = H'01)
2. Set ICCR.ICE to activate the IIC module. (Write H'81 to ICCR)
3. Set ICIC[7]+ICCL, ICIC[6]+ICCH, and ICIC[3:0], according to the operating mode.
4. Write H'94 to ICCR to issue the START condition. The DTE bit in ICSR is then set to 1.
5. Wait for a DTE interrupt.
6. Clear ICIC.DTEE to disable a DTE interrupt when the ICSR.DTE bit becomes 1.
And write transmission data (slave address and write bit) to ICDR. The ICSR.DTE bit is then automatically cleared. Data transmission starts after data is sent to the internal transmit buffer.
7. Wait for a WAIT interrupt.
8. If more transmission data remains, write transmission data (ex. register address) to ICDR. And clear ICSR.WAIT to exit from the WAIT status. After exiting from the WAIT status, go back to step 7.
9. If no more transmission data remains, clear ICSR.WAIT to exit from the WAIT status. After exiting, go to step 10.
10. Write H'94 to ICCR to issue the repeated START condition, and set ICIC.DTEE to enable a DTE interrupt.
11. Wait for a DTE interrupt.
12. Clear ICIC.DTEE to disable a DTE interrupt when the ICSR.DTE bit becomes 1. And write transmission data (slave address and read bit) to ICDR. The ICSR.DTE bit is then automatically cleared. Data transmission starts after data is sent to the internal transmit buffer.
13. Wait for a WAIT interrupt.
14. Write H'81 to ICCR to changing the transmission mode to the receive mode. And clear ICSR.WAIT to exit from the WAIT status.
15. Wait for a WAIT interrupt.
16. If this interrupt is not final reception data for I2C bus, execute below action.
 - Read ICDR to get reception data, then ICSR.DTE is cleared. *1
 - Clear ICSR.WAIT to exit from the WAIT status.

*1 This reading result is previous receiving result. So, if this WAIT interruption is 1st time WAIT of reading phase, skip this ICDR reading.
17. If this interrupt is final reception data for I2C bus, execute below action.
 - Read ICDR to get reception data, then ICSR.DTE is cleared. *2
 - Set H'C0 to ICCR to make STOP condition.
 - Set ICIC.DTEE to enable a DTE interrupt.
 - Clear ICSR.WAIT to exit from the WAIT status.

*2 This read result is previous one from final reception data.
18. Wait for a DTE interrupt.
19. Clear ICIC.DTEE to disable a DTE interrupt when the ICSR.DTE bit becomes 1.
Read data from ICDR. The ICSR.DTE bit is then automatically cleared. *3
- *3 This reading result is final reception data
20. Wait until ICSR.BUSY is cleared.
21. Clear ICCR.ICE to 0 to reset the IIC module. (Write ICCR = H'01)

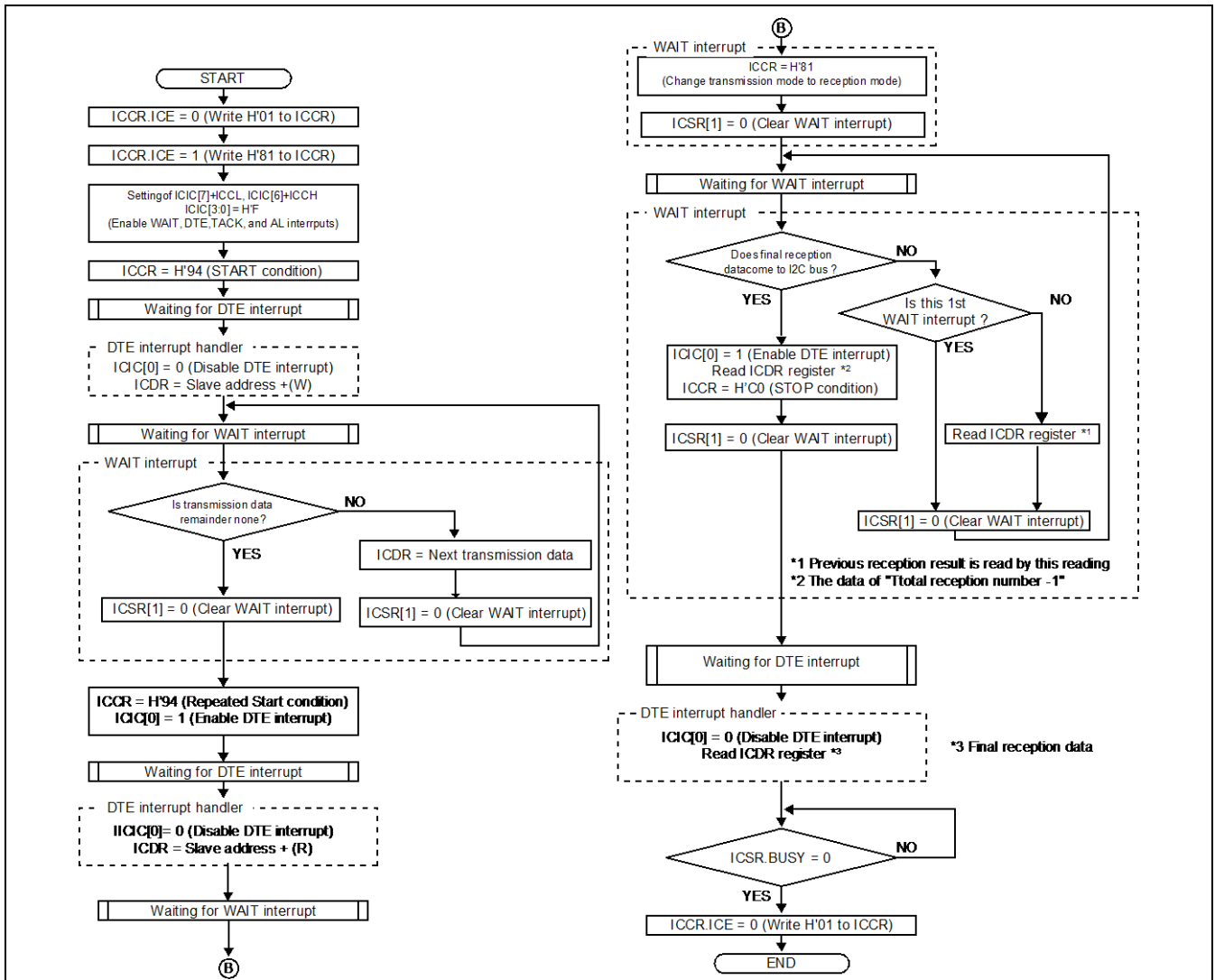


Figure 53.8 Example Flow for n-Byte Write/n-Byte Read (c) The Direction is Converted from Transmission to Reception (Using WAIT Function)

Figure 53.8 is the flow of this.

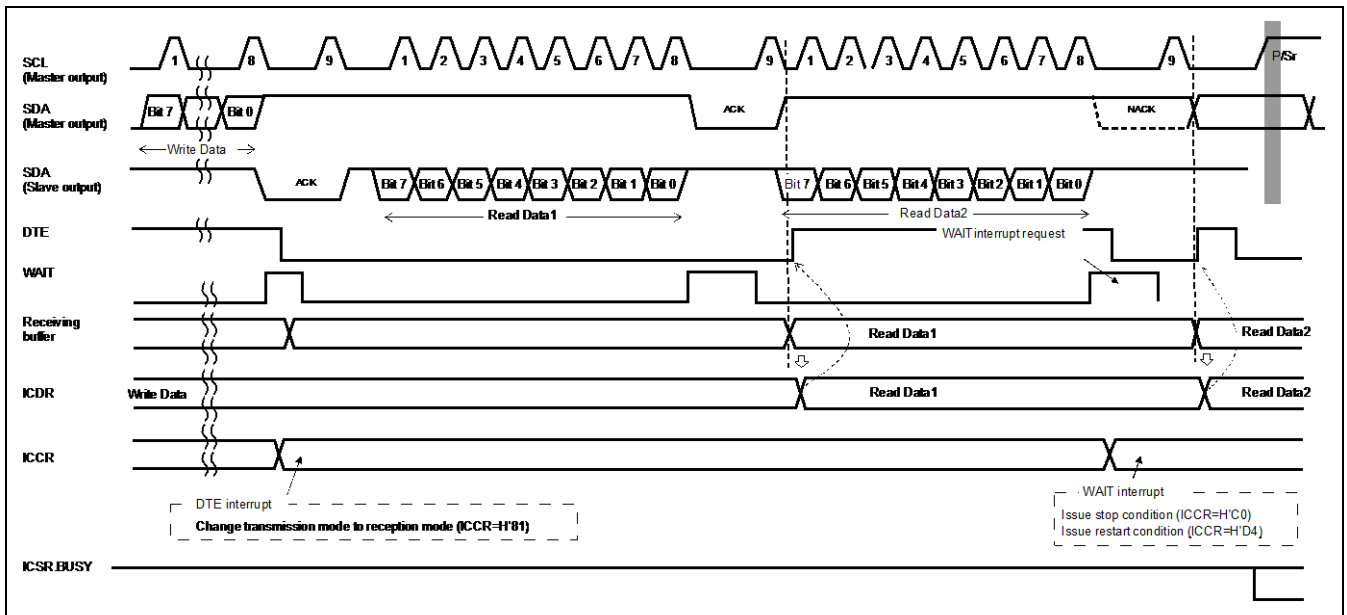


Figure 53.9 Example for Read (c) Master Reception Mode (Using WAIT Function)

(2) When only one byte of data is received

When only one byte data is received, use a WAIT interrupt to issue the STOP or repeated START condition by writing to ICCR.

1. Write ICCR.ICE to 0 to reset the IIC module. (Write ICCR = H'01)
2. Set ICCR.ICE to activate the IIC module. (Write H'81 to ICCR)
3. Set ICIC[7]+ICCL, ICIC[6]+ICCH, and ICIC[3:0], according to the operating mode.
4. Write H'94 to ICCR to issue the START condition. The DTE bit in ICSR is then set to 1.
5. Wait for a DTE interrupt.
6. Clear ICIC.DTEE to disable a DTE interrupt when the ICSR.DTE bit becomes 1.
And write transmission data (slave address and write bit) to ICDR. The ICSR.DTE bit is then automatically cleared.
Data transmission starts after data is sent to the internal transmit buffer.
7. Wait for a WAIT interrupt.
8. Write H'81 to ICCR for changing the transmission mode to the receive mode. And clear ICSR.WAIT to exit from the WAIT status.
9. Wait for a WAIT interrupt.
10. Set ICCR to H'C0 for the STOP condition. Set ICIC.DTEE to enable a DTE interrupt, then clear ICSR.WAIT to exit from the WAIT status.
11. Wait for a DTE interrupt.
12. Clear ICIC.DTEE to disable a DTE interrupt when the ICSR.DTE bit becomes 1. Read data from ICDR. The ICSR.DTE bit is then automatically cleared.
13. Wait until ICSR.BUSY is cleared.
14. Clear ICCR.ICE to 0 to reset the IIC module. (Write ICIC = H'01)

Figure 53.10 shows the flowchart of 1-byte reception.

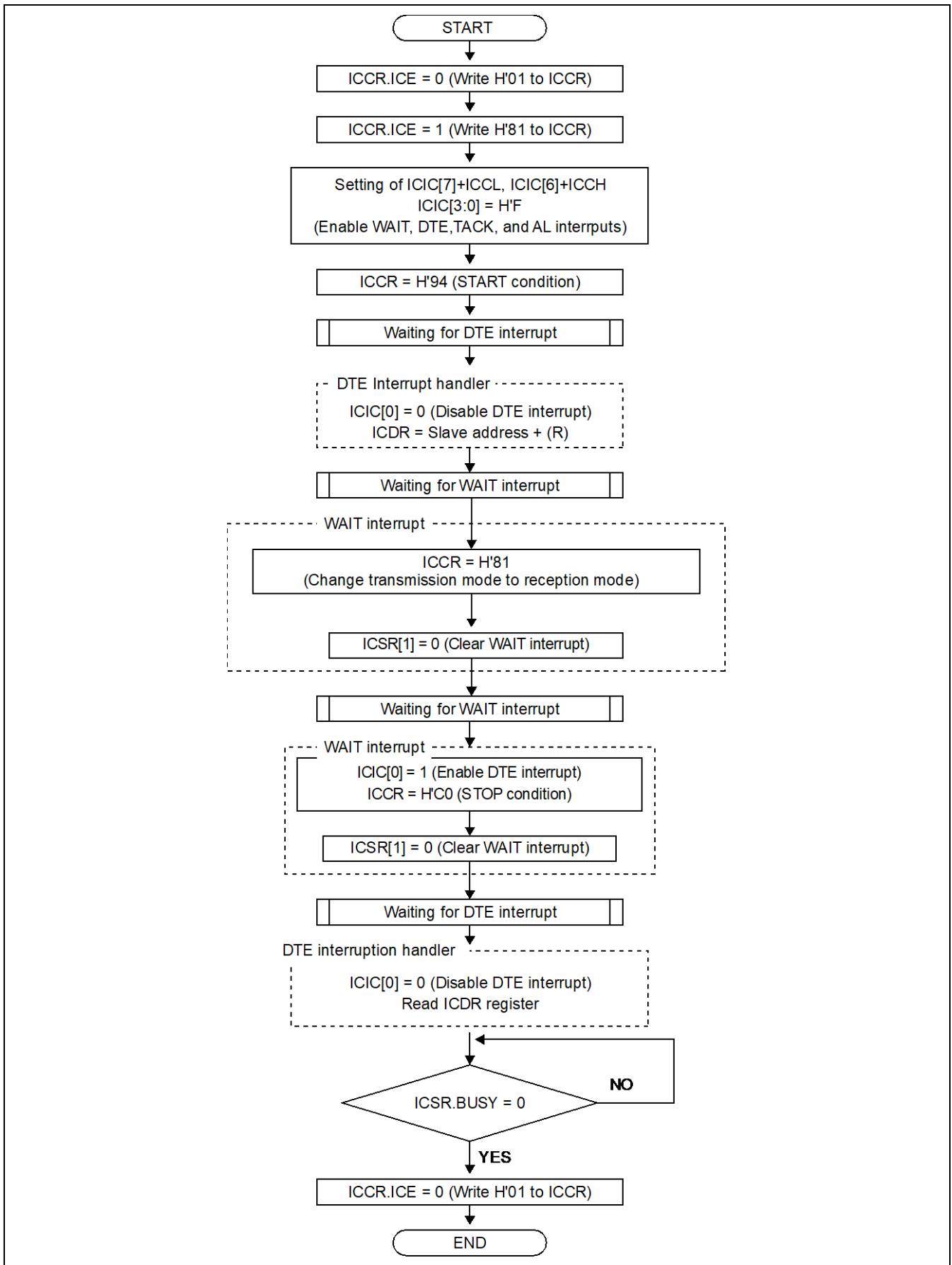


Figure 53.10 Example of Software Flow for 1-Byte Write/1-Byte Read (Using WAIT Function)

53.3.4 IIC Continuous Transmission by DMA

In DMA transmit mode, the IIC module asserts a DMA request to the DMAC. The received DMAC writes data to ICDR, and returns DMARACK to the IIC module. The transmission procedure and operations are described below.

1. Write ICCR.ICE to 0 to reset the IIC module. (Write ICCR = H'01)
2. Set ICCR.ICE to activate the IIC module. (Write H'81 to ICCR)
3. Set ICIC[7]+ICCL, ICIC[6]+ICCH, and ICIC[3:0], and ICIC[5], according to the operating mode.
4. Set DMAC including number of transmission data.
5. Write H'94 to ICCR to issue the START condition.
6. DMAC writes the transmit data to ICDR from the memory space specified by DMAC register until transmission finishes.
7. Wait for a transfer end interrupt from DMAC.
8. Clear ICIC.TDMA to disable DMA transmission after a transfer end interrupt is asserted.
9. Wait for a WAIT interrupt.
10. Clear ICSR.WAIT to exit from the WAIT status. After the acknowledge clock, the STOP condition is generated and operation is stopped.
11. Wait until ICSR.BUSY is cleared.
12. Clear ICCR.ICE to 0 to reset the IIC module. (Write ICIC = H'01)

Figure 53.11 shows the flow of data transmission with DMA.

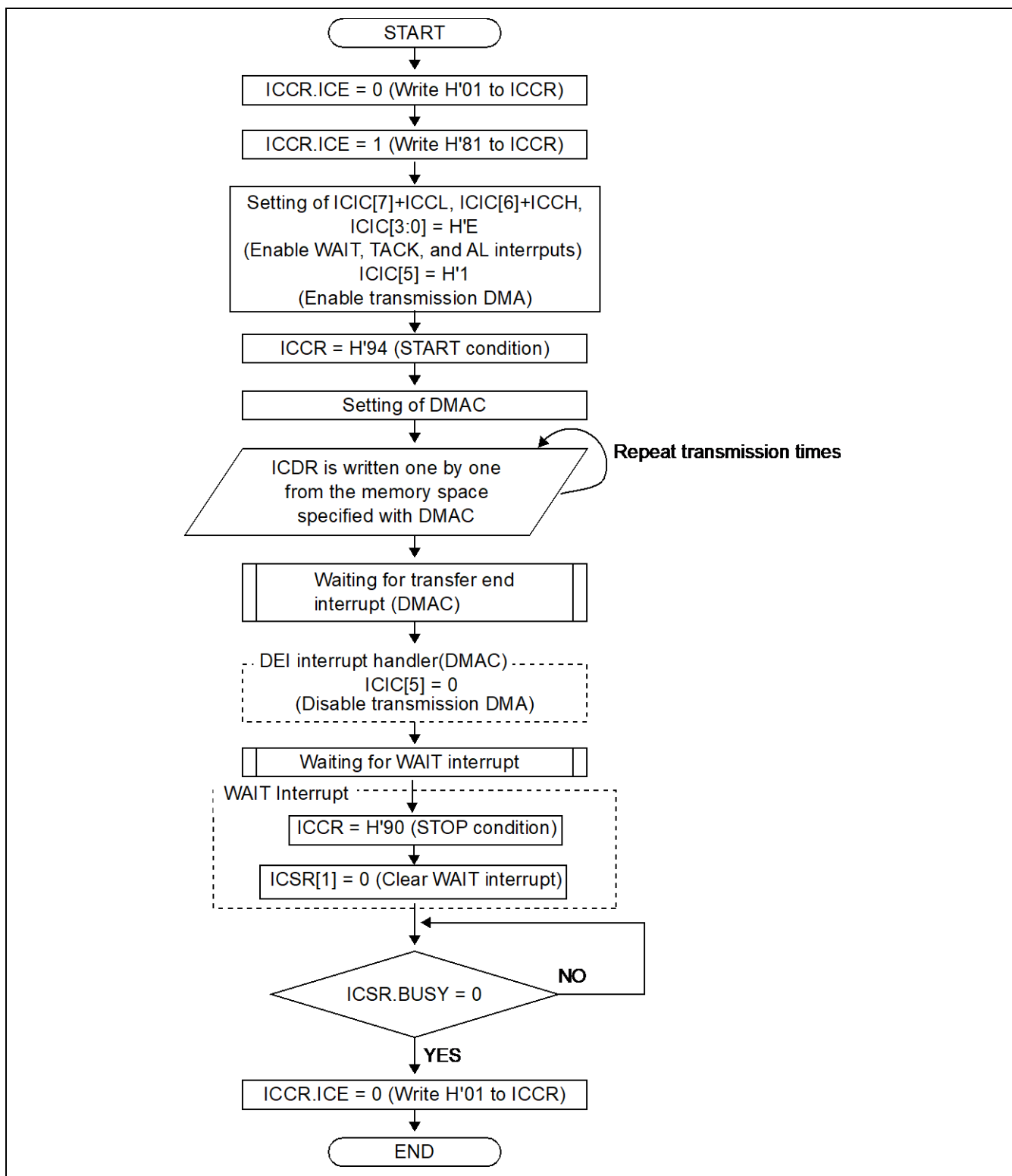


Figure 53.11 Example Flow for Transmission (a) Master Transmission Mode (Using WAIT and DMA Function)

53.3.5 IIC Continuous Receive by DMA

In DMA receive mode, the master device asserts DMA request to DMAC. The DMAC that received the request reads the received data from ICDR, and returns DMARACK to the IIC module. In case of receiving data continuously by DMA, 4 bytes or more receive data is required. The reception procedure and operations are described below.

1. Write ICCR.ICE to 0 to reset the IIC module. (Write ICCR = H'01)
2. Set ICCR.ICE to activate the IIC module. (Write H'81 to ICCR)
3. Set ICIC[7] + ICCL, ICIC[6]+ICCH, and ICIC[3:0], according to the operating mode.
4. Write H'94 to ICCR to issue the START condition. The DTE bit in ICSR is then set to 1.
5. Wait for a DTE interrupt.
6. Clear ICIC.DTEE to disable a DTE interrupt when the ICSR.DTE becomes 1.
And write transmission data (slave address and write bit) to ICDR. The ICSR.DTE bit is then automatically cleared. Data transmission starts after data is sent to the internal transmit buffer.
7. Wait for a WAIT interrupt.
8. If one or more transmission data remains, write transmission data (ex. register address) to ICDR. And clear ICSR.WAIT to exit from the WAIT status. After exiting from the WAIT state, go back to step 7.
9. If no more transmission data remains, clear ICSR.WAIT to exit from the WAIT status. After exiting, go to step 10.
10. Write H'94 to ICCR to issue the repeated START condition, and set ICIC.DTEE to enable a DTE interrupt,
11. Wait for a DTE interrupt.
12. Clear ICIC.DTEE to disable a DTE interrupt when the ICSR.DTE bit becomes 1. And write transmission data (slave address and read bit) to ICDR. The ICSR.DTE is then automatically cleared. Data transmission starts after data is sent to the internal transmit buffer.
13. Wait for a WAIT interrupt.
12. Write H'81 to ICCR for changing the transmission mode to the receive mode. And clear ICSR.WAIT to exit from the WAIT status.
14. DMAC reads the received data from ICDR, and stores them into the destination area until receiving data is finished.
15. Wait for a transfer end interrupt of DMAC.
16. Clear ICIC.RDMA to disable the received DMA.
17. Wait for a WAIT interrupt.
20. Read data from ICDR. And ICCR = H'C0 for the STOP condition.
21. Set ICIC.DTEE to enable a DTE interrupt. And clear ICSR.WAIT to exit from the WAIT status.
22. Wait for a DTE interrupt.
23. Clear ICIC.DTEE to disable a DTE interrupt when the ICSR.DTE bit becomes 1. Read data from ICDR. The ICSR.DTE bit is then automatically cleared.
24. Wait until CSR.BUSY is cleared.
25. Clear ICCR.ICE to 0 to reset the IIC module. (Write ICCR = H'01)

Notice: DMA reception is allowed that reception total times is 3 or more.

Figure 53.12 is the flow of this.

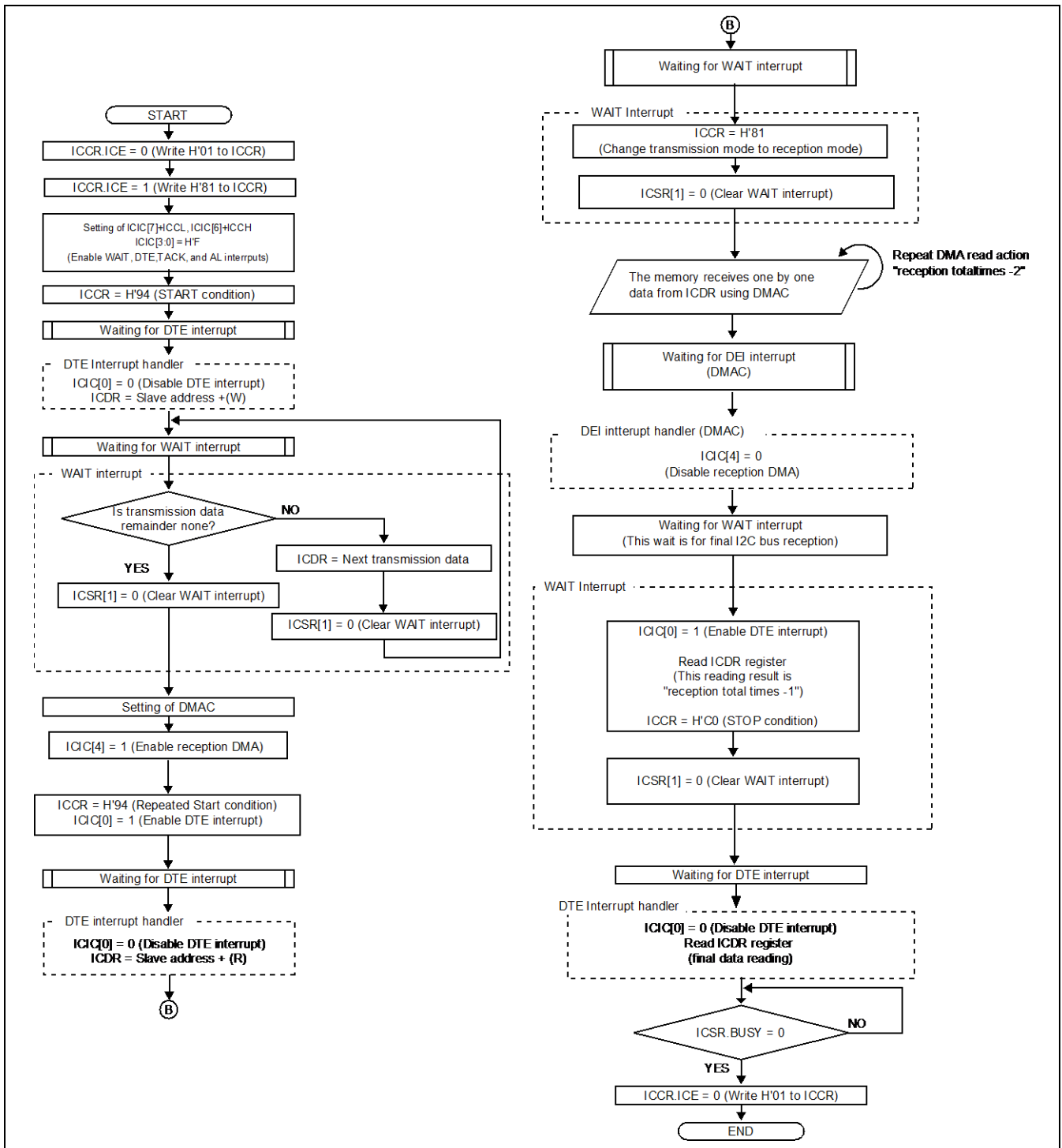


Figure 53.12 Example Flow for n-Byte Write/n-Byte Read (c) The Direction is Converted From Transmission to Reception (Using WAIT and DMA Function)

53.3.6 Automatic transfer mode

An automatic transfer mode can automatically carry out transfer of IIC by appointing ICATFR.Format.

1. Write ICCR.ICE to 0 to reset the IIC module. (Write ICCR = H'01)
2. Set ICCR.ICE to activate the IIC module. (Write H'81 to ICCR)
3. Set ICIC[7] + ICCL, ICIC[6] + ICCH, ICIC[1] = 1, ICACE, and ICIMSK.
4. Set transfer format to ICATFR.Format. And set the number of the transfer and data like Table 53.7.
5. Write H'80 in the ICSTART to start IIC transfer.
6. Wait for a AEND interrupt.
7. Clear ICINT.AEND to disable a AEND interrupt.
8. Because data are stored away at the time of reception by ICARD**, begin to read data. (** = 00 to 09, 10 to 19)
9. Clear ICCR.ICE to 0 to reset the IIC module. (Write ICCR = H'01)

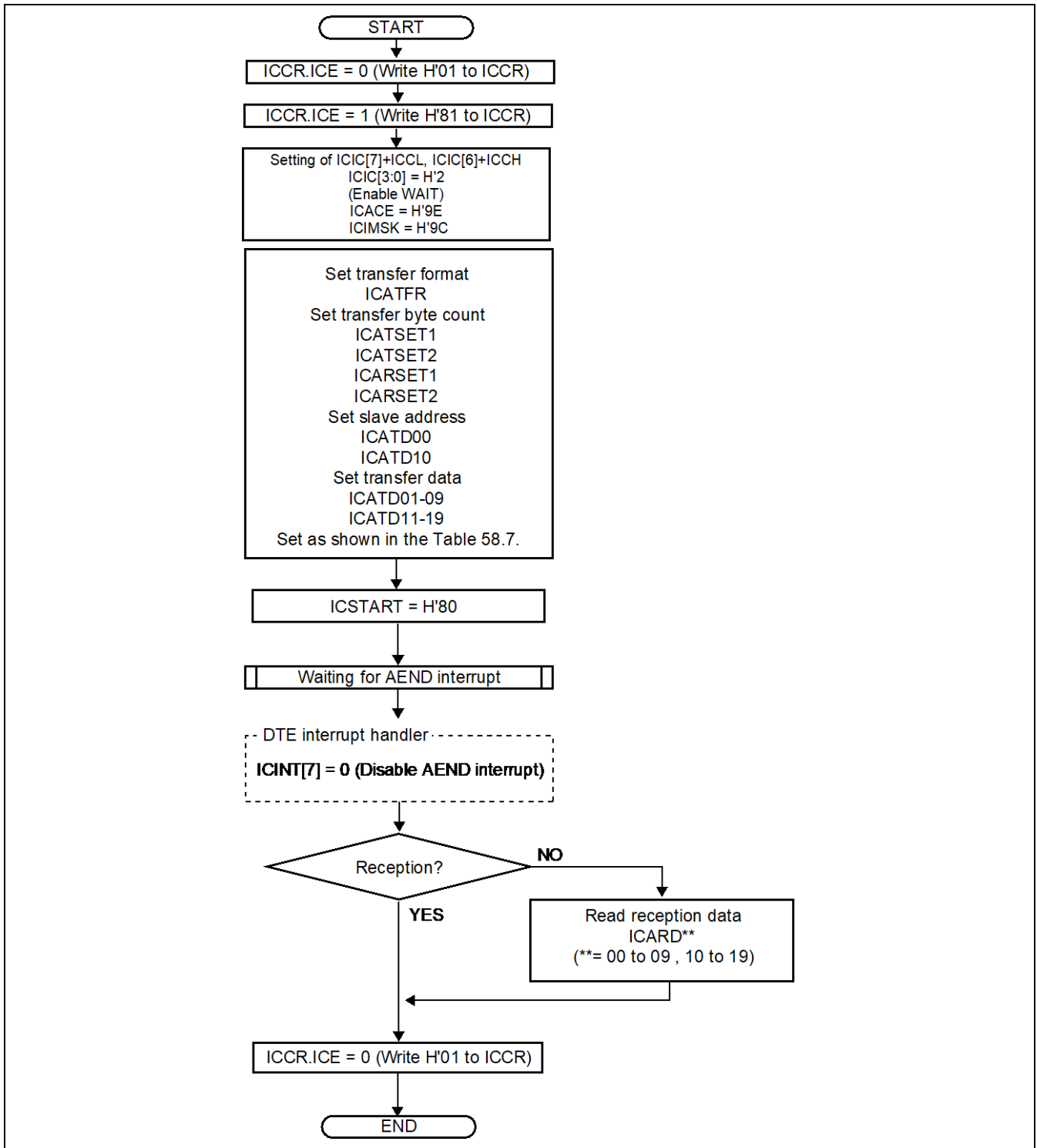


Figure 53.13 Example Flow for n-Byte Write/n-Byte Read (c) The Direction is Converted From Transmission to Reception (Using WAIT and DMA Function)

Table 53.7 Register setting list according to Format (without DMA)

| Format | 0000 | 0001 | 0010 | 0011 | 0100 | 0101 |
|----------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|
| ICATSET1 | value | H'00 | value | value | H'00 | H'00 |
| ICATSET2 | — | — | value | H'00 | value | H'00 |
| ICARSET1 | — | value - 1 | — | value - 1 | H'00 | value - 1 |
| ICARSET2 | — | — | — | — | — | value - 1 |
| ICATD00 | slave address + (W) | slave address + (R) | slave address + (W) | slave address + (W) | slave address + (R) | slave address + (R) |
| ICATD01 | data | — | data | data | — | — |
| : | | | | | | |
| ICATD09 | | | | | | |
| ICATD10 | — | — | slave address + (W) | slave address + (R) | slave address + (W) | slave address + (R) |
| ICATD11 | — | — | data | — | data | — |
| : | | | | | | |
| ICATD19 | | | | | | |

value: Data transfer count (excluding slave address)

—: There is no need to set it.

(W): write bit (R): read bit

53.3.7 Automatic transfer mode by DMA

An automatic transfer mode by DMA can automatically carry out transfer of IIC by appointing ICATFR.Format.

1. Write ICCR.ICE to 0 to reset the IIC module. (Write ICCR = H'01)
2. Set ICCR.ICE to activate the IIC module. (Write H'81 to ICCR)
3. Set ICIC[7] + ICCL, ICIC[6] + ICCH, ICIC[1] = 1, ICACE, and ICIMSK.
4. Set transfer format to ICATFR.Format. And set the number of the transfer and data like Table 53.8.
5. Write H'81 or H'82 or H'83 in the ICSTART to start IIC transfer.
6. When transmitting, ICATD00 is written one by one from the memory space specified by DMAC. When receiving, the memory receives one by one from ICARD00 using DMAC.
6. Wait for a AEND interrupt.
7. Clear ICINT.AEND to disable a AEND interrupt.
8. Clear ICCR.ICE to 0 to reset the IIC module. (Write ICCR = H'01)

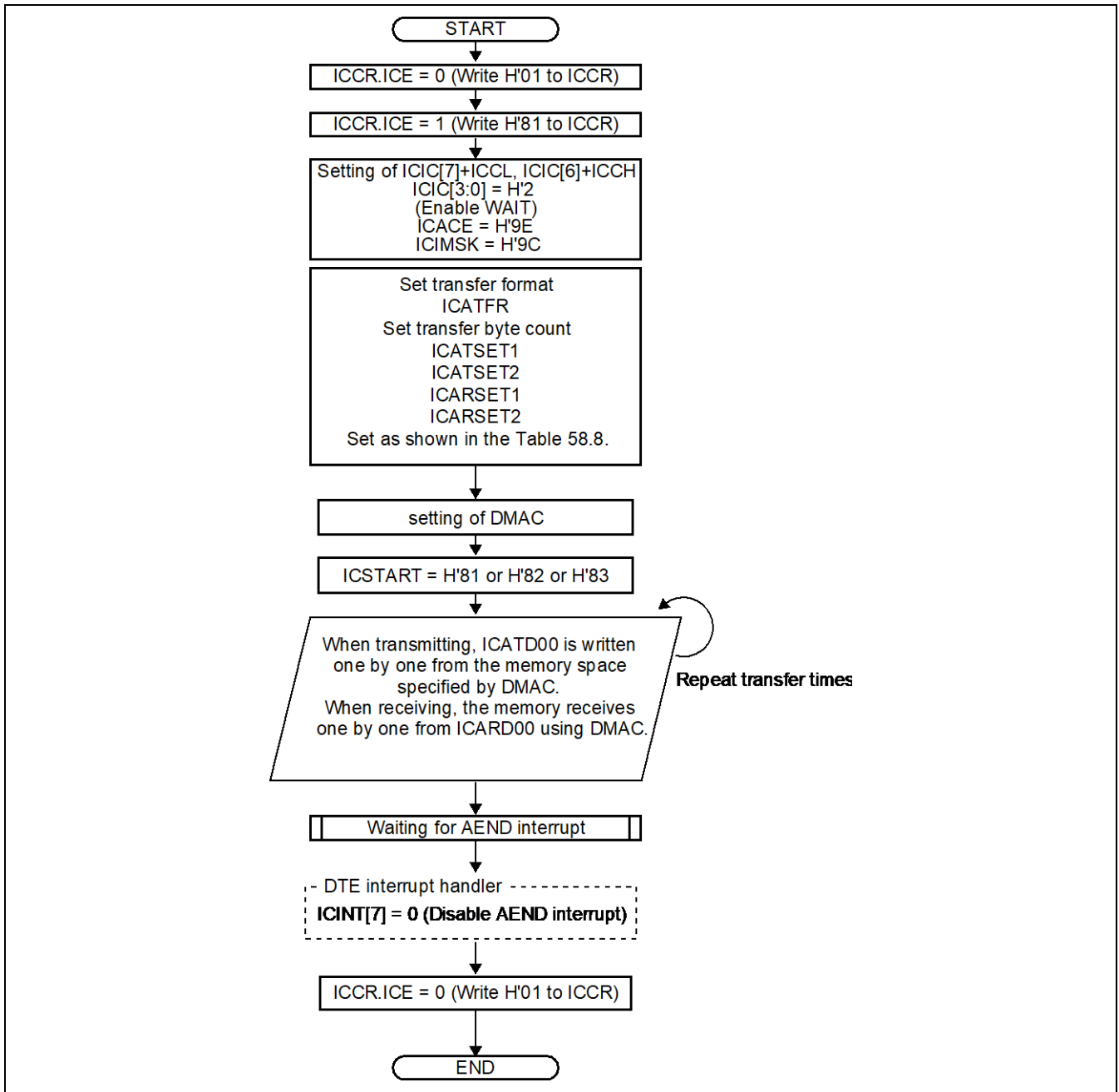


Figure 53.14 Example Flow for n-Byte Write/n-Byte Read (c) The Direction is Converted From Transmission to Reception (Using WAIT and DMA Function)

Table 53.8 Register setting list according to Format (with DMA)

| Format | 0000 | 0001 | 0010 | 0011 | 0100 | 0101 |
|----------|-------------------------------------|-----------|-------|-----------|-------|-----------|
| ICATSET1 | value | H'00 | value | value | H'00 | H'00 |
| ICATSET2 | — | — | value | H'00 | value | H'00 |
| ICARSET1 | — | value - 1 | — | value - 1 | H'00 | value - 1 |
| ICARSET2 | — | — | — | — | — | value - 1 |
| ICATD00 | The data are transferred using DMA. | | | | | |
| ICATD10 | | | | | | |
| ICARD00 | | | | | | |
| ICARD10 | | | | | | |

value: Data transfer count (excluding slave address)

—: There is no need to set it.

53.3.8 Synchronizing SCL Line

When the SCL line state is changed from high to low, connected devices may start counting in the low period. When the clock of a device becomes low, the device holds the SCL line low until its clock becomes high (see Figure 53.15). However, even if the clock of the device is changed from low to high, the SCL line state does not change when clocks of other devices are in the low period. Therefore, the low period of the SCL line is determined by the device with the longest low period. At this time, a device with the short low period waits in the high state.

When all devices complete the low period, the clock line is open and in the high state. Therefore, the device clock and SCL line are in the same state, and start counting both of pulses in high level period. The SCL line enters the low state again by the device that has first completed the high period.

The low period is determined by the device with the longest low period, and the high period is determined by the device with the shortest high period. The SCL line is then synchronized.

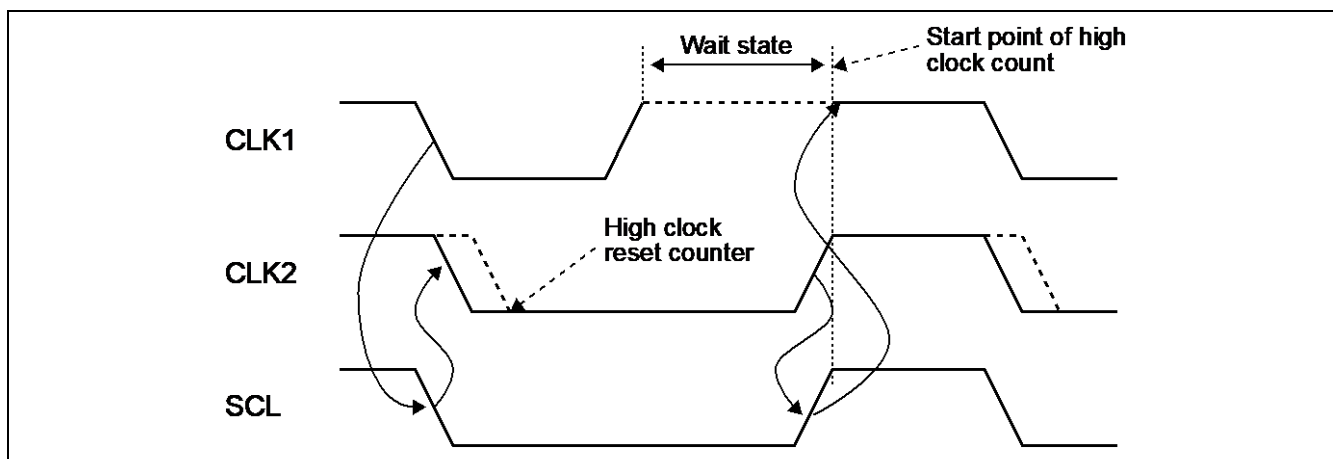


Figure 53.15 Synchronizing SCL Line

53.3.9 Noise Canceller

For removing spike noise on the SCL and SDA line, the noise canceller circuit is implemented. Figure 53.16 shows a block diagram of the noise canceller circuit.

The noise canceller consists of two cascaded FFs and a match detector. The SCL/SDA signal is sampled by FFs with module clock, and recognize that SCL/SDA signals are toggled if both of two FF's outputs have toggled. If only one output of FFs have toggled, previously latched value is output to the internal logic.

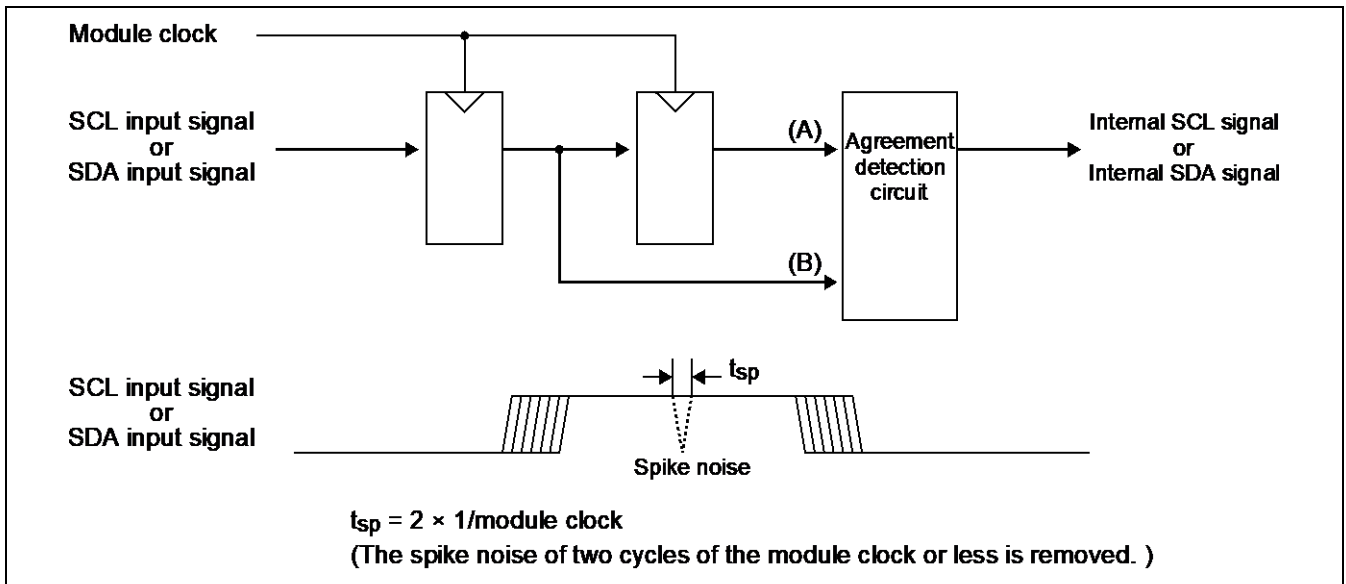


Figure 53.16 Block Diagram of Noise Canceller

53.3.10 Arbitration Lost Operation

This module can assert an arbitration lost interrupt request when bus conflict is detected.

Interrupt assert conditions are described below.

- (1) When the level of the data being received changes during counting of the period at high level for the SCL clock.
- (2) When the level on the SCL line becomes low, while the SCL clock is controlled by this module, and not driven by this module.
- (3) When this module detects the START condition before issuing the START condition.
(This AL case does not detect it in Automatic Mode.)
- (4) When the SDA data driven by this module and monitored SDA signals are not equal.

In case of conditions above, the AL bit in ICSR is set to 1 and an arbitration lost interrupt is asserted.

When a loss of arbitration is detected in transmission mode, the SDA line is released right now. For SCL clock line, clocks kept be output until the end of the frame including acknowledge cycle and released. Figure 53.17 shows an example of arbitration lost interrupt operation timing.

In case of conditions below, the AL bit in ICSR is set to 0.

- When data is written to ICDR (in transmission mode), or data is read (in receive mode) while DTE = 1.
- When 0 is written to the AL bit in ICSR.

When arbitration is lost by the acknowledge bit, hold SCL and SDA at a low level.

ICE reset must be performed by the software to release the bus.

Do not clear the ALE bit to 0 while the AL bit is 1. It is recommended to set ALE bit only before the start of communication.

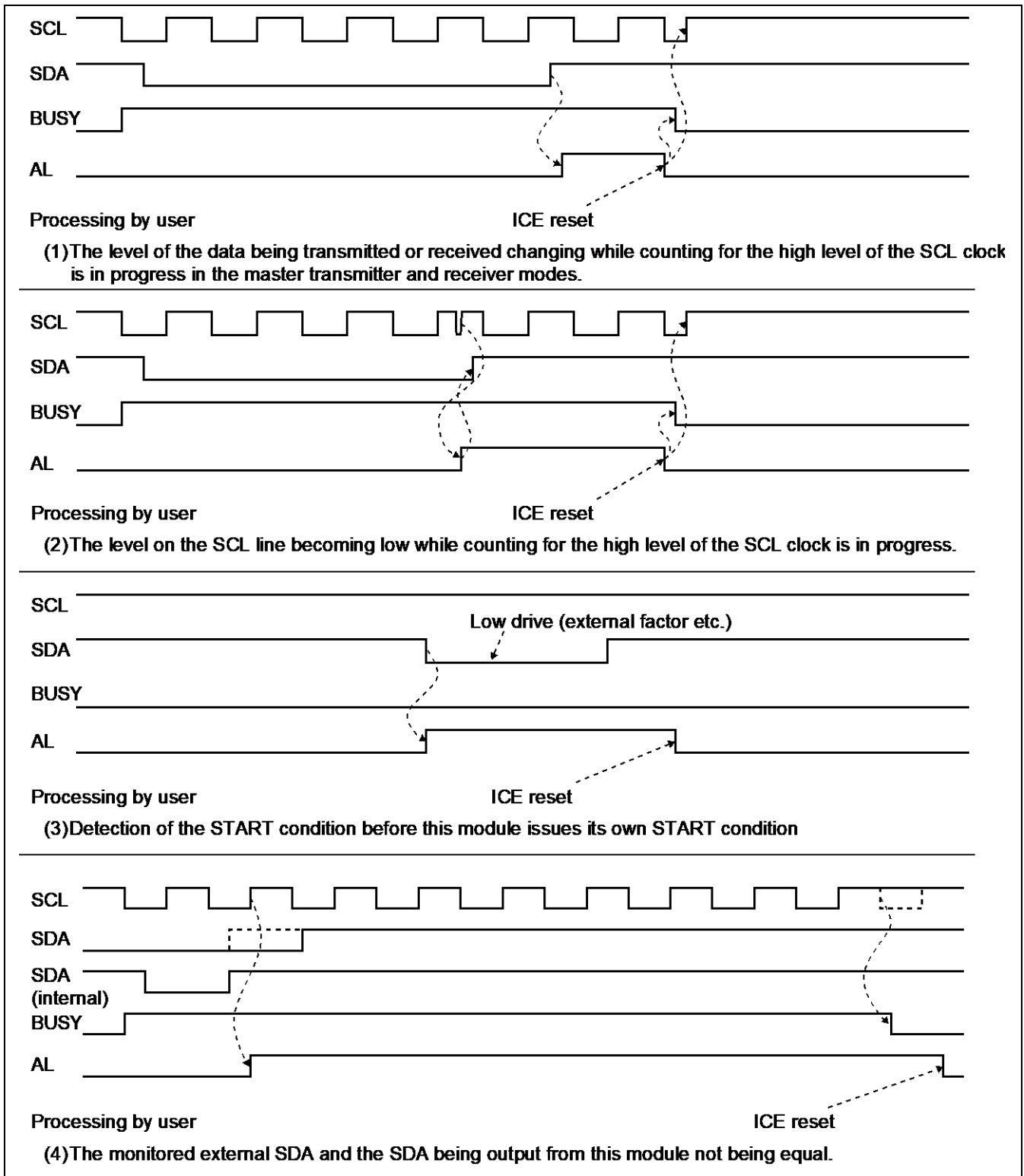


Figure 53.17 Example of Arbitration Lost Interrupt Operation Timing

53.3.11 Non-Acknowledge Operation

This module can assert a non-acknowledge interrupt when there is no acknowledge from the receiving device in transmission.

An acknowledge bit from the receiving device is stored in the TACK bit in ICSR. A non-acknowledge interrupt is asserted if the TACK bit is set to 1 when TACKE in ICIC is 1.

Generating a STOP condition after a non-acknowledge interrupt is recommended. The STOP condition should be generated even if the ICCR is already set for further processing (for example, generating the repeated START condition). Figure 53.19 shows an example processing of generating the STOP condition when non-acknowledge interrupt is asserted.

Do not set the TACKE bit in ICIC if processing for non-acknowledgement is not required.

Do not clear TACKE bit to 0 while TACK bit is 1. It is recommended to set TACKE bit only before start of communication.

Figure 53.18 shows an example of non-acknowledge operation timing.

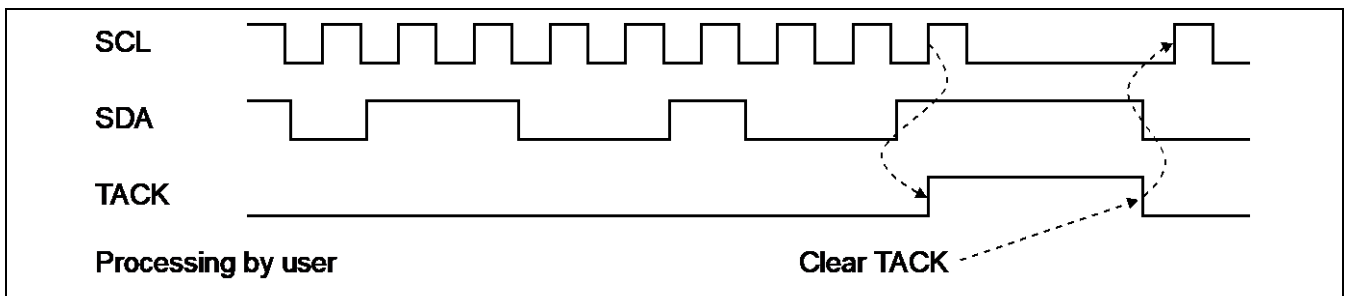


Figure 53.18 Example of Non-Acknowledge Operation Timing

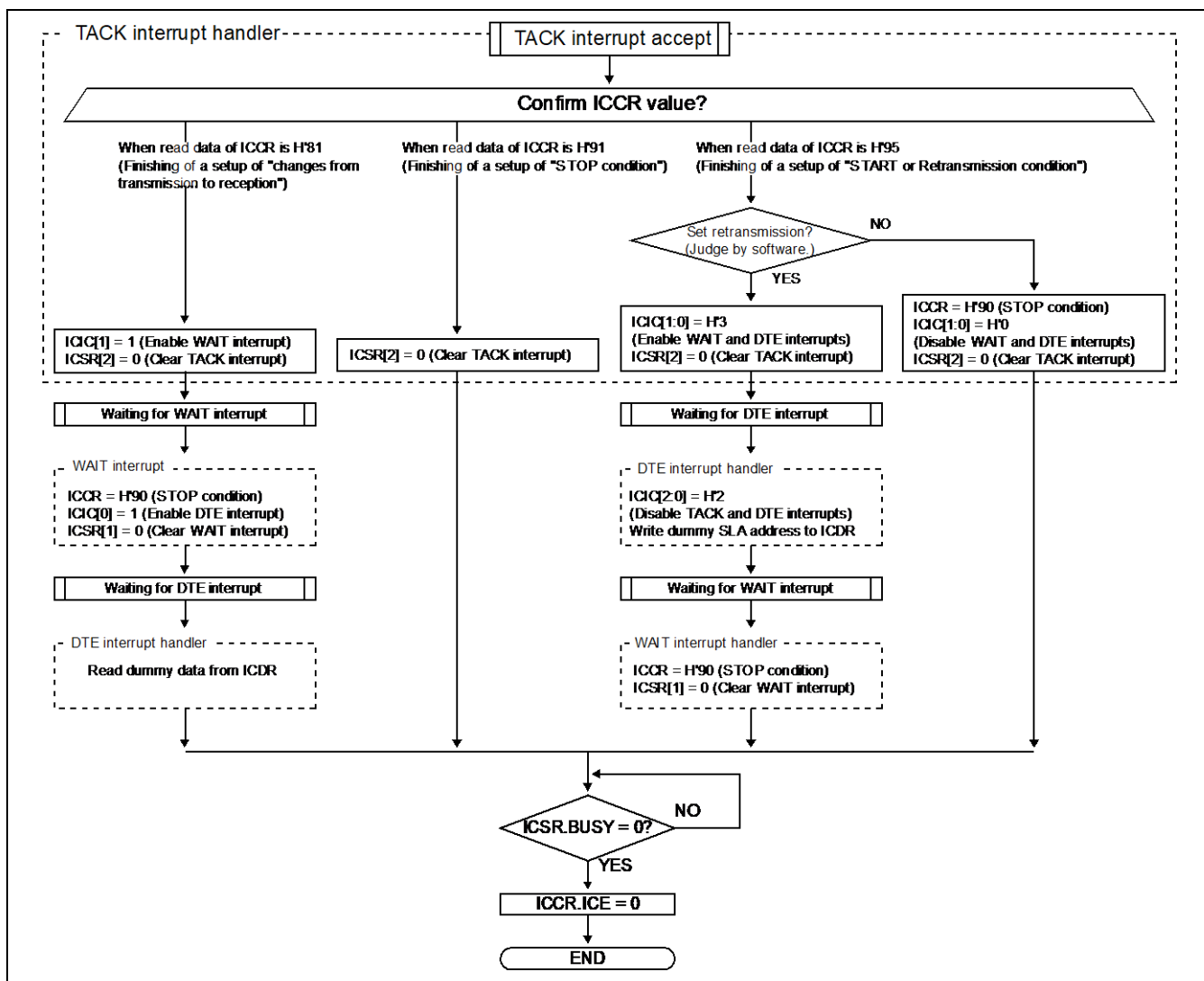


Figure 53.19 Generating a STOP Condition When Non-Acknowledge Interrupt Asserted

53.3.12 Wait Operation

This module can go into the wait state by assert of WAIT bit in ICSR. WAIT is set to 1 at the falling edge of the eighth transmit clock while this module is operating in master mode.

After that, the transmit clock is fixed to low until the WAIT bit is cleared to 0.

When the WAIT bit is cleared to 0, the ninth transmit clock is generated and the state is recovered from the wait state.

In case of using WAIT interrupt, data transfer by the I2C bus is suspended from the time WAIT is asserted by the hardware until the time WAIT is cleared by software.

It is recommended to set the WAITE bit only before the start of communication.

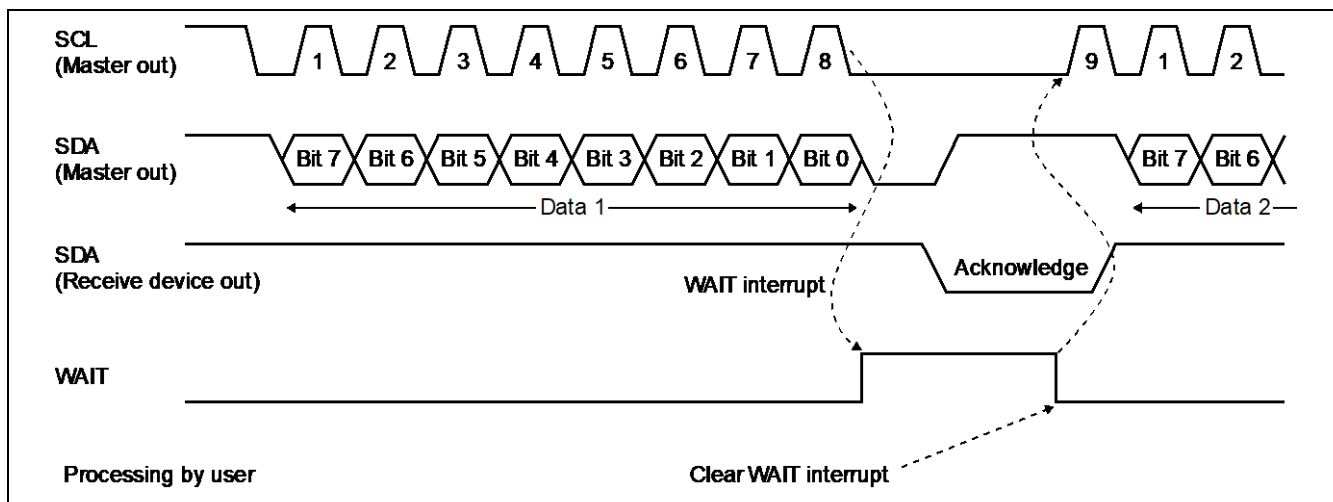


Figure 53.20 Example of Wait Operation Timing

53.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

(1) Arbitration lost

When a loss of arbitration is detected, the I2C bus may not be released soon depending on the status of slave devices. Reset the IIC module by the ICE bit in ICCR register when re-starting data transfer after a loss of arbitration is detected.

(2) ICE reset

When asserting ICE reset during data transfer, SCL clock is stopped immediately and the bus may not be released depending on the status of the slave device.

(3) Processing for stop condition generation

When data is transmitted or received after a STOP condition is generated, it may not be possible to generate a STOP condition hereafter. Initialize all registers in the IIC module by clearing the ICE bit in ICCR to 0, after a STOP condition has been generated or before making settings to trigger data transmission or reception after a STOP condition has been generated.

(4) Handling of IIC when not in use

Set the ICE bit in ICCR to 0 to disable the unused channels while the IIC bus interface is not used.

(5) Abort of data transfer in the middle of IIC communication

Aborting data transfer may violate the I2C protocol. Make sure that the external device accepts such violations when you need to abort transfer in progress.

54. Clock-Synchronized Serial Interface with FIFO (MSIOF)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

54.1 Overview

This LSI includes clock-synchronized serial I/O module with FIFO (MSIOF). In this module, MSO ϕ is used as a module clock.

54.1.1 Features

- Number of channels: Four channels
- FIFO capacity: 32 bits \times 64 stages for transmission and 32 bits \times 256 stages for reception
 - For reception FIFO, receiving 1 FRAME with 2 group with 64 words requires at most 128 FIFO stages.
- MSB first or LSB first selectable for data transmission and reception
- Synchronization by level [RZ/G2H]
- Synchronization by frame synchronization pulse, level, or left/right channel switch [Except for RZ/G2H]
- Supports master and slave modes
- Interrupts: One interrupt line, with 12 internal factors per module.
- Serial clock
- DMA transfer
- Serial format
 - Supports serial formats: SPI (master and slave modes). [RZ/G2H].
 - Supports serial formats: IIS, SPI (master and slave modes). [Except for RZ/G2H]
- CLK/SYNC (SPI) in common transmit/receive mode

54.1.2 Block diagram

Figure 54.1 shows a block diagram of the MSIOF.

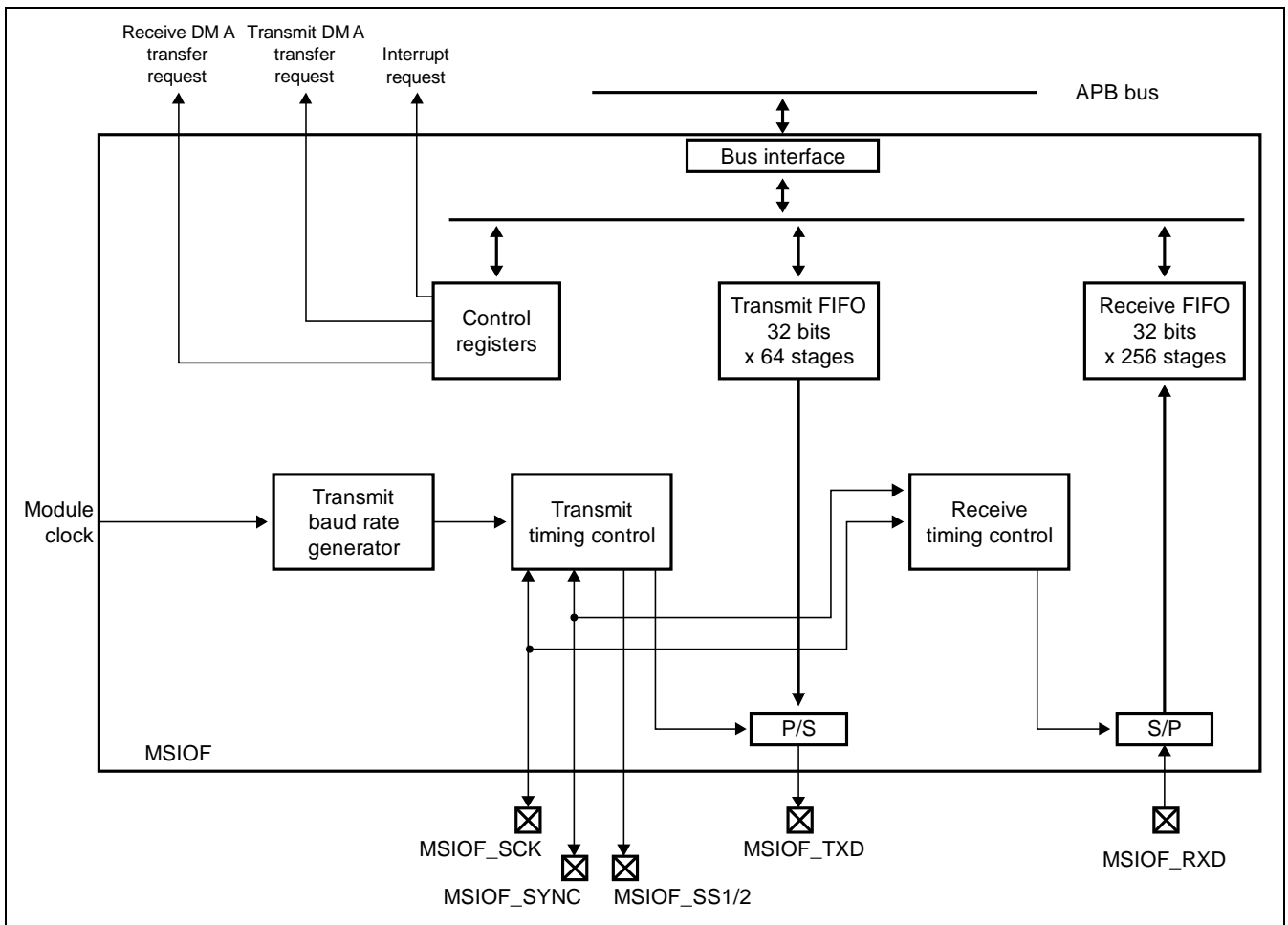


Figure 54.1 Block Diagram of MSIOF

54.1.3 External Pins

The pin configuration of this module is shown in Table 54.1 and Table 54.2.

Table 54.1 Pin Configuration

| Name | Abbreviation | I/O | Function | Second Generation RZ/G Series Products | | | | |
|------------|--------------------|--------|---|---|----------------------------|--------|--------|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| MSIOF_SCK | SCK | I/O | MSIOF serial clock for transmission | √ | √ | √ | √ | |
| MSIOF_SYNC | SYNC | I/O | MSIOF frame synchronization signal for transmission | √ | √ | √ | √ | |
| MSIOF_SS1 | SS1 | Output | MSIOF frame synchronization signal 1 for transmission This signal can work only master mode. | √ | √ | √ | √ | |
| MSIOF_SS2 | SS2 | Output | MSIOF frame synchronization signal 2 for transmission This signal can work only master mode. | √ | √ | √ | √ | |
| MSIOF_TXD | TXD (MOSI/MISO) | Output | MSIOF transmit data | √ | √ | √ | √ | |
| MSIOF_RXD | RXD (MISO/MOSI) | Input | MSIOF receive data | √ | √ | √ | √ | |

Table 54.2 Pin Group

| Module Name | MOD_SEL setting (refer to section PFC for each products) | Group Suffix | Pin Name XXX is Abbreviation of Table 54.1 | Second Generation RZ/G Series Products | | | |
|-------------|---|--------------|---|---|----------------------------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| MSIOF0 | — | — | MSIOF0_XXX | √ | √ | √ | √ |
| MSIOF1 | — | — | MSIOF1_XXX | — | — | — | √ |
| | {sel_msiof1_n} = H'0 | _A | MSIOF1_XXX_A | √ | √ | √ | — |
| | {sel_msiof1_n} = H'1 | _B | MSIOF1_XXX_B | √ | √ | √ | — |
| | {sel_msiof1_n} = H'2 | _C | MSIOF1_XXX_C | √ | √ | √ | — |
| | {sel_msiof1_n} = H'3 | _D | MSIOF1_XXX_D | √ | √ | √ | — |
| | {sel_msiof1_n} = H'4 | _E | MSIOF1_XXX_E | √ | √ | √ | — |
| | {sel_msiof1_n} = H'5 | _F | MSIOF1_XXX_F | √ | √ | √ | — |
| | {sel_msiof1_n} = H'6 | _G | MSIOF1_XXX_G | √ | √ | √ | — |
| MSIOF2 | — | — | MSIOF2_XXX | — | — | — | — |
| | {sel_msiof2_m} = H'0 | _A | MSIOF2_XXX_A | √ | √ | √ | √ |
| | {sel_msiof2_m} = H'1 | _B | MSIOF2_XXX_B | √ | √ | √ | √ |
| | {sel_msiof2_m} = H'2 | _C | MSIOF2_XXX_C | √ | √ | √ | — |
| | {sel_msiof2_m} = H'3 | _D | MSIOF2_XXX_D | √ | √ | √ | — |
| MSIOF3 | — | — | MSIOF3_XXX | — | — | — | — |
| | {sel_msiof3_l} = H'0 | _A | MSIOF3_XXX_A | √ | √ | √ | √ |
| | {sel_msiof3_l} = H'1 | _B | MSIOF3_XXX_B | √ | √ | √ | √*1 |
| | {sel_msiof3_l} = H'2 | _C | MSIOF3_XXX_C (not exist SS1 and SS2) | √ | √ | √ | — |
| | {sel_msiof3_l} = H'3 | _D | MSIOF3_XXX_D (not exist SS2) | √ | √ | √ | — |
| | {sel_msiof3_l} = H'4 | _E | MSIOF3_XXX_E | √ | √ | √ | — |

n = 2 to 0, m = 1 to 0, l = 2 to 0

Notes: 1. RZ/G2E does not exist SS2 at this MOD_SEL setting.

54.1.4 Register Configuration

Table 54.3 and Table 54.4 show the MSIOF register configuration and Table 54.5 shows the register state in each processing mode. Do not write to any address other than those listed in the tables. Otherwise, the operation is not guaranteed. Reading a non-listed address returns an undefined value. Access the registers should be accessed with the “Access Size” listed below. Otherwise, the operation is not guaranteed.

Table 54.3 Module Base Address Summary

| | | Second Generation RZ/G Series Products | | | |
|-------------|--------------|--|----------------------------|--------|--------|
| Module Name | Base Address | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| MSIOF0 | H'E6E9_0000 | √ | √ | √ | √ |
| MSIOF1 | H'E6EA_0000 | √ | √ | √ | √ |
| MSIOF2 | H'E6C0_0000 | √ | √ | √ | √ |
| MSIOF3 | H'E6C1_0000 | √ | √ | √ | √ |

Table 54.4 Register Configuration

| | | | | | Second Generation RZ/G Series Products | | | |
|--------------------------------------|--------------|---------|----------------|--------------|--|----------------------------|--------|--------|
| Name | Abbreviation | R/W | Address Offset | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| MSIOF transmit mode register 1 | SITMDR1 | R/W | H'0000 | 32 | √ | √ | √ | √ |
| MSIOF transmit mode register 2 | SITMDR2 | R/W | H'0004 | 32 | √ | √ | √ | √ |
| MSIOF transmit mode register 3 | SITMDR3 | R/W | H'0008 | 32 | √ | √ | √ | √ |
| MSIOF receive mode register 1 | SIRMDR1 | R/W | H'0010 | 32 | √ | √ | √ | √ |
| MSIOF receive mode register 2 | SIRMDR2 | R/W | H'0014 | 32 | √ | √ | √ | √ |
| MSIOF receive mode register 3 | SIRMDR3 | R/W | H'0018 | 32 | √ | √ | √ | √ |
| MSIOF transmit clock select register | SITSCR | R/W | H'0020 | 16 | √ | √ | √ | √ |
| MSIOF control register | SICTR | R/W | H'0028 | 32 | √ | √ | √ | √ |
| MSIOF FIFO control register | SIFCTR | R/W | H'0030 | 32 | √ | √ | √ | √ |
| MSIOF status register | SISTR | R/WC1*1 | H'0040 | 32 | √ | √ | √ | √ |
| MSIOF interrupt enable register | SIIER | R/W | H'0044 | 32 | √ | √ | √ | √ |
| MSIOF transmit FIFO data register | SITFDR | W | H'0050 | 32, 16, 8 | √ | √ | √ | √ |
| MSIOF receive FIFO data register | SIRFDR | R | H'0060 | 32, 16, 8 | √ | √ | √ | √ |

Notes: 1. For this register, bits 28 and 12 are read-only bits. For details, refer to Section 54.2.11 MSIOF Status Register (SISTR)

Table 54.5 Register States in Each Operating Mode

| Abbreviation | Reset | Module Standby |
|--------------|-------------|----------------|
| SITMDR1 | Initialized | Retained |
| SITMDR2 | Initialized | Retained |
| SITMDR3 | Initialized | Retained |
| SIRMDR1 | Initialized | Retained |
| SIRMDR2 | Initialized | Retained |
| SIRMDR3 | Initialized | Retained |
| SITSCR | Initialized | Retained |
| SICTR | Initialized | Retained |
| SIFCTR | Initialized | Retained |
| SISTR | Initialized | Retained |
| SIIER | Initialized | Retained |
| SITFDR | Initialized | Retained |
| SIRFDR | Retained | Retained |

54.1.5 Connected Module

Table 54.6 shows the connected modules to the MSIOF.

Table 54.6 Connected modules

| Module name | Connected module name | Function of connected module |
|-------------|-----------------------|------------------------------|
| MSIOF | AP-System Core | Access the Register |
| | CPG | Output Clocks |
| | PFC | Select External pins |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |
| | INTC-AP | Control to interrupt |
| | SYS-DMAC | Control Direct Memory Access |

54.2 Register Description

[Legend for Register Descriptions]

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC1: Readable/writable. Writing 1 initializes the bit, and writing 0 is ignored.

R: Read-only. The write value should always be initial value.

—/WB: Write-only. The read value is undefined.

54.2.1 MSIOF Transmit Mode Register 1 (SITMDR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SITMDR1 is a 32-bit readable/writable register that specifies the MSIOF transmit mode.

The settings of this register should be not changed when SICTR.RXE or SICTR.TXE are B'1. Otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|------|------|-------------|-------------|------------|--------|-----|-----|----|-----------|-----|-----|-----|----------|-----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TRMD | PCON | SYNCMD[1:0] | SYNCCH[1:0] | SYNCA C | BITLSB | — | — | — | DTDL[2:0] | — | — | — | — | — | — |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | FLD[1:0] | — | TXSTP |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|--------|-------------|---------------|-----|--|
| 31 | TRMD | B'1 | R/W | Transfer Mode Selects the transfer mode. 0: Slave mode 1: Master mode |
| 30 | PCON | B'0 | R/W | Transfer Signal Connection 0: Setting prohibited 1: MSIOF_SCK and MSIOF_SYNC are used as common signals for transmission and reception (MSIOF_SCK and MSIOF_SYNC). |
| 29, 28 | SYNCMD[1:0] | B'00 | R/W | SYNC Mode These bits specify the mode for the MSIOF_SYNC signal. [RZ/G2H] B'10: Level mode/SPI Other than above: Setting prohibited [Except for RZ/G2H] B'00: Frame start synchronization pulse B'01: Reserved B'10: Level mode/SPI B'11: L/R mode |
| 27, 26 | SYNCCH[1:0] | B'00 | R/W | Synchronization Signal Channel Select These bits are valid only in master mode. B'00: The frame synchronization signal output at MSIOF_SYNC. B'01: The frame synchronization signal output at MSIOF_SS1. B'10: The frame synchronization signal output at MSIOF_SS2. B'11: Setting prohibited |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|--------------|---------------|-----|--|
| 25 | SYNCAC | B'0 | R/W | MSIOF_SYNC Polarity 0: Active-high signal in synchronization pulse or level mode, or driven high then low in L/R mode 1: Active-low signal in synchronization pulse or level mode, or driven low then high in L/R mode |
| 24 | BITLSB | B'0 | R/W | MSB/LSB First 0: MSB first 1: LSB first |
| 23 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 22 to 20 | DTDLE[2:0] | B'001 | R/W | Data Pin Bit Delay for MSIOF_SYNC Pin The value specified with TXDIZ in SICTR is output during transmission. B'1xx is valid only in SPI mode. In SPI mode, these bits should be specified so that the sum of the delays specified through DTDLE and SYNCDEL becomes an integer value. [RZ/G2H] B'010: 2-clock-cycle delay Other than above: Setting prohibited [Except for RZ/G2H] B'000: No bit delay B'001: 1-clock-cycle delay B'010: 2-clock-cycle delay B'101: 0.5-clock-cycle delay B'110: 1.5-clock-cycle delay Other than above: Setting prohibited |
| 19 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 18 to 16 | SYNCDEL[2:0] | B'000 | R/W | Frame Synchronization Signal Timing Delay These bits extend the transmit frame synchronization signal. The value specified with TXDIZ in SICTR is output during transmission. These bits are invalid when SYNCMD [1:0] = B'01. B'1xx is valid only in SPI mode. In SPI mode, these bits should be specified so that the sum of the delays specified through DTDLE and SYNCDEL becomes an integer value. In L/R mode, these bits should always be set to B'000. B'000: No bit delay B'001: 1-clock-cycle delay B'010: 2-clock-cycle delay B'011: 3-clock-cycle delay B'101: 0.5-clock-cycle delay B'110: 1.5-clock-cycle delay Other than above: Setting prohibited |
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|------|-----------|---------------|-----|---|
| 3, 2 | FLD [1:0] | B'00 | R/W | <p>Frame Synchronization Signal Interval</p> <p>Specify the minimum idle time between frames in the number of serial clock cycles. This setting is valid only in master mode.</p> <p>In L/R mode, these bits should always be set to B'00.</p> <p>In SPI mode, these bits shall not be set to B'00.</p> <p>B'00: 0-clock-cycle delay B'01: 1-clock-cycle delay B'10: 2-clock-cycle delay B'11: 3-clock-cycle delay</p> |
| 1 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 0 | TXSTP | B'1 | R/W | <p>Transmission Stop</p> <p>0: Setting prohibited</p> <p>1: Stop a frame from starting to transmit until either the number of FIFO stages equals the frame size or the FIFO is full and simultaneously a frame started cannot be interrupted in between.</p> |

54.2.2 MSIOF Transmit Mode Register 2 (SITMDR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SITMDR2 is a 32-bit readable/writable register that specifies the MSIOF transmit mode.

The settings of this register should be not changed when SICTR.RXE or SICTR.TXE are B'1. Otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|-----|----|--------------|-----|-----|-----|-----|----|-------------|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | GRP | — | BITLEN1[4:0] | | | | — | — | WDLEN1[5:0] | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

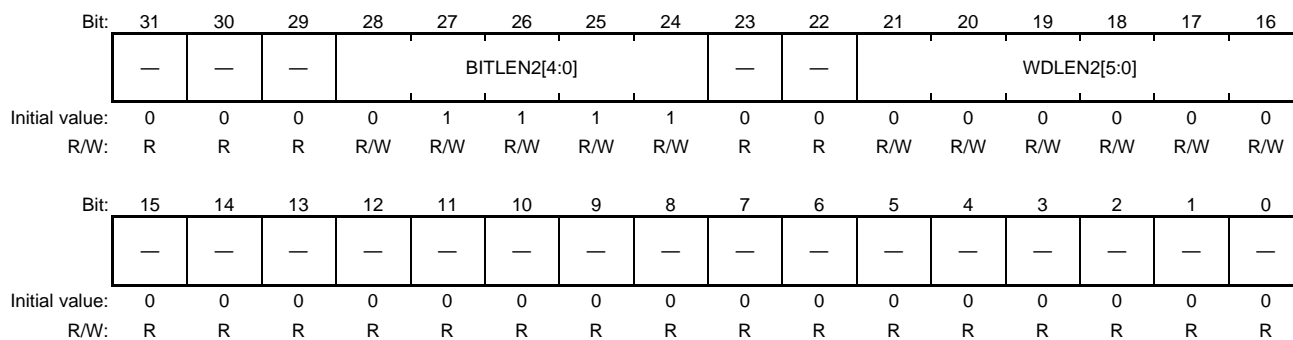
| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|--------------|---------------|-----|--|
| 31 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 30 | GRP | B'0 | R/W | Group Count 0: Group count 1 1: Group count 2 |
| 29 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 28 to 24 | BITLEN1[4:0] | B'0_1111 | R/W | Data Size (8 to 32 bits) The word size (bits) of Group 1 is set to the value specified in these bits + 1. Either 8, 16, 24, or 32 bits can be specified as word size. |
| 23, 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 to 16 | WDLEN1[5:0] | H'00 | R/W | Word Count (1 to 64 words) The word count of Group 1 is set to WDLEN1 + 1. |
| 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

54.2.3 MSIOF Transmit Mode Register 3 (SITMDR3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SITMDR3 is a 32-bit readable/writable register that specifies the MSIOF transmit mode.

The settings of this register should be not changed when SICTR.RXE or SICTR.TXE are B'1. Otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|--------------|---------------|-----|---|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 24 | BITLEN2[4:0] | B'0_1111 | R/W | Word Size (8 to 32 bits) The word size (bits) of Group 2 is set to the value specified in these bits + 1. Either 8, 16, 24, or 32 bits can be specified as word size. When setting SITMDR1.BITLSB = B'1 and SITMDR2.GRP = 1, set the same value as SITMDR2.BITLEN1. |
| 23, 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 to 16 | WDLEN2[5:0] | H'00 | R/W | Word Count (1 to 64 words) The word count of Group 2 is set to WDLEN2 + 1. |
| 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

54.2.4 MSIOF Receive Mode Register 1 (SIRMDR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SIRMDR1 is a 32-bit readable/writable register that specifies the MSIOF receive mode. The settings of this register should be not changed when SICTR.RXE or SICTR.TXE are B'1. Otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|------|----|-------------|-----|----|------------|--------|-----|-----------|-----|-----|-----|-------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TRMD | — | SYNCMD[1:0] | — | — | SYNCA C | BITLSB | — | DTDL[2:0] | | | — | SYNCDL[2:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R/W | R/W | R | R | R/W | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

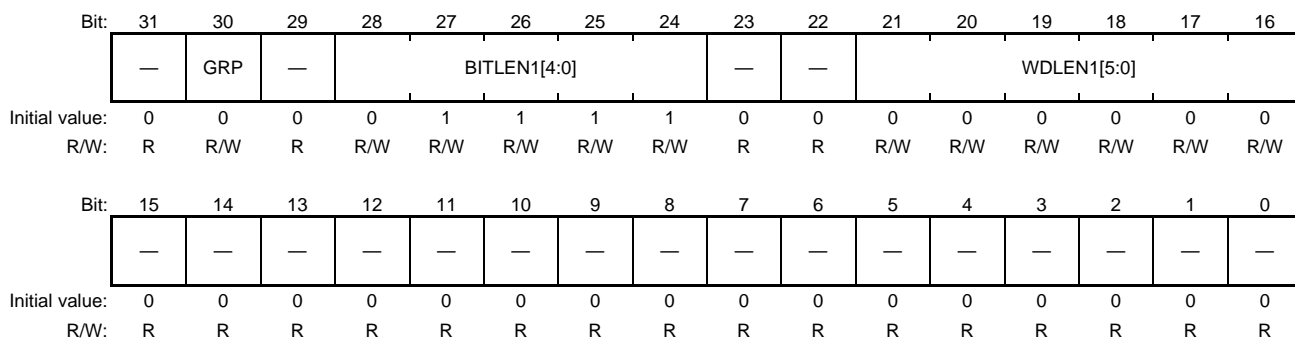
| Bit | Bit Name | Initial Value | R/W | Descriptions |
|--------|--------------|---------------|-----|---|
| 31 | TRMD | B'0 | R/W | Transfer Mode Selects the transfer mode. This bit should be set to 0. |
| 30 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 29, 28 | SYNCMD [1:0] | B'00 | R/W | SYNC Mode The mode setting in these bits should be the same as that in SITMDR1.SYNCMD. [RZ/G2H] B'10: Level mode/SPI Other than above: Setting prohibited [except for RZ/G2H] B'00: Frame start synchronization pulse B'01: Reserved B'10: Level mode/SPI B'11: L/R mode |
| 27, 26 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25 | SYNCA C | B'0 | R/W | SYNC Polarity B'0: Active-high signal in synchronization pulse or level mode, or driven high then low in L/R mode B'1: Active-low signal in synchronization pulse or level mode, or driven low then high in L/R mode This bit must have the same setting as the SYNCA C bit of SITMDR1. |
| 24 | BITLSB | B'0 | R/W | MSB/LSB First B'0: MSB first B'1: LSB first |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|-----------------|---------------|-----|--|
| 23 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 22 to 20 | DTDL[2:0] | B'001 | R/W | Data Pin Bit Delay for MSIOF_SYNC Pin B'000: No bit delay B'001: 1-clock-cycle delay B'010: 2-clock-cycle delay Other than above: Setting prohibited In case of SPI mode, only B'000 is allowed to set to this field. |
| 19 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 18 to 16 | SYNCDL [2:0] | B'000 | R/W | MSIOF_SYNC Timing Delay These bits should always be set to B'000. |
| 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

54.2.5 MSIOF Receive Mode Register 2 (SIRMDR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SIRMDR2 is a 32-bit readable/writable register that specifies the MSIOF receive mode. The settings of this register should be not changed when SICTR.RXE or SICTR.TXE are B'1. Otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|--------------|---------------|-----|--|
| 31 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 30 | GRP | B'0 | R/W | Group Count 0: Group count 1 1: Group count 2 When using reception in master mode, set the same value as the value specified by the GRP bit in SITMDR2. |
| 29 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 28 to 24 | BITLEN1[4:0] | B'0_1111 | R/W | Word Size (8 to 32 bits) The word size (bits) of Group 1 is set to the value specified in these bits + 1. Either 8, 16, 24, or 32 bits can be specified as word size. When using reception in master mode, set the same value as the value specified by the BITLEN1 bits in SITMDR2. |
| 23, 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 to 16 | WDLEN1[5:0] | H'00 | R/W | Word Count (1 to 64 words) The word count of Group 1 is set to WDLEN1 + 1. When using reception in master mode, set the same value as the value specified by the WDLEN1 bits in SITMDR2. |
| 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

54.2.6 MSIOF Receive Mode Register 3 (SIRMDR3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SIRMDR3 is a 32-bit readable/writable register that specifies the MSIOF receive mode.

The settings of this register should be not changed when SICTR.RXE or SICTR.TXE are B'1. Otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|--------------|-----|-----|-----|-----|----|-------------|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | BITLEN2[4:0] | | | | — | — | WDLEN2[5:0] | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|--------------|---------------|-----|--|
| 31 to 29 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 28 to 24 | BITLEN2[4:0] | B'0_1111 | R/W | Word Size (8 to 32 bits) The word size (bits) of Group 2 is set to the value specified in these bits + 1. Either 8, 16, 24, or 32 bits can be specified as word size. When using reception in master mode, set the same value as the value specified by the BITLEN2 bits in SITMDR3. |
| 23, 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 to 16 | WDLEN2[5:0] | H'00 | R/W | Word Count (1 to 64 words) The word count of Group 2 is set to WDLEN2 + 1. When using reception in master mode, set the same value as the value specified by the WDLEN2 bits in SITMDR3. |
| 15 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

54.2.7 MSIOF Control Register (SICTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SICTR is a 32-bit readable/writable register that specifies the MSIOF operating state.

The values written to the TXE, RXE, TFSE, RFSE, TSCKE, and RSCKE bits become valid (can be read from the bits) several cycles after writing. These bits should not be modified at the same time. Modify one bit at a time, and check that the new value can be read from the bit, and then modify another bit.

| | | | | | | | | | | | | | | | | |
|----------------|-------------|------|-------------|------|------|------|-----|-----|------------|-----|----|----|----|----|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TSCKIZ[1:0] | | RSCKIZ[1:0] | | TEDG | REDG | — | — | TXDIZ[1:0] | | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TSCKE | TFSE | RSCKE | RFSE | — | — | TXE | RXE | — | — | — | — | — | — | TXRST | RXRST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|--------|-------------|---------------|-----|--|
| 31, 30 | TSCKIZ[1:0] | B'00 | R/W | Transmit Clock Input/output Polarity Select in SPI Mode When Transmission is Disabled When SPI mode is not used, these bits must always be set to B'00. These bits must always be set B'10 or B'11 when SPI mode. [Master mode] B'00: Outputs MSIOF_SCK when transmission is disabled. B'01: Setting prohibited B'10: Outputs 0. B'11: Outputs 1. [Slave mode] B'00: Inputs MSIOF_SCK when transmission is disabled. B'01: Setting prohibited B'10: Inputs 0 through MSIOF_SCK when transmission is disabled. B'11: Inputs 1 through MSIOF_SCK when transmission is disabled. |
| 29, 28 | RSCKIZ[1:0] | B'00 | R/W | Receive Clock Polarity Select in SPI Mode Set the same value as the value specified by the TSCKIZ bits. Other setting is prohibited. |
| 27 | TEDG | B'0 | R/W | Transmit Timing 0: Outputs transmit data at the rising edge of the clock. 1: Outputs transmit data at the falling edge of the clock. |
| 26 | REDG | B'0 | R/W | Receive Timing 0: Samples receive data at the falling edge of the clock. 1: Samples receive data at the rising edge of the clock. |
| 25, 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|------------|---------------|-----|---|
| 23, 22 | TXDIZ[1:0] | B'00 | R/W | <p>Pin Output When Transmission is Disabled</p> <p>These bits specify the MSIOF_TXD pin output state when transmission is disabled.</p> <p>B'00: Outputs 0. B'01: Outputs 1. B'10: Setting prohibited B'11: Setting prohibited</p> |
| 21 to 16 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 15 | TSCKE | B'0 | R/W | <p>Transmit Serial Clock Output Enable</p> <p>This bit is valid in master mode.</p> <p>When this bit is set to 1, the MSIOF initializes the baud rate generator and initiates the operation. At the same time, the MSIOF outputs the clock generated by the baud rate generator to the MSIOF_SCK pin. After the clock is output, 1 can be read.</p> <p>When transmitting data, set this bit to 1 before setting the TFSE and TXE bits. After data transmission, clear the TFSE and TXE bits and then clear this bit to 0.</p> <p>[Write data] 0: Disables the MSIOF_SCK output. (Outputs the value specified with TSCKIZ.) 1: Enables the MSIOF_SCK output.</p> <p>[Read data] 0: Does not output MSIOF_SCK. (Outputs the value specified with TSCKIZ.) 1: Outputs MSIOF_SCK.</p> |
| 14 | TFSE | B'0 | R/W | <p>Transmit Frame Synchronization Signal Output Enable</p> <p>This bit is valid in master mode.</p> <p>When this bit is set to 1, the MSIOF initializes the frame counter and initiates the operation. After the transmit frame synchronization signal is output, 1 can be read. When 0 is written to this bit, 0 is set after transmission of the frame.</p> <p>[Write data] 0: Disables the MSIOF_SYNC output. (Outputs the value specified with SYNCAC.) 1: Enables the MSIOF_SYNC output.</p> <p>[Read data] 0: Does not output MSIOF_SYNC. (Outputs the value specified with SYNCAC.) 1: Outputs MSIOF_SYNC.</p> |
| 13 | RSCKE | B'0 | R/W | <p>Receive Serial Clock Output Enable</p> <p>The write value should always be 0.</p> |
| 12 | RFSE | B'0 | R/W | <p>Receive Frame Synchronization Signal Output Enable</p> <p>The write value should always be 0.</p> |
| 11, 10 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|--------|----------|---------------|-----|--|
| 9 | TXE | B'0 | R/W | <p>Transmit Enable</p> <p>When this bit is set to 1, the MSIOF starts data transmission from the beginning of the next frame (at the rising edge of the frame synchronization signal). After the valid data is output, 1 can be read.</p> <p>When value 1 set to this bit becomes valid, the MSIOF issues a transmit data transfer request according to the setting of the TFWM bits in SIFCTR. When transmit data is stored in the transmit FIFO, transmission of data from the MSIOF_TXD pin begins. When 0 is written to this bit, 0 is set after transmission of the frame.</p> <p>This bit is initialized upon a transmit reset.</p> <p>[Write data]</p> <p>0: Disables the MSIOF_TXD output. (Outputs the value specified with TXDIZ.)</p> <p>1: Enables the MSIOF_TXD output.</p> <p>[Read data]</p> <p>0: Does not output MSIOF_TXD. (Outputs the value specified with TXDIZ.)</p> <p>1: Outputs MSIOF_TXD.</p> |
| 8 | RXE | B'0 | R/W | <p>Receive Enable</p> <p>When this bit is set to 1, the MSIOF starts data reception from the beginning of the next frame (at the rising edge of the frame synchronization signal).</p> <p>When value 1 set to this bit becomes valid, the MSIOF starts receiving data through the MSIOF_RXD pin. When receive data is stored in the receive FIFO, the MSIOF issues a receive data transfer request according to the setting of the RFWM bits in SIFCTR.</p> <p>This bit is initialized upon a receive reset.</p> <p>[Write data]</p> <p>0: Disables data reception through MSIOF_RXD.</p> <p>1: Enables data reception through MSIOF_RXD.</p> <p>[Read data]</p> <p>0: Data is not received through MSIOF_RXD.</p> <p>1: Data can be received through MSIOF_RXD.</p> |
| 7 to 2 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|---|
| 1 | TXRST | B'0 | R/W | <p>Transmit Reset</p> <p>When value 1 set to this bit becomes valid, the MSIOF immediately sets transmit data through the MSIOF_TXD pin to 0, and initializes the transmit data registers and transmit-related status. When value 1 set to this bit becomes valid when PCON = 1, the RXRST bit is also becomes valid. The following registers and bits are initialized.</p> <p>SITFDR (Transmit FIFO write pointer)</p> <p>TDREQ bit in SISTR</p> <p>TXE bit</p> <p>This bit is read as 1 until the reset operation is completed for about 20 cycles of module clock. No data should be written to SICTR or the transmit FIFO during this time period.</p> <p>[Write data]</p> <p>0: Does not reset transmit operation. 1: Resets transmit operation.</p> <p>[Read data]</p> <p>0: Transmit operation reset is completed. 1: Transmit operation is being reset.</p> |
| 0 | RXRST | B'0 | R/W | <p>Receive Reset</p> <p>When value 1 set to this bit becomes valid, the MSIOF immediately disables reception through the MSIOF_RXD pin, and initializes the receive data registers and receive-related status. The following registers and bits are initialized.</p> <p>SIRFDR (Receive FIFO read pointer)</p> <p>RDREQ bit in SISTR</p> <p>RXE bit</p> <p>This bit is read as 1 until the reset operation is completed for about 20 cycles of module clock. No data should be written to SICTR or read the receive FIFO during this time period.</p> <p>[Write data]</p> <p>0: Does not reset receive operation. 1: Resets receive operation.</p> <p>[Read data]</p> <p>0: Receive operation reset is completed. 1: Receive operation is being reset.</p> |

54.2.8 MSIOF Transmit Clock Select Register (SITSCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SITSCR is a 16-bit readable/writable register that specifies the conditions for generating transmit serial clock in master mode. SITSCR can be specified when the TRMD bit in SITMDR1 is set to B'1.

The settings of this register should be not changed when SICTR.TSCKE is B'1. Otherwise, the operation cannot be guaranteed.

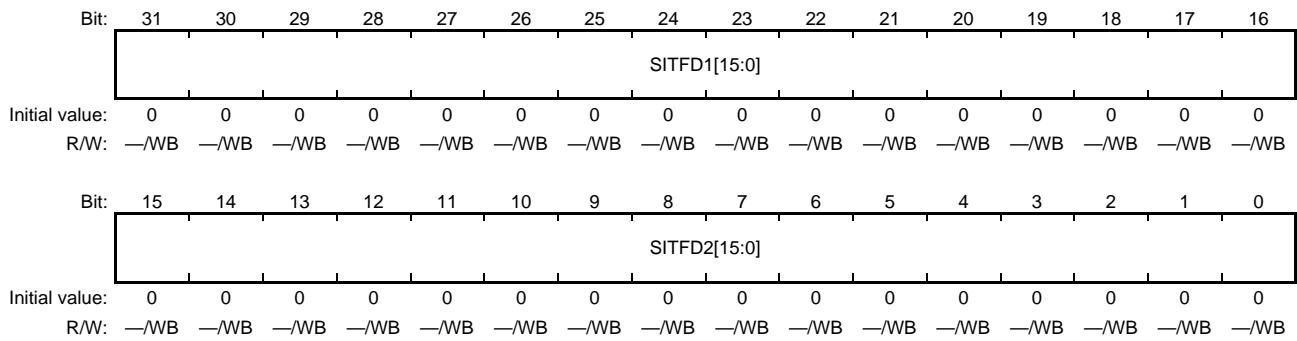
| | | | | | | | | | | | | | | | | |
|----------------|------------|-----|-------|-----------|-----|-----|-----|-----|---|---|---|---|-----------|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MSSEL[1:0] | | MSIMM | BRPS[4:0] | | | | — | — | — | — | — | BRDV[2:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|------------|---------------|-----|---|
| 15, 14 | MSSEL[1:0] | B'00 | R/W | Master Clock Source Select The master clock is the clock input to the baud rate generator. B'00: Selects module clock as the source of master clock. Others: Setting prohibited |
| 13 | MSIMM | B'0 | R/W | Master Clock Direct Select B'0: Selects the clock output from the baud rate generator as the serial clock. B'1: Setting prohibited |
| 12 to 8 | BRPS[4:0] | B'0_0000 | R/W | Prescaler Setting These bits specify the master clock (MSO ϕ) division ratio in the count value of the prescaler in the baud rate generator. The specifiable value is from B'0_0000 ($\times 1/1$) to B'1_1111 ($\times 1/32$). |
| 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 to 0 | BRDV[2:0] | B'000 | R/W | Baud Rate Generator's Division Ratio These bits specify the frequency division ratio for the output stage of the baud rate generator. The final frequency division ratio of the baud rate generator is determined as $BRPS \times BRDV$ (1/1024 max.). B'000: Prescaler output $\times 1/2$ B'001: Prescaler output $\times 1/4$ B'010: Prescaler output $\times 1/8$ B'011: Prescaler output $\times 1/16$ B'100: Prescaler output $\times 1/32$ B'101: Setting prohibited B'110: Setting prohibited B'111: Prescaler output $\times 1/1$ Note: - B'111 is valid only when the BRPS [4:0] bits are set to B'0_0001. - $BRPS \times BRDV = 1/1$ is invalid ($BRPS \times BRDV = 1/2$ min.). |

54.2.9 MSIOF Transmit FIFO Data Register (SITFDR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SITFDR is a 32-bit write-only register that specifies the transmit FIFO data of the MSIOF.

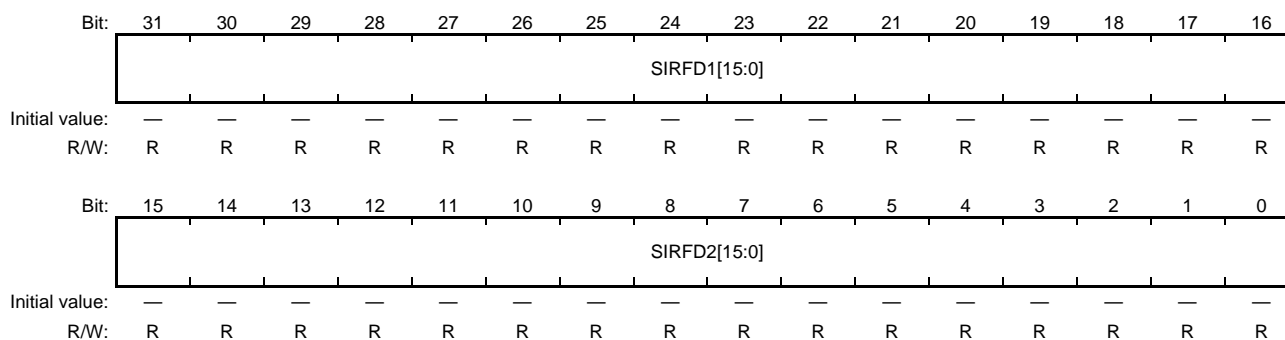


| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|--------------|---------------|------|---|
| 31 to 16 | SITFD1[15:0] | H'0000 | -/WB | These bits specify the upper 16 bits of the FIFO data to be output through MSIOF_TXD. |
| 15 to 0 | SITFD2[15:0] | H'0000 | -/WB | These bits specify the lower 16 bits of the FIFO data to be output through MSIOF_TXD. |

54.2.10 MSIOF Receive FIFO Data Register (SIRFDR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SIRFDR is a 32-bit read-only register that stores the receive FIFO data of the MSIOF.



| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|--------------|---------------|-----|--|
| 31 to 16 | SIRFD1[15:0] | Undefined | R | Store the upper 16 bits of the FIFO data received through MSIOF_RXD. |
| 15 to 0 | SIRFD2[15:0] | Undefined | R | Store the lower 16 bits of the FIFO data received through MSIOF_RXD. |

54.2.11 MSIOF Status Register (SISTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Each bit in SISTR becomes an MSIOF interrupt source when the corresponding bit in SIIER is set to 1.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|-------|-----------|----|----|----|----|-------|----|------------|-------|-------|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | TFEMP | TDREQ | — | — | — | — | TEOF | — | TFS ERR | TFOVF | TFUDF | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/WC1 | R | R | R | R | R | R/WC1 | R | R/WC1 | R/WC1 | R/WC1 | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | RFFUL | RDRE Q | — | — | — | — | REOF | — | RFS ERR | RFUDF | RFOVF | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/WC1 | R | R | R | R | R | R/WC1 | R | R/WC1 | R/WC1 | R/WC1 | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-------|---|
| 31, 30 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 29 | TFEMP | B'0 | R/WC1 | Transmit FIFO Empty This bit is set to 1, when the TXE bit in SICTR is 1, and transmit FIFO is empty. Writing 1 to this bit clears this bit. Writing 0 to this bit is ignored. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: Transmit FIFO is not empty. 1: Transmit FIFO is empty. |
| 28 | TDREQ | B'0 | R | Transmit Data Transfer Request This bit is set, when the empty space in the transmit FIFO exceeds the size specified by the TFWM bits in SIFCTR. When transferring transmit data through the DMAC, this bit is always cleared after one DMAC access. After DMAC access, when the conditions for setting this bit are satisfied, the MSIOF again sets this bit to 1. This bit is valid when the TXE bit in SICTR is 1. This bit indicates a state; if the size of empty space in the transmit FIFO becomes less than the size specified by the TFWM bits in SIFCTR, the MSIOF clears this bit. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. However, when the TDMAE bit is 1, only a DMAC transfer request is issued. 0: The size of empty space in the transmit FIFO has not exceeded the size specified by the TFWM bits in SIFCTR. 1: The size of empty space in the transmit FIFO has exceeded the size specified by the TFWM bits in SIFCTR. |
| 27 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-------|---|
| 23 | TEOF | B'0 | R/WC1 | <p>Frame Transmission End</p> <p>This bit is set, when one-frame data transmission is completed.</p> <p>This bit is valid when the TXE bit in SICTR is 1.</p> <p>Writing 1 to this bit clears this bit.</p> <p>Writing 0 to this bit is ignored.</p> <p>If the interrupt request from this bit is enabled, an MSIOF interrupt is issued.</p> <p>0: One-frame transmission end is not detected.</p> <p>1: One-frame transmission end is detected.</p> |
| 22 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 21 | TFSEERR | B'0 | R/WC1 | <p>Transmit Frame Synchronization Error</p> <p>This bit is set, when the next transmit frame synchronization timing arrives before the previous data transmission has been completed.</p> <p>If a transmit frame synchronization error occurs, the MSIOF performs transmission only for the slots that can be transferred.</p> <p>This bit is valid when the TXE bit in SICTR is 1.</p> <p>Writing 1 to this bit clears this bit.</p> <p>Writing 0 to this bit is ignored.</p> <p>If the interrupt request from this bit is enabled, an MSIOF interrupt is issued.</p> <p>0: No transmit frame synchronization error has occurred.</p> <p>1: A transmit frame synchronization error has occurred.</p> |
| 20 | TFOVF | B'0 | R/WC1 | <p>Transmit FIFO Overflow</p> <p>A transmit FIFO overflow means that there has been an attempt to write to SITFDR when the transmit FIFO is full.</p> <p>If a transmit FIFO overflow occurs, the MSIOF ignores the write operation causing the overflow.</p> <p>This bit is valid when the TXE bit in SICTR is 1.</p> <p>Writing 1 to this bit clears this bit.</p> <p>Writing 0 to this bit is ignored.</p> <p>If the interrupt request from this bit is enabled, an MSIOF interrupt is issued.</p> <p>0: No transmit FIFO overflow has occurred.</p> <p>1: A transmit FIFO overflow has occurred.</p> |
| 19 | TFUDF | B'0 | R/WC1 | <p>Transmit FIFO Underflow</p> <p>A transmit FIFO underflow means that loading for transmission has occurred when the transmit FIFO is empty. Output data is unknown when under flow is occurred.</p> <p>This bit is valid when the TXE bit in SICTR is 1.</p> <p>Writing 1 to this bit clears this bit.</p> <p>Writing 0 to this bit is ignored.</p> <p>If the interrupt request from this bit is enabled, an MSIOF interrupt is issued.</p> <p>0: No transmit FIFO underflow has occurred.</p> <p>1: A transmit FIFO underflow has occurred.</p> <p>Note: If transmission size is larger than 64-words per frame, do not use this bit. See 54.4.5 for details.</p> |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-------|--|
| 18 to 14 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 13 | RFFUL | B'0 | R/WC1 | Receive FIFO Full This bit is valid when the RXE bit in SICTR is 1. Writing 1 to this bit clears this bit. Writing 0 to this bit is ignored. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: Receive FIFO is not full. 1: Receive FIFO is full. |
| 12 | RDREQ | B'0 | R | Receive Data Transfer Request This bit is set, when the valid data space in the receive FIFO exceeds the size specified by the RFWM bits in SIFCTR. When transferring receive data through the DMAC, this bit is always cleared by one DMAC access. After DMAC access, when the conditions for setting this bit are satisfied, the MSIOF again sets this bit to 1. This bit is valid when the RXE bit in SICTR is 1. This bit indicates a state; if the size of valid data space in the receive FIFO becomes less than the size specified by the RFWM bits in SIFCTR, the MSIOF clears this bit. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. However, when the RDMAE bit is 1, only a DMAC transfer request is issued. 0: The size of valid data space in the receive FIFO has not exceeded the size specified by the RFWM bits in SIFCTR. 1: The size of valid data space in the receive FIFO has exceeded the size specified by the RFWM bits in SIFCTR. |
| 11 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | REOF | B'0 | R/WC1 | Frame Reception End The MSIOF issues the frame reception end flag upon completion of one-frame data reception. This bit is valid when the RXE bit in SICTR is 1. Writing 1 to this bit clears this bit. Writing 0 to this bit is ignored. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: One-frame reception end is not detected. 1: One-frame reception end is detected. |
| 6 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|--------|----------|---------------|-------|---|
| 5 | RFSERR | B'0 | R/WC1 | <p>Receive Frame Synchronization Error</p> <p>A receive frame synchronization error occurs when the next receive frame synchronization timing arrives before the previous data reception has been completed.</p> <p>If a receive frame synchronization error occurs, the MSIOF performs reception only for the slots that can be transferred. This bit is valid when the RXE bit in SICTR is 1.</p> <p>Writing 1 to this bit will clear this bit.</p> <p>Writing 0 to this bit will be ignored.</p> <p>If the interrupt request from this bit is enabled, an MSIOF interrupt is issued.</p> <p>0: No receive frame synchronization error has occurred 1: A receive frame synchronization error has occurred</p> |
| 4 | RFUDF | B'0 | R/WC1 | <p>Receive FIFO Underflow</p> <p>A receive FIFO underflow means that reading of SIRFDR has occurred when the receive FIFO is empty.</p> <p>If a receive FIFO underflow occurs, the value read from SIRFDR is not guaranteed.</p> <p>This bit is valid when the RXE bit in SICTR is 1.</p> <p>Writing 1 to this bit clears this bit.</p> <p>Writing 0 to this bit is ignored.</p> <p>If the interrupt request from this bit is enabled, an MSIOF interrupt is issued.</p> <p>0: No receive FIFO underflow has occurred. 1: A receive FIFO underflow has occurred.</p> |
| 3 | RFOVF | B'0 | R/WC1 | <p>Receive FIFO Overflow</p> <p>A receive FIFO overflow means that writing has been caused by receiving operation when the receive FIFO is full.</p> <p>If a receive FIFO overflow occurs, the receive data causing the overflow is lost.</p> <p>This bit is valid when the RXE bit in SICTR is 1.</p> <p>Writing 1 to this bit clears this bit.</p> <p>Writing 0 to this bit is ignored.</p> <p>If the interrupt request from this bit is enabled, an MSIOF interrupt is issued.</p> <p>0: No receive FIFO overflow has occurred. 1: A receive FIFO overflow has occurred.</p> |
| 2 to 0 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

54.2.12 MSIOF Interrupt Enable Register (SIER)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SIER is a 32-bit readable/writable register that enables the issuance of MSIOF interrupts. When each bit in this register is set to 1 and the corresponding bit in SISTR is set to 1, the MSIOF issues an interrupt.

| | | | | | | | | | | | | | | | | |
|----------------|-------|----|--------|--------|----|----|----|----|-------|----|---------|--------|--------|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TDMAE | — | TFEMPE | TDREQE | — | — | — | — | TEOFE | — | TFSERRE | TFOVFE | TFUDFE | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R/W | R/W | R | R | R | R | R/W | R | R/W | R/W | R/W | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RDMAE | — | RFFULE | RDREQE | — | — | — | — | REOFE | — | RFSERRE | RFUDFE | RFOVFE | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R/W | R/W | R | R | R | R | R/W | R | R/W | R/W | R/W | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|---|
| 31 | TDMAE | B'0 | R/W | Transmit Data DMA Transfer Request Enable Specifies whether to send an interrupt as an interrupt request to the CPU or a transfer request to the DMAC. The TDREQE bit can be set as the request source. 0: Sends an interrupt request to the CPU. 1: Sends a DMA transfer request to the DMAC. |
| 30 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 29 | TFEMPE | B'0 | R/W | Transmit FIFO Empty Enable 0: Disables interrupts due to transmit FIFO empty. 1: Enables interrupts due to transmit FIFO empty. |
| 28 | TDREQE | B'0 | R/W | Transmit Data Transfer Request Enable 0: Disables interrupts due to transmit data transfer requests. 1: Enables interrupts due to transmit data transfer requests. |
| 27 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 | TEOFE | B'0 | R/W | Frame Transmission End Enable 0: Disables a frame transmission end interrupt. 1: Enables a frame transmission end interrupt. |
| 22 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 21 | TFSERRE | B'0 | R/W | Transmit Frame Synchronization Error Enable 0: Disables interrupts due to transmit frame synchronization errors. 1: Enables interrupts due to transmit frame synchronization errors. |
| 20 | TFOVFE | B'0 | R/W | Transmit FIFO Overflow Enable 0: Disables interrupts due to transmit FIFO overflow. 1: Enables interrupts due to transmit FIFO overflow. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|--|
| 19 | TFUDFE | B'0 | R/W | Transmit FIFO Underflow Enable 0: Disables interrupts due to transmit FIFO underflow. 1: Enables interrupts due to transmit FIFO underflow. |
| 18 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | RDMAE | B'0 | R/W | Receive Data DMA Transfer Request Enable Specifies whether to send an interrupt as an interrupt request to the CPU or a transfer request to the DMAC. The RDREQE bit can be set as the request source. 0: Sends an interrupt request to the CPU. 1: Sends a DMA transfer request to the DMAC. |
| 14 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 13 | RFFULE | B'0 | R/W | Receive FIFO Full Enable 0: Disables interrupts due to receive FIFO full. 1: Enables interrupts due to receive FIFO full. |
| 12 | RDREQE | B'0 | R/W | Receive Data Transfer Request Enable 0: Disables interrupts due to receive data transfer requests. 1: Enables interrupts due to receive data transfer requests. |
| 11 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | REOFE | B'0 | R/W | Frame Reception End Enable 0: Disables a frame reception end interrupt. 1: Enables a frame reception end interrupt. |
| 6 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 5 | RFSERRE | B'0 | R/W | Receive Frame Synchronization Error Enable 0: Disables interrupts due to receive frame synchronization errors. 1: Enables interrupts due to receive frame synchronization errors. |
| 4 | RFUDFE | B'0 | R/W | Receive FIFO Underflow Enable 0: Disables interrupts due to receive FIFO underflow. 1: Enables interrupts due to receive FIFO underflow. |
| 3 | RFOVFE | B'0 | R/W | Receive FIFO Overflow Enable 0: Disables interrupts due to receive FIFO overflow. 1: Enables interrupts due to receive FIFO overflow. |
| 2 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

54.2.13 MSIOF FIFO Control Register (SIFCTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SIFCTR is a 32-bit readable/writable register that indicates the area available for the transmit/receive FIFO transfer.

| | | | | | | | | | | | | | | | | |
|----------------|-----------|-----|-----|-----------|----|-----------|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | TFWM[2:0] | | | — | — | TFUA[6:0] | | | | | | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RFWM[2:0] | | | RFUA[8:0] | | | | | | — | — | — | — | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|-----------|---------------|-----|---|
| 31 to 29 | TFWM[2:0] | B'000 | R/W | <p>Transmit FIFO Watermark</p> <p>A transfer request of the transmit FIFO is issued by the TDREQE bit in SIIER.</p> <p>The transmit FIFO always operates as a 64-stage FIFO regardless of the setting of these bits.</p> <p>B'000: Issues a transfer request when 64 stages of the transmit FIFO are empty.</p> <p>B'001: Issues a transfer request when 32 or more stages of the transmit FIFO are empty.</p> <p>B'010: Issues a transfer request when 24 or more stages of the transmit FIFO are empty.</p> <p>B'011: Issues a transfer request when 16 or more stages of the transmit FIFO are empty.</p> <p>B'100: Issues a transfer request when 12 or more stages of the transmit FIFO are empty.</p> <p>B'101: Issues a transfer request when 8 or more stages of the transmit FIFO are empty.</p> <p>B'110: Issues a transfer request when 4 or more stages of the transmit FIFO are empty.</p> <p>B'111: Issues a transfer request when 1 or more stages of transmit FIFO are empty.</p> |
| 28, 27 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 26 to 20 | TFUA[6:0] | H'40 | R | <p>Transmit FIFO Usable Area</p> <p>Indicate the number of words that can be transferred by the CPU or DMAC as B'000_0000 (full) to B'100_0000 (empty).</p> |
| 19 to 16 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|-----------|---------------|-----|--|
| 15 to 13 | RFWM[2:0] | B'000 | R/W | <p>Receive FIFO Watermark</p> <p>A transfer request of the receive FIFO is issued by the RDREQE bit in SIIER.</p> <p>The receive FIFO always operates as a 256-stage FIFO regardless of the setting of these bits.</p> <p>B'000: Issues a transfer request when 1 stage or more of the receive FIFO are valid.</p> <p>B'001: Issues a transfer request when 4 or more stages of the receive FIFO are valid.</p> <p>B'010: Issues a transfer request when 8 or more stages of the receive FIFO are valid.</p> <p>B'011: Issues a transfer request when 16 or more stages of the receive FIFO are valid.</p> <p>B'100: Issues a transfer request when 32 or more stages of the receive FIFO are valid.</p> <p>B'101: Issues a transfer request when 64 or more stages of the receive FIFO are valid.</p> <p>B'110: Issues a transfer request when 128 or more stages of the receive FIFO are valid.</p> <p>B'111: Issues a transfer request when 256 stages of the receive FIFO are valid.</p> |
| 12 to 4 | RFUA[8:0] | All 0 | R | <p>Receive FIFO Usable Area</p> <p>Indicate the number of words that can be transferred by the CPU or DMAC as B'0_0000_0000 (empty) to B'1_0000_0000 (full).</p> |
| 3 to 0 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

54.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

54.3.1 Operating Mode

(1) Common transmit/receive mode

Common clock and frame synchronization signals are used for transmission and reception.

Master mode: MSIOF_SCK and MSIOF_SYNC are output.

Slave mode: MSIOF_SCK and MSIOF_SYNC are input.

54.3.2 Serial Clocks

(1) Clock output in master mode

In master mode, the baud rate generator is used to generate the serial clock. The division ratio is from 1/2 to 1/1024.

(2) Clock input in slave mode

In slave mode, the clock input for transmission and reception is used as the serial clock.

(3) Multiple channel function

This module provides the multiple channel function as shown in Figure 54.2.

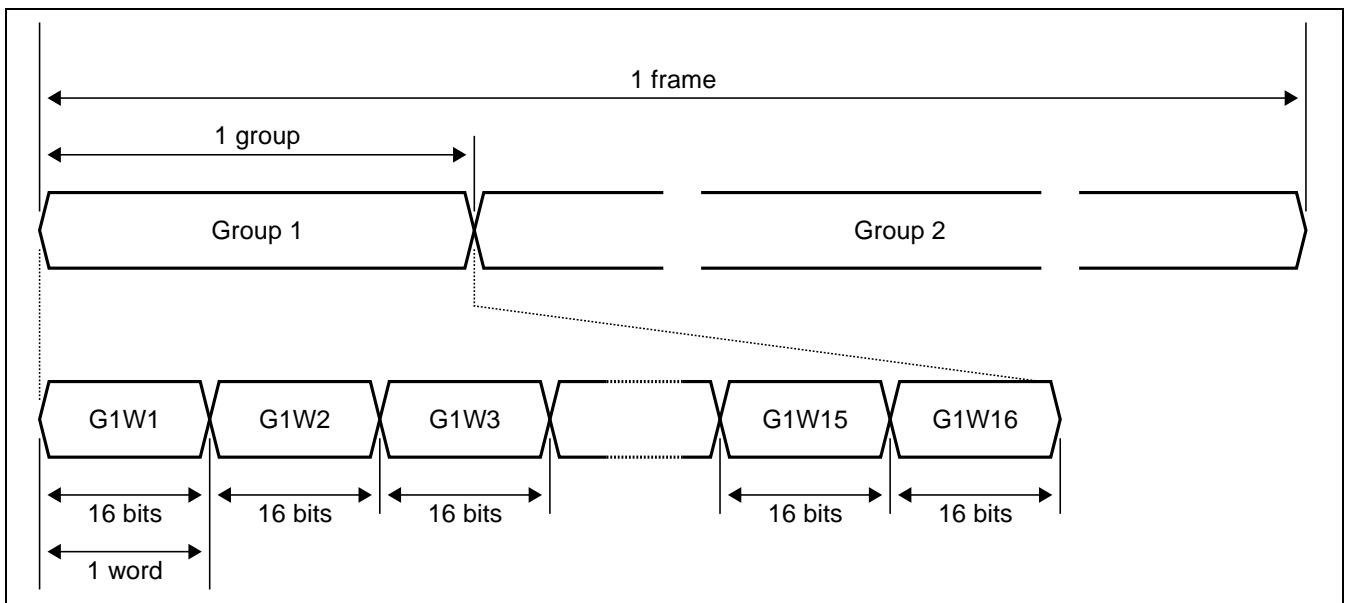


Figure 54.2 Multiple Channel Structure

The following conditions can be specified for the multiple channel function.

- Up to 2 groups in one frame
- Up to 64 words in one group
- 8, 16, 24, or 32 bits in one word

54.3.3 Serial Timing

(1) MSIOF_SYNC

The MSIOF_SYNC is a frame synchronization signal. The following three modes are available.

[RZ/G2H]

- Level/SPI: Level signal asserted during frame transmission

[Except for RZ/G2H]

- Frame start synchronization pulse: 1-bit-width pulse indicating the start of a frame
- Level/SPI: Level signal asserted during frame transmission
- L/R: 1/2-frame-width pulse indicating the first-half groups in a high level and the last-half groups in a low level

Figure 54.3 to Figure 54.5 show the synchronization timing in these modes using the MSIOF_SYNC signal.

(a) Frame start synchronization pulse

The rising edge of the synchronization pulse indicates the start of a frame.

The delay between the rising edge of the synchronization signal and the start of data transmission or reception can be specified with the DTDL bits. The width of the synchronization pulse can be extended through the SYNCDL bit setting.

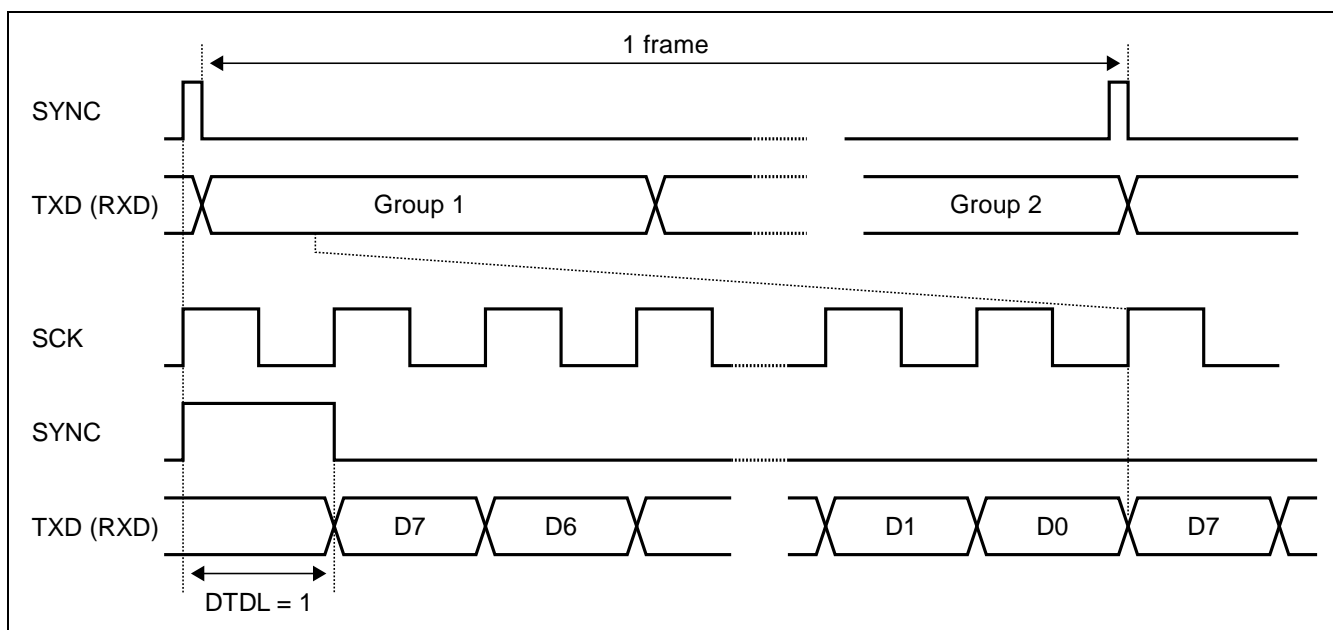


Figure 54.3 Synchronization Timing in Start Frame Synchronization Pulse Mode

(b) Level synchronization

The synchronization signal is driven high for the frame length.

The delay between the rising edge of the synchronization signal and the start of transmission or reception can be specified with the DTDL bits and the delay between the end of transmission or reception and the falling edge of the synchronization signal can be specified with the SYNCNL bits (DTDL = 0 to 2(RZ/G2H is only 2) and SYNCNL = 0 to 3). These settings are valid only in master mode. This mode is available for the SPI master or slave operation.

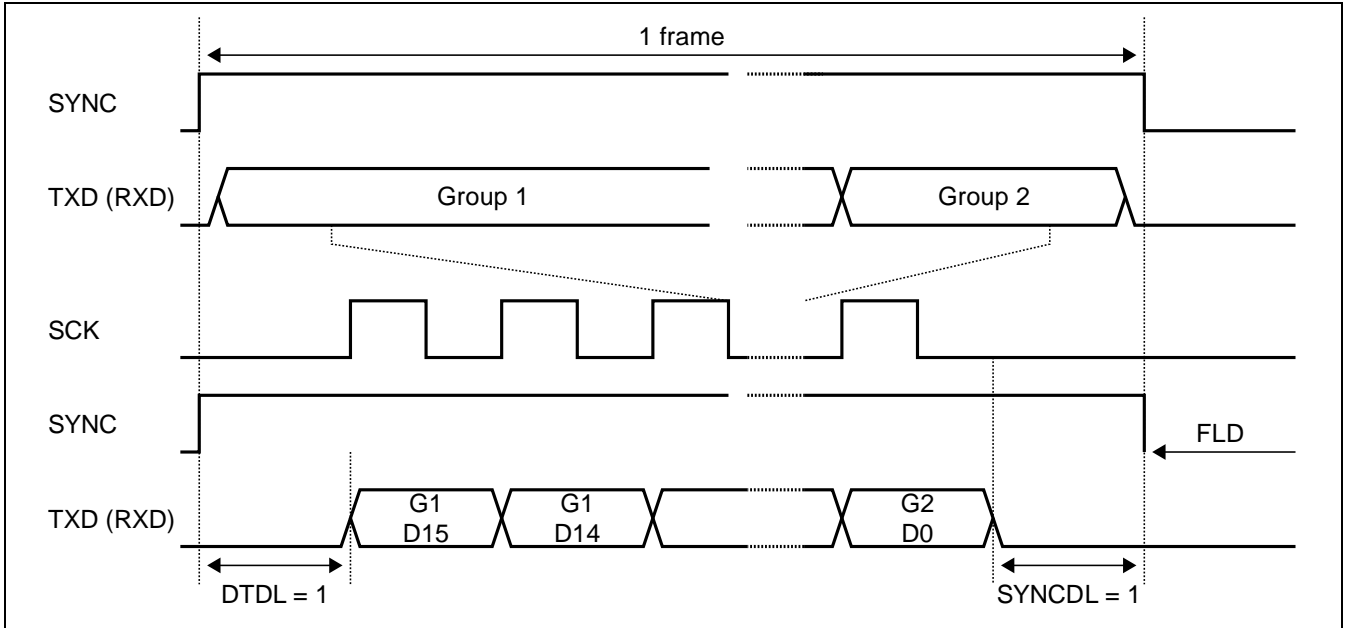


Figure 54.4 Synchronization Timing in Level Synchronization Mode

(c) L/R synchronization

A high level in the synchronization signal indicates the first-half groups and a low level indicates the last-half groups.

The delay between the rising or falling edge of the synchronization signal and the start of transmission or reception can be specified with the DTDL bits.

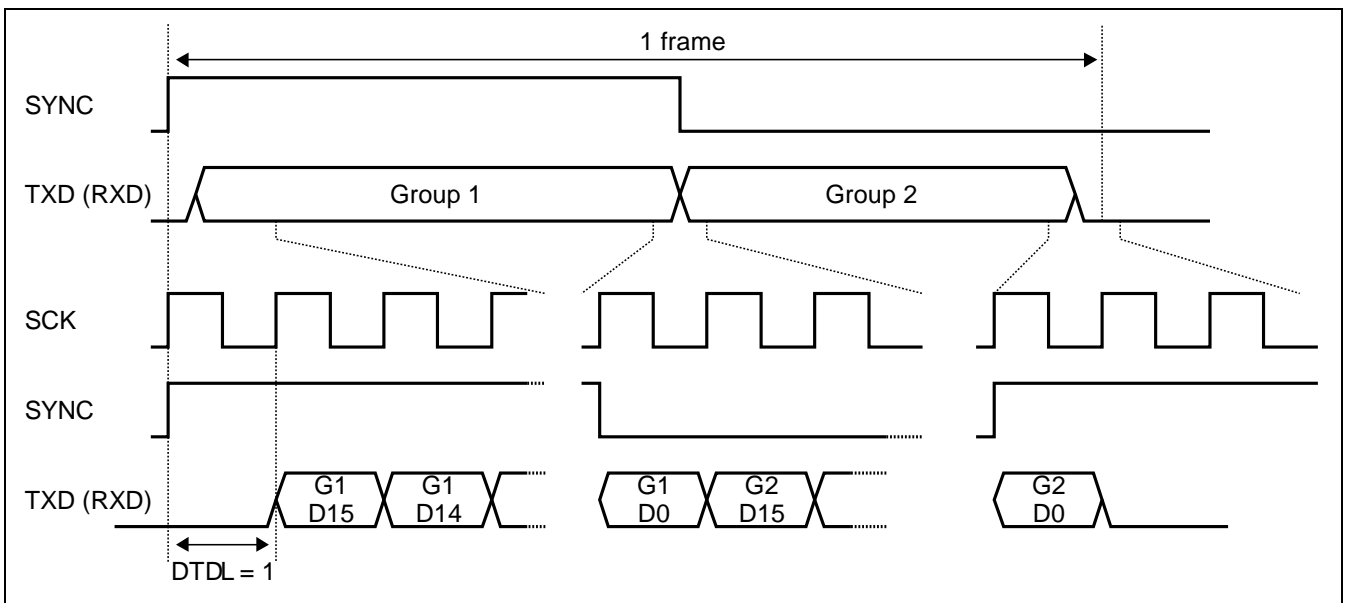


Figure 54.5 Synchronization Timing in L/R Synchronization Mode

(2) Transmit/receive timing

The timing of MSIOF_TXD transmission relative to MSIOF_SCK and of MSIOF_RXD can be specified as sampling on either edge, as listed below. The respective settings are made in TEDG and REDG bits of SICTR.

- Falling-edge sampling
- Rising-edge sampling

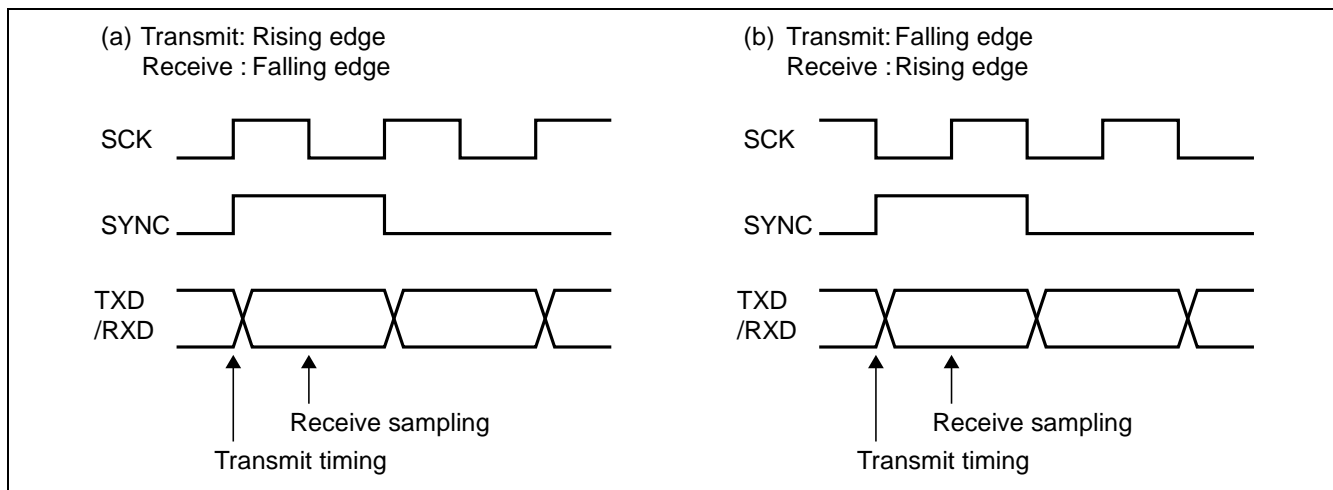


Figure 54.6 Transmit/Receive Timing.

54.3.4 Transfer Data Allocation to Registers

(1) Transmit/receive data

Transmit/receive data (FIFO data) should be written to or read from the following registers.

- Transmit FIFO data writing: SITFDR (32-bit access)
- Receive FIFO data reading: SIRFDR (32-bit access)

Figure 54.7 shows the bit alignment of these registers.

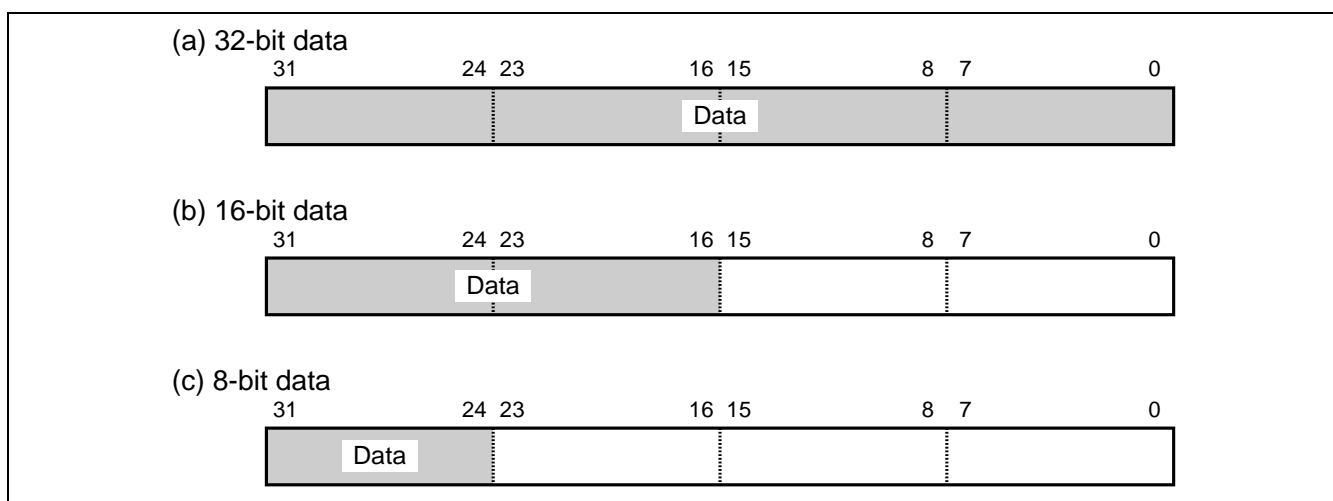


Figure 54.7 Transmit/Receive Data Bit Alignment

54.3.5 FIFO

(1) Overview

The transmission and receive FIFOs of the MSIOF have the following features.

- 32 bits × 64 stages for transmission and 32 bits × 256 stages for reception
- The FIFO pointer is updated in one read or write cycle regardless of the access size of the CPU or DMAC. (One access cannot be divided into multiple accesses.)

(2) Transfer request

A request for FIFO data transfer can be issued to the CPU or DMAC as the following interrupt sources.

- FIFO transmit request: TDREQ (interrupt source for transmission)
- FIFO receive request: RDREQ (interrupt source for reception)

The conditions for requesting FIFO data transfer can be specified separately for transmission and reception. The conditions for the transmit FIFO and receive FIFO are specified in the TFWM[2:0] bits and RFWM[2:0] bits in SIFCTR, respectively.

Table 54.7 Condition for Issuing Transmit Request

| TFWM[2:0] | Number of Requested Stages | Condition for Transmit Request | Areas Used |
|-----------|----------------------------|---------------------------------|------------|
| B'000 | 1 | 64 stages of empty area | Smallest |
| B'001 | 32 | 32 or more stages of empty area | |
| B'010 | 40 | 24 or more stages of empty area | |
| B'011 | 48 | 16 or more stages of empty area | |
| B'100 | 52 | 12 or more stages of empty area | |
| B'101 | 56 | 8 or more stages of empty area | |
| B'110 | 60 | 4 or more stages of empty area | |
| B'111 | 64 | 1 or more stages of empty area | |

Table 54.8 Condition for Issuing Receive Request

| RFWM[2:0] | Number of Requested Stages | Condition for Receive Request | Areas Used |
|-----------|----------------------------|----------------------------------|------------|
| B'000 | 1 | 1 or more stages of valid data | Smallest |
| B'001 | 4 | 4 or more stages of valid data | |
| B'010 | 8 | 8 or more stages of valid data | |
| B'011 | 16 | 16 or more stages of valid data | |
| B'100 | 32 | 32 or more stages of valid data | |
| B'101 | 64 | 64 or more stages of valid data | |
| B'110 | 128 | 128 or more stages of valid data | |
| B'111 | 256 | 256 stages of valid data | |

The maximum FIFO stages are always available even if the data area or empty area exceeds the size specified for the transfer condition. Accordingly, an overflow error or an underflow error occurs if the data area or empty area exceeds the maximum number of FIFO stages.

The FIFO transfer request is canceled when the specified condition is not satisfied even if the FIFO does not become empty or full.

(3) Number of FIFOs

The number of transmit and receive FIFO stages used are indicated by the following registers.

- Transmit FIFO: The number of empty FIFO stages is indicated by the TFUA [6:0] bits in SIFCTR.
- Receive FIFO: The number of valid data stages is indicated by the RFUA [8:0] bits in SIFCTR.

These registers show the number of data stages that can be transferred by the CPU. DMA transfer should be set the following setting.

SIFCTR.TFWM[2:0] = B'111

SIFCTR.RFWM[2:0] = B'000

54.3.6 Transmit and Receive Procedures

(1) Transmission in master mode

Figure 54.8 and Figure 54.9 shows an example of settings and operation for transmission in master mode.

| No. | Operating flow | Description | Operation |
|-----|--|--|----------------------------------|
| 1 | SITMDR1/2/3 SITSCR SIFCTR SIIER setting | Operating mode Clock setting FIFO water mark Setting interrupt/DMA enable | |
| 2 | SICTR Set SICTR.TSCKE to 1 | Transmission rate setting Baud rate generator starting | Output serial clock |
| 3 | | Read SICTR.TSCKE = 1 | |
| 4 | SITFDR | Write transmission data to FIFO | |
| 5 | Set SICTR.TXE to 1 | Transmission enable | |
| 6 | | Read SICTR.TXE = 1 | |
| 7 | Set SICTR.TFSE to 1 | Starting MSIOF_SYNC output | |
| 8 | | Read SICTR.TFSE = 1 | |
| 9 | Starting transmission | | Data transmission |
| 10 | | | Generating interrupt/DMA request |
| 11 | Write transmission data to FIFO | | |
| 12 | | | |
| 13 | Set SICTR.TFSE to 0 | Frame sync disable | |
| 14 | | Read SICTR.TFSE = 0 | |
| 15 | Set SICTR.TXE to 0 | Transmission disable | |
| 16 | | Read SICTR.TXE = 0 | |

Figure 54.8 Example of Transmit Procedure in Master Mode (without DMAC)

| No. | Operating flow | Description | Operation |
|-----|--|--|-----------------------------------|
| 1 | <ul style="list-style-type: none"> Setting interrupt controller Setting source address of DMAC channel Setting Destination address of DMAC channel is SITFDR Setting number of Transfer in DMAC transfer count register Setting transfer as fixed destination address mode. Setting Resouse Select register of DMAC channel, please refer to Chapter describe DMAC module to get the value. DMAC start. | <ul style="list-style-type: none"> DMAC setting, Refer To Chapter Describe DMAC module for More detail Setting to interrupt controller for MSIOF interrupt event, refer to chapter describe interrupt controller for detail. | |
| 2 | SITMDR1/2/3 SITSCR SIFCTR SIIER setting | <ul style="list-style-type: none"> operating mode clock setting fifo water mark interrupt: Refer to Section 54.3.7 or SIIER register description (54.2.12) for MSIOF's Interrupt Factor. dma enable (SIIER.TDMAE) | |
| 3 | SICTR Set SICTR.TSCKE to 1 | Transmission rate setting Baud rate generator starting | Output serial clock |
| 4 | | Read SICTR.TSCKE = 1 | |
| 5 | Set SICTR.TXE to 1 | Transmission enable | |
| 6 | | Read SICTR.TXE = 1 | |
| 7 | Set SICTR.TFSE to 1 | Starting MSIOF_SYNC output | |
| 8 | | Read SICTR.TFSE = 1 | |
| 9 | Starting transmission | | Data transmission |
| 10 | | | Generating interrupt/DM A request |
| 11 | Write transmission data to FIFO | | |
| 12 | | Wait bit TE in DMAC CHCR register set to 1, refer to chapter describe DMAC module for detail. Or use interrupt to avoid software polling, DMAC module support interrupt when the transfer end. | |
| 13 | Set SICTR.TFSE to 0 | Frame sync disable | |
| 14 | | Read SICTR.TFSE = 0 | |
| 15 | Set SICTR.TXE to 0 | Transmission disable | |
| 16 | | <ul style="list-style-type: none"> Read SICTR.TXE = 0 Disable DMAC channel, refer to chapter describe DMAC module for detail | |

Figure 54.9 Example of Transmit Procedure in Master Mode (with DMAC)

(2) Reception in master mode (with SITMDR1.PCON = 1)

Figure 54.10 and Figure 54.11 show an example of settings and operation for reception in master mode with SITMDR1.PCON = 1.

| No. | Operating flow | Description | Operation |
|-----|---|---|----------------------------------|
| 1 | SIRMDR1/2/3 SITMDR1/2/3 SITSCR SIFCTR SIER setting | Operating mode Operating mode (for SPI dummy transmission) Serial clock FIFO water mark Interrupt/DMA setting | |
| 2 | SICTR Set SICTR.TSCKE to 1 | Baud rate setting Baud rate generator starting | Output serial clock |
| 3 | | Read SICTR.TSCKE = 1 | |
| 4 | SITFDR | Write dummy transmission data to SITFDR The number of dummy data should equal the number of reception data. | |
| 5 | Set SICTR.TXE to 1 Set SICTR.RXE to 1 | Transmission enable Reception enable | |
| 6 | | Read SICTR.TXE = 1 Read SICTR.RXE = 1 | |
| 7 | Set SICTR.TFSE to 1 | Frame sync enable | Output MSIOF_SYNC |
| 8 | | Read SICTR.TFSE = 1 | |
| 9 | Synchronizing with MSIOF_SYNC (1) Transmission FIFO dummy data (2) Reception data | | Transmission and reception |
| 10 | | | Generating interrupt/DMA request |
| 11 | Write dummy data to FIFO Read fifo data from SIRFDR | Transmission dummy data Reception data reading | |
| 12 | | | |
| 13 | Set SICTR.TFSE to 0 | Disable frame sync enable | |
| 14 | | Read SICTR.TFSE = 0 | |
| 15 | Set SICTR.TXE and SICTR.RXE to 0 | Disable transmission enable and reception enable | |
| 16 | | Read SICTR.TXE = 0 Read SICTR.RXE = 0 | |

Figure 54.10 Example of Receive Procedure in Master Mode (SITMDR1.PCON = 1) (without DMAC)

| No. | Operating flow | Description | Operation |
|-----|---|---|------------------------------------|
| 1 | <ul style="list-style-type: none"> Setting Interrupt controller In receive master mode, dummy data need to be written into SITFDR, so 2 DMAC channel must be used. <u>Setting DMAC for transfer dummy data:</u> <ul style="list-style-type: none"> - Setting Source Address is memories address which locate dummy data. - Setting Destination Address is SITFDR. - Setting the number of transfer in DMAC transfer count register equal the number of received data. - DMAC transfer as fixed destination address mode. <u>Setting DMAC for received data:</u> <ul style="list-style-type: none"> - Setting Source address is SIRFDR register. - Setting Destination address is memories address which locate the received data. - Setting the number of transfer in DMAC transfer count register equal the number of received data. - DMAC transfer as incremental destination address mode. DMAC start for 2 channel. | <ul style="list-style-type: none"> Setting Interrupt Controller to enable MSIOF interrupt, refer to chapter describe Interrupt Controller for detail DMAC setting, refer to chapter describe DMAC module for detail. | |
| 2 | SIRMDR1/2/3 SITMDR1/2/3 SITSCR SIFCTR SIIER setting | <ul style="list-style-type: none"> operating mode operating mode (for SPI dummy transmission) serial clock fifo water mark interrupt : Refer to section 54.3.7 or SIIER register description (54.2.12) for MSIOF's Interrupt Factor. dma enable (SIIER.RDMAE) | |
| 3 | SICTR Set SICTR.TSCKE to 1 | baud rate setting baud rate generator starting | output serial clock |
| 4 | | Read SICTR.TSCKE = 1 | |
| 5 | Set SICTR.TXE to 1 Set SICTR.RXE to 1 | Trasnmission enable Reception enable | |
| 6 | | Read SICTR.TXE = 1 Read SICTR.RXE = 1 | |
| 7 | Set SICTR.TFSE to 1 | Frame sync enable | Output MSIOF_SYNC |
| 8 | | Read SICTR.TFSE = 1 | |
| 9 | Synchronizing with MSIOF_SYNC (1) Transmission fifo dummy data (2) Reception data | | transmission and reception |
| 10 | | | Generating interrupt / DMA request |
| 11 | Write dummy data to fifo Read fifo data from SIRFDR | Transmission dummy data Reception data reading | |
| 12 | | Wait bit TE in DMAC CHCR register set to 1, refer to chapter describe DMAC module for detail. Or use interrupt to avoid software polling, DMAC module support interrupt when the transfer end. | |
| 13 | Set SICTR.TFSE to 0 | Disable frame sync enable | |
| 14 | | Read SICTR.TFSE = 0 | |
| 15 | Set SICTR.TXE and SICTR.RXE to 0 | Disable transmission enable and reception enable | |
| 16 | | | |

Figure 54.11 Example of Receive Procedure in Master Mode (SITMDR1.PCON = 1) (with DMAC)

(3) Transmission in slave mode

Figure 54.12 shows an example of settings and operation for transmission in slave mode. (transmission size is less than or equal to 64-words per frame). Figure 54.13 shows an example of settings and operation for transmission in slave mode (transmission size is larger than 64-words per frame without DMAC). Figure 54.14 shows an example of settings and operation for transmission in slave mode (transmission size is more than 64-words per frame with DMAC).

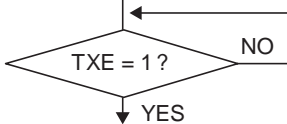
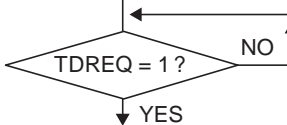
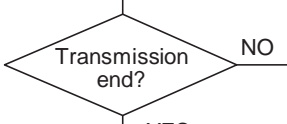
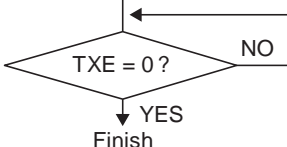
| No. | Operating flow | Description | Operation |
|-----|---|---|----------------------------------|
| 1 | Master device setting | Depend on master device | Signal polarity |
| 2 | SITMDR1/2/3 SIFCTR SIIER setting | Operating mode FIFO water mark Interrupt/DMA enable (Do not use SIIER.TFUDFE) | |
| 3 | SITFDR | Transmission data setting | |
| 4 | Set SICTR.TXE to 1 | Setting transmission enable | |
| 5 |  | Read SICTR.TXE = 1 | |
| 6 | Data transmission synchronize with MSIOF_SYNC | | Data transmission |
| 7 |  | | Generating interrupt/DMA request |
| 8 | Write FIFO data to SITFDR | Data transmission | |
| 9 |  | | |
| 10 | Set SICTR.TXE to 0 | Transmission disable | |
| 11 |  | Read SICTR.TXE = 0 | |

Figure 54.12 Example of Transmit Procedure in Slave Mode (transmission size is less than or equal to 64-words per frame)

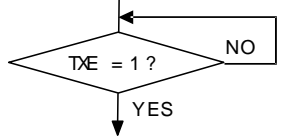
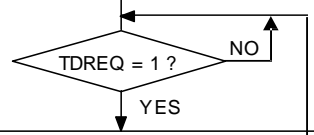
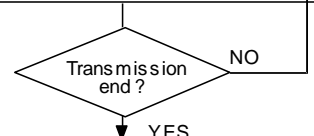
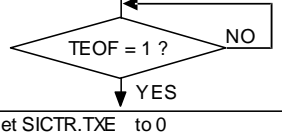
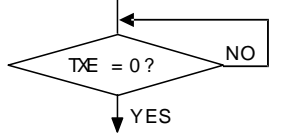
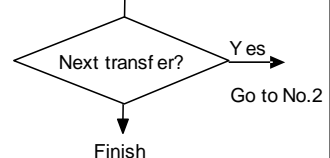
| No. | Operating flow | Description | Operation |
|-----|---|--|---|
| 1 | Master device setting | Depend on master device | Signal polarity |
| 2 | SITMDR1/2/3 SIFCTR_SIIER setting | Operating mode FIFO watermark Interrupt/DMA enable (Do not use SIIER.TFUDFE) | Do not include dummy data when setting the transfer word count.(SITMDR2.WDLEN1, SITMDR3.WDLEN2) |
| 3 | PFC setting | Change MSIOF_TXD, MSIOF_RXD, MSIOF_SCK, MSIOF_SYNC to module side | |
| 4 | SITFDR | Transmission data setting | Need to transmit FIFO full |
| 5 | Set SICTR.TXE to 1 | Setting transmission enable | |
| 6 |  | Read SICTR.TXE = 1 | |
| 7 | Data transmission synchronize with MSIOF_SYNC | | Data transmission |
| 8 |  | | Generating interrupt/DMA request |
| 9 | Write FIFO data to SITFDR | Data transmission After the last data is stored, store additional dummy data. | Do not need to transmit dummy data(1-word). |
| 10 |  | | |
| 11 |  | | |
| 12 | Set SICTR.TXE to 0 | Transmission disable | |
| 13 |  | Read SICTR.TXE = 0 | |
| 14 | PFC setting | Change MSIOF_SCK, MSIOF_SYNC to GPIO(In) side | |
| 15 | Assert SW reset for MSIOF | SRCLR2.MSIOF (Refer to sec 8A. Module Standby, Software Reset) | Clear dummy data |
| 16 | De-assert SW reset for MSIOF | SRSTCLR2.MSIOF (Refer to sec 8A. Module Standby, Software Reset) | |
| 17 |  | | |

Figure 54.13 Example of Transmit Procedure in Slave Mode (transmission size is more than 64-words per frame without DMAC)

| No. | Operating flow | Description | Operation |
|-----|---|--|--|
| 1 | Master device setting | Depend on master device | Signal polarity |
| 2 | <ul style="list-style-type: none"> Setting interrupt controller Setting source address of DMAC channel Setting Destination address of DMAC channel is SITFDR Setting number of Transfer in DMAC transfer count register (add 1-word data as dummy) Setting transfer as fixed destination address mode. Setting Reuse Select register of DMAC channel, please refer to Chapter describe DMAC module to get the value. DMAC start. | <ul style="list-style-type: none"> DMAC setting, Refer To Chapter Describe DMAC module for More detail Setting to interrupt controller for MSIOF interrupt event, refer to chapter describe interrupt controller for detail. | |
| 3 | SITMDR1/2/3 SIFCTR SIIER setting | <ul style="list-style-type: none"> Operating mode FIFO water mark interrupt: Refer to Section 59.3.7 or SIIER register description (59.2.12) for MSIOF's Interrupt Factor (Don't use SIIER.TFUDFE). dma enable (SIIER.TDMAE) | Do not include dummy data when setting the transfer word count. (SITMDR2.WDLEN1, SITMDR3.WDLEN2) |
| 4 | PFC setting | Change MSIOF_TXD, MSIOF_RXD, MSIOF_SCK, MSIOF_SYNC to module side | |
| 5 | Set SICTR.TXE to 1 | Setting transmission enable | |
| 6 | | Read SICTR.TXE = 1 | |
| 7 | Wait for MSIOF Tx FIFO full | SIFCTR.TFUA = B'00_0000 | |
| 8 | Data transmission synchronize with MSIOF_SYNC | | Data transmission |
| 9 | | | Generating interrupt/DMA request |
| 10 | Write transmission data to FIFO 11 | | |
| 11 | | Wait bit TE in DMAC CHCR register set to 1, refer to chapter describe DMAC module for detail. Or use interrupt to avoid software polling, DMAC module support interrupt when the transfer end. | Do not need to transmit dummy data. |
| 12 | | | |
| 13 | Set SICTR.TXE to 0 | Transmission disable | |
| 14 | | Read SICTR.TXE = 0 | |
| 15 | PFC setting | Change MSIOF_SCK, MSIOF_SYNC to GPIO(In) side | |
| 16 | Assert SW reset for MSIOF | SRCLR2.MSIOF (Refer to sec 8A. Module Standby, Software Reset) | Clear dummy data |
| 17 | De-assert SW reset for MSIOF | SRSTCLR2.MSIOF (Refer to sec 8A. Module Standby, Software Reset) | |
| 18 | | | |

Figure 54.14 Example of Transmit Procedure in Slave Mode (transmission size is more than 64-words per frame with DMAC)

(4) Reception in slave mode

Figure 54.15 shows an example of settings and operation for reception in slave mode.

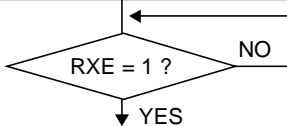
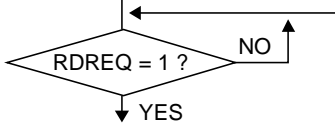
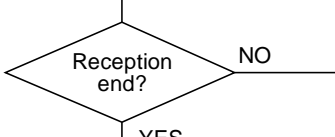
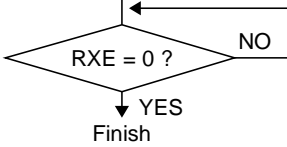
| No. | Operating flow | Description | Operation |
|-----|---|---|----------------------------------|
| 1 | Master device setting | Depend on master device | Signal polarity |
| 2 | SITMDR1 SIRMDR1/2/3 SIFCTR SIIER setting | Operating mode (For slave setting) Operating mode FIFO water mark Interrupt/DMA enable | TRMD = 0, PCON = 1 |
| 3 | Set SICTR.RXE to 1 | Setting receive enable | |
| 4 |  | Read SICTR.RXE = 1 | |
| 5 | Data reception | | Receive data |
| 6 |  | | Generating interrupt/DMA request |
| 7 | Read SIRFDR | Data read from FIFO | |
| 8 |  | | |
| 9 | Set SICTR.RXE to 0 | Reception disable | |
| 10 |  | Read SICTR.RXE = 0 | |

Figure 54.15 Example of Receive Procedure in Slave Mode

(5) Reset

After a power-on reset is applied, module stop state is canceled, or a reset signal is asserted through the TXRST or RXRST setting in SICTR, it takes about 20 cycles of module clock to complete the internal reset of the module. During a period from the beginning of reset until the TXRST or RXRST value is read as 0, do not modify the control register or access the FIFOs.

The MSIOF can reset the transmit and receive units by setting the following bits to 1.

- Transmit reset: TXRST bit in SICTR
- Receive reset: RXRST bit in SICTR

Table 54.9 Transmit and Receive Reset

| Type | Objects Initialized |
|----------------|--|
| Transmit reset | SITFDR (Transmit FIFO write pointer) TDREQ bit in SISTR TXE bit in SICTR |
| | (When SITMDR1.PCON = 1, Receive reset is also asserted.) |
| Receive reset | SIRFDR (Receive FIFO read pointer) RDREQ bit in SISTR RXE bit in SICTR |

(6) Initial operating mode

After a power-on reset, both transmit and receive units are initialized to master mode (0 is output through MSIOF_SCK and MSIOF_SYNC). When using slave mode, keep the connected device outputting 0 until the operating mode setting is completed.

54.3.7 Interrupts

The MSIOF has one type of interrupt.

(1) Interrupt sources

Interrupts can be generated by several sources. Each source is shown as an MSIOF status in SISTR. Table 54.10 lists the MSIOF interrupt sources.

Table 54.10 MSIOF Interrupt Sources

| No. | Classification | Bit Name | Function Name | Description |
|-----|----------------|----------|--------------------------------|--|
| 1 | Transmission | TDREQ | Transmit FIFO transfer request | The transmit FIFO has empty space of specified size or more. |
| 2 | | TFEMP | Transmit FIFO empty | The transmit FIFO is empty. |
| 3 | | TEOF | Frame transmission end | Transmission of data with a length of one frame is completed. |
| 4 | Reception | RDREQ | Receive FIFO transfer request | The receive FIFO stores data of specified size or more. |
| 5 | | RFFUL | Receive FIFO full | The receive FIFO is full. |
| 6 | | REOF | Frame reception end | Reception of data with a length of one frame is completed. |
| 7 | Error | TFUDF | Transmit FIFO underflow | Serial data transmit timing has arrived while the transmit FIFO is empty. |
| 8 | | TFOVF | Transmit FIFO overflow | The transmit FIFO is written to while it is full. |
| 9 | | RFOVF | Receive FIFO overflow | Serial data is received while the receive FIFO is full. |
| 10 | | RFUDF | Receive FIFO underflow | The receive FIFO is read while the receive FIFO is empty. |
| 11 | | TFSEERR | Transmit FS error | A transmit frame synchronization signal is input before the specified number of bits has been reached (in slave mode). |
| 12 | | RFSERR | Receive FS error | A receive frame synchronization signal is input before the specified number of bits has been reached (in slave mode). |

Whether an interrupt is issued from an interrupt source is determined by the SIIER settings. If an interrupt source bit is set to 1 and the corresponding bit in SIIER is set to 1, an MSIOF interrupt is issued.

(2) TDREQ and RDREQ

The transmit FIFO transfer request (TDREQ) and receive FIFO transfer request (RDREQ) are signals indicating the state. If the state changes after TDREQ or RDREQ is set, it is automatically cleared by the MSIOF.

When the DMA transfer is used, a DMA transfer request signal is pulled low for one cycle at the end of DMA transfer.

(3) Processing when errors occur

- Transmit FIFO underflow (TFUDF)
An undefined value is output.
- Transmit FIFO overflow (TFOVF)
The contents of the transmit FIFO are protected, and the write operation causing the overflow is ignored.
- Receive FIFO overflow (RFOVF)
Data causing the overflow is discarded and lost.
- Receive FIFO underflow (RFUDF)
An undefined value is output on the bus.
- Transmit FS error (TFSERR)
The internal counter is reset according to the synchronization signal in which an error occurs.
- Receive FS error (RFSERR)
The internal counter is reset according to the synchronization signal in which an error occurs.

54.3.8 Transmit and Receive Timing

Examples of the MSIOF serial transmission and reception are shown in Figure 54.16 and Figure 54.17.

(1) 16-bit synchronization pulse

Synchronization pulse method, one group, one word, 16 bits in a word, no bit delay

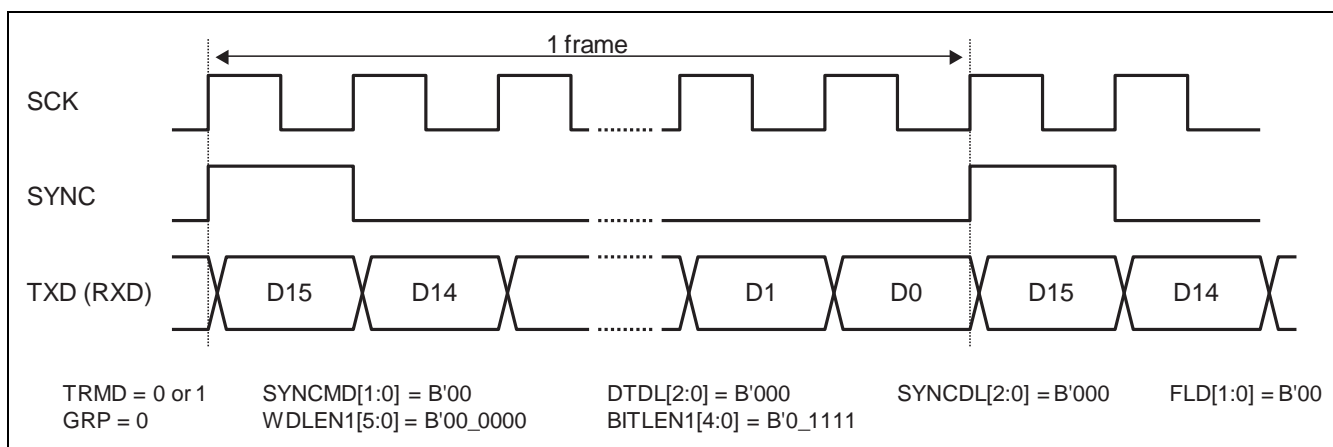


Figure 54.16 Transmit and Receive Timing (16 Bits)

(2) 32-bit synchronization pulse

Synchronization pulse method, one group, 32 words, 32 bits in a word, 1-bit delay

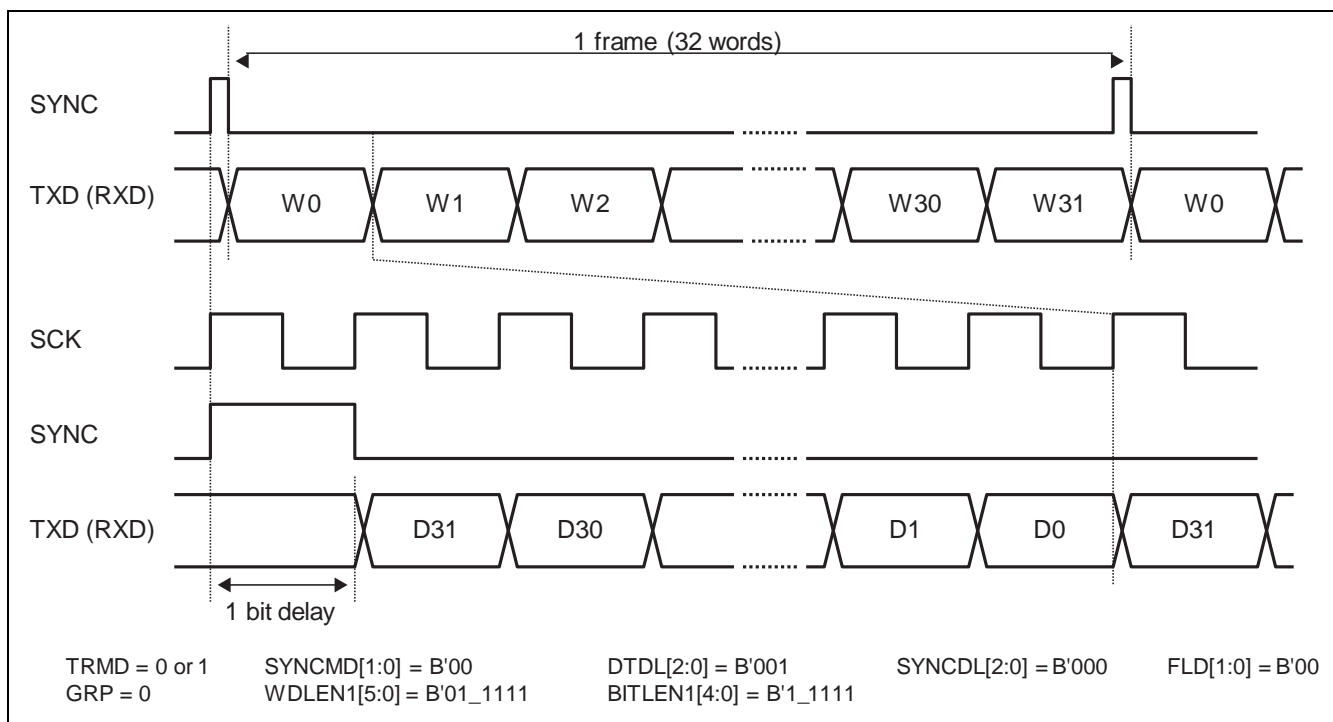


Figure 54.17 Transmit and Receive Timing (32 Bits)

(3) 32-bit IIS transmission

L/R mode, 2 groups, 1 word for each group, 32-bit word for each group, 1-bit delay

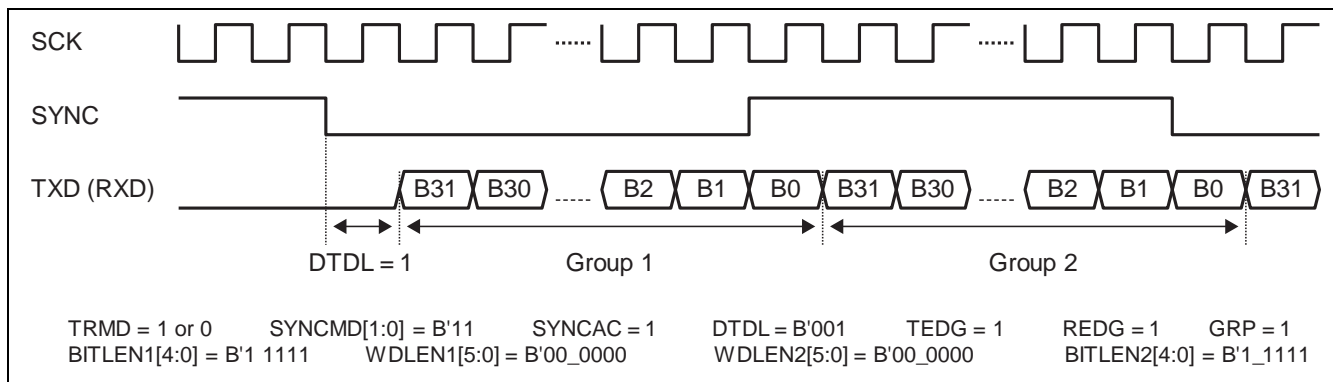


Figure 54.18 Transmit Timing (24-Bit IIS)

54.3.9 SPI

With the appropriate register settings, the MSIOF can be used as an SPI device.

(1) Example of SPI device connection

Figure 54.19 shows an example of connection with an SPI device.

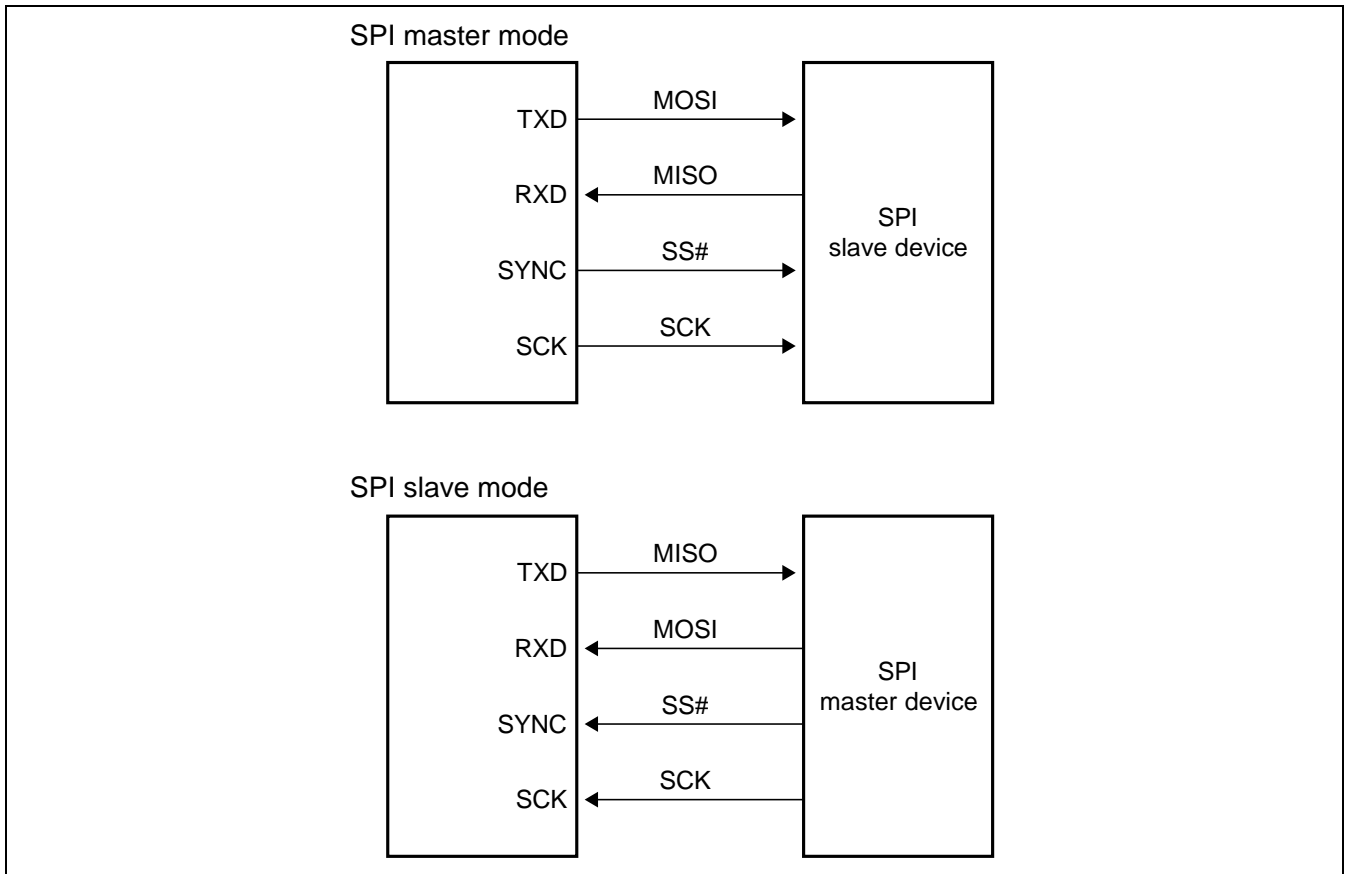


Figure 54.19 Example of SPI Device Connection

(2) SPI serial clock timing

Figure 54.20 and Figure 54.21 show the data and clock timing in SPI mode. As shown in the figures, four types of serial transfer formats can be used.

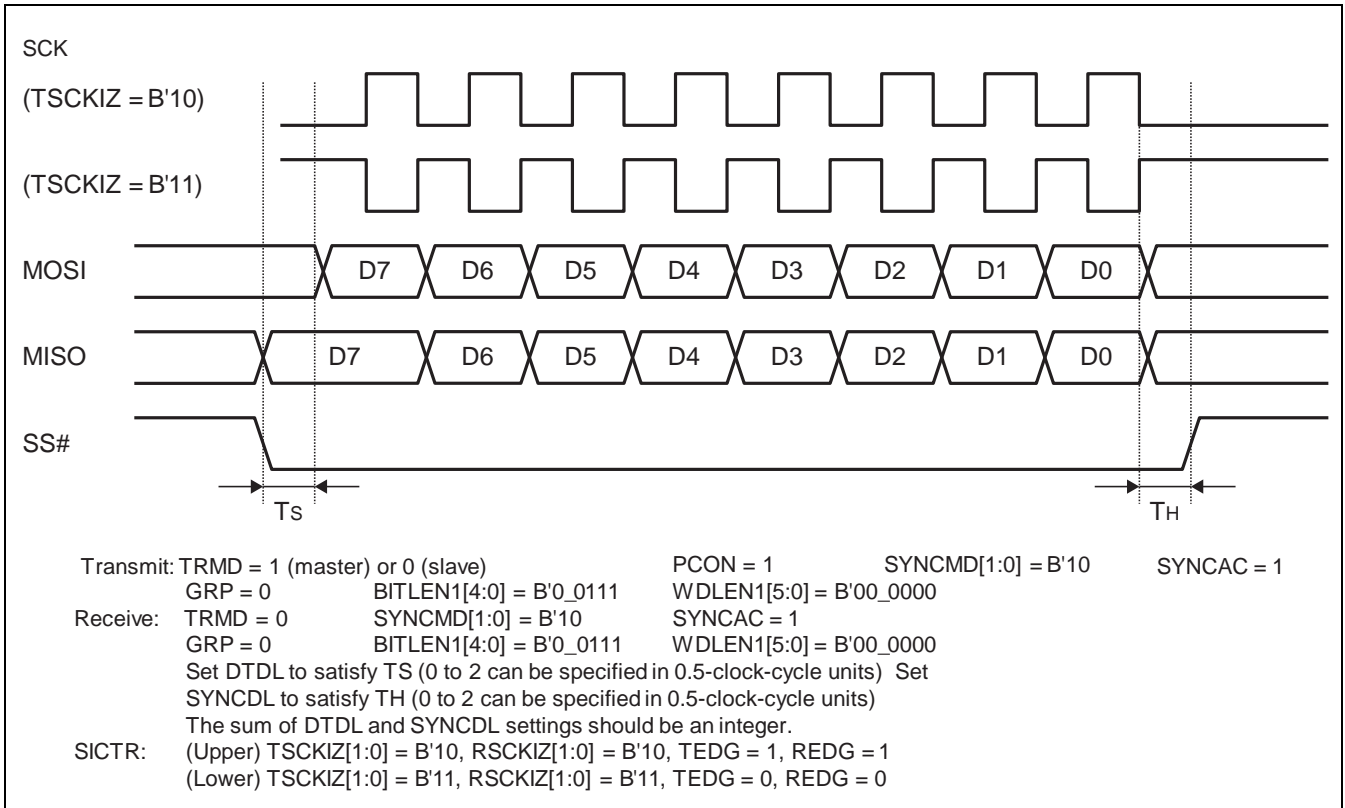


Figure 54.20 SPI Clock and Data Timing 1

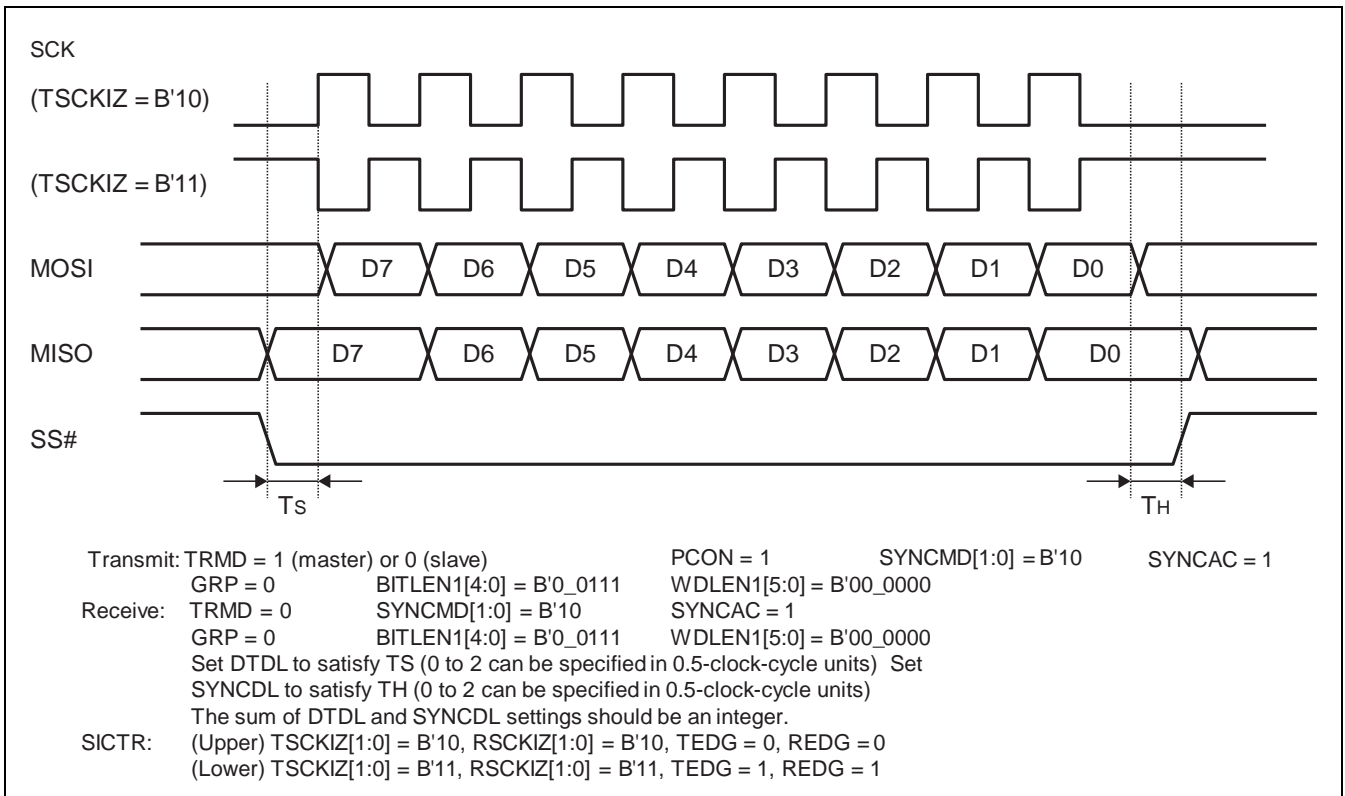


Figure 54.21 SPI Clock and Data Timing 2

54.3.10 IIS

This module can support IIS format with following setting which is shown in Figure 54.22 and Figure 54.23. BITLEN1 and BITLEN2 bits will change by usage.

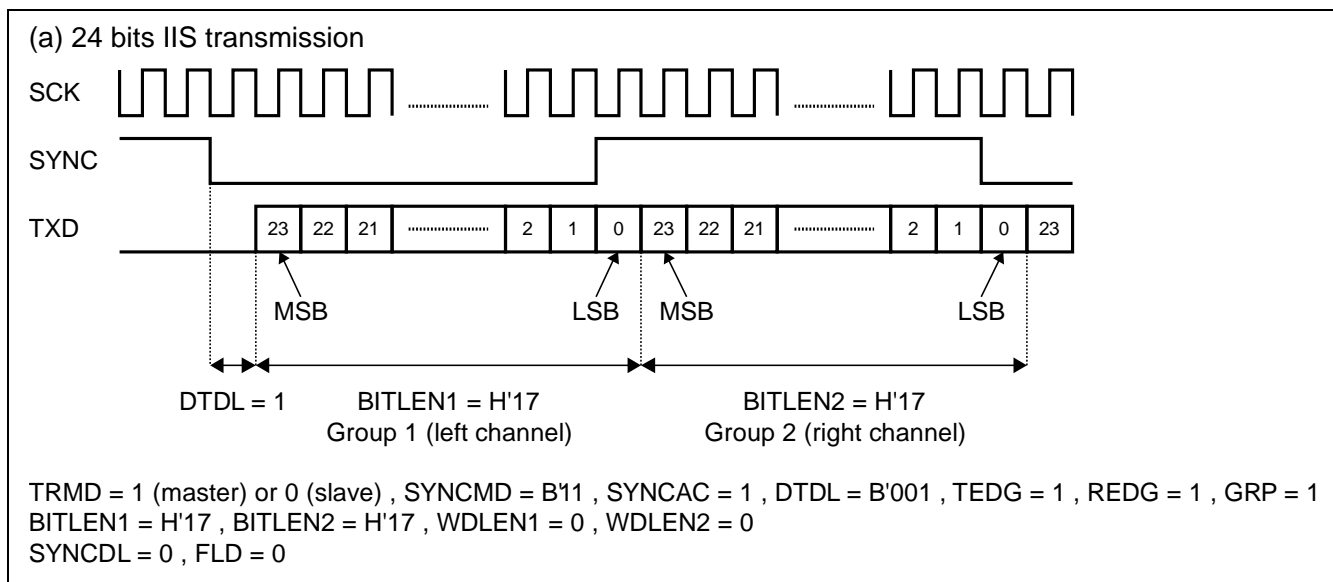


Figure 54.22 24 Bits IIS Transmission

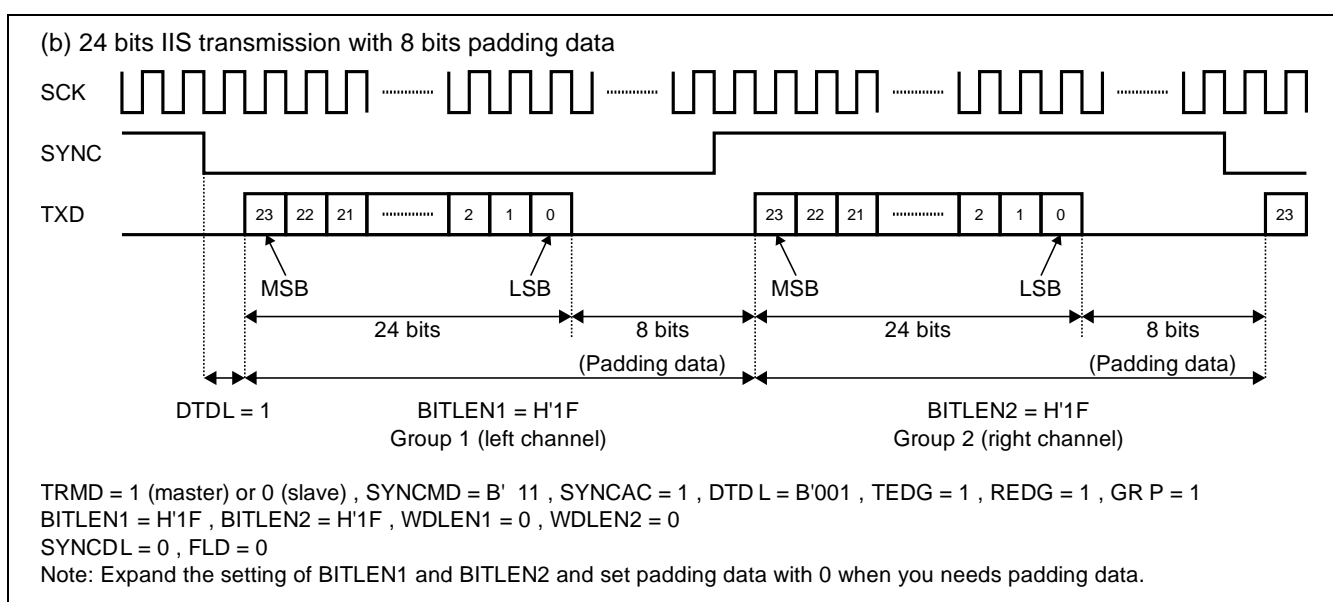


Figure 54.23 24 Bits IIS Transmission with 8 Bits Padding Data

54.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

54.4.1 Notes for Using SPI Mode

- For SPI master mode operation, set the FLD bits in SITMDR1 for at least 3 cycles of module clock.
- For SPI slave mode operation, set the frame synchronization signal interval to at least 4 cycles of module clock.
- For SPI slave mode operation, the setting for data delay should be always set to 0.
- For SPI slave mode operation, TXDIZ bits work on SS signal invalid state.

54.4.2 Notes for Using SPI slave Mode

1. In case of SPI slave mode, this module can't receive any data from external device, before SICTR [8]. RXE is set to 1. Prepare scheme to guarantee data will arrive only after SICTR [8]. RXE is set to 1, by higher level arbitration.
2. In case of SPI slave mode, this module needs to receive assert of SYNC signal to start receiving data. For this purpose, SYNC signal from master device must be connected to SYNC port, even in the environment, in which only two devices (master and slave) exist. Also, SYNC signal must be asserted after SICTR [8]. RXE bit is set. SYNC signal assertion while SICTR [8]. RXE is 0, received data can't guarantee.
3. When the MSIOF operates with all the following condition 1 to 3, the MSIOF in SPI slave mode cannot start the next transmission. The MSIOF cannot support SPI mode under following condition.
 - Condition
 - 1) MSIOF is in SPI slave mode (SITMDR1.TRMD = 0, SITMDR1.SYNCMD[1:0] = 10)
 - 2) SYNC (CS) signal is being asserted from master
 - 3) TXE or RXE in SICTR is set to 1 during SYNC (CS) is asserted

This limitation means the MSIOF cannot respond to the SYNC (CS).

It is necessary to control the MSIOF enabling timing by the master and system.

[Workaround]

Use master mode for SPI.

If use slave mode for SPI, consider following alternative.

[Alternative]

If the master device can support handshaking, use the handshaking transmission.

For the MSIOF in SPI slave mode, use with the GPIO output for SPI handshaking; the system operation must be tested and evaluated thoroughly by user.

4. In case of SPI slave mode, this module must not set SICTR[1].TXRST or SICTR[0].RXRST during SYNC signal asserted.

54.4.3 Notes for Transmitting Audio data via MSIOF

If you want to transmit audio data via MSIOF, don't set the master clock (MSO ϕ) to SSCG.

54.4.4 Notes on Fixed Value related to MSIOF registers

It is necessary to change from the initial value to the specific fixed value for these bits in the MSIOF registers. Setting other value is prohibited.

Table 54.11 Fixed Value for the MSIOF Registers without any condition

| Section | Bit Number | Fixed Value |
|---|------------|-------------|
| 54.2.1 MSIOF Transmit Mode Register 1 (SITMDR1) | 30 | B'1 |

54.4.5 Notes for Transmit FIFO Underflow (SISTR.TFUDF)

If transmission size is less than or equal to 64-words per frame, transmit FIFO Underflow (SISTR.TFUDF) needs to be ignored. If transmission size is larger than 64-words per frame, use Transmit FIFO Empty (SISTR.TFEMP) instead of this bit. To skip Empty (SISTR.TFEMP) state after the transmission completion, write 1-word dummy data to the transmit FIFO and perform reset after the transferring. If the next transfer data can be prepared, it can be possible to store next data instead of dummy data. In that case, the reset process can be skipped. If Empty is detected during transfer, reset after the stop processing of each module completed, and redo initialization.

The following is an example flow when Transmit FIFO Empty (SISTR.TFEMP) interrupt is detected.

1. SISTR.TFEMP interrupt detection
2. Clear interrupt enable of MSIOF
3. Wait for SISTR.TEOF/REOF=1 state of MSIOF
4. Stop processing of DMAC (After DMACHCR.DE=0 at DMAC, wait for DMACHCR.DE=0)
5. DMAC channel reset (DMACHCLR.CLR[m]=1 at DMAC, m=usage channel)
6. Change PFC setting (MSIOF_SCK, MSIOF_SYNC) to GPIO(In) side
7. Assert SW reset for activating MSIOF channels in use (SRCLR2.MSIOF)
8. De-assert SW reset for using MSIOF channels in use (SRSTCLR2.MSIOF) (Regarding procedure of SW reset, please refer to the sec 8A. Module Standby, Software Reset)
9. Reconfigure MSIOF and DMAC When using PIO access, the No4 and No5 and the reconfiguration of DMAC in No9 are unnecessary. (The reconfiguration of MSIOF in No9 is necessary.)

Note: DMAC is SYS-DMAC

55. PWM

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

55.1 Overview

This LSI incorporates seven channels for , RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E;

Internal bus clock:

- RZ/G2H: S0D12 ϕ
- RZ/G2M V1.3: S0D12 ϕ
- RZ/G2M V3.0: S0D12 ϕ
- RZ/G2N: S0D12 ϕ
- RZ/G2E: S3D4C ϕ

55.1.1 Features

- PWM output cycle settable (10 bits)
- PWM output cycle settable within the range from 2 cycles to $2^{24} \times 1023$ cycles of internal bus clock (i.e. from 30 ns to 257.5 seconds when internal bus clock = 66.66 MHz)
- High-level width of the PWM output signal settable (10 bits)
- Continuous pulse output or single pulse output selectable

55.1.2 Block Diagram

Figure 55.1 shows a block diagram of the PWM timer.

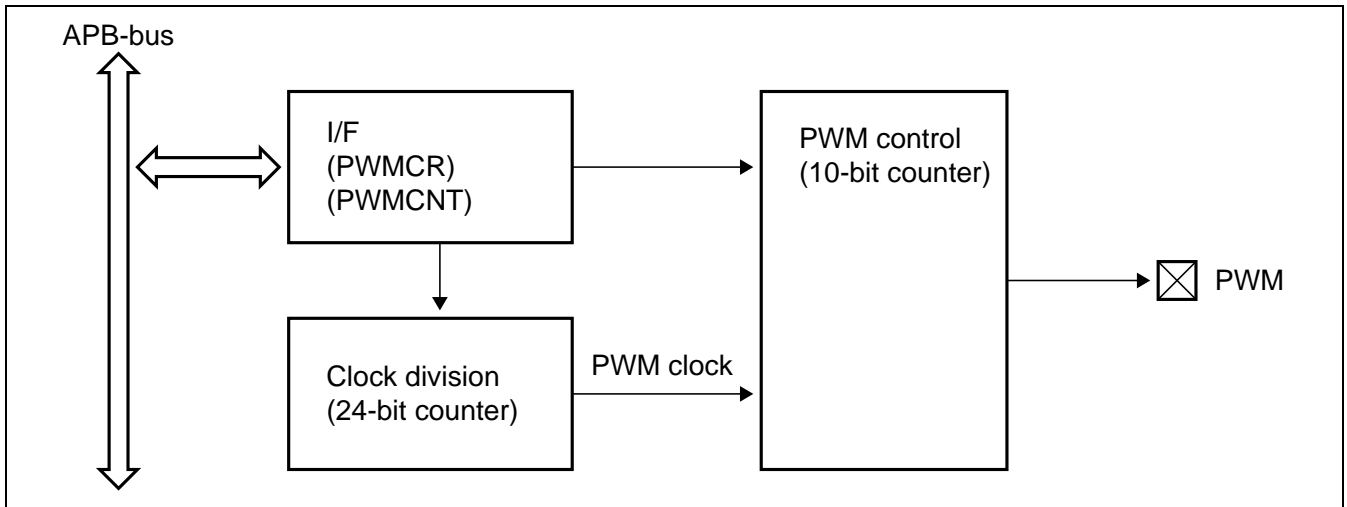


Figure 55.1 Block Diagram of PWM Timer

55.1.3 External Pins

Table 55.1 shows the pin configuration of the PWM timer.

Table 55.1 Pin Configuration

| Pin Name | I/O | Function | Second Generation RZ/G Series Products | | | |
|----------|--------|-------------------------|---|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| PWM0 | Output | PWM0 timer pulse output | √ | √ | √ | √ |
| PWM1 | Output | PWM1 timer pulse output | √ | √ | √ | √ |
| PWM2 | Output | PWM2 timer pulse output | √ | √ | √ | √ |
| PWM3 | Output | PWM3 timer pulse output | √ | √ | √ | √ |
| PWM4 | Output | PWM4 timer pulse output | √ | √ | √ | √ |
| PWM5 | Output | PWM5 timer pulse output | √ | √ | √ | √ |
| PWM6 | Output | PWM6 timer pulse output | √ | √ | √ | √ |

55.1.4 Register Configuration

Table 55.2 lists the registers of the PWM timer. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined. Access the register as a longword (32 bits). Operation cannot be guaranteed if the register is not accessed as a longword.

Table 55.2 List of Registers

| Channel | Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | | |
|---------|----------------------|--------------|-----|-------------|-------------|---|----------------------------|--------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | RZ/G2E |
| 0 | PWM control register | PWMCR | R/W | H'E6E3_0000 | 32 | √ | √ | √ | √ | |
| | PWM count register | PWMCNT | R/W | H'E6E3_0004 | 32 | √ | √ | √ | √ | |
| 1 | PWM control register | PWMCR | R/W | H'E6E3_1000 | 32 | √ | √ | √ | √ | |
| | PWM count register | PWMCNT | R/W | H'E6E3_1004 | 32 | √ | √ | √ | √ | |
| 2 | PWM control register | PWMCR | R/W | H'E6E3_2000 | 32 | √ | √ | √ | √ | |
| | PWM count register | PWMCNT | R/W | H'E6E3_2004 | 32 | √ | √ | √ | √ | |
| 3 | PWM control register | PWMCR | R/W | H'E6E3_3000 | 32 | √ | √ | √ | √ | |
| | PWM count register | PWMCNT | R/W | H'E6E3_3004 | 32 | √ | √ | √ | √ | |
| 4 | PWM control register | PWMCR | R/W | H'E6E3_4000 | 32 | √ | √ | √ | √ | |
| | PWM count register | PWMCNT | R/W | H'E6E3_4004 | 32 | √ | √ | √ | √ | |
| 5 | PWM control register | PWMCR | R/W | H'E6E3_5000 | 32 | √ | √ | √ | √ | |
| | PWM count register | PWMCNT | R/W | H'E6E3_5004 | 32 | √ | √ | √ | √ | |
| 6 | PWM control register | PWMCR | R/W | H'E6E3_6000 | 32 | √ | √ | √ | √ | |
| | PWM count register | PWMCNT | R/W | H'E6E3_6004 | 32 | √ | √ | √ | √ | |

Note: Do not access addresses other than those listed above. If access is attempted, a malfunction may occur.

Table 55.3 Register States in Each Operating Mode

| Name | Abbreviation | Power-on Reset/Software Reset | Module Standby |
|----------------------|--------------|-------------------------------|----------------|
| PWM control register | PWMCR | H'0000_0000 | Retained |
| PWM count register | PWMCNT | H'0000_0000 | Retained |

The registers of the PWM timer are mapped into the internal bus address space.

55.2 Register Description

Registers in the PWM timer are allocated to and arranged in the address space of the internal bus.

Legend for Register Description

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

R: Read-only. When the bit is reserved, the write value should always be 0. Writing 1 to these bits can cause a malfunction of PWM.

—/W: Write-only. The read value is undefined.

55.2.1 PWM Control Register (PWMCr)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------|----|----|----|------|----|----|----|----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | CC0 | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CCMD | — | — | — | SYNC | — | — | — | — | — | — | SS0 | — | — | — | EN0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R/W | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 20 | — | All 0 | R | Reserved The read value is always 0. The write value should always be 0. |
| 19 to 16 | CC0 | B'0000 | R/W | <p>Clock Control</p> <p>The clock frequency of the PWM counter is obtained by dividing the frequency of the clock signal of internal bus clock.</p> <p>When the internal bus clock frequency is f MHz, the following PWM clock frequencies are obtained.</p> <p>B'0000: f MHz B'0001: f MHz/2² B'0010: f MHz/2⁴ B'0011: f MHz/2⁶ B'0100: f MHz/2⁸ B'0101: f MHz/2¹⁰ B'0110: f MHz/2¹² B'0111: f MHz/2¹⁴ B'1000: f MHz/2¹⁶ B'1001: f MHz/2¹⁸ B'1010: f MHz/2²⁰ B'1011: f MHz/2²² B'1100: f MHz/2²⁴</p> <p>When B'1101, B'1110, or B'1111 is set, the PWM clock frequency is f MHz/2²⁴.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 15 | CCMD | B'0 | R/W | <p>CC0 Frequency Division Mode</p> <p>Changes the PWM clock frequency set by the clock control (CC0) bits.</p> <p>0: The PWM clock frequency set by CC0 is used.</p> <p>1: The PWM clock frequency set by CC0 is changed as follows:</p> <ul style="list-style-type: none"> 1 - B'0000: $f \text{ MHz}/2$ 1 - B'0001: $f \text{ MHz}/2^3$ 1 - B'0010: $f \text{ MHz}/2^5$ 1 - B'0011: $f \text{ MHz}/2^7$ 1 - B'0100: $f \text{ MHz}/2^9$ 1 - B'0101: $f \text{ MHz}/2^{11}$ 1 - B'0110: $f \text{ MHz}/2^{13}$ 1 - B'0111: $f \text{ MHz}/2^{15}$ 1 - B'1000: $f \text{ MHz}/2^{17}$ 1 - B'1001: $f \text{ MHz}/2^{19}$ 1 - B'1010: $f \text{ MHz}/2^{21}$ 1 - B'1011: $f \text{ MHz}/2^{23}$ <p>If B'1100, B'1101, B'1110, or B'1111 is set for CC0, the PWM clock frequency is $f \text{ MHz}/2^{24}$ irrespective of the CCMD value.</p> |
| 14 to 12 | — | All 0 | R | <p>Reserved</p> <p>The read value is always 0. The write value should always be 0.</p> |
| 11 | SYNC | B'0 | R/W | <p>Specifies whether to allow the set values in the PWM count register (PWMCNT) to be reflected in the timer operation synchronously with setting the PWM control register (PWMCR).</p> <p>0: Allows the PWMCNT set values to be reflected in the timer operation irrespective of PWMCR setting.</p> <p>1: Allows the PWMCNT set values to be reflected in the timer operation synchronously with PWMCR setting.</p> <p>For details, refer to section 55.3 Operation.</p> |
| 10 to 7 | — | All 0 | R | <p>Reserved</p> <p>The read value is always 0. The write value should always be 0.</p> |
| 6, 5 | — | All 0 | R/W | <p>Reserved</p> <p>The read value is always 0. The write value should always be 0.</p> |
| 4 | SS0 | B'0 | R/W | <p>Single Pulse Output</p> <p>0: The timer operates in continuous pulse output mode.</p> <p>1: The timer operates only for a single cycle and then stops.</p> |
| 3 | — | B'0 | R/W | <p>Reserved</p> <p>The read value is always 0. The write value should always be 0.</p> |
| 2, 1 | — | All 0 | R | <p>Reserved</p> <p>The read value is always 0. The write value should always be 0.</p> |
| 0 | EN0 | B'0 | R/W | <p>Channel Enable</p> <p>0: The channel is held in the idle state, and outputs a high level.</p> <p>1: The channel outputs high and low levels in a predetermined cycle.</p> <p>This bit is automatically cleared in single pulse output mode.</p> |

55.2.2 PWM Count Register (PWMCNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | | |
| | — | — | — | — | — | — | CYC0 | | | | | | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | |
| | — | — | — | — | — | — | PH0 | | | | | | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 26 | — | All 0 | R | Reserved The read value is always 0. The write value should always be 0. |
| 25 to 16 | CYC0 | H'000 | R/W | PWM Cycle Sets the PWM output cycle. The cycle set by these bits is the sum of the high-level and low-level periods. Setting H'000 is prohibited. |
| 15 to 10 | — | All 0 | R | Reserved The read value is always 0. The write value should always be 0. |
| 9 to 0 | PH0 | H'000 | R/W | PWM High-Level Period Sets the period for which the PWM timer outputs a high-level signal. Setting H'000 is prohibited. |

Note: For the CYC0 and PH0 bits, set the number of cycles to be counted of the PWM clock signal whose frequency has been set by the CC0 and CCMD bits in the PWM control register.

55.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The duty ratio of a PWM output pulse can be obtained by setting a high-level period and a cycle. The timer channel counts the PWM clock signal pulses using a 10-bit counter to generate the PWM output pulse having the specified period and cycle. When the channel function is enabled ($EN0 = 1$), the timer outputs a high level until the counter value reaches the value set in the PH0 bits of the PWM count register (PWMCNT). The output goes low when the PH0 value is reached, and is held low until the counter value reaches the value set in the CYC0 bits of PWMCNT. When the CYC0 value is reached, the output goes high and the counter is reset.

Unless the channel function is enabled, the output is held high, and the counter is held in the reset state.

Figure 55.2 shows the PWM state transition during the operation described above.

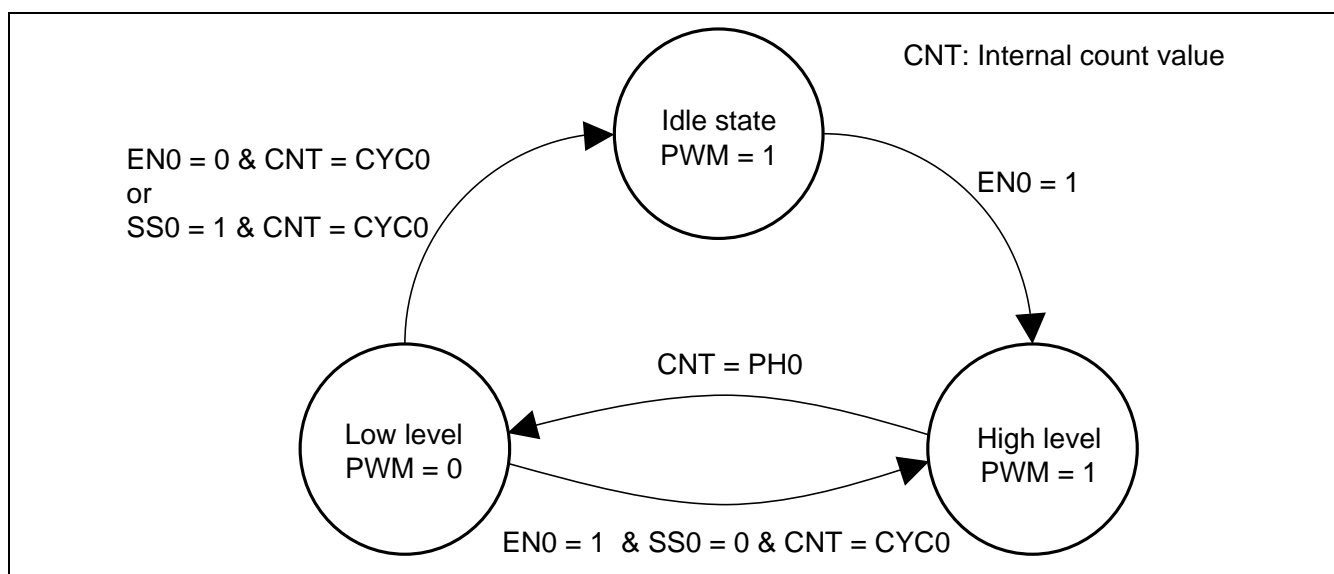


Figure 55.2 PWM Timer State Transition

Note that after a channel has been enabled by writing 1 to EN0, the number of clock pulses before output of the first falling edge from a channel is greater than the value set in the PWM control register and PWM count register by up to four cycles of the internal bus clock. After that, the PWM output rises and falls according to a set timing. Figure 55.3 schematically shows this output timing. This figure shows an internal operation.

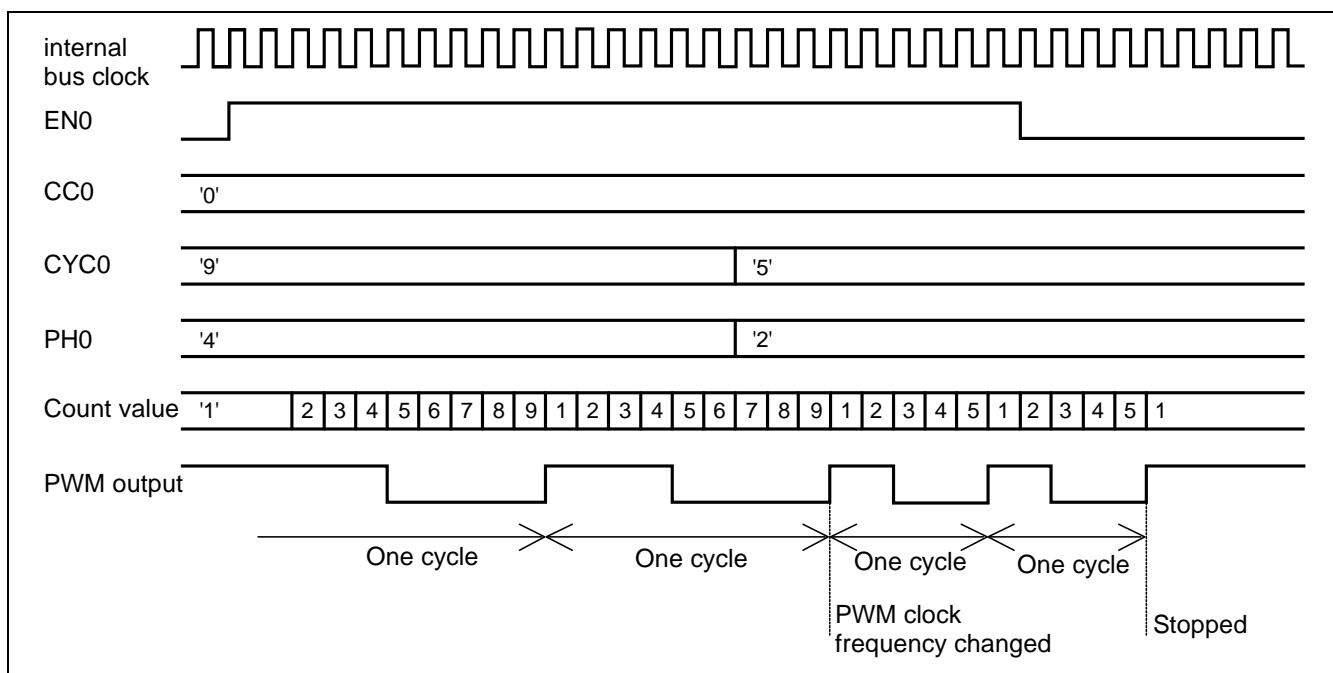


Figure 55.3 PWM Output Timing

1. The period from enabling the channel to the first falling edge output from the channel is the sum of a set value and up to four cycles of the internal bus clock.
2. After that, the PWM high and low widths are the same as the set values.
3. When the PWM count register is changed during operation ($EN0 = 1$), timing of the output will be in accord with the new settings after output of the current cycle has been completed.
4. When the CC0 and CCMD bit values in the PWM control register are changed during operation ($EN0 = 1$), timing of the output will be in accord with the new settings after output of the current cycle has been completed.
5. To stop the timer operation, write 0 to the EN0 bit. The timer stops operation after completing the output of the current cycle. Note that, however, if 1 is written to EN0 again before completing the output of the current cycle, the timer does not stop but continues to operate.
6. For the PWM timer output, continuous pulse output mode and single pulse output mode are supported. In continuous pulse output mode, the timer outputs pulses continuously while EN0 is 1. In single pulse output mode, the timer outputs a single pulse. In this mode, EN0 is automatically cleared.
7. To change the values of the clock control bits (CC0) and the CC0 frequency division mode bit (CCMD) in the PWMCR register and of the PWM cycle bits (CYC0) and the PWM high-level period bits (PH0) in the PWMCNT register at the same time during the PWM operation, the following procedure should be followed:
 1. Write 1 to the SYNC bit. (Write the original set values to the other bits in PWMCR.)
 2. Change the PWMCNT value.
 3. Change the CC0 and CCMD values in PWMCR. (Write the original set values to the other bits in PWMCR. The SYNC bit should be held at 1.)

By following the above procedure, the new PWMCNT value is reflected in the timer operation synchronously with PWMCR setting.

When changing only the PWMCNT value while SYNC = 1, writing to PWMCR is required after changing the PWMCNT value. In this case, write the same values as the original set values to the PWMCR bits. On the other hand, when changing only the PWMCR value, writing to PWMCNT is not required.

If PWMCNT and PWMCR values are changed while SYNC = 0, the change to the values might not be completed within a single PWM output cycle but actually take two or more cycles. This may disturb the period or duty cycle of the PWM output.

Note that, when the PWCNT value is changed while SYNC = 1, the new values are read from PWMCNT even before PWMCR is set.

55.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

When using the PWM timer, note the following points.

When the CYC0 value is equal to or smaller than PH0 value in the PWM count register, the PWM output is held high. In this case, to start pulse output, clear EN0 to 0, set CYC0 to the value greater than the PH0 value, and then set EN0 to 1.

Note that, a maximum period equivalent to CYC0 = H'3FF at the set PWM clock frequency may be required before pulse output is started.

56. SPI Multi I/O Bus Controller (RPC-IF)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

56.1 Overview

The SPI multi I/O bus controller enables direct connection of the serial flash memory, HyperFlash, or Octal-SPI Flash memory to this LSI.

56.1.1 Features

This module allows the serial flash memory or HyperFlash memory connected to this LSI to be accessed by the external address space read mode or manual mode.

The external address space read mode is able to read from Flash memory only. The manual mode is able to write to and read from Flash memory.

(1) Serial Flash Memory Interface

- Up to two serial flash memory devices can be connected. Note that two serial flash memory devices are operated in parallel interface and each 1xQSPI operation is not supported.
- When 1xQSPI configuration, QSPI0 pins in this LSI are connected to the serial flash memory.
- A data bus size of 1 bit or 4 bits can be selected for a serial flash memory device.
- Support single data rate (SDR) and Double data rate (DDR) transfer
- Only the master operation is supported (the slave operation is not supported).
- Support Octal-SPI flash interface by using 2ch mode with 4bit bus size

Note: “Octal-SPI flash” indicates 8-bit SPI flash with one ChipSelect, one Clock source and one DataStrobe configuration. Serial flash memory specification without “8-bit” description specifies general SPI flash interface (including SPI compatible mode) in this manual.

Note: The RZ/G2M V1.3 (not RZ/G2M V3.0) does not support DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit).

Note that the RZ/G2M V1.3 (not RZ/G2M V3.0) supports the Hyperflash and Octal-SPI memory devices even though it requires DDR operation.

(2) HyperFlash Interface

- One HyperFlash can be connected.
- The data bus size is fixed to 8 bits.
- Wrapped Burst is not supported.

(3) External Address Space Read Mode

- A read access from the bus master to the SPI multi I/O bus space is automatically converted into a read command to the device and the read data are returned to the bus master.
- Normal read and burst read operation
- Efficient data reception is possible by utilizing the on-chip read cache memory (64-bit line × 32 entries) during burst read operation.

(4) Manual Mode

- Desired command, read, and write accesses to the serial flash memory or HyperFlash memory are possible.
- Efficient data writing is possible by using write buffer.
- Not support the on-chip read cache memory during read operation.

56.1.2 Block Diagram

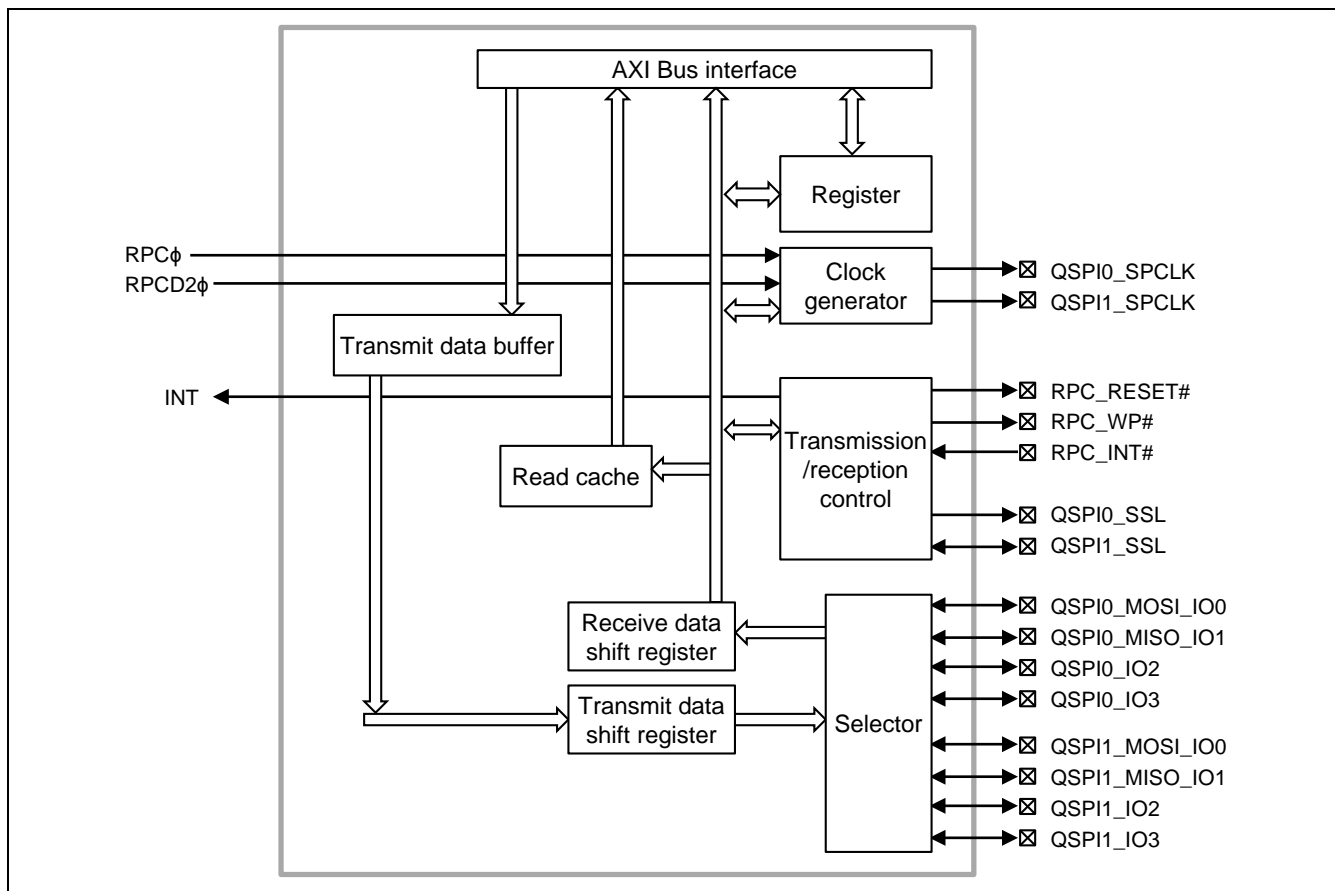


Figure 56.1 RPC-IF Block Diagram

56.1.3 External Pins

Table 56.1 shows the pin configuration of the RPC-IF.

Table 56.1 Pin Name

| Pin Name | Symbol | I/O | Function | | Second Generation RZ/G Series Products | | | |
|---|----------------|--------|--------------------------------|------------------------------|--|----------------------------|--------|--------|
| | | | When serial flash is selected. | When HyperFlash is selected. | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Channel 0 clock pin*2 | QSPIO_SPCLK | Output | Clock output | Differential clock output + | √ | √ | √ | √ |
| Channel 0 master transmit data/data 0 | QSPIO_MOSI_IO0 | I/O | Master transmit data/data 0 | Data 0 | √ | √ | √ | √ |
| Channel 0 slave transmit data/data 1 | QSPIO_MISO_IO1 | I/O | Slave transmit data/data 1 | Data 1 | √ | √ | √ | √ |
| Channel 0 data 2 pin | QSPIO_IO2 | I/O | Data 2 | Data 2 | √ | √ | √ | √ |
| Channel 0 data 3 pin | QSPIO_IO3 | I/O | Data 3 | Data 3 | √ | √ | √ | √ |
| Channel 0 slave select pin | QSPIO_SSL | Output | Slave selection | Chip selection | √ | √ | √ | √ |
| Channel 1 clock pin*1,*2,*4 | QSPI1_SPCLK | Output | Clock output | Differential clock output - | √ | √ | √ | √ |
| Channel 1 master transmit data/data 0 pin | QSPI1_MOSI_IO0 | I/O | Master transmit data/data 0 | Data 4 | √ | √ | √ | √ |
| Channel 1 slave transmit data/data 1 pin | QSPI1_MISO_IO1 | I/O | Slave transmit data/data 1 | Data 5 | √ | √ | √ | √ |
| Channel 1 data 2 pin | QSPI1_IO2 | I/O | Data 2 | Data 6 | √ | √ | √ | √ |
| Channel 1 data 3 pin | QSPI1_IO3 | I/O | Data 3 | Data 7 | √ | √ | √ | √ |
| Channel 1 slave select pin*1,*4 | QSPI1_SSL | I/O | Slave selection or Data strobe | Read data strobe (RDS) | √ | √ | √ | √ |
| Flash-RESET pin*1,*3 | RPC_RESET# | Output | Reset signal output | Reset signal output | √ | √ | √ | √ |
| Write protect pin*1 | RPC_WP# | Output | Write protect (WP) | Write protect (WP) | √ | √ | √ | — |
| INT pin*1 | RPC_INT# | Input | Interrupt input | Interrupt input | √ | √ | √ | √ |

Notes: When a serial flash memory is to be connected with BSZ[1:0] being set to 00, the serial flash device should be connected to the QSPIO_* pins.

In the text, the channel number is omitted (e.g. QSPIn_SSL).

These pins are 1.8V operation only [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

These pins are 1.8/3.3V operation selectable [RZ/G2E]

1. These pins should be connected depending on the serial flash specification.

2. The frequency of QSPIn_SPCLK is same as RPCD2φ.

3. RPC_RESET# cannot be used with Serial Flash boot. Instead, connect PRESETOUT# to the Serial Flash RESET# terminal. Also, when using 1.8V serial flash, step down voltage from 3.3V to 1.8V. (3.3V serial flash is unnecessary.) [RZ/G2E]

4. When MD[4:1]=B'0010/B'0011(dma HyperFlash), B'1010/B'1011(xip HyperFlash),

- During RPC_RESET# is low level, QSPI1_SPCLK is low output and QSPI1_SSL is high output.

- During RPC_RESET# is high level, QSPI1_SPCLK is high output and QSPI1_SSL is Hi-Z.

56.1.4 Register Configuration

Table 56.2 shows the register configuration of the RPC-IF.

Table 56.2 Register Configuration

| Register Name | Abbreviation | R/W | Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | |
|---|--------------|-----|-------------|---------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Common control register | CMNCR | R/W | H'EE20_0000 | *1 | 32 | √ | √ | √ | √ |
| SSL delay register | SSLDR | R/W | H'EE20_0004 | H'0000_0400 | 32 | √ | √ | √ | √ |
| Data read control register | DRCR | R/W | H'EE20_000C | *1 | 32 | √ | √ | √ | √ |
| Data read command setting register | DRCMR | R/W | H'EE20_0010 | *1 | 32 | √ | √ | √ | √ |
| Data read extended address setting register | DREAR | R/W | H'EE20_0014 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Data read option setting register | DROPR | R/W | H'EE20_0018 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Data read enable setting register | DRENr | R/W | H'EE20_001C | *1 | 32 | √ | √ | √ | √ |
| Manual mode control register | SMCR | R/W | H'EE20_0020 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Manual mode command setting register | SMCMR | R/W | H'EE20_0024 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Manual mode address setting register | SMADR | R/W | H'EE20_0028 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Manual mode option setting register | SMOPR | R/W | H'EE20_002C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Manual mode enable setting register | SMENR | R/W | H'EE20_0030 | H'0000_4000 | 32 | √ | √ | √ | √ |
| Manual mode read data register 0 | SMRDR0 | R | H'EE20_0038 | Undefined | 8, 16, 32 | √ | √ | √ | √ |
| Manual mode read data register 1 | SMRDR1 | R | H'EE20_003C | Undefined | 8, 16, 32 | √ | √ | √ | √ |
| Manual mode write data register 0 | SMWDR0 | R/W | H'EE20_0040 | H'0000_0000 | 8, 16, 32 | √ | √ | √ | √ |
| Manual mode write data register 1 | SMWDR1 | R/W | H'EE20_0044 | H'0000_0000 | 8, 16, 32 | √ | √ | √ | √ |
| Common status register | CMNSR | R | H'EE20_0048 | H'0000_0001 | 32 | √ | √ | √ | √ |
| Data read dummy cycle setting register | DRDMCR | R/W | H'EE20_0058 | *1 | 32 | √ | √ | √ | √ |
| Data read DDR enable register | DRDRENr | R/W | H'EE20_005C | *1 | 32 | √ | √ | √ | √ |

| | | | | | | Second Generation RZ/G Series Products | | | |
|--|----------------|-----|------------------------|-------------------|-------------|--|----------------------------|--------|--------|
| Register Name | Abbreviation | R/W | Address | Initial value | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Manual mode dummy cycle setting register | SMDMCR | R/W | H'EE20_0060 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Manual mode DDR enable register | SMDRENR | R/W | H'EE20_0064 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PHY Address Register | PHYADD | R/W | H'EE20_0070 | H'0000_0000 | 32 | — | — | — | √ |
| PHY Write Data Register | PHYWR | R/W | H'EE20_0074 | H'0000_0000 | 32 | — | — | — | √ |
| PHY Read Data Register | PHYRD | R/W | H'EE20_0078 | H'0000_0000 | 32 | — | — | — | √ |
| PHY control register | PHYCNT | R/W | H'EE20_007C | *1 | 32 | √ | √ | √ | √ |
| PHY offset register 1 | PHYOFFSET 1 | R/W | H'EE20_0080 | *1 | 32 | √ | √ | √ | √ |
| PHY offset register 2 | PHYOFFSET 2 | R/W | H'EE20_0084 | H'0000_0431 | 32 | √ | √ | √ | √ |
| PHY interrupt register | PHYINT | R/W | H'EE20_0088 | *1 | 32 | √ | √ | √ | √ |
| Address division register 1 | ADD_DIV1 | R/W | H'EE20_00AC | H'0000_0000 | 32 | √ | — | — | — |
| Address division register 2 | ADD_DIV2 | R/W | H'EE20_00B0 | H'0000_0000 | 32 | √ | — | — | — |
| Address division register 3 | ADD_DIV3 | R/W | H'EE20_00B4 | H'0000_0000 | 32 | √ | — | — | — |
| Secure configuration register | SEC_CONF | R/W | H'EE20_00B8 | H'0000_0100 *2 | 32 | √ | — | — | — |
| Access right register | ARIGHT | R/W | H'EE20_00BC | H'0000_0000 | 32 | √ | — | — | — |
| Secure command register n (n = 0 to 15) | SEC_CMDn | R/W | H'EE20_00C0 + H'4*n | H'0000_0000 | 32 | √ | — | — | — |
| Non-Secure command register n (n = 0 to 15) | NON_SEC_CMDn | R/W | H'EE20_0100 + H'4*n | H'0000_0000 | 32 | √ | — | — | — |
| Erase Command List n (n = 1, 2) | ERASELISTn | R/W | H'EE20_01E0 + H'4*n | H'0000_0000 | 32 | √ | — | — | — |
| Write Command List n (n = 1, 2) | WRITELISTn | R/W | H'EE20_01E8 + H'4*n | H'0000_0000 | 32 | √ | — | — | — |

Notes: 1. The initial value of this register changes depending on the boot mode. Refer to Table 56.3.

2. The setting value of this register changes to H'0000_0155 after boot operation.

Note: Do not write to any addresses other than listed above. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed above are undefined.

Table 56.3 shows the register initial value depending on the boot mode.

Table 56.3 Register initial value

| Register Name | Initial value | | | |
|---------------|--------------------------------------|-------------------------------------|-------------------------------------|---------------|
| | MD4-1 = 0100 (dma qspi single) | MD4-1 = 001* (dma HyperFlash) | MD4-1 = 101* (xip HyperFlash) | MD4-1 = other |
| CMNCR | H'0155_7300 | H'0155_7301 | H'0155_7301 | H'0155_7301 |
| DRCR | H'001F_0100 | H'001F_0100 | H'001F_0100 | H'001F_0101 |
| DRCMR | H'0003_0000 | H'00A0_0000 | H'00A0_0000 | H'0003_0000 |
| DRENDR | H'0000_4700 | H'A222_D400 | H'A222_D400 | H'0000_4700 |
| DRDMCR | H'0000_0000 | H'0000_000E | H'0000_000E | H'0000_0000 |
| DRDRENDR | H'0000_0000 | H'0000_5101 | H'0000_5101 | H'0000_0000 |
| PHYCNT | H'0007_0260 | H'0007_0263 | H'0003_0263 | H'0007_0260 |
| PHYOFFSET1 | H'3151_1144 | H'2151_1144 | H'2151_1144 | H'3151_1144 |
| PHYINT | H'0606_0002 | H'0707_0002 | H'0707_0002 | H'0606_0002 |

56.2 Register Description

The registers of the RPC-IF are assigned to and located in the address space of the internal bus.

[Legend for Register Descriptions]

Initial value: Register value after a reset.

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

R: Read-only. The write value should always be initial value. For details of initial value, refer to Table 56.3.

—/WB: Write-only. The read value is undefined.

56.2.1 Common Control Register (CMNCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMNCR is a 32-bit register that controls the SPI multi I/O bus controller. The settings of this register are reflected both in external address space read mode and manual operating mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|-----|----|----|----|----|----|-------|-----|--------|--------|--------|--------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MD | — | — | — | — | — | — | — | MOIIO3 | MOIIO2 | MOIIO1 | MOIIO0 | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | IO0FV | — | — | — | — | — | — | — | BSZ | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R | R/W | R/W |

Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | MD | — | R/W | Operating Mode Switch Switches the operating modes. 0: External address space read mode 1: Manual mode |
| 30 to 25 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|---|
| 24 | — | — | R | Reserved This bit is always read as initial value. The write value should always be initial value. |
| 23, 22 | MOIO3[1:0] | — | R/W | QSPIn_SSL Output Idle Value Fix QSPIn_IO3 Fixes output value of QSPIn_IO3 in QSPIn_SSL negation period. B'00: 0 B'01: 1 B'10: The value in the last bit of the previous transfer (Hi-Z, if Hi-Z was the value in the last bit of the previous transfer). B'11: Hi-Z Notes: 1. This bit should be specified to one when Octal-SPI flash is connected 2. When connecting to HyperFlash, this function is always "10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer)" |
| 21, 20 | MOIO2[1:0] | — | R/W | QSPIn_SSL Output Idle Value Fix QSPIn_IO2 Fixes output value of QSPIn_IO2 in QSPIn_SSL negation period. B'00: Output value 0 B'01: Output value 1 B'10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer). B'11: Output value Hi-Z Notes: 1. This bit should be specified to one when Octal-SPI flash is connected 2. When connecting to HyperFlash, this function is always "10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer)" |
| 19, 18 | MOIO1[1:0] | — | R/W | QSPIn_SSL Output Idle Value Fix QSPIn_IO1 Fixes output value of QSPIn_IO1 in QSPIn_SSL negation period. B'00: Output value 0 B'01: Output value 1 B'10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer). B'11: Output value Hi-Z Notes: 1. This bit should be specified to one when Octal-SPI flash is connected 2. When connecting to HyperFlash, this function is always "10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer)" |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 17, 16 | MOIO0[1:0] | — | R/W | <p>QSPIn_SSL Output Idle Value Fix QSPIn_IO0</p> <p>Fixes output value of QSPIn_IO0 in QSPIn_SSL negation period.</p> <p>B'00: Output value 0</p> <p>B'01: Output value 1</p> <p>B'10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer).</p> <p>B'11: Output value Hi-Z</p> <p>Notes: 1. This bit should be specified to one when Octal-SPI flash is connected</p> <p>2. When connecting to HyperFlash, this function is always "B'10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer)"</p> |
| 15 to 10 | — | — | R | <p>Reserved</p> <p>These bits are always read as initial value. The write value should always be initial value.</p> |
| 9, 8 | IO0FV[1:0] | — | R/W | <p>QSPIn_IO0 Fixed Value for 1-bit Size Input</p> <p>Fixes the output value of QSPIn_IO0 pin for 1-bit size input.</p> <p>B'00: Output value 0</p> <p>B'01: Output value 1</p> <p>B'10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer).</p> <p>B'11: Output value Hi-Z</p> |
| 7 to 2 | — | — | R | <p>Reserved</p> <p>These bits are always read as initial value. The write value should always be initial value.</p> |
| 1, 0 | BSZ[1:0] | — | R/W | <p>Data Bus Size</p> <p>Specifies the number of serial flash memory devices to be connected. When HyperFlash is connected, these bits should always be set to 01.</p> <p>B'00: Serial flash memory x 1</p> <p>B'01: Serial flash memory x 2 or HyperFlash x 1 or Octal-SPI flash memory x 1</p> <p>B'1X: Setting prohibited</p> <p>Note: After changing the value of this bit field, all the entries in the read cache must be cleared by setting the RCF bit in DRCR to B'1.</p> |

Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

56.2.2 SSL Delay Register (SSLDR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SSLDR is a 32-bit register that adjusts the timing between the QSPIn_SSL signal and the QSPIn_SPCLK signal.

The settings of this register are reflected both in external address space read mode and manual operating mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|-------|-----|-----|----|----|----|----|-------|-------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | SPNDL | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | SLNDL | | — | — | — | — | — | SCKDL | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 19 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 18 to 16 | SPNDL[2:0] | B'000 | R/W | Next Access Delay Sets the period from transfer end to next transfer start (next access). B'000: 1 cycle of QSPIn_SPCLK B'001: 2 cycles of QSPIn_SPCLK B'010: 3 cycles of QSPIn_SPCLK B'011: 4 cycles of QSPIn_SPCLK B'100: 5 cycles of QSPIn_SPCLK B'101: 6 cycles of QSPIn_SPCLK B'110: 7 cycles of QSPIn_SPCLK B'111: 8 cycles of QSPIn_SPCLK |
| 15 to 11 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 to 8 | SLNDL[2:0] | B'100 | R/W | QSPIn_SSL Negation Delay Sets the period from the time the last QSPIn_SPCLK edge is sent of a transfer to QSPIn_SSL pin negation (QSPIn_SSL negation delay). B'100: 5.5 cycles of QSPIn_SPCLK B'101: 6.5 cycles of QSPIn_SPCLK B'110: 7.5 cycles of QSPIn_SPCLK B'111: 8.5 cycles of QSPIn_SPCLK Other than above: Setting prohibited |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|--|
| 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 to 0 | SCKDL[2:0] | B'000 | R/W | Clock Delay Sets the period from QSPIn_SSL pin assertion to QSPIn_SPCLK oscillation (clock delay). B'000: 1 cycle of QSPIn_SPCLK B'001: 2 cycles of QSPIn_SPCLK B'010: 3 cycles of QSPIn_SPCLK B'011: 4 cycles of QSPIn_SPCLK B'100: 5 cycles of QSPIn_SPCLK B'101: 6 cycles of QSPIn_SPCLK B'110: 7 cycles of QSPIn_SPCLK B'111: 8 cycles of QSPIn_SPCLK |

56.2.3 Data Read Control Register (DRCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DRCR is a 32-bit register that sets the operation in the external address space read mode.

For other than SSLN bit, the settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|------|------|----|----|----|--------|-----|-----|-----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | SSLN | — | — | — | RBURST | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | —/WB | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | RCF | RBE | — | — | — | — | — | — | — | SSLE |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | —/WB | R/W | R | R | R | R | R | R | R | R/W |

- Notes: 1. Initial value varies depending on boot mode. (Refer to Table 56.3.)
 2. This function is available for write operation only. In read case, use external address space mode to read more than 32bit (1xQSPI) or 64bit (2xQSPI) data.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|------|--|
| 31 to 25 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 24 | SSLN | — | —/WB | QSPIn_SSL Negation Asserted QSPIn_SSL can be negated by writing 1 to this bit when both the RBE and SSLE bits are 1. This bit is always read as 0. To start next access after QSPIn_SSL negation using this bit, read SSLF in CMNSR = 0 to confirm that the QSPIn_SSL has been negated. Note: This setting is only valid while serial flash memory is connected. |
| 23 to 21 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 20 to 16 | RBURST[4:0] | — | R/W | Read Data Burst Length Sets the burst length (data unit count) when reading. This bit field is enabled when the RBE bit is set to 1. B'0_0000: 1 data unit B'0_0001: 2 continuous data units : B'1_1110: 31 continuous data units B'1_1111: 32 continuous data units One data unit is 64 bits long. |
| 15 to 10 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|------|--|
| 9 | RCF | — | —/WB | <p>Read Cache Flush</p> <p>When 1 is written to this bit, all the entries in the read cache are cleared.</p> <p>This bit is always read as 0.</p> <p>After flushing the read cache by writing 1 to the RCF bit, read the DRCCR before proceeding to read from the external address space. After operating by manual mode or using cache area as Write Buffer, this cache area must be cleared by writing 1 to the RCF bit.</p> |
| 8 | RBE | — | R/W | <p>Read Burst</p> <p>Enables or disables burst read.</p> <p>0: Normal read Data is read according to the access size.</p> <p>1: Burst read As many data units as the burst count specified in RBURST[4:0] bits are read. Read cache is enabled.</p> <p>Note: If an access spreads over the last address of the flash while a cache is enabled, the access address does not agree with the internal address of the flash. Manage the access so that the read address and the data size of the access does not spread over the last address.</p> |
| 7 to 1 | — | — | R | <p>Reserved</p> <p>These bits are always read as <u>initial value</u>. The write value should always be initial value.</p> |
| 0 | SSLE | — | R/W | <p>QSPIn_SSL Negation Setting</p> <p>Sets the conditions for QSPIn_SSL negation during read burst. QSPIn_SSL is negated for each access during normal read.</p> <p>0: QSPIn_SSL is negated after transfer of data set in burst length.</p> <p>1: QSPIn_SSL is negated when the accessed address is not continuous with the previously transferred address.</p> <p>Note: This setting is only valid while serial flash memory is connected.</p> |

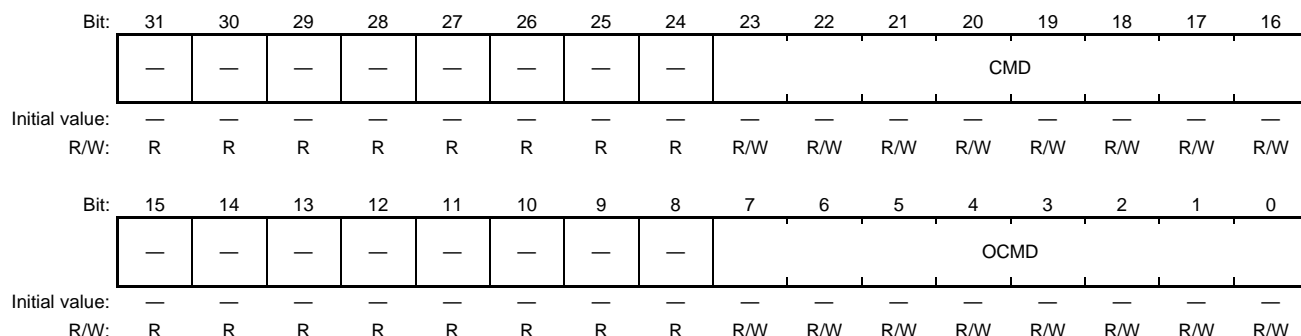
Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

56.2.4 Data Read Command Setting Register (DRCMR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DRCMR is a 32-bit register that sets the commands issued in external address space read mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 24 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 23 to 16 | CMD[7:0] | — | R/W | Command Sets the command. |
| 15 to 8 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 7 to 0 | OCMD[7:0] | — | R/W | Optional Command Sets the optional command. |

Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

56.2.5 Data Read Extended Address Setting Register (DREAR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DREAR is a 32-bit register that sets the address when the serial flash address is output in 32-bit mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed. This register is valid while the serial flash is connected.

| | | | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|-----|-----|---|---|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | | | |
| | — | — | — | — | — | — | — | — | EAV | | | | | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | EAC | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | | | | |

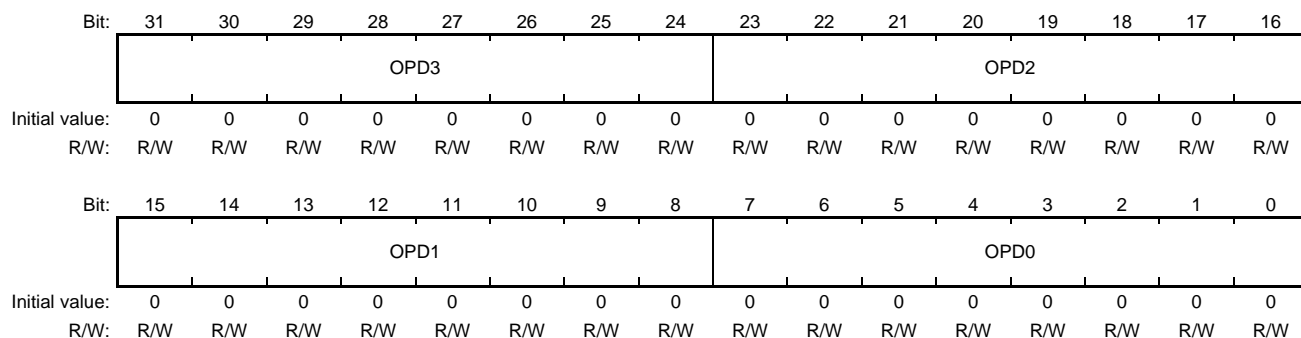
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | EAV[7:0] | H'00 | R/W | 32-Bit Extended Upper Address Fixed Value Sets the upper address bit values of the external address specified by the EAC[2:0] bits when the serial flash address is output in 32-bit mode. Bit 0 corresponds to the serial flash address bit [25], and bit 7 corresponds to the bit [32]. This setting is valid when the ADE[3] bit in DRENr is 1. When EAC[2:0] are 000, serial flash address [32:25] fixed values should set to EAV[7:0]. When EAC[2:0] are 001, serial flash address [32:26] fixed values should set to EAV[7:1]. (1) BSZ[1:0] bits in CMNCR = 00 (one serial flash memory connected) (2) BSZ[1:0] bits in CMNCR = 01 (two serial flash memory devices connected) Serial flash addresses [32:1] are used for accessing. |
| 15 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 to 0 | EAC[2:0] | B'000 | R/W | 32-Bit Extended External Address Valid Range Sets the range of the external address to be used as serial flash address when the serial flash address is output in 32-bit mode. This setting is valid when the ADE[3] bit in DRENr is 1. B'000: External address bits [24:0] enabled B'001: External address bits [25:0] enabled Other than above: Setting prohibited |

56.2.6 Data Read Option Setting Register (DROPR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DROPR is a 32-bit register that sets the option data in external address space read mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 24 | OPD3[7:0] | H'00 | R/W | Option Data 3 Sets the option data 3. |
| 23 to 16 | OPD2[7:0] | H'00 | R/W | Option Data 2 Sets the option data 2. |
| 15 to 8 | OPD1[7:0] | H'00 | R/W | Option Data 1 Sets the option data 1. |
| 7 to 0 | OPD0[7:0] | H'00 | R/W | Option Data 0 Sets the option data 0. |

Note: OPD3, OPD2, OPD1, and OPD0 are output in this order.

56.2.7 Data Read Enable Setting Register (DRENr)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DRENr is a 32-bit register that sets the bit size of the command, optional command, address, option data, and read data in external address space read mode and enables output of data other than read data. When connecting to HyperFlash, refer to Table 56.7 (Enable Registers (HyperFlash)).

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|------|------|-----|-----|-----|------|-----|-----|------|-----|----|----|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CDB | | OCDB | | — | — | ADB | | — | — | OPDB | | — | — | DRDB | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R | R | R/W | R/W | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DME | CDE | — | OCDE | ADE | | | OPDE | | | — | — | — | — | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R |

Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|--|
| 31, 30 | CDB[1:0] | — | R/W | Command Bit Size Sets the command size in bit units. B'00: 1 bit B'10: 4 bits Other than above: Setting prohibited |
| 29, 28 | OCDB[1:0] | — | R/W | Optional Command Bit Size Sets the optional command size in bit units. B'00: 1 bit B'10: 4 bits Other than above: Setting prohibited |
| 27, 26 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 25, 24 | ADB[1:0] | — | R/W | Address Bit Size Sets the address size in bit units. B'00: 1 bit B'10: 4 bits Other than above: Setting prohibited |
| 23, 22 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 21, 20 | OPDB[1:0] | — | R/W | Option Data Bit Size Sets the option data size in bit units. B'00: 1 bit B'10: 4 bits Other than above: Setting prohibited |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 19, 18 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 17, 16 | DRDB[1:0] | — | R/W | Data Read Size in Bit Units Sets the data read size in bit units. B'00: 1 bit B'10: 4 bits Other than above: Setting prohibited |
| 15 | DME | — | R/W | Dummy Cycle Enable Enables or disables insertion of a dummy cycle. The dummy cycle is inserted before the read data. Setting of a transfer starting with a dummy cycle is prohibited. 0: Insertion of a dummy cycle is disabled. 1: Insertion of a dummy cycle is enabled. |
| 14 | CDE | — | R/W | Command Enable Enables or disables output of commands. 0: Output disabled 1: Output enabled |
| 13 | — | — | R | Reserved This bit is always read as initial value. The write value should always be initial value. |
| 12 | OCDE | — | R/W | Optional Command Enable Enables or Disables output of Optional commands. 0: Output disabled 1: Optional command output enabled |
| 11 to 8 | ADE[3:0] | — | R/W | Address Enable Sets the address to be output. Be sure to use the following setting. Otherwise, the operation is not guaranteed. (1) BSZ[1:0] in CMNCR = 00 (one serial flash memory connected) B'0000: Output disabled B'0111: Data in address[23:0] is output. B'1111: Data in address[31:0] is output. Other than above: Setting prohibited (2) BSZ[1:0] in CMNCR = 01 (two serial flash memory devices connected) B'0000: Output disabled B'0111: Address[24:1] is output. B'1111: Address[32:1] is output. Other than above: Setting prohibited (3) PHYMEM in PHYCNT = 11 (HyperFlash connected) B'0100: HyperFlash protocol is output. Other than above: Setting prohibited Refer to Table 56.7 for the detail setting. (4) OCT in PHYCNT = 1 (Octal-SPI connected) with 8-8-8 protocol B'1100: Octal-SPI with 8-8-8 protocol is output. Other than above: Setting prohibited |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|---|
| 7 to 4 | OPDE[3:0] | — | R/W | <p>Option Data Enable</p> <p>Sets the option data to be output.</p> <p>Be sure to use the following setting. Otherwise, the operation is not guaranteed.</p> <p>B'0000: Output disabled</p> <p>B'1000: OPD3 is output.</p> <p>B'1100: OPD3 and OPD2 are output.</p> <p>B'1110: OPD3, OPD2, and OPD1 are output.</p> <p>B'1111: OPD3, OPD2, OPD1, and OPD0 are output.</p> <p>Other than above: Setting prohibited</p> |
| 3 to 0 | — | — | R | <p>Reserved</p> <p>These bits are always read as initial value. The write value should always be initial value.</p> |

Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

56.2.8 Manual Mode Control Register (SMCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SMCR is a 32-bit register that sets the operation in manual mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-------|----|----|----|----|----|-------|-------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | SSLKP | — | — | — | — | — | SPIRE | SPIWE | SPIE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R/W | R/W | —/WB |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | SSLKP | B'0 | R/W | QSPIn_SSL Signal Level Determines the QSPIn_SSL status after the end of transfer. 0: QSPIn_SSL signal is negated at the end of transfer. 1: QSPIn_SSL signal level is maintained from the end of transfer to the start of next access. Note: This function is available for write operation only. In read case, use external address space mode to read more than 32bit (1xQSPI) or 64bit (2xQSPI) data. |
| 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | SPIRE | B'0 | R/W | Data Read Enable Enables reading in manual mode. 0: Data reading disabled 1: Data reading enabled When the transfer data bit size is set to 4 bits with the SPIDB[1:0] bits, the SPIRE and SPIWE bits should not be set to 1 at the same time. |
| 1 | SPIWE | B'0 | R/W | Data Write Enable Enables writing in manual mode. 0: Data writing disabled 1: Data writing enabled When the transfer data bit size is set to 4 bits with the SPIDB[1:0] bits, the SPIRE and SPIWE bits should not be set to 1 at the same time. |

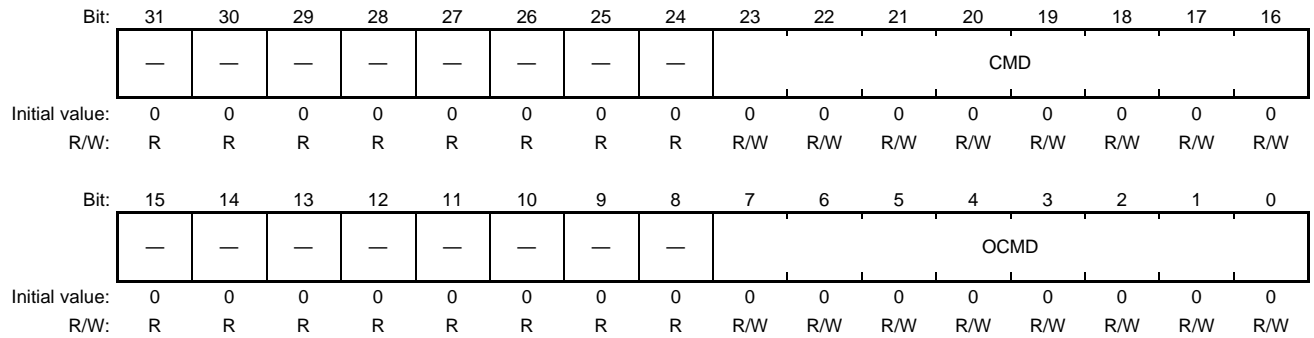
| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|---|
| 0 | SPIE | B'0 | —/WB | <p>SPI Data Transfer Enable</p> <p>Data is transferred by setting this bit to B'1.</p> <p>This bit is enabled only when the TEND bit in CMNSR is set to B'1. The operation cannot be guaranteed when this bit is set to 1 with the TEND bit set to B'0.</p> <p>This bit is always read as B'0.</p> <p>Note: When the QSPIn_SSL pin is negated, the command, optional command, address, and option data that are output enabled are output even if the SPIRE and SPIWE bits are set to B'0. When the QSPIn_SSL pin is asserted, follow the notes described in section 56.4.3, "Notes on Starting Transfer from the QSPIn_SSL Retained State in Manual Mode".</p> |

56.2.9 Manual Mode Command Setting Register (SMCMR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SMCMR is a 32-bit register that sets the commands issued in manual mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



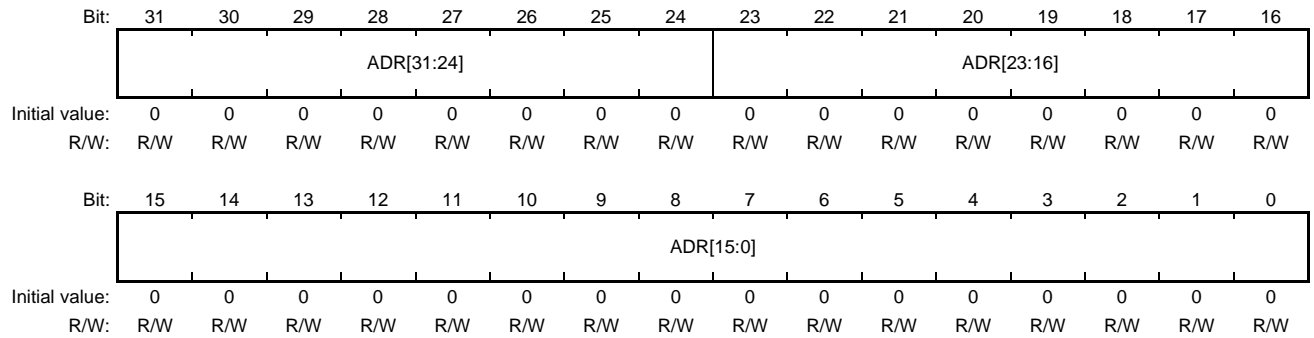
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | CMD[7:0] | H'00 | R/W | Command Sets the command. |
| 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | OCMD[7:0] | H'00 | R/W | Optional Command Sets the optional command. |

56.2.10 Manual Mode Address Setting Register (SMADR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SMADR is a 32-bit register that sets the addresses in manual mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



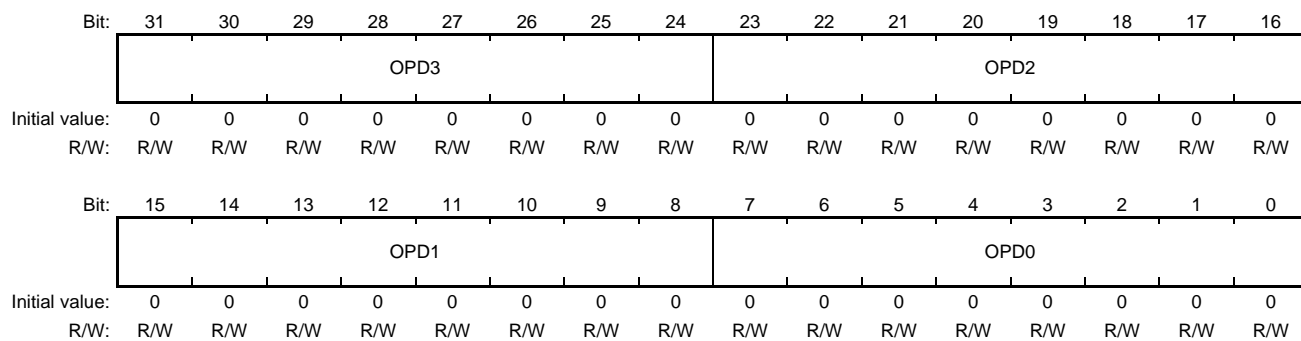
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 24 | ADR[31:24] | H'00 | R/W | Address Sets the value of bits 31 to 24 when the serial flash address is output in 32-bit units. This setting is valid when ADE[3] in SMENR is 1. |
| 23 to 0 | ADR[23:0] | H'00_0000 | R/W | Address Sets the address. |

56.2.11 Manual Mode Option Setting Register (SMOPR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SMOPR is a 32-bit register that sets the option data in manual mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 24 | OPD3[7:0] | H'00 | R/W | Option Data 3 Sets the option data 3. |
| 23 to 16 | OPD2[7:0] | H'00 | R/W | Option Data 2 Sets the option data 2. |
| 15 to 8 | OPD1[7:0] | H'00 | R/W | Option Data 1 Sets the option data 1. |
| 7 to 0 | OPD0[7:0] | H'00 | R/W | Option Data 0 Sets the option data 0. |

Note: OPD3, OPD2, OPD1, and OPD0 are output in this order.

56.2.12 Manual Mode Enable Setting Register (SMENR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SMENR is a 32-bit register that sets the bit size of the command, optional command, address, option data, and transfer data in manual mode and enables their output. SMENR also enables dummy cycle insertion. Disabling all of the command, optional command, address, option data, dummy cycle, and transfer data is prohibited. At least one of them except dummy cycle must be enabled. When connecting to HyperFlash, refer to Table 56.7 (Enable Registers (HyperFlash)).

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|------|------|-----|-----|-----|-----|------|-----|------|-----|-------|-----|-------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CDB | | OCDB | | — | — | ADB | | — | — | OPDB | | — | — | SPIDB | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R | R | R/W | R/W | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DME | CDE | — | OCDE | ADE | | | | OPDE | | | | SPIDE | | | |
| Initial value: | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|--|
| 31, 30 | CDB[1:0] | B'00 | R/W | Command Bit Size Sets the command size in bit units. B'00: 1 bit B'10: 4 bits Other than above: Setting prohibited |
| 29, 28 | OCDB[1:0] | B'00 | R/W | Optional Command Bit Size Sets the optional command size in bit units. B'00: 1 bit B'10: 4 bits Other than above: Setting prohibited |
| 27, 26 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 25, 24 | ADB[1:0] | B'00 | R/W | Address Bit Size Sets the address size in bit units. B'00: 1 bit B'10: 4 bits Other than above: Setting prohibited |
| 23, 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|---|
| 21, 20 | OPDB[1:0] | B'00 | R/W | Option Data Bit Size Sets the option data size in bit units. B'00: 1 bit B'10: 4 bits Other than above: Setting prohibited |
| 19, 18 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 17, 16 | SPIDB[1:0] | B'00 | R/W | Transfer Data Bit Size Sets the transfer data size in bit units. B'00: 1 bit B'10: 4 bits Other than above: Setting prohibited |
| 15 | DME | B'0 | R/W | Dummy Cycle Enable Enables or disables insertion of the dummy cycle before the read data. Note: Dummy cycle insertion is prohibited for write in manual mode including the case in which a transfer ends with a dummy cycle. A setting is prohibited for a transfer starting with a dummy cycle. 0: Dummy cycle insertion disabled 1: Dummy cycle insertion enabled |
| 14 | CDE | B'1 | R/W | Command Enable Sets the command to be output. 0: Output disabled 1: Output enabled Note: Specify this bit to 0 in external address space read mode with Octal-SPI flash. |
| 13 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 12 | OCDE | B'0 | R/W | Optional Command Enable Sets the optional command to be output. 0: Optional command output disabled 1: Optional command output enabled |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 11 to 8 | ADE[3:0] | B'0000 | R/W | <p>Address Enable</p> <p>Sets the address to be output.</p> <p>The following settings must be used. Otherwise, the operation cannot be guaranteed.</p> <p>(1) serial flash memory, or Octal-SPI without 8-8-8 protocol connected</p> <p>0000: Output disabled</p> <p>B'0100: ADR[23:16] is output</p> <p>B'0110: ADR[23:8] is output</p> <p>B'0111: ADR[23:0] is output</p> <p>B'1111: ADR[31:0] is output</p> <p>Other than above: Setting prohibited</p> <p>Note: These bits should be specified B'0111 or B'1111 with Write Buffer function.</p> <p>Refer to Figure 56.18.</p> <p>(2) Octal-SPI with 8-8-8 protocol connected</p> <p>B'1100: Octal-SPI with 8-8-8 protocol is output.</p> <p>Other than above: Setting prohibited</p> <p>(3) HyperFlash connected</p> <p>B'0100: HyperFlash protocol is output.</p> <p>Other than above: Setting prohibited</p> <p>Refer to Table 56.7 for the detail setting.</p> |
| 7 to 4 | OPDE[3:0] | B'0000 | R/W | <p>Option Data Enable</p> <p>Sets the option data to be output.</p> <p>The following settings must be used. Otherwise, the operation cannot be guaranteed.</p> <p>B'0000: Output disabled</p> <p>B'1000: OPD3 is output.</p> <p>B'1100: OPD3 and OPD2 are output.</p> <p>B'1110: OPD3, OPD2, and OPD1 are output.</p> <p>B'1111: OPD3, OPD2, OPD1, and OPD0 are output.</p> <p>Other than above: Setting prohibited</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|--|
| 3 to 0 | SPIDE[3:0] | B'0000 | R/W | <p>Transfer Data Enable</p> <p>Sets valid transfer data.</p> <p>Valid data differs depending on the BSZ[1:0] bit setting in CMNCR. The following settings must be used. Otherwise, the operation cannot be guaranteed.</p> <p>(1) BSZ[1:0] in CMNCR = 00 (one serial flash memory connected)</p> <p>B'0000: Not transferred</p> <p>B'1000: 8 bits transferred (enables data at address 0 of the manual mode read/write data registers 0)</p> <p>B'1100: 16 bits transferred (enables data at addresses 0 to 1 of the manual mode read/write data registers 0)</p> <p>B'1111: 32 bits transferred (enables data at addresses 0 to 3 of the manual mode read/write data registers 0)</p> <p>Other than above: Setting prohibited</p> <p>(2) BSZ[1:0] in CMNCR = 01 (two serial flash memory devices connected)</p> <p>B'0000: Not transferred</p> <p>B'1000: 16 bits transferred (enables data at addresses 0 to 1 of the manual mode read/write data registers 0)</p> <p>B'1100: 32 bits transferred (enables data at addresses 0 to 3 of the manual mode read/write data registers 0)</p> <p>B'1111: 64 bits transferred (enables data at addresses 0 to 3 of the manual mode read/write data registers 0 and data at addresses 0 to 3 of the manual mode read/write data registers 1)</p> <p>Other than above: Setting prohibited</p> |

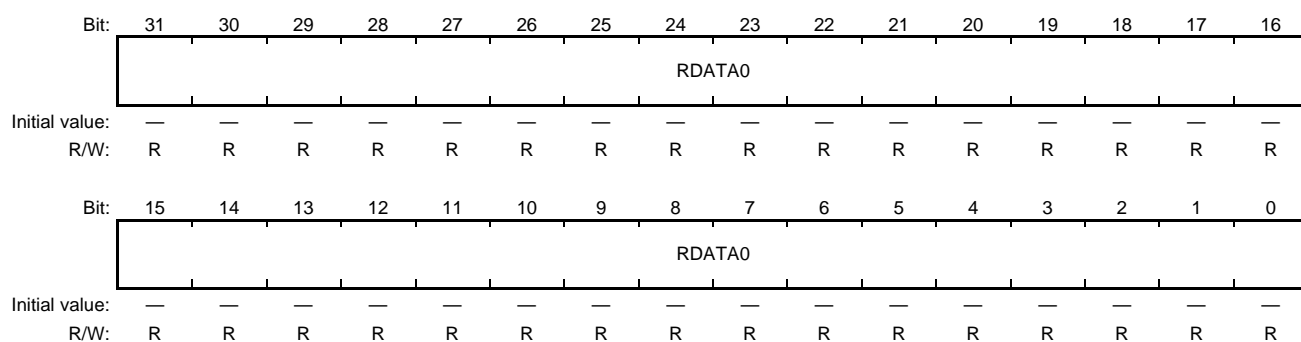
56.2.13 Manual Mode Read Data Register 0 (SMRDR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SMRDR0 is a 32-bit register that holds the read data in manual mode.

Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the manual mode enable setting register (SMENR). Be sure to access from LSB.

The data should be read when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 0 | RDATA0 [31:0] | Undefined | R | Read Data Holds the data read in manual mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = B'00: Read data[31:0] BSZ[1:0] = B'01: Read data[63:32] |

Note: The contents of this register and SMRDR1 are modified upon completion of reception in manual mode. Be sure to read data when reception in manual mode is completed.

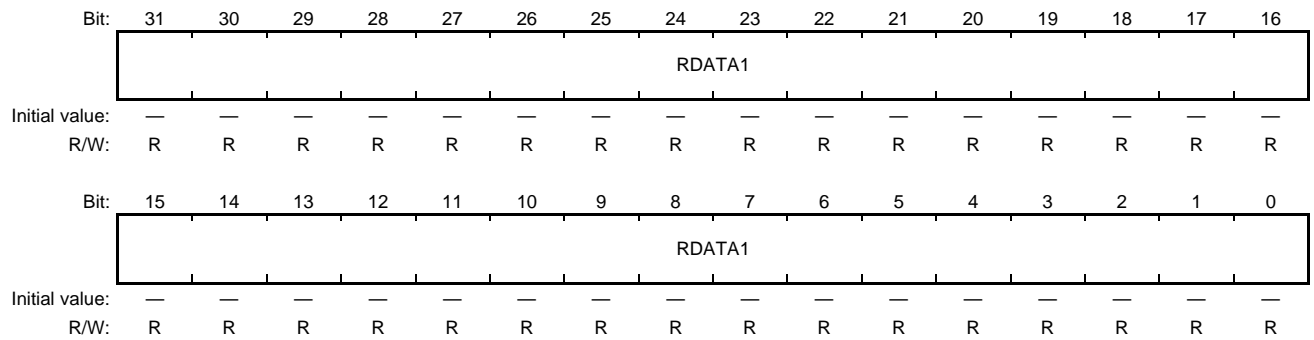
56.2.14 Manual Mode Read Data Register 1 (SMRDR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SMRDR1 is a 32-bit register that holds the read data in manual mode.

Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the manual mode enable setting register (SMENR). Be sure to access from LSB.

The data should be read when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|--|
| 31 to 0 | RDAT1 [31:0] | Undefined | R | Read Data Holds the data read in manual mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = B'00: Bits in this register are disabled. BSZ[1:0] = B'01: Read data[31:0] |

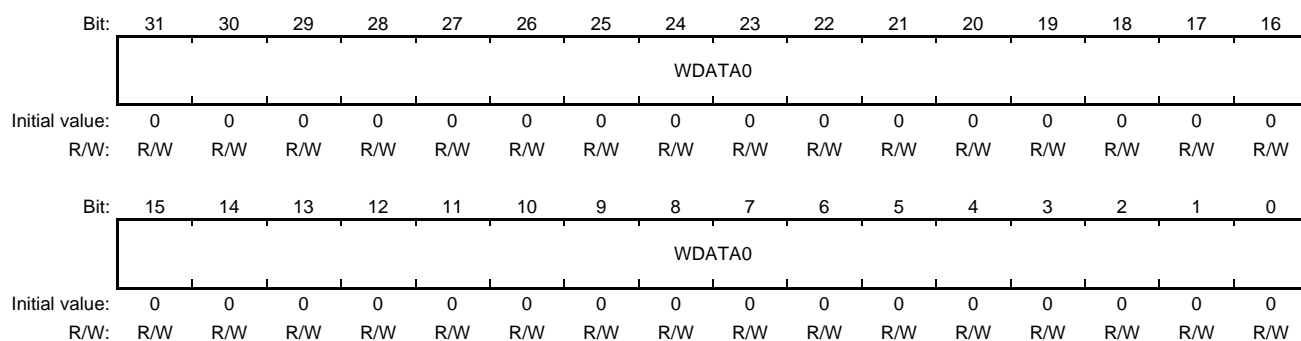
56.2.15 Manual Mode Write Data Register 0 (SMWDR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SMWDR0 is a 32-bit register that sets the write data in manual mode.

Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the manual mode enable setting register (SMENR). Be sure to access from address 0.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 0 | WDATA0 [31:0] | H'0000_0000 | R/W | Write Data Holds the data to be written in manual mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = B'00: Write data[31:0]. BSZ[1:0] = B'01: Write data[63:32]. |

56.2.16 Manual Mode Write Data Register 1 (SMWDR1)

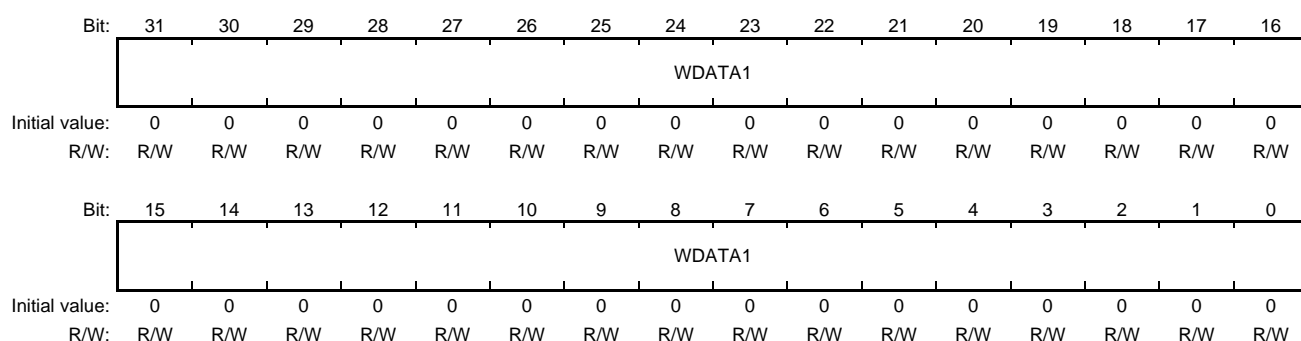
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SMWDR1 is a 32-bit register that sets the write data in manual mode.

This register is enabled when the BSZ[1:0] bits in CMNCR are set to 01 (two serial flash memory devices or HyperFlash connected) and disabled when BSZ[1:0] in CMNCR are set to 00 (one serial flash memory connected).

Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the manual mode enable setting register (SMENR). Be sure to access from address 0.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 31 to 0 | WDATA1 [31:0] | H'0000_0000 | R/W | Write Data Holds the data to be written in manual mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = B'00: Bits in this register are disabled. BSZ[1:0] = B'01: Write data[31:0]. |

56.2.17 Common Status Register (CMNSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMNSR is a 32-bit register that holds flags indicating the operating state.

The settings of this register are reflected both in external address space read mode and manual operating mode.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SSLF | TEND |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | SSLF | B'0 | R | QSPIn_SSL Pin State Monitor B'0: QSPIn_SSL pin is negated. B'1: QSPIn_SSL pin is asserted. |
| 0 | TEND | B'1 | R | Transfer End Flag Indicates whether the data transfer has ended. 0: Indicates that data transfer is in progress. 1: Indicates that data transfer has ended. |

56.2.18 Data Read Dummy Cycle Setting Register (DRDMCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DRDMCR is a 32-bit register that sets the size and number of dummy cycles to be inserted in external address space read mode.

The settings of this register are enabled when the DME bit in the data read enable setting register (DRENr) is 1.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|-------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | DMCYC | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 to 5 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 4 to 0 | DMCYC[4:0] | — | R/W | Number of Dummy Cycles Setting Sets the number of dummy cycles to be inserted when the DME bit in the data read enable setting register (DRENr) is 1. B'0_0000: 1 cycle B'0_0001: 2 cycles B'0_0010: 3 cycles B'1_0010: 19 cycles B'1_0011: 20 cycles Other than above: Setting prohibited |

Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

56.2.19 Data Read DDR Enable Register (DRDRENr)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DRDRENr is a 32-bit register that specifies the SDR or DDR transfer of the address, option data, and read data in external address space read mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|------|-----|-----|----|----|-------|-----|----|----|-------|-----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | HYPE | | — | — | — | ADDRE | — | — | — | OPDRE | — | — | — | — | DRDRE |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R/W | R/W | R/W | R | R | R | R/W | R | R | R | R/W | R | R | R | R/W |

Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 15 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 14 to 12 | HYPE | — | R/W | HyperFlash or Octal-SPI flash in DDR mode Enable B'101*: HyperFlash or Octal-SPI flash in DDR mode B'100*: Octal-SPI flash to issue only command in DDR mode B'000: SPI flash mode Other than above: Setting prohibited Setting prohibited when using DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit) (RZ/G2M V1.3 Ver.1.2) (not RZ/G2M V3.0) |
| 11 to 9 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 8 | ADDRE | — | R/W | Address DDR Enable Specifies the SDR or DDR transfer of the address. 0: SDR transfer 1*: DDR transfer Note: When HyperFlash is connected, specify 1 in this bit. Setting prohibited when using DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit) (RZ/G2M V1.3 Ver.1.2) (not RZ/G2M V3.0) |
| 7 to 5 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 4 | OPDRE | — | R/W | Option Data DDR Enable Specifies the SDR or DDR transfer of the option data. 0: SDR transfer 1*: DDR transfer Setting prohibited when using DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit) (RZ/G2M V1.3 Ver.1.2) (not RZ/G2M V3.0) |
| 3 to 1 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 0 | DRDRE | — | R/W | Data Read DDR Enable Specifies the SDR or DDR transfer of the data read. 0: SDR transfer 1*: DDR transfer Note: When Hyperflash is connected, specify 1 in this bit. Setting prohibited when using DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit) (RZ/G2M V1.3 Ver.1.2) (not RZ/G2M V3.0) |

Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

56.2.20 Manual Mode Dummy Cycle Setting Register (SMDMCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SMDMCR is a 32-bit register that sets the size and number of dummy cycles to be inserted in manual mode.

The settings of this register are enabled when the DME bit in the manual mode enable setting register (SMENR) is 1.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|-------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | DMCYC | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 to 0 | DMCYC[4:0] | B'0_0000 | R/W | Number of Dummy Cycles Setting Sets the number of dummy cycles to be inserted when the DME bit in the manual mode enable setting register (SMENR) is 1. B'0_0000: 1 cycle B'0_0001: 2 cycles B'0_0010: 3 cycles B'1_0010: 19 cycles B'1_0011: 20 cycles Other than above: Setting prohibited |

56.2.21 Manual Mode DDR Enable Register (SMDRENDR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SMDRENDR is a 32-bit register that specifies the SDR or DDR transfer of the address, option data, and data for transfer in manual mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|------|-----|-----|----|----|----|-------|----|----|----|-------|----|----|----|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | HYPE | | | — | — | — | ADDRE | — | — | — | OPDRE | — | — | — | SPIDRE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R | R | R | R/W | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 15 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 14 to 12 | HYPE | B'000 | R/W | HyperFlash or Octal-SPI flash in DDR mode Enable B'101*: HyperFlash or Octal-SPI flash in DDR mode B'100*: Octal-SPI flash to issue only command in DDR mode 000: SPI flash mode Other than above: Setting prohibited Setting prohibited when using DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit) (RZ/G2M V1.3) (not RZ/G2M V3.0) |
| 11 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | ADDRE | B'0 | R/W | Address DDR Enable Specifies the SDR or DDR transfer of the address. 0: SDR transfer 1*: DDR transfer Note: When HyperFlash is connected, specify 1 in this bit. Setting prohibited when using DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit) (RZ/G2M V1.3) (not RZ/G2M V3.0) |
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | OPDRE | B'0 | R/W | Option Data DDR Enable Specifies the SDR or DDR transfer of the option data. 0: SDR transfer 1*: DDR transfer Setting prohibited when using DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit) (RZ/G2M V1.3) (not RZ/G2M V3.0) |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | SPIDRE | B'0 | R/W | Transfer Data DDR Enable Specifies the SDR or DDR transfer of the data for transfer. 0: SDR transfer 1*: DDR transfer Note: When HyperFlash is connected, specify 1 in this bit. Setting prohibited when using DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit) (RZ/G2M V1.3) (not RZ/G2M V3.0) |

56.2.22 PHY Control Register (PHYCNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

PHYCNT is a 32-bit register that sets the PHY operating mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|------------|----|----|----|----|----|----|----|------|------|-----|------------|-----|--------|--------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CAL | — | — | — | — | — | — | — | OCTA | EXDS | OCT | DDRCA L | HS | STRTIM | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | —/WB | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | STRTI M | — | — | — | — | — | — | — | — | — | — | WBUF2 | — | WBUF | PHYMEM | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R/W | R | R | R | R | R | R | R | R | R | R | R/W | R | R/W | R/W | R/W |

Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|------|---|
| 31 | CAL | — | —/WB | PHY Calibration Executes calibration of the PHY. 1: Calibration is executed. 0: Calibration is not executed. Note: When set this bit to 1, STRTIM[2:0] should be set to B'110 (RZ/G2M V1.3, RZ/G2M V3.0) or B'111. (Other products). |
| 30 to 24 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 23, 22 | OCTA | — | R/W | Octal-SPI flash alignment Specify data alignment with Octal-SPI flash memory. Refer to 56.3.14 for detail. These bits should be set to 00 when HyperFlash or serial flash memory is connected. B'00: HyperFlash or serial flash memory x 1ch or 2ch is connected B'10: Support sequential alignment B'01: Support alternate alignment B'11: Support alternate alignment Note: Alternate alignment with OCTA[1:0] = 01 is available in manual mode only. |
| 21 | EXDS | — | R/W | External Data Strobe Uses external Data Strobe signal with serial flash memory when connecting to serial flash with Data Strobe. When Octal-SPI flash memory is connected, specify this bit to 1. Note: Set this bit to 0 when HyperFlash is connected. 1: Use external Data Strobe signal 0: Not use external Data Strobe signal |
| 20 | OCT | — | R/W | Octal-SPI flash protocol mode 1: Use Octal-SPI DDR/SDR protocol mode 0: Specify 0 in the other than above mode |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 19 | DDRCAL | — | R/W | This bit is specified to 1 in SW calibration for DDR transfer of serial flash operation. Keep 1 in DDR transfer of serial flash operation after SW calibration. When SW calibration for DDR transfer of serial flash is not executed, this bit should be specified to 0. Note: Specify 0 to this bit. (RZ/G2M V1.3 Ver.1.2) (not RZ/G2M V3.0) |
| 18 | HS | — | R/W | High Speed response mode Specify high speed response mode. 1: The read data is output to bus master in parallel of device access when DRCR.RBE = 1. 0: The read data is output to bus master after the number of data which is specified in DRCR.RBURST is read. Notes: 1. This bit should be specified to 1 when using DMA transfer. DRCR.RBURST should be set to B'1_1111. 2. Accessing to register area during DMA transfer is prohibited. 3. When this bit set to 1, access address alignment from RPC to flash memory should be 256Byte align. |
| 17 to 15 | STRTIM[2:0] | — | R/W | Strobe Timing Adjustment bit These bits specify the internal strobe delay which is used for the flash device without external strobe, like serial flash memory. When connecting to HyperFlash or any other flash device which has strobe, set these bits to B'110 (RZ/G2M V1.3) or B'111 (Other products). B'111: The delay is smallest. B'110: The delay is 2nd smallest. : B'001: The delay is 2nd biggest. B'000: The delay is biggest. Note: Only STRTIM[2:1] is available in RZ/G M3W (not RZ/G2M V3.0). STRTIM[0] should always be 0. |
| 14 to 5 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 4 | WBUF2 | — | R/W | Write Buffer Enable2 The write buffer is used when the flash memory is written to. Refer to 56.3.13 for usage of the write buffer. 1: The write buffer is used to write data to the flash memory. 0: The write buffer is not used. |
| 3 | — | — | R | Reserved This bit is always read as initial value. The write value should always be initial value. |
| 2 | WBUF | — | R/W | Write Buffer Enable The write buffer is used when the flash memory is written to. Refer to 56.3.13 for usage of the write buffer. 1: The write buffer is used to write data to the flash memory. 0: The write buffer is not used. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 1, 0 | PHYMEM | — | R/W | Device Selection Selects a device to connect. B'00: Serial flash in SDR mode B'01*: Serial flash in DDR mode B'11: HyperFlash Other than above: Setting prohibited Setting prohibited when using DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit) (RZ/G2M V1.3 Ver.1.2) (not RZ/G2M V3.0) |

Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

56.2.23 PHY Timing Offset Register 1 (PHYOFFSET1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

PHYOFFSET1 is a 32-bit register that sets the timing adjustment in DDR operation.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|--------|-----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | DDRTMG | | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31, 30 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 29, 28 | DDRTMG | — | R/W | DDR Operation Timing Register Specify the timing adjustment in DDR read operation. B'10: When SMDREN.R.SPIDRE or DRDREN.R.DRDRE = 1 in read operation. B'11: When SMDREN.R.SPIDRE or DRDREN.R.DRDRE = 0 in read operation Other than above: Setting prohibited Setting prohibited when using DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit) (RZ/G2M V1.3) (not RZ/G2M V3.0) |
| 27 to 0 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |

56.2.24 Timing Offset Register 2 (PHYOFFSET2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

PHYOFFSET2 is a 32-bit register that sets the timing adjustment in DDR operation.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|--------|-----|-----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | OCTTMG | | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 11 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 to 8 | OCTTMG | B'100 | R/W | Octal-SPI flash Operation Timing Specifies the timing adjustment when write to Octal-SPI flash B'000: Write Buffer Operation of serial flash with 1-1-4 or 1-4-4 of Table 56.14 B'100: serial flash or Hyperflash B'011: Octal-SPI flash operation Other than above: Setting prohibited Note: Specify B'100 when serial flash compatible mode of Octal-SPI flash. |
| 7 to 0 | — | B'0011_0001 | R | Reserved These bits are always read as 0. The write value should always be initial value. |

56.2.25 PHY Interrupt Register (PHYINT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

PHYINT is a 32-bit register that sets the interrupt signal and the pins.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-------|----|----|----|----|----|--------|-----------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | INTIE | — | — | — | — | — | RSTEN | WPEN (*) | INTEN |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R/W | (*) | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | RSTVAL | WPVAL (*) | INT |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | (*) | R |

Notes: Initial value varies depending on boot mode. (Refer to Table 56.3.)

* RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, and RZ/G2N support these bits as WPEN and WPVAL. These bits are assigned R/W.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 25 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 24 | INTIE | — | R/W | RPC_INT# Pin Input Enable Input Enables or disables the RPC_INT# pin. 0: The RPC_INT# pin input is disabled. 1: The RPC_INT# pin input is enabled. Note: This bit and PHYINT.INTEN bit should be 1 when using interrupt from external device. |
| 23 to 19 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 18 | RSTEN | — | R/W | RPC_RESET# Pin Enable Enables or disables the RPC_RESET# pin. 0: The RPC_RESET# pin is disabled. 1: The RPC_RESET# pin is enabled, and the value on RSTVAL is reflected. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 17 | WPEN | — | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, and RZ/G2N] RPC_WP# Pin Enable Enables or disables the RPC_WP# pin. 0: The RPC_WP# pin is disabled. 1: The RPC_WP# pin is enabled, and the settings of WPVAL are reflected |
| | — | — | R | [RZ/G2E] Reserved This bit is always read as initial value. The write value should always be initial value. |
| 16 | INTEN | — | R/W | RPC_INT# Pin Enable Enables or disables the RPC_INT# pin. 0: The RPC_INT# pin is disabled. 1: The RPC_INT# pin is enabled, and the interrupt signal from HyperFlash is enabled. |
| 15 to 3 | — | — | R | Reserved These bits are always read as initial value. The write value should always be initial value. |
| 2 | RSTVAL | — | R/W | RPC_RESET# Pin Output Value Specifies the value output from the RPC_RESET# pin. This setting is enabled when RSTEN = 1. 0: RPC_RESET# = H 1: RPC_RESET# = L |
| 1 | WPVAL | — | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, and RZ/G2N] RPC_WP# Pin Output Value Specifies the value output from the RPC_WP# pin. This setting is enabled when WPEN = 1. 0: RPC_WP# = H 1: RPC_WP# = L |
| | — | — | R | [RZ/G2E] Reserved This bit is always read as initial value. The write value should always be initial value. |
| 0 | INT | — | R | Interrupt Status When RPC_INT# is set to L, this bit becomes H to notice interrupt occurs in the connected device. |

Note: Initial value varies depending on boot mode. (Refer to Table 56.3.)

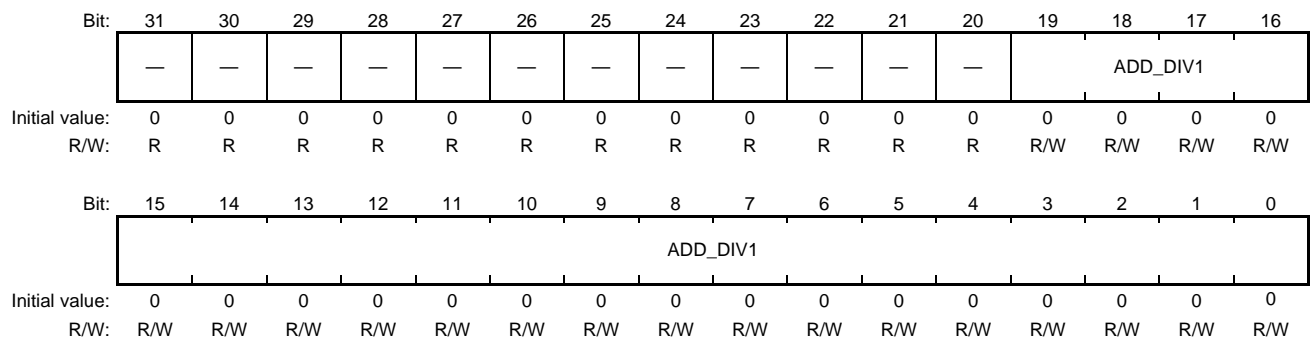
56.2.26 Address Division Register 1 (ADD_DIV1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | — | — |

ADD_DIV1 is a 32-bit register that specifies the address which divides the RPC-IF area for security domain.

Only Secure CPU can access to this register.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 to 0 | ADD_DIV1 [19:0] | All 0 | R/W | Address Division bit 1 Specifies address boundary of secure domain in RPC-IF area. These bits are used with ADD_DIV2 and ADD_DIV3. Refer to 56.3.18 for the usage. |

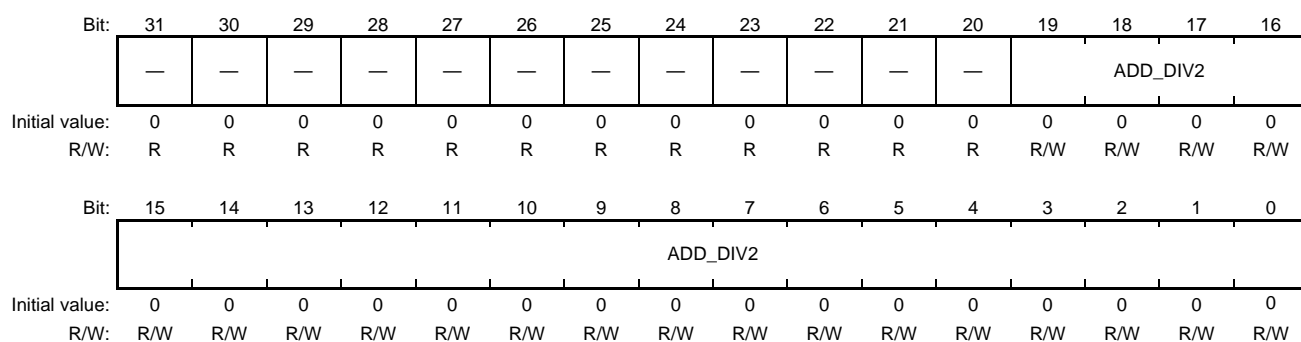
56.2.27 Address Division Register 2 (ADD_DIV2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | — | — |

ADD_DIV2 is a 32-bit register that specifies the address which divides the RPC-IF area for security domain.

Only Secure CPU can access to this register.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 to 0 | ADD_DIV2 [19:0] | All 0 | R/W | Address Division bit 2 Specifies address boundary of secure domain in RPC-IF area. These bits are used with ADD_DIV1 and ADD_DIV3. Refer to 56.3.18 for the usage. |

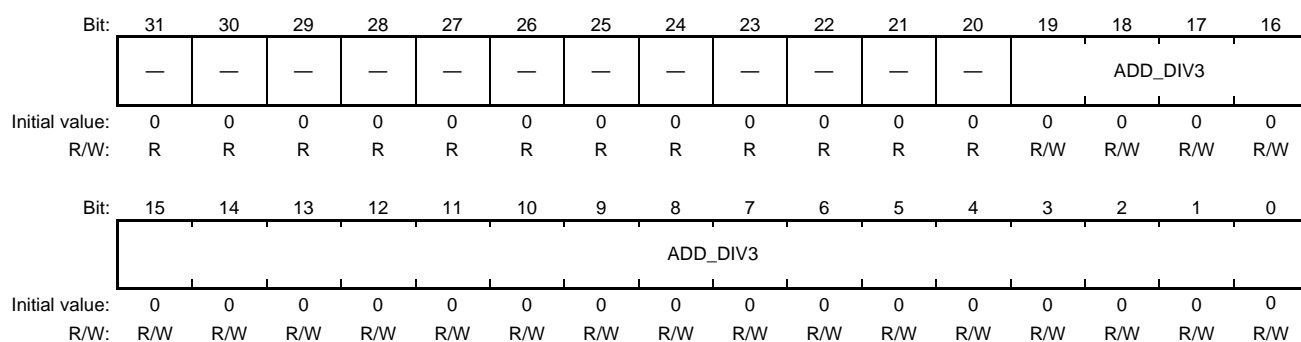
56.2.28 Address Division Register 3 (ADD_DIV3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | — | — |

ADD_DIV3 is a 32-bit register that specifies the address which divides the RPC-IF area for security domain.

Only Secure CPU can access to this register.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 to 20 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 19 to 0 | ADD_DIV3 [19:0] | All 0 | R/W | Address Division bit 3 Specifies address boundary of secure domain in RPC-IF area. These bits are used with ADD_DIV1 and ADD_DIV2. Refer to 56.3.18 for the usage. |

56.2.29 Secure Configuration Register (SEC_CONF)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | — | — |

SEC_CONF is a 32-bit register that specifies the attribution of each domain which is divided by ADD_DIVn (n = 1, 2, 3).

Only Secure CPU can access to this register.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-------|-------|-------|-------|-------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | SECEN | AREA3 | AREA2 | AREA1 | AREA0 | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | SECEN | B'1 | R/W | Secure function enable bit Enables secure function command restriction. 0: secure 1: non-secure The others: reserved Refer to 56.3.18 for the usage. |
| 7, 6 | AREA3[1:0] | All 0 | R/W | Area Attribution 3 Specifies the attribution of Area3 which is specified by ADD_DIVn register in RPC-IF area. B'00: secure B'01: non-secure The others: reserved Refer to 56.3.18 for the usage. |
| 5, 4 | AREA2[1:0] | All 0 | R/W | Area Attribution 2 Specifies the attribution of Area2 which is specified by ADD_DIVn register in RPC-IF area. B'00: secure B'01: non-secure The others: reserved Refer to 56.3.18 for the usage. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|------------|---------------|-----|--|
| 3, 2 | AREA1[1:0] | All 0 | R/W | <p>Area Attribution 1</p> <p>Specifies the attribution of Area1 which is specified by ADD_DIVn register in RPC-IF area.</p> <p>B'00: secure</p> <p>B'01: non-secure</p> <p>The others: reserved</p> <p>Refer to 56.3.18 for the usage.</p> |
| 1, 0 | AREA0[1:0] | All 0 | R/W | <p>Area Attribution 0</p> <p>Specifies the attribution of Area0 which is specified by ADD_DIVn register in RPC-IF area.</p> <p>B'00: secure</p> <p>B'01: non-secure</p> <p>The others: reserved</p> <p>Refer to 56.3.18 for the usage.</p> |

56.2.30 Access Right Register (ARIGHT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | — | — |

ARIGHT is a 32-bit register that specifies the semaphore function which limits the access to the register of this controller.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|------------|----|-----------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | RIGHT_NONS | — | RIGHT_SEC | RIGHT_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|---|
| 31 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 | RIGHT_NON S | B'0 | R | Right Bit For Non-Secure Specifies the right bit for Non-Secure CPU. This bit is set to 1 when Non-Secure CPU writes 1 on RIGHT_EN bit. 1: Non-Secure CPU has a right to access 0: Non-Secure CPU doesn't have a right to access Note this bit can be cleared by both Secure and Non-Secure CPU. |
| 2 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 1 | RIGHT_SEC | B'0 | R | Right Bit For Secure Specifies the right bit for Secure CPU. This bit is set to 1 when Secure CPU writes 1 on RIGHT_EN bit. 1: Secure CPU has a right to access 0: Secure CPU doesn't have a right to access |
| 0 | RIGHT_EN | B'0 | R/W | Access Right Enable Specifies the access right function to enable. When accessing to this register with RIGHT_EN = 1, the right is reflected in each bit 3 and 1 according to the attribution of each master. To cancel the access right, the same attribution master should access to this register with RIGHT_EN = 0. The access is ignored when the attribution of access is different. 0: Not use access right function 1: Use access right function Note: Secure attribution can overwrite Non-Secure right. |

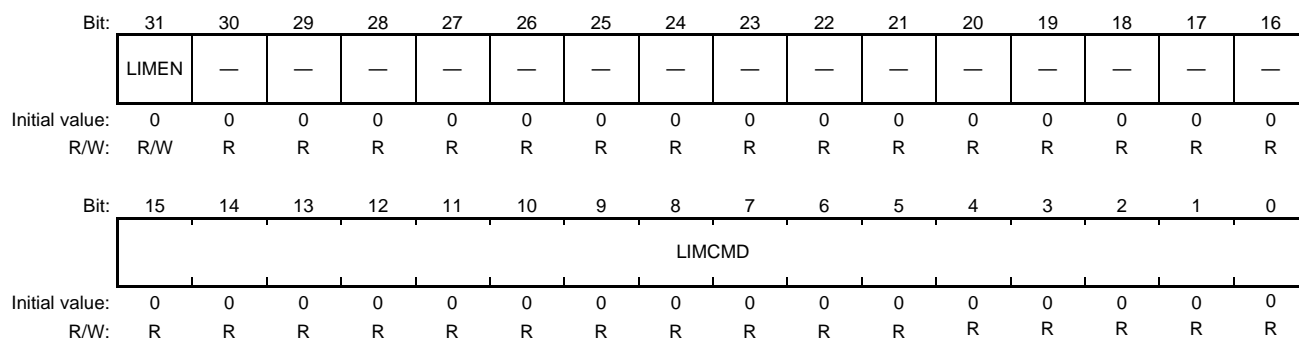
56.2.31 Secure Command Register (SEC_CMDn (n = 0 to 15))

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | — | — |

SEC_CMDn (n = 0 to 15) is a 32-bit register that specifies the command which is prohibited for Secure CPU.

Only Secure CPU can access to this register.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | LIMEN | B'0 | R/W | Command Limitation Enable for Secure CPU Specifies the limited command for Secure CPU. 1: The command specified in LIMCMD[15:0] is limited 0: The command specified in LIMCMD[15:0] is not limited |
| 30 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | LIMCMD | All 0 | R | Limited Command for Secure CPU Specifies the command which is limited for Secure CPU. When LIMEN = 1, the command specified in these bits are checked and not issued to the device. |

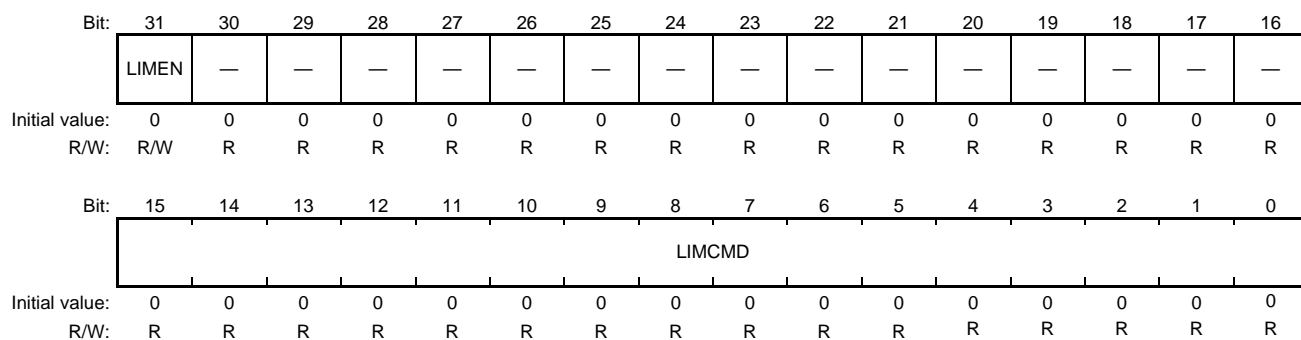
56.2.32 Non-Secure Command Register (NON_SEC_CMDn (n = 0 to 15))

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | — | — |

NON_SEC_CMDn (n = 0 to 15) is a 32-bit register that specifies the command which is prohibited for Non-Secure CPU.

Only Secure CPU can access to this register.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | LIMEN | B'0 | R/W | Command Limitation Enable for Non-Secure CPU Specifies the limited command for Non-Secure CPU. 1: The command specified in LIMCMD[15:0] is limited 0: The command specified in LIMCMD[15:0] is not limited |
| 30 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | LIMCMD | All 0 | R | Limited Command for Non-Secure CPU Specifies the command which is limited for Non-Secure CPU. When LIMEN = 1, the command specified in these bits are checked and not issued to the device. |

56.2.33 Erase Command List Register (ERASELISTn (n = 1, 2))

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | — | — |

ERASELISTn (n = 1, 2) is a 32-bit register that specifies Erase Command to limit.

Only Secure CPU can access to this register.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | | | | | |
|----------------|-------------|----|----|----|----|----|----|----|-------|-----|-----|-----|-----|-----|-----|-----|--|--|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | | | |
| | ECMD EN1 | — | — | — | — | — | — | — | ECMD1 | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| R/W: | R/W | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | |
| | ECMD EN2 | — | — | — | — | — | — | — | ECMD2 | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| R/W: | R/W | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | ECMDEN1 | B'0 | R/W | Erase Command 1 Enable Enables to limit the erase command which is specified In ECMD1. |
| 30 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | ECMD1 | All 0 | R/W | Erase Command 1 Specifies the Erase command to limit. |
| 15 | ECMDEN2 | B'0 | R/W | Erase Command 2 Enable Enables to limit the erase command which is specified In ECMD2. |
| 14 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | ECMD2 | All 0 | R/W | Erase Command 2 Specifies the Erase command to limit. |

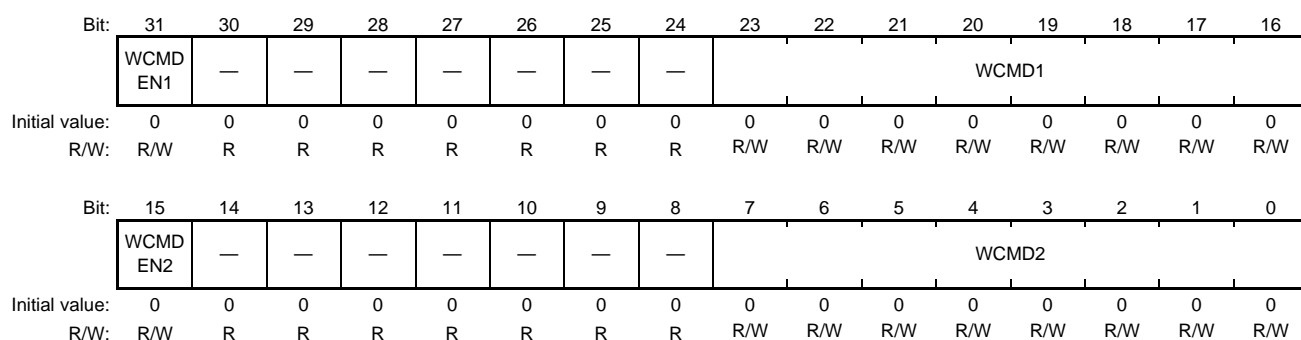
56.2.34 Write Command List Register (WRITELISTn (n = 1, 2))

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | — | — |

WRITELISTn (n = 1, 2) is a 32-bit register that specifies Write Command to limit.

Only Secure CPU can access to this register.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | WCMDEN1 | B'0 | R/W | Write Command 1 Enable Enables to limit the Write command which is specified In WCMD1. 0: Disable to limit the Write command which is specified In WCMD1. 1: Enable to limit the Write command which is specified In WCMD1. |
| 30 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 16 | WCMD1 | All 0 | R/W | Write Command 1 Specifies the Write command to limit. |
| 15 | WCMDEN2 | B'0 | R/W | Write Command 2 Enable Enables to limit the Write command which is specified In WCMD2. 0: Disable to limit the Write command which is specified In WCMD2. 1: Enable to limit the Write command which is specified In WCMD2. |
| 14 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | WCMD2 | All 0 | R/W | Write Command 2 Specifies the Write command to limit. |

56.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

56.3.1 System Configuration

With this module, one or two serial flash memory devices (data size of 1, 2, and 4 bits) or one HyperFlash (data size of 8 bits) can be connected. The connected devices can be selected using the BSZ[1:0] bits in CMNCR.

Examples of system configuration are shown in Figure 56.2, 56.3, 56.4 and 56.5.

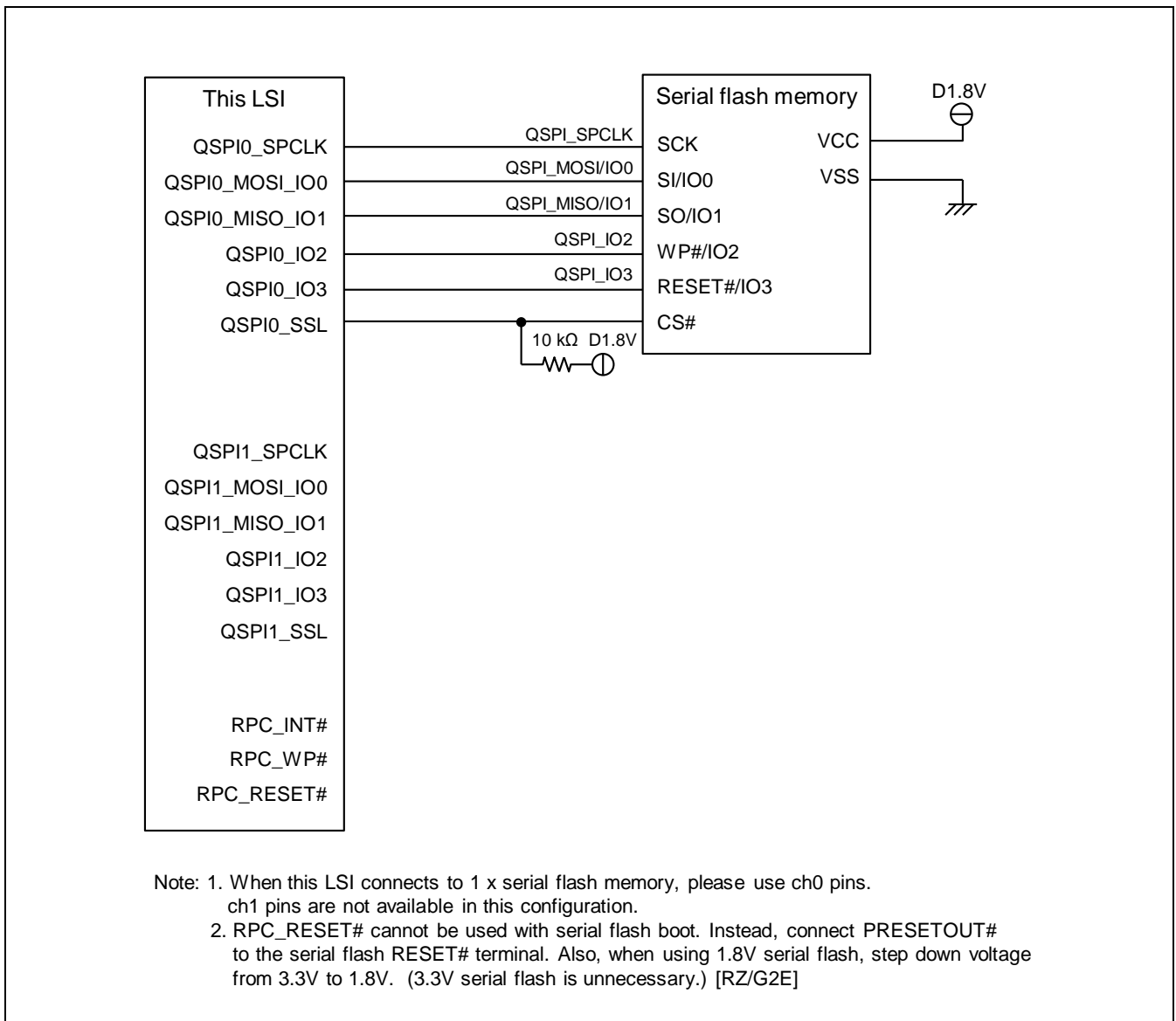


Figure 56.2 Example of System Configuration with One 4-Bit Serial Flash Memory Device Connected

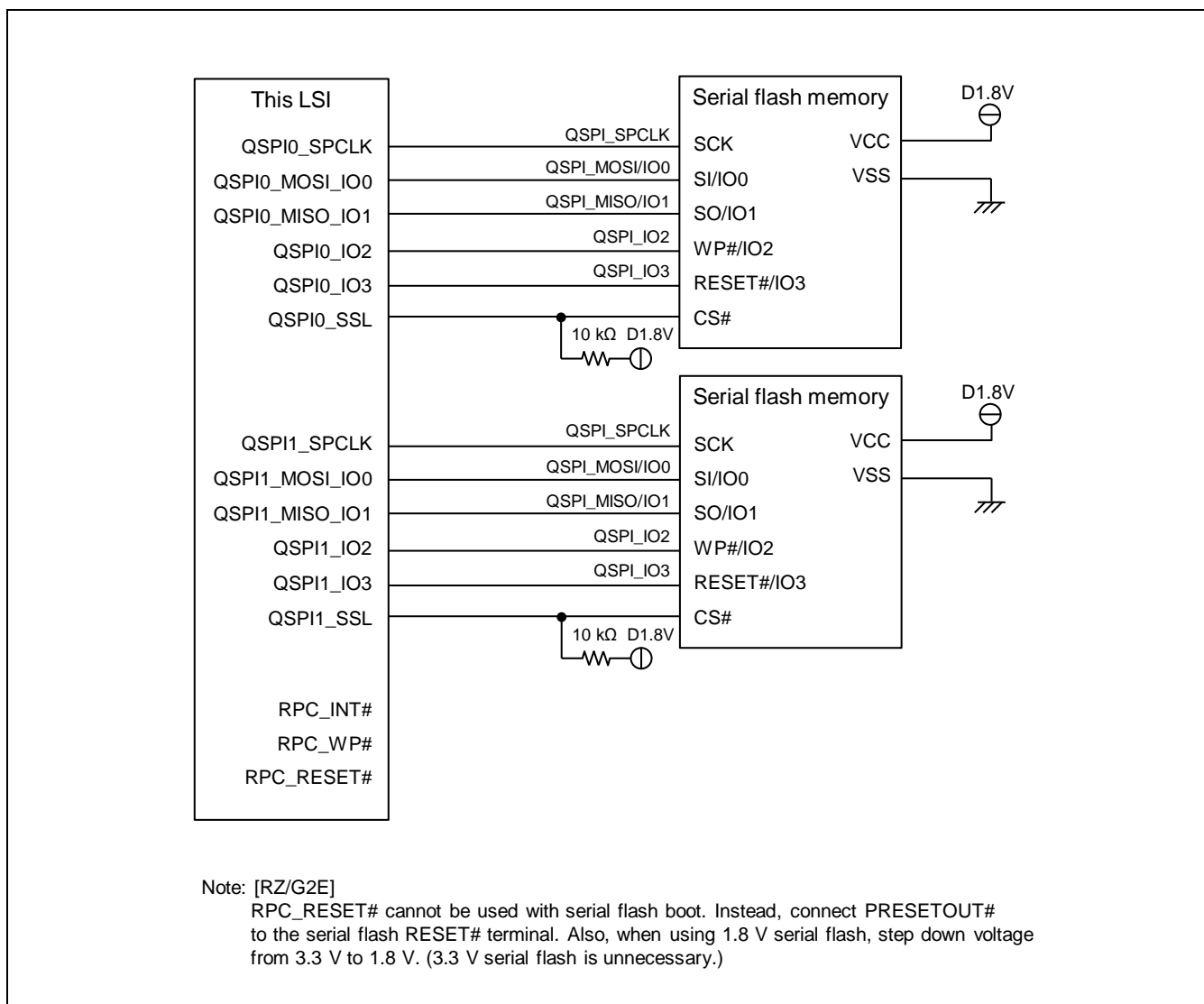


Figure 56.3 Example of System Configuration with Two 4-Bit Serial Flash Memory Devices Connected (BSZ[1:0] Bits in CMNCR = 01)

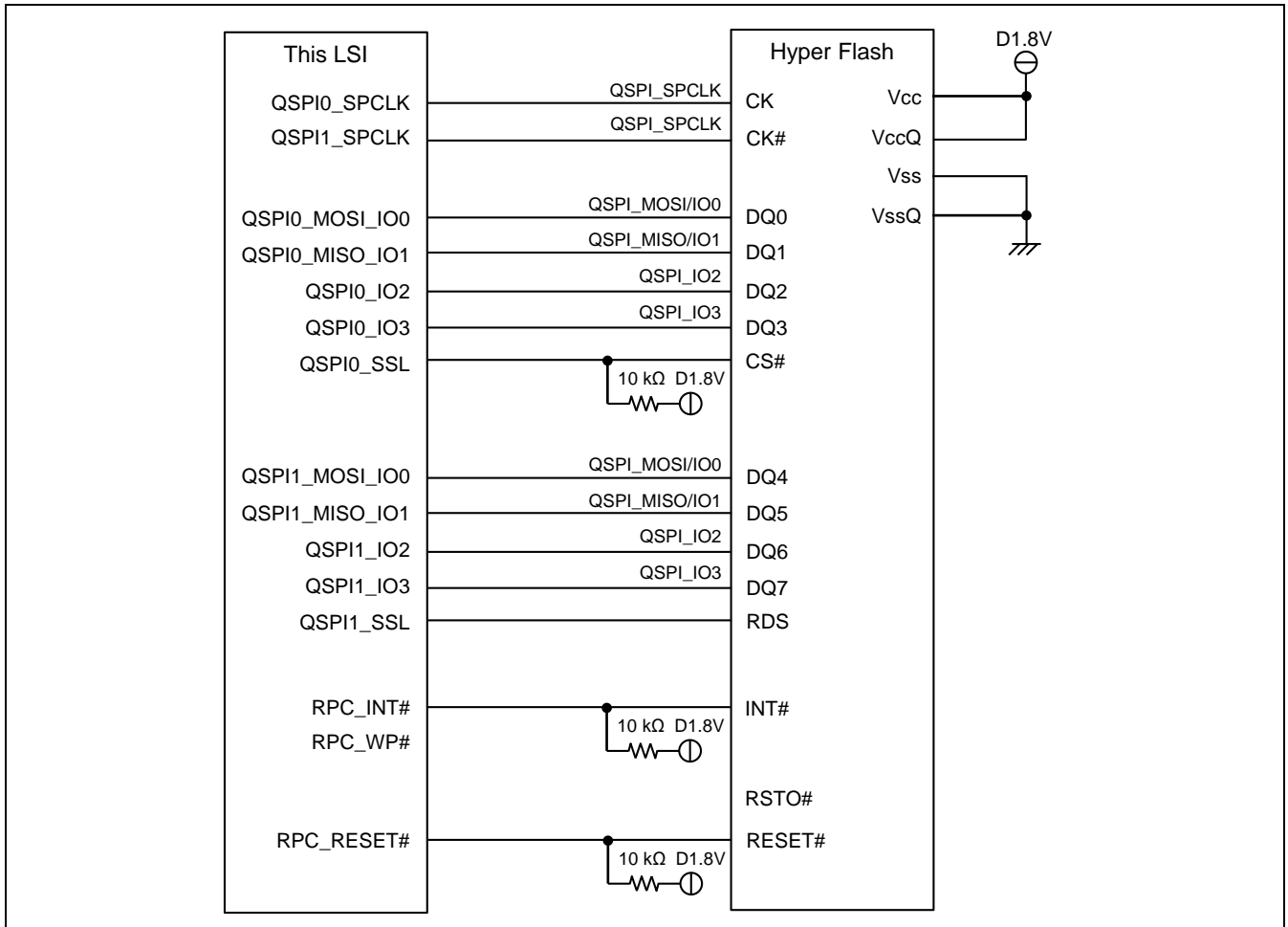


Figure 56.4 Example of System Configuration with HyperFlash Connected

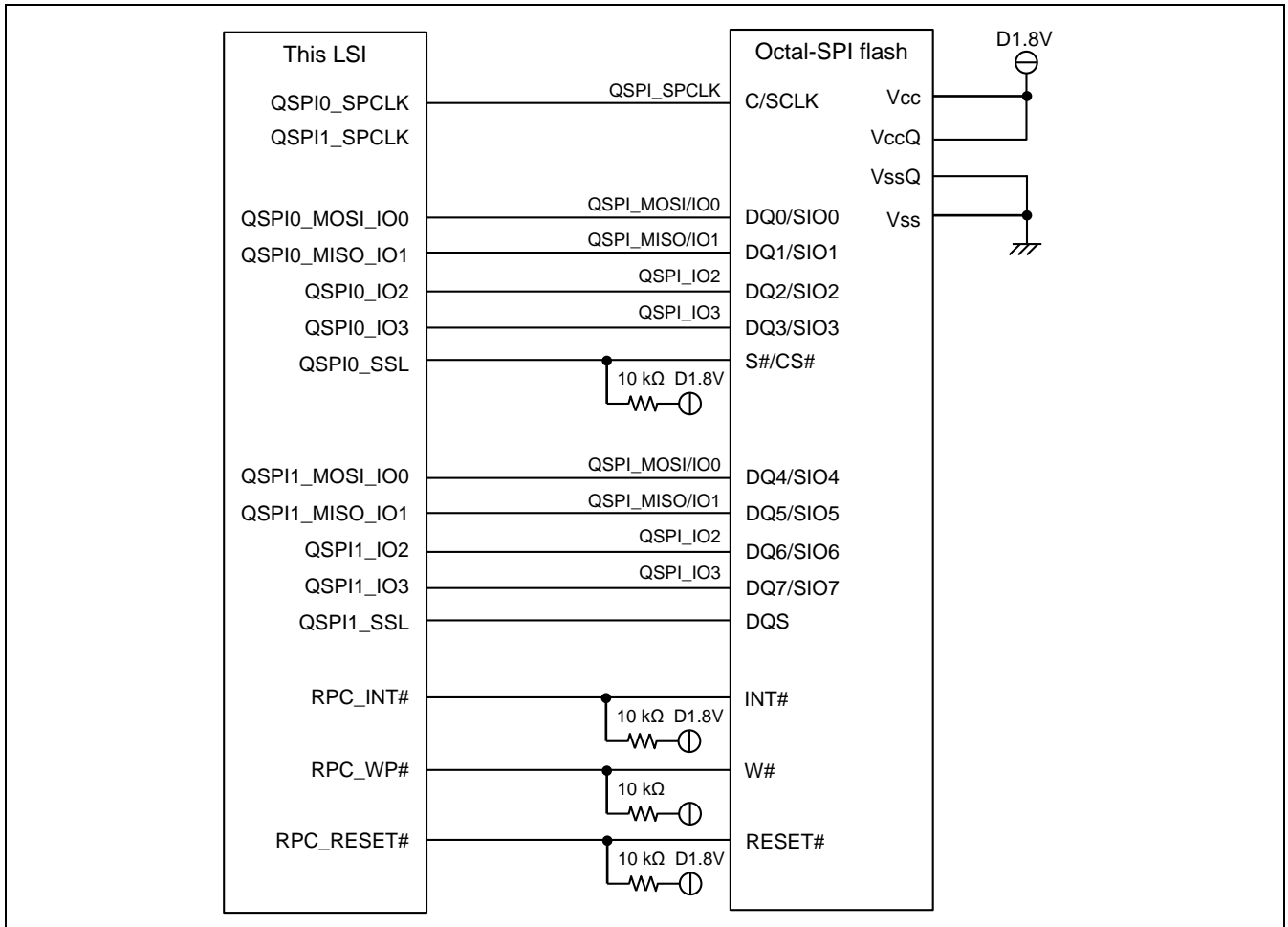


Figure 56.5 Example of System Configuration with Octal-SPI Flash Memory Device Connected

56.3.2 Address Map

In external address space read mode, the serial flash memory and HyperFlash to be connected are assigned in the SPI multi I/O bus space. The internal address space is 64MB, but with the DREAR register to specify the offset, a maximum of 4 Gbytes can be accessed when one serial flash memory is connected, and a maximum of 8 Gbytes can be accessed when two memory devices or a HyperFlash or an Octal-SPI flash memory are connected.

Table 56.4 Address Map

| Internal Address | Maximum Access Area |
|----------------------------|---|
| H'0800_0000 to H'0BFF_FFFF | Serial flash × 1: 4 Gbytes Serial flash × 2 or HyperFlash × 1 or Octal-SPI flash × 1: 8 Gbytes |

56.3.3 32-Bit Address Area in the Serial Flash Memory

Since the SPI multi I/O bus space is 64 Mbytes (with 26-bit address width), only a part of the 32-bit serial flash address area can be directly accessed. The fixed value set in the DREAR register is used as the upper-order bit value of a 32-bit address.

To output the address of the serial flash memory in 32 bits, set the ADE[3] bit in DRENr to 1, set the range of the external addresses used as the serial flash addresses to the EAC[2:0] bits in DREAR, and set the upper bit value of the 32-bit address as the fixed value to the EAV[7:0] bits in DREAR.

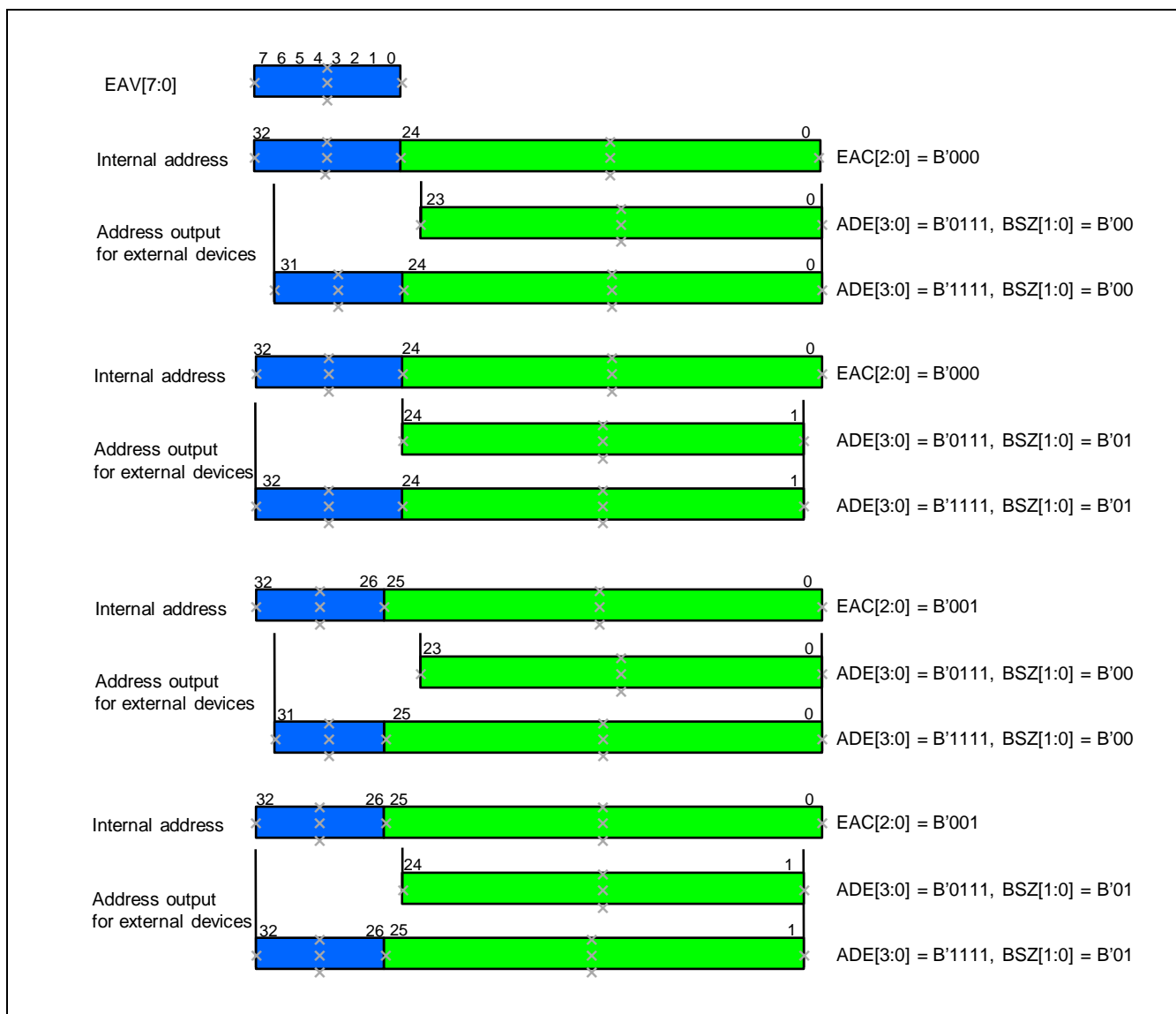


Figure 56.6 32-Bit Address Setting

Setting the ADE[3] bit in DRENr to 1 allows the serial flash address to be output using [31:0] bits. When EAC[2:0] = 000, external address bits [24:0] are valid. Set the value for [32:25] bits to EAV[7:0]. When EAC[2:0] = 001, external address bits [25:0] are valid. Set the value for [32:26] bits to EAV[7:1]. The address bits actually used for the external device are shown below.

When one serial flash memory is connected, internal address bits [31:0] are used.

When two serial flash memory devices are connected, internal address bits [32:1] are used.

When HyperFlash or an Octal SPI flash is connected, this address extension with DREN.R.ADE[3] is not available, and internal address bits [25:0] (64 Mbytes area) are used. Use manual mode when access to the bigger area than 64 Mbytes area.

56.3.4 Operating Modes

This module has two operating modes: external address space read mode and manual mode.

In external address space read mode, a read access to the SPI multi I/O bus space is converted into the data read protocol and data is received from the external device. After data acquisition from the external device, data is returned to the bus master that is the issuing source of read access. For details, see section 56.3.5, External Address Space Read Mode.

The manual mode carries out an arbitrary protocol by using the register settings. For details, see section 56.3.7, Manual Mode.

56.3.5 External Address Space Read Mode

By external address space read mode, a read access to the SPI multi I/O bus space is converted into the data read protocol. Further, the commands, optional commands, option data, and dummy cycle issued for reading external device can be modified using registers.

In external address space read mode, either normal read operation or burst read operation can be selected. The transfer format is determined based on the common control register (CMNCR), SSL delay register (SSLDR), data read control register (DRCR), data read command setting register (DRCMR), data read extended address setting register (DREAR), data read option setting register (DROPR), data read enable setting register (DRENr), data read dummy cycle setting register (DRDMCR), and data read DDR enable register (DRDRENr).

Note: The RZ/G2M V1.3 (not RZ/G2M V3.0) does not support DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit). In the following description, do not set the related operation.

(1) Normal Read Operation

When the RBE bit in DRCR is set to 0, normal read operation is performed.

In the normal read operation, the data is read from external device by reading access from bus master. After reading from external device, the QSPIn_SSL pin is negated.

The normal read operation timing with serial flash is shown in Figure 56.7, Normal Read Operation Timing.

t_1 is the time period from QSPIn_SSL pin assertion to QSPIn_SPCLK oscillation (clock delay), t_2 is the time period from transmission of the last QSPIn_SPCLK edge of a transfer to QSPIn_SSL pin negation (QSPIn_SSL negation delay), and t_3 is the time period from one transfer end to the next transfer start (next access).

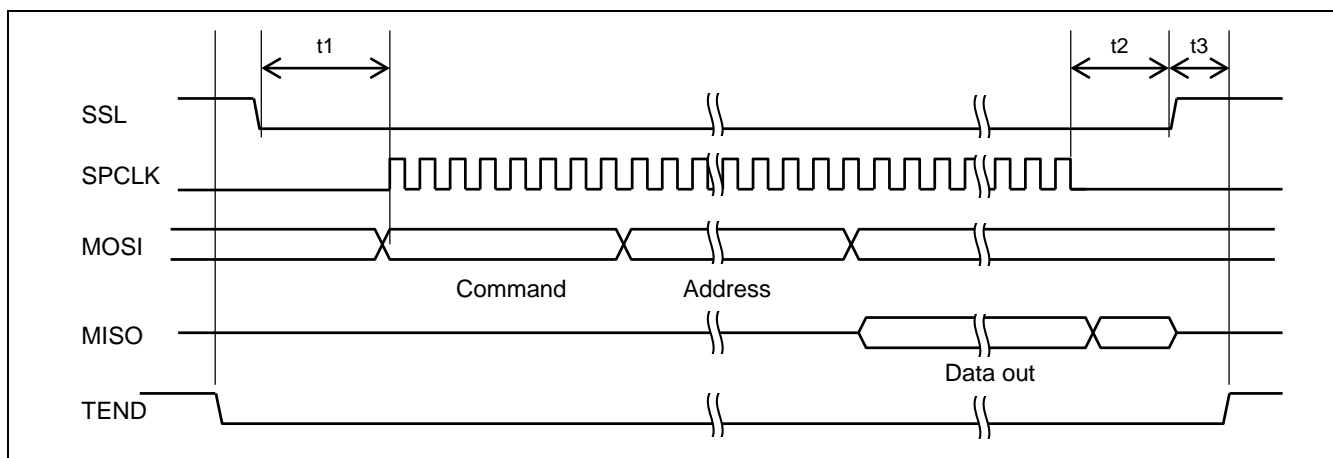


Figure 56.7 Normal Read Operation Timing

Note: t_1 to t_3 are specified in SSLDR.SCKDL, SSLDR.SLNDL and SSLDRSPNDL bits respectively.

(2) Burst Read Operation

When the RBE bit in DRCR is set to 1, burst read operation is performed.

Read cache is enabled in the burst read operation. For read cache operation, see section 56.3.6, Read Cache.

For reading access from bus master, the read cache is first referred to for checking whether the data of read address of bus master is contained in the read cache memory. When the read cache memory contains the data, the data contained in the cache memory is returned to the bus master without accessing the external device. When the read cache memory does not contain the data, burst read is performed in the external device, the read data is stored in the read cache memory and the data is returned to the bus master. The data transfer length at that time is 64 bits x RBURST[4:0] bits and the data is always read from the 64-bit boundary.

The QSPIn_SSL pin status after data transfer can be selected by using the SSLE bit in DRCR. When the SSLE bit is set to 0, the QSPIn_SSL pin is always negated after data transfer. For an operation performed when the SSLE bit is set to 1, see (3) Burst Read Operation with Automatic QSPIn_SSL Negation, just below.

A pattern diagram of this operation and a burst read operation timing diagram with serial flash when SSLE bit is set to 0 are shown in Figure 56.8 and 56.9.

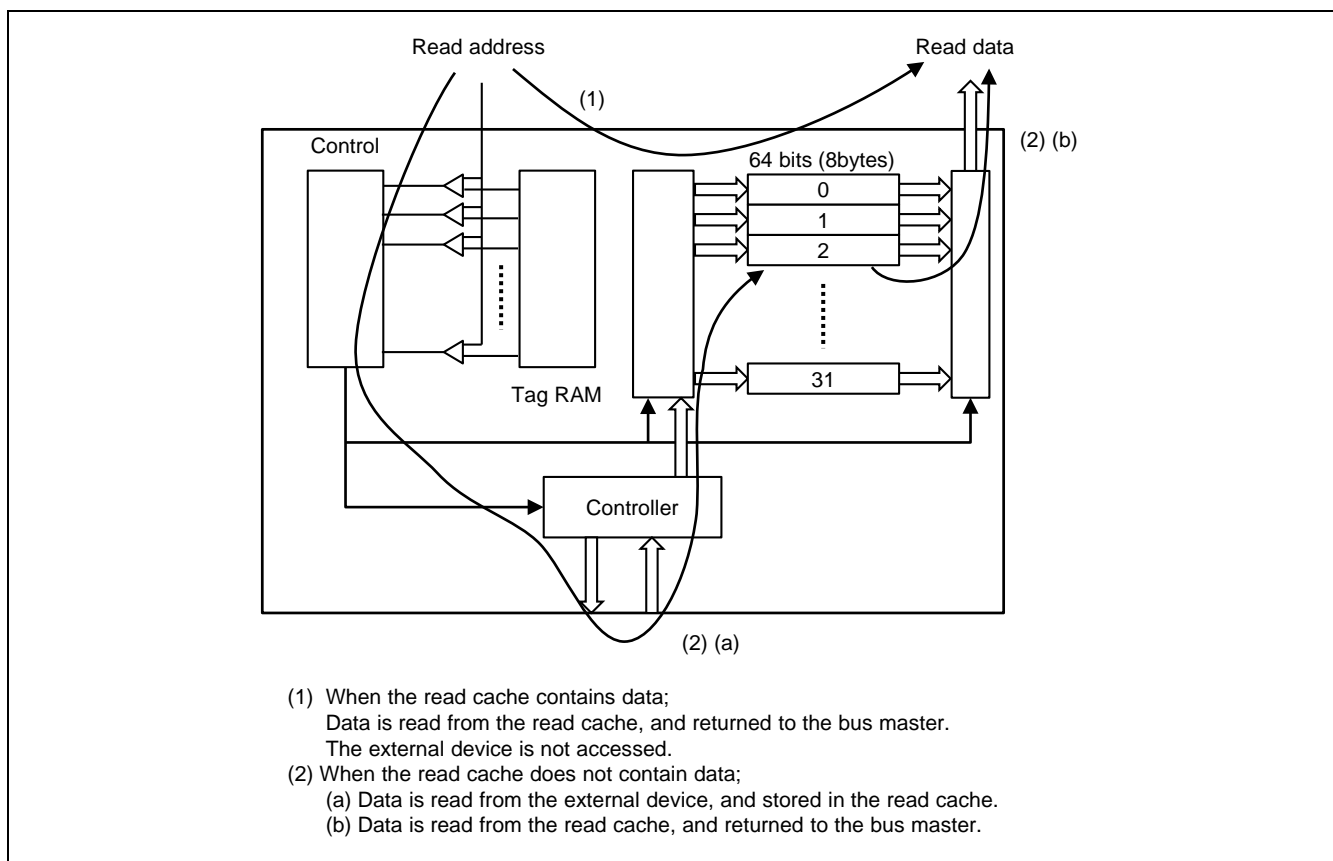


Figure 56.8 Read Operation in Burst Read Operation

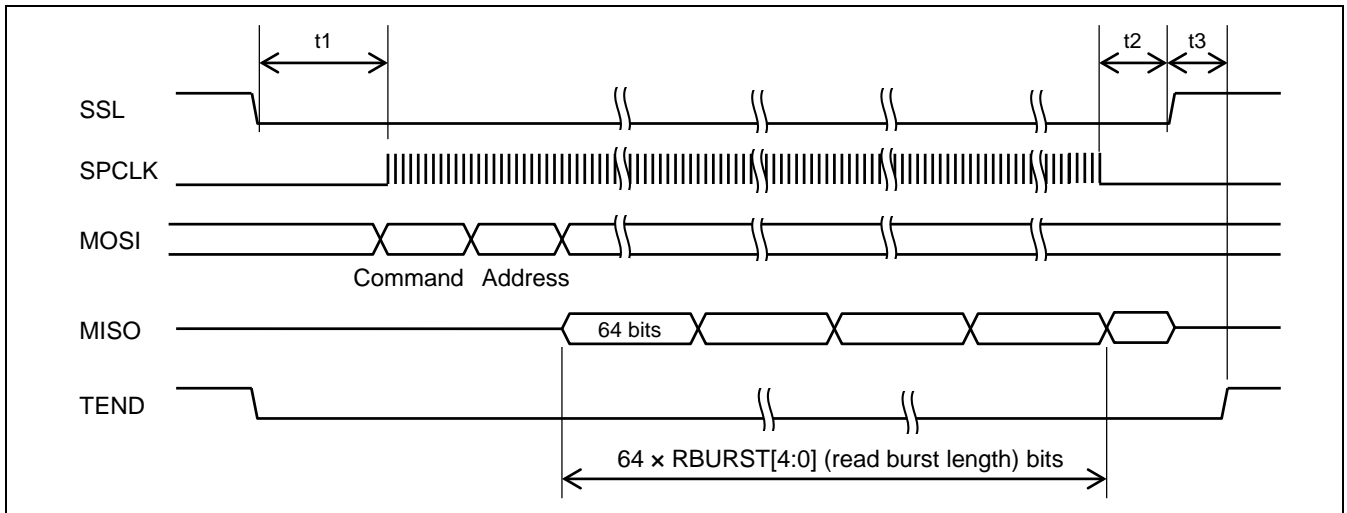


Figure 56.9 Read Timing in Burst Read Operation (SSLE Bit = 0)

Note: t_1 to t_3 are specified in SSLDR.SCKDL, SSLDR.SLNDL and SSLDRSPNDL bits respectively.

(3) Burst Read Operation with Automatic QSPIn_SSL Negation

When SSLE bit in DRCCR is set to 1, this module does not negate the QSPIn_SSL pin after the burst read transfer on serial flash connection. When accessing the next time, if the address is continuous with the previous read address, the burst read operation is performed without issuing the command, optional command, address, option data, or dummy cycle. If the address is not continuous with the previous read address, the QSPIn_SSL pin is once negated and the burst read operation is performed after issuing the command, optional command, address, option data, or dummy cycle.

Burst read timing diagrams for continuous address and non-continuous address with serial flash are shown in Figure 56.10 and 56.11.

For the next access after negation of the QSPIn_SSL with the SSLN bit in DRCCR with this operation, read SSLF = 0 in CMNSR to confirm that the QSPIn_SSL has been negated. This function is only supported to connect to serial flash memory.

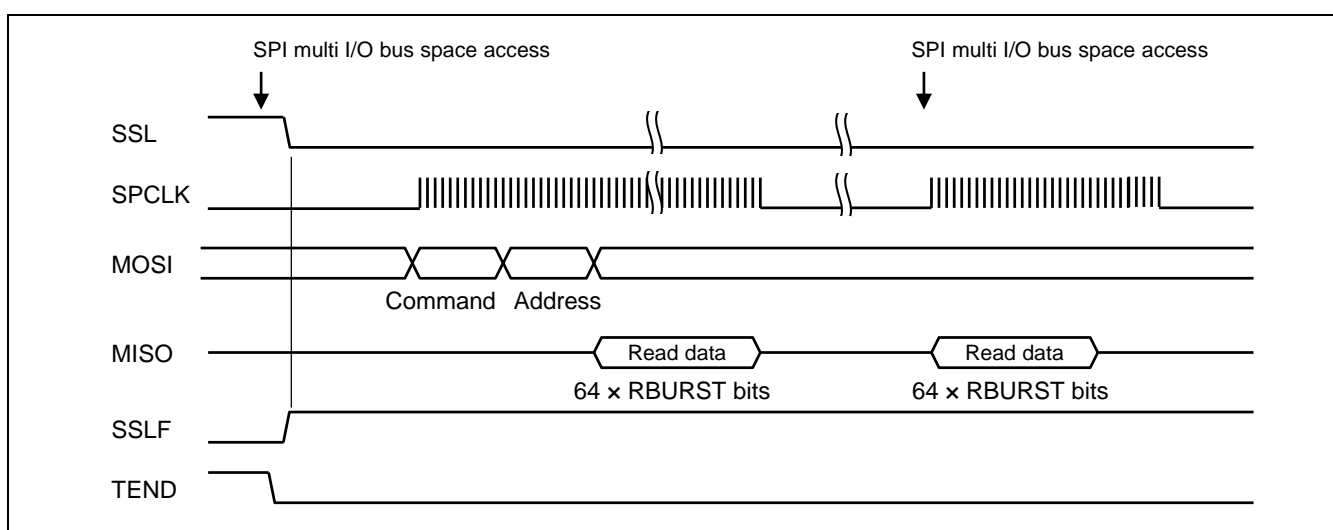


Figure 56.10 Burst Read Timing for Continuous Address (SSLE Bit = 1)

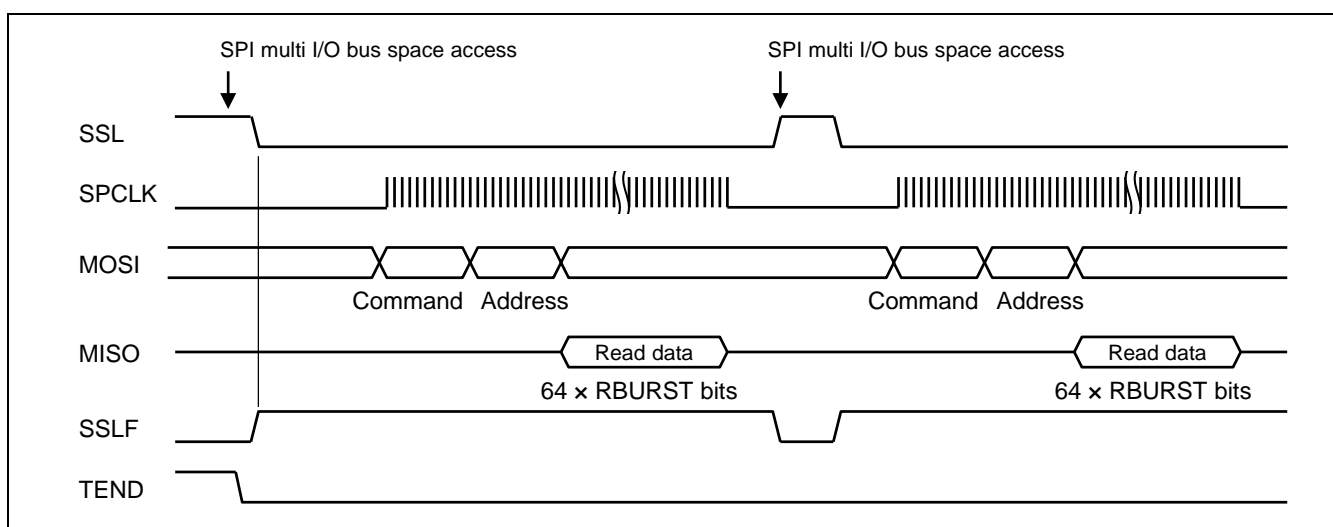


Figure 56.11 Burst Read Timing for Non-Continuous Address (SSLE Bit = 1)

(4) Initial Setting Flow

An example of an initial setting flow in external address space read mode is shown in Figure 56.12.

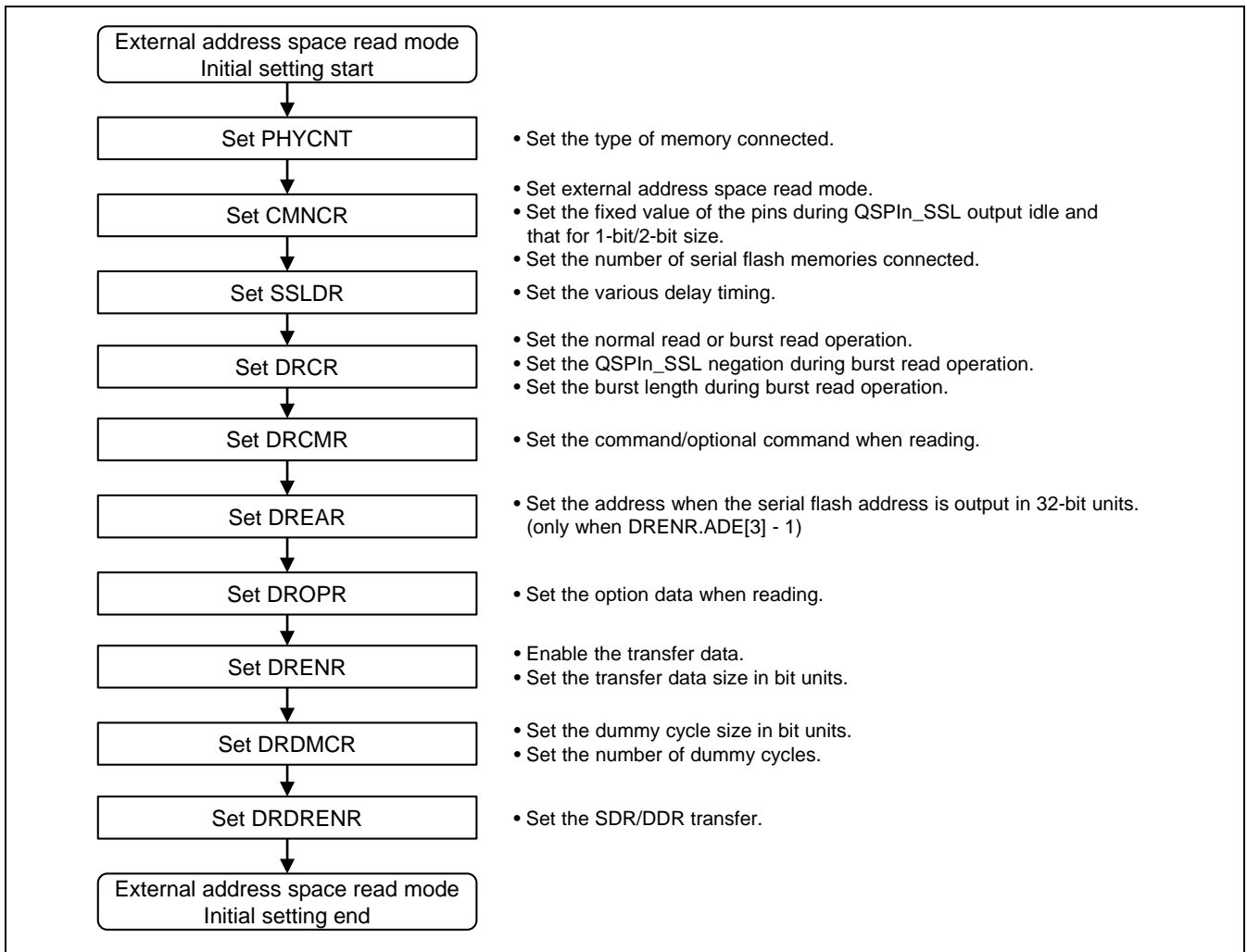


Figure 56.12 Example of Initial Setting Flow in External Address Space Read Mode

56.3.6 Read Cache

This module has a simple built-in read cache. The read cache can be used during external address space read mode and burst read operation. The read cache is configured with a line size of 64 bits and 32 entries.

Read cache configuration is shown in Figure 56.13.

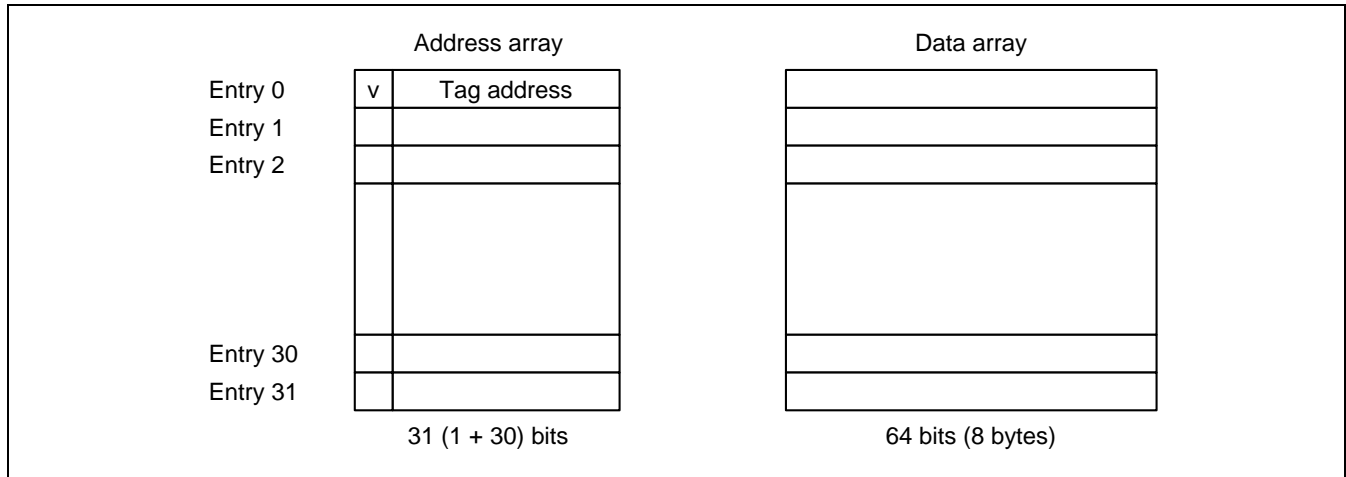


Figure 56.13 Read Cache Configuration

(1) Address Array

The V bit indicates whether the entry data is valid. When the V bit is 1, the data is valid and when V bit is 0, the data is invalid.

The tag address bits hold the address used for the serial flash memory. Address bits 32 to 3 are used for the purpose. Address bits 23 to 3 are enabled when address output is 24 bits and one serial flash memory is connected; and address bits 24 to 3 are enabled when two serial flash memory devices are connected.

Address bits 31 to 3 are enabled when address output is 32 bits and one serial flash memory is connected; and address bits 32 to 3 are enabled when two serial flash memory devices are connected.

(2) Data Array

It retains the 64-bit read data. Registration in the read cache is performed in line units.

(3) Read Operation

In case of read-hit, data is read from the read cache. In case of read-miss, after the $64 \times \text{RBURST}$ (read burst length) data is read from the serial flash memory and the read cache is updated, the data is returned to the bus master.

(4) Data Replacement

The write pointer is used to update data. In case of read-miss, the RBURST (read burst length) portion data is replaced starting at the entry specified by the write pointer. In other words, the data is replaced in the storage order of the data. Whether data is referred to or not will not affect the replacement order of data.

56.3.7 Manual Mode

This module can carry out an arbitrary serial transfer operation by using the register settings.

The transfer format is determined based on the common control register (CMNCR), SSL delay register (SSLDR), manual mode control register (SMCR), manual mode command setting register (SMCMR), manual mode address setting register (SMADR), manual mode option setting register (SMOPR), and manual mode enable setting register (SMENR), manual mode read data register (SMRDR), manual mode write data register (SMWDR), manual mode dummy cycle setting register (SMDMCR), and manual mode DDR enable register (SMDRENDR).

Manual mode can be used for reading the status of the serial flash memory and writing to the serial flash memory.

In this mode, one transfer refers to the operation from when the SPIE bit in SMCR is set to 1 to when the TEND bit is set to 1.

Note: The RZ/G2M V1.3 (not RZ/G2M V3.0) does not support DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit). In the following description, do not set the related operation.

(1) Transfer Start

The transfer of data is started in the set transfer format by setting the SPIE bit in SMCR to 1. When write operation is enabled, the manual mode write data register is transmitted to the serial flash memory. When read operation is enabled, data read from the serial flash memory is stored into the manual mode read data register.

The operation timing of serial transfer is shown in Figure 56.14.

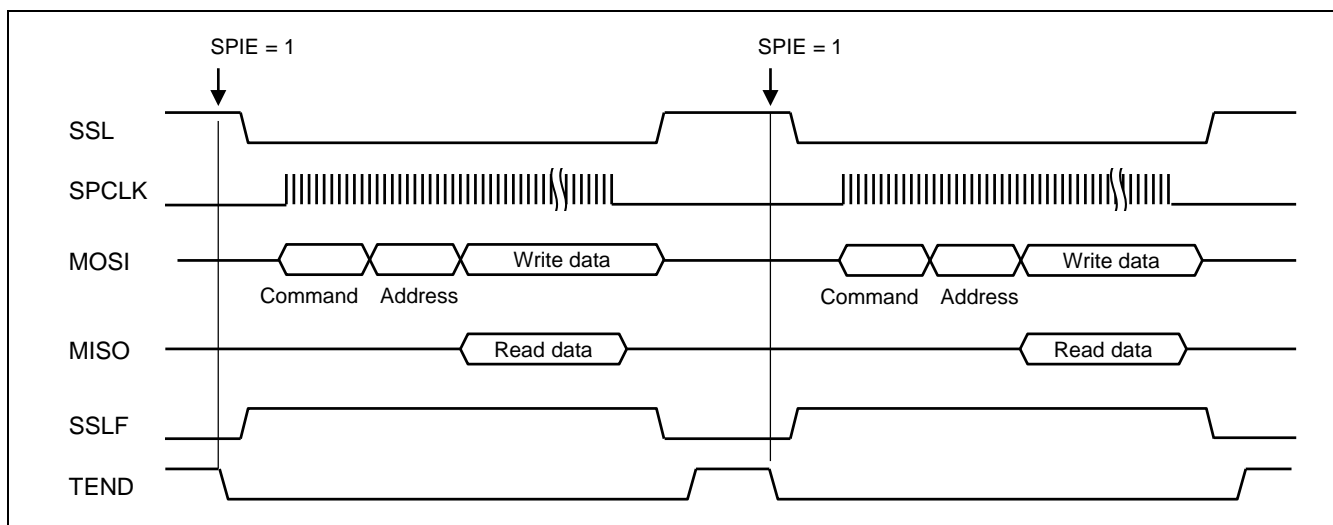


Figure 56.14 Operation Timing of Serial Transfer

(2) Read/Write Enable

Read operation: Data can be read by setting the SPIRE bit in SMCR to 1. The read data is stored into SMRDR.

Write operation: Data can be written by setting the SPIWE bit in SMCR to 1. The data stored in SMWDR is output.

When the data size is set to 1 bit using the SPIDB[1:0] bits in SMENR, data can be transmitted and received by setting the SPIRE and SPIWE bits to 1. However, when the data size is set to 4 bits by using the SPIDB[1:0] bits, only one of the SPIRE and SPIWE bits should be enabled. The operation is not guaranteed if both the bits are enabled.

(3) Retention of QSPIn_SSL Pin Assertion

By setting the SSLKP bit in SMCR to 1, assertion of the QSPIn_SSL pin can be continued till the next transfer. With this function, the transfer can be carried out continuously with the QSPIn_SSL kept in the asserted state.

The data transfer timing using the SSLKP bit is shown in Figure 56.15.

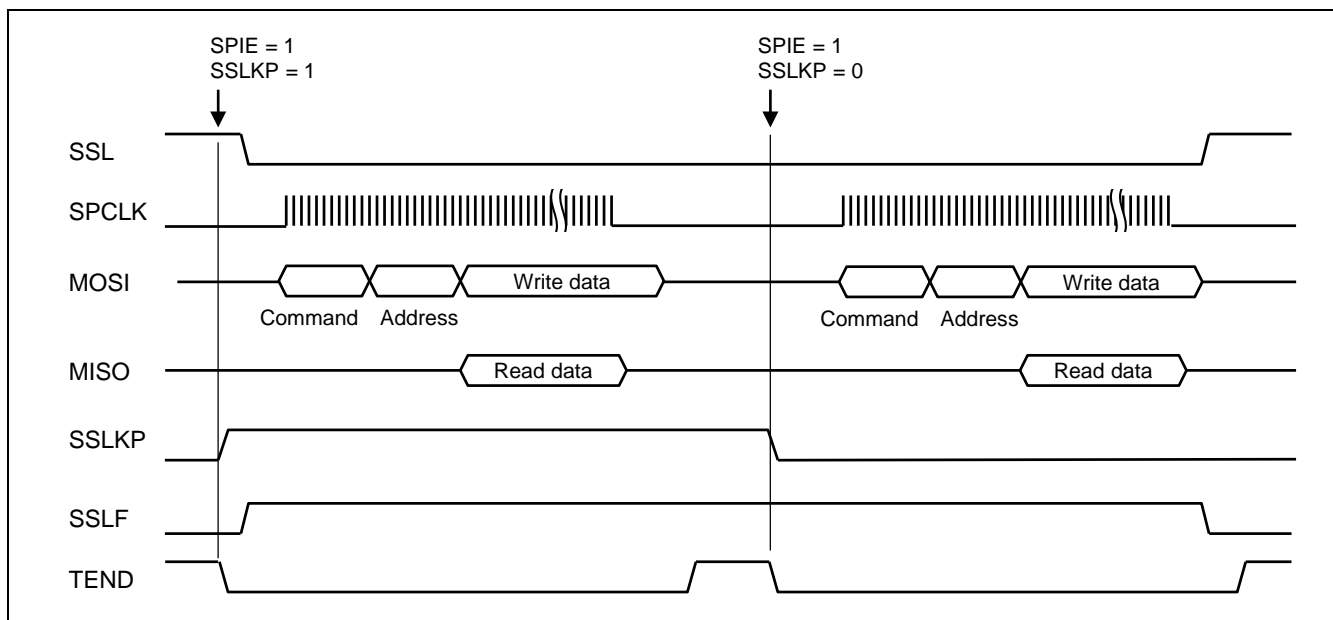


Figure 56.15 Data Transfer Timing Chart Using the SSLKP Bit

(4) Initial Setting Flow

An example of an initial setting flow in manual mode is shown in Figure 56.16.

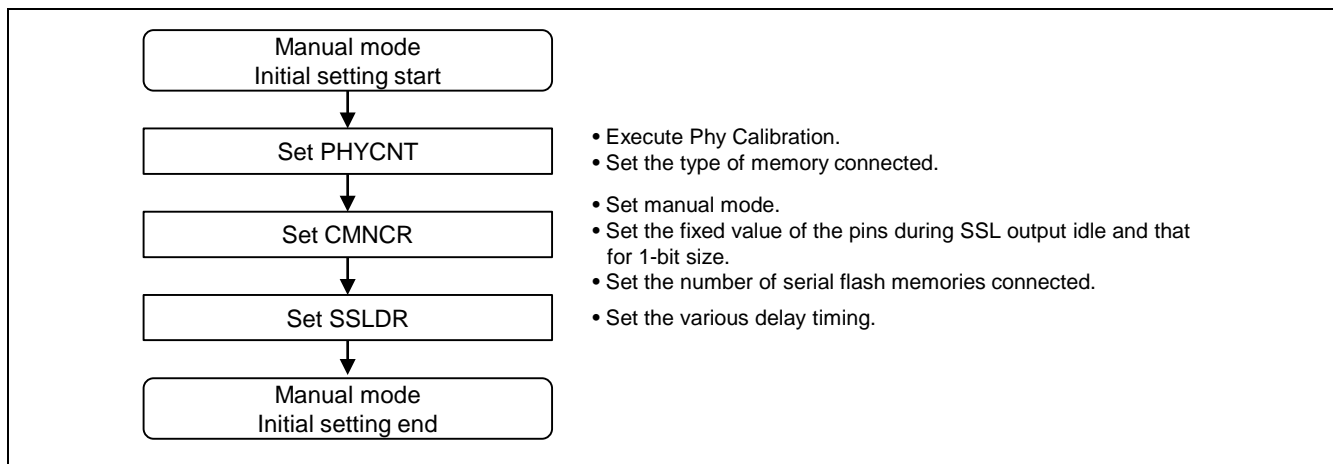


Figure 56.16 Example of Initial Setting Flow in Manual Operating Mode

(5) Data Transfer Setting Flow

An example of a data transfer setting flow in manual mode is shown in Figure 56.17.

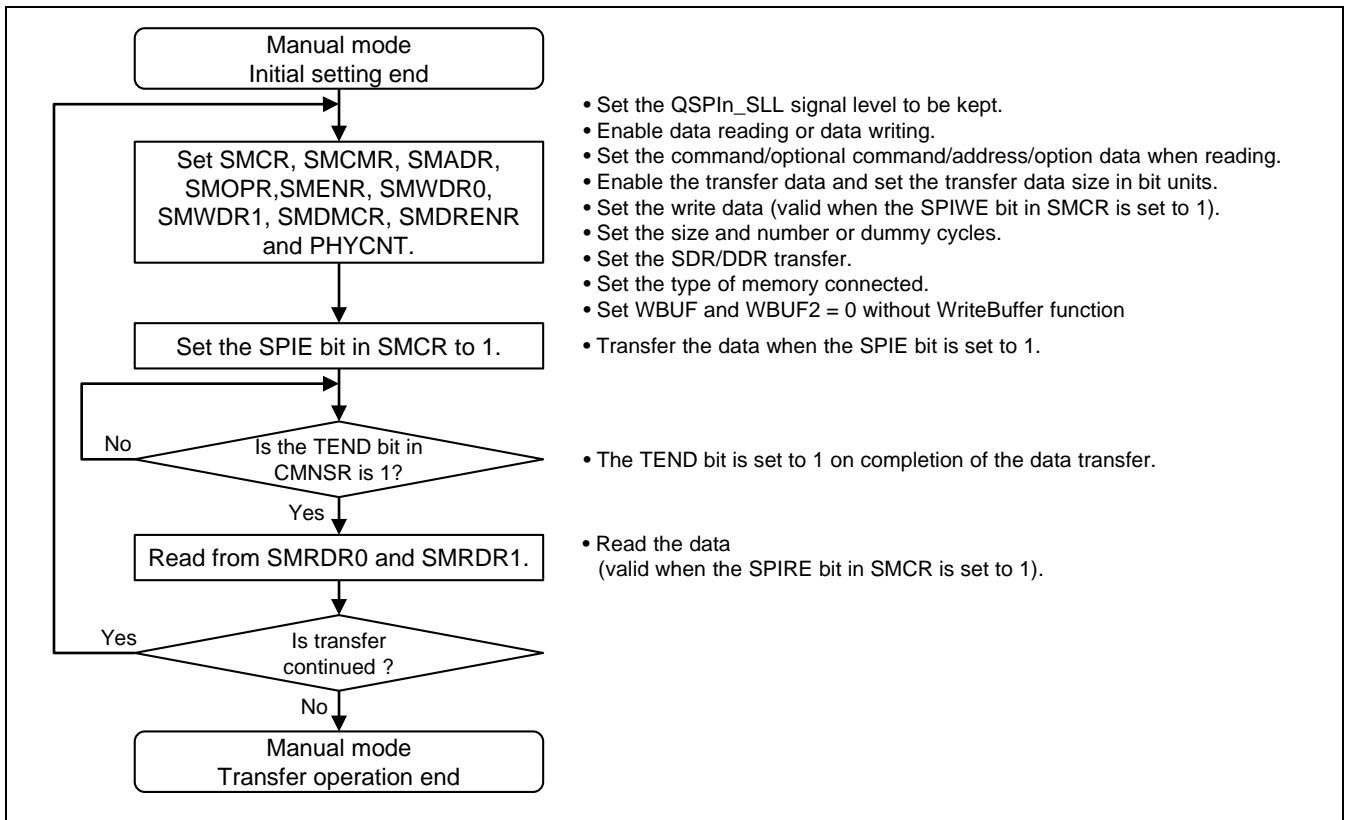


Figure 56.17 Example of Data Transfer Setting Flow in Manual Operating Mode

56.3.8 Command Sequence

This module handles input and output data in the order of command, optional command, address, option data, dummy cycle, and data.

Note: The RZ/G2M V1.3 (not RZ/G2M V3.0) does not support DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit). In the following description, do not set the related operation.

(1) Data Registers

Table 56.5 and Table 56.6 shows the input and output data.

Table 56.5 Data Registers (serial flash)

| Data | External Address Space | Read Operation | Manual Operation |
|------------------------------|--|--|---|
| Command (8 bits) | CMD[7:0] bits in DRCCMR | | CMD[7:0] bits in SMCMDR |
| Optional Command (8 bits) | OCMD[7:0] bits in DRCCMR | | OCMD[7:0] bits in SMCMDR |
| Address (32/24 bits) | BSZ[1:0] = 00 (one serial flash memory connected) | 32 bits: DREAR.EAV[7:1 to 0] bits + lower [25 to 24:0] bits of the read address. | 32 bits: ADR[31:0] bits in SMADR. |
| | | 24 bits: Lower [23:0] bits of the read address | 24 bits: ADR[23:0] bits in SMADR |
| | BSZ[1:0] = 01 (two serial flash memory devices connected) | 32bits: DREAR.EAV[7:1 to 0] bits + lower [25 to 24:1] bits of the read address | |
| | | 24bits: Lower [24:1] bits of the read address | |
| Option data (8 bits × 4) | DROPR | | SMOPR |
| Dummy cycle (1 to 20 cycles) | DRDMC | | SMDMC register (only when read) |
| Transfer Data | Normal read: 8, 16, and 32 bits, Burst read: 64 x RBURST bits | | Read: SMRDR0 and SMRDR1 Write: SMWDR0 and SMRDR1 |

Table 56.6 Data Registers (HyperFlash)

| Data | External Address Space | Read Operation | Manual Operation |
|------------------------------|--|----------------|---|
| CA0 (47-40) | CMD[7:5] bits in DRCCMR + [31:27] bits of the read address | | CMD[7:5] bits in SMCMDR + ADR[31:27] bits in SMADR |
| CA0 (39-32) | [26:19] bits of the read address | | ADR[26:19] bits in SMADR |
| CA0 (31-24) | [18:11] bits of the read address | | ADR[18:11] bits in SMADR |
| CA0 (23-16) | [10:3] bits of the read address | | ADR[10:3] bits in SMADR |
| CA0 (15-8) | DROPR[15:8] | | SMOPR[15:8] |
| CA0 (7-0) | DROPR[7:3] + [2:0] bits of the read address | | SMOPR[7:3] + ADR[2:0] bits in SMADR |
| Dummy cycle (1 to 20 cycles) | DRDMC | | SMDMC register |
| Transfer Data | Normal read: 8, 16, and 32 bits, Burst read: 64 x RBURST bits | | Read: SMRDR0 and SMRDR1 Write: SMWDR0 and SMRDR1 |

(2) Data Enable

In external address space read mode, transfer enable or disable of the command, optional command, address, option data, and dummy cycle can be controlled with the CDE, OCDE, ADE[3:0], OPDE[3:0], and DME bits in DRENr, respectively. The size and number of dummy cycles can be controlled with the data read dummy cycle setting register (DRDMCR).

Similarly, in manual mode, enable or disable of the command, optional command, address, option data, dummy cycle, and transfer data can be controlled with the CDE, OCDE, ADE[3:0], OPDE[3:0], DME, and SPIDE[3:0] bits in SMENr, respectively. However, disabling all the above parameters is prohibited in manual mode. At least one of them except dummy cycle must be enabled. The size and number of dummy cycles can be controlled with the manual mode dummy cycle setting register (SMDMCR).

For the address and option data in external address space read mode; and the address, option data, and transfer data in manual mode, the enable bit setting allowed is determined according to the transfer data size. For the allowed setting combinations of the enable bits and transfer data size, refer to the description of the pertinent register.

If data is disabled, that data is skipped, and input and output of the next data is carried out. The command, optional command, address, and option data are always output. During dummy cycles, the state of the used pins is Hi-Z. In external address space read mode, data is always input; and in manual mode, input and output of data is determined based on the settings of the SPIRE and SPIWE bits in SMCR.

There are some restrictions on dummy cycle insertion. Refer to the description of the DME bits in DRENr and SMENr for details.

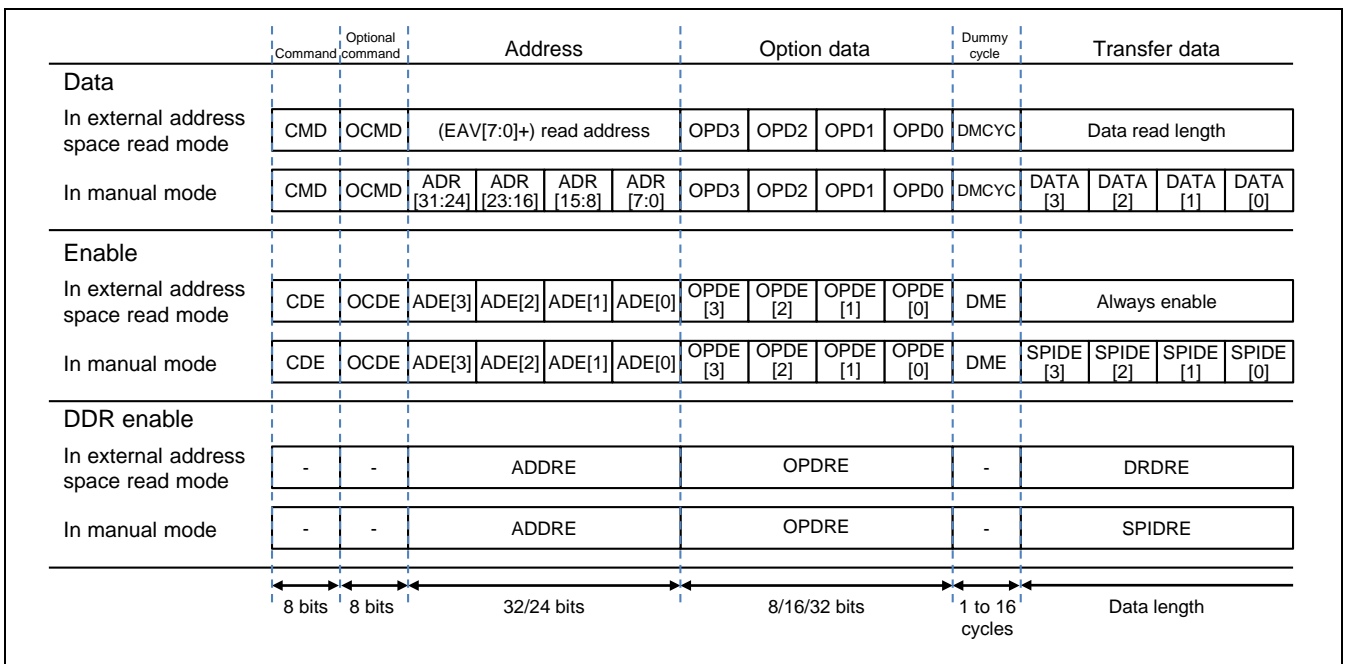


Figure 56.18 The relationship between Data and Enable (serial flash)

When HyperFlash is connected to this LSI, specify enable bits in CDE, OCDE, ADE[3:0], OPDE[3:0], and DME bits in DRENr and HYPE[2:0], ADDRE, OPDRE and DRDRE in DRDRENr in external address space read mode, and CDE, OCDE, ADE[3:0], OPDE[3:0], DME and SPIDE[3:0] in SMENr and HYPE[2:0], ADDRE, OPDRE and SPIDRE in SMDRENr manual mode.

Table 56.7 Enable Registers (HyperFlash)

| External Address Space Read Operation | | | Manual Operation | | |
|---------------------------------------|-----------|---------------|------------------|------------|---|
| Register | Bits | Setting Value | Register | Bits | Setting Value |
| DRENr | CDE | B'1 | SMENr | CDE | B'1 |
| | OCDE | B'1 | | OCDE | B'1 |
| | ADE[3:0] | B'0100 | | ADE[3:0] | B'0100 |
| | OPDE[3:0] | B'0000 | | OPDE[3:0] | B'0000 |
| | DME | B'1 | | DME | Read: B'1 Write: B'0 |
| DRDRENr | HYPE[2:0] | B'101 | SMDRENr | SPIDE[3:0] | 16bit: B'1000 32bit: B'1100 64bit: B'1111 |
| | ADDRE | B'1 | | HYPE[2:0] | B'101 |
| | OPDRE | B'0 | | ADDRE | B'1 |
| | DRDRE | B'1 | | OPDRE | B'0 |
| | | | | SPIDRE | B'1 |

(3) Bit Size

In external address space read mode, the size of the command, optional command, address, option data, and the read data in bit units is respectively controlled with the CDB[1:0], OCDB[1:0], ADB[1:0], OPDB[1:0], and DRDB[1:0] bits in DRENr. The size of the dummy cycle in bit units is also controlled with the DMDB[1:0] bits in DRDMCR.

Similarly, in manual mode, the size of the command, optional command, address, option data, and read write data in bit units is controlled with the CDB[1:0], OCDB[1:0], ADB[1:0], OPDB[1:0], and SPIDB[1:0] bits in SMENr. The size of the dummy cycle in bit units is also controlled with the DMDB[1:0] bits in SMDMCR.

(a) 1-bit Size

When the size is set to 1 bit, QSPI0_MISO_IO1 and QSPI1_MISO_IO1 pins will be the input pins and QSPI0_MOSI_IO0 and QSPI1_MOSI_IO0 pins will be the output pins. QSPI0_IO2, QSPI1_IO2, QSPI0_IO3, and QSPI1_IO3 pins are not used.

Figure 56.19 and 56.20 show the examples of the transfer format.

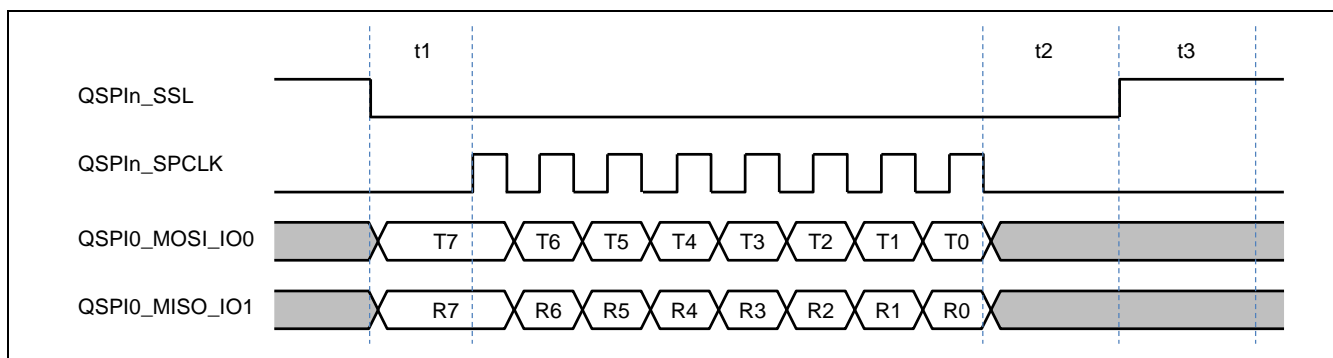


Figure 56.19 Example of Transfer Format with 1-Bit Data Size and One Serial Flash Memory Connected

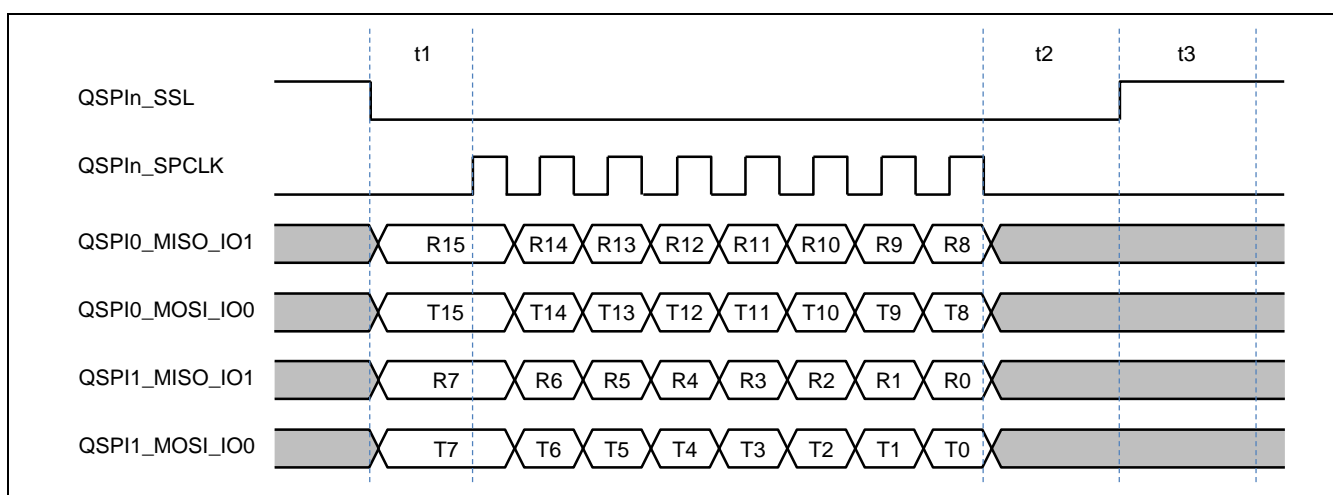


Figure 56.20 Example of Transfer Format with 1-Bit Data Size and Two Serial Flash Memory Devices Connected

(b) 4-bit Size (serial flash)

When the size is set to 4-bit, QSPI0_MOSI_IO0, QSPI1_MOSI_IO0, QSPI0_MISO_IO1, QSPI1_MISO_IO1, QSPI0_IO2, QSPI1_IO2, QSPI0_IO3, and QSPI1_IO3 pins will be either the input pins or the output pins. Figure 56.21 and 56.22 show the examples of the transfer format.

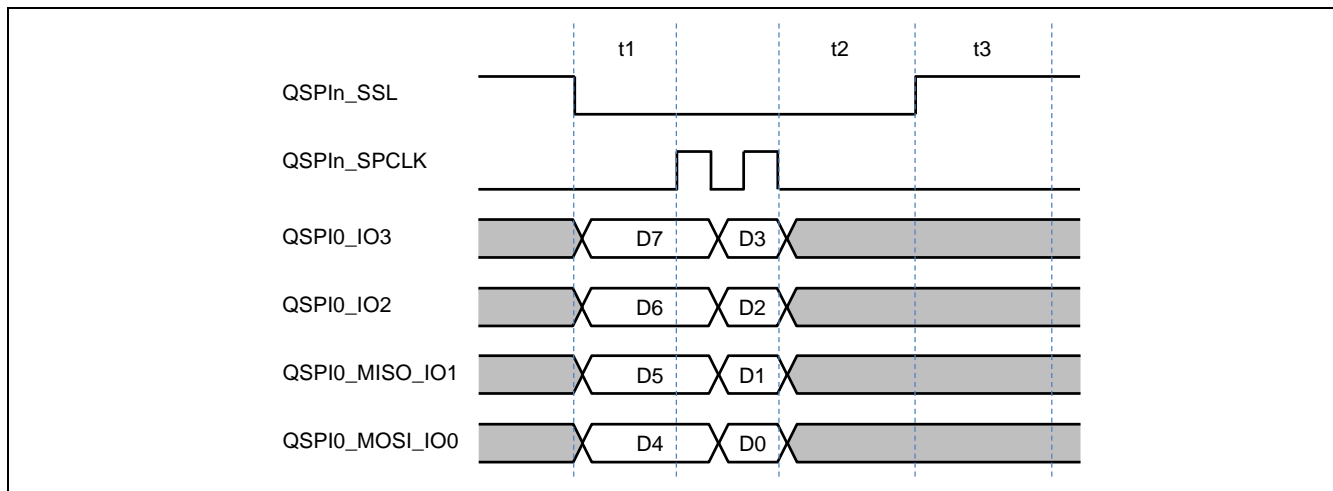


Figure 56.21 Example of Transfer Format with 4-Bit Data Size and One Serial Flash Memory Connected

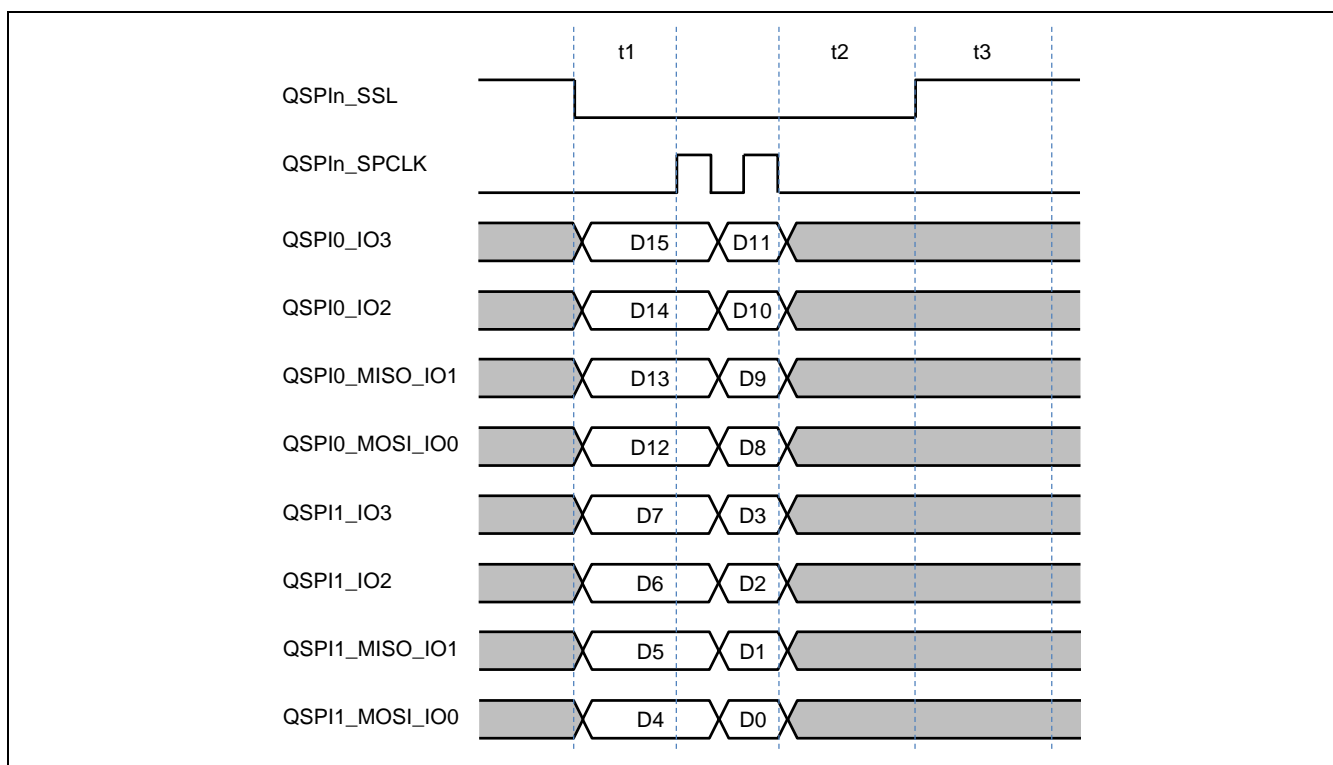


Figure 56.22 Example of Transfer Format with 4-Bit Data Size and Two Serial Flash Memory Devices Connected

(c) 4-bit Size (Hyperflash)

When HyperFlash is connected to this LSI, QSPI0_SPCLK and QSPI1_SPCLK will be differential clock, QSPI1_SSL will be input pin for RDS (Read Data Strobe) and QSPI0_MOSI_IO0, QSPI1_MOSI_IO0, QSPI0_MISO_IO1, QSPI1_MISO_IO1, QSPI0_IO2, QSPI1_IO2, QSPI0_IO3, and QSPI1_IO3 pins will be either the input pins or the output pins. Figure 56.23 shows the command/address sequence, and Figure 56.24 shows example of read data. In manual mode, the read data alignment is sequential, {DQ3[7:0], DQ2[7:0], DQ1[7:0], DQ0[7:0]} is stored in SMRDR1[31:0] and {DQ7[7:0], DQ6[7:0], DQ5[7:0], DQ4[7:0]} is stored in SMRDR0[31:0]. In external address space read mode, the read data is stored in little endian. For example, the 64-byte read data on RPC-IF area (H'0800_0000 ~ H'0BFF_FFFF) is {DQ7[31:0], DQ6[7:0], DQ5[7:0], DQ4[7:0], DQ3[7:0], DQ2[7:0], DQ1[7:0], DQ0[7:0]}.

Figure 56.25 shows example of write data(64bit). In manual mode, the write data alignment is sequential, the data in SMWDR1[7:0] is on DQ0, SMWDR1[15:8] is on DQ1, ..., SMWDR0[31:24] is on DQ7. Figure 56.26 shows example of write data(16 bit). In manual mode, the write data alignment is sequential, the data in SMWDR0[23:16] is on DQ6, SMWDR0[31:24] is on DQ7. In Write Buffer mode, the write data alignment is little endian. The data stored in Write Buffer is output from H'EE20_8000 to H'EE20_80FF sequentially.

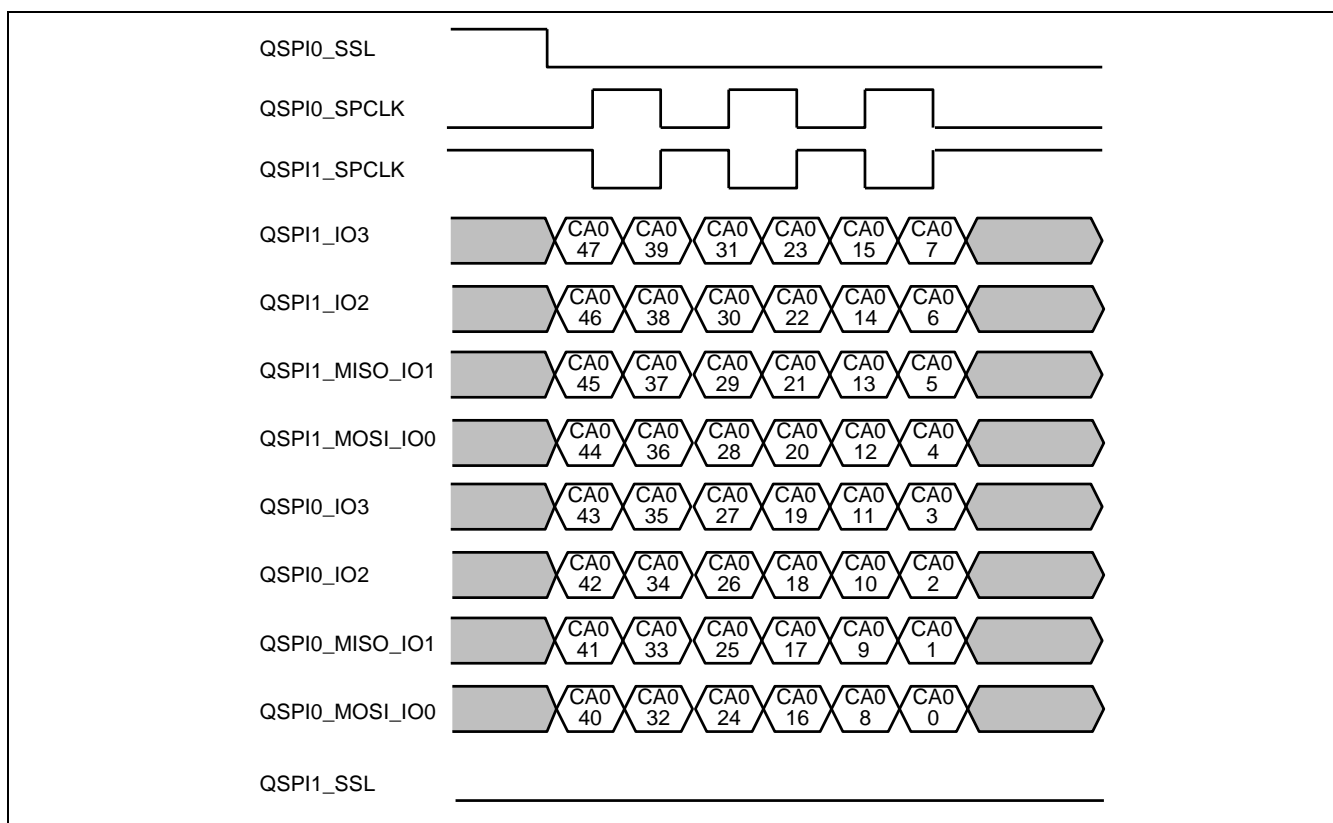


Figure 56.23 Example of Transfer Format in Command/Address phase with HyperFlash

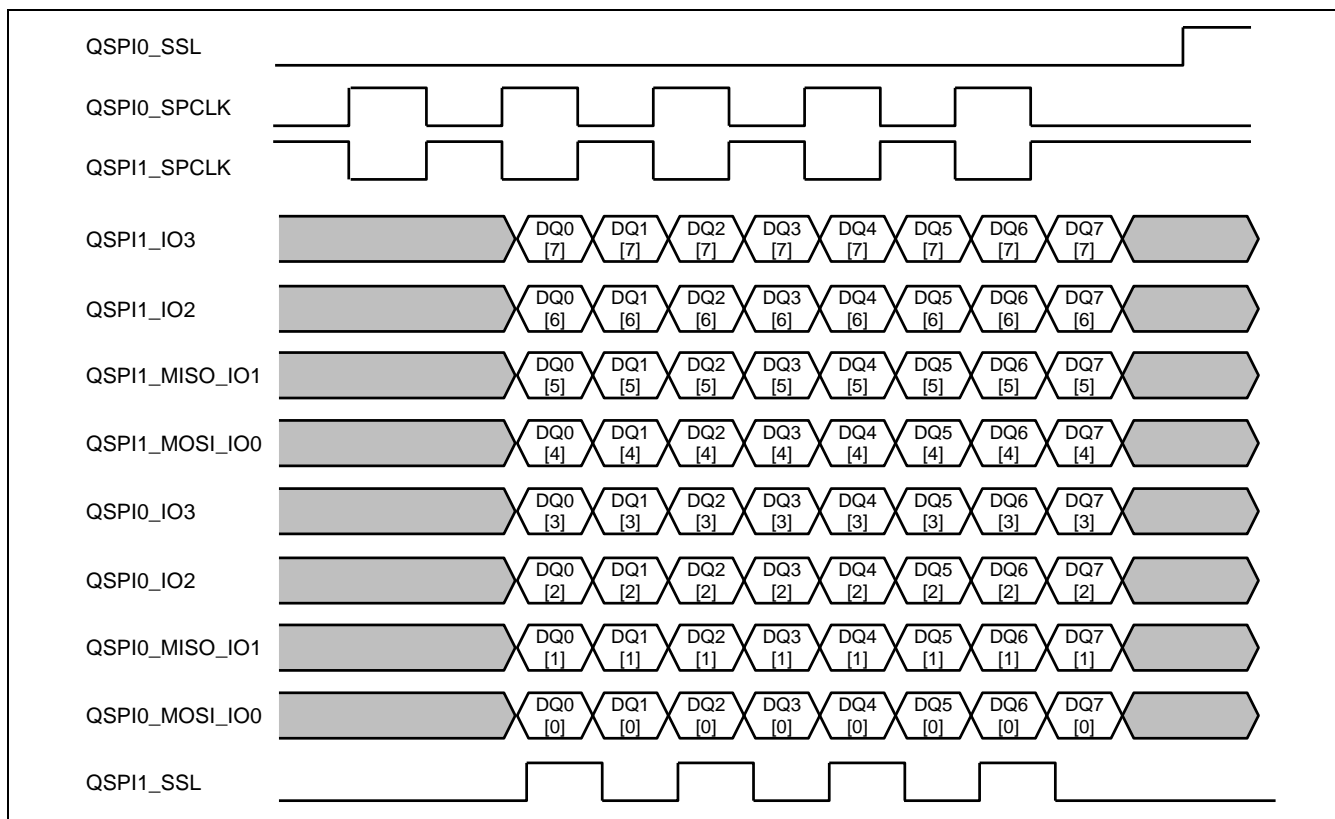


Figure 56.24 Example of Transfer Format in Read phase with HyperFlash

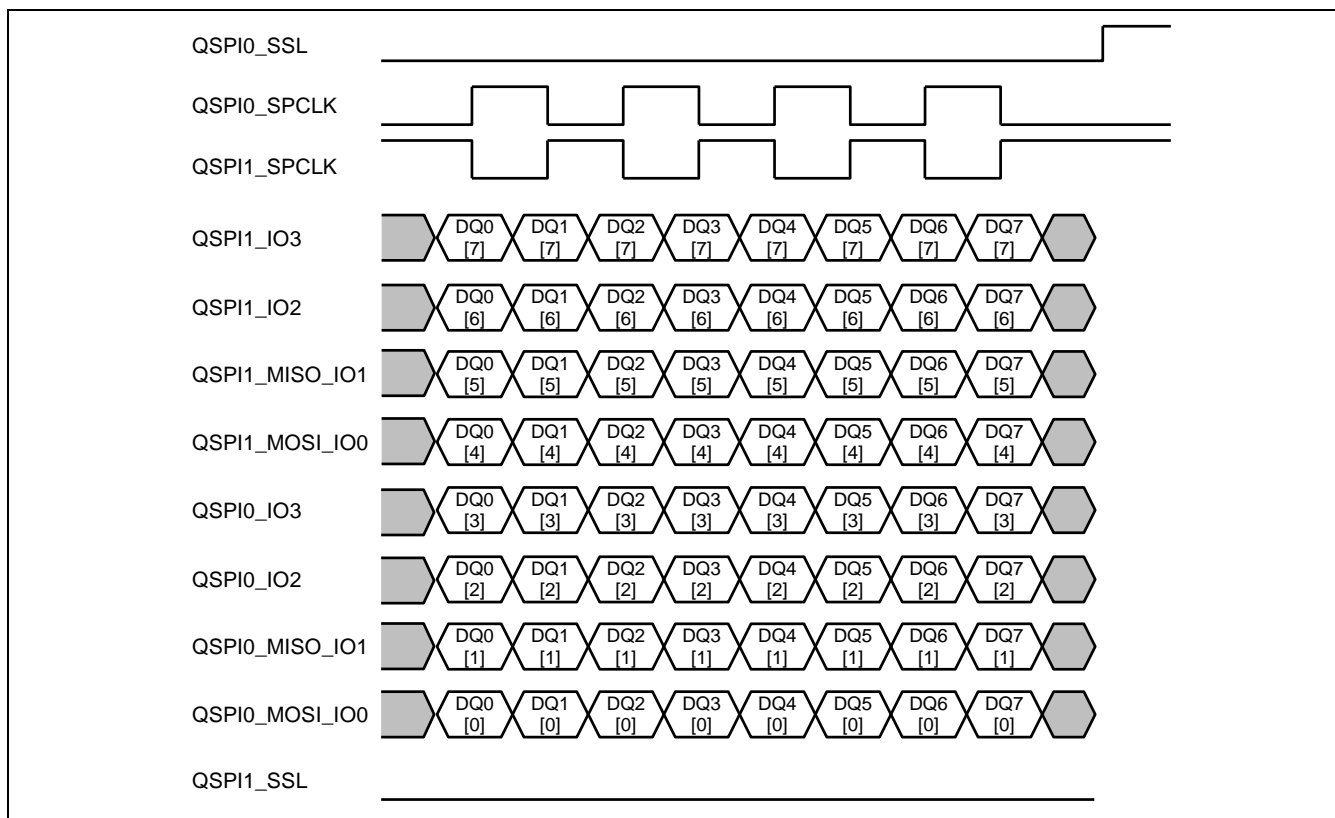


Figure 56.25 Example of Transfer Format in Write phase with HyperFlash (64 bit)

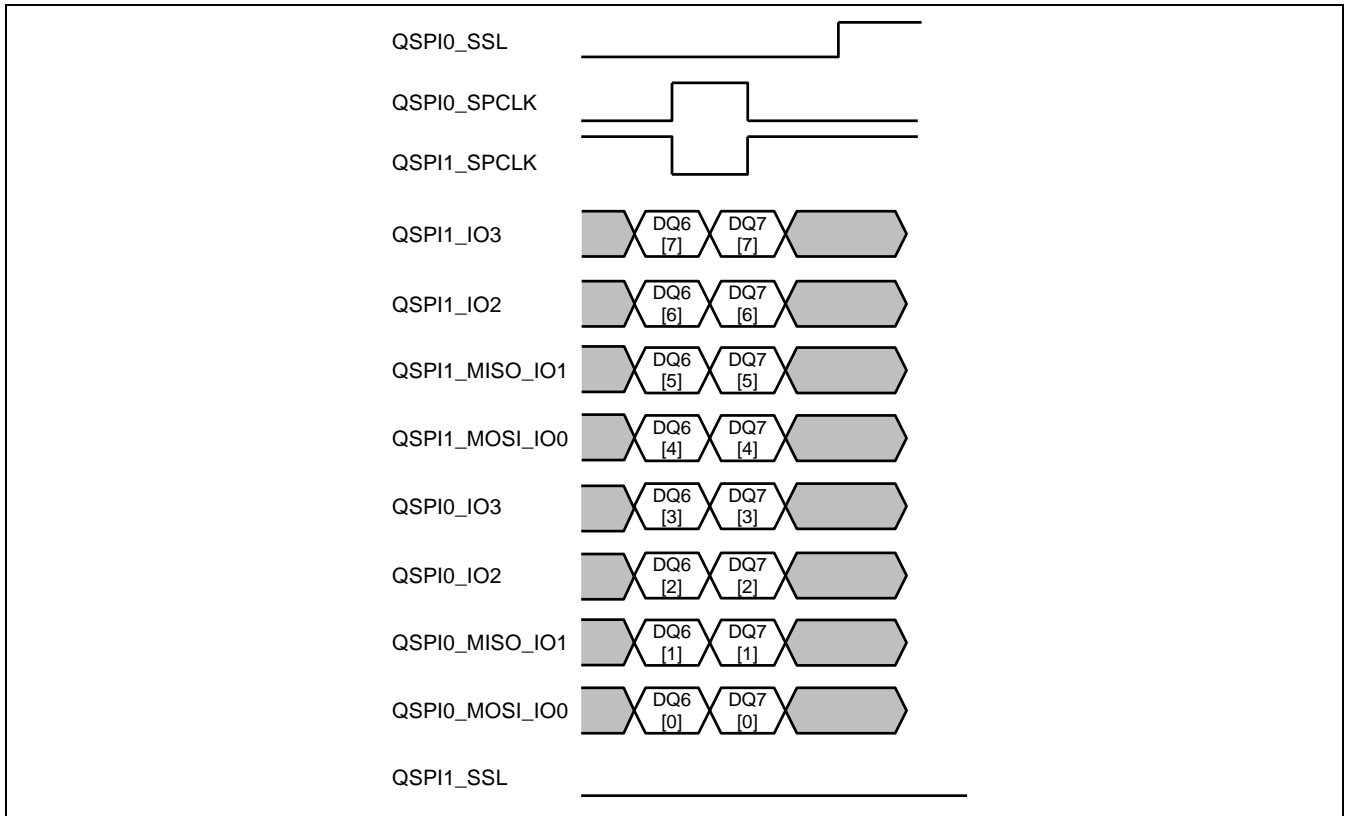


Figure 56.26 Example of Transfer Format in Write phase with HyperFlash (16 bit)

56.3.9 Data Pin Control

With this module, the status of pins can be automatically changed based on the data size to be used and the read/write settings. The pin status during the QSPIn_SSL negation can be set with the MOIIIO3, MOIIIO2, MOIIIO1, and MOIIIO0 bits in CMNCR.

The QSPI0_SSL and QSPIn_SPCLK pins are always output pins. The status of respective pins is specified in Table 56.8 to 56.11.

Note that connection with ch1 pins in 1 x serial flash configuration is prohibited.

Table 56.8 Pin Status (1)

| Pin | QSPIn_SSL Negation | QSPIn_SSL Assertion | |
|-------------------------|--------------------|---|------------|
| | | Command, Optional Command, Address, Option Data | |
| | | 1-bit Size | 4-bit Size |
| QSPI0_IO0, QSPI1_IO0 | MOIIIO0 bit value | Output | Output |
| QSPI0_IO1, QSPI1_IO1 | MOIIIO1 bit value | Hi-Z | Output |
| QSPI0_IO2, QSPI1_IO2 | MOIIIO2 bit value | specified in PFC * | Output |
| QSPI0_IO3, QSPI1_IO3 | MOIIIO3 bit value | specified in PFC * | Output |

Note: * QSPI0_IO2, QSPI1_IO2, QSPI0_IO3 and QSPI1_IO3 pin state should be controlled by PFC function. Refer to "8. and 9. Pin Function Controller"

Table 56.9 Pin Status (2)

| Pin | Transfer Data | | | |
|-------------------------|---------------------------------------|------------|------------------------------|------------|
| | External Address Space Read Operation | | SPI Operation | |
| | 1-bit Size | 4-bit Size | SPIRE Bit = 1, SPIWE Bit = 0 | |
| | | | 1-bit Size | 4-bit Size |
| QSPI0_IO0, QSPI1_IO0 | IO0FV bit value | Input | IO0FV bit value | Input |
| QSPI0_IO1, QSPI1_IO1 | Input | Input | Input | Input |
| QSPI0_IO2, QSPI1_IO2 | specified in PFC * | Input | specified in PFC * | Input |
| QSPI0_IO3, QSPI1_IO3 | specified in PFC * | Input | specified in PFC * | Input |

Note: * QSPI0_IO2, QSPI1_IO2, QSPI0_IO3 and QSPI1_IO3 pin state should be controlled by PFC function. Refer to the section 8. and 9. Pin Function Controller in User's Manual.

Table 56.10 Pin Status (3)

| Pin | Transfer Data | | | |
|-------------------------|------------------------------|------------|------------------------------|--------------------|
| | SPI Operation | | | |
| | SPIRE Bit = 0, SPIWE Bit = 1 | | SPIRE Bit = 1, SPIWE Bit = 1 | |
| | 1-bit Size | 4-bit Size | 1-bit Size | 4-bit Size |
| QSPI0_IO0, QSPI1_IO0 | Output | Output | Output | Setting prohibited |
| QSPI0_IO1, QSPI1_IO1 | Hi-Z | Output | Input | Setting prohibited |
| QSPI0_IO2, QSPI1_IO2 | specified in PFC * | Output | specified in PFC * | Setting prohibited |
| QSPI0_IO3, QSPI1_IO3 | specified in PFC * | Output | specified in PFC * | Setting prohibited |

Note: * QSPI0_IO2, QSPI1_IO2, QSPI0_IO3 and QSPI1_IO3 pin state should be controlled by PFC function. Refer to the section 8. and 9. Pin Function Controller in User's Manual.

Table 56.11 Pin Status (4)

| Pin | Dummy Cycle | |
|----------------------|--------------------|------------|
| | 1-bit Size | 4-bit Size |
| QSPI0_IO0, QSPI1_IO0 | IO0FV bit value | Hi-Z |
| QSPI0_IO1, QSPI1_IO1 | Hi-Z | Hi-Z |
| QSPI0_IO2, QSPI1_IO2 | specified in PFC * | Hi-Z |
| QSPI0_IO3, QSPI1_IO3 | specified in PFC * | Hi-Z |

Note: * QSPI0_IO2, QSPI1_IO2, QSPI0_IO3 and QSPI1_IO3 pin state should be controlled by PFC function. Refer to "8. and 9. Pin Function Controller"

56.3.10 QSPIn_SSL Pin Control

Negation conditions of the QSPIn_SSL pin are as follows.

(1) External Address Space Read Operation

(a) Normal read operation (RBE bit in DRCR = 0)

QSPIn_SSL negated after completing the data transfer and t2 cycle.

(b) Burst read operation without automatic QSPIn_SSL negation (RBE bit in DRCR = 1, SSLE bit in DRCR = 0)

QSPIn_SSL negated after completing the data transfer and t2 cycle.

(c) Burst read operation with automatic QSPIn_SSL negation (RBE bit in DRCR = 1, SSLE bit in DRCR = 1)

- QSPIn_SSL negated after t2 cycle when the read address is not continuous with the previously read address
- QSPIn_SSL negated after the SSLN bit in DRCR is set to 1.

Note: This function is available when connecting to serial flash.

(2) Manual Mode

(a) QSPIn_SSL pin assertion not retained (SSLKP bit in SMCR = 0)

QSPIn_SSL negated after completing the data transfer.

(b) QSPIn_SSL pin assertion retained (SSLKP bit in SMCR = 1)

QSPIn_SSL not negated.

When to be negated, data should be transferred after setting the SSLKP bit to 0.

Note: QSPIn_SSL is used as input pin for data strobe when connecting to HyperFlash or Octal-SPI flash.

56.3.11 QSPIn_SPCLK Pin Control

QSPIn0_SPCLK and QSPIn1_SPCLK are output depending on the following setting

Table 56.12 QSPIn_SPCLK output

| Pin | HyperFlash (CMNCR.BSZ = 01 PHYCNT.PHYMEM = 11) | Serial flash x 1ch (CMNCR.BSZ = 00 PHYCNT.PHYMEM = 00 or 01) | Serial flash x 2ch (CMNCR.BSZ = 01 PHYCNT.PHYMEM = 00 or 01) |
|--------------|---|--|--|
| QSPIn0_SPCLK | Output | Output | Output |
| QSPIn1_SPCLK | Output in differential relation with QSPIn0_SPCLK | Hi-Z | Output |

56.3.12 Flags

This module has two flag bits SSLF and TEND in CMNSR. These bits are read-only bits.

(1) SSLF Bit

This bit indicates the QSPIn_SSL pin status. The status is 1 when the QSPIn_SSL is asserted, and the status is 0 when the QSPIn_SSL is negated.

(2) TEND Bit

This bit indicates whether transfer of data is in progress or the transfer of data has ended.

During t1 time period, data transfer, t2 time period, t3 time period, and waiting for read access by burst read and QSPIn_SSL automatic negation, the TEND bit is read as 0 to indicate that the transfer of data is in progress.

When other than the above, the TEND bit is read as 1 to indicate that transfer of data has ended.

(3) Register Re-writing Timing

The status of the TEND bit determines the rewritable registers.

The registers which can be written to, except the SSLN bit in DRCR, should be modified when TEND = 1.

The data should be read when the TEND flag in SMRDR0 or SMRDR1 is 1;

CMNSR can always be read.

56.3.13 Write Buffer Operation

This module uses read cache as write buffer in write operation. This write buffer improve the performance.

Table 56.13 Write Buffer Area

| Address | Access Size |
|----------------------------|-----------------------|
| H'EE20_8000 to H'EE20_80FF | 4, 8, 16, 32, 64 Byte |

Note: This area should be accessed sequentially from the start address and transfer size to the device is 256-byte unit. All Write Buffer area should be filled with transfer data in every transfer. When accessing non-sequentially or at random, the operation is not guaranteed. After Write Buffer Operation, this cache area must be cleared by DRCR.RCF bit. Write Buffer operation supports 24bit or 32bit address only. SMENR.ADE should be specified 0111 (ADR[23:0]) or 1111 (ADR[31:0]).

Figure 56.27 shows the sequence to use write buffer. About the initial setting of Manual mode, refer to Figure 56.16.

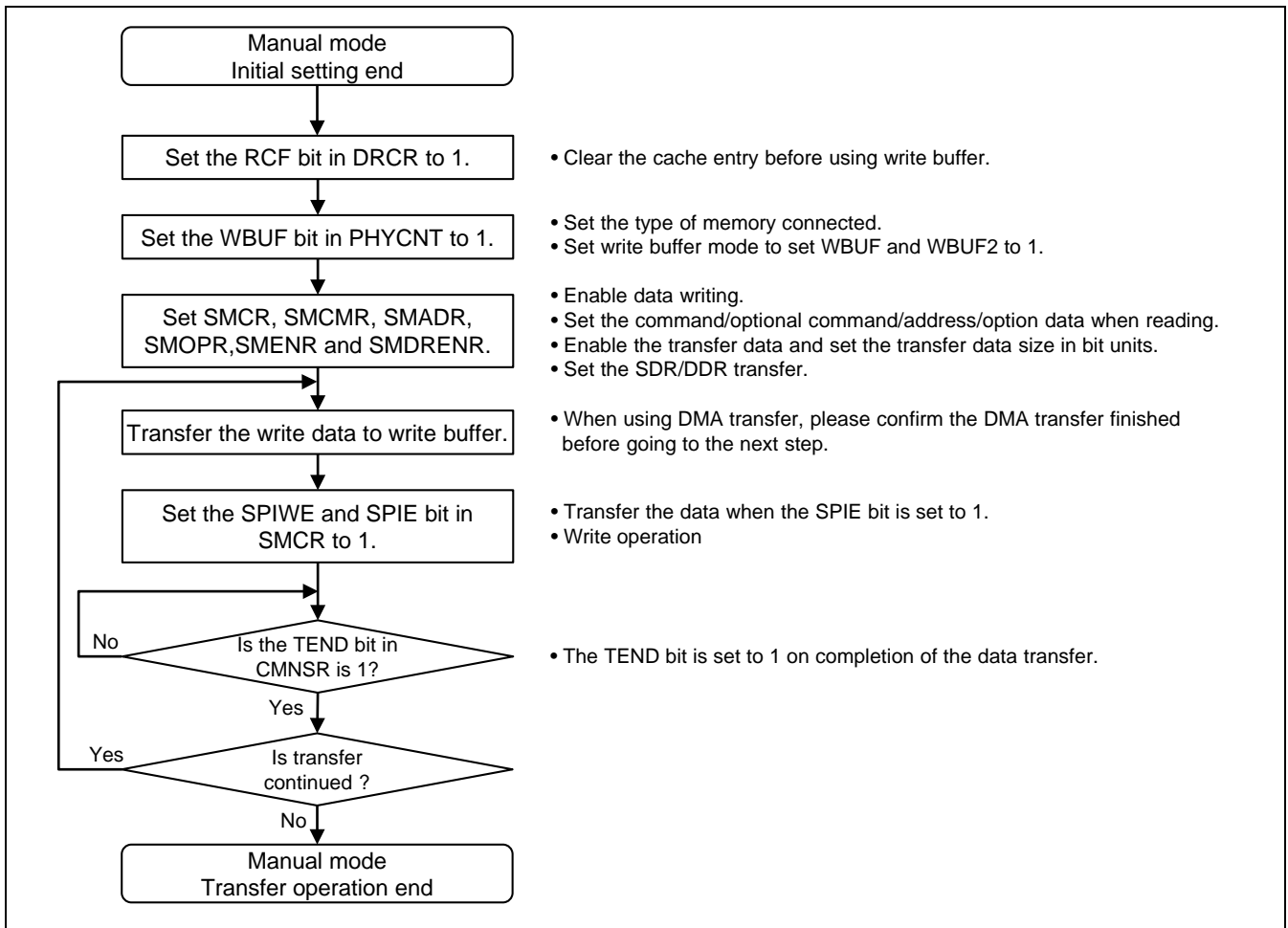


Figure 56.27 Write Buffer Usage Sequence

56.3.14 Data Alignment with Octal-SPI flash memory

When this LSI connects to Octal-SPI flash memory, this controller support two types of data alignment. Figure 56.28 and Figure 56.29 show sequential and alternate alignment respectively. Write data alignment in Write Buffer operation is same.

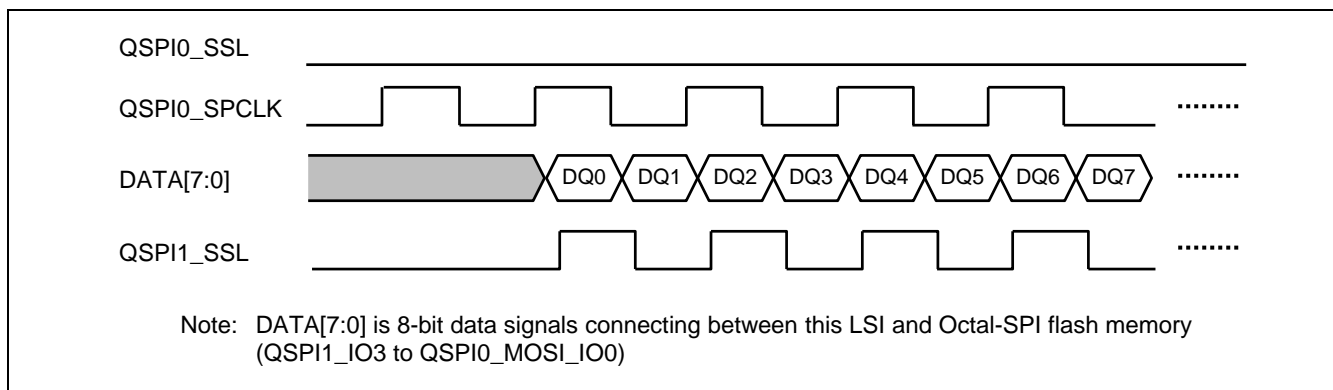


Figure 56.28 Sequential Data Alignment with Octal-SPI flash (PHYCNT.OCTA = 10)

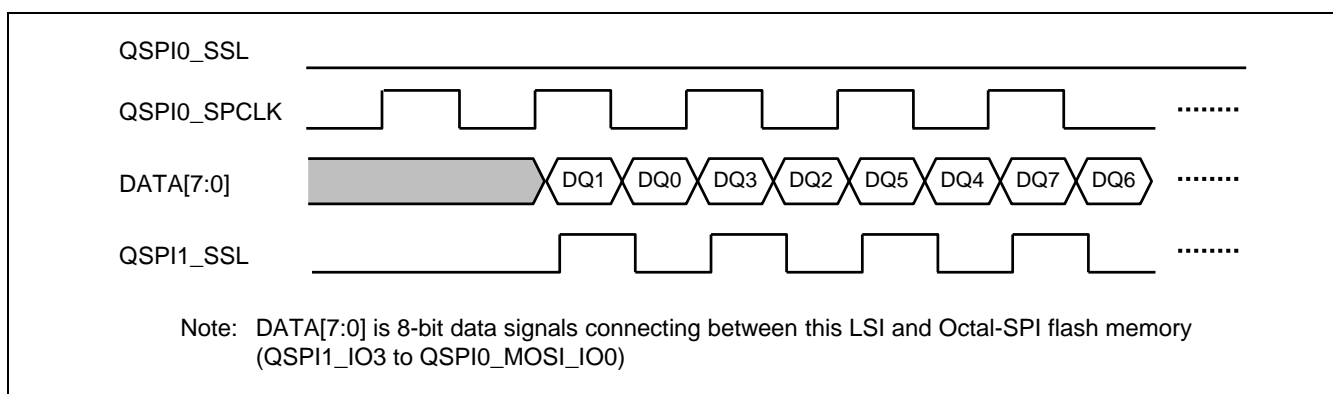


Figure 56.29 Alternate Data Alignment with Octal-SPI flash (PHYCNT.OCTA = 01)

Figure 56.30 shows the write data alignment in 8-8-8 protocol in manual mode operation.

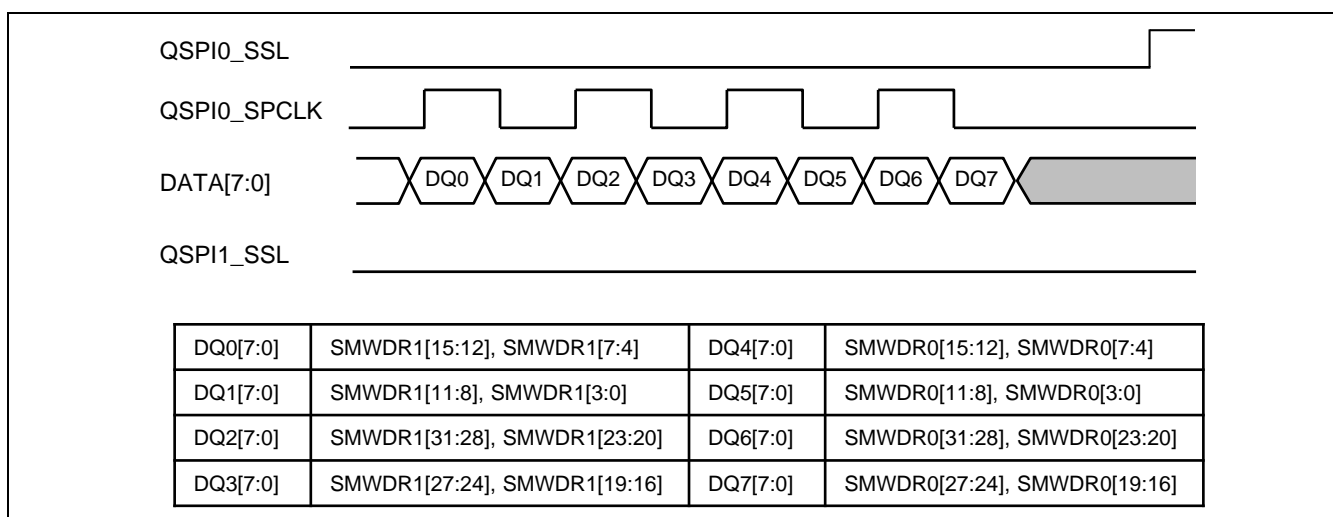


Figure 56.30 Write Data Alignment with Octal-SPI flash in manual mode

56.3.15 Supported Serial Flash Protocol

Table 56.14 shows the protocols of serial flash memory which are supported in this LSI.

Note: The RZ/G2M V1.3 (not RZ/G2M V3.0) does not support DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit). In the following description, do not set the related operation.

Table 56.14 Supported protocols of serial flash memory

| Bit width (Command-Address-Data) | Command | Address | Data |
|----------------------------------|---------|---------|------|
| 1-0-0 | SDR | NA | NA |
| 1-0-1 | SDR | NA | SDR |
| 1-1-1 | SDR | SDR | SDR |
| | SDR | DDR | DDR |
| 1-1-4 | SDR | SDR | SDR |
| 1-4-4 | SDR | SDR | SDR |
| | SDR | DDR | DDR |

56.3.16 Supported Octal-SPI Flash Protocol

Table 56.15 shows the protocols of Octal-SPI flash memory which are supported in this LSI.

Table 56.15 Supported protocols of Octal-SPI flash memory

| Bit width (Command-Address-Data) | Command | Address | Data |
|----------------------------------|---------|---------|------|
| 1-0-0 | SDR | NA | NA |
| 1-0-1 | SDR | NA | SDR |
| 1-1-1 | SDR | SDR | SDR |
| 8-0-0 | SDR | NA | NA |
| | DDR | NA | NA |
| 8-0-8 | DDR | NA | DDR |
| 8-8-8 | DDR | DDR | DDR |

56.3.17 DDR operation

This controller supports hardware calibration to adjust the timing of data strobe to support DDR protocol. When this LSI connects to the device with data strobe like HyperFlash or Octal SPI flash, this controller adjusts the strobe point automatically. When this LSI connects to the device without data strobe like serial flash memory in DDR mode, more adjustment is necessary to support DDR operation.

To adjust the timing of DDR operation without data strobe flash device, dummy read operation is necessary by software to change the delay of the internal strobe. PHYCNT.STRTIM bits change the delay. The delay depends on the specified value, 000(biggest) > 001 > ... > 110 > 111 (smallest), of PHYCNT.STRTIM bits. The maximum value of tVSD in Table 73.21.1 can be achieved among these setting value, so software can find the appropriate timing for changing these values one by one. Figure 56.31 shows the flow of this timing adjustment.

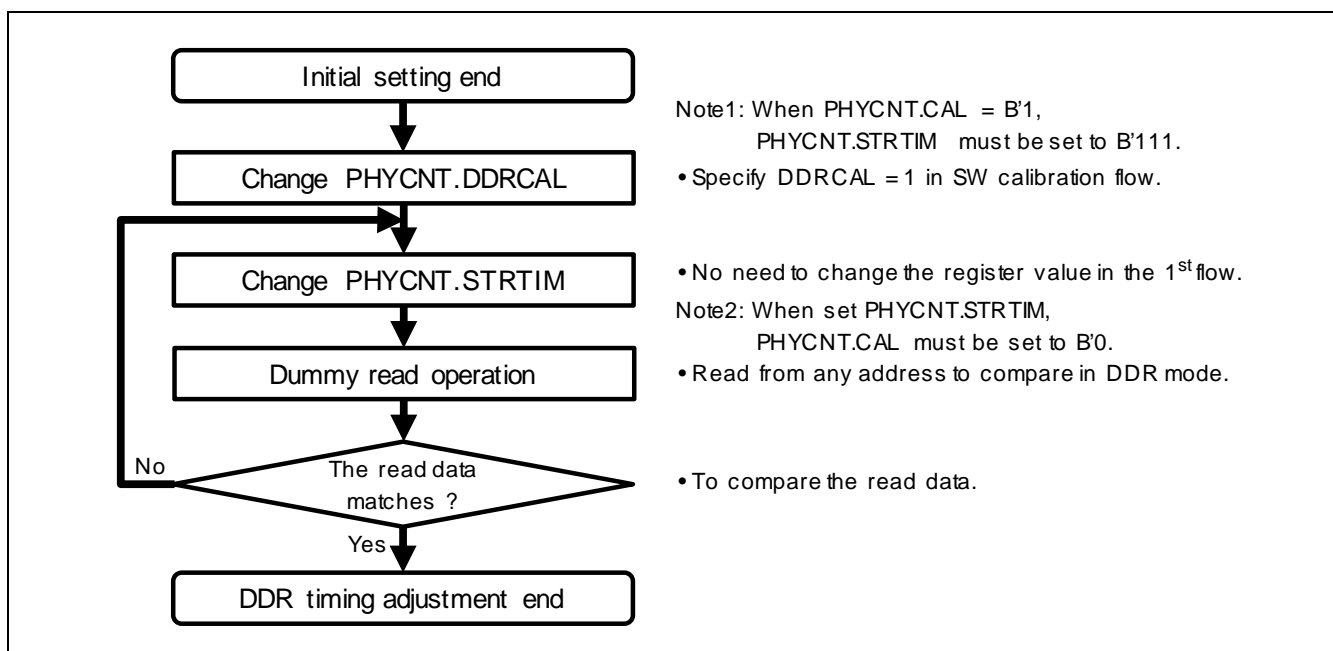


Figure 56.31 DDR timing adjustment to connect to the device without data strobe

This timing adjustment should be done in periodically because the timing changes with PVT condition in both this LSI and the connected device.

Note: The RZ/G2M V1.3 (not RZ/G2M V3.0) does not support DDR operation with QSPI (1/4bit) and QSPIx2 (8bit).

56.3.18 Security function

This controller supports security function to protect the data on the Flash device.

(1) SPI multi I/O bus space division

ADD_DIVn.ADD_DIVn (n = 1, 2, 3) bit specifies the address space division to separate the area into four. And SEC_CONF.AREAn (n = 0, 1, 2, 3) specifies the attribution of each area. The address specified ADD_DIVn.ADD_DIVn[19:0] bit are shifted and evaluate the bus address with 4K granularity. For example, when ADD_DIV1.ADD_DIV1[19:0] is specified to 20'h09FFF and SEC_CONF.AREA0 is specified to 2'b01 (non-secure) setting, Area0 becomes H'0800_0000 – H'09FF_FFFF and Non-Secure attribution, and Area1 starts from H'0A00_0000.

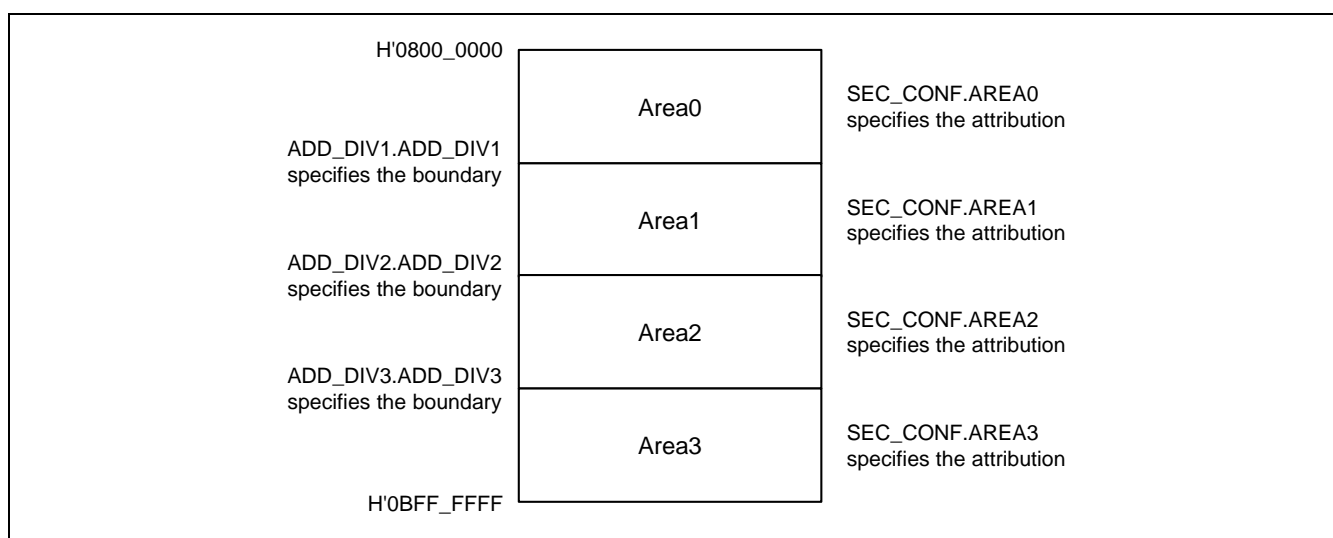


Figure 56.32 Area protection to divide the address space

Note that ADD_DIVn (n = 1 to 3) should be specified to keep the order of H'0800_0000 < ADD_DIV1.ADD_DIV1 < ADD_DIV2.ADD_DIV2 < ADD_DIV3.ADD_DIV3 < H'0BFF_FFFF. If this order is not kept, the operation of this module is not guaranteed. Specifying ADD_DIV1.ADD_DIV1 = 20h'0_8000 to make H'0800_0000 ~ H'0800_0FFF in Area0 is not available.

(2) Command restriction for serial flash

This controller can limit the command to issue to the connected serial flash. SEC_CMDn (n = 0 to 15) and NON_SEC_CMDn (n = 0 to 15) specify the limited command. For example, when H'FF is specified in SEC_CMD0.LIMCMD[7:0] bit and H'FF command is specified in SMCMDR.COMD bit, the operation by Secure CPU is not issued. LIMEN bit in each register enables this function.

(3) Write/Erase protect function for serial flash

This controller can protect the mistaken overwrite or erase on the other attribution to limit the Write/Erase command to the specific area on serial flash. ERASELISTn (n = 1, 2) and WRITELISTn (n = 1, 2) specify the prohibited command. When the target address belongs to secure attribution area specified in ADD_DIVn and SEC_CONF registers, the operation by non-secure attribution is not issued.

(4) Access Right

This controller has access right function for the register setting. There are two attribution, Secure and Non-Secure same as area protection. The master which specifies ARIGHT.RIGHT_EN bit has the right to access. The access right is reflected in each bit 3 to 1 according to the attribution of each master. To cancel the access right, the same attribution master should access to this register with RIGHT_EN = 0. The access is ignored when the attribution of access is different.

56.3.19 PHY register interface

This register interface is for the additional timing adjustment of DDR operation without data strobes from flash devices in 56.3.17.

(1) Register Configuration

Table 56.16 shows the register configuration of the PHY register interface.

Table 56.16 Register Configuration of PHY register interface

| Register Name | Abbreviation | R/W | Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | |
|-------------------------|--------------|-----|-------------|---------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| PHY Address register | PHYADD | R/W | H'EE20_0070 | H'0000_0000 | 32 | — | — | — | √ |
| PHY Write Data register | PHYWR | R/W | H'EE20_0074 | H'0000_0000 | 32 | — | — | — | √ |
| PHY Read Data register | PHYRD | R/W | H'EE20_0078 | H'0000_0000 | 32 | — | — | — | √ |

Note: Do not write to any other addresses than listed above. Operations are not guaranteed if such attempt is made. Values read from addresses other than those listed above are undefined.

Table 56.17 shows the register configuration of PHY. The registers in Table 56.17 can only be accessed by the registers in Table 56.16.

Table 56.17 Register Configuration of PHY register interface

| Register Name | Abbreviation | R/W | Address | Initial value | Access Size | Second Generation RZ/G Series Products | | | |
|------------------------------------|--------------|-----|---------|---------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Mode register | MD | R/W | H'00 | H'0000_0000 | 32 | — | — | — | √ |
| Rise Edge Delay Line Tap Selection | RDLSEL | R/W | H'22 | H'0000_0000 | 32 | — | — | — | √ |
| Fall Edge Delay Line Tap Selection | FDLSEL | R/W | H'24 | H'0000_0000 | 32 | — | — | — | √ |
| Rise Edge Delay Line Tap Monitor | RDLMON | R | H'26 | undefined | 32 | — | — | — | √ |
| Fall Edge Delay Line Tap Monitor | FDLMON | R | H'28 | undefined | 32 | — | — | — | √ |

Note: Do not write to any other addresses than listed above. Operations are not guaranteed if such attempt is made. Values read from addresses other than those listed above are undefined.

(2) Register Description

The registers of the RPC-IF are assigned to and located in the address space of the internal bus.

[Legend for Register Descriptions]

Initial value: Register value after a reset.

—: Undefined value

R/W: Readable/writable. The written value can be read.

R: Read-only. The write value should always be initial value. For details of initial value, refer to Table 56.3.

W: Write-only.

(a) PHY Address Register (PHYADD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

PHYADD is a 32-bit register that controls access to the PHY registers.

This register is used for the flow in Figure 56.34.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|-------|----|----|----|----|----|----|----|----|----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | ACCEN | RW | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | W | W | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | ADD | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 | ACCEN | B'0 | W | When writing or reading from a PHY register, set this bit to 1. The RW bit should be set simultaneously to access the PHY registers. |
| 30 | RW | B'0 | W | Specifies write or read from a PHY register. 1: Read 2: Write The ACCEN bit should be set simultaneously to access the PHY registers. |
| 29 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 to 0 | ADD[5:0] | H'00 | R/W | Specifies the addresses of the PHY registers. |

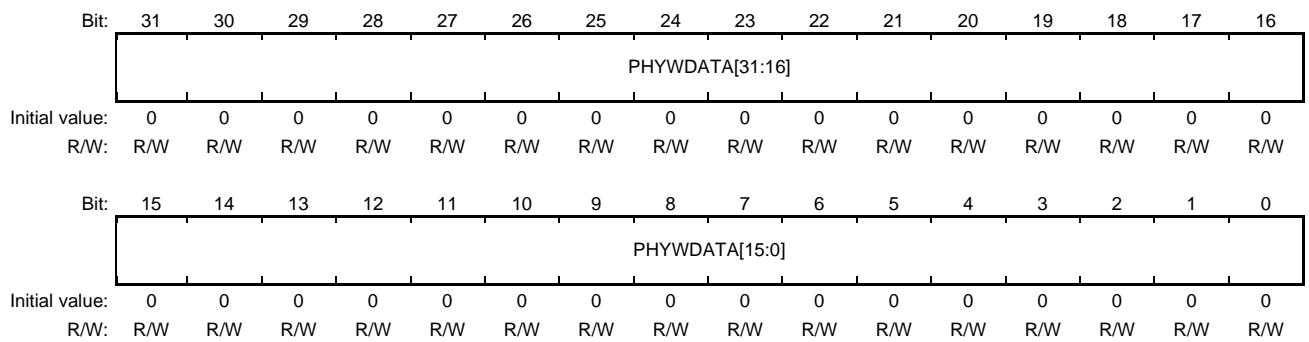
(b) PHY Write Data Register (PHYWR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

PHYWR is a 32-bit register that writes data to the PHY registers.

This register is used for the flow in Figure 56.34.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|-----------------------------------|
| 31 to 0 | PHYWDATA[31:0] | H'0000_0000 | R/W | Writes data to the PHY registers. |

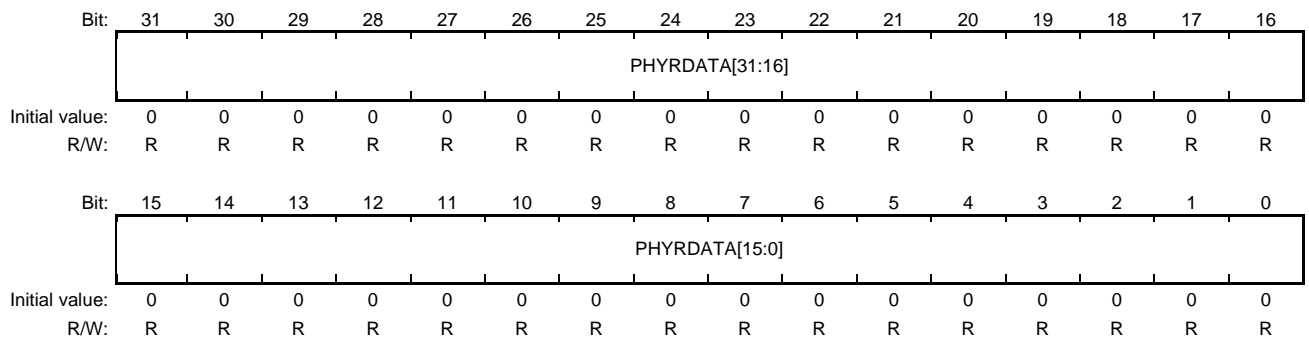
(c) PHY Write Data Register (PHYRD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

PHYRD is a 32-bit register bit register where the value of the PHY registers is read when reading from the PHY registers.

This register is used for the flow in Figure 56.34.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|--|
| 31 to 0 | PHYRDATA[31:0] | H'0000_0000 | R | When reading from a PHY register, the register value is read from this register. |

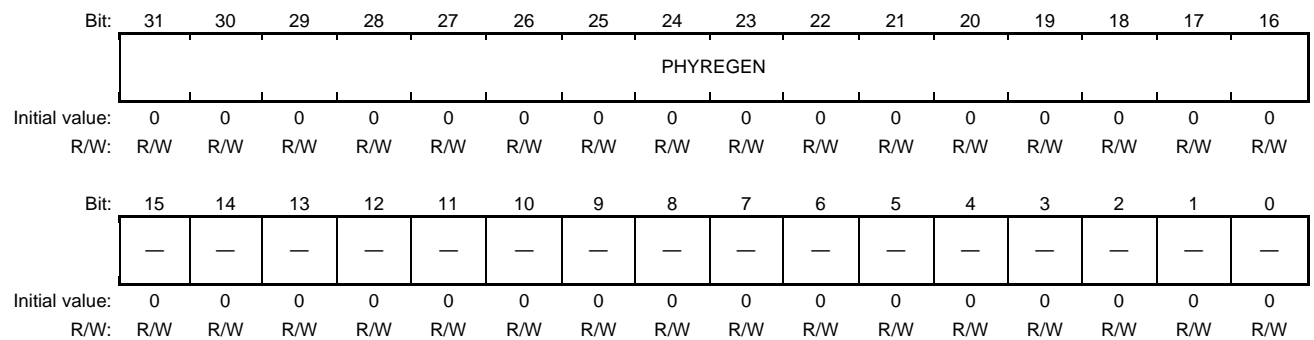
(d) Mode Register (MD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

MD is a 32-bit register that specifies the mode of the PHY registers.

This register is used for the flow in Figure 56.34.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 16 | PHYREGEN[15:0] | H'0000_0000 | R/W | Writes data to the PHY registers. |
| 15 to 0 | — | All 0 | R | Reserved The write value should always be set to the initial value. |

(e) Rise Edge Delay Line Tap Selection (RDLSEL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

RDLSEL is a 32-bit register that specifies Rise Edge Delay Line Tap Selection.

This register is used for the flow in Figure 56.34.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----------------|----|----|---------------|-----|-----|-----|-----|-----|----|----|---------------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | QSPI1 DLSET | — | — | QSPI1DLTAPSEL | | | | — | — | — | — | QSPI0DLTAPSEL | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved The write value should always be set to the initial value. |
| 15 | QSPI1DLSETEN | B'0 | R/W | When this bit is 1, QSPI1DLTAP is used to set Delay Line Tap Selection. |
| 14 to 13 | — | All 0 | R | Reserved The write value should always be set to the initial value. |
| 12 to 8 | QSPI1DLTAPSEL [4:0] | H'00 | R/W | Select the delay line tap of the QSPI1 side. B'0_0000: The delay is the smallest. . . B'1_1111: The delay is the largest. |
| 7 | QSPI0DLSETEN | B'0 | R/W | When this bit is 1, QSPI0DLTAP is used to set Delay Line Tap Selection. |
| 6 to 5 | — | All 0 | R | Reserved The write value should always be set to the initial value. |
| 4 to 0 | QSPI0DLTAPSEL [4:0] | H'00 | R/W | Select the delay line tap of the QSPI0 side. B'0_0000: The delay is the smallest. . . B'1_1111: The delay is the largest. |

(f) Fall Edge Delay Line Tap Selection (FDLSEL)

| | | | |
|--------|--------------------------|---------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | \RZ/G2N | RZ/G2E |
| — | — | — | √ |

FDLSEL is a 32-bit register that specifies Fall Edge Delay Line Tap Selection.

This register is used for the flow in Figure 56.34.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----------------|----|----|---------------|-----|-----|-----|----------------|-----|----|---------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | QSPI1 DLSET | — | — | QSPI1DLTAPSEL | | | | QSPI0 DLSET | — | — | QSPI0DLTAPSEL | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved The write value should always be set to the initial value. |
| 15 | QSPI1DLSETEN | B'0 | R/W | When this bit is 1, QSPI1DLTAP is used to set Delay Line Tap Selection. |
| 14 to 13 | — | All 0 | R | Reserved The write value should always be set to the initial value. |
| 12 to 8 | QSPI1DLTAPSEL [4:0] | H'00 | R/W | Select the delay line tap of the QSPI1 side. B'0_0000: The delay is the smallest. . . B'1_1111: The delay is the largest. |
| 7 | QSPI0DLSETEN | B'0 | R/W | When this bit is 1, QSPI0DLTAP is used to set Delay Line Tap Selection. |
| 6 to 5 | — | All 0 | R | Reserved The write value should always be set to the initial value. |
| 4 to 0 | QSPI0DLTAPSEL [4:0] | H'00 | R/W | Select the delay line tap of the QSPI0 side. B'0_0000: The delay is the smallest. . . B'1_1111: The delay is the largest. |

(g) Rise Edge Delay Line Tap Monitor (RDLMON)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

RDLMON is a 32-bit register that specifies Rise Edge Delay Line Tap Selection.

This register is used for the flow in Figure 56.34.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|---------------|----|----|----|----|----|----|---------------|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | QSPI1DLTAPMON | | | | — | — | — | QSPI0DLTAPMON | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------|---------------|-----|---|
| 31 to 13 | — | — | R | Reserved |
| 12 to 8 | QSPI1DLTAPMON[4:0] | — | R | Monitor the selected delay line tap of the QSPI1 side. When QSPI1DLSETEN of the Delay Line Tap Selection register is 1, the read value is QSPI1DLTAPSEL of the Delay Line Tap Selection register. When QSPI1DLSETEN of the Delay Line Tap Selection register is 0, the read value is the calibration result of PHYCNT.CAL. B'0_0000: The delay is the smallest. . . B'1_1111: The delay is the largest. |
| 7 to 5 | — | — | R | Reserved |
| 4 to 0 | QSPI0DLTAPMON[4:0] | — | R | Monitor the selected delay line tap of the QSPI0 side. When QSPI1DLSETEN of the Delay Line Tap Selection register is 1, the read value is QSPI1DLTAPSEL of the Delay Line Tap Selection register. When QSPI1DLSETEN of the Delay Line Tap Selection register is 0, the read value is the calibration result of PHYCNT.CAL. B'0_0000: The delay is the smallest. . . B'1_1111: The delay is the largest. |

(h) Fall Edge Delay Line Tap Monitor (FDLMON)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

FDLMON is a 32-bit register that specifies Fall Edge Delay Line Tap Selection.

This register is used for the flow in Figure 56.34.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|---------------|----|----|----|----|----|----|---------------|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | QSPI1DLTAPMON | | | | — | — | — | QSPI0DLTAPMON | | | | | |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------|---------------|-----|---|
| 31 to 13 | — | — | R | Reserved |
| 12 to 8 | QSPI1DLTAPMON[4:0] | — | R | Monitor the selected delay line tap of the QSPI1 side. When QSPI1DLSETEN of the Delay Line Tap Selection register is 1, the read value is QSPI1DLTAPSEL of the Delay Line Tap Selection register. When QSPI1DLSETEN of the Delay Line Tap Selection register is 0, the read value is the calibration result of PHYCNT.CAL. B'0_0000: The delay is the smallest. . . B'1_1111: The delay is the largest. |
| 7 to 5 | — | — | R | Reserved |
| 4 to 0 | QSPI0DLTAPMON[4:0] | — | R | Monitor the selected delay line tap of the QSPI0 side. When QSPI1DLSETEN of the Delay Line Tap Selection register is 1, the read value is QSPI1DLTAPSEL of the Delay Line Tap Selection register. When QSPI1DLSETEN of the Delay Line Tap Selection register is 0, the read value is the calibration result of PHYCNT.CAL. B'0_0000: The delay is the smallest. . . B'1_1111: The delay is the largest. |

(3) The ways of access to the PHY registers

For both the write and read sequences, it is mandatory to set the MD.PHYREGEN register to H'A539.

(a) The write sequence

Execute the write sequence according to the following procedures.

1. Specify the write data in PHYWR.
2. Specify the PHY register address in PHYADD.ADD.PHYADDRW=0 (write) and PHYADD.ACCEN=1 to start the write sequence.

(b) The read sequence

Execute the read sequence according to the following procedures.

1. Specify the PHY register address in PHYADD.ADD.PHYADDRW=1 (read) and PHYADD.ACCEN=1 to start the readsequence.
2. Dummy read PHYRD to make time to finish reading from the PHY registers.
3. Read the PHYRD register to get the read value.

If the result of a dummy read does not match the settings of PHYCN.STRTIM bits, the Delay Line in Figure 56.33 is adjusted in an additional process by the software according to the flowchart in Figure 56.34. Figure 56.35 shows the adjustment method of the delay circuit. Refer to section 56.3 for registers set in Figure 56.35.

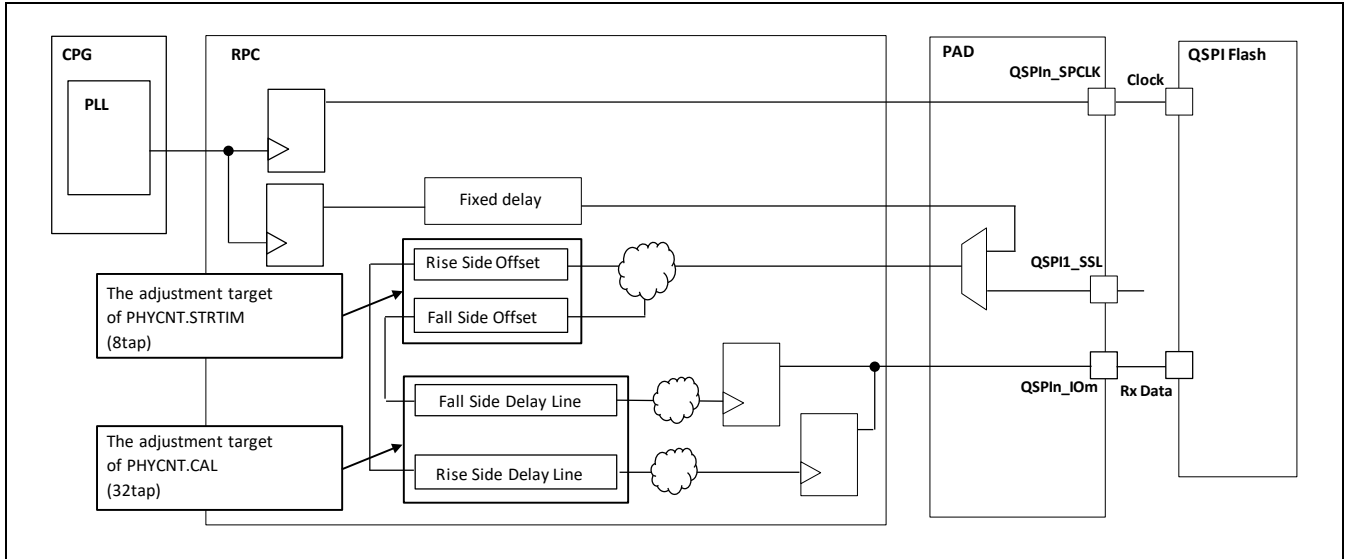


Figure 56.33 Circuit of data receiving

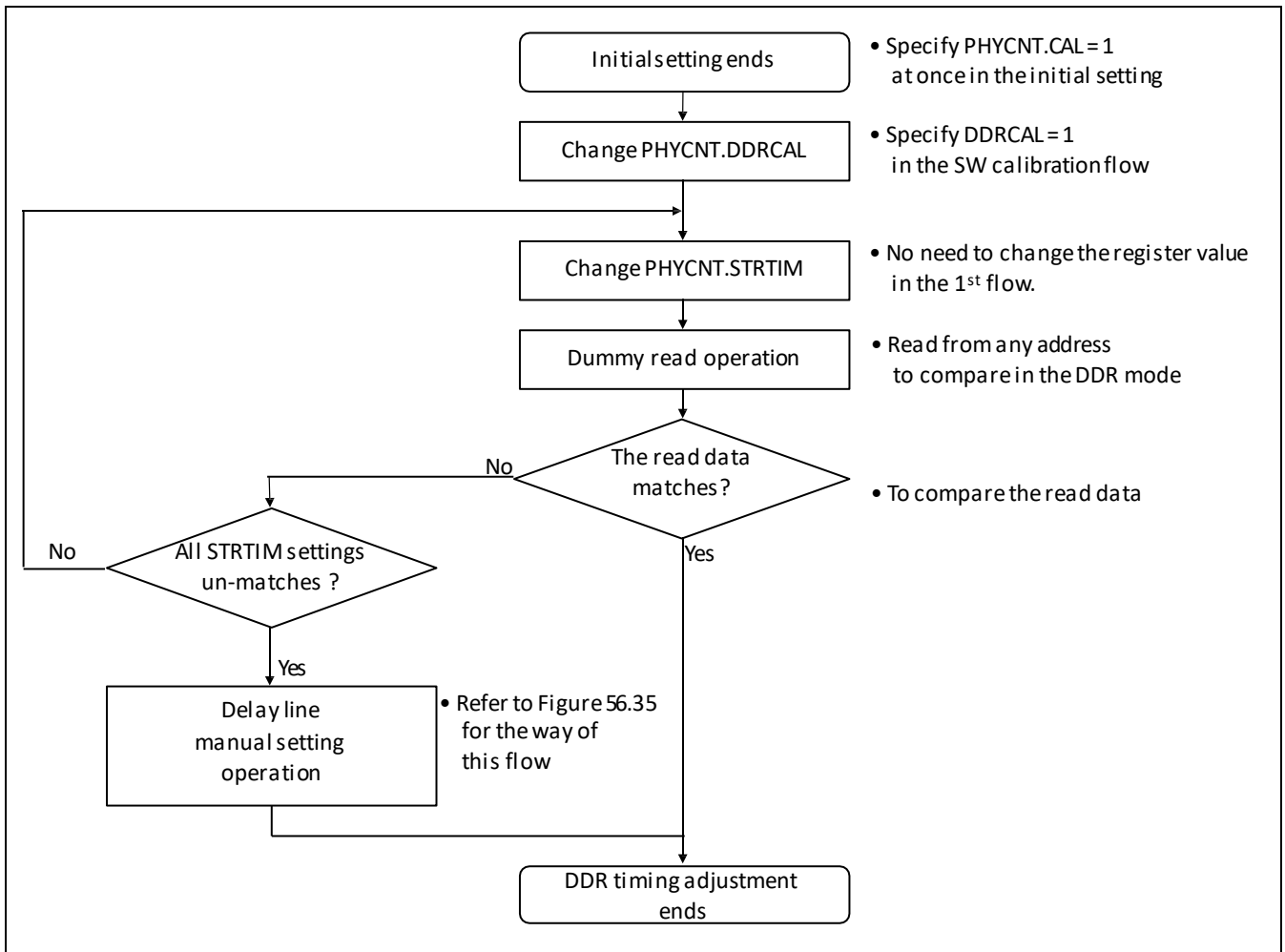


Figure 56.34 Additional timing adjustment flow

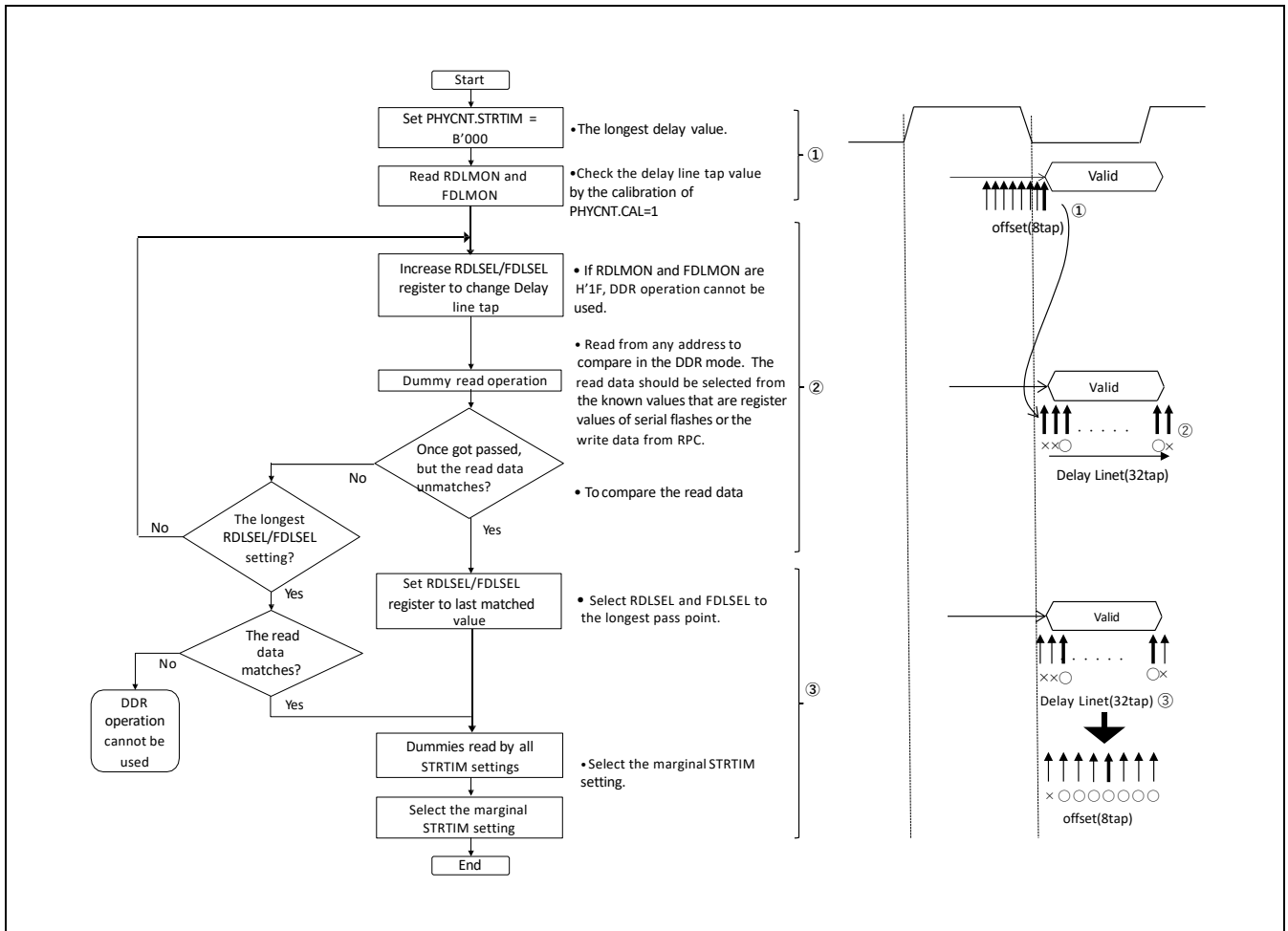


Figure 56.35 Delay line manual setting operation

56.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

56.4.1 Transfer to Read Data while the Signal on the QSPIn_SSL Pin is Negated

Set the SMENR.SPIDE[3:0] bits to B'1100 or B'1111 when transfer only for reading data is to proceed.

Transfer will not proceed normally if the setting of the SMENR.SPIDE[3:0] bits is B'1000.

56.4.2 Transfer to Read Data while the Signal on the QSPIn_SSL Pin is Asserted

When transfer only for reading data is to proceed, set the SMENR.SPIDE[3:0] bits to B'1100 or B'1111, or end the immediately preceding transfer with reading data.

When the immediately preceding transfer is of a command, optional command, address, or option data, or is transfer for writing data, the subsequent transfer only for reading data will not proceed normally if the setting of the SMENR.SPIDE[3:0] bits is B'1000.

56.4.3 Notes on Starting Transfer from the QSPIn_SSL Retained State in Manual Mode

Be sure to set the SPIWE bit in the SMCR register to 1 when the transfer of a command, optional command, address, or option data is started while the QSPIn_SSL pin is being asserted in manual mode.

56.4.4 RPC_RESET# operation

There are two ways to operate RPC_RESET# pins.

1. Use PHYINT.RSTEN, PHYINT.RSTVAL to operate.
The RPC_RESET# state depends on PHYINT.RSTVAL value.
2. Activate reset to this module
During reset period by Power-on-reset or Software Reset, RPC_RESET# pin state is L. After reset is deactivated, RPC_RESET# pin state is H.

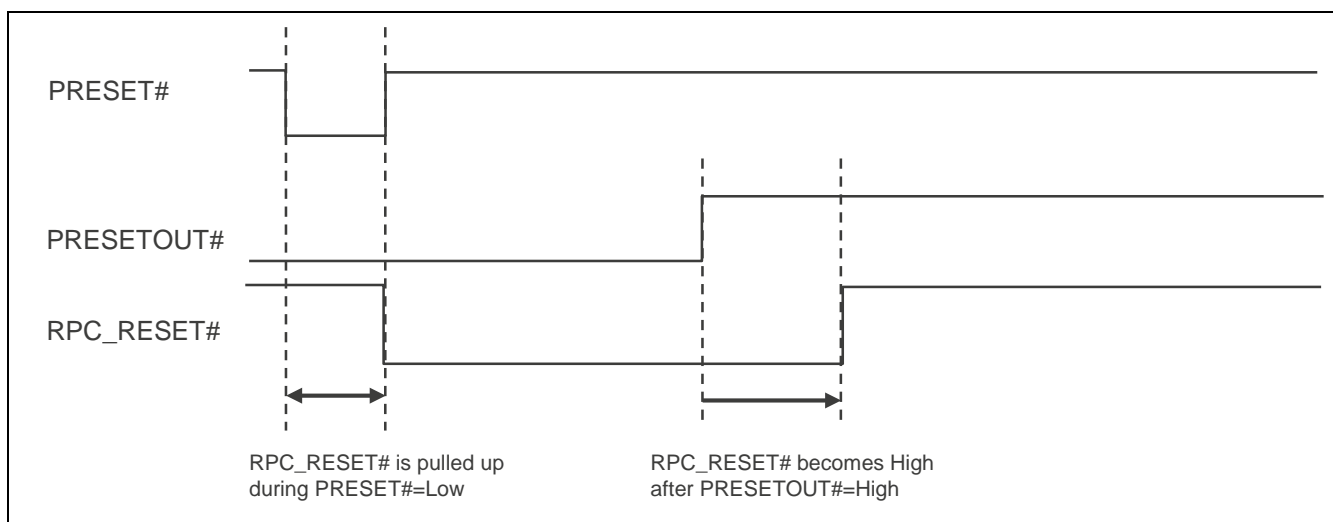


Figure 56.36 Relationship of RPC_RESET# and PRESET# timing

56.4.5 Software Reset after Module Stop operation

Be sure to supply software reset to this module after module stop is deactivated. Note that RPC_RESET# pin state is L while software reset is activated. Guarantee the reset period and accessible period after reset which are required in the connected devices.

56.4.6 DMA transfer

In read operation by DMA transfer, external address space read mode is available. The read address (source address in DMAC) is the memory area (H'0800_0000 – H'0BFF_FFFF) by Auto-Request mode. In High Speed response mode (PHYCNT.HS = 1), DRCCR.RBURST should be set to B'1_1111 and access address alignment from RPC to flash memory should be 256Byte align. Accessing to register area during DMA transfer is prohibited during High Speed response mode.

In write operation, refer to 56.3.13.

56.4.7 Frequency change

(1) RPCCKCR

When the operation frequency changes by setting RPCCKCR in CPG, read after write operation on this register is necessary to guarantee the setting to be applied.

56.4.8 QSPI1 pins switched to high impedance [RZ/G2M V1.3] (not RZ/G2M V3.0)

Applying QSPI0 single boot (Mode pins MD[4:1] = 0100) the QSPI1 terminals are switched to high impedance. In case using a second QSPI flash device on the QSPI1 interface for 8-bit access later on, apply external pull-up resistors to the signal lines.

57. SD Card/MMC Interfaces

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

57.1 Overview

57.1.1 Features

This LSI has four SD card interfaces (SDHI0 to SDHI3), two of which can also be used as MMC interfaces (SDHI2 and SDHI3). The select setting of SDHI and MMC is unnecessary.

- SD memory/IO card interface (1-bit/4-bit SD bus)
- SD, SDHC, and SDXC SD memory card access supported
- Default, high-speed, UHS-I/SDR50, and SDR104 transfer modes supported
- SD clock (SDCLK) frequency = internal SDx ϕ frequency/2ⁿ (n = 0 to 9)
SDCLK = 200 MHz (SDR104)
SDCLK = 100 MHz (SDR50)
- Error check function: CRC7 (for command/response), CRC16 (for data)
- Interrupt request: 2
- Card detect function
- Write protect support
- MMC interface (1-bit/4-bit/8-bit MMC bus)
- e-MMC device access supported
- Backward-compatible, high-speed, HS200, and HS400 transfer modes supported
SDCLK = 200 MHz (HS200)
- High-priority interrupt (HPI) supported
- Alternative Boot operation supported
- Sequencer function (SEQ)
The SEQ can issue maximum 8 commands sequentially.

Development of the SD-related products needs the conclusion of the following agreement.

SD Host/Ancillary Product License Agreement (SD HALA)

x = 0 to 3

57.1.2 Block Diagram

Figure 57.1 shows a block diagram of the SD host interface.

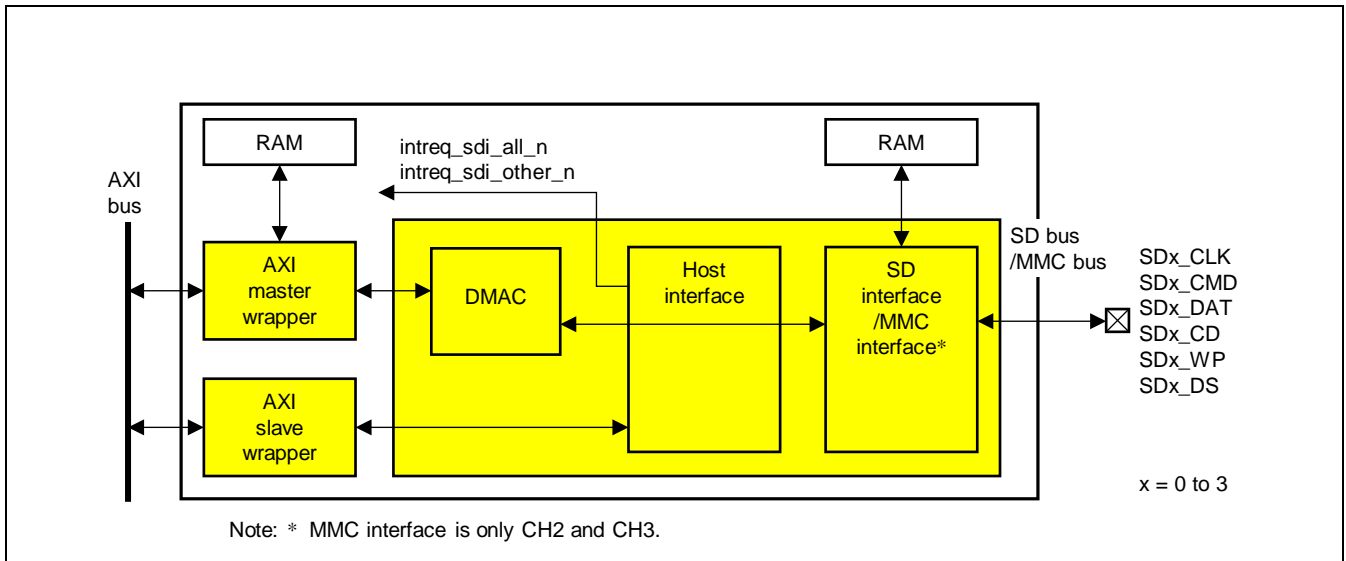


Figure 57.1 Block Diagram of SD Host Interface

57.1.3 External Pins

Table 57.1 lists the input and output pins used by the interface. The voltage of the SDHI terminal is variable. Therefore a power network is separate. When choosing the function of GPIO and the other ones, be careful.

Table 57.1 Pin Configuration

| Name | Abbreviation | I/O | Function | Second Generation RZ/G Series Products | | | |
|-------------|--------------|-----|-----------------------------|---|----------------------------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SD0_CLK | — | O | SD clock | √ | √ | √ | √ |
| SD0_CMD | — | I/O | Command/response | √ | √ | √ | √ |
| SD0_DAT0 | — | I/O | Data [bit 0] | √ | √ | √ | √ |
| SD0_DAT1 | — | I/O | Data [bit 1]/SDIO interrupt | √ | √ | √ | √ |
| SD0_DAT2 | — | I/O | Data [bit 2]/Read wait | √ | √ | √ | √ |
| SD0_DAT3 | — | I/O | Data [bit 3]/Card detect | √ | √ | √ | √ |
| SD0_CD | — | I | Card detect *1 | √ | √ | √ | √ |
| SD0_WP | — | I | Write protect *1 | √ | √ | √ | √ |
| SD1_CLK | — | O | SD clock | √ | √ | √ | √ |
| SD1_CMD | — | I/O | Command/response | √ | √ | √ | √ |
| SD1_DAT0 | — | I/O | Data [bit 0] | √ | √ | √ | √ |
| SD1_DAT1 | — | I/O | Data [bit 1]/SDIO interrupt | √ | √ | √ | √ |
| SD1_DAT2 | — | I/O | Data [bit 2]/Read wait | √ | √ | √ | √ |
| SD1_DAT3 | — | I/O | Data [bit 3]/Card detect | √ | √ | √ | √ |
| SD1_CD | — | I | Card detect *1 | √ | √ | √ | √ |
| SD1_WP | — | I | Write protect *1 | √ | √ | √ | √ |
| SD2_CLK *2 | — | O | SD clock | √ | √ | √ | — |
| SD2_CMD *2 | — | I/O | Command/response | √ | √ | √ | — |
| SD2_DAT0 *2 | — | I/O | Data [bit 0] | √ | √ | √ | — |
| SD2_DAT1 *2 | — | I/O | Data [bit 1]/SDIO interrupt | √ | √ | √ | — |
| SD2_DAT2 *2 | — | I/O | Data [bit 2]/Read wait | √ | √ | √ | — |
| SD2_DAT3 *2 | — | I/O | Data [bit 3]/Card detect | √ | √ | √ | — |
| SD2_DAT4 *3 | — | I/O | Data [bit 4] | √ | √ | √ | — |
| SD2_DAT5 *3 | — | I/O | Data [bit 5] | √ | √ | √ | — |
| SD2_DAT6 *3 | — | I/O | Data [bit 6] | √ | √ | √ | — |
| SD2_DAT7 *3 | — | I/O | Data [bit 7] | √ | √ | √ | — |
| SD2_DS *3 | — | I | Data Strobe | √ | √ | √ | — |
| SD2_CD | — | I | Card detect *1 | √ | √ | √ | — |
| SD2_WP | — | I | Write protect *1 | √ | √ | √ | — |
| SD3_CLK *2 | — | O | SD clock | √ | √ | √ | √ |
| SD3_CMD *2 | — | I/O | Command/response | √ | √ | √ | √ |
| SD3_DAT0 *2 | — | I/O | Data [bit 0] | √ | √ | √ | √ |
| SD3_DAT1 *2 | — | I/O | Data [bit 1]/SDIO interrupt | √ | √ | √ | √ |
| SD3_DAT2 *2 | — | I/O | Data [bit 2]/Read wait | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Name | Abbreviation | I/O | Function | Second Generation RZ/G Series Products | | | |
|-------------|--------------|-----|--------------------------|--|----------------------------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| SD3_DAT3 *2 | — | I/O | Data [bit 3]/Card detect | √ | √ | √ | √ |
| SD3_DAT4 *3 | — | I/O | Data [bit 4] | √ | √ | √ | √ |
| SD3_DAT5 *3 | — | I/O | Data [bit 5] | √ | √ | √ | √ |
| SD3_DAT6 *3 | — | I/O | Data [bit 6] | √ | √ | √ | √ |
| SD3_DAT7 *3 | — | I/O | Data [bit 7] | √ | √ | √ | √ |
| SD3_DS *3 | — | I | Data Strobe | √ | √ | √ | √ |
| SD3_CD | — | I | Card detect *1 | √ | √ | √ | √ |
| SD3_WP | — | I | Write protect *1 | √ | √ | √ | √ |

Notes: 1. When a pin is not in use, fix the corresponding signal to High level by external input.

2 x = 2 to 3: This is used by both of SDHI and MMC.

3 x = 2 to 3: This is used only MMC.

57.1.4 Register Configuration

The base addresses for each interface are as follows. The select setting of SDHI and MMC is unnecessary. The difference between SDHI and MMC is only the command type by Command type register (SD_CMD).

SDHI0: H'00_EE10_0000

SDHI1: H'00_EE12_0000

SDHI2: H'00_EE14_0000

SDHI3: H'00_EE16_0000

MMC0: H'00_EE14_0000

MMC1: H'00_EE16_0000

Note: The two pairs of interfaces (SDHI2 and MMC0, SDHI3 and MMC1) can only be used for either the SDHI or MMC interface.

57.2 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

(1) Designing the board

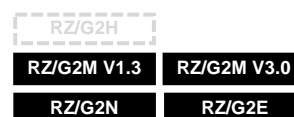
SDHI and MMC interface transfer performance has been improved. When designing the board, run the SI simulation using the IBIS model.

(2) Initial value of drivability at eMMC boot (RZ/G2N)

At eMMC boot, initial value of eMMC pin drivability of RZ/G2N is set to B'100 in the eMMC boot program. When operating eMMC with HS200, HS400, reset the optimum setting value that matches the characteristics of the board on which RZ/G Gen 2 is used with the eMMC driver.

Since RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 uses the initial value of PFC at eMMC boot time, drivability can't be adjusted. Adjusting the damping resistance for the eMMC terminal on the board at eMMC boot is needed time to suppress overshoot / undershoot.

58. rawNAND Controller (R-NANDC)



58.1 Overview

The NAND Flash Memory Interface controller implements the function of a high level interface to one NAND flash device. It supports the functionality of the high speed NAND Flash devices described in the ONFI 1.x specifications.

58.1.1 Features

- Compatibility
 - Compatible with the ONFi 1.x specification. (8/16-bit) except the "EDO data output timing mode" which is not supported.
 - * No full support of timing mode 4 and 5 due to missing EDO mode.
 - * Timing mode 4 or 5 memory devices can be used in non-EDO mode.
 - * NFMA fast transition mode can be used for timing mode 4 or 5 memory devices. (Refer to section 58.2.2 Main configurations register (CONTROL)).
 - Support for the Small Block devices - only devices that allow to disable CE signal when device is in the busy state are supported.
 - Support for the low capacity device (which use four address bytes) with the help of generic sequence.
- DMA Controller
 - Scatter-gather mode is supported.
 - Single transfer mode is supported.
 - Supports precise and imprecise burst modes.
 - Supports configurable burst types: single transfer, incrementing and stream.
 - 32-bit system memory addressing.
 - Triggering mechanism that depends on the programmable FIFO data level.
- Interrupt Controller
 - Each interrupt can be masked.
 - Each interrupt has its own status flag.
 - The status flags are also valid when the given interrupt is masked, and can be checked by the software polling mechanism.
 - Single interrupt signal output
- Data buffering
 - FIFO module based on dual port memory.
- ECC unit
 - The ECC module is based on the BCH algorithms.
 - The syndrome calculation, error detection and error correction phases in the BCH-based modules are pipelined.
 - The ECC unit supports the following options: 256, 512 bytes and 1024 bytes data blocks. The data block size depends on the selected ECC option.
- Advanced features
 - The page cache read/write sequences are supported.
 - The multiple planes read/write sequences are supported.
 - The multi-LUN work mode is supported.
 - Command queuing mechanism. *1

Note 1.

The queue is used when write access to Registers is performed by the host. It is valid for all registers except:

- INT_STATUS
- STATUS
- FIFO_INIT
- ECC_STAT
- FIFO_DATA

Write to registers listed above will be performed immediately. Write to any other register will be performed through the command queue.

- The advanced Bad Blocks Management system
 - Records tables are stored in the system memory.
 - Hardware implementation of the search algorithm.

58.1.2 Block Diagram

Figure 58.1 is a block diagram of the R-NANDC.

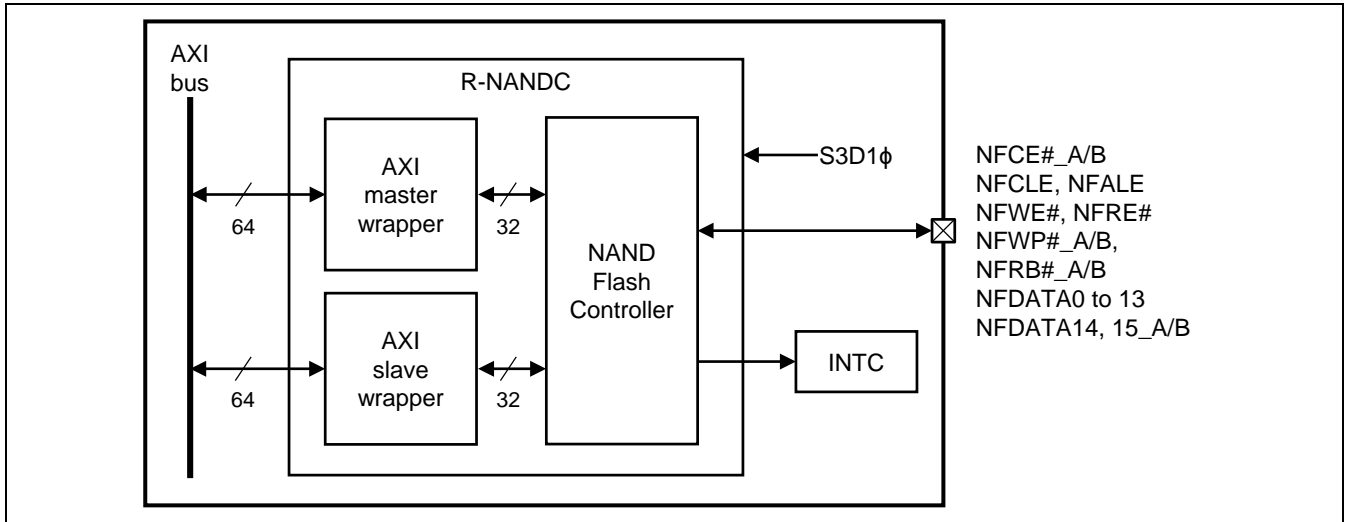


Figure 58.1 Block Diagram [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N]

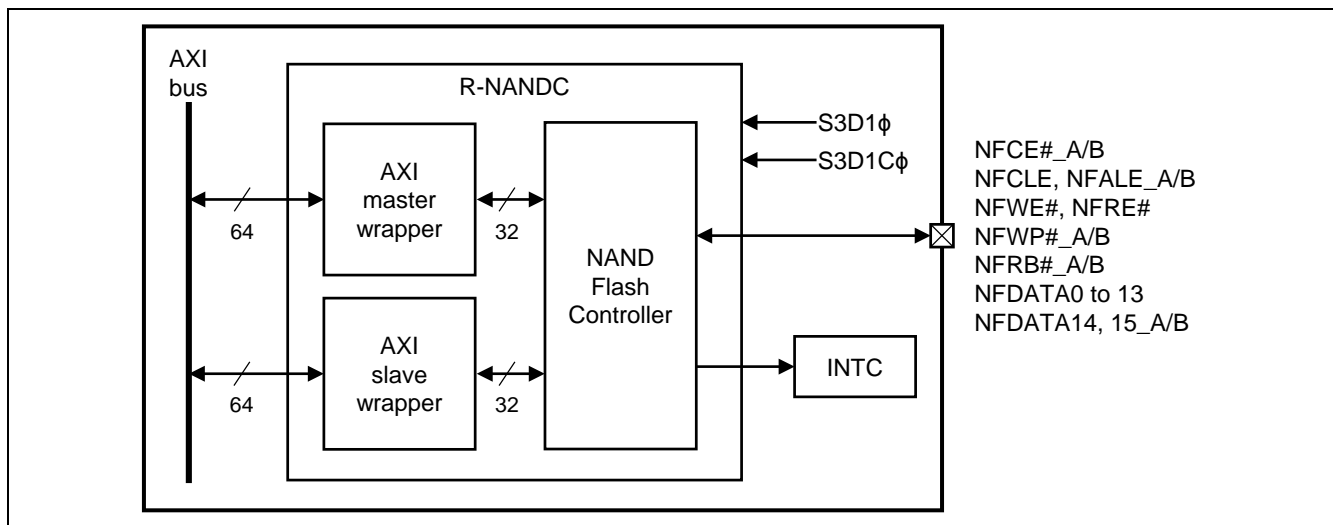


Figure 58.2 Block Diagram [RZ/G2E]

58.1.3 External Pins

Table 58.1 lists the input and output pins used by the interface. The voltage of the R-NANDC terminal is variable. Therefore a power network is separate. When choosing the function of GPIO and the other ones, be careful.

Table 58.1 Pin Configuration [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Name | I/O | Function | Voltage (1.8V/3.3V) | Second Generation RZ/G Series Products | | | |
|--------------|-----|----------------------|--|---|----------------------------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| NFRE# | O | Read Enable | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFCE#_A/B | O | Chip Enable | 3.3V(NFCE#_A) and Voltage switch of 1.8V/3.3V(NFCE#_B) | — | √ | √ | √ |
| NFCLE | O | Command Latch Enable | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFALE | O | Address Latch Enable | Voltage switch of 1.8V/3.3V | — | √ | √ | — |
| NFALE_A/B | O | Address Latch Enable | 3.3V(NFALE_A) and Voltage switch of 1.8V/3.3V(NFALE_B) | — | — | — | √ |
| NFWE# | O | Write Enable | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFWP#_A/B | O | Write Protect | 3.3V(NFWP#_A) and Voltage switch of 1.8V/3.3V(NFWP#_B) | — | √ | √ | √ |
| NFDATA0 | I/O | Data [bit 0] | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFDATA1 | I/O | Data [bit 1] | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFDATA2 | I/O | Data [bit 2] | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFDATA3 | I/O | Data [bit 3] | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFDATA4 | I/O | Data [bit 4] | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFDATA5 | I/O | Data [bit 5] | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFDATA6 | I/O | Data [bit 6] | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFDATA7 | I/O | Data [bit 7] | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFDATA8 | I/O | Data [bit 8] | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFDATA9 | I/O | Data [bit 9] | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFDATA10 | I/O | Data [bit 10] | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFDATA11 | I/O | Data [bit 11] | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFDATA12 | I/O | Data [bit 12] | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFDATA13 | I/O | Data [bit 13] | Voltage switch of 1.8V/3.3V | — | √ | √ | √ |
| NFDATA14_A/B | I/O | Data [bit 14] | 3.3V(NFDATA14_A) and Voltage switch of 1.8V/3.3V(NFDATA14_B) | — | √ | √ | √ |
| NFDATA15_A/B | I/O | Data [bit 15] | 3.3V(NFDATA15_A) and Voltage switch of 1.8V/3.3V(NFDATA15_B) | — | √ | √ | √ |
| NFRB#_A/B | I | Ready/busy status | 3.3V(NFRB#_A) and Voltage switch of 1.8V/3.3V (NFRB#_B) | — | √ | √ | √ |

58.1.4 Register Configuration

Table 58.2 lists the R-NANDC related registers and their configurations.

Table 58.2 Configurations of R-NANDC-related Registers

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|--|---------------|-----|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Controller commands register | COMMAND | R/W | H'EE18_0000 | H'0000_0000 | 32 | — | √ | √ | √ |
| Main configurations register | CONTROL | R/W | H'EE18_0004 | H'0000_0000 | 32 | — | √ | √ | √ |
| Controller status register | STATUS | R | H'EE18_0008 | H'0000_0001 | 32 | — | √ | √ | √ |
| Mask register for the READ STATUS commands | STATUS_MASK | R/W | H'EE18_000C | H'0000_4040 | 32 | — | √ | √ | √ |
| Interrupts mask register | INT_MASK | R/W | H'EE18_0010 | H'0000_0000 | 32 | — | √ | √ | √ |
| Interrupts status register | INT_STATUS | R/W | H'EE18_0014 | H'0000_0000 | 32 | — | √ | √ | √ |
| ECC module control register | ECC_CTRL | R/W | H'EE18_0018 | H'0000_0000 | 32 | — | √ | √ | √ |
| ECC offset in the spare area register | ECC_OFFSET | R/W | H'EE18_001C | H'0000_0000 | 32 | — | √ | √ | √ |
| ECC module status register | ECC_STAT | R/W | H'EE18_0020 | H'0000_0000 | 32 | — | √ | √ | √ |
| Column address register 0 | ADDR0_COL | R/W | H'EE18_0024 | H'0000_0000 | 32 | — | √ | √ | √ |
| Row address register 0 | ADDR0_ROW | R/W | H'EE18_0028 | H'0000_0000 | 32 | — | √ | √ | √ |
| Column address register 1 | ADDR1_COL | R/W | H'EE18_002C | H'0000_0000 | 32 | — | √ | √ | √ |
| Row address register 1 | ADDR1_ROW | R/W | H'EE18_0030 | H'0000_0000 | 32 | — | √ | √ | √ |
| FIFO module interface | FIFO_DATA | R/W | H'EE18_0038 | — | 32 | — | √ | √ | √ |
| Data register | DATA_REG | R/W | H'EE18_003C | H'0000_0000 | 32 | — | √ | √ | √ |
| Data register size selection register | DATA_REG_SIZE | R/W | H'EE18_0040 | H'0000_0000 | 32 | — | √ | √ | √ |
| Records table pointer register | DEV0_PTR | R/W | H'EE18_0044 | H'0000_0000 | 32 | — | √ | √ | √ |
| DMA base address - least significant part register | DMA_ADDR_L | R/W | H'EE18_0064 | H'0000_0000 | 32 | — | √ | √ | √ |
| DMA module counter initial value | DMA_CNT | R/W | H'EE18_006C | H'0000_0000 | 32 | — | √ | √ | √ |
| DMA control register | DMA_CTRL | R/W | H'EE18_0070 | H'0000_0001 | 32 | — | √ | √ | √ |
| Bad block management (BBM) module control register | BBM_CTRL | R/W | H'EE18_0074 | H'0000_0000 | 32 | — | √ | √ | √ |
| Page size value register | DATA_SIZE | R/W | H'EE18_0084 | H'0000_0000 | 32 | — | √ | √ | √ |
| Timing configuration register | TIMINGS_ASYN | R/W | H'EE18_0088 | H'0000_0000 | 32 | — | √ | √ | √ |

| Name | Abbreviation | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|--|----------------|-----|-------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Command sequence timing configuration register 0 | TIME_SEQ_0 | R/W | H'EE18_0090 | H'0000_0000 | 32 | — | √ | √ | √ |
| Command sequence timing configuration register 1 | TIME_SEQ_1 | R/W | H'EE18_0094 | H'0000_0000 | 32 | — | √ | √ | √ |
| Generic command sequence timing configuration register 0 | TIME_GEN_SEQ_0 | R/W | H'EE18_0098 | H'0000_0000 | 32 | — | √ | √ | √ |
| Generic command sequence timing configuration register 1 | TIME_GEN_SEQ_1 | R/W | H'EE18_009C | H'0000_0000 | 32 | — | √ | √ | √ |
| Generic command sequence timing configuration register 2 | TIME_GEN_SEQ_2 | R/W | H'EE18_00A0 | H'0000_0000 | 32 | — | √ | √ | √ |
| FIFO module control register | FIFO_INIT | R | H'EE18_00B0 | H'0000_0000 | 32 | — | √ | √ | √ |
| FIFO module status register | FIFO_STATE | R | H'EE18_00B4 | H'0000_0095 | 32 | — | √ | √ | √ |
| GENERIC_SEQ register | GEN_SEQ_CTRL | R/W | H'EE18_00B8 | H'0000_0000 | 32 | — | √ | √ | √ |
| MLUN register | MLUN | R/W | H'EE18_00BC | H'0000_0000 | 32 | — | √ | √ | √ |
| Records table size register | DEV0_SIZE | R/W | H'EE18_00C0 | H'0000_0000 | 32 | — | √ | √ | √ |
| DMA trigger level value register | DMA_TRIG_TLVL | R/W | H'EE18_0114 | H'0000_0000 | 32 | — | √ | √ | √ |
| CMD ID initial value register | CMD_MARK | W | H'EE18_0124 | H'0000_0000 | 32 | — | √ | √ | √ |
| LUN per device status register | LUN_STATUS 0 | R | H'EE18_0128 | H'0000_00FF | 32 | — | √ | √ | √ |
| Generic command sequence timing configuration register 3 | TIME_GEN_SEQ_3 | R/W | H'EE18_0134 | H'0000_0000 | 32 | — | √ | √ | √ |
| ECC error level counter register | ECC_CNT | R/W | H'EE18_014C | H'0000_0000 | 32 | — | √ | √ | √ |
| Memory device control register | MEM_CTRL | R/W | H'EE18_0080 | H'0000_0000 | 32 | — | √ | √ | √ |

58.1.5 Connected Module**Table 58.3 Connected Module**

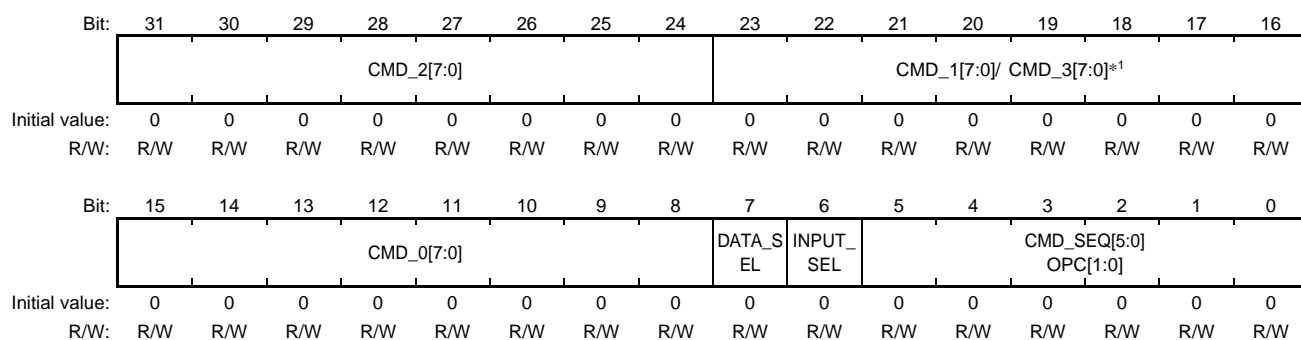
| Module name | Connected module name | Function of connected module |
|--------------------|--------------------------------|---|
| R-NANDC | AXI-bus | Register access of CPU DMA access of R-NANDC |
| | CPG | Clock output |
| | PFC | Selection of external pins |
| | Module Standby, Software Reset | Clock stop control |
| | RST | reset execution |
| | INTC | Interrupt control |

58.2 Register Description

58.2.1 Controller commands register (COMMAND)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The write of the command sequence code to the COMMAND register triggers the programmed command sequence execution as soon as it is possible. If the execution cannot be started immediately, the transfer to this register is prolonged by the series of the WAIT responses. Each command sequence can trigger the interrupt when it is completed.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------------------|---------------|-----|---|
| 31 to 24 | CMD_2[7:0] | H'00 | R/W | Code of the third command in a sequence. |
| 23 to 16 | CMD_1[7:0]/ CMD_3[7:0]*1 | H'00 | R/W | Code of the second command in a sequence. |
| 15 to 8 | CMD_0[7:0] | H'00 | R/W | Code of the first command in a sequence. |
| 7 | DATA_SEL | B'0 | R/W | Data / FIFO selection flag: 0 – the FIFO module selected 1 – the DATA register selected |
| 6 | INPUT_SEL | B'0 | R/W | Input module selection flag: 0 – select the registers Slave I/F Unit (SIU) module as input 1 – select the DMA module as input |
| 5 to 0 | CMD_SEQ[5:0] | B'00_0000 | R/W | Command code. |

Note 1. Depending on the selected command sequence, this field will store CMD1 or CMD3 code.
Generic Sequence is only one case where both commands are used in a single sequence

58.2.2 Main configurations register (CONTROL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The CONTROL register stores the configuration parameters that are common to all controller modules.

| | | | | | | | | | | | | | | | | |
|----------------|-------------|----|--------|----------|----|----|----|----|-------------------|---------|----------------|-----|----------------------|-----|-----------------|-----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | AUTO_READ_STAT_EN | MLUN_EN | SMALL_BLOCK_EN | — | — | — | ADDR1_AUTO_INCR | ADDR0_AUTO_INCR |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | NFMA_FASTTR | — | BBM_EN | IO_WIDTH | — | — | — | — | BLOCK_SIZE [1:0] | ECC_EN | INT_EN | — | ECC_BLOCK_SIZE [1:0] | — | — | READ_STATUS_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R/W | R | R | R | R | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 23 | AUTO_READ_STAT_EN | B'0 | R/W | Auto Read Status mode enable. If active, the controller reads the status after the PROGRAM PAGE and BLOCK ERASE commands. It can trigger interrupt. The ERROR_MASK field in the STATUS_MASK register must be configured when this feature is enabled. It enables operation status checking after PROGRAM_PAGE/ERASE commands. It allow to detect that those operations failed. 0: Auto Read Status mode disabled 1: Auto Read Status mode enabled |
| 22 | MLUN_EN | B'0 | R/W | Multi LUN mode enables. If active, it enables controller multi LUN work mode 0: Multi LUN mode disabled 1: Multi LUN mode enabled For more details, see section 58.3.4, Multi LUN Work Mode. |
| 21 | SMALL_BLOCK_EN | B'0 | R/W | Enable small block mode. In this mode, the controller sends only a single byte as the column address instead of two bytes, as done in the big block NAND Flash devices. 0: big block mode enabled 1: small block mode enabled Note: Only devices that allow to disable CE signal when device is in the busy state are supported. |
| 20 to 18 | — | B'000 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------|---------------|-----|---|
| 17 | ADDR1_ AUTO_INCR | B'0 | R/W | Address auto increment for row address register 1 (ADDR1_ROW). 0: auto increment disabled 1: auto increment enabled When this bit is set, sending any command sequence using address register 1 causes the increment of address register 1. |
| 16 | ADDR0_ AUTO_INCR | B'0 | R/W | Address auto increment for row address register 0 (ADDR0_ROW). 0: auto increment disabled 1: auto increment enabled When this bit is set, sending any command sequence using address register 0 causes the increment of address register 0 |
| 15 | NFMA FASTTR | B'0 | R/W | NFMA fast transition mode control 0: normal mode 1: fast mode When using NAND flash memory which support ONFI 1.0 EDO mode, TIMINGS_ASYN.TRWP value can be one reduced (-1) from normal setting by setting this bit. With this setting, data read transfer speed is improved. |
| 14 | — | B'0 | R | Reserved bit. This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to the bit. |
| 13 | BBM_EN | B'0 | R/W | Bad Block Management enable flag For more details, see section 58.3.5, Remapping Mechanism. |
| 12 | IO_WIDTH | B'0 | R/W | NAND Flash I/O width 0: 8 bits 1: 16 bits |
| 11 to 8 | — | H'0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 7, 6 | BLOCK_SIZE [1:0] | B'00 | R/W | The Block Size. B'00: 32 pages per block B'01: 64 pages per block B'10: 128 pages per block B'11: 256 pages per block |
| 5 | ECC_EN | B'0 | R/W | Hardware ECC support enable 0: ECC disabled 1: ECC enabled Hardware ECC can be used only when $m * (ECC_BLOCK_SIZE) \leq DATA_SIZE \leq m * (ECC_BLOCK_SIZE + 32)$, where m is 1,2,3,.. |
| 4 | INT_EN | B'0 | R/W | Global Interrupt enable 0: Interrupts disabled 1: Interrupts enabled For more details, see section 58.3.6, Interrupts Mechanism. |
| 3 | — | 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|---------------------|---------------|-----|---|
| 2, 1 | ECC_BLOCK_SIZE[1:0] | B'00 | R/W | <p>The ECC Block Size (depends on core configuration):</p> <p>B'00: 256 bytes</p> <p>B'01: 512 bytes</p> <p>B'10: 1024 bytes</p> <p>B'11: not available</p> <p>The ECC block size can be changed only when all memory devices are ready.</p> |
| 0 | READ_STATUS_EN | B'0 | R/W | <p>Automatically READ STATUS / check RnB lines. The STATUS_MASK field in the STATUS_MASK register must be configured when this feature is enabled.</p> <p>It select how controller detect device ready/busy state:</p> <p>0: The controller checks RnB lines</p> <p>1: The controller sends READ STATUS commands</p> <p>Note:</p> <p>Automatically sent READ STATUS command is only available in devices compatible with ONFI 1.0</p> |

58.2.3 GENERIC_SEQ register (GEN_SEQ_CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The GENERIC_SEQ register is used to parameterize the generic command sequences. For more details, see section 58.3.2, Generic Command Sequence. For examples, see Table 58.7

| | | | | | | | | | | | | | | | | |
|----------------|---------|---------------|---------|-------------|-------------|-------------|-------------|---------|------------|---------|---------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | CMD_3[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | IMD_SEQ | DELAY_EN[1:0] | DATA_EN | ROW_A1[1:0] | ROW_A0[1:0] | COL_A1[1:0] | COL_A0[1:0] | CMD3_EN | CMD2_EN | CMD1_EN | CMD0_EN | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 23 to 16 | CMD_3[7:0] | H'00 | R/W | Command 3 code value. |
| 15 | IMD_SEQ | B'00 | R/W | Enable immediate command execution. This bit allows the command sequence to be executed without checking the selected target state. 0: feature disabled 1: feature enable |
| 14, 13 | DELAY_EN [1:0] | B'00 | R/W | Enable the busy 0 or 1 phase. This bit allows enabling or disabling the presence of the “busy” phase in the universal command sequence. B'00: disable both delays B'01: enable delay 0 B'10: enable delay 1 B'11: disable both delays |
| 12 | DATA_EN | B'0 | R/W | Enable data part sequence This bit allows enabling or disabling the data phase of the universal command sequence. 0: disable data phase 1: enable data phase |
| 11, 10 | ROW_A1[1:0] | B'00 | R/W | Row Address Cycles. Number of the row address bytes sent to NAND Flash device. B'00: 0 address cycles B'01: 1 address cycles B'10: 2 address cycles B'11: 3 address cycles |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|-------------|---------------|-----|--|
| 9, 8 | ROW_A0[1:0] | B'00 | R/W | Row Address Cycles. Number of the row address bytes sent to NAND Flash device. B'00: 0 address cycles B'01: 1 address cycles B'10: 2 address cycles B'11: 3 address cycles |
| 7, 6 | COL_A1[1:0] | B'00 | R/W | Column Address Cycles. Number of the column address bytes sent to NAND Flash device. B'00: 0 address cycles B'01: 1 address cycles B'10: 2 address cycles B'11: not available |
| 5, 4 | COL_A0[1:0] | B'00 | R/W | Column Address Cycles. Number of the column address bytes sent to NAND Flash device. B'00: 0 address cycles B'01: 1 address cycles B'10: 2 address cycles B'11: not available |
| 3 | CMD3_EN | B'0 | R/W | This bit allows enabling or disabling the presence of the "command 3" phase in the universal command sequence. 1: enable 0: disable |
| 2 | CMD2_EN | B'0 | R/W | Enable Command 2 phase This bit allows enabling or disabling the presence of the "command 2" phase in the universal command sequence. 1: enable 0: disable |
| 1 | CMD1_EN | B'0 | R/W | Enable Command 1 phase. This bit allows enabling or disabling the presence of the "command 1" phase in the universal command sequence. 1: enable 0: disable |
| 0 | CMD0_EN | B'0 | R/W | Enable Command 0 phase. This bit allows enabling or disabling the presence of the "command 0" phase in the universal command sequence. 1: enable 0: disable |

58.2.4 Controller status register (STATUS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The STATUS register stores the NAND Flash controller and connected device status flags. These flags can be used to obtain the current controller internal state.

The CTRL_STAT flag is set after the controller starts to execute the requested command for the selected NAND Flash device, and it is active while the command execution is not completed. Command execution can be divided into two phases. In the first phase, the command sequence is executed at the moment when the NAND Flash device goes into the busy state. After that, the controller stores information about the pending operation on the selected device, but is also able to execute a new command on any other device. In the second phase, the controller automatically finishes the pending command execution based on the previously stored data. As long as this flag is set, the controller does not accept any new commands.

The MEMx_ST flags correspond to the NAND Flash device with the same index value. The flags give information about the NAND Flash devices' state. The flag has the same function as the NAND Flash device RnB line.

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|-------------|-------------------|-----------|-------------|----|----|----|----|----|----|---------|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | CMD_ID[7:0] | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | DATA_REG_ST | DATASIZE_ERROR_ST | CTRL_STAT | — | — | — | — | — | — | — | MEMO_ST | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------|---------------|-----|---|
| 31 to 24 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 23 to 16 | CMD_ID[7:0] | H'00 | R | Command ID The current command under execution identification marker added before command was put in to the command FIFO. |
| 15 to 11 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 10 | DATA_REG_ST | B'0 | R | The DATA_REG: Resetting of this flag is possible only by reading the data from the DATA_REG register. 1: data in DATA_REG is available 0: data in DATA_REG is not available |
| 9 | DATASIZE_ERROR_ST | B'0 | R | The Data Size value error: When the ECC is enabled, this bit signal incorrect value in the DATA_SIZE register. The algorithm for correct data size value can be found in the DATA_SIZE register description. 0: correct value 1: incorrect value |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|---|
| 8 | CTRL_STAT | B'0 | R | The main controller status bit: 0: controller ready 1: controller busy |
| 7 to 1 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 0 | MEMO_ST | B'1 | R | Device 0 status flag: 1: device ready 0: device busy |

Note: When using "READ PARAMETER PAGE" command, ECC Block Size should same with DATA SIZE.
When using "SET FEATURES" or "GET FEATURES" or "READ ID" commands, ECC function should be disabled.

58.2.5 LUN per device status register (LUN_STATUS_0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The LUN_STATUS0 register allows to access the LUN status information for devices 0-3. Each bit of the LUN status field contain the status of single LUN of device. The busy status is marked as logical '0', the ready state is marked as logical '1'.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | MEMO_LUN[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 7 to 0 | MEMO_LUN [7:0] | H'FF | R | Memory 0 LUN-s status field. |

58.2.6 Interrupts mask register (INT_MASK)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The INT_MASK register allows masking of the selected interrupts source in the NAND Flash controller. The masked interrupts still set the appropriate bits in the status register, but do not assert the interrupt signal.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----------------|----|------------------|----|--------------|------------|-----------------|----------------|------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | ECC_INT0_EN | — | — | — | — | — | — | — | STAT_ERR_INT0_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | MEMO_RDY_INT_EN | — | PG_SZ_ERR_INT_EN | — | TRANS_ERR_EN | DMA_INT_EN | DATA_REG_INT_EN | CMD_END_INT_EN | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R/W | R | R/W | R/W | R/W | R/W | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|---|
| 31 to 25 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 24 | ECC_INT0_EN | B'0 | R/W | Enables the interrupt from the ECC module status for Memory device 0. 0: interrupt disabled 0: interrupt disabled 1: interrupt enabled |
| 23 to 17 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 16 | STAT_ERR_INT0_EN | B'0 | R/W | Enables the interrupt when the most recently finished operation on the Memory device 0 failed. This applies to PROGRAM PAGE and BLOCK ERASE operations. It is not valid following a READ-series operation. 0: interrupt disabled 1: interrupt enabled |
| 15 to 9 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 8 | MEMO_RDY_INT_EN | B'0 | R/W | The memory device 0 is ready for the new command: 0: interrupt disabled 1: interrupt enabled For more details see Figure 58.32 |
| 7 | — | B'0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|------------------|---------------|-----|---|
| 6 | PG_SZ_ERR_INT_EN | B'0 | R/W | Data Size error occur. 0: interrupt disabled 1: interrupt enabled |
| 5 | — | B'0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 4 | TRANS_ERR_EN | B'0 | R/W | The transfer on the slave interface error: 0: interrupt disabled 1: interrupt enabled |
| 3 | DMA_INT_EN | B'0 | R/W | DMA transfer ended. 0: interrupt disabled 1: interrupt enabled |
| 2 | DATA_REG_INT_EN | B'0 | R/W | Data in DATA_REG is available. 0: interrupt disabled 1: interrupt enabled |
| 1 | CMD_END_INT_EN | B'0 | R/W | Command sequence ended 0: interrupt disabled 1: interrupt enabled For more details see Figure 58.32 |
| 0 | — | B'0 | R | Reserved bit. This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to the bit. |

58.2.7 Interrupts status register (INT_STATUS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The INT_STATUS register stores the NAND Flash controller interrupt flags. If the given bit is 0, the corresponding interrupt condition is not met. If the given bit is 1, the corresponding interrupt condition is met.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----------------|----|------------------|----|--------------|------------|-----------------|----------------|------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | ECC_INT0_FL | — | — | — | — | — | — | — | STAT_ERR_INT0_FL |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | MEMO_RDY_INT_FL | — | PG_SZ_ERR_INT_FL | — | TRANS_ERR_FL | DMA_INT_FL | DATA_REG_INT_FL | CMD_END_INT_FL | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R/W | R | R/W | R/W | R/W | R/W | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|--|
| 31 to 25 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 24 | ECC_INT0_FL | B'0 | R/W | Selected flag (source is ECC_UNC_0, ECC_ERROR_0 or ECC_OVER_0) in the ECC module is set. |
| 23 to 17 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 16 | STAT_ERR_INT0_FL | B'0 | R/W | Most recently finished operation on the Memory device 0 failed. This applies to PROGRAM PAGE and BLOCK ERASE operations. It is not valid following a READ-series operation. |
| 15 to 9 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 8 | MEMO_RDY_INT_FL | B'0 | R/W | The memory device 0 is ready for the new command. |
| 7 | — | B'0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 6 | PG_SZ_ERR_INT_FL | B'0 | R/W | Data Size error flag. When the ECC is enabled, the value written into the DATA_SIZE register has some restrictions. Interrupt condition is met when the value written to the DATA_SIZE register is not correct. |
| 5 | — | B'0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------------|---------------|-----|--|
| 4 | TRANS_ERR_FL | B'0 | R/W | The transfer on the slave interface error. The flag is set when the access to the FIFO memory has the opposite direction to the current FIFO configuration. |
| 3 | DMA_INT_FL | B'0 | R/W | DMA transfer ended flag. |
| 2 | DATA_REG_INT_FL | B'0 | R/W | Data in DATA_REG is available. |
| 1 | CMD_END_INT_FL | B'0 | R/W | Transfer sequence ended. |
| 0 | — | B'0 | R | Reserved bit. This bit is always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to the bit. |

58.2.8 ECC module control register (ECC_CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The ECC_CTRL register stores all configuration parameters required by the ECC.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|--------------------|-----|-----|-----|-----|-----|----|----|----|----|--------------|-----|--------------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ECC_SEL[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | ERR_THRESHOLD[5:0] | | | | | — | — | — | — | — | ECC_CAP[2:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|---|
| 31 to 18 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 17,16 | ECC_SEL [1:0] | B'00 | R/W | The ECC interrupt source select. These bits selects the ECC module flag that will be used as a source for the interrupt signal: B'00: select ECC_ERROR (correctable error) flag as interrupt source. B'01: select ECC_UNC (uncorrectable error) flag as interrupt source. B'1x: select ECC_OVER (acceptable errors level overflow) flag as interrupt source. (ECC_OVER flag is not set by uncorrectable errors (ECC_UNC_0 = 1)) |
| 15,14 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 13 to 8 | ERR_THRES HOLD[5:0] | B'00_0000 | R/W | The acceptable errors level. The value of this field contains the number of errors that is acceptable. This field must be initialized by the application program. |
| 7 to 3 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 2 to 0 | ECC_CAP [2:0] | B'000 | R/W | The ECC module correction ability. The correction ability can be changed only when all memory devices are ready. B'000: 2 B'001: 4 B'010: 8 B'011: 16 B'100: 24 All others: 32 |

58.2.9 ECC module status register (ECC_STAT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The ECC_STAT register stores all ECC module status information.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----------|----|----|----|----|----|----|----|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ECC_OVER_0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | ECC_UNC_0 | — | — | — | — | — | — | — | ECC_ERROR_0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|---|
| 31 to 17 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 16 | ECC_OVER_0 | B'0 | R/W | The Memory device 0 acceptable errors level overflow. The bit is set when the number of errors is bigger than the value of declared ERR_THRESHOLD bits. (uncorrectable errors are not counted) |
| 15 to 9 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 8 | ECC_UNC_0 | B'0 | R/W | The Memory device 0 uncorrectable error flag. The bit is set when the uncorrectable errors occur during the read operation |
| 7 to 1 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 0 | ECC_ERRO_R_0 | B'0 | R/W | The Memory device 0 correctable error flag. The bit is set when the correctable errors or the uncorrectable errors (ECC_UNC_0 = 1) occur during the read operation. |

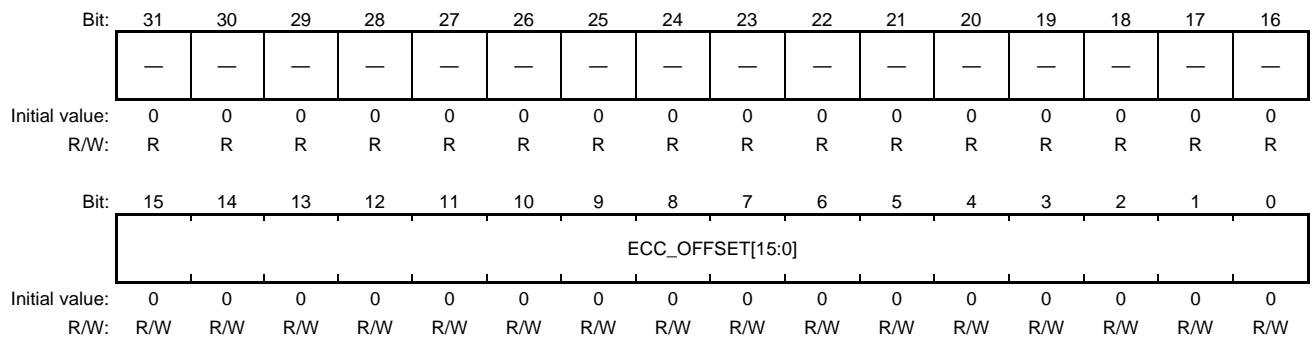
58.2.10 ECC offset in the spare area register (ECC_OFFSET)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The ECC_OFFSET register stores the offset value from the beginning of the page to the place where correction words will be stored.

The value of the ECC_OFFSET register must be bigger than the value in the DATA_SIZE register.

In small block mode, the value in the ECC_OFFSET is ignored and the correction words are located in the NAND Flash memory device just behind the data.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 15 to 0 | ECC_OFFSET [15:0] | H'0000 | R/W | Correction words block offset. |

58.2.11 ECC error level counter register (ECC_CNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The ECC_CNT register stores number of value detected during last page read operation. This register content is not automatically cleared, it must be done by software. The new page read operation does not overwrite previous register value. Register contain value of the largest error level detected in processed ECC blocks.

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|--------------|-----|-----|-----|-----|-----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | — | — | — | — | — | ERR_LVL[5:0] | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | |

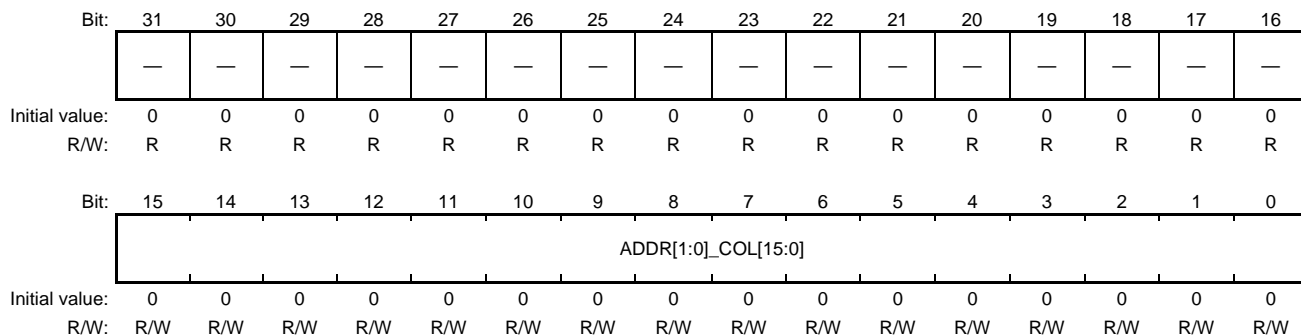
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 6 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 5 to 0 | ERR_LVL [5:0] | H'00 | R/W | Detected error level. (uncorrectable errors are not counted) |

58.2.12 Column/row address registers (ADDR[1:0]_COL, ADDR[1:0]_ROW)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

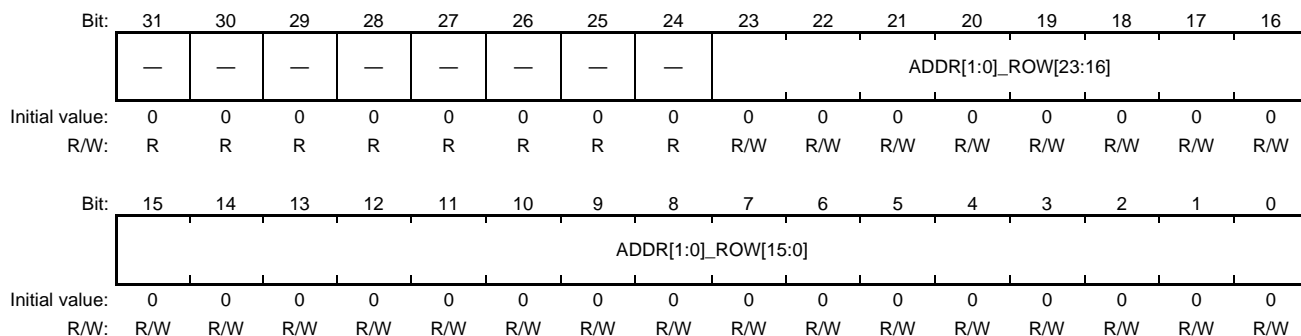
The address registers store the packaged version of the address that will be used by the next command sequence during access to the NAND Flash device.

ADDR [1:0] _COL:



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 15 to 0 | ADDR[1:0]_COL[15:0] | H'0000 | R/W | Column address. A15-A0 address bits. |

ADDR [1:0] _ROW:



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|---|
| 31 to 24 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 23 to 0 | ADDR[1:0]_ROW[23:0] | H'00_0000 | R/W | Row address. A39-A16 address bits (Page address, Block address and LUN address in the ONFI case). |

Notes:

1. There is no register that defines the total memory size of the NAND Flash memory chip, thus the controller is not able to determine which address bits in ADDR [1:0]_COL and ADDR[1:0]_ROW are important and which must be zero. For this reason, the software must take special care with the values written to these registers. Incorrect values of unused address bits (none '0' values) can cause errors in memory access.

A relation between address registers and memory device address width is configured by the command sequence field of the COMMAND register. This field determines which command sequence has to be used and how many address bytes are used when addressing a NAND Flash memory device. (Example: In order to erase blocks, the three address cycles containing the row address are written into the NAND Flash memory device. The NAND Flash Controller automatically writes bits A39-A16 to the NAND Flash device). Refer to section 58.3.1.(2), Command Sequence Encoding and Table 58.8 in order to see how many address cycles are written into the NAND Flash memory device by each Command Sequence.

The address written to the address register must be aligned according to the NAND Flash device. Unused bits must be padded with zeros.

2. When the auto increment for the row address register is enabled, the proper value of this register can be read only when the bit CTRL_STAT in STATUS register is clear.

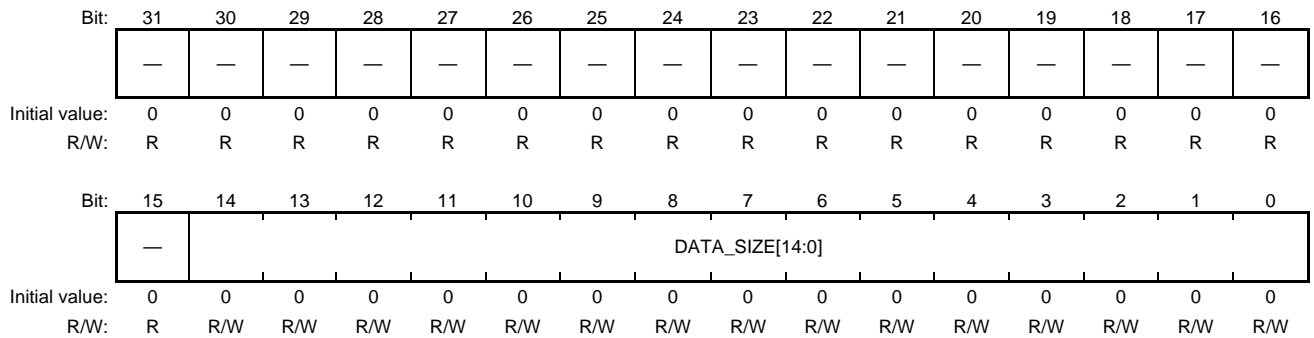
Table 58.4 Address Registers and Address Bytes Relationship

| Address Cycle | I/O 0 | I/O 1 | I/O 2 | I/O 3 | I/O 4 | I/O 5 | I/O 6 | I/O 7 |
|-----------------------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1 st cycle | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 |
| 2 nd cycle | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 |
| 3 rd cycle | A16 | A17 | A18 | A19 | A20 | A21 | A22 | A23 |
| 4 th cycle | A24 | A25 | A26 | A27 | A28 | A29 | A30 | A31 |
| 5 th cycle | A32 | A33 | A34 | A35 | A36 | A37 | A38 | A39 |

58.2.13 Page size value register (DATA_SIZE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The DATA_SIZE register stores the value of the data block size. The data size value is remembered as the number of bytes per transferred block, but its size must be declared as the multiple of the chosen NAND Flash word size. The unused bits for the given word size configuration are ignored and replaced with zeros.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|---|
| 31 to 15 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 14 to 0 | DATA_SIZE [14:0] | H'0000 | R/W | Data size. The value of this field defines data size. |

Note:

Write the proper value to the DATA_SIZE register when ECC is enabled:

$$(ECC_BLOCK_SIZE) \times m \leq DATA_SIZE \leq (ECC_BLOCK_SIZE + 32) \times m,$$

Where m is 1, 2, 3 ...

Here is an example of which values are correct when ECC_BLOCK_SIZE equals 512 bytes:

Table 58.5 Example, in which DATA_SIZE Values are Correct

| DATA_SIZE | Value Correctness |
|-----------|-------------------|
| H'0000 | Not available |
| | |
| H'001F | Not available |
| H'0020 | Not available |
| H'0021 | Not available |
| | |
| H'01FF | Not available |

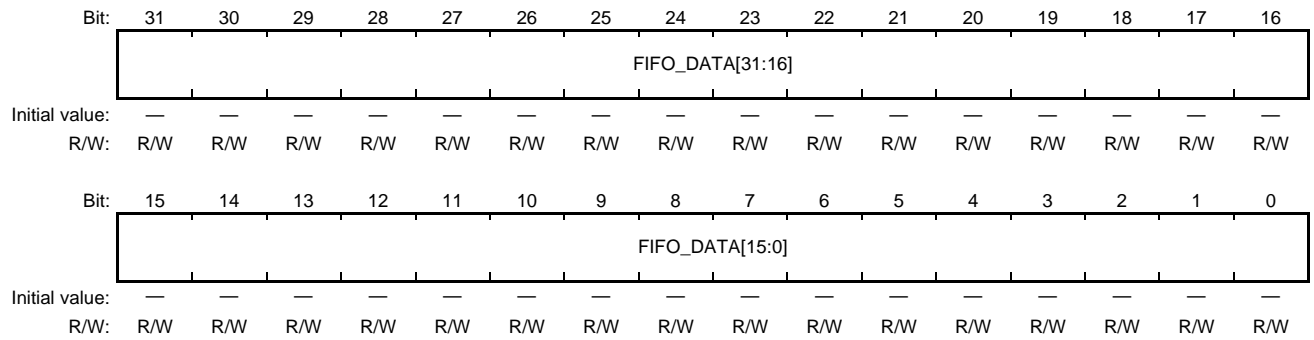
| DATA_SIZE | Value Correctness |
|------------------|--------------------------|
| H'0200 | Available |
| H'0201 | Available |
| | |
| H'0220 | Available |
| H'0221 | Not available |
| | |
| H'03FF | Not available |
| H'0400 | Available |
| H'0401 | Available |
| | |
| H'0440 | Available |
| H'0440 | Not available |
| | |

Note: ECC_BLOCK_SIZE equals 512 bytes.

58.2.14 FIFO module interface register (FIFO_DATA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The FIFO_DATA register is used as an entry point to the FIFO module. The CPU can access the FIFO module by reading from or writing to the FIFO_DATA register in the same way as it accesses any other register.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------|---------------|-----|-------------|
| 31 to 0 | FIFO_DATA [31:0] | — | R/W | FIFO Data. |

Note:

FIFO_DATA register is able to access after issuing read or write command.

58.2.15 Bad block management (BBM) module control register (BBM_CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The BBM control register stores the BBM specific control parameters.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | RMP_INIT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are read as 0. The write value should be 0. |
| 0 | RMP_INIT | B'0 | R/W | Remap initial flag. If set, this flag forces the BBM module to reread the remapping table after it was updated by software. This flag is set by software and cleared by hardware after rereading the remapping table. |

58.2.16 Records table pointer register (DEV0_PTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The bad block management mechanism uses the tables in the system memory to store the remapping records. The DEV0_PTR register stores the table of the remapping record address.

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|----|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | PTR_ADDR[11:2] | | | | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 11 to 2 | PTR_ADDR [11:2] | B'00_0000_000 | R/W | Remap table pointer. The field contains an address of the remap table in the internal memory. |
| 1 to 0 | — | ALL 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |

58.2.17 Records table size register (DEV0_SIZE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The bad block management mechanism implemented in the controller uses the tables in the system memory to store the remapping records. Each table can store a variable number of records depending on the number of bad blocks in the NAND Flash device.

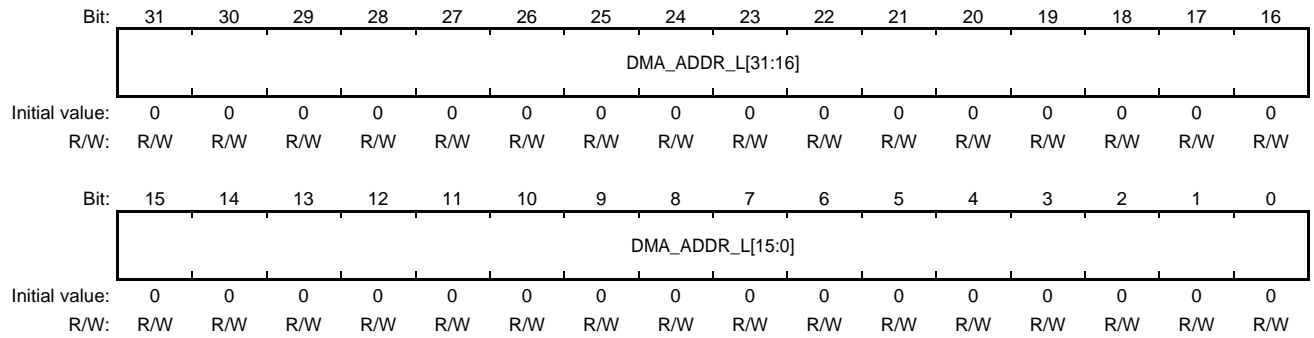
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | DEV_SIZE[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 11 to 0 | DEV_SIZE [11:0] | H'000 | R/W | Number of record |

58.2.18 DMA base address - least significant part register (DMA_ADDR_L)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The DMA_ADDR_L register is the least significant part of the 64-bit DMA base address. The register contains an address for the first data in the data block in the system memory, or the address of the first descriptor. The DMA module can read data from the memory location set by DMA_ADDR and write it to the FIFO module, or read data from the FIFO module and write it to the memory, starting from the location indicated by the DMA_ADDR.

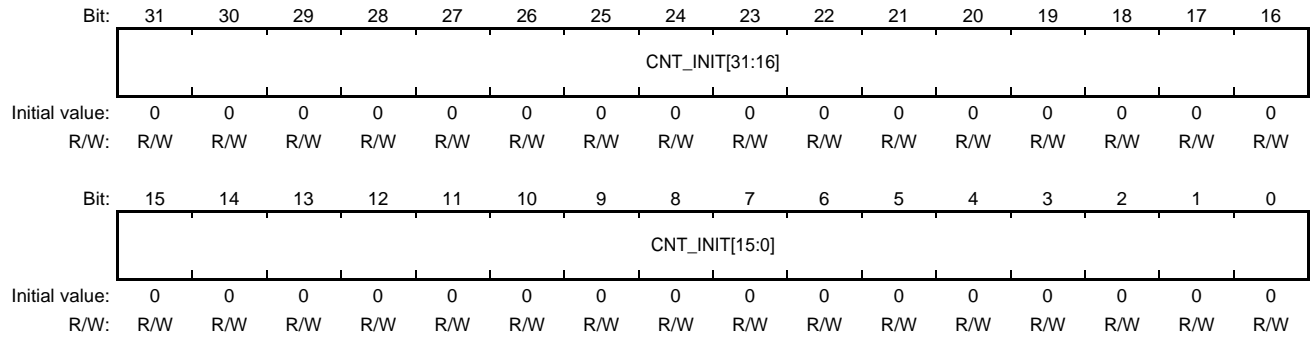


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------|---------------|-----|---|
| 31 to 0 | DMA_ADDR_L[31:0] | H'0000_0000 | R/W | The least significant part of the DMA base address. The two least significant bits are ignored, thus the address must be aligned to 32-bit words |

58.2.19 DMA counter initial value register (DMA_CNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The DMA_CNT defines the number of bytes that will be transferred by the DMA module. The register remains unchanged during the transfer process.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|--|
| 31 to 0 | CNT_INIT [31:0] | H'0000_0000 | R/W | Bytes counter initial value. The field contains data page length in bytes (H'0000_0004 – H'FFFF_FFFC). The number of the bytes has to be divided by 4. |

58.2.20 DMA control register (DMA_CTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

DMA_CTRL is a control register for the DMA channel. This register defines the parameters of the DMA transfer.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----------|-----|----------|----------------|-----|-----|-----|-----------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | DMA_START | — | DMA_MODE | DMA_BURST[2:0] | | | — | DMA_READY | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R | R/W | R/W | R/W | R/W | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 7 | DMA_START | B'0 | R/W | Set bit DMA_START to start DMA when the command sequence is sent to the NAND Flash memory. For more details see section 58.4.2.(2), DMA Description. |
| 6 | — | B'0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 5 | DMA_MODE | B'0 | R/W | DMA work mode: 0: the registers managed mode 1: the Scatter-Gather mode |
| 4 to 2 | DMA_BURST [2:0] | B'000 | R/W | Burst type These bits define the main transfer type used by the DMA to precede the requested transfer B'000: incrementing precise burst of precisely four transfers (address increment) B'001: stream precise burst of precisely sixteen transfers (address constant) B'010: single transfer (address increment) B'011: burst of unspecified length (address increment) B'100: incrementing precise burst of precisely eight transfers (address increment) B'101: incrementing precise burst of precisely sixteen transfers (address increment) |
| 1 | — | B'0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 0 | DMA_READY | B'1 | R | DMA ready flag. The flag is set when transfer is completed. |

58.2.21 DMA trigger level value register (DMA_TRIG_TLVL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

This register allows the setting of the data FIFO occupancy level that will trigger the DMA module. For more details see section 58.4.2. (2), DMA Description.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|--------------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | DMA_TRIG_TLVL[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 7 to 0 | DMA_TRIG_TLVL[7:0] | H'00 | R/W | DMA trigger level The trigger level is counted using the 32-bit words as entity. |

58.2.22 Mask register for the READ STATUS commands (STATUS_MASK)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The STATE_MASK field is used to mark the ready/busy bits in the NAND Flash device status byte. This field is used during the internal read status operation. In the case of ONFI, the user must mask all fields except RDY or ARDY (depending on the application).

The ERROR_MASK field is used to mask unused fields when the controller automatically reads the status of the NAND Flash memory device. In the case of ONFI, the user must mask all fields except FAIL or FAILC (depending on the application).

| | | | | | | | | | | | | | | | | |
|----------------|-----------------|-----|-----|-----|-----|-----|-----|-----|-----------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | ERROR_MASK[7:0] | | | | | | | | STATE_MASK[7:0] | | | | | | | |
| Initial value: | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 15 to 8 | ERROR_MASK[7:0] | H'40 | R/W | Error State Mask used to mask the error bits if automatic read status feature is enabled. |
| 7 to 0 | STATE_MASK[7:0] | H'40 | R/W | State Mask used to mask status bits when the read status command is used to obtain the NAND flash status |

58.2.23 Command sequence timing configuration register 0 (TIME_SEQ_0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

[RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N]

Some waveform configuration parameters are defined in the TIME_SEQ_0 register. The time delay generated by the controller equals the minimum value written into the register, increased by 1. All the timings are generated using the S3D1 ϕ clock signal.

[RZ/G2E]

Some waveform configuration parameters are defined in the TIME_SEQ_0 register. The time delay generated by the controller equals the minimum value written into the register, increased by 1. All the timings are generated using the S3D1C ϕ clock signal.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|-----------|-----|-----|-----|-----|-----|----|-----------|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | TWHR[5:0] | | | | | — | — | TRHW[5:0] | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | TADL[5:0] | | | | | — | — | TCCS[5:0] | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 30 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 29 to 24 | TWHR[5:0] | B'00_0000 | R/W | NFWE# high to NFRE# low time |
| 23 to 22 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 21 to 16 | TRHW[5:0] | B'00_0000 | R/W | NFRE# high to NFWE# low time |
| 15 to 14 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 13 to 8 | TADL[5:0] | B'00_0000 | R/W | ALE to data start time |
| 7 to 6 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 5 to 0 | TCCS[5:0] | B'00_0000 | R/W | Change column setup |

58.2.24 Command sequence timing configuration register 1 (TIME_SEQ_1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

[RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N]

Some waveform configuration parameters are defined in the TIME_SEQ_1 register. The time delay generated by the controller equals the minimum value written into the register, increased by 1. All the timings are generated using the S3D1 ϕ clock signal.

[RZ/G2E]

Some waveform configuration parameters are defined in the TIME_SEQ_1 register. The time delay generated by the controller equals the minimum value written into the register, increased by 1. All the timings are generated using the S3D1C ϕ clock signal.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----------|-----|-----|-----|-----|-----|----|----------|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | TRR[5:0] | | | | | — | — | TWB[5:0] | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 14 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 13 to 8 | TRR[5:0] | B'00_0000 | R/W | Read high to Read low TRR time period from rising edge on read/busy input line to the moment when the read enable signal can be asserted |
| 7 to 6 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 5 to 0 | TWB[5:0] | B'00_0000 | R/W | tWB delay. Time period measured from rising edge of the NFWE# or CLK signal to a falling edge on the RnB line or the NAND flash device SR[6] low. |

58.2.25 Generic command sequence timing configuration register 0 (TIME_GEN_SEQ_0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

[RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N]

Some waveform configuration parameters for the Generic Sequence are defined in the TIME_GEN_SEQ_0 register. The time delay generated by the controller equals the minimum value written into the register, increased by 1. For more detail see section 58.3.2, Generic Command Sequence. All the timings are generated using the S3D1 ϕ clock signal.

[RZ/G2E]

Some waveform configuration parameters for the Generic Sequence are defined in the TIME_GEN_SEQ_0 register. The time delay generated by the controller equals the minimum value written into the register, increased by 1. For more detail see section 58.3.2, Generic Command Sequence. All the timings are generated using the S3D1C ϕ clock signal.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|------------|-----|-----|-----|-----|-----|----|----|------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | t0_d3[5:0] | | | | | | — | — | t0_d2[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | t0_d1[5:0] | | | | | | — | — | t0_d0[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 30 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 29 to 24 | t0_d3[5:0] | B'00_0000 | R/W | Command to Data time. The time between sending a command and data transferring. |
| 23 to 22 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 21 to 16 | t0_d2[5:0] | B'00_0000 | R/W | Command to Delay time. The time between sending a command to the NAND Flash memory device and waiting until the memory is ready |
| 15 to 14 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 13 to 8 | t0_d1[5:0] | B'00_0000 | R/W | Command to Command time. The time between two subsequent commands sent to the NAND Flash memory device. |
| 7 to 6 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------------|-----------------|----------------------|------------|---|
| 5 to 0 | t0_d0[5:0] | B'00_0000 | R/W | Command to Address time. The time between sending a command and sending the address to the NAND Flash memory device. |

58.2.26 Generic command sequence timing configuration register 1 (TIME_GEN_SEQ_1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

[RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N]

Some waveform configuration parameters for the Generic Sequence are defined in the TIME_GEN_SEQ_1 register. The time delay generated by the controller equals the minimum value written into the register, increased by 1. For more detail see section 58.3.2, Generic Command Sequence. All the timings are generated using the S3D1φ clock signal.

[RZ/G2E]

Some waveform configuration parameters for the Generic Sequence are defined in the TIME_GEN_SEQ_1 register. The time delay generated by the controller equals the minimum value written into the register, increased by 1. For more detail see section 58.3.2, Generic Command Sequence. All the timings are generated using the S3D1Cφ clock signal.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|------------|-----|-----|-----|-----|-----|----|----|------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | t0_d7[5:0] | | | | | | — | — | t0_d6[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | t0_d5[5:0] | | | | | | — | — | t0_d4[5:0] | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 30 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 29 to 24 | t0_d7[5:0] | B'00_0000 | R/W | Address to Data time. The time between sending the address and data transferring. |
| 23 to 22 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 21 to 16 | t0_d6[5:0] | B'00_0000 | R/W | Address to Delay time. The time between sending the address to the NAND Flash memory device and waiting until the memory is ready. |
| 15 to 14 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 13 to 8 | t0_d5[5:0] | B'00_0000 | R/W | Address to Address time. The time between two subsequent addresses sent to the NAND Flash memory device. |
| 7 to 6 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|---|
| 5 to 0 | t0_d4[5:0] | B'00_0000 | R/W | Address to Command time. The time between sending the address and sending a command to the NAND Flash memory device. |

58.2.27 Generic command sequence timing configuration register 2 (TIME_GEN_SEQ_2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

[RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N]

Some waveform configuration parameters for the Generic Sequence are defined in the TIME_GEN_SEQ_2 register. The time delay generated by the controller equals the minimum value written into the register, increased by 1. For more detail see section 58.3.2, Generic Command Sequence. All the timings are generated using the S3D1 ϕ clock signal.

[RZ/G2E]

Some waveform configuration parameters for the Generic Sequence are defined in the TIME_GEN_SEQ_2 register. The time delay generated by the controller equals the minimum value written into the register, increased by 1. For more detail see section 58.3.2, Generic Command Sequence. All the timings are generated using the S3D1C ϕ clock signal.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|-------------|-----|-----|-----|-----|-----|----|-------------|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | t0_d11[5:0] | | | | | — | — | t0_d10[5:0] | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | t0_d9[5:0] | | | | | — | — | t0_d8[5:0] | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 30 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 29 to 24 | t0_d11[5:0] | B'00_0000 | R/W | Data to Delay time. The time between data transferring and waiting until the memory is ready. |
| 23 to 22 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 21 to 16 | t0_d10[5:0] | B'00_0000 | R/W | Data to Command time. The time between data transferring and sending a command to the NAND Flash memory device. |
| 15 to 14 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 13 to 8 | t0_d9[5:0] | B'00_0000 | R/W | Delay to Command time. The time between waiting until the memory is ready and sending a command to the NAND Flash memory device. |
| 7 to 6 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|--|
| 5 to 0 | t0_d8[5:0] | B'00_0000 | R/W | Delay to Data time. The time between waiting until the memory is ready and data transferring. |

58.2.28 Generic command sequence timing configuration register 3 (TIME_GEN_SEQ_3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

[RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N]

Some waveform configuration parameters for the Generic Sequence are defined in the TIME_GEN_SEQ_3 register. For more details see section 58.3.2, Generic Command Sequence. All the timings are generated using the S3D1 ϕ clock signal.

[RZ/G2E]

Some waveform configuration parameters for the Generic Sequence are defined in the TIME_GEN_SEQ_3 register. For more details see section 58.3.2, Generic Command Sequence. All the timings are generated using the S3D1C ϕ clock signal.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-------------|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | t0_d12[5:0] | | | | | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|--|
| 31 to 6 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 5 to 0 | t0_d12[5:0] | B'00_0000 | R/W | Data to sequence end time. The time between the data transferring phase and sequence end. The sequence is ended when any other sequence phases after the data transfer phase are not enabled. |

58.2.29 Timing configuration register (TIMINGS_ASYNC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

[RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N]

Two waveform configuration parameters are defined in the TIMINGS_ASYNC register. The time delay generated by the controller equals the minimum value written into the register, increased by 1. All the timings are generated using the S3D1φ clock signal.

[RZ/G2E]

Two waveform configuration parameters are defined in the TIMINGS_ASYNC register. The time delay generated by the controller equals the minimum value written into the register, increased by 1. All the timings are generated using the S3D1Cφ clock signal. Figure 58.3 shows how timings parameters are mapped to the NAND flash interface. The upper part of figure shows read transfer, lower part shows write transfer.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-----------|-----|-----|-----------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | TRWH[3:0] | | | TRWP[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 7 to 4 | TRWH[3:0] | H'0 | R/W | NFRE# or NFWE# high hold time. |
| 3 to 0 | TRWP[3:0] | H'0 | R/W | NFRE# or NFWE# pulse width. |

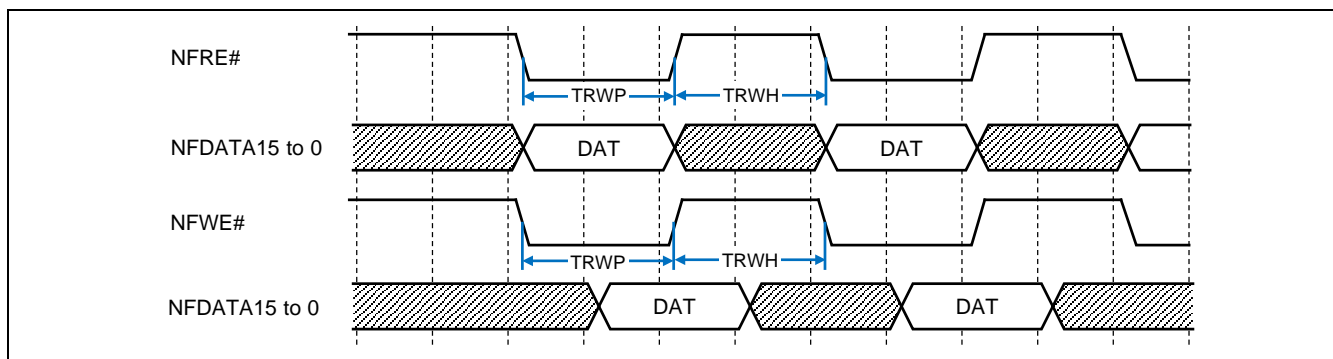
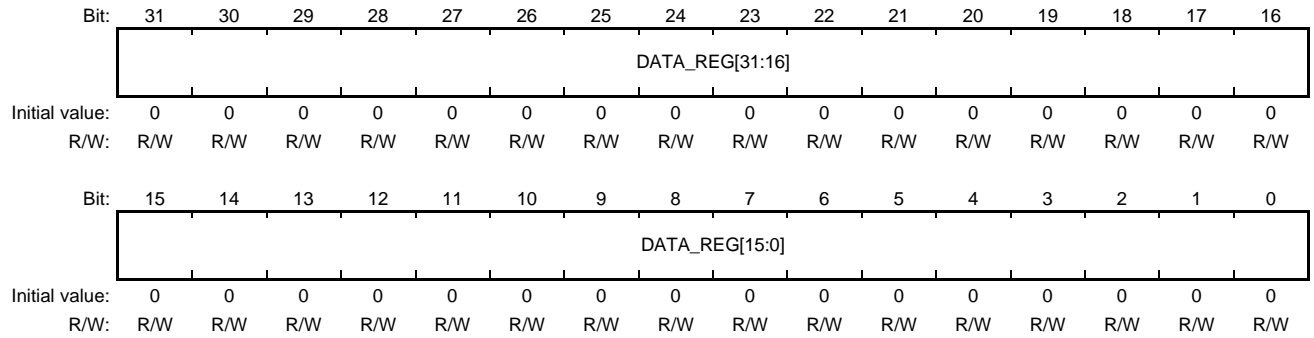


Figure 58.3 Asynchronous timings

58.2.30 Data register (DATA_REG)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The DATA register is used for storage of the data that is read in the registered mode. The registered mode is allowed in the read direction only.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|-------------|
| 31 to 0 | DATA_REG [31:0] | H'0000_0000 | R/W | FIFO DATA |

58.2.31 Data register size selection register (DATA_REG_SIZE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The DATA_REG_SIZE register allows the selection of data size in the registered work mode. The data size in the registered mode is limited to four bytes.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|--------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | DATA_REG_SIZE[1:0] |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------|---------------|-----|--|
| 31 to 2 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 1 to 0 | DATA_REG_SIZE[1:0] | B'00 | R/W | DATA_REG_SIZE register. Allows selection of the number of valid bytes in the DATA register: B'00: single byte valid B'01: two lower bytes valid B'10: three lower bytes valid B'11: all four bytes valid |

58.2.32 FIFO module control register (FIFO_INIT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The FIFO_INIT register contains the FIFO_INIT bit.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | FIFO_INIT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 0 | FIFO_INIT | B'0 | W | FIFO init bit The setting of this bit causes the flushing of FIFO. It is not necessary to set this bit before sending each command to the NAND Flash memory device. This feature is reserved only for a situation where previous FIFO content must be purged before a new operation. |

58.2.33 FIFO module status register (FIFO_STATE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The FIFO_STATE register contains the data buffer status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------------|-----------|-------------|-------------|---------|----------|-----------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | DF_W_EMPTY | DF_R_FULL | CF_AC_CPT_W | CF_AC_CPT_R | CF_FULL | CF_EMPTY | DF_W_FULL | DF_R_EMPTY |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 7 | DF_W_EMPTY | B'1 | R | FIFO empty state bit. This bit indicates that there is no data in the FIFO available. This flag is valid for the write direction. |
| 6 | DF_R_FULL | B'0 | R | FIFO full state bit. This bit indicates that there is no free space for the data in FIFO available. This flag is valid for the read direction. |
| 5 | CF_ACCPT_W | B'0 | R | Command FIFO accept flag – write direction. If this flag is set then next write access will finish with any additional delay. |
| 4 | CF_ACCPT_R | B'1 | R | Command FIFO accept flag – read direction. |
| 3 | CF_FULL | B'0 | R | Command FIFO full flag. It can't be used to check if next transfer will be accepted. |
| 2 | CF_EMPTY | B'1 | R | Command FIFO empty flag. It can't be used to check if next transfer will be accepted. |
| 1 | DF_W_FULL | B'0 | R | FIFO full state bit. This bit indicates that there is no free space for the data in FIFO available. This flag is valid for the write direction. |
| 0 | DF_R_EMPTY | B'1 | R | FIFO empty state bit. This bit indicates that there is no data in the FIFO available This flag is valid for the read direction. |

58.2.34 MLUN register (MLUN)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The MLUN register contains the LUN address offset bits and number of available LUNs.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|--------------|-----|----|----|----|----|----|---------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | LUN_SEL[1:0] | — | — | — | — | — | — | MLUN_IDX[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 to 10 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 9,8 | LUN_SEL [1:0] | B'00 | R/W | LUN number. B'00: two LUN-es B'01: four LUN-es. B'10: eight LUN-es. |
| 7 to 3 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 2 to 0 | MLUN_IDX [2:0] | B'000 | R/W | LUN address offset. The content of this field is used as the bit offset value from the bit zero of the last address byte. This fields point to the bit index in the address that identify active LUN. |

58.2.35 CMD ID initial value register (CMD_MARK)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The CMD MARK register allows to write initial value to the command marking generator in the register Slave I/F Unit (SIU).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | CMD_ID[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | W | W | W | W | W | W | W | W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 7 to 0 | CMD_ID[7:0] | H'00 | W | CMD ID initial value. |

58.2.36 Memory device control register (MEM_CTRL)

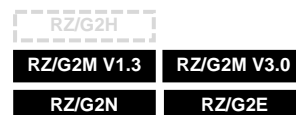
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | √ | √ | √ |

The MEM_CTRL register stores the WP line control.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|---------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | MEMO_WP | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 9 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |
| 8 | MEMO_WP | B'0 | R/W | The WP line control bit. |
| 7 to 0 | — | All 0 | R | Reserved bits These bits are always read as 0. The write value should always be 0. Operation is not guaranteed if a value other than 0 is written to these bits. |

58.3 Operation



58.3.1 Command Generation

The NAND Flash devices are constantly extended and evaluated to the larger capacities and higher data

Through put. Frequently during this process, new commands appear in the next generation of devices.

To allow use of the NAND Flash controller with future generations of devices, the parameterizable command sequences have been added.

The parameterization allows for a defining set of parameters for each supported command sequence.

(1) Instruction Encoding

The controller instruction field is constant and has 32 bits. The instruction field contains the command sequence code and optional parameters. Those parameters are:

- The command codes present in the instruction sequence.
- The flag used to select data destination; possible options are data register and FIFO module.
- The flag used to select data source/sink for the command sequence. The possible choices are the registers or the DMA unit.
- The command sequence code.

If the given command sequence does not use all parameter fields, unused fields are ignored. The instruction encoding scheme is presented in the table below:

Table 58.6 Instruction Encoding

| Field Name | Bits | Description |
|-------------|---------|---|
| CMD_2 | [31:24] | Code of the third command in a sequence. |
| CMD_1/CMD3* | [23:16] | Code of the second or fourth command in a sequence. |
| CMD_0 | [15:8] | Code of the first command in a sequence. |
| DATA_SEL | [7] | Data register / FIFO select flag: 0 – the FIFO selected 1 – the DATA register selected |
| INPUT_SEL | [6] | Input module select flag: 0 – select the register as input 1 – select the DMA module as input |
| CMD_SEQ | [5:0] | Command code. |

Note: Depending on the selected command sequence, this field will store the CMD1 or CMD3 code. Both commands are never used in a single sequence.

(2) Command Sequence Encoding

The NAND Flash devices use the same set of signals independently of the memory capacity. This allows upgrading of obsolete devices with newer ones without PCB redesign. The NAND Flash devices use common IO bus to transfer commands, addresses and data. The predefined set of command sequences is used for the read and write operations on those devices. The set of supported command sequences is not constant for all NAND Flash device producers and evolves as the devices are more capable.

The NAND Flash controller must be able to support the new NAND Flash device features as they appear with a minimum effort from the designer side. This objective can be achieved in many cases because most of the new instructions use the previously defined sequence of commands and addresses, along with new command codes, data page size, data spare area size, etc.

The NAND Flash controller defines the set of commands, addresses and data sequences that allows implementation of all present and many future instructions. The following description of those sequences is adequate to define most of the future NAND Flash device instructions.

Table 58.7 contains the command sequence encoding details. Each sequence is encoded according to the fields defined in the GEN_SEQ_CTRL register.

For more details, section 58.2.3, GENERIC_SEQ register (GEN_SEQ_CTRL)

Table 58.7 Command Sequence Encoding

| Sequence symbol | Sequence encoding | CMD0 | CMD1 | CMD2 | CMD3 | COL_A0*2 | COL_A1 | ROW_A0 | ROW_A1 | DATA_EN | DELAY_EN | IMD_SEQ*3 |
|-----------------|-------------------|------|------|------|------|----------|--------|--------|--------|---------|----------|-----------|
| SEQ_0 | B'00_0000 | √ | — | — | — | — | — | — | — | — | DELAY_1 | — |
| SEQ_1 | B'10_0001 | √ | — | — | — | 1 | — | — | — | √ | — | — |
| SEQ_2 | B'10_0010 | √ | — | — | — | 1 | — | — | — | √ | DELAY_0 | — |
| SEQ_3 | B'00_0011 | √ | — | — | — | 1 | — | — | — | √ | DELAY_1 | — |
| SEQ_4 | B'10_0100 | √ | — | — | — | — | — | — | — | √ | — | √ |
| SEQ_5 | B'10_0101 | √ | — | — | — | — | — | 3 | — | √ | — | √ |
| SEQ_6 | B'10_0110 | √ | — | √ | — | 2(1) | — | — | — | √ | — | — |
| SEQ_7 | B'10_0111 | √ | — | √ | — | 2(1) | — | 3 | — | √ | DELAY_0 | — |
| SEQ_8 | B'00_1000 | √ | — | — | — | 2(1) | — | — | — | √ | — | — |
| SEQ_9 | B'10_1001 | √ | √ | — | — | 2(1) | — | 3 | — | — | DELAY_1 | — |
| SEQ_10 | B'10_1010 | √ | — | √ | — | 2(1) | — | 3 | — | √ | DELAY_0 | — |
| SEQ_11 | B'10_1011 | √ | — | — | — | — | — | — | — | √ | DELAY_0 | — |
| SEQ_12 | B'00_1100 | √ | √ | — | — | 2(1) | — | 3 | — | √ | DELAY_1 | — |
| SEQ_13 | B'00_1101 | √ | — | — | — | 2(1) | — | 3 | — | √ | DELAY_1 | — |
| SEQ_14 | B'00_1110 | √ | √ | — | — | — | — | 3 | — | — | DELAY_1 | — |
| SEQ_15 | B'10_1111 | √ | — | √ | √ | 2 | 2 | 3 | 3 | √ | DELAY_0 | — |
| SEQ_17 | B'11_0001 | √ | — | — | — | 2(1) | — | 3 | — | √ | DELAY_1 | — |
| SEQ_18 | B'11_0010 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 |
| SEQ_19 | B'01_0011 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 | *1 |
| SEQ_20 | B'11_0100 | √ | — | — | — | — | — | 3 | — | — | DELAY_1 | — |
| SEQ_21 | B'01_0101 | √ | — | — | — | 1 | — | — | — | — | — | — |
| SEQ_22 | B'11_0110 | √ | — | √ | — | 2(1) | 2 | — | 3 | √ | DELAY_0 | — |
| SEQ_23 | B'01_0111 | √ | √ | — | — | — | — | 3 | — | √ | DELAY_1 | — |
| SEQ_24 | B'01_1000 | √ | — | √ | √ | — | — | 3 | 3 | — | DELAY_0 | — |
| SEQ_25 | B'11_1001 | √ | — | √ | √ | 2(1) | 2 | 3 | — | √ | — | — |

Note 1. SEQ_18 and SEQ_19 are the parameterized Generic Sequences. The GENE_SEQ_CTRL register defines which parts of sequences are executed.

Note 2. The value given in the brackets relates to the small block mode. In this mode, the controller sends only a single byte as the column address.

Note 3. IMD_SEQ - The command will be sent immediately.

Note: Gray rows – read from NAND Flash memory; White rows – write to NAND Flash memory; Blue rows – Non-directional commands

(a) Sequence SEQ_0

This non-directional sequence is composed from only one command. After the command is written to the NAND Flash device, the controller waits until the device goes into the busy state and drives the RnB line low, or sends the READ STATUS command. When delay time (t_{WB}) passes or the device is ready, the sequence ends. The figure below shows the sequence:

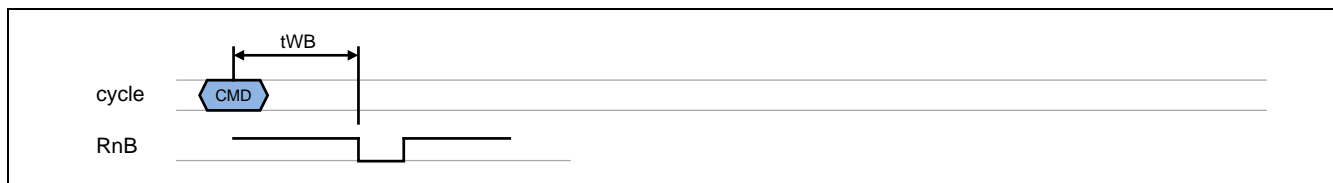


Figure 58.4 SEQ_0 Sequence

(b) Sequence SEQ_1

This is a read-sequence that is composed from a single command cycle, single address cycle and the single data cycle with a programmable number of read sequences. After the address sequence is finished, the controller measures the standard delay of first data read after the last write (t_{WHR}). Next, the read data words are written to the FIFO module. The input module is selected by the INPUT_SEL field of the COMMAND register, the source for the address is placed in the ADDR0_COL register, and the command code is stored in the CMD_0 field. The figure below shows the sequence execution:

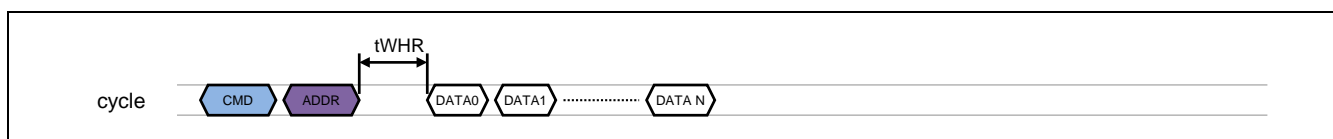


Figure 58.5 SEQ_1 Sequence

(c) Sequence SEQ_2

This is a read-sequence and it is similar to the SEQ_1 sequence except that after the address cycle the controller expects that device goes to the busy state. The controller sequentially checks state of the RnB line or send READ STATUS command to obtain the NAND Flash device status. The figure below shows the sequence execution:

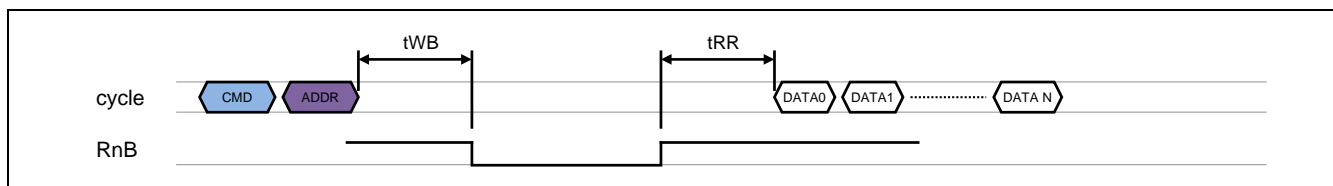


Figure 58.6 SEQ_2 Sequence

(d) Sequence SEQ_3

This is a write-sequence that is composed from a single command cycle, single address cycle and single data cycle with a programmable number of write sequences. After the address sequence is finished, the controller measures the standard delay of first data write after the last address cycle (t_{ADL}). The written words are read from the FIFO module. The controller sequentially checks the state of the RnB line or sends the READ STATUS command to obtain the NAND Flash device status. The input module is selected by the INPUT_SEL field of the COMMAND register; the source for the address is placed in the ADDR0_COL register, and the command code is stored in the CMD_0 field. The figure below shows the sequence execution:

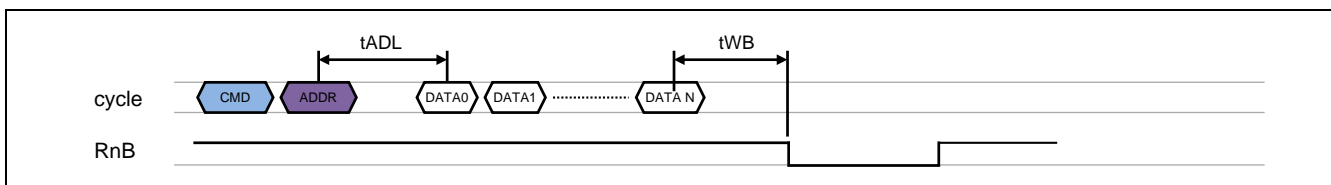


Figure 58.7 SEQ_3 Sequence

(e) Sequence SEQ_4

This is a special read-sequence that is used to implement the read status command sequences. The command is sent immediately. The sequence is composed of a single command cycle and a single data cycle. Between those cycles, the delay is counted (t_{WHR}). The command code is read from the CMD_0 field.

When the DATA register is selected in the COMMAND register, the data is stored in the DATA register. The user defines the number of data in the DATA_REG_SIZE register. The registered mode is allowed only for the read direction.

When the FIFO register is selected in the COMMAND register, the data is stored in the FIFO. Because user has to change DATA_SIZE register, the command will be sent when all memories are ready and the Controller is in the idle state. The figure below shows the sequence execution:

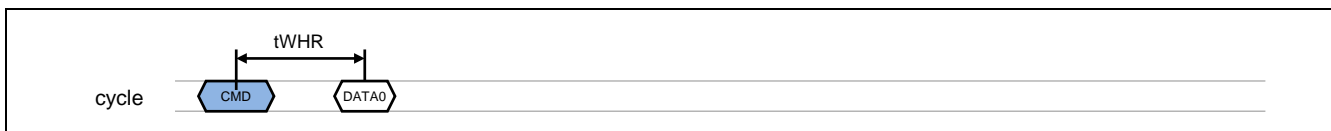


Figure 58.8 SEQ_4 Sequence

(f) Sequence SEQ_5

This is a read-sequence and it is similar to the SEQ_4 sequence. The command is sent immediately. The only difference is that after the command cycle, an additional address cycle is performed. The ADDR0_ROW register is used in this sequence. The figure below shows the sequence execution:

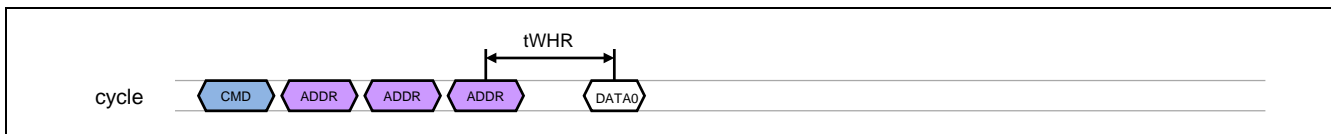


Figure 58.9 SEQ_5 Sequence

(g) Sequence SEQ_6

This is a read-sequence. The sequence of command cycle, address cycle, command cycle is executed. After that, the delay from the change column to the next operation (tCCS) is measured. Finally, the read data cycle is executed. The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_2 instruction field; the ADDR0_COL register is used in this sequence; the input module is selected by the INPUT_SEL field. The figure below shows the sequence execution:

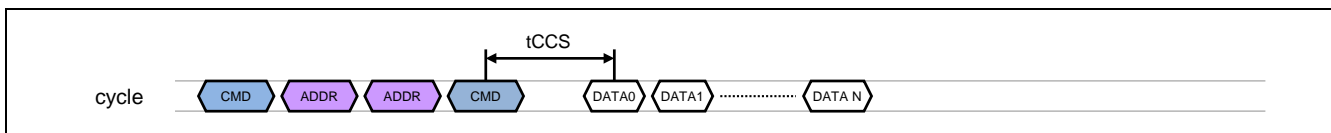


Figure 58.10 SEQ_6 Sequence

(h) Sequence SEQ_7

This read-sequence is similar to the SEQ_6 sequence, differing only because the address cycle in this sequence is composed of five bytes (ADDR0_COL and ADDR0_ROW) instead of three bytes. All else is the same as in the SEQ_6 sequence. The figure below shows the sequence execution:

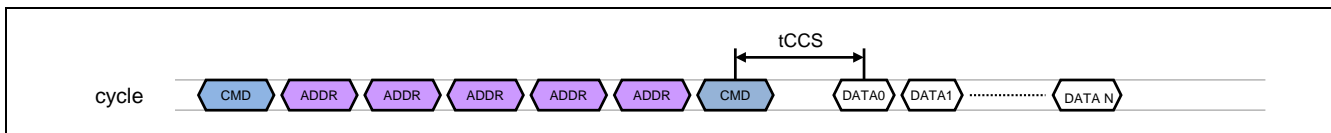


Figure 58.11 SEQ_7 Sequence

(i) Sequence SEQ_8

This is a write-sequence. First, the sequence of command cycle and two bytes address cycle is executed. Next, the delay after the column address changes (t_{CCS}) is measured. Finally, the single data cycle with programmable number of write sequences is executed. The first command code is encoded in the CMD_0 instruction field; the ADDR0_COL register is used in this sequence; the input module is selected by the INPUT_SEL field. The figure below shows the sequence execution:

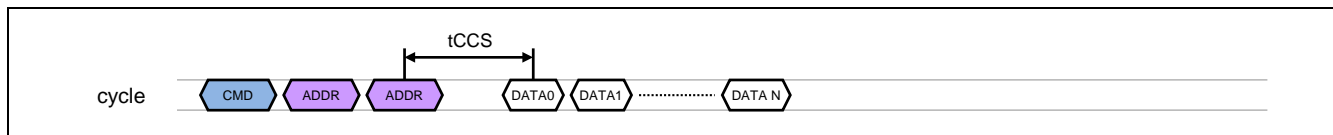


Figure 58.12 SEQ_8 Sequence

(j) Sequence SEQ_9

This is a non-directional sequence. The first step is to execute the five bytes address command cycle. The controller sequentially checks the state of the RnB line or sends the READ STATUS command to obtain the NAND Flash device status. The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_1 instruction field; the ADDR0_COL and ADDR0_ROW registers are used in this sequence. The figure below shows the sequence execution:

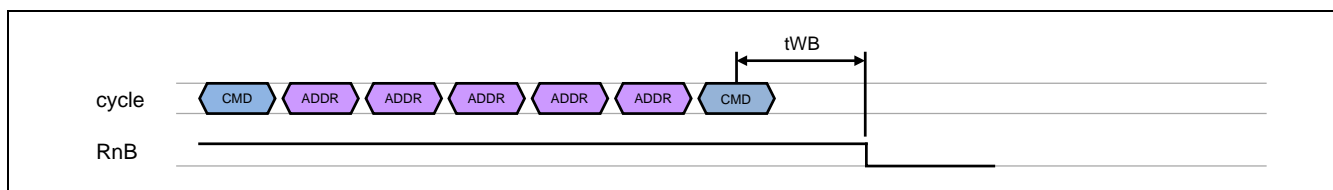


Figure 58.13 SEQ_9 Sequence

(k) Sequence SEQ_10

This is a non-directional sequence. The first step is to execute the five bytes address command cycle. The controller sequentially checks the state of the RnB line or sends the READ STATUS command to obtain the NAND Flash device status. Finally data block is read. The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_2 instruction field; the ADDR0_COL and ADDR0_ROW registers are used in this sequence. The figure below shows the sequence execution:

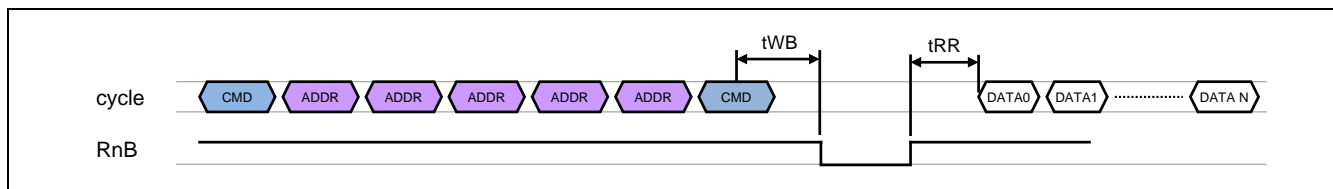


Figure 58.14 SEQ_10 Sequence

(l) Sequence SEQ_11

This is the read-sequence. The first step is to execute the command cycle. Next, the device goes to the busy state. The controller sequentially checks the state of the RnB line or sends the READ STATUS command to obtain the NAND Flash device status. As soon as the device reaches the ready state, the write data cycle with configurable read sequences is executed. The command code is encoded in the CMD_0 instruction field; the input module is selected by the INPUT_SEL field. The number of transferred bytes are configured using the DATA_SIZE register. The figure below shows the sequence execution:

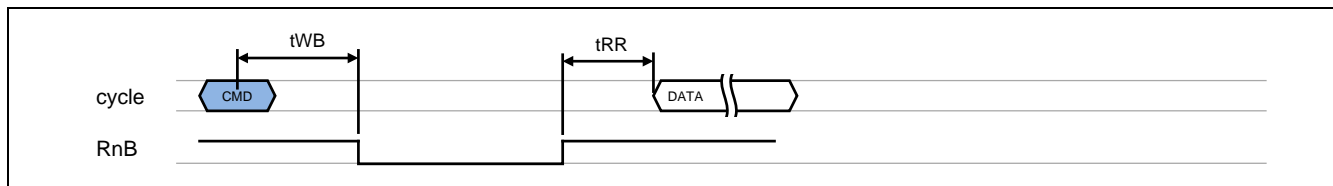


Figure 58.15 SEQ_11 Sequence

(m) Sequence SEQ_12

This is a write-sequence. The SEQ_12 sequence is a series of command cycle, address cycle, and data cycle with a configurable number of write operations and another command cycle. Between the last address cycle and first data cycle, a delay is measured (tADL) and, after the second command cycle, another delay is measured (tWB). The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_1 instruction field; the ADDR0_COL and ADDR_ROW registers are used in this sequence; the input module is selected by the INPUT_SEL field. The figure below shows the sequence execution:

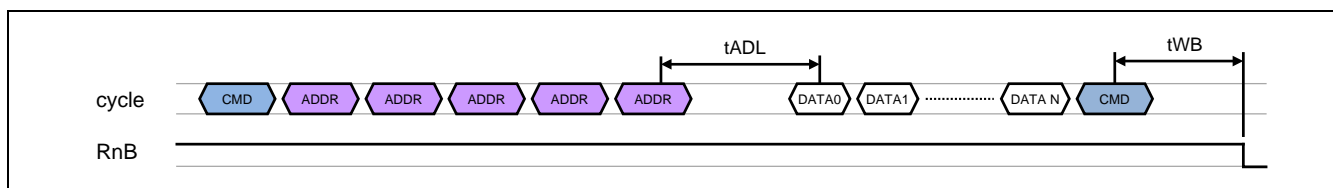


Figure 58.16 SEQ_12 Sequence

(n) Sequence SEQ_13

This is a write-sequence. The SEQ_13 sequence is a series of command cycle and address cycles, data cycle with a configurable number of write operations. Between the last address cycle and first data cycle, a delay is measured (tADL). The command code is encoded in the CMD_0 instruction field; the ADDR0_COL and ADDR0_ROW registers are used in this sequence; the input module is selected by the INPUT_SEL field. The figure below shows the sequence execution:

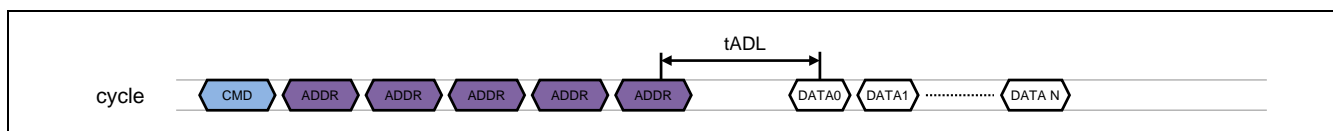


Figure 58.17 SEQ_13 Sequence

(o) Sequence SEQ_14

This is a non-directional sequence. First, the series of command cycle, address cycle, command cycle is executed. The controller sequentially checks the state of the RnB line or sends the READ STATUS command to obtain the NAND Flash device status. The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_1 instruction field; the ADDR0_ROW and ADDR0_COL registers are used in this sequence. The figure below shows the sequence execution:

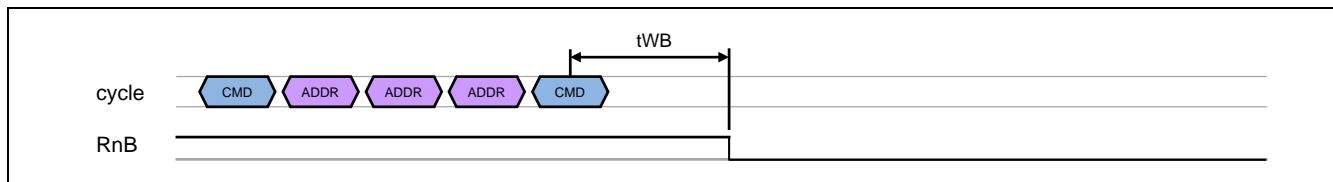


Figure 58.18 SEQ_14 Sequence

(p) Sequence SEQ_15

This is a read-sequence. First, the series of command cycle, address cycle, second command cycle, second address cycle, third command cycle is executed. The controller sequentially checks the state of the RnB line or sends the READ STATUS command to obtain the NAND Flash device status. After the NAND Flash device returns to the ready state, the data sequence, with a configurable number of read operations, is executed. The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_1 instruction field; the third command code is encoded in the CMD_2 instruction field. In this sequence, both address registers are used. The ADDR0_ROW and ADDR0_COL register content is sent after the first command in the sequence, the ADDR1_ROW and ADDR1_COL register content is sent after the second command in the sequence. The figure below shows the sequence execution:

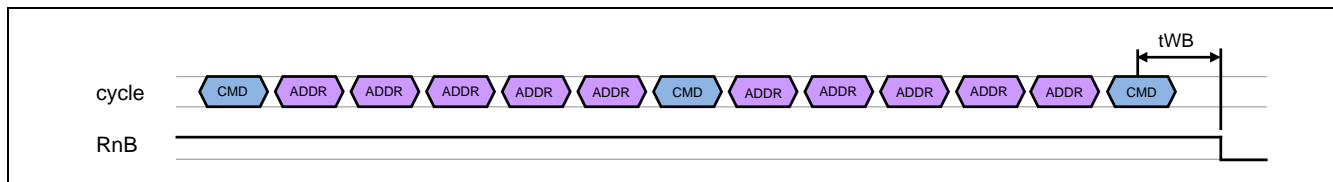


Figure 58.19 SEQ_15 Sequence

(q) Sequence SEQ_17

This is a read-sequence. This sequence is similar to the SEQ_10 sequence, except that the second command cycle is omitted. This sequence is implemented to use small block memories. The controller sends only four bytes of the address when the small block mode is enabled. The figure below shows the sequence execution:

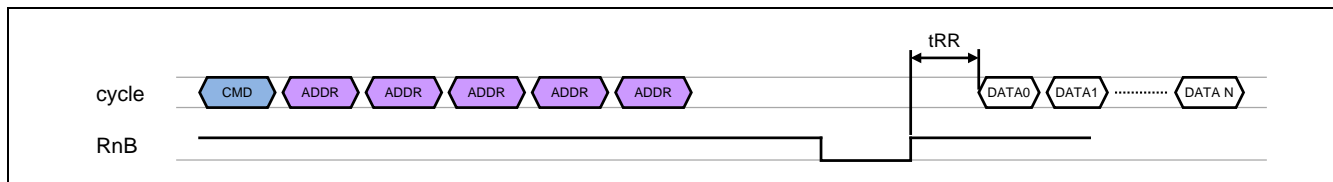


Figure 58.20 SEQ_17 Sequence

(r) Sequence SEQ_18

This generic read sequence is described in detail in section 58.3.2, Generic Command Sequence.

(s) Sequence SEQ_19

This generic write sequence is described in detail in section 58.3.2, Generic Command Sequence.

(t) Sequence SEQ_20

This non-directional sequence is composed from one command and three addresses bytes. After the command and addresses are written to the NAND Flash device, the controller waits until the device goes into the busy state and drives the RnB line low, or sends the READ STATUS command. When delay time (t_{WB}) passes or the device is ready, the sequence ends. The figure below shows the sequence execution:

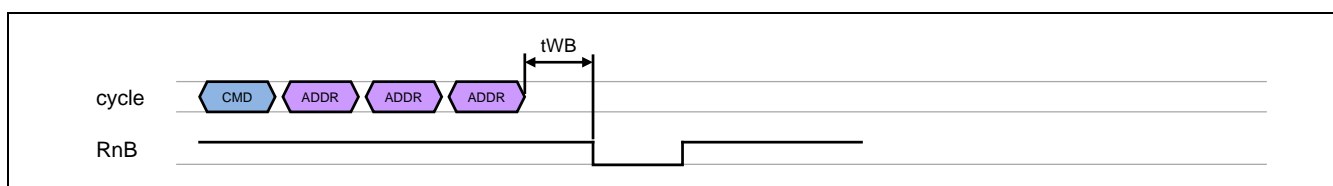


Figure 58.21 SEQ_20 Sequence

(u) Sequence SEQ_21

This non-directional sequence is composed from one command and one addresses byte. After the command and address are written to the NAND Flash device, the sequence ends. The figure below shows the sequence execution:

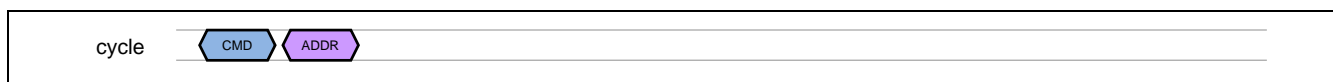


Figure 58.22 SEQ_21 Sequence

(v) Sequence SEQ_22

This is a read-sequence. The first step is to execute the five bytes address command cycle. The controller sequentially checks the state of the RnB line or sends the READ STATUS command to obtain the NAND Flash device status. The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_1 instruction field; the ADDR0_COL, ADDR1_COL and ADDR1_ROW registers are used in this sequence. The figure below shows the sequence execution:

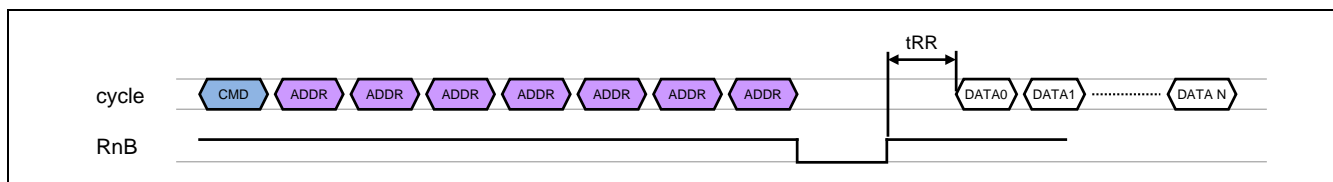


Figure 58.23 SEQ_22 Sequence

(w) Sequence SEQ_23

This is a write-sequence. The SEQ_23 sequence is a series of command cycle, address cycle, and data cycle with a configurable number of write operations and another command cycle. Between the last address cycle and first data cycle, a delay is measured (t_{ADL}) and, after the second command cycle, another delay is measured (t_{WB}). The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_1 instruction field; the ADDR0_ROW registers is used in this sequence; the input module is selected by the INPUT_SEL field. The figure below shows the sequence execution:

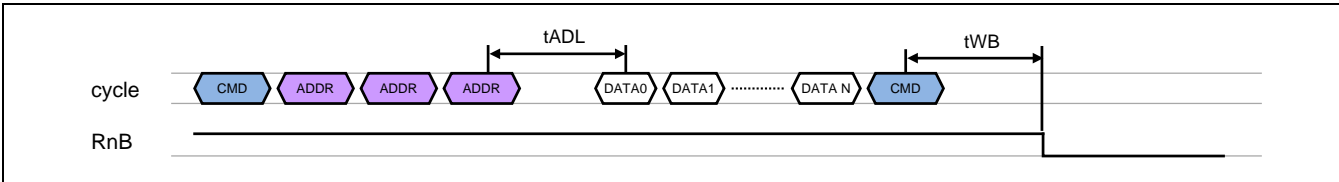


Figure 58.24 SEQ_23Sequence

(x) Sequence SEQ_24

This is a write-sequence. It is composed from the three commands cycles and two addresses cycles. Both addresses cycle contain the row address part. After the last command cycle the t_{WB} delay is measured. The figure below shows the sequence execution:

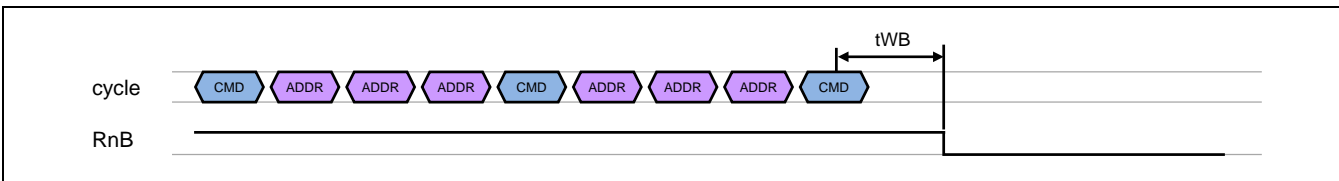


Figure 58.25 SEQ_24 Sequence

(y) Sequence SEQ_25

This is a read-sequence. It is composed from the three commands cycles and two addresses cycles. the first addresses cycle contain the column and row address part. The second address cycle contain only the column address part. After the last command cycle the t_{WHR} delay is measured. The figure below shows the sequence execution:

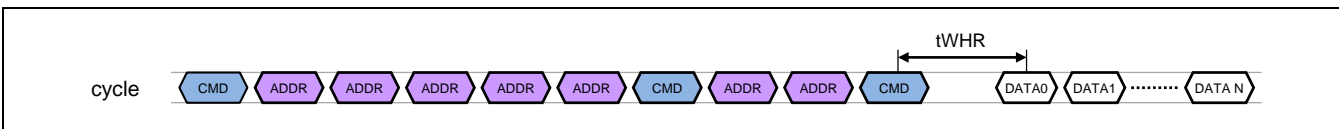


Figure 58.26 SEQ_25 Sequence

(z) Sequence SEQ_26

This is a write-sequence. The SEQ_26 sequence is a series of command cycle, address cycle, and data cycle with a configurable number of write operations and another command cycle. Between the last address cycle and first data cycle, a delay is measured (t_{ADL}). The first command code is encoded in the CMD_0 instruction field; the second command code is encoded in the CMD_1 instruction field; the ADDR0_ROW registers is used in this sequence; the input module is selected by the INPUT_SEL field. The figure below shows the sequence execution:

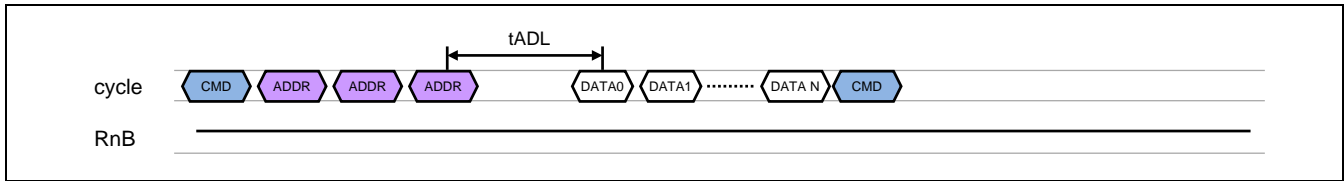


Figure 58.27 SEQ_26 Sequence

58.3.2 Generic Command Sequence

There will be cases where the set of predefined sequences above will not be sufficient to handle a new command sequence. If that occurs, the generic command sequence feature of the NAND flash controller can be used. There will be cases where the set of predefined sequences above will not be sufficient to handle a new command sequence. If that occurs, the generic command sequence feature of the NAND flash controller can be used.

This sequence is designed to mimic almost every available command supported by the NAND Flash devices; however, additional effort required to trigger such commands.

Generic command sequence is executed in the following steps:

CMD0 – The first command in the sequence. The value of this command is stored in the CMD_0 field of the COMMAND register, described in 58.2.1 Controller commands register (COMMAND)

ADDR0 – The first address sequence. This is enabled if the COL_A0 and ROW_A0 fields in the GEN_SEQ_CTRL register described in 58.2.3 GENERIC_SEQ register (GEN_SEQ_CTRL) have values other than zero. In this phase, the address is sent to the NAND Flash device and is read from the ADDR0_COL and ADDR0_ROW registers. The number of the bytes in the address cycle is configured by the COL_A0 and ROW_A0 fields of the GEN_SEQ_CTRL register.

CMD1 – The fourth command in the sequence. The value of this command is stored in the CMD_1 field of the COMMAND register, described in 58.2.1 Controller commands register (COMMAND). This is enabled by the CMD1_EN field in the GEN_SEQ_CTRL register, described in 58.2.3 GENERIC_SEQ register (GEN_SEQ_CTRL).

ADDR1 – The second address in the sequence. This is enabled if the COL_A1 and ROW_A1 fields of the GEN_SEQ_CTRL register described in 58.2.3 GENERIC_SEQ register (GEN_SEQ_CTRL) have a value other than zero. In this phase, the address is sent to the NAND Flash device from the ADDR1_COL and ADDR1_ROW registers. The number of bytes in the address cycle is configured by the COL_A1 and ROW_A1 fields of the GEN_SEQ_CTRL register.

CMD2 – The third command in the sequence. The value of this command is stored in the CMD_2 field of the COMMAND register described in 58.2.1 Controller commands register (COMMAND). This is enabled by the CMD2_EN field in the GEN_SEQ_CTRL register, described in 58.2.3 GENERIC_SEQ register (GEN_SEQ_CTRL).

DELAY0 – Waiting for the device to return to the ready state and continue the sequence. This is enabled by the DELAY_EN field in the GEN_SEQ_CTRL register, described in 58.2.3 GENERIC_SEQ register (GEN_SEQ_CTRL). Only one delay phase can be present in the generic sequence. For more details, see 58.2.3 GENERIC_SEQ register (GEN_SEQ_CTRL).

DATA – The data phase of the sequence. This is enabled by the DATA_EN field in the GEN_SEQ_CTRL register described in 58.2.3 GENERIC_SEQ register (GEN_SEQ_CTRL). Additionally, the transfer direction must be selected by the sequence number. Sequence number 18 reads data from the NAND Flash memory, sequence number 19 writes data to the NAND Flash memory. The size of the transferred data block is configured by the DATA_SIZE register value.

CMD3 – The second command in the sequence. This is enabled by the CMD3_EN field in the GEN_SEQ_CTRL register described in the 58.2.3 GENERIC_SEQ register (GEN_SEQ_CTRL). The value of this command is stored in the CMD_3 field of the GEN_SEQ_CTRL register.

DELAY1 – Waiting for the device to return to the ready state and finish the sequence. This is enabled by the

DELAY_EN field in the GEN_SEQ_CTRL register described in 58.2.3 GENERIC_SEQ register (GEN_SEQ_CTRL). The controller waits for the device to return to the ready state and finish the sequence. Only one delay phase can be present in generic sequence.

The figure below presents the generic sequence composition:

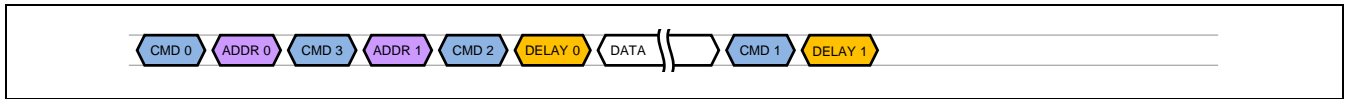


Figure 58.28 Generic Sequence

There are few constraints on the generic sequence usage:

- The DEL0 and DEL1 delay phases cannot both be enabled in a single command sequence
- The TIMINGS _ASYN register must be set even when the generic sequence is used.

It is possible to force an immediate command sequence execution by enabling the IMD_SEQ bit in the GEN_SEQ_CTRL register. In this case, the triggered command will be executed even if the previously sent command for a selected device is not completed. If both the IMD_SEQ and DATA_EN flags are enabled in a single sequence, then the register must be selected as data source/sink. The IMD_SEQ is valid only for the read direction. This feature is intended to implement all state read operations.

After each step of the generic sequence, the programmable time delay is measured. These delays are configured using the TIME_GEN_SEQ[0-3] registers. Refer to the TIME_GEN_SEQ[0-3] registers description for further information and see figures below.

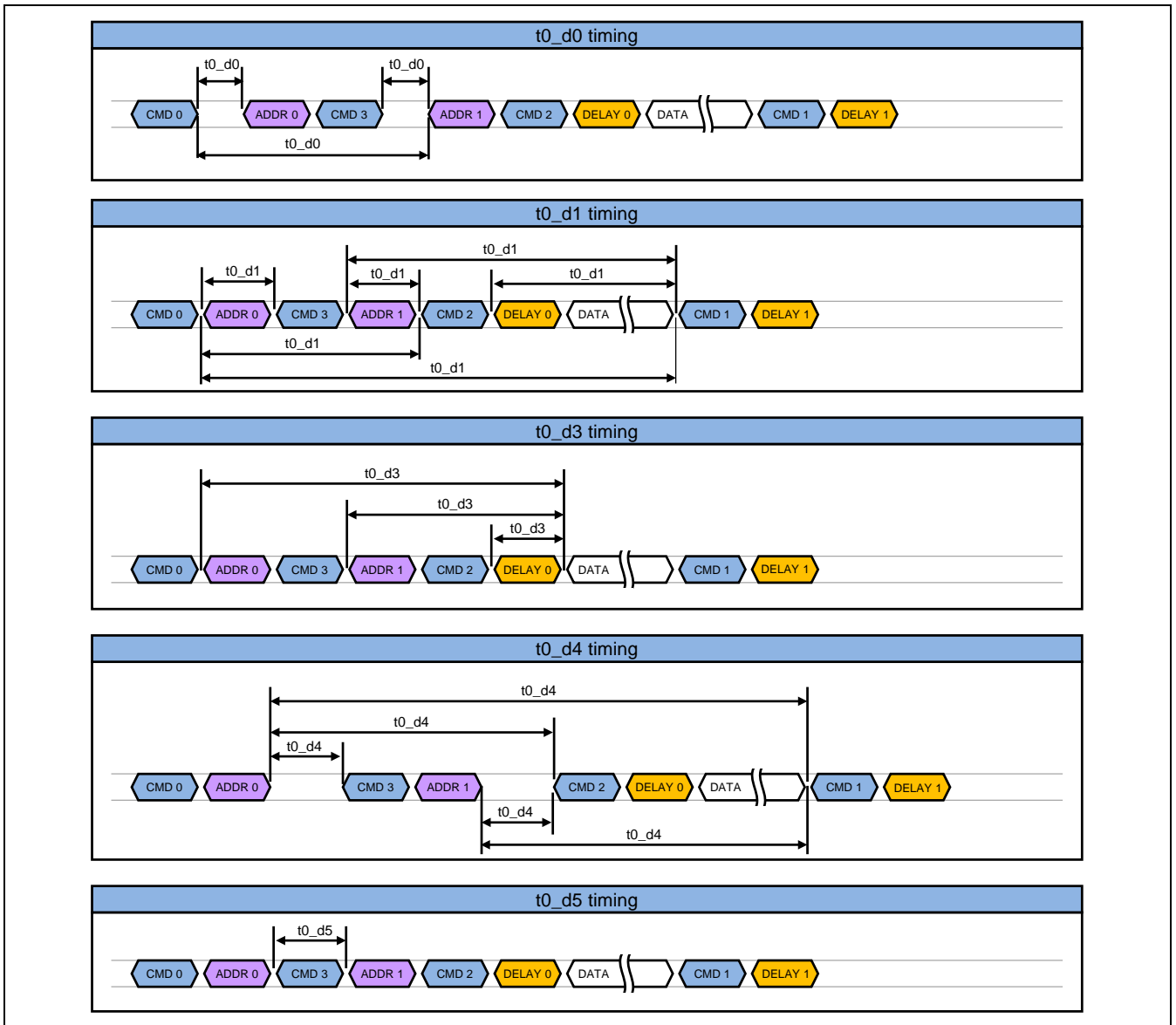


Figure 58.29 Generic Sequence Timing parameters

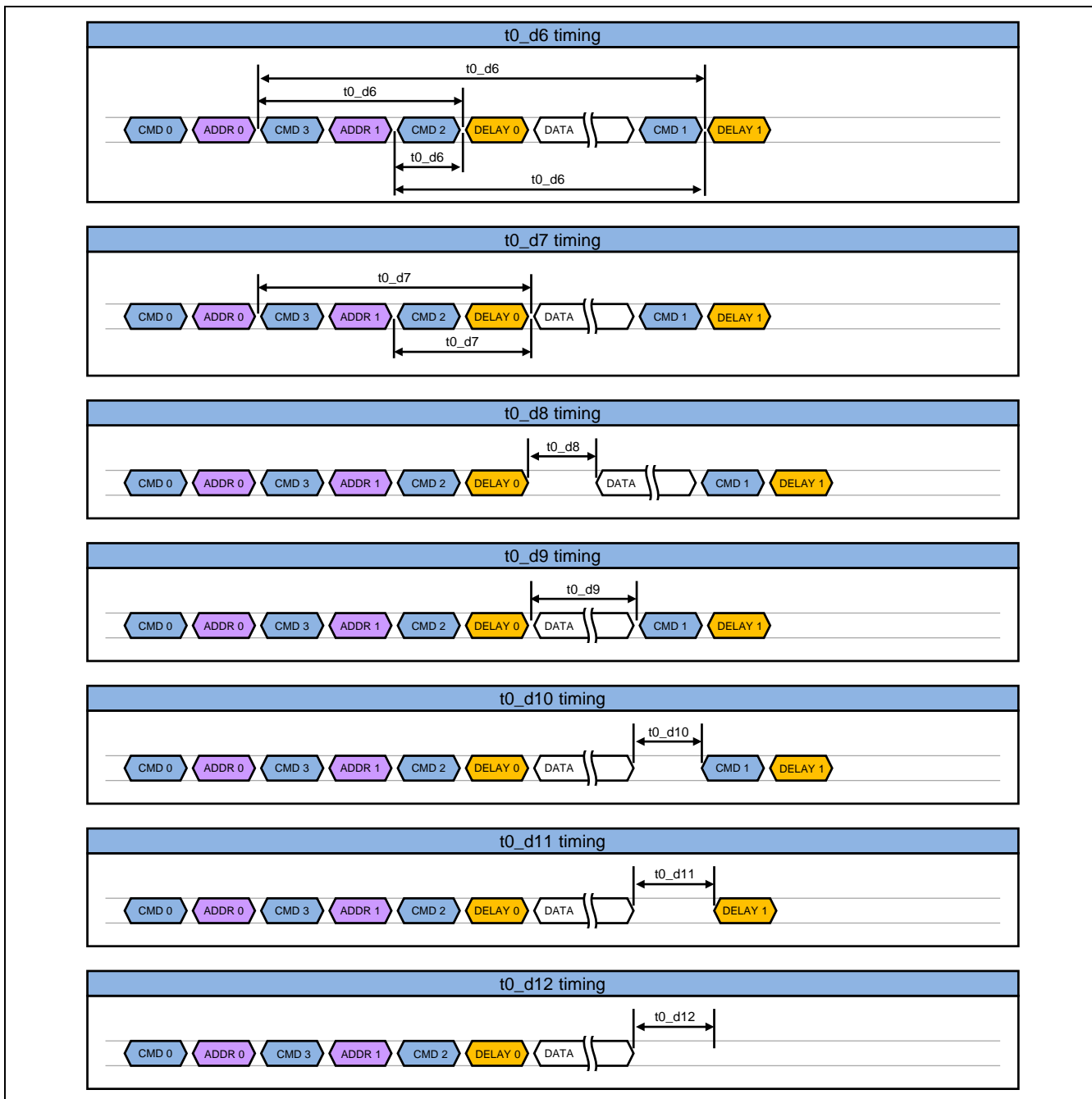


Figure 58.30 Generic Sequence Timing parameters

58.3.3 Instructions

The implementation of the instruction set presented at this point is an example of how to use the instruction encoding scheme presented in section 58.3.1 (1), Instruction Encoding. This chapter and the command sequences presented previously give sufficient information to implement new commands and command sequences for future NAND Flash devices.

(1) Instructions Set

The table below contains the basic instruction defined to implement the all command sequences accessible in the ONFI 1.x standards.

Table 58.8 Instructions set

| Instruction | CMD_0 | CMD_1/ CMD_3 | CMD_2 | CD_SEQ | Send when memory is busy |
|----------------------------------|-------|-----------------|-------|--------|-----------------------------|
| Reset Commands | | | | | |
| RESET | H'FF | — | — | SEQ_0 | No |
| Identification Operations | | | | | |
| READ ID | H'90 | — | — | SEQ_1 | No |
| READ PARAMETER PAGE | H'EC | — | — | SEQ_2 | No |
| READ UNIQUE ID | H'ED | — | — | SEQ_3 | No |
| Configuration Operations | | | | | |
| GET FEATURES | H'EE | — | — | SEQ_2 | No |
| SET FEATURES | H'EF | — | — | SEQ_3 | No |
| Status Operations | | | | | |
| READ STATUS | H'70 | — | — | SEQ_4 | Yes |
| SELECT LUN WITH STATUS | H'78 | — | — | SEQ_5 | Yes |
| LUN STATUS | H'71 | — | — | SEQ_5 | Yes |
| DEVICE STATUS | H'72 | — | — | SEQ_4 | Yes |
| VOLUME_SELECT | H'E1 | — | — | SEQ21 | Yes |
| Column Address Operations | | | | | |
| CHANGE READ COLUMN | H'05 | — | H'E0 | SEQ_6 | No |
| SELECT CACHE REGISTER | H'06 | — | H'E0 | SEQ_7 | No |
| CHANGE WRITE COLUMN | H'85 | — | — | SEQ_8 | No |
| CHANGE ROW ADDRESS | H'85 | H'11 | — | SEQ_12 | No |
| Read Operations | | | | | |
| READ PAGE | H'00 | — | H'30 | SEQ_10 | No |
| READ PAGE CHANGE | H'31 | — | — | SEQ_11 | No |
| READ PAGE CHANGE LAST | H'3F | — | — | SEQ_11 | No |
| READ MULTIPLANE | H'00 | H'32 | — | SEQ_9 | No |
| TWO PLANE PAGE READ | H'00 | H'00 | H'30 | SEQ_15 | No |
| QUEUE PAGE READ | H'07 | — | H'37 | SEQ_22 | No |
| Program Operation | | | | | |
| PROGRAM PAGE | H'80 | H'10 | — | SEQ_12 | No |
| PROGRAM PAGE IMD | H'80 | H'10 | — | SEQ_23 | No |
| PROGRAM PAGE DEL | H'80 | H'13 | — | SEQ_23 | No |
| PROGRAM PAGE 1 | H'80 | — | — | SEQ_13 | No |
| PROGRAM PAGE CACHE | H'80 | H'15 | — | SEQ_12 | No |

| Instruction | CMD_0 | CMD_1/ CMD_3 | CMD_2 | CD_SEQ | Send when memory is busy |
|----------------------------|-------|-----------------|-------|--------|-----------------------------|
| PROGRAM MULTIPLANE | H'80 | H'11 | — | SEQ_12 | No |
| WRITE PAGE | H'10 | — | — | SEQ_0 | No |
| WRITE PAGE CACHE | H'15 | — | — | SEQ_0 | No |
| WRITE MULTIPLANE | H'11 | — | — | SEQ_0 | No |
| Erase Operations | | | | | |
| ERASE BLOCK | H'60 | H'D0 | — | SEQ_14 | No |
| ERASE MULTIPLANE | H'60 | H'D1 | — | SEQ_14 | No |
| Copyback Operations | | | | | |
| COPYBACK READ | H'00 | — | H'35 | SEQ_10 | No |
| COPYBACK PROGRAM | H'85 | H'10 | — | SEQ_9 | No |
| COPYBACK PROGRAM 1 | H'85 | — | — | SEQ_13 | No |
| COPYBACK MULTIPLANE | H'85 | H'11 | — | SEQ_12 | No |
| OTP Operations | | | | | |
| PROGRAM OTP | H'A0 | H'10 | — | SEQ_12 | No |
| DATA PROTECT OTP | H'A5 | H'10 | — | SEQ_9 | No |
| READ PAGE OTP | H'AF | — | H'30 | SEQ_10 | No |

(2) RESET Command**(a) Command Description**

The RESET command is used to put a target into a known condition and to abort command sequences in progress.

(b) Command Encoding

The RESET instruction uses the SEQ_0 commands sequence. The command is encoded, as shown in the table below:

Table 58.9 RESET Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | — | H'FF | — | — | SEQ_0_ID |

(3) READ ID Command**(a) Command Description**

The READ ID command is used to read identifier codes programmed into the target. This command is accepted by the target only when all LUNs on the target are in the IDLE state.

When the command is followed by an address cycle of H'00, the target returns a 5-byte identifier code that includes the manufacturer's ID, device configuration, and part-specific information.

When the READ ID command is followed by an address cycle of H'20, the target returns the 4-byte ONFI identifier code.

(b) Command Encoding

The READ ID instruction uses the SEQ_1 commands sequence. The command is encoded, as shown in the table below:

Table 58.10 READ ID Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | — | H'90 | 0/1 | 0/1 | SEQ_1_ID |

(4) READ PARAMETER PAGE Command**(a) Command Description**

The READ PARAMETER PAGE command is used to read the ONFI parameter page programmed into the target. This command is accepted by the target only when all LUNs on the target are idle.

When the command is followed by an address cycle of H'00, the target goes into busy state. After the read process is completed, the controller enables the data output mode to read the parameter page.

(b) Command Encoding

The READ PARAMETER PAGE instruction uses the SEQ_2 commands sequence. The command is encoded, as shown in the table below:

Table 58.11 READ PARAMETER PAGE Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | — | H'EC | 0/1 | 0/1 | SEQ_2_ID |

(5) READ UNIQUE ID Command**(a) Command Description**

The READ UNIQUE ID instruction is used to read a unique identifier programmed into the target. This command is accepted by the target only when all LUNs on the target are idle.

When the address cycle of H'00 is written to the target, then the target goes into busy state. After the read process is complete, the controller enables the data output mode to read the unique ID. Sixteen copies of the unique ID data are stored in the device. Each copy is 32 bytes. The first 16 bytes are unique data and the second 16 bytes are the complement of the first 16 bytes. The application program will XOR the first 16 bytes with the second 16 bytes. If the result is 16 bytes of H'FF, then that copy of the unique ID data is correct. In the event that there is a non-H'FF result, the application program repeats the XOR operation on a subsequent copy of the unique ID data.

(b) Command Encoding

The READ UNIQUE ID instruction uses the SEQ_2 commands sequence. The command is encoded, as shown in the table below:

Table 58.12 READ UNIQUE ID Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | — | H'ED | 0/1 | 0/1 | SEQ_2_ID |

(6) GET FEATURES Command**(a) Command Description**

The GET FEATURES instruction reads the sub-feature parameters (P1-P4) from the specified feature address. This command is accepted by the target only when all LUNs on the target are idle.

When the H'EE command is followed by a feature address, the target goes into busy state. After the target internal read operation completes, the controller enables the data output mode to read the subfeature parameters.

(b) Command Encoding

The GET FEATURES instruction uses the SEQ_2 commands sequence. The command is encoded, as shown in the table below:

Table 58.13 GET FEATURES Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | — | H'EE | 0/1 | 0/1 | SEQ_2_ID |

(7) SET FEATURES Command**(a) Command Description**

The SET FEATURES instruction writes the sub-feature parameters (P1-P4) to the specified feature address to enable or disable target-specific features. This command is accepted by the target only when all LUNs on the target are idle.

The H'EF command is followed by a valid feature address. The possible address value depends on the features set implemented in the target device. The address cycle is followed by the configurable number of data cycles. Values of the address and data encoding scheme allowed are found in the device vendor documentation.

(b) Command Encoding

The SET FEATURES instruction uses the SEQ_3 commands sequence. The command is encoded, as shown in the table below:

Table 58.14 SET FEATURES Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | — | H'EF | — | 0/1 | SEQ_3_ID |

(8) READ STATUS Command**(a) Command Description**

Each LUN provides its status independently of other LUNs on the same target through its 8-bit status register. Once the READ STATUS instruction is issued, status register output is enabled. The contents of the status register are returned on DQ[7: 0] for each data output request.

The READ STATUS command returns the status of the most recently selected LUN.

(b) Command Encoding

The READ STATUS instruction uses the SEQ_4 commands sequence. The command is encoded, as shown in the table below:

Table 58.15 READ STATUS Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | — | H'70 | 0/1 | — | SEQ_4_ID |

This instruction has special meaning because it can be executed when the target is in the BUSY state.

The FIFO cannot be used to access the read data because it can be occupied by the operation under execution. For the sequence the data field of command must select the DATA_REG register as data destination. The DATA_REG_SIZE must be select single byte.

(9) DEVICE STATUS Command**(a) Command Description**

Each LUN provides its status independently of other LUNs on the same target through its 8-bit status register. Once the DEVICE STATUS instruction is issued, status register output is enabled. The contents of the status register are returned on DQ[7:0] for each data output request.

The DEVICE STATUS command returns the status of the most recently selected LUN.

(b) Command Encoding

The DEVICE STATUS instruction uses the SEQ_4 commands sequence. The command is encoded, as shown in the table below:

Table 58.16 DEVICE STATUS Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | — | H'72 | 0/1 | — | SEQ_4_ID |

This instruction has special meaning because it can be executed when the target is in the BUSY state.

The FIFO cannot be used to access the read data because it can be occupied by the operation under execution. For the sequence the data field of command must select the DATA_REG register as data destination. The DATA_REG_SIZE must be select single byte.

(10) VOLUME SELECT Command**(a) Command Description**

This command is accepted by all initialized devices that share a CE pin. The command may be executed with any volume on the target in any state. When the VOLUME SELECT command is issued, all volumes with unselected volume addresses will be deselected to save power. If the Volume address entered is invalid or does not match any appointed volume address, all volume addresses will be deselected. If the VOLUME SELECT command is not issued after CE high time then all volumes.

(b) Command Encoding

The VOLUME SELECT instruction uses the SEQ_21 commands sequence. The command is encoded, as shown in the table below:

Table 58.17 VOLUME SELECT Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | — | H'E1 | 0/1 | — | SEQ_21_ID |

(11) SELECT LUN WITH STATUS Command**(a) Command Description**

Each LUN provides its status independently of other LUNs on the same target through its 8-bit status register. Once the SELECT LUN WITH STATUS instruction is issued, status register output is enabled. The contents of the status register are returned on DQ[7:0] for each data output request.

The SELECT LUN WITH STATUS command returns the status of the selected LUN.

(b) Command Encoding

The SELECT LUN WITH STATUS instruction uses the SEQ_5 commands sequence. The command is encoded, as shown in the table below:

Table 58.18 SELECT LUN WITH STATUS Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | — | H'78 | 0/1 | — | SEQ_5_ID |

This instruction has special meaning because it can be executed when the target is in the BUSY state.

The FIFO cannot be used to access the read data because it can be occupied by the operation under execution. For the sequence the data field of command must select the DATA_REG register as data destination. The DATA_REG_SIZE must be select single byte.

(12) LUN STATUS Command**(a) Command Description**

Each LUN provides its status independently of other LUNs on the same target through its 8-bit status register. Once the SELECT LUN WITH STATUS instruction is issued, status register output is enabled. The contents of the status register are returned on DQ[7: 0] for each data output request.

The LUN STATUS command returns the status of the selected LUN.revert to their previous state.

(b) Command Encoding

The LUN STATUS instruction uses the SEQ_5 commands sequence. The command is encoded, as shown in the table below:

Table 58.19 LUN STATUS Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | — | H'71 | 0/1 | — | SEQ_5_ID |

This instruction has special meaning because it can be executed when the target is in the BUSY state.

The FIFO cannot be used to access the read data because it can be occupied by the operation under execution. For the sequence the data field of command must select the DATA_REG register as data destination. The DATA_REG_SIZE must be select single byte.

(13) CHANGE READ COLUMN Command**(a) Command Description**

The CHANGE READ COLUMN command changes the column address of the selected cache register and enables data output of the last selected LUN. This command is accepted by the selected LUN when it is ready. Writing H'05 to the target command register, followed by two column address cycles containing the column address, followed by the H'E0 command puts the selected LUN into data output mode. The selected LUN stays in data output mode until another valid command is issued.

(b) Command Encoding

The CHANGE READ COLUMN instruction uses the SEQ_6 commands sequence. The command is encoded, as shown in the table below:

Table 58.20 CHANGE READ COLUMN Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| H'E0 | — | H'05 | 0/1 | — | SEQ_6_ID |

(14) SELECT CACHE REGISTER Command**(a) Command Description**

The SELECT CACHE REGISTER command enables data output on the addressed LUN and cache register at the specified column address. This command is accepted by a LUN when it is ready. Writing the H'06 to the target internal command register, followed by two column address cycles and three row address cycles, followed by H'E0 enables data output mode on the address LUN and cache register at the specified column address.

(b) Command Encoding

The SELECT CACHE REGISTER instruction uses the SEQ_7 commands sequence. The command is encoded, as shown in the table below:

Table 58.21 SELECT CACHE REGISTER Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| H'E0 | — | H'06 | 0/1 | — | SEQ_7_ID |

(15) CHANGE WRITE COLUMN Command**(a) Command Description**

The CHANGE WRITE COLUMN command changes the column address of the selected cache register and enables data input on the last selected LUN. Writing the H'85 to the target internal command register, followed by two column address cycles containing the column address puts the selected LUN into data input mode.

(b) Command Encoding

The CHANGE WRITE COLUMN instruction uses the SEQ_8 commands sequence. The command is encoded, as shown in the table below:

Table 58.22 CHANGE WRITE COLUMN Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | — | H'85 | — | — | SEQ_8_ID |

(16) CHANGE ROW ADDRESS Command**(a) Command Description**

The CHANGE ROW ADDRESS command changes the row address (block and page) where the cache register contents are to be programmed in the NAND array. It also changes the column address of the selected cache register and enables data input on the specified LUN.

(b) Command Encoding

The CHANGE ROW ADDRESS instruction uses the SEQ_12 commands sequence. The command is encoded, as shown in the table below:

Table 58.23 CHANGE ROW ADDRESS Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | H'11 | H'85 | 0 | — | SEQ_12_ID |

(17) READ PAGE Command**(a) Command Description**

The READ PAGE command copies a page from the NAND Flash array to its respective cache register and enables data output. This command is accepted by the LUN when it is ready. To read a page from the NAND Flash array, the controller writes the H'00 command to the target internal command register, then writes 5 address cycles to the address registers, and concludes with the H'30 command. The selected LUN will go into busy state as the data is transferred. When the LUN is ready, data output is enabled for the cache register linked to the plane addressed in the READ PAGE command. The controller reads the programmed number of bytes to the FIFO.

(b) Command Encoding

The READ PAGE instruction uses the SEQ_10 commands sequence. The command is encoded, as shown in the table below:

Table 58.24 READ PAGE Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| H'30 | — | H'00 | 0/1 | 0/1 | SEQ_10_ID |

(18) READ PAGE CACHE Command**(a) Command Description**

The READ PAGE CACHE command reads the next sequential page within a block into the data register, while the previous page is output from the cache register. To issue this command, the controller writes H'31 to the target internal command register. After this command is issued, the R/B# goes LOW and the LUN goes into busy state. Afterwards, the R/B# goes HIGH and the LUN is busy with a cache operation, indicating that the cache register is available and that the specified page is copying from the NAND Flash array to the data register. At this point, data is read from the cache register.

(b) Command Encoding

The READ PAGE CACHE instruction uses the SEQ_11 commands sequence. The command is encoded, as shown in the table below:

Table 58.25 READ PAGE CACHE Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | — | H'31 | 0/1 | 0/1 | SEQ_11_ID |

(19) READ PAGE CACHE LAST Command**(a) Command Description**

The READ PAGE CACHE LAST command ends the READ PAGE CACHE sequence and copies a page from the data register to the cache register. This command is accepted by the LUN when it is ready. To issue the READ PAGE CACHE LAST command, the controller writes H'3F to the target internal command register. After this command is issued, R/B# goes LOW and the LUN goes into busy state. Afterwards, the R/B# goes HIGH and the LUN is ready. At this point, data from the targets cache register is read into the FIFO.

(b) Command Encoding

The READ PAGE CACHE LAST instruction uses the SEQ_11 commands sequence. The command is encoded, as shown in the table below:

Table 58.26 READ PAGE CACHE LAST Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | — | H'3F | 0/1 | 0/1 | SEQ_11_ID |

(20) READ MULTIPLANE Command**(a) Command Description**

The READ MULTIPLANE command queues a plane to transfer data from the NAND array to its cache register. This command can be issued one or more times. Each time a new plane address is specified, that plane is also queued for data transfer. To select the final plane and to begin the read operation for all previously queued planes, issue the READ PAGE command. All queued planes will transfer data from the NAND array to their cache registers.

(b) Command Encoding

The READ MULTIPLANE instruction uses the SEQ_9 commands sequence. The command is encoded, as shown in the table below:

Table 58.27 READ MULTIPLANE Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | H'32 | H'00 | — | — | SEQ_9_ID |

(21) QUEUE PAGE READ Command**(a) Command Description**

The QUEUE PAGE READ operation allows for partial-page reads by using a 7-address cycle. The first two bytes of the address cycle indicate the length of the page to read - those bytes are stored in ADDR0 register, followed by the column and row addresses - those bytes are stored in the ADDR1 register.

This can help overall performance should only a portion of the page data be needed because only the codeword containing the requested data will have ECC decoded. This command is exclusive for the CLEAR NAND devices.

(b) Command Encoding

The QUEUE PAGE READ instruction uses the SEQ_22 commands sequence. The command is encoded, as shown in the table below:

Table 58.28 QUEUE PAGE READ Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | H'37 | H'07 | — | — | SEQ_22_ID |

(22) TWO PLANE PAGE READ command**(a) Command Description**

This command was implemented to preserve the compatibility with the ONFI 1.x and some older devices.

The TWO PLANE PAGE READ (H'00-H'00-H'30) operation is similar to the PAGE READ (H'00-H'30) operation. It transfers two pages of data from the NAND Flash array to the data registers. Each page must be from a different plane on the same die. The software is responsible for generating correct addresses for the requested pages. Both the ADDR0 and ADDR1 address registers are used in this case.

(b) Command Encoding

The TWO PLANE PAGE READ instruction uses the SEQ_15 commands sequence. The command is encoded, as shown in the table below. In this case, both the address registers are used.

Table 58.29 TWO PLANE PAGE READ Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| H'30 | H'00 | H'00 | 0/1 | 0/1 | SEQ_15_ID |

(23) PROGRAM PAGE Command**(a) Command Description**

The PROGRAM PAGE command allows the application program to input data to a cache register and moves the data from the cache register to the specified block and page address in the array of the selected LUN. This command is accepted by the LUN when it is ready. To input a page to the cache register and move it to the NAND array at the block and page address specified, the controller writes H'80 to the target internal command register. Then five address cycles containing the column address and row address are written. Data input cycles follow. Serial data is input, beginning at the column address specified. When data input is complete, the controller writes H'10 to the target internal command register. The selected LUN goes into the busy state.

(b) Command Encoding

The PROGRAM PAGE instruction uses the SEQ_12 commands sequence. The command is encoded, as shown in the table below:

Table 58.30 PROGRAM PAGE Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | H'10 | H'80 | 0 | 0/1 | SEQ_12_ID |

(24) PROGRAM PAGE IMMEDIATE Command**(a) Command Description**

The PROGRAM PAGE IMMEDIATE command allows the application program to input data to a cache register and moves the data from the cache register to the specified block and page address in the array of the selected LUN. This command is accepted by the LUN when it is ready. To input a page to the cache register and move it to the NAND array at the block and page address specified, the controller writes H'80 to the target internal command register. Then five address cycles containing the column address and row address are written. Data input cycles follow. Serial data is input, beginning at the column address specified. When data input is complete, the controller writes H'10 to the target internal command register. The selected LUN goes into the busy state. This command writes only the row address to the NAND flash device

(b) Command Encoding

The PROGRAM PAGE IMMEDIATE instruction uses the SEQ_23 commands sequence. The command is encoded, as shown in the table below:

Table 58.31 PROGRAM PAGE IMD Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | H'10 | H'80 | 0 | 0/1 | SEQ_23_ID |

(25) PROGRAM PAGE DELAYED Command**(a) Command Description**

The device's internal controller can automatically manage multi-plane programming. It does this with PROGRAM PAGE DELAYED command. When this command issued, the controller will delay issuing the program operation to the array until the address for the subsequent program operation is examined. If that operation allows the previous operation to complete as part of multi-plane operation, the controller will issue a multi-plane program to the array. Multi-plane operations are only completed if an LUN address from plane 0 is issued prior to a LUN address from plane 1. If the subsequent program operation does not allow the multi-plane operation, the controller will immediately start the previous program. It is presumed that the application program may use this command to initiate all the program operations. In this way the application program does not have to maintain information regarding multi-plane operation usage.

(b) Command Encoding

The PROGRAM PAGE DELAY instruction uses the SEQ_12 commands sequence. The command is encoded, as shown in the table below:

Table 58.32 PROGRAM PAGE DELAY Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | H'13 | H'80 | 0 | 0/1 | SEQ_23_ID |

(26) PROGRAM PAGE 1 Command**(a) Command Description**

The PROGRAM PAGE 1 command allows the application program to input data to a cache register and moves the data from the cache register to the specified block and page address in the array of the selected LUN. This command is accepted by the LUN when it is ready. To input a page to the cache register and move it to the NAND array at the block and page address specified, the controller writes H'80 to the target internal command register. Then five address cycles containing the column address and row address are written. Data input cycles follow. Serial data is input, beginning at the column address specified. When data input is complete, the commands sequence ends.

(b) Command Encoding

The PROGRAM PAGE 1 instruction uses the SEQ_13 commands sequence. The command is encoded, as shown in the table below:

Table 58.33 PROGRAM PAGE 1 Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | — | H'80 | 0 | 0/1 | SEQ_13_ID |

(27) PROGRAM PAGE CACHE Command**(a) Command Description**

The PROGRAM PAGE CACHE command allows the controller to input data to a cache register, copies the data from the cache register to the data register, and then moves the data register contents to the specified block and page address in the array of the selected LUN. After the data is copied to the data register, the cache register is available for additional PROGRAM PAGE CACHE or PROGRAM PAGE commands. The PROGRAM PAGE CACHE command is accepted by the LUN when it is ready.

To input a page to the cache register to move it to the NAND array at the block and page address specified, the controller writes H'80 to the target internal command register. Then five address cycles containing the column address and row address are written. Data input cycles follow. Serial data is input, beginning at the column address specified. When data input is complete, H'15 is written to the command register. The selected LUN goes into busy state to allow the data register to become available from a previous program cache operation, to copy data from the cache register to the data register, and then to begin moving the data register contents to the specified page and block address.

(b) Command Encoding

The PROGRAM PAGE CACHE instruction uses the SEQ_12 commands sequence. Command is encoded as shown in the table below.

Table 58.34 PROGRAM PAGE CACHE Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | H'15 | H'80 | 0 | 0/1 | SEQ_12_ID |

(28) PROGRAM MULTIPLANE Command**(a) Command Description**

The PROGRAM MULTIPLANE command allows the controller to input data to the addressed plane's cache register and queue the cache register to ultimately be moved to the NAND array. This command can be issued one or more times. Each time a new plane address is specified that plane is also queued for data transfer. This command is accepted by the LUN when it is ready. The controller writes H'80 to the target internal command register. Then five address cycles containing the column address and row address are written. Data input cycles follow. Serial data is input beginning at the column address specified. When data input is complete, the controller writes H'11 to the target internal command register.

(b) Command Encoding

The PROGRAM PAGE CACHE instruction uses the SEQ_12 commands sequence. Command is encoded as shown in the table below.

Table 58.35 PROGRAM MULTIPLANE Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | H'11 | H'80 | 0 | 0/1 | SEQ_12_ID |

(29) WRITE PAGE Command**(a) Command Description**

The WRITE PAGE command allows the controller to move data from the targets cache register to the NAND array. This command is accepted by the LUN when it is ready. The controller writes H'10 to the target internal command register.

(b) Command Encoding

The WRITE PAGE instruction uses the SEQ_0 commands sequence. The command is encoded, as shown in the table below:

Table 58.36 WRITE PAGE Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | — | H'10 | — | — | SEQ_0_ID |

(30) WRITE PAGE CACHE Command**(a) Command Description**

The WRITE PAGE CACHE command allows the controller to move data from the targets cache register to the targets data register. This command is accepted by the LUN when it is ready. The controller writes H'15 to the target internal command register.

(b) Command Encoding

The WRITE PAGE CACHE instruction uses the SEQ_0 commands sequence. The command is encoded, as shown in the table below:

Table 58.37 WRITE PAGE CACHE Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | — | H'15 | — | — | SEQ_0_ID |

(31) WRITE MULTIPLANE Command**(a) Command Description**

The WRITE MULTIPLANE command allows the controller to queue data from the targets cache register to the NAND array. This command is accepted by the LUN when it is ready. The controller writes H'11 to the target internal command register.

(b) Command Encoding

The WRITE MULTIPLANE instruction uses the SEQ_0 commands sequence. The command is encoded, as shown in the table below:

Table 58.38 WRITE MULTIPLANE Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | — | H'11 | — | — | SEQ_0_ID |

(32) ERASE BLOCK Command**(a) Command Description**

The ERASE BLOCK command erases the specified block in the NAND array. This command is accepted by the LUN when it is ready. To erase a block, the controller writes H'60 to the target internal command register. Then three address cycles containing the row address are written; the column address is ignored. Finally, H'D0 is written to the target internal command register.

(b) Command Encoding

The ERASE BLOCK instruction uses the SEQ_14 commands sequence. The command is encoded, as shown in the table below:

Table 58.39 ERASE BLOCK Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | H'D0 | H'60 | — | — | SEQ_14_ID |

(33) ERASE MULTIPLANE Command**(a) Command Description**

The ERASE MULTIPLANE command queues a block in the specified plane to be erased from the NAND array. This command can be issued one or more times. Each time a new plane address is specified, that plane is also queued for a block to be erased. This command is accepted by the LUN when it is ready. To queue a block to be erased, the controller writes H'60 to the command register.

Then three address cycles containing the row address are written; the column address is ignored.

Finally, H'D1 is written to the command register.

(b) Command Encoding

The ERASE MULTIPLANE instruction uses the SEQ_14 commands sequence. The command is encoded, as shown in the table below:

Table 58.40 ERASE MULTIPLANE Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | H'D1 | H'60 | — | — | SEQ_14_ID |

(34) COPYBACK READ Command**(a) Command Description**

The COPYBACK READ command is functionally identical to the READ PAGE command, except that H'35 is written to the target internal command register instead of H'30. For more details, see section 58.3.3 (17), READ PAGE Command.

(b) Command Encoding

The COPYBACK READ instruction uses the SEQ_10 commands sequence. The command is encoded, as shown in the table below:

Table 58.41 COPYBACK READ Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| H'35 | — | H'00 | 0/1 | 0/1 | SEQ_10_ID |

(35) COPYBACK PROGRAM Command**(a) Command Description**

The Copyback function reads a page of data from one location and then moves that data to a second location. The COPYBACK PROGRAM command is functionally identical to the PROGRAM PAGE command, except that when H'85 is written to the target internal command register, the cache register contents are not cleared. The SEQ_9 command sequence does not have the data phase, so the data from the cache register are written into the second location without modification. If data must be written with modification to the second location, the SEQ_12 command sequence, which includes the data phase, is used.

(b) Command Encoding

The COPYBACK PROGRAM instruction uses the SEQ_9 commands sequence. The command is encoded, as shown in the table below:

Table 58.42 COPYBACK PROGRAM Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | H'10 | H'85 | — | — | SEQ_9_ID |

(36) COPYBACK PROGRAM 1 Command**(a) Command Description**

The COPYBACK PROGRAM 1 command is functionally identical to the PROGRAM PAGE 1 command, except that when H'85 is written to the target internal command register, the cache register contents are not cleared. See section 58.3.3 (26), PROGRAM PAGE 1 Command for further details.

(b) Command Encoding

The COPYBACK PROGRAM 1 instruction uses the SEQ_13 commands sequence. The command is encoded as shown in the table below:

Table 58.43 COPYBACK PROGRAM 1 Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | — | H'85 | 0 | — | SEQ_13_ID |

(37) COPYBACK MULTIPLANE Command**(a) Command Description**

The COPYBACK MULTIPLANE command is functionally identical to the PROGRAM MULTIPLANE command, except that when H'85 is written to the target internal command register, cache register contents are not cleared. See section 58.3.3 (28), PROGRAM MULTIPLANE Command for further details.

(b) Command Encoding

The COPYBACK MULTIPLANE instruction uses the SEQ_12 commands sequence. The command is encoded, as shown in the table below:

Table 58.44 COPYBACK MULTIPLANE Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | H'11 | H'85 | 0 | 0/1 | SEQ_12_ID |

(38) PROGRAM OTP Command**(a) Command Description**

The PROGRAM OTP command is used to write data to the pages within the OTP area. An entire page can be programmed at one time, or a page can be partially programmed up to four times. There is no ERASE operation for the OTP pages. To use the PROGRAM OTP command, the controller issues the H'A0 command. Next, 5 address cycles are issued. The address write is followed by a programmable number of data cycles. After data input is complete, the controller issues the H'10 command. The internal control logic automatically executes the proper programming algorithm and controls the necessary timing for programming and verification.

(b) Command Encoding

The PROGRAM OTP instruction uses the SEQ_12 commands sequence. The command is encoded as shown in the table below:

Table 58.45 PROGRAM OTP Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|-----------|
| — | H'10 | H'A0 | 0 | 0/1 | SEQ_12_ID |

(39) DATA PROTECT OTP Command**(a) Command Description**

The DATA PROTECT OTP command is used to protect all the data in the OTP area. After the data is protected, it cannot be further programmed. When the OTP area is protected, the pages within the area are no longer programmable and cannot be unprotected. To use the DATA PROTECT OTP command, the controller issues the H'A5 command. Next, the controller issues the following 5 addresses cycles.

Finally, the H'10 command is issued.

(b) Command Encoding

The DATA PROTECT OTP instruction uses the SEQ_9 commands sequence. The command is encoded, as shown in the table below:

Table 58.46 DATA PROTECT OTP Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|-------|-------------|-------|----------|-----------|----------|
| — | H'10 | H'A5 | — | — | SEQ_9_ID |

(40) PAGE READ OTP Command**(a) Command Description**

The PAGE READ OTP command is used to read data from a page within the OTP area. An OTP page within the OTP area is available for reading data whether or not the area is protected. To use the PAGE READ OTP command, the controller issues the H'AF command. Next, 5 address cycles are issued.

Finally, the H'30 command is issued. After internal read from the NAND matrix is ended, the data is copied to the FIFO.

(b) Command Encoding

The PAGE READ OTP instruction uses the SEQ_9 commands sequence. The command is encoded, as shown in the table below:

Table 58.47 PAGE READ OTP Instruction Encoding

| CMD_2 | CMD_1/CMD_3 | CMD_0 | DATA_SEL | INPUT_SEL | CMD_SEQ |
|--------------|--------------------|--------------|-----------------|------------------|----------------|
| H'30 | — | H'AF | 0/1 | 0/1 | SEQ_10_ID |

58.3.4 Multi LUN Work Mode

The multi LUN work mode is enabled by setting the MLUN_EN bit in the CONTROL register. In this mode, each LUN is handled as separate target. The active LUN number is decoded directly from the address value.

This multi LUN mode is parameterized by the following parameters:

The LUN_IDX field in the MLUN register. This parameter provides the bits index for the last address byte where the LUN select bits is present. This parameter must be set according to the NAND Flash device datasheet in use.

The LUN_SEL bits in MLUN register is used to configure the number of LUN-s per device.

The LUN STATUS 0 and 1 holds the LUN-s statuses, each bit correspond to the single LUN status.

The STATE_MASK field in the STAT_RSP register. This parameter is used to mask the part of the LUN status byte that will be ignored during the LUN ready/busy check.

58.3.5 Remapping Mechanism

The remapping mechanism supports the bad blocks management (BBM) solution. The hardware remapping mechanism relieves software from the time-consuming operations of finding the physical address for the given linear address in the requested operation. The software initializes only the remapping tables for uses in the application of the NAND Flash device. It sorts those tables in ascending order, and then the whole operation of searching tables and substituting addresses is accomplished automatically.

The remapping solution uses two groups of the control registers:

- The pointer register DEV0_PTR. This register stores the address in the internal memory where record tables used by the BBM mechanism are placed.
- The size register DEV0_SIZE. This register stores the number of records in a table.

Note for Figure 58.31:

Block Number are written to Destination Address and Source Address.

The table have to set by ascending order of Source Address.

When some Address of one record is unused, these Address should filled by H'FFFF_FFFF.

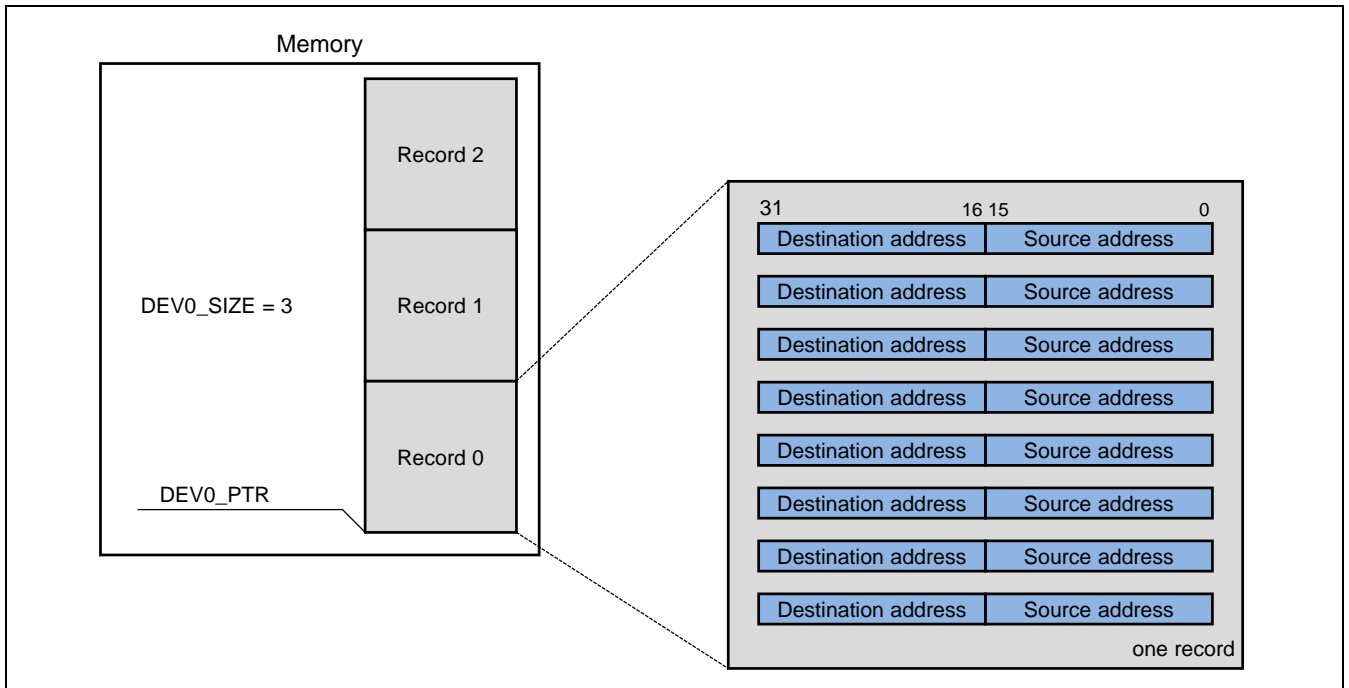


Figure 58.31 Example of BBM records

58.3.6 Interrupts Mechanism

The NAND Flash controller interrupt system uses two control registers:

The interrupt mask register – each bit of this register masks a single interrupt. The register is described in more detail in section 58.2.6, Interrupts mask register (INT_MASK).

The interrupt flag register – each bit of this register is an active flag from the single interrupt source.

The register is described in more detail in section 58.2.7, Interrupts status register (INT_STATUS).

In case an interrupt event occurs, i.e. the respective interrupt flag in INT_STATUS is 1, and its respective mask bit in INT_MASK is 1, the interrupt signal INTNFMA is asserted.

Note that the interrupt mask bits in INT_MASK affects only the interrupt signal INTNFMA, but not the flags in INT_STATUS.

A set flag in INT_STATUS must be cleared by the application program.

The available interrupt sources and their respective INT_STATUS flags are:

INT_STATUS.CMD_END_INT_FL

Command sequence finished interrupt – This interrupt occurs when the previously triggered command sequence is finished and the new one can be started. The command sequence is marked as finished when the full sequence is executed or when the NAND Flash device goes into the busy state.

INT_STATUS.ECC_INT0_FLAG

The ECC module detects an uncorrectable error in the transmitted data.

The ECC module detects when the configured errors threshold level is exceeded.

INT_STATUS.MEM0_RDY_INT_FL

The memory device is ready – This interrupt occurs when the NAND Flash device finishes executing the programmed command sequence and is ready for the new one. Each NAND Flash device has a single interrupt flag. The difference between “command sequence finished” interrupt and “memory device is ready” interrupt is presented in Figure 58.32.

INT_STATUS.TRANS_ERR_FL

The error on the slave interface during access to the controller FIFO – This interrupt occurs if the access to the FIFO memory has the opposite direction to the current FIFO configuration: the FIFO is read when it is configured for write, or the FIFO is write when it is configured for read.

INT_STATUS.STAT_ERR_INT0_FL

Most recently finished operation on the memory device failed. This applies to PROGRAM PAGE and BLOCK ERASE operations. It is not valid following a READ-series operation.

INT_STATUS.PG_SZ_ERR_INT_FL

Data Size error flag.

When the ECC is enabled, the value written into the DATA_SIZE register has some restrictions.

Interrupt condition is met when the value written to the DATA_SIZE register is not correct.

INT_STATUS.DMA_INT_FL

DMA transfer ended flag.

INT_STATUS.DATA_REG_INT_FL

Data in DATA_REG is available

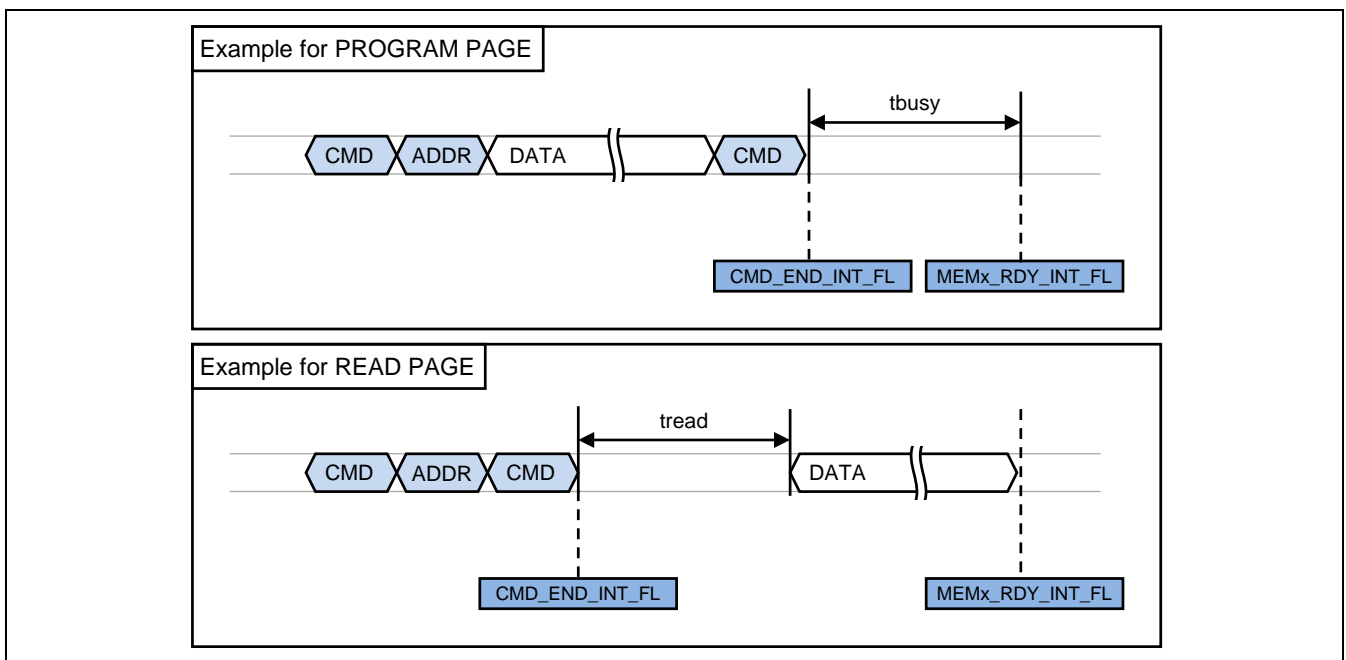


Figure 58.32 Command Sequence End and Memory-Ready Interrupts

Notes:

Interrupt is asserted before bus transaction completion.

Dedicated workaround to avoid data coherency issues needs to be implemented:

Always load additional, not used 64 byte of data in that way it is ensured that the interrupt asserted when the previous intended data was completely transferred to SDRAM.

58.3.7 Setup and Configuration

The CONTROL register is the main control register in the NAND Flash controller.

The following bits configure basic settings of the controller:

- INT_EN – Bit which enables Global Interrupt.
- ECC_EN – Bit which enables Hardware ECC.
- BLOCK_SIZE – Bits which configure block size.
- IO_WIDTH – Bit which configures width of the I/O bus connected to the NAND Flash memory device.
- BBM_EN – Bit which enables remapping process.
- ADDR_x_AUTO_INCR - Addresses auto increment for row address register 0 or 1.
- SMALL_BLOCK_EN – Bit which enables Small Block Mode.
- MLUN_EN – Bit which enables Multi LUN mode.
- AUTO_READ_STAT_EN – Bit which activates automatic read status after the PROGRAM PAGE and BLOCK ERASE commands.
- READ_STATUS_EN – Bit which chooses whether the controller checks RnB lines or sends READ_STATUS commands.
- ECC_BLOCK_SIZE – Bits which define ECC Block Size.

The registers described below are configured according to the settings of other bits in the CONTROL register:

- If INT_EN bit is set, the software must write the mask into the INT_MASK register, which masks the selected interrupts source in the NAND Flash controller.
- If ECC_EN bit is set, the software must correctly configure the ECC module by writing the appropriate configuration into the ECC_CTRL. Additionally, the software configures the offset in the ECC_OFFSET register. In small block mode, the value in the ECC_OFFSET is ignored and the correction words are located in the NAND Flash memory device, right behind the data.
- The write number of the data which will be transferred by the controller (DATA_SIZE register).
When ECC is enabled, there are some restrictions to the DATA_SIZE value.
- If the BBM_EN bit is set, the software must initialize the remapping tables (DEVn_PTR and DEVn_SIZE registers).
- Additionally, the software must configure time parameters which can be found in the TIMINGS_ASYN register. Additionally, the software must configure the TIME_SEQ_0 and TIME_SEQ_1 registers.

When the NAND Flash controller uses DMA to transfer data, the software must configure the DMA_ADDR, DMA_CTRL and DMA_CNT registers. The software can modify these registers before any transfer or during the initialization procedure.

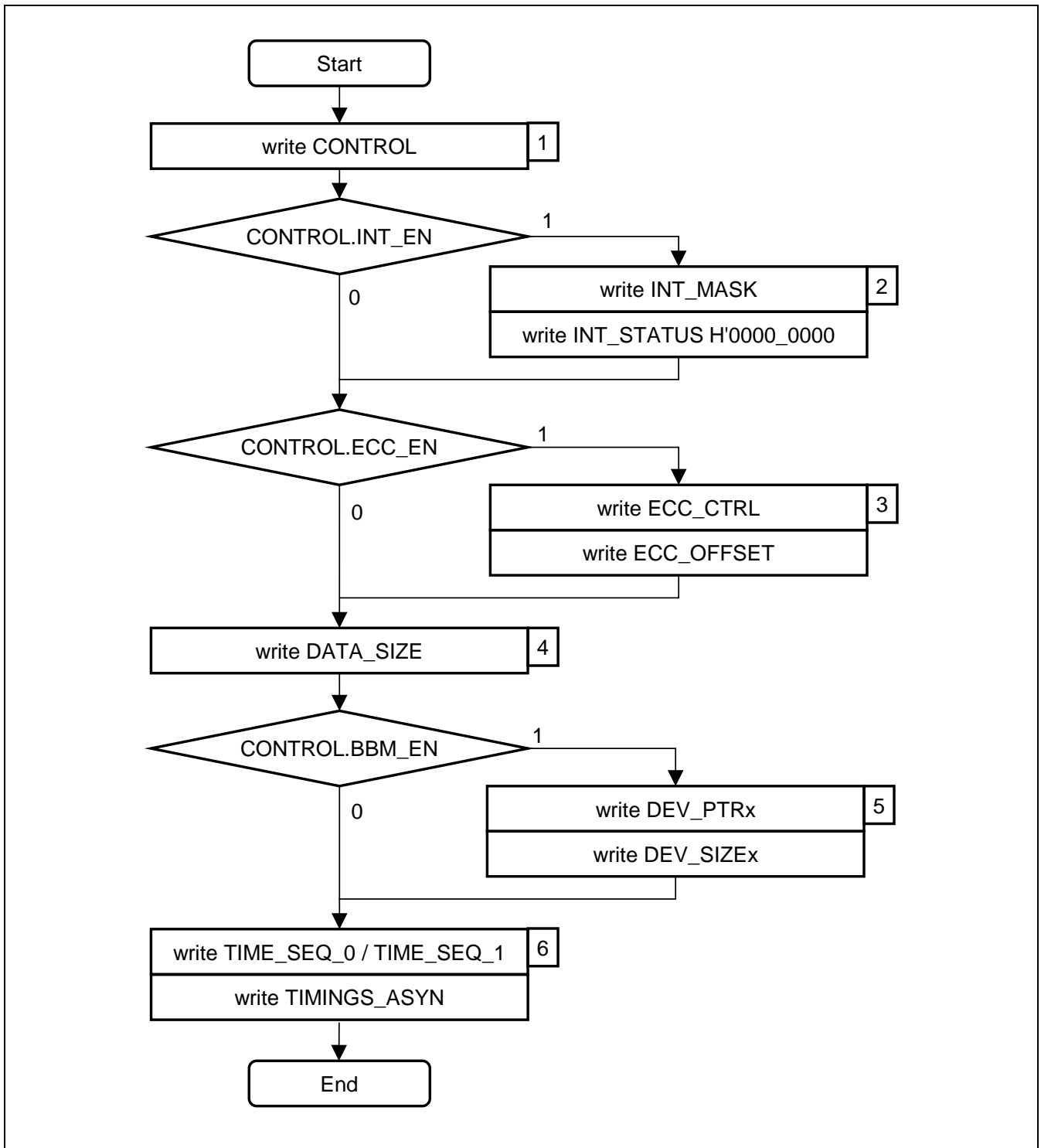


Figure 58.33 Configuration

(1) Send Data to NAND Flash via Slave Interface

The NAND Flash controller must be correctly configured before sending data to the NAND Flash memory device. The setup process is described in detail in section 58.3.7, Setup and Configuration and section 58.2, Register Descriptions.

1. Write the address of the data in NAND Flash memory into the address register 0 (ADDR_COL and ADDR_ROW registers). Write the number of data which you want to read (DATA_SIZE register).
2. To use the simplest program command, write H'0010_800C to the COMMAND register (PROGRAM PAGE command, FIFO module selected, registers as input).
3. Write data to the FIFO using the FIFO_DATA register. Data is sent to the NAND Flash memory device.

When the memory is ready for further work, the appropriate bit MEMx_ST is set. When the software needs to send the command to another memory, it is not necessary to wait for the MEMx_ST bit to be set.

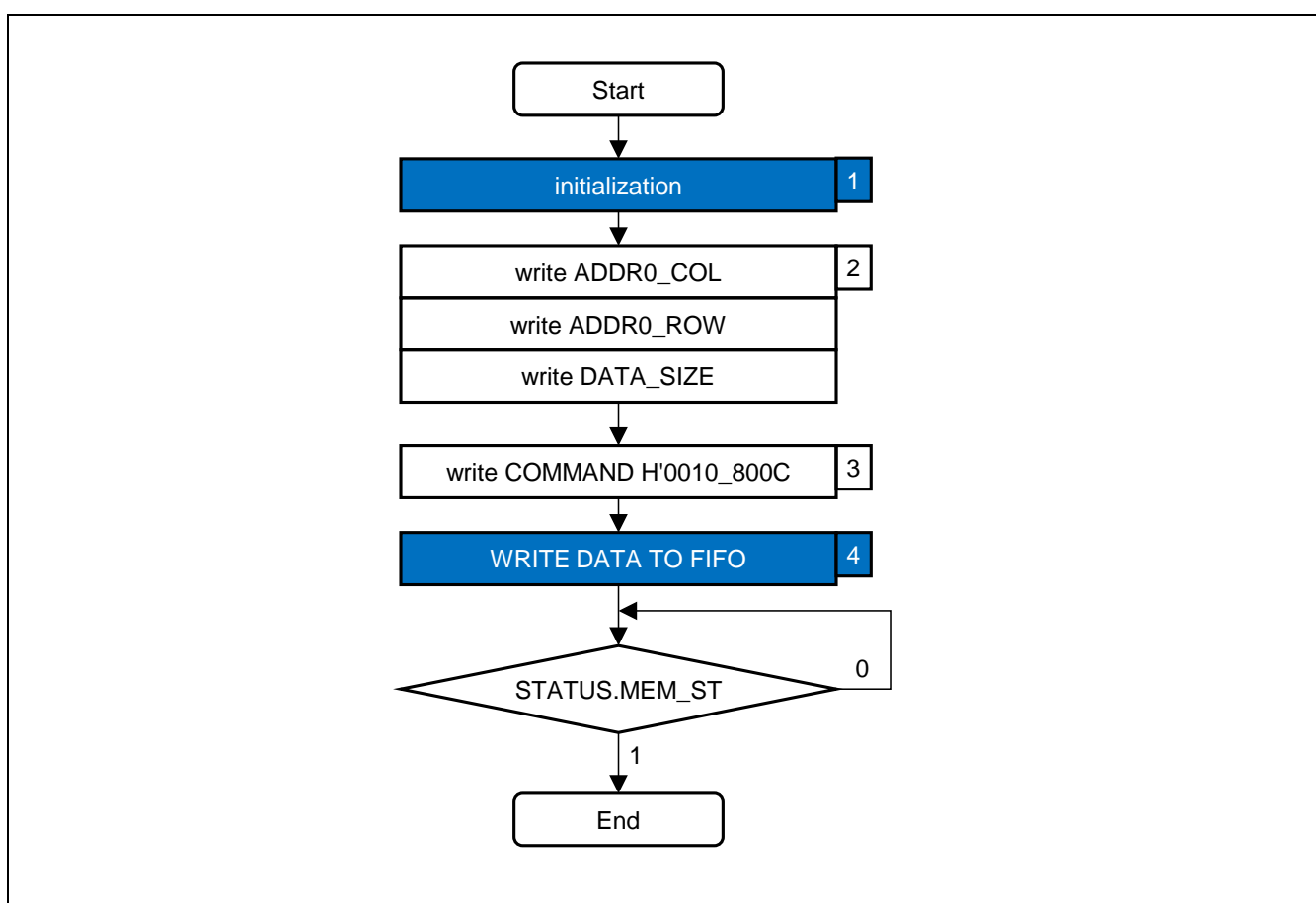


Figure 58.34 Write data to NAND Flash memory via Slave Interface

(2) Read Data from NAND Flash via Slave Interface

1. The NAND Flash controller must be correctly configured before sending data to the NAND Flash memory device. The setup process is described in detail in section 58.3.7, Setup and Configuration and section 58.2, Register Descriptions.
2. Write the address of the data in NAND Flash memory into the address register 0 (ADDR_COL and ADDR_ROW registers).
3. To use the simplest read command, write H'3000_002A to the COMMAND register (READ PAGE command, FIFO module selected, registers as input).
4. The application program can read data immediately after sending the command but in this case the NAND flash controller will send a very few WAIT replies.
If this situation shall be avoided the application program must read FIFO_STATE register and wait when CF_EMPTY bit is set. After that wait for the DF_R_EMPTY bit in the FIFO_STATE register to be clear.
5. Read data from the FIFO using the FIFO_DATA register.

When the memory is ready for further work, the appropriate bit MEMx_ST is set. When the software needs to send the command to another memory, it is not necessary to wait for the MEMx_ST bit to be set.

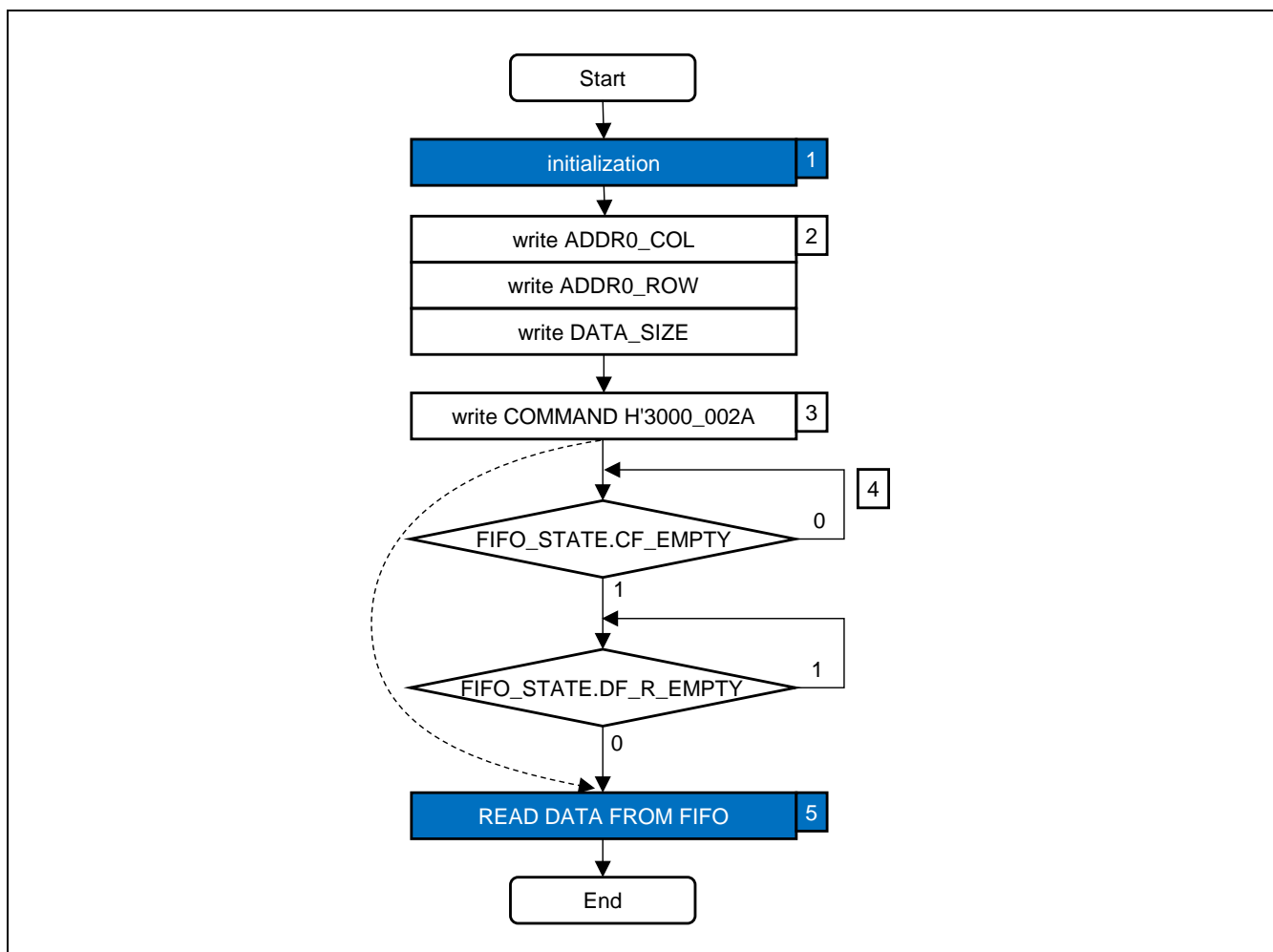
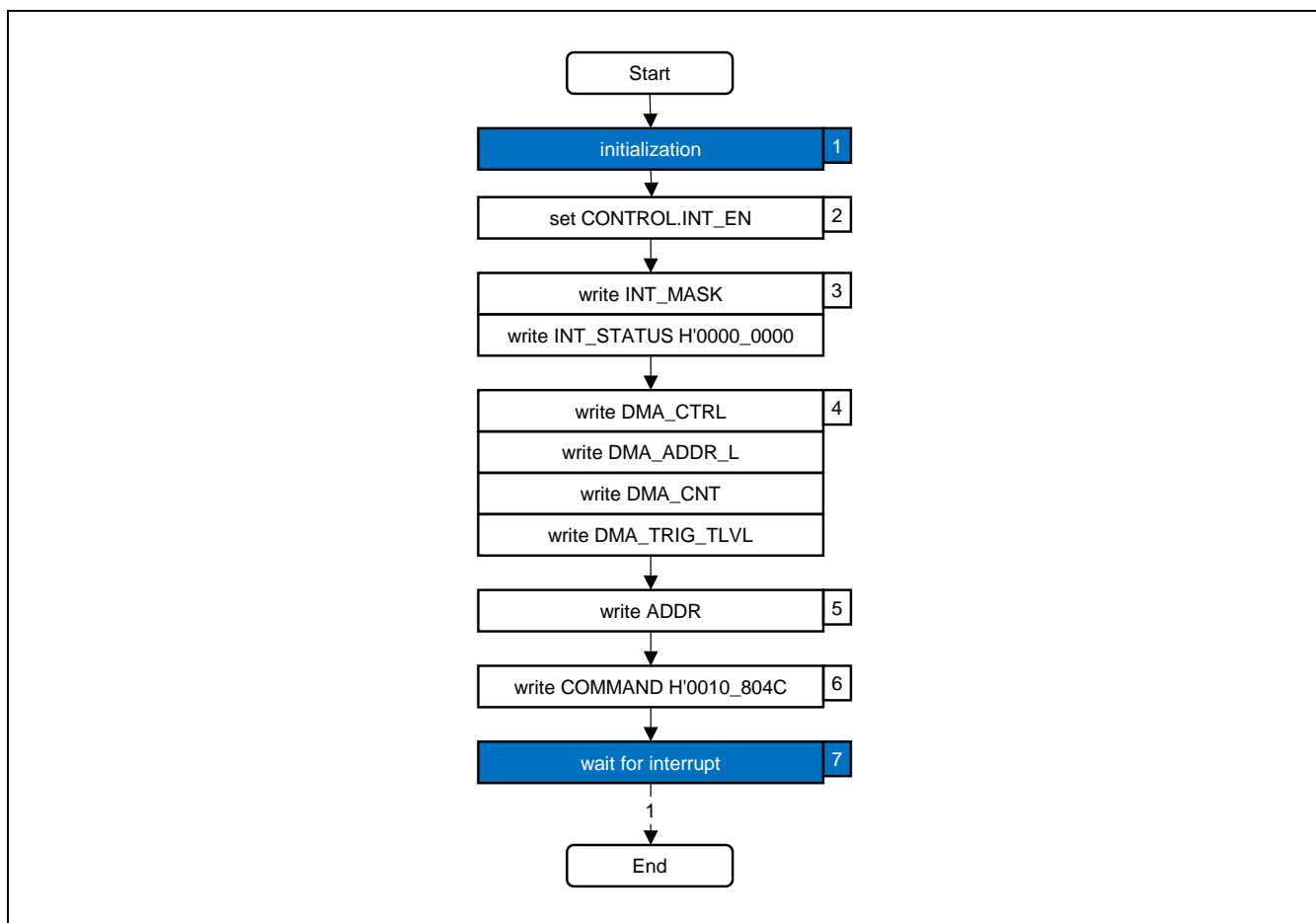


Figure 58.35 Read data from NAND Flash memory via Slave Interface

(3) Send Data to NAND Flash via Master Interface (using DMA)

1. The NAND Flash controller must be correctly configured before sending data to the NAND Flash memory device. The setup process is described in detail in section 58.3.7, Setup and Configuration and section 58.2, Register Descriptions.
2. Set the INT_EN bit in the CONTROL register to enable global interrupt.
3. Select active interrupt in the INT_MASK register and write H'0000_0000 into the INT_STATUS register to clear all interrupts.
4. Select the DMA work mode to configure the DMA module correctly:
In registers managed mode, the address of the data is written in the system memory (DMA_ADDR_L registers). Write the number of the transferred data into the DMA_CNT register. Set bit DMA_START to start DMA when the command sequence is sent to the NAND Flash memory. Bits ERR_FLAG and DMA_READY are read-only. The first indicates the error on the internal bus while DMA is transferring data; the second indicates when DMA is ready (transfer is completed).
5. Write the address of the data in the NAND Flash memory device into the address register 0.
6. To use the simplest program command, write H'0010_804C to the COMMAND register (PROGRAM PAGE command, FIFO module selected, DMA module as input).
7. When the memory is ready for further work, the appropriate bit MEMx_ST is set and interrupt is active. (recommended interrupt is MEM0_RDY_INT)

**Figure 58.36 Send Data to NAND Flash Using DMA, Interrupt Enable**

(4) Fast Writing and Reading of Several Pages from the Memory using DMA

1. The NAND Flash controller must be correctly configured before sending data to the NAND Flash memory device. The setup process is described in detail in section 58.3.7, Setup and Configuration and section 58.2, Register Descriptions.
2. Set INT_EN bit in CONTROL register to enable global interrupt.
3. Select the active interrupt (CMD_END_INT_EN) in the INT_MASK register and write H'0000_0000 into the INT_STATUS register to clear all interrupts.
4. In Scatter-Gather mode, it is necessary to write the descriptors into the system memory. Select the DMA work mode to correctly configure the DMA module. In Scatter-Gather mode, it is not necessary to configure the DMA_CNT register. For more details, see section 58.4.2 (2), DMA Description, (b)
5. Set DMA_START bit to start the DMA when the command sequence is sent to the NAND Flash memory. Bits ERR_FLAG and DMA_READY are read-only. The former indicates the error on the internal bus while DMA is transferring data, the latter indicates when the DMA is ready (transfer is completed).
6. Set the ADDR0_INCR bit to auto increment row address 0 register after each command. Write address of the data in the NAND Flash memory device into the 0 address register.
7. Write H'0000_0000 into the INT_STATUS register to clear all interrupts.
8. Write address of the first descriptor into the DMA_ADDR_L registers.
9. Write PROGRAM PAGE CACHE command to the NAND Flash memory device by writing H'0015_804C into the COMMAND register (DMA module as input, FIFO module selected). When the controller is ready for further work, the appropriate CMD_END_INT_FL bit is set and the interrupt is activated.
10. When the number of the pages to transfer does not equal one, go to step 7 of this procedure.
11. Write H'0000_0000 into the INT_STATUS register to clear all interrupts.
Write address of the first descriptor into the DMA_ADDR_L register.
The last command in the sequence of sending data is the PROGRAM PAGE (write H'0010_804C) into the COMMAND register (DMA module as input, FIFO module selected).
12. Wait for interrupts and write H'0000_0000 into the INT_STATUS register to clear all interrupts.
Write the address of the read data to the NAND Flash memory device into address register 0.
13. Write the new descriptors to the system memory.
14. If DMA should work in the same work mode and burst type do not modify DMA_BURST and DMA_MODE bits.
Set DMA_START bit to start DMA when the command sequence is sent to the NAND Flash memory.
15. Write the READ PAGE command into the NAND Flash memory device by writing H'3000_0069 into the COMMAND register (FIFO module selected, DMA module as input).
16. Write H'0000_0000 into the INT_STATUS register to clear all interrupts.
17. Write the address of the first descriptor to the DMA_ADDR_L registers.
18. Write READ PAGE CACHE SEQUENTIAL command to the NAND Flash memory device by writing H'0000_316B into the COMMAND register (FIFO module selected, DMA module as input).
19. When the controller is ready for further work, the appropriate CMD_END_INT_FL bit is set and the interrupt is activated. When the number of the pages to transfer does not equal zero, go to step 16 of this procedure.
20. Write H'0000_0000 into the INT_STATUS register to clear all interrupts. Write the address of the first descriptor to the DMA_ADDR_L register. The last command in the sequence of reading data is READ PAGE CACHE LAST (write H'0000_3F6B into the COMMAND register).

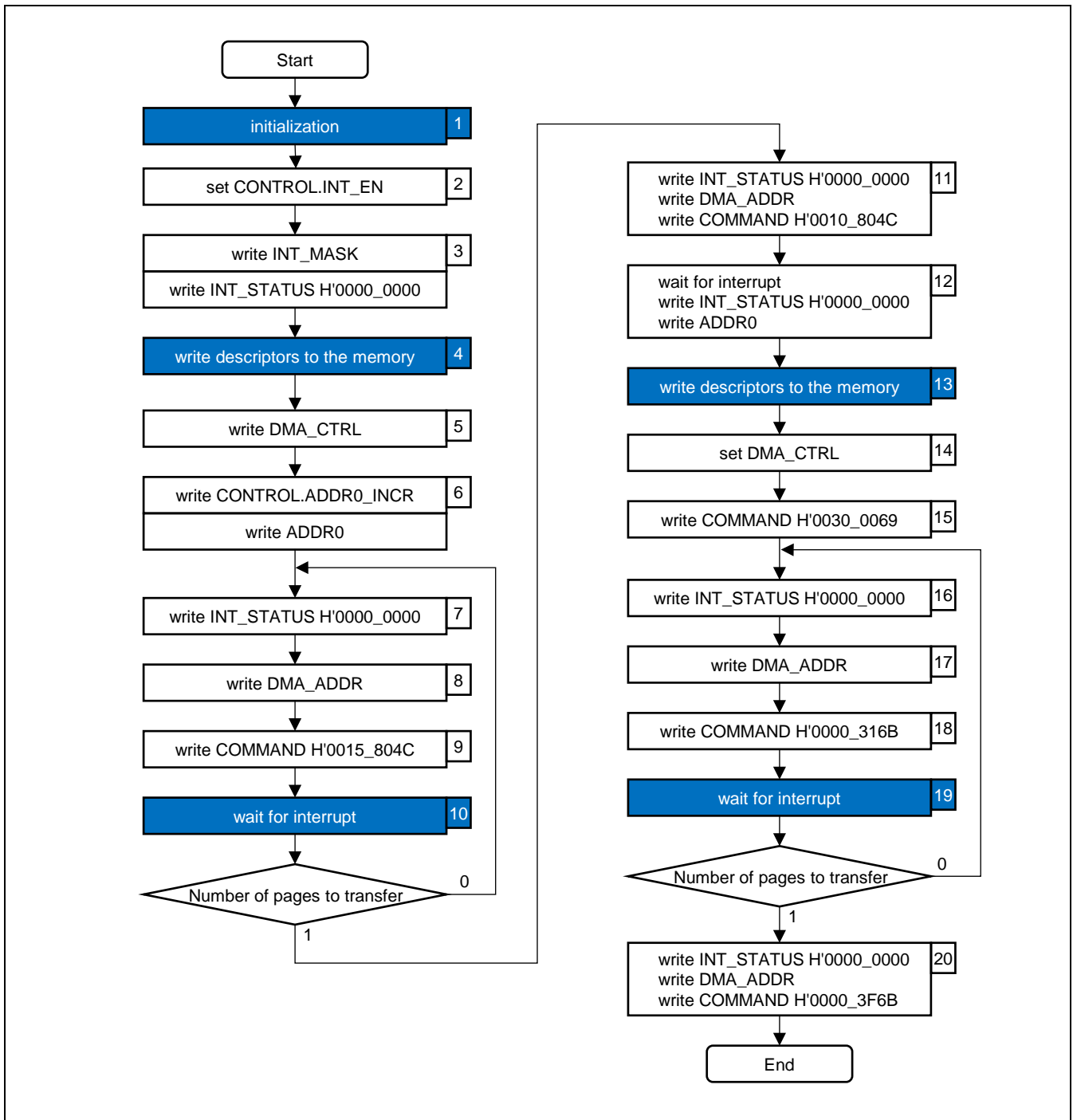


Figure 58.37 Fast Writing and Reading of Several Pages from the Memory Using DMA

(5) Writing Partial Pages

The following procedure need to be used to write partial pages with ECC engine enabled:

1. The NAND Flash controller must be correctly configured before sending data to the NAND Flash memory device. The setup process is described in detail in section 58.3.7, Setup and Configuration and section 58.2, Register Descriptions.
2. Write the address of the data in NAND Flash memory into the address register 0 (ADDR_COL and ADDR_ROW registers). You need to write partial page offset into the ADDR_COL register. Write the number of data which you want to read (DATA_SIZE register) – in this case you need to set partial sector size. Write the offset value of the ecc data into the ECC_OFFSET register.
3. To use the simplest program command, write H'0010_800C to the COMMAND register (PROGRAM PAGE command, FIFO module selected, registers as input).
4. Write data to the FIFO using the FIFO_DATA register. Data is sent to the NAND Flash memory device.

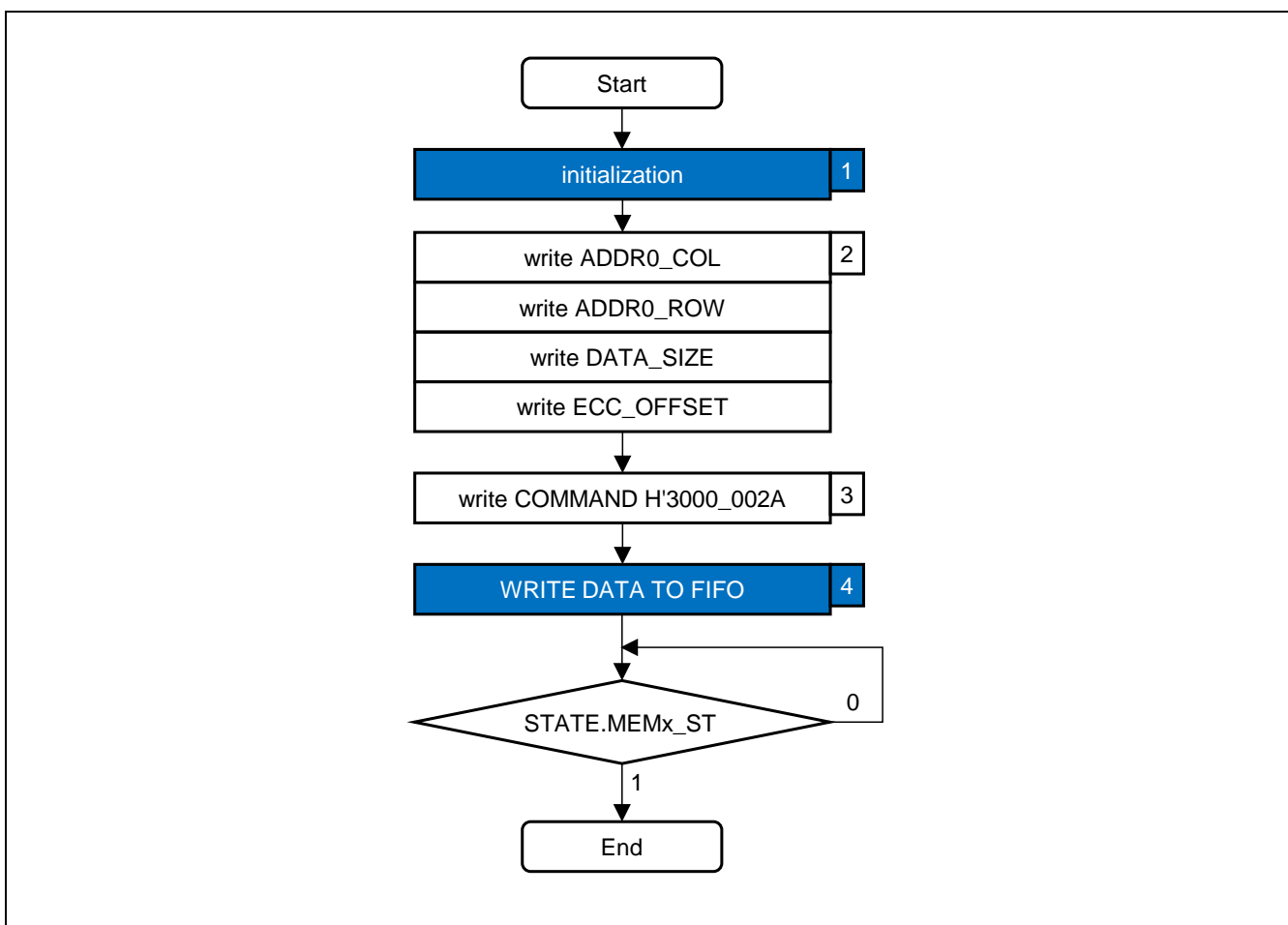


Figure 58.38 Writing Partial Pages

(6) Reading Partial Pages

1. The NAND Flash controller must be correctly configured before sending data to the NAND Flash memory device. The setup process is described in detail in section 58.3.7, Setup and Configuration and section 58.2, Register Descriptions.
2. Write the address of the data in NAND Flash memory into the address register 0 (ADDR_COL and ADDR_ROW registers). You need to write partial page offset into the ADDR_COL register. Write the number of data which you want to read (DATA_SIZE register) – in this case you need to set partial page size.
3. To use the simplest read command, write H'3000_002A to the COMMAND register (READ PAGE command, FIFO module selected, registers as input).
4. You can read data immediately after sending the command but in this case the NAND flash controller will send a very few WAIT replies. If this situation shall be avoided the application program must read FIFO_STATE register and wait when CF_EMPTY bit is set. After that wait for the DF_R_EMPTY bit in the FIFO_STATE register to be clear.
5. Read data from the FIFO using the FIFO_DATA register.

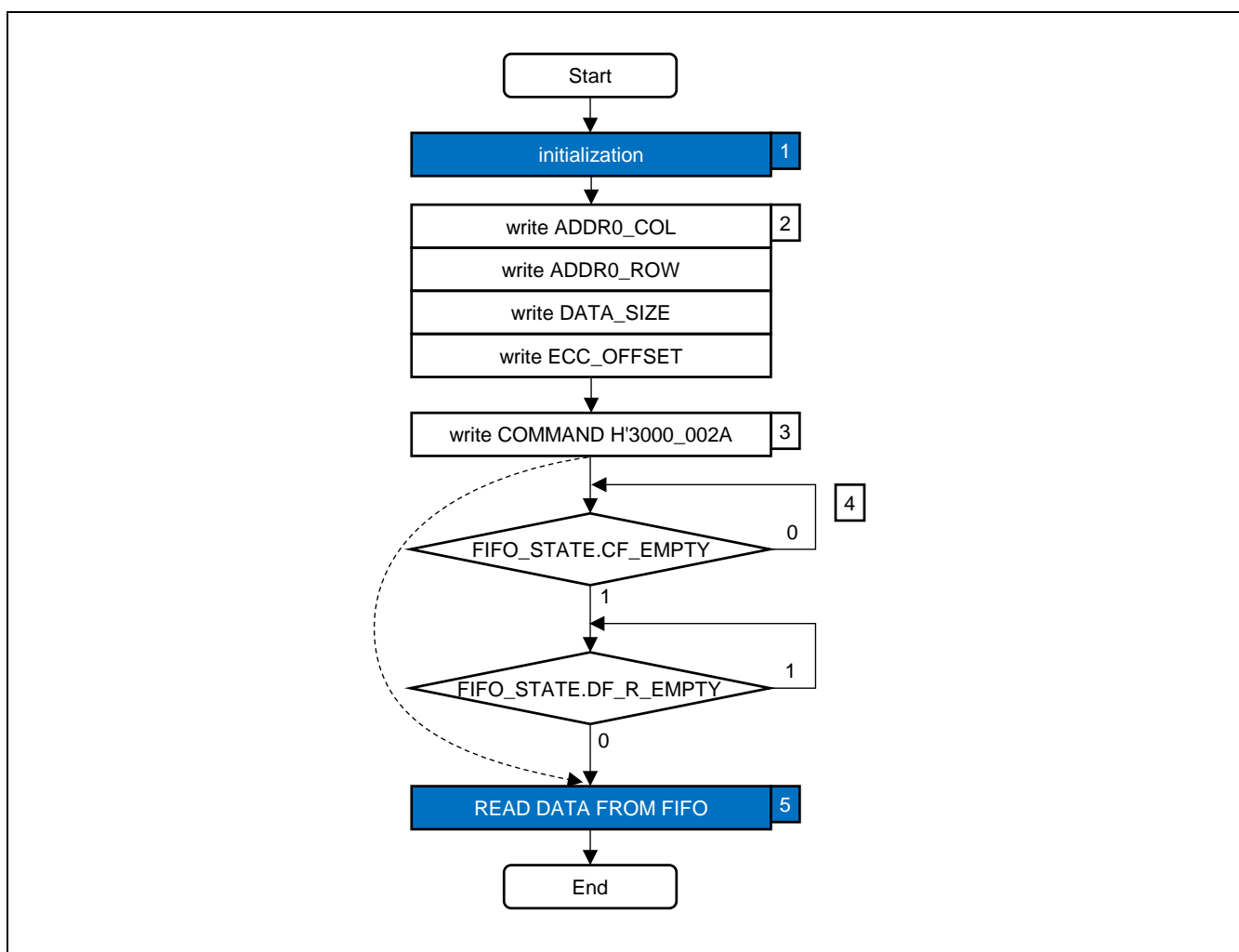


Figure 58.39 Reading Partial Pages

58.4 Functional Details

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

58.4.1 Block Diagram

The figure below shows the NAND Flash controller block diagram.

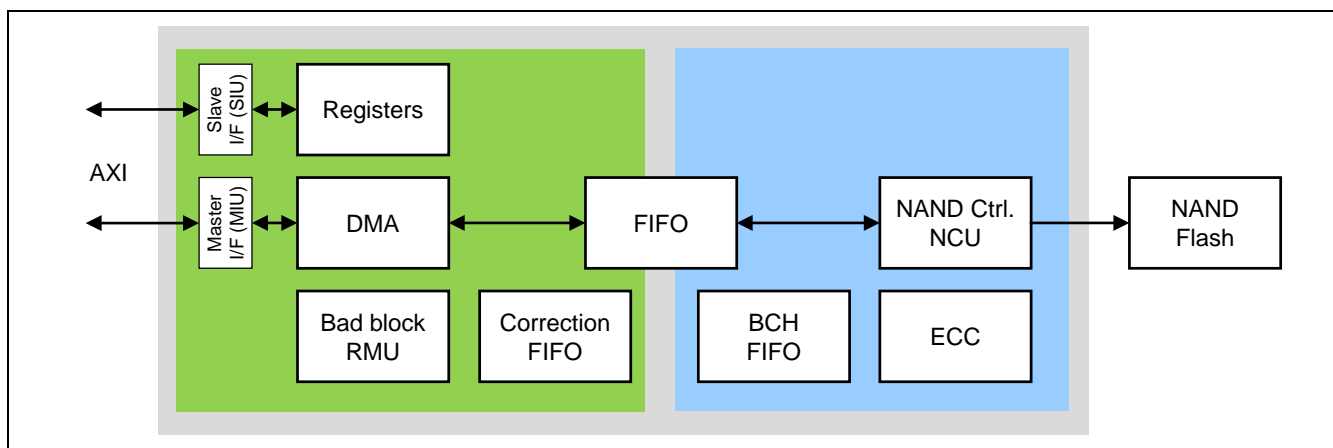


Figure 58.40 Block Diagram

The main components of the controller are:

- **FIFO** – This unit provides the FIFO queue interface to the other controller modules. Using the FIFO interface allows closing of the read and write pointers in the same module; thus, two modules can read and write in parallel. Depending on the controller software configuration, one side of the queue is the SIU or DMA module, the second one is always the NAND Control Unit (NCU).
- **DMA** – This unit is responsible for the fast transfer of data between the external memory location and the controller. The unit can work in two modes: the Scatter-Gather mode and the internally triggered single mode. DMA functions as the master side of the OCP socket. This block is optional and can be disabled during the compile process.
- **NAND control unit (NCU)** – This unit is responsible for the generation of the NAND Flash device access sequences.
- **ECC** – An error correction code calculator. A correction word is calculated for each 256B, 512B or 1024B ECC block of the NAND Flash memory page. During the read operation, the unit can automatically correct bad bits without any interaction with the CPU. It has a status register, the bits of which signal errors occurring during a read, and then it informs when errors are corrected.

The ECC module has an integrated FIFO that is used to transfer the calculated words to the NCU modules during the encode process and to store the calculated partial syndromes during the decode process. An additional FIFO module is used to transfer correction data between the correction module and the BCH-CHIEN module.

- **Bad block (RMU)** – It is a bad block remapping unit. This module remaps blocks excluded from the poll of the usable blocks into the blocks that will be its substitute. Whole process is transparent for the software.

58.4.2 DMA Module

(1) DMA Overview

The DMA engine integrated into the NAND Flash controller has the following properties:

- Two work modes:
 - The registers managed mode
 - The Scatter-Gather mode
- Supported burst types:
 - Imprecise bursts
 - Precise bursts
- Supported burst sequences:
 - The stream burst mode
 - The incrementing mode
- The DMA module preserves the configuration registers. These are copied into the DMA working registers.
- DMA trigger level enables DMA to send packet data.

(2) DMA Description

DMA_ADDR_L gives a 32-bit address when the DMA is configured as 32-bit.

DMA supports two modes:

- The registers managed mode – the DMA does one transfer depending on the contents of DMA_ADDR, DMA_CNT, and DMA_CTRL registers.
- The Scatter-Gather mode – new DMA transfer algorithm.

The DMA mode for the current transfer is selected via the DMA_CTRL.DMA_MODE bit and cannot be changed during each consecutive data transfer. The application program can change the DMA mode when the transfer is completed and the DMA_CTRL.DMA_READY bit is set.

In order to begin the transfer, the DMA should be in an IDLE state (DMA_CTRL.DMA_READY = 1).

Setting the DMA_CTRL.DMA_START bit triggers the DMA after the commands are sent to the NAND Flash memory (writing CONTROL register). When the DMA_START bit is clear, sending commands to the NAND Flash memory will not trigger the DMA. Because of this solution, it is possible to program the DMA to transfer more data than the page size of the NAND Flash memory (example: the DMA is configured to 32 kB data transfer, the size of the page in NAND Flash memory is 4 kB. In order to send 32 kB of data into the NAND Flash memory, it is necessary to set the DMA_START bit in the register DMA_CTRL, then it is necessary to send the PROGRAM PAGE command to the NAND Flash memory, clear the DMA_START bit in the DMA_CTRL register, and send seven commands to the NAND Flash memory). In this example, the DMA is set free only once at the beginning.

The DMA transfers four 32-bit words in one package when working in a 4-beat incrementing burst, eight words in an 8-beat incrementing burst, and sixteen words in a 16-beat incrementing burst. The burst size indicates the number of 32-bit words in the burst, not the number of bytes transferred.

The DMA trigger level enables the DMA to send packet data. It indicates when the DMA has to start transferring data.

The application program must set trigger level depending on arbitration type, system and NAND Flash memory performance. For example:

1. If the DMA trigger level equals 0, the DMA does not wait, but transfers data immediately because the trigger level is always exceeded.
2. For read from the NAND Flash memory: If the DMA trigger level equals H'40 (256 bytes), the DMA waits until data level reach the previously programmed trigger level (256 bytes). Then the DMA will transfer data until FIFO gets empty status. Next the wait for required data level cycle is repeated as long as remain data size allow to reach programmed trigger level. The trigger level value should be chosen to ensure that only the continuous data block will be transferred and the FIFO will not be overflowed.
3. For write to NAND Flash memory: If the DMA trigger level equals H'40 (256 bytes), the DMA writes data to the FIFO until the FIFO is full. Then, the DMA waits until 256 bytes are written in to the NAND Flash memory. Next, again the DMA transfers data until FIFO is full. The trigger level value should be chosen to ensure that only the continuous data block will be transferred.

(a) Registers Managed Mode:

When the DMA works in this mode, it transfers only one continuous block (DMA_CNT is block length) to or from the system memory at address DMA_ADDR. Block can be multiple of DATA_SIZE value in the DATA_SIZE register. All the registers which modify this transfer are described in 58.2.19 DMA base address - least significant part register (DMA_ADDR_L), 58.2.21 DMA counter initial value register (DMA_CNT) and 58.2.22 DMA control register (DMA_CTRL).

The DMA_BURST bit in register DMA_CTRL defines the main transfer type used by the DMA to precede the requested transfer.

(b) Scatter-Gather Mode:

The Scatter-Gather Direct Memory Access (SG-DMA) controller implements high-speed data transfer between two components, when DMA transfers and merges non-contiguous memory to a continuous address space, and vice versa.

The DMA in Scatter-Gather mode uses the Descriptors List to describe data transfers. The NAND Flash registers only points to the base address of the Descriptors List. The base address for this list is set in the DMA_ADDR register whether it is a read or write transfer. The base address, sizes of the data blocks, and flags are defined inside the descriptors.

The DMA_BURST bit in register DMA_CTRL defines the main transfer type used by the DMA to precede the requested transfer.

When in Scatter-Gather mode, the DMA transfers data from data blocks. Each descriptor can define only one data block.

The DMA adopts the scatter-gather DMA algorithm so that higher data transfer speed is available. The application program can program a list of data transfers between the system memory and NAND Flash to the Descriptor Table before executing Scatter-Gather mode.

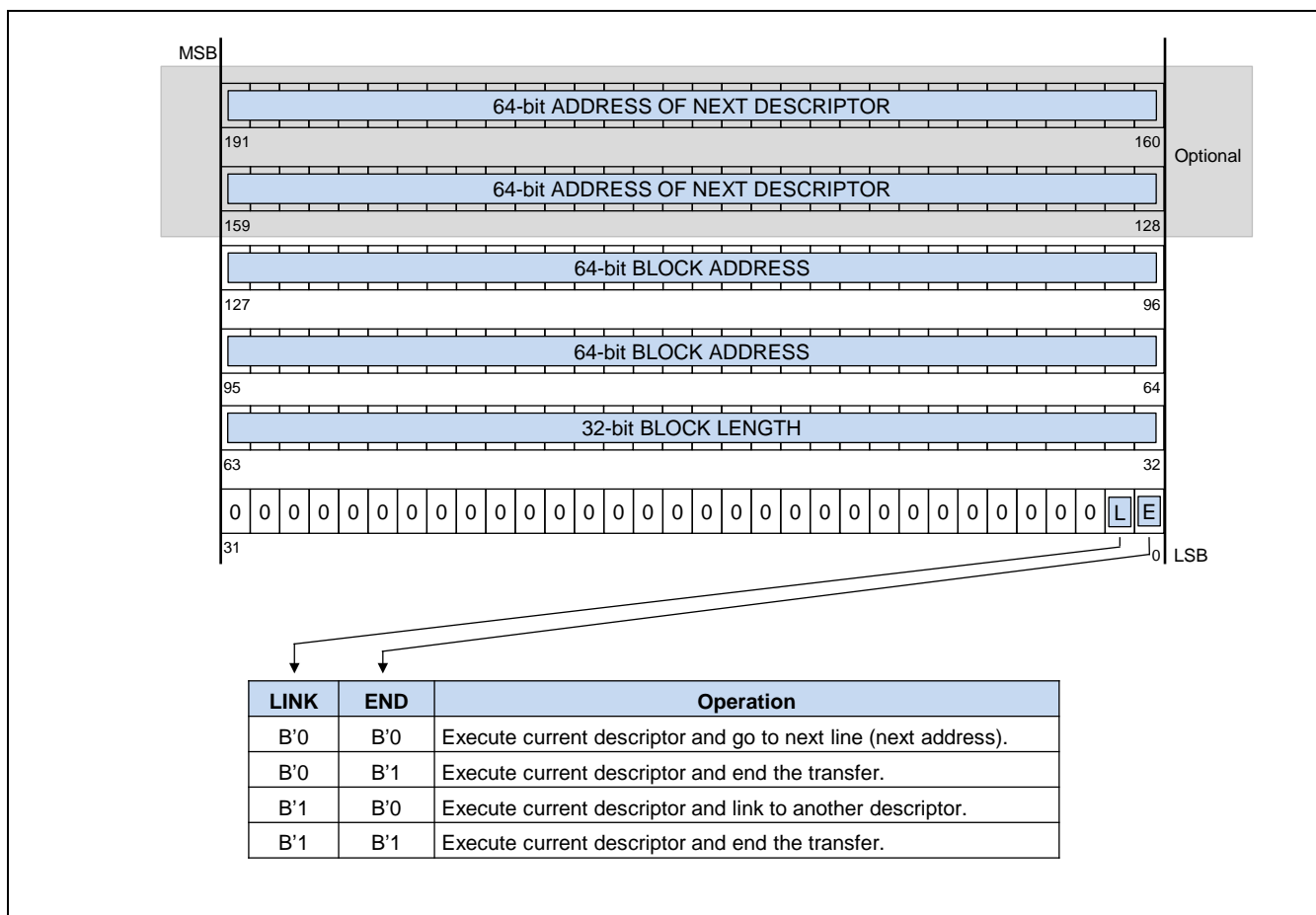


Figure 58.41 Scatter-Gather DMA Descriptor Layout (64-bit system memory addressing)

The table below defines the SG-DMA descriptor fields and their functions:

Table 58.48 Scatter-Gather DMA Descriptor Fields (64-bit system memory addressing)

| Bit | Symbol | Description |
|-----------|---------|--|
| [191:128] | ADDRESS | This optional field contains the address of the next descriptor list when LINK = 1 and END = 0. |
| [127:64] | ADDRESS | The field contains the data block address. |
| [63:32] | LENGTH | The field contains the data page length in bytes (H'0000_0004 - H'FFFF_FFFD). The number of bytes has to be divided by 4 (two least significant bits should equal 00). |
| [31:2] | — | Reserved for future use. |
| [1] | LINK | |
| [0] | END | |

| LINK | END | Comment |
|------|-----|---|
| B'0 | B'0 | Execute current descriptor and go to next descriptor on the list. |
| B'0 | B'1 | Execute current descriptor and end the transfer. |
| B'1 | B'0 | Execute current descriptor and link to another descriptor. |
| B'1 | B'1 | Execute current descriptor and end the transfer. |

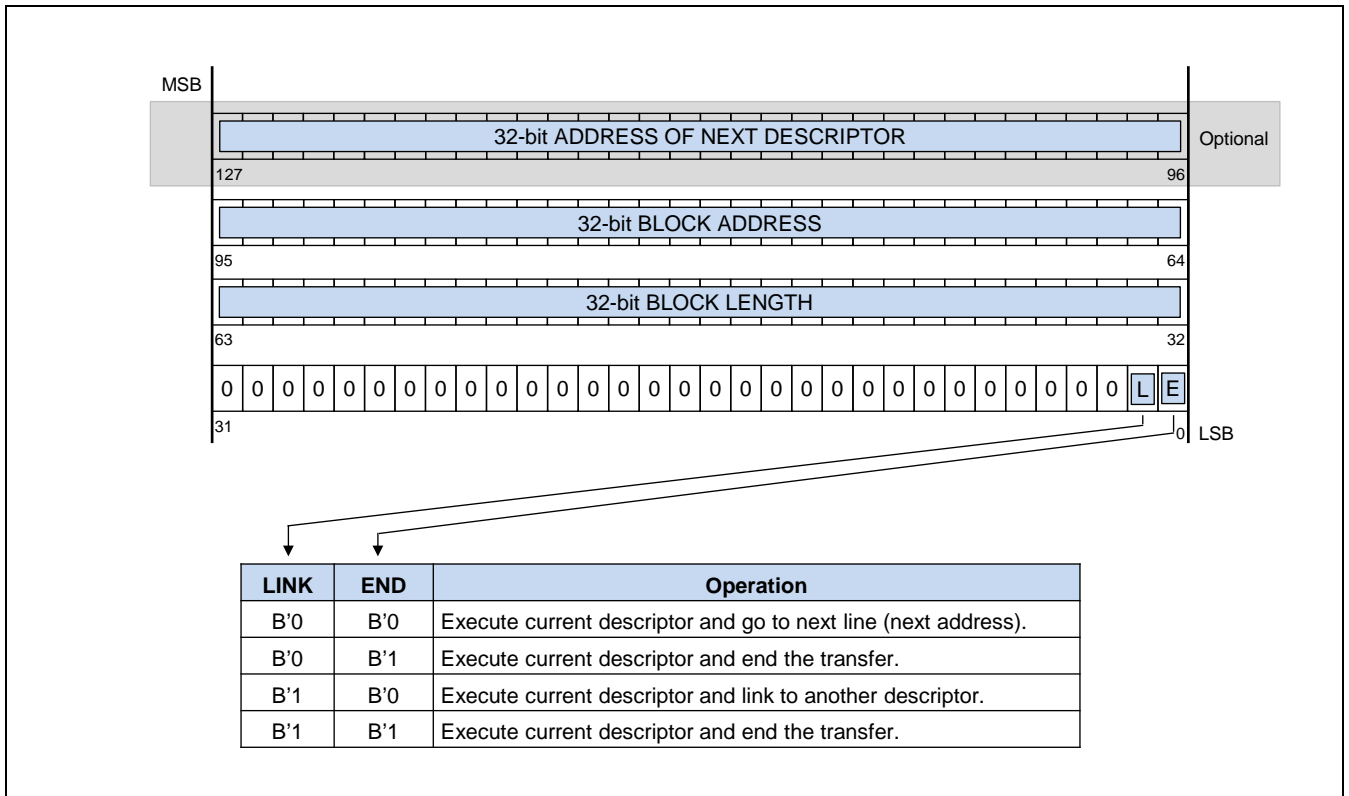


Figure 58.42 Scatter-Gather DMA Descriptor Fields (32-bit system memory addressing)

The table below defines the SG-DMA descriptor fields and their functions:

Table 58.49 Scatter-Gather DMA Descriptor Fields (32-bit system memory addressing)

| Bit | Symbol | Description | | | | | | | | | | | | | | | |
|----------|---------|---|------|-----|---------|-----|-----|---|-----|-----|--|-----|-----|--|-----|-----|--|
| [127:96] | ADDRESS | This optional field contains the address of the next descriptor lit when LINK = 1 and END = 0. | | | | | | | | | | | | | | | |
| [96:64] | ADDRESS | The field contains the data block address. | | | | | | | | | | | | | | | |
| [63:32] | LENGTH | The field contains the data page length in bytes (H'0000_0004 - H'FFFF_FFFD). The number of bytes has to be divided by 4 (two least significant bits should equal 00). | | | | | | | | | | | | | | | |
| [31:2] | — | Reserved for future use. | | | | | | | | | | | | | | | |
| [1] | LINK | | | | | | | | | | | | | | | | |
| | | <table border="1"> <thead> <tr> <th>LINK</th> <th>END</th> <th>Comment</th> </tr> </thead> <tbody> <tr> <td>B'0</td> <td>B'0</td> <td>Execute current descriptor and go to next descriptor on the list.</td> </tr> <tr> <td>B'0</td> <td>B'1</td> <td>Execute current descriptor and end the transfer.</td> </tr> <tr> <td>B'1</td> <td>B'0</td> <td>Execute current descriptor and link to another descriptor.</td> </tr> <tr> <td>B'1</td> <td>B'1</td> <td>Execute current descriptor and end the transfer.</td> </tr> </tbody> </table> | LINK | END | Comment | B'0 | B'0 | Execute current descriptor and go to next descriptor on the list. | B'0 | B'1 | Execute current descriptor and end the transfer. | B'1 | B'0 | Execute current descriptor and link to another descriptor. | B'1 | B'1 | Execute current descriptor and end the transfer. |
| LINK | END | Comment | | | | | | | | | | | | | | | |
| B'0 | B'0 | Execute current descriptor and go to next descriptor on the list. | | | | | | | | | | | | | | | |
| B'0 | B'1 | Execute current descriptor and end the transfer. | | | | | | | | | | | | | | | |
| B'1 | B'0 | Execute current descriptor and link to another descriptor. | | | | | | | | | | | | | | | |
| B'1 | B'1 | Execute current descriptor and end the transfer. | | | | | | | | | | | | | | | |
| [0] | END | | | | | | | | | | | | | | | | |

Table 58.50 Scatter-Gather DMA Descriptor Fields (32-bit system memory addressing)

| Length Field | Value of Length |
|--------------|-----------------|
| H'0000_0000 | Not Available |
| H'0000_0001 | Not Available |
| H'0000_0002 | Not Available |
| H'0000_0003 | Not Available |
| H'0000_0004 | 4 bytes |
| H'0000_0005 | Not Available |
| H'0000_0006 | Not Available |
| H'0000_0007 | Not Available |
| H'0000_0008 | 8 bytes |
| | |

The Descriptor Table is created in system memory by the application program. SG-DMA can support only 32-bit and 64-bit system memory addressing. Each descriptor line (one executable unit) consists of the address, length and flags. The flags specify operation of the descriptor line.

The figure below shows the example of Scatter-Gather programming:

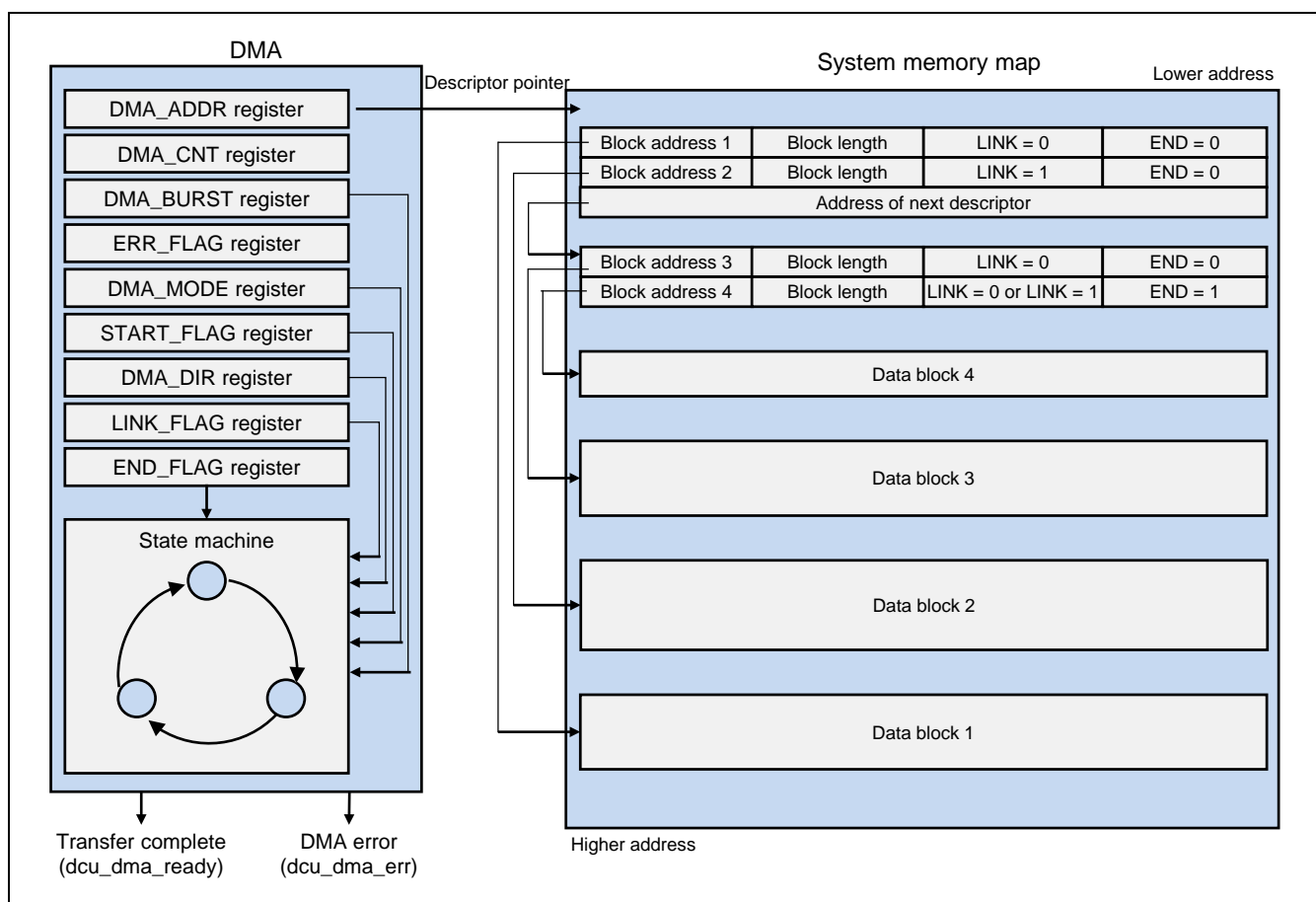


Figure 58.43 Example of Scatter-Gather DMA Data Transfer

58.4.3 ECC Module

(1) Module Overview

The ECC module is based on one of the BCH algorithms and allows correction of multiple bit errors.

The ECC engine integrated into the NAND Flash controller has the following properties:

- The encoder and decoder works on the 256, 512, 1024 bytes data blocks.
- Programmable correction capability: 2, 4, 8, 16, 24 or 32 errors.
- The corrected data words are aligned to the 32 bits.
- The correction words are aligned to the 32 bits.
- Correction words are placed after the data.

(2) Block Diagram

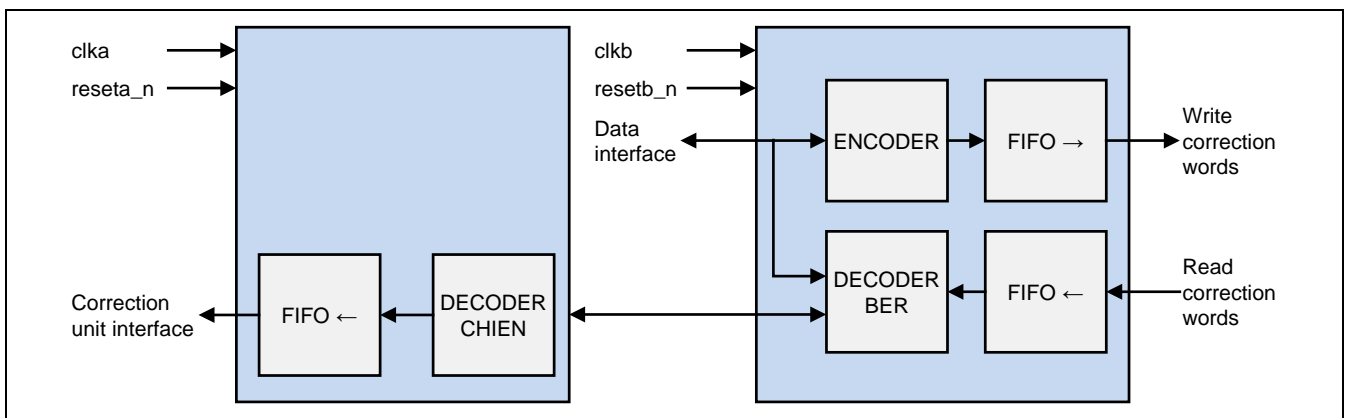


Figure 58.44 ECC Module Block Diagram

58.4.4 BCH Algorithm Implementation

- Data block length:
- Programmable memory page length, multiple of the block length by any power of 2 (optional).
- Programmable correction capability: 2, 4, 8, 16, 24 or 32 errors.
- Separate encoder and decoder modules.
- Calculation of correction data performed during write to memory.
- Error detection performed during read data from memory.
- Internal pipeline allows the correction of errors in one data block simultaneously with detection of errors in the following data block.

Table 58.51 Size of Correction Bytes

| Corrcion Capability | ECC Block Size | Size of Correction Bytes per one ECC Block(IO_WIDTH = 8) | Size of Correction Bytes per one ECC Block(IO_WIDTH = 16) |
|---------------------|----------------|--|---|
| 2 | 256/512/1024 B | 4 | 4 |
| 4 | 256/512/1024 B | 7 | 8 |
| 8 | 256/512/1024 B | 14 | 14 |
| 16 | 256/512/1024 B | 28 | 28 |
| 24 | 256/512/1024 B | 42 | 42 |
| 32 | 256/512/1024 B | 56 | 56 |

58.5 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

58.5.1 Initial Setting

(1) Clock Supply

The clock signal needs to be supplied before R-NANDC is used. Figure 58.45 shows the example of initial setting.

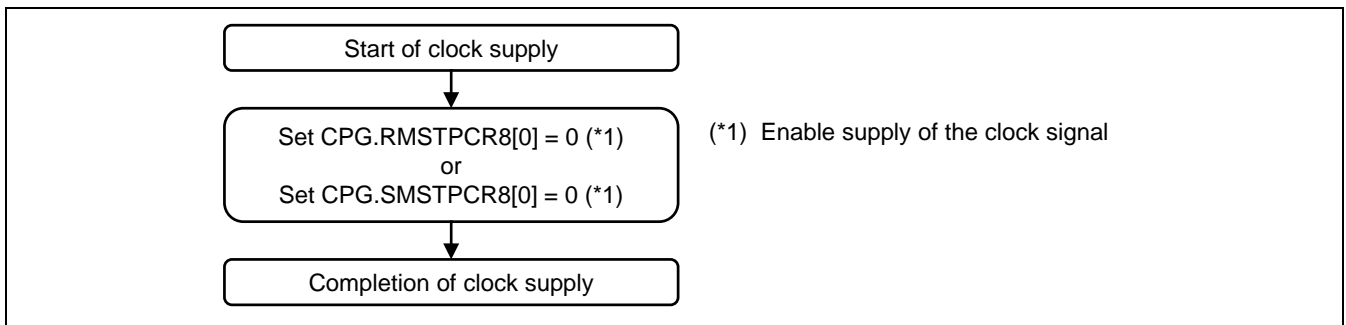


Figure 58.45 Clock Supply Flowchart [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] [RZ/G2E]

58.5.2 Clock Stop

Refer to Figure 58.46 Clock Stop Flowchart when it is necessary to stop the clock signal.

Also, refer to Figure 58.45 Clock Supply Flowchart when restarting clock supply.

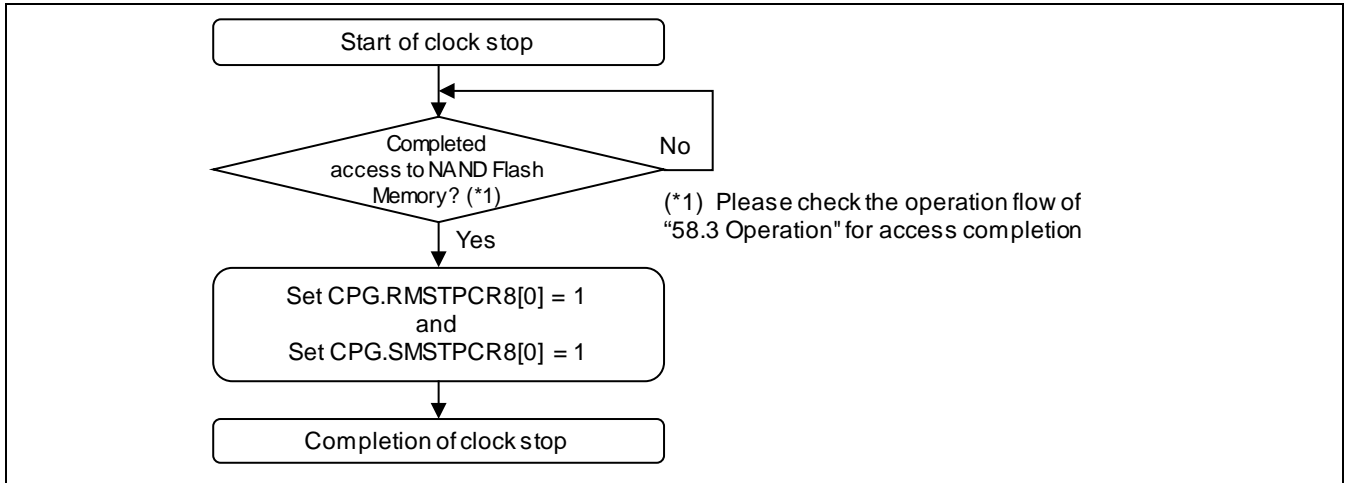
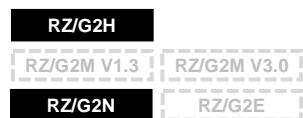


Figure 58.46 Clock Stop Flowchart [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N] [RZ/G2E]

59. Serial-ATA Gen3



59.1 Overview

The serial-ATA interface provides a serial ATA physical interface which complies with the serial-ATA standard.

59.1.1 Features

- Supports the following transfer rates: 6.0 Gbps, 3.0 Gbps, and 1.5 Gbps (respectively for the third, second, and first generation of the SATA interface standard)
- Parallel-ATA emulation
- Packet protocol transfer
- Supports SATA standard 8b/10b encoding, cyclic redundancy check (CRC), and scrambling modes.
- Supports the SATA power management mode (partial mode/slumber mode).
- Does not support reset speed negotiation (RSN).
- PHY is shared with PCIE Controller. SATA is supported with MD12 (mode pin) = 1.

59.1.2 Block Diagram

Figure 59.1 shows the block diagram of the SATA interfaces.

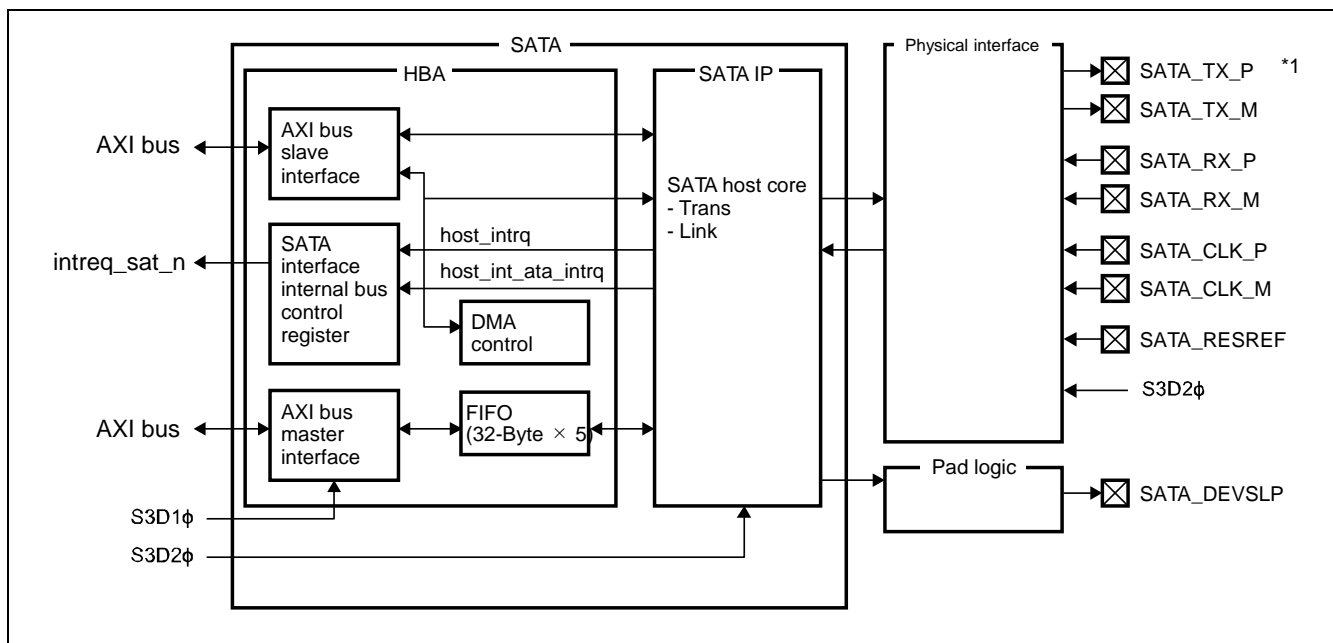


Figure 59.1 Block Diagram of SATA Interfaces

host_intrq and host_int_ata_intrq are internal interrupt request signals. For the interrupt sources of these signals, refer to the description of the SATA INT status register.

Note: 1. Refer Table 59.1.

59.1.3 External Pins

Table 59.1 shows the external pins of the SATA interface.

Table 59.1 External Pins

| | | | | | Second Generation RZ/G Series Products | | | |
|-------------------------------------|--|----------------|------------------|--|---|----------------------------|--------|--------|
| Name | Pin Name | SATA Symbol | I/O | Description | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Reference clock | PCIE1_CLK_P (SATA_CLK_P) PCIE1_CLK_M (SATA_CLK_M) | — | Input | Reference clock input to the PLL circuit in the Serial-ATA module (differential input). Apply a 100-MHz clock. | √ | — | √ | — |
| Transmit data | PCIE1_TX_P (SATA_TX_P) PCIE1_TX_M (SATA_TX_M) | TX+ TX- | Output | Pins for data transmission pins. A 6.0-GHz (third generation) signal or 3.0-GHz (second generation) signal or 1.5-GHz (first generation) signal is transmitted through these pins. The pins with names ending in P and M are combined to provide a differential signal. | √ | — | √ | — |
| Receive data | PCIE1_RX_P (SATA_RX_P) PCIE1_RX_M (SATA_RX_M) | RX+ RX- | Input | Pins for data reception. A 1.5-GHz (first generation) or 3.0-GHz (second generation) signal or 6.0-GHz (third generation) signal is received through these pins. The pins with names ending in P and M are combined to provide a differential signal. | √ | — | √ | — |
| Reference resistor connection | PCIE1_RESREF (SATA_RESREF) | — | Input/ Output | Attach a 200-Ω ±1% ±100 ppm/°C precision resistor-to-ground on the board. | √ | — | √ | — |
| Device Sleep | SATA_DEVSLP | DEVSL P | Output | Low power consumption mode of device(HDD,SSD) | √ | — | √ | — |

59.1.4 Register Configuration

Table 59.2 shows the register configuration of the SATA interface. The set of registers shown below are allocated to the register map space.

The area allocated for the SATA interface is from H'EE30_0000 to H'EE4F_FFFF (2-Mbyte space).

These registers must not be accessed in any access sizes other than those listed in the table.

Addresses other than those listed below must not be write-accessed to. If written to, operation is not guaranteed. If read, an undefined value is read.

Table 59.2 Register Configuration

Add the offsets under “address” below to H'EE30_0000.

| Name | Symbol | R/W | Address | Initial Value | Access Size (Available Bit Size) | Second Generation RZ/G Series Products | | | |
|--|------------------|-------|---------|---------------|----------------------------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Serial-ATA HOST control registers | | | | | | | | | |
| BIST config register | BISTCONF | R/W | H'102C | H'0000_0000 | 32 | √ | — | √ | — |
| Shadow data register | SDATA | R/W | H'1100 | H'0000_FFFF | 32 | √ | — | √ | — |
| Shadow error register | SSERR | R | H'1104 | H'0000_00FF | 32 | √ | — | √ | — |
| Shadow features register | SSFEATURES | W | H'1104 | H'0000_00FF | 32 | √ | — | √ | — |
| Shadow sector CNT register | SSECCNT | R/W | H'1108 | H'0000_00FF | 32 | √ | — | √ | — |
| Shadow LBA low register | SLBALOW | R/W | H'110C | H'0000_00FF | 32 | √ | — | √ | — |
| Shadow LBA mid register | SLBAMID | R/W | H'1110 | H'0000_00FF | 32 | √ | — | √ | — |
| Shadow LBA high register | SLBAHIGH | R/W | H'1114 | H'0000_00FF | 32 | √ | — | √ | — |
| Shadow device/head register | SDEVHEAD | R/W | H'1118 | H'0000_00FF | 32 | √ | — | √ | — |
| Shadow status register | SSSTATUS | R | H'111C | H'0000_007F | 32 | √ | — | √ | — |
| Shadow command register | SSCOM | W | H'111C | H'0000_00FF | 32 | √ | — | √ | — |
| Shadow alternate status register | SSALTSTS | R | H'1204 | H'0000_007F | 32 | √ | — | √ | — |
| Shadow device control register | SSDEVCON | W | H'1204 | H'0000_0000 | 32 | √ | — | √ | — |
| SATA extend ICC register | SATAEICCR | R | H'1220 | H'0000_0000 | 32 | √ | — | √ | — |
| SATA extend auxiliary register | SATAEAUXR | R | H'1224 | H'0000_0000 | 32 | √ | — | √ | — |
| SATA extend DEVSLP register | SATAEDEVSLP R | R/W | H'1228 | H'0000_0000 | 32 | √ | — | √ | — |
| SCR Sstatus register | SCRSSSTS | R | H'1400 | H'0000_0000 | 32 | √ | — | √ | — |
| SCR Serror register | SCRSEERR | R/WC1 | H'1404 | H'0000_0000 | 32 | √ | — | √ | — |
| SCR Scontrol register | SCRSCON | R/W | H'1408 | H'0000_0000 | 32 | √ | — | √ | — |
| SCR Sactive register | SCRSACT | R/W | H'140C | H'0000_0000 | 32 | √ | — | √ | — |
| SATA INT status register | SATAINTSTAT | RC | H'1508 | H'7F00_0000 | 32 | √ | — | √ | — |
| SATA INT mask register | SATAINTMASK | R/W | H'150C | H'0000_09FC | 32 | √ | — | √ | — |
| PHY STOP Register | PHYSTOP | R/W | H'1568 | H'0000_0000 | 32 | √ | — | √ | — |

| | | | | | | Second Generation RZ/G Series Products | | | |
|--------------------------------------|--------|-----|---------|---------------|---|---|----------------------------|--------|--------|
| Name | Symbol | R/W | Address | Initial Value | Access Size (Available Bit Size) | | | | |
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Rx DMA setup FIS dword 0 register | DMADW0 | R/W | H'1620 | H'0000_0000 | 32 | √ | — | √ | — |
| Rx DMA setup FIS dword 1 register | DMADW1 | R/W | H'1624 | H'0000_0000 | 32 | √ | — | √ | — |
| Rx DMA setup FIS dword 2 register | DMADW2 | R/W | H'1628 | H'0000_0000 | 32 | √ | — | √ | — |
| Rx DMA setup FIS dword 3 register | DMADW3 | R/W | H'162C | H'0000_0000 | 32 | √ | — | √ | — |
| Rx DMA setup FIS dword 4 register | DMADW4 | R/W | H'1630 | H'0000_0000 | 32 | √ | — | √ | — |
| Rx DMA setup FIS dword 5 register | DMADW5 | R/W | H'1634 | H'0000_0000 | 32 | √ | — | √ | — |
| Rx DMA setup FIS dword 6 register | DMADW6 | R/W | H'1638 | H'0000_0000 | 32 | √ | — | √ | — |

Notes: See the descriptions of individual registers for the R/W attributes of the valid bits.

1. Bits 15 to 0 of the data bus are used.
2. Bits 7 to 0 of the data bus are used.

59.1.5 Connected Module

Table 59.3 Connected modules

| Module name | Connected module name | Function of connected module |
|--------------------|------------------------------|-------------------------------------|
| SATA | AP-System Core | Access the Registers |
| | CPG | Output clocks |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |
| | INTC-AP | Control to interrupt |

59.2 Register Description

- Legend:

| | |
|----------------|---|
| Address: | Address of the register |
| Bit map: | Bit configuration of the entire register. The initial value and R/W of the individual bits are also indicated. |
| Bit: | Bit number and range |
| Bit name: | Bit name or field name |
| Initial value: | Register value after a reset |
| —: | Undefined value |
| R/W: | Readable/writable. The written value can be read. |
| R/WC0: | Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored. |
| R/WC1: | Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored. |
| R: | Read-only. The write value should always be 0. (If a read/write value is specified in the bit description, however, it must be observed.) |
| —/W: | Write-only. The read value is undefined. |
| RC: | Read-only. The bit is cleared to 0 after it has been read. |

59.2.1 Note on the Access to the Registers Allocated to H'1018 to H'10A8

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | — | √ | — |

No normal access can basically be made to the registers that are allocated to H'1018 to H'10A8 unless the PHYRDY state is established. The PHYRDY state here refers to the state in which the rate of transfer between the host and device is established and they are ready for data transfer. In this state, bits 3-0 of the SCR SStatus register described in section 59.2.27, SCR SStatus Register (SCRSSSTS), is set to B'0011*. The integrity of the value of a register that is accessed in the non-PHYRDY state is not guaranteed.

59.2.2 ATAPI Control Register (ATAPI_CONTROL1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|------------|----|----|----|-------|----|----|----|------|-----|------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ISM |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | DTA32 M | — | — | — | RESET | — | — | — | DESE | R/W | STOP | START |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R | R | R | R/W | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 17 | — | — | R | Reserved The write value should always be 0. |
| 16 | ISM | B'0 | R/W | Interrupt Status mode 0: IP compatible mode Normal operating mode of the SATA INT status register (SATAINTSTAT). 1: CIS interrupt mode A valid interrupt source specified by the SATA INT mask register is set in the SATA INT status register (SATAINTSTAT). No interrupts are generated by valid sources specified in the SATA INT mask register (SATAINTMASK). Read access to the SATA INT status register (SATAINTSTAT) will not negate the ATA-sourced DEVINT interrupt signal. Note: For details, refer to section 59.3.6, Interrupt Modes. Set ISM to 1 while the SATA-IP host core block is being reset (bit 7 (RESET) is 1). Do not change the value of ISM after the SATA-IP host core block has been released from the reset state (bit 7 (RESET) is 0). |
| 15 to 12 | — | All 0 | R | This bit is fixed to 0 and cannot be set to 1. |
| 11 | DTA32M | B'0 | R/W | DTA32M enables bits 31 to 29 of the descriptor DMA start address in descriptor table operation mode. This causes the termination flag to be changed from bit 31 to bit 0. 0: Reserved 1: Enabled bits of the descriptor DMA start address are bits 31 to 2. |
| 10 to 8 | — | All 0 | R | These bits are fixed to 0 and cannot be set to 1. |
| 7 | RESET | B'0 | R/W | RESET controls the SATA-IP host core block. Setting this bit to 1 resets the SATA-IP host core block. |
| 6 to 4 | — | All 1 | R | These bits are fixed to 1. Attempting to set these bits to 0 is ignored and not reflected in the value of the bits. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | DESE | B'0 | R/W | DESE controls the descriptor table operation mode. 0: Descriptor functions disabled 1: Descriptor functions enabled |
| 2 | R/W | B'0 | R/W | R/W controls FIFO read/write. 0: FIFO write (data-out operation at DMA transfer) 1: FIFO read (data-in operation at DMA transfer) Set this bit to 1 when reading data from the SATA device. Clear it to 0 when writing data to the SATA device. |
| 1 | STOP | B'0 | R/W | This bit forces termination of DMA transfer. When writing 0: Ignored 1: Forcibly terminates data transfer When reading 0: Forced termination command is not issued. 1: Forced termination of data transfer command is issued. This bit is cleared to 0 when the next DMA starts. To power down (SATA internal clock turns off) while DMA is active (ACT = 1), DMA transfer should be forcibly terminated. Note: Transfer cannot always be resumed from the address at which DMA transfer has been forcibly terminated. |
| 0 | START | B'0 | R/W | This bit initiates DMA transfer. If this bit set to 1 then the DMA transfer is started. When cleared to 0, this bit is ignored. When writing 0: Ignored 1: Starts DMA transfer When reading 0: DMA transfer is not active 1: DMA transfer is in busy state Note: Access to the task file register is prohibited while DMA is active. |

59.2.3 ATAPI Status Register (ATAPI_STATUS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|-------|--------|----|----|---------|----|----|-------|----|-------|--------|--------|----|-------|-------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SSBSY | SSDRDY | — | — | SSDRQ | — | — | SSERR | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | — | — | 0 | — | — | 0 | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | SATAINT | — | — | SWERR | — | DNEND | DEVTRM | DEVINT | — | ERR | NEND | ACT |
| Initial value: | — | — | — | — | 0 | — | — | 0 | — | 0 | 0 | 0 | — | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/WC0 | R | R/WC0 | R/WC0 | R | R | R/WC0 | R/WC0 | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 | SSBSY | B'0 | R | SSBSY corresponds to INTBSY of the SATA INT status register (SATAINTSTAT), the BSY bit value of the shadow status register (SSTATUS). This bit is enabled when ISM = 1 and disabled (0) when ISM = 0. The write value should always be 0. |
| 30 | SSDRDY | B'0 | R | SSDRDY corresponds to INTDRDY of the SATA INT status register (SATAINTSTAT), the DRDY bit value of the shadow status register (SSTATUS). This bit is enabled when ISM = 1 and disabled (0) when ISM = 0. The write value should always be 0. |
| 29, 28 | — | — | R | Reserved The write value should always be 0. |
| 27 | SSDRQ | B'0 | R | SSDRQ corresponds to INTDRQ of the SATA INT status register (SATAINTSTAT), the DRQ bit value of the shadow status register (SSTATUS). This bit is enabled when ISM = 1 and disabled (0) when ISM = 0. The write value should always be 0. |
| 26, 25 | — | — | R | Reserved The write value should always be 0. |
| 24 | SSERR | B'0 | R | SSER corresponds to INTERR of the SATA INT status register (SATAINTSTAT), the ERR bit value of the shadow status register (SSTATUS). This bit is enabled when ISM = 1 and disabled (0) when ISM = 0. The write value should always be 0. |
| 23 to 12 | — | — | R | Reserved The write value should always be 0. |
| 11 | SATAINT | B'0 | R | SATAINT indicates the status of host_intrq of the SATA-IP block. This bit is read-only. host_intrq is the internal interrupt request signal. For the interrupt source of this signal, refer to the description of the SATA INT Status register. The write value should always be 0. |
| 10, 9 | — | — | R | Reserved The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 8 | SWERR | B'0 | R/WC0 | Software error bit. 1 in this bit indicates that the task file register is accessed when the DMA is active. This bit is set to 1 when a PIO transfer is performed during DMA transfer. In this case, no output is directed to the SATA-IP block and the access is ignored. Writing 0 resets this bit. |
| 7 | — | — | R | Reserved The write value should always be 0. |
| 6 | DNEND | B'0 | R/WC0 | DNEND indicates that all DMAs have successfully ended in the descriptor mode. Writing 0 resets this bit. |
| 5 | DEVTRM | B'0 | R/WC0 | DEVTRM is set to 1 when the DMA mode for the SATA_IP block is terminated before the number of DMA transfer bytes defined in the DMA transfer count register (ATAPI_DMA_TRANS_CNT) is reached (corresponds to P-ATA device's device termination operation). Writing 0 resets this bit. |
| 4 | DEVINT | B'0 | R | DEVINT indicates the status of host_int_ata_intrq in the SATA-IP block (host_int_ata_intrq signal corresponds to the IDEINT signal of the P-ATA device and ATA of the SATA INT status register (SATAINTSTAT)). This bit is read-only. Since this LSI chip preserves no status, this bit is cleared to 0 when host_int_ata_intrq is set to 0. The SATA_HBA block regards any interrupt signal from the SATA-IP block as a level trigger input. host_int_ata_intrq is negated by the SATA device to clear the interrupt pending state after the shadow status register (SSTATUS) is read out. There is no time specification. The write value should always be 0. |
| 3 | — | — | R | Reserved The write value should always be 0. |
| 2 | ERR | B'0 | R/WC0 | ERR is set to 1 when the host forcibly terminated the DMA transfer. Writing 0 resets this bit. |
| 1 | NEND | B'0 | R/WC0 | NEND indicates that DMA has been successfully terminated. Writing 0 resets this bit. |
| 0 | ACT | B'0 | R | ACT indicates that DMA is active. To power down (SATA internal clock turns off) while DMA is active (ACT = 1), DMA transfer should be forcibly terminated (STOP = 1). The write value should always be 0. |

59.2.4 Interrupt Enable Register (ATAPI_INT_ENABLE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

When 1 is written to a bit, the interrupt signal corresponding to the bit in the ATAPI status register (ATAPI_STATUS) is enabled.

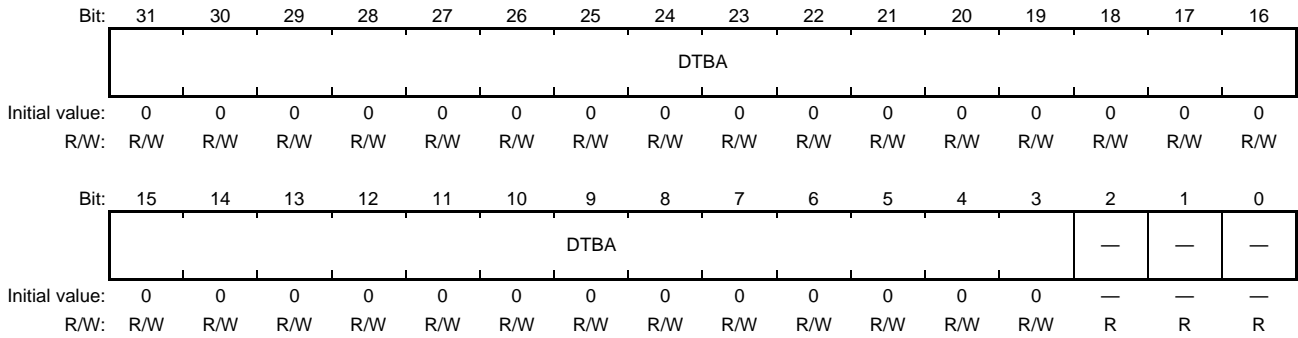
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|--------------|----|----|------------|----|------------|-------------|-------------|----|------|-------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | iSATAI NT | — | — | iSWER R | — | iDNEN D | iDEV TRM | iDEVIN T | — | iERR | iNEND | iACT |
| Initial value: | — | — | — | — | 0 | — | — | 0 | — | 0 | 0 | 0 | — | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R | R | R/W | R | R/W | R/W | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 12 | — | — | R | Reserved The write value should always be 0. |
| 11 | iSATAINT | B'0 | R/W | SATAINT interrupt enable |
| 10, 9 | — | — | R | Reserved The write value should always be 0. |
| 8 | iSWERR | B'0 | R/W | SWERR interrupt enable |
| 7 | — | — | R | Reserved The write value should always be 0. |
| 6 | iDNEND | B'0 | R/W | DNEND interrupt enable |
| 5 | iDEVTRM | B'0 | R/W | DEVTRM interrupt enable |
| 4 | iDEVINT | B'0 | R/W | DEVINT interrupt enable |
| 3 | — | — | R | Reserved The write value should always be 0. |
| 2 | iERR | B'0 | R/W | ERR interrupt enable |
| 1 | iNEND | B'0 | R/W | NEND interrupt enable |
| 0 | iACT | B'0 | R/W | ACT interrupt enable bit. Since ACT is cleared automatically when a DMA transfer is completed, interrupt processing should be completed during assertion. |

Note: Writing 1 to a bit enables the interrupt signal corresponding to the bit in the ATAPI status register. Do not clear the interrupt mask of the INTC while the SATA module is being reset. For resetting of the SATA module, refer to the descriptions of the CPG module. For the interrupt mask of INTC, refer to the description in the section on the INTC module.

59.2.5 Descriptor Table Base Address Register (ATAPI_DTB_ADR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|--|-----|--|
| 31 to 3 | DTBA | B'0_0000_0000_0000_0000_0000_0000_0000 | R/W | These bits indicate the descriptor table base address. Bits 31 to 0 are used to set the descriptor table base address. |
| 2 to 0 | — | — | R | Reserved The write value should always be 0. |

Note: This address does not change and the set value is retained even after the DMA becomes active.

(1) Descriptor Table

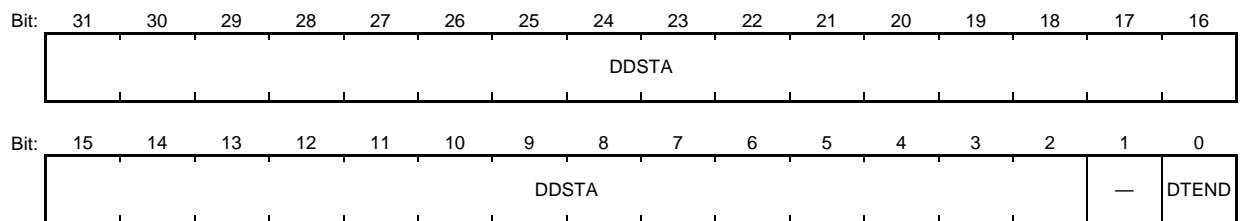
The descriptor table consists of the termination flag, the descriptor DMA start address (DDSTA), and the descriptor DMA transfer count (DDTRC).

(a) When Bits 31 to 29 of Descriptor DMA Start Address are Valid (DTA32M = 1)

Table 59.4 Descriptor Table Map in Memory

| Address | Data Description |
|-----------------------|---|
| DTBA | The first termination flag (bit 0 = 0) and DDSTA |
| DTBA + 4 | The first DDTRC |
| DTBA + 8 | The second termination flag (bit 0 = 0) and DDSTA |
| DTBA + 12 | The second DDTRC |
| ... | ... |
| DTBA + 8* (n - 1) | The n-th termination flag (bit 0 = 1) and DDSTA |
| DTBA + 8* (n - 1) + 4 | The n-th DDTRC |

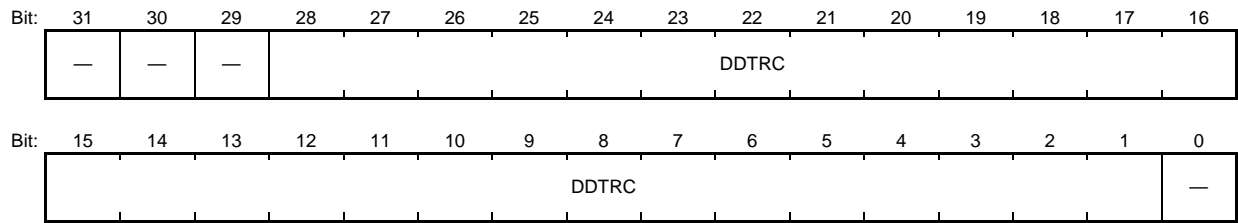
• Termination Flag and Descriptor DMA Start Address



| Bit | Bit Name | Description |
|---------|----------|---|
| 31 to 2 | DDSTA | DDSTA shows the DMA start address in descriptor operation. Bits 31 to 0 are used to set the descriptor table start address on a byte basis. Bits 1 and 0 are ignored because it is necessary to secure the boundary of 32-bit addresses in the DMA start address. |
| 1 | — | Reserved The write value should always be 0. |
| 0 | DTEND | DTEND controls the termination of a descriptor DMA operation. 0: Validates the descriptor table When DTEND is 0, the DMA transfer count is read, DMA transfer is performed, and the next descriptor table is read. 1: Terminates the descriptor DMA operation When DTEND is 1, the last descriptor table is detected. |

The valid flag and descriptor DMA start address should be set in the descriptor table base address + "m" in the memory, where m is multiple of 2 (such as 0, 2, 4 ...).

• Descriptor DMA Transfer Count

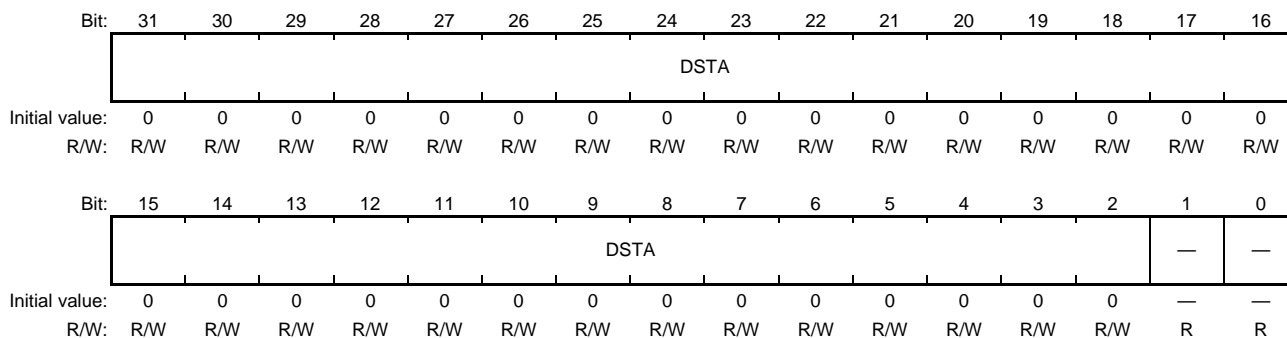


| Bit | Bit Name | R/W | Description |
|----------|----------|-----|--|
| 31 to 29 | — | R | Reserved The write value should always be 0. |
| 28 to 1 | DDTRC | R/W | These bits set the DMA transfer count during descriptor operation. Bits 28 to 0 are used to set the DMA transfer count on a byte basis. Bit 0 is ignored because the SATA data bus is handled on a 16-bit basis (on a word basis). |
| 0 | — | R | Reserved The write value should always be 0. |

The descriptor DMA transfer count should be set in the descriptor table base address + "m" in the memory, where the value of m is any multiple number of 2 plus 1 (such as 1, 3, 5,...).

59.2.6 DMA Start Address Register (ATAPI_DMA_START_ADR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

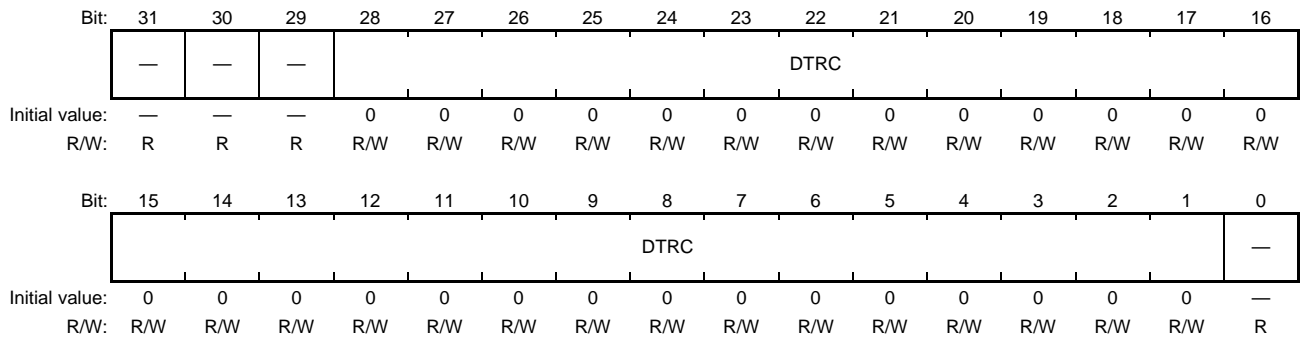


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---|-----|---|
| 31 to 2 | DSTA | B'00_0000_0000_0000_0000_0000_0000_0000 | R/W | DSTA sets a DMA start address that indicates the data transfer start address in the memory. Bits 31 to 0 are used to set the DMA start address on a byte basis. Bits 1 and 0 are ignored because it is necessary to secure the boundary of 32-bit addresses in the DMA start address. |
| 1, 0 | — | — | R | Reserved The write value should always be 0. |

Note: This address does not change and the set value is retained even after the DMA becomes active.

59.2.7 DMA Transfer Count Register (ATAPI_DMA_TRANS_CNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 29 | — | — | R | Reserved The write value should always be 0. |
| 28 to 1 | DTRC | H'000_0000 | R/W | DTRC sets the DMA transfer count. Bits 28 to 0 are used to set the DMA transfer count on a byte basis. Bit 0 is ignored because the ATAPI data bus is handled on a 16-bit basis (on a word basis). |
| 0 | — | — | R | Reserved The write value should always be 0. |

Note: This count value does not change and the set value is retained even after the DMA becomes active.

59.2.8 ATAPI Control 2 Register (ATAPI_CONTROL2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|------------|-----------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | LWORD SWAP | WORD SWAP | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | 0 | 0 | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 3 | — | — | R | Reserved The write value should always be 0. |
| 2 | LWORDSWAP | B'0 | R/W | LWORDSWAP controls the swapping of the upper 32-bit data and the lower 32-bit data on a 2-longword basis (64-bit data) on the AXI bus. 0: Longword swap is not executed. The 64-bit data on the AXI bus appears in a big endian format. 1: Longword swap is executed between the SATA interface and the AXI bus interface. Note that longword swap is only available on data transfer when bit 0 (START) in the ATAPI control register (ATAPI_CONTROL1) = 1: DMA mode start. |
| 1 | WORDSWAP | B'0 | R/W | WORDSWAP controls the swapping of the upper 16-bit data and the lower 16-bit data on a longword basis when the 64-bit data bus is enabled in the AXI bus. 0: Word swap is not executed. 1: Word swap is executed between the SATA interface and the AXI bus interface. Note that word swap is only available on data transfer when bit 0 (START) in the ATAPI control register (ATAPI_CONTROL1) = 1: DMA mode start. |
| 0 | — | — | R | Reserved The write value should always be 0. |

59.2.9 ATAPI Signal Status Register (ATAPI_SIG_ST)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------------|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | DPIOR DY | DMAR Q |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 2 | — | — | R | Reserved The write value should always be 0. |
| 1 | DPIORDY | — | R | DPIORDY indicates the state of the IF signal for PIO transfer in the SATA_IP block. |
| 0 | DMARQ | — | R | DMARQ indicates ATAPIDMARQ (IDEDREQ) signal state. |

Note: Since the SATA interface has separate PIO and DMA interfaces, the DDMARDY signal of conventional ATAPI cannot be monitored.
The IORDY signal is not emulated. DPIORDY monitors the PIO-transfer interface signal only. Monitoring of the DMA-transfer interface signals is not possible.

59.2.10 Byte Swap Register (ATAPI_BYTE_SWAP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | BYTE SWAP |
| Initial value: | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | — | R | Reserved The write value should always be 0. |
| 0 | BYTESWAP | B'0 | R/W | BYTESWAP controls the swapping of the upper 8-bit data and the lower 8-bit data in the SATA interface. 0: Byte swap is not executed between the SATA interface and the AXI bus. 1: Byte swap is executed between the SATA interface and the AXI bus. Note that byte swap is only available on data transfer when START in the ATAPI control register (ATAPI_CONTROL1) = 1: DMA mode start. |

59.2.11 BIST Config Register (BISTCONF)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The BISTCONF register is used to configure the operation to be performed in BIST mode.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | LOOPB |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

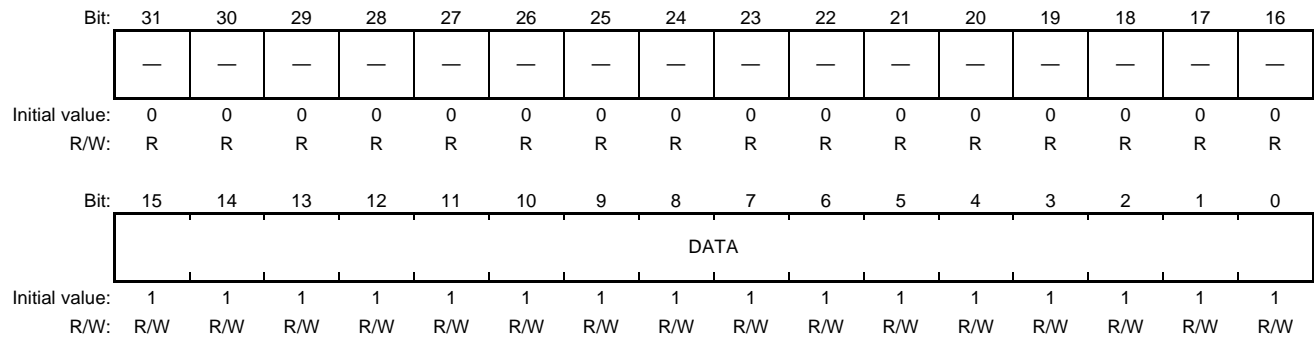
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | LOOPB | B'0 | R/W | Loopback mode bit (Loopback Mode) 0: Loopback (for far-end retimed loopback mode) is not formed in the link layer of the host. 1: Loopback (for far-end retimed loopback mode) is formed in the link layer of the host. |

Note: No read access can be made to this register and no correct data can be read from this register until the PHYRDY state is established. (See section 59.2.1, Note on the Access to the Registers Allocated to H'1018 to H'10A8.)

59.2.12 Shadow Data Register (DATA) (SDATA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The Shadow Data Register corresponds to the ATA standard's Data Register/Port. This is a data register that is read and written in PIO transfer mode.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | DATA | H'FFFF | R/W | Transmit/receive data (SHADOW_DATA) |

- Notes:
1. Refer to the ATA/ATAPI-7 and Serial-ATA specification for further information about this register.
 2. In PIO transfer mode, that is, except when the host_shdw_stat_DRQ signal is being asserted, both of R/W operations are canceled, and therefore the initial value H'FFFF is not read out but H'0000 is read out. When a read attempt is made after the PIO mode read transfer, the last read value is read out.
 3. Access to the data register is prohibited while DMA is active. The SDATA register is accessible.

59.2.13 Shadow Error Register (SERR[Read Mode]) (SSERR[Read Mode])

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The shadow error register corresponds to the ATA standard's error register. (The actual register is on the device side; the host side value is handled as a mirror value.)

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------|----|----|----|----|----|----|----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | SERR | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. |
| 7 to 0 | SERR | H'FF | R | Error monitor bit (SHADOW_ERROR) The bit definitions vary with issued commands. (Except for Bit 2) ABRT bit (Bit 2): The value is 1 when issued command is Abort. |

- Notes:
1. Refer to the ATA/ATAPI-7 and Serial-ATA specification for further information about this register. Note that a write to the address of this register essentially causes the shadow features register (SFEATURES[Write Mode]) (SSFEATURES[Write Mode]) to be accessed.
 2. The initial value of this register cannot be read out since the value of this register is normally rewritten by the device immediately after the interface gets ready for communication following a power-on reset. See section 59.3.13, Signature of the Shadow Register (INFORMATIVE).
 3. Access to the SERR register is prohibited while DMA is active. The SSERR register is accessible.

59.2.14 Shadow Features Register (SFEATURES[Write Mode]) (SSFEATURES[Write Mode])

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The shadow features register corresponds to the ATA standard's features register. (* The real entity of the register lies on the device side; the host side value is handled as a mirror value.)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-----------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | SFEATURES | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | W | W | W | W | W | W | W | W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved The write value should always be 0. |
| 7 to 0 | SFEATURES | H'FF | W | Subcommand setup bits (SHADOW_FEATURES) The bit definitions vary with the commands. |

- Notes:
1. Refer to the ATA/ATAPI-7 and Serial-ATA specification for further information about this register. Note that a read from the address of this register essentially causes the shadow error register (SERR[Read Mode]) (SSERR[Read Mode]) to be accessed.
 2. This register functions in a 2-stage FIFO configuration when a device that is compatible with the 48-bit Address features set is connected. For details, see section 59.3.12, Extended ATA Registers.
 3. Access to the SFEATURES register is prohibited while DMA is active. The SSFEATURES register is accessible.

59.2.15 Shadow Sector CNT Register (SECCNT) (SSECCNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The shadow sector CNT register corresponds to the ATA standard's sector count register.

(* The real entity of the register lies on the device side; the host side value is handled as a mirror value.)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|--------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | SECCNT | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | SECCNT | H'FF | R/W | Sector count bits (SHADOW_SECTOR_CNT) The bit definitions vary with the commands. |

- Notes:
1. Refer to the ATA/ATAPI-7 and Serial-ATA specification for further information about this register.
 2. The initial value of this register cannot be read out since the value of this register is normally rewritten by the device immediately after the interface gets ready for communication following a power-on reset. See section 59.3.13, Signature of the Shadow Register (INFORMATIVE).
 3. This register functions in a 2-stage FIFO configuration when a device that is compatible with the 48-bit Address features set is connected. For details, see section 59.3.12, Extended ATA Registers.
 4. Access to the SECCNT register is prohibited while DMA is active. The SSECCNT register is accessible.

59.2.16 Shadow LBA Low Register (LBALOW) (SLBALOW)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The shadow LBA low register corresponds to the ATA standard's LBA low (sector number) register.
 (* The real entity of the register lies on the device side; the host side value is handled as a mirror value.)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|--------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | LBALOW | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | LBALOW | H'FF | R/W | LBA_low address bits (SHADOW_LBA_LOW) The bit definitions vary with the commands. |

- Notes:
1. Refer to the ATA/ATAPI-7 and Serial-ATA specification for further information about this register.
 2. The initial value of this register cannot be read out since the value of this register is normally rewritten by the device immediately after the interface gets ready for communication following a power-on reset. See section 59.3.13, Signature of the Shadow Register (INFORMATIVE).
 3. This register functions in a 2-stage FIFO configuration when a device that is compatible with the 48-bit Address features set is connected. For details, see section 59.3.12, Extended ATA Registers.
 4. Access to the LBALOW register is prohibited while DMA is active. The SLBALOW register is accessible.

59.2.17 Shadow LBA Mid Register (LBAMID) (SLBAMID)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The shadow LBA mid register corresponds to the ATA standard's LBA mid (cylinder low) register. (* The real entity of the register lies on the device side; the host side value is handled as a mirror value.)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|--------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | LBAMID | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | LBAMID | H'FF | R/W | LBA_MID address bits (SHADOW_LBA_MID) The bit definitions vary with the commands. |

- Notes:
1. Refer to the ATA/ATAPI-7 and Serial-ATA specification for further information about this register.
 2. The initial value of this register cannot be read out since the value of this register is normally rewritten by the device immediately after the interface gets ready for communication following a power-on reset. See section 59.3.13, Signature of the Shadow Register (INFORMATIVE).
 3. This register functions in a 2-stage FIFO configuration when a device that is compatible with the 48-bit Address features set is connected. For details, see section 59.3.12, Extended ATA Registers.
 4. Access to the LBAMID register is prohibited while DMA is active. The SLBAMID register is accessible.

59.2.18 Shadow LBA High Register (LBAHIGH) (SLBAHIGH)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The shadow LBA high register corresponds to the ATA standard's LBA high (cylinder high) register. (* The real entity of the register lies on the device side; the host side value is handled as a mirror value.)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | LBAHIGH | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | LBAHIGH | H'FF | R/W | LBA_high address bits (SHADOW_LBA_HIGH) The bit definitions vary with the commands. |

- Notes:
1. Refer to the ATA/ATAPI-7 and Serial-ATA specification for further information about this register.
 2. The initial value of this register cannot be read out since the value of this register is normally rewritten by the device immediately after the interface gets ready for communication following a power-on reset. See section 59.3.13, Signature of the Shadow Register (INFORMATIVE).
 3. This register functions in a 2-stage FIFO configuration when a device that is compatible with the 48-bit Address features set is connected. For details, see section 59.3.12, Extended ATA Registers.
 4. Access to the LBAHIGH register is prohibited while DMA is active. The SLBAHIGH register is accessible.

59.2.19 Shadow Device/Head Register (DEVHEAD) (SDEVHEAD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The shadow device/head register corresponds to the ATA standard's device/head register.

(* The real entity of the register lies on the device side; the host side value is handled as a mirror value.)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | DEVHEAD | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 to 0 | DEVHEAD | H'FF | R/W | Device function select bits (SHADOW_DEVHEAD) The bit definitions except those for bits 7, 5, and 4 vary with the commands. Bits 7 and 5: Obsolete Bit 4: DEV bit. Selects either the Master or Slave. This bit must always be set to 0. |

- Notes:
1. Refer to the ATA/ATAPI-7 and Serial-ATA specification for further information about this register.
 2. The initial value of this register cannot be read out since the value of this register is normally rewritten by the device immediately after the interface gets ready for communication following a power-on reset. See section 59.3.13, Signature of the Shadow Register (INFORMATIVE).
 3. Access to the DEVHEAD register is prohibited while DMA is active. The SDEVHEAD register is accessible.

59.2.20 Shadow Status Register (SSTATUS [Read Mode]) (SSSTATUS [Read Mode])

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The SSTATUS or SSSTATUS register corresponds to the ATA standard's status register. Its value can be read out correctly only when the DEV bit (bit 4) of the shadow device/head register (DEVHEAD) (SDEVHEAD) is set to 0. Its read value is always H'0000_0000 when the DEV bit is set to 1.

Accessing this register for read causes the ATA-sourced host_intrq interrupt signal to be negated. (* The real entity of the register lies on the device side; the host side value is handled as a mirror value.)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-----|------|------|---------|-----|----|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | BSY | DRDY | DFSE | SSTATUS | DRQ | — | — | ERR |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. |
| 7 | BSY | B'0 | R | BSY bit |
| 6 | DRDY | B'1 | R | DRDY bit |
| 5 | DFSE | B'1 | R | DF/SE bit |
| 4 | SSTATUS | B'1 | R | Definition varies with the command. |
| 3 | DRQ | B'1 | R | DRQ bit |
| 2, 1 | — | B'11 | R | Reserved This bit is always read as 1. |
| 0 | ERR | B'1 | R | ERR/CHK bit |

- Notes:
1. Refer to the ATA/ATAPI-7 and Serial-ATA specification for further information about this register. Note that a write to the address of this register essentially causes the shadow command register (SCOM[Write Mode]) (SSCOM[Write Mode]) to be accessed.
 2. The initial value of this register cannot be read out since the value of this register is normally rewritten by the device immediately after the interface gets ready for communication following a power-on reset. See section 59.3.13, Signature of the Shadow Register (INFORMATIVE).
 3. Access to the SSTATUS register is prohibited while DMA is active. The SSSTATUS register is accessible.

59.2.21 Shadow Command Register (SCOM [Write Mode]) (SSCOM [Write Mode])

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The shadow command register corresponds to the ATA standard's command register. The ATA command code is set by this register.

Accessing this register for write causes the command to be issued to the device (sending command register FIS). (* The real entity of the register lies on the device side; the host side value is handled as a mirror value.)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | SCOM | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | W | W | W | W | W | W | W | W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved The write value should always be 0. |
| 7 to 0 | SCOM | H'FF | W | Transmit command setup bits (SHADOW_COMMAND) |

- Notes:
1. Refer to the ATA/ATAPI-7 and Serial-ATA specification for further information about this register. Note that a read from the address of this register essentially causes the shadow status register (SSTATUS[Read Mode]) (SSSTATUS[Read Mode]) to be accessed.
 2. Access to the SCOM register is prohibited while DMA is active. The SSCOM register is accessible.

59.2.22 Shadow alternate status register (SALTSTS [Read Mode]) (SSALTSTS [Read Mode])

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The SALTSTS or SSALTSTS register is the ATA standard's alternate status register. Its value can be read out correctly only when the DEV bit (bit 4) of the shadow device/head register (DEVHEAD) (SDEVHEAD) is set to 0. Its read value is always H'0000_0000 when the DEV bit is set to 1.

The bits of this register always have the same bit value as the corresponding bits of the shadow status register. Any attempt to access this register for read, however, does not cause the ATA-sourced host_intrq interrupt signal to be negated.

(* The real entity of the register lies on the device side; the host side value is handled as a mirror value.)

control register (SDEVCON[Write Mode])

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|------|-------|-------|---------|------|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | ABSY | ADRDY | ADFSE | SALTSTS | ADRQ | — | — | AERR |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved These bits are always read as 0. |
| 7 | ABSY | B'0 | R | BSY bit |
| 6 | ADRDY | B'1 | R | DRDY bit |
| 5 | ADFSE | B'1 | R | DF/SE bit |
| 4 | SALTSTS | B'1 | R | Definition varies with the command. |
| 3 | ADRQ | B'1 | R | DRQ bit |
| 2, 1 | — | B'11 | R | Reserved These bits are always read as 1. |
| 0 | AERR | B'1 | R | ERR/CHK bit |

- Notes:
1. Refer to the ATA/ATAPI-7 and Serial-ATA specification for further information about this register. Note that a write to the address of this register essentially causes the shadow device control register (SDEVCON[Write Mode]) (SSDEVCON[Write Mode]) to be accessed.
 2. The initial value of this register cannot be read out since the value of this register is normally rewritten by the device immediately after the interface gets ready for communication following a power-on reset. See section 59.3.13, Signature of the Shadow Register (INFORMATIVE).
 3. Access to the SALTSTS register is prohibited while DMA is active. The SSALTSTS register is accessible.

59.2.23 Shadow Device Control Register (SDEVCON [Write Mode]) (SSDEVCON [Write Mode])

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The shadow device control register is the ATA standard's device control register.

(* The real entity of the register lies on the device side; the host side value is handled as a mirror value.)

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-----|----|----|----|----|------|------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | HOB | — | — | — | — | SRST | NIEN | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | W | R | R | R | R | W | W | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved The write value should always be 0. |
| 7 | HOB | B'0 | W | HOB bit |
| 6 to 3 | — | All 0 | R | Reserved The write value should always be 0. |
| 2 | SRST | B'0 | W | SRST bit |
| 1 | NIEN | B'0 | W | nIEN bit |
| 0 | — | B'0 | R | Reserved The write value should always be 0. |

- Notes:
1. Refer to the ATA/ATAPI-7 and Serial-ATA specification for further information about this register. Note that a read from the address of this register essentially causes the Shadow alternate status register (SALTSTS[Read Mode]) (SSALTSTS[Read Mode]) to be accessed.
 2. Access to the SDEVCON register is prohibited while DMA is active. The SSDEVCON register is accessible.

59.2.24 SATA Extend ICC Register (SATAEICCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | ICC | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved The write value should always be 0. |
| 7 to 0 | ICC | H'00 | R | ICC stands for Isochronous Command Completion |

Note: Refer to the Serial-ATA specification for further information about this register.

59.2.25 SATA Extend Auxiliary Register (SATAEAUXR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Auxiliary | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Auxiliary | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | Auxiliary | H'0000_0000 | R | Auxiliary Contains parameter values specified on per command basis |

Note: Refer to the Serial-ATA specification for further information about this register.

59.2.26 SATA Extend DEVSLP Register (SATAEDEVSLPR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | DEVSLP_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. |
| 0 | DEVSLP_EN | B'0 | R/W | Sets the device sleep mode of the SATA host IP. 0: The device sleep mode is not set. 1: The device sleep mode is set. |

59.2.27 SCR SStatus Register (SCRSTS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-----|----|----|----|-----|----|----|----|-----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | IPM | | | | SPD | | | | DET | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. |
| 11 to 8 | IPM | B'0000 | R | Interface power manager state monitor bits These bits indicate the interface's Power Management mode or state. B'0000: No device is connected or the device is not ready for communication. B'0001: Active mode B'0010: Partial mode B'0110: Slumber mode B'1000: DevSleep mode Other: Setting prohibited. After a transition to the slumber mode from the partial mode by automatic partial to slumber, IPM is set to B'0110 to indicate the slumber mode. |
| 7 to 4 | SPD | B'0000 | R | Communication speed monitor bits These bits indicate the speed at which the interface communicates with the device. B'0000: No device is connected or the device is not ready for communication. B'0001: First generation (1.5 Gbps) B'0010: Second generation (3.0 Gbps) B'0011: Third generation (6.0 Gbps) Other: Setting prohibited. |
| 3 to 0 | DET | B'0000 | R | Device communication state monitor bits These bits indicate the PHY state with respect to device detection. B'0000: No device is detected or the device is not ready for communication. B'0001: A device is detected but the device is not ready for communication. B'0011: A device is detected and the device is ready for communication. B'0100: Offline mode or loopback mode Other: Setting prohibited. |

Note: Refer to the Serial-ATA specification for further information about this register.

59.2.28 SCR SError Register (SCRSERR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | DIAGA | DIAGX | DIAGF | DIAGT | DIAGS | DIAGH | DIAGC | DIAGD | DIAGB | DIAGW | DIAGI | DIAGN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | ERRE | ERRP | ERRC | ERRT | — | — | — | — | — | — | ERRM | ERRI |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R | R | R | R | R | R | R/WC1 | R/WC1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|---|
| 31 to 28 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 27 | DIAGA | B'0 | R/WC1 | DIAGA is used to sense COMWAKE before detecting a device. Port Selector Presence Detect (set when COMWAKE is detected before the Phy layer detects a device) |
| 26 | DIAGX | B'0 | R/WC1 | DIAGX is used to sense the change in device connection state. Exchanged (Set when the Phy layer detects a device connection or disconnection) |
| 25 | DIAGF | B'0 | R/WC1 | FIS type error bit Unrecognized FIS Type |
| 24 | DIAGT | B'0 | R/WC1 | Abnormal timing transfer result reception bit Transport state transition error (Set when transfer result notification is received at an invalid timing) |
| 23 | DIAGS | B'0 | R/WC1 | Abnormal state transition bit Link Sequence Error (Set when an illegal state transition occurs in the Link layer) |
| 22 | DIAGH | B'0 | R/WC1 | Handshake error bit |
| 21 | DIAGC | B'0 | R/WC1 | CRC error bit |
| 20 | DIAGD | B'0 | R/WC1 | Disparity error bit |
| 19 | DIAGB | B'0 | R/WC1 | 10b/8b decode error bit |
| 18 | DIAGW | B'0 | R/WC1 | COMWAKE detect bit COMWAKE Detect (Set when COMWAKE is detected at any timing) |
| 17 | DIAGI | B'0 | R/WC1 | Error state bit Phy Internal Error (Set when an error occurs in the Phy layer) |
| 16 | DIAGN | B'0 | R/WC1 | Transfer Ready state change bit PHYRDY change (Set when the PhyRdy signal indicating the Phy layer's transfer ready state is toggled) |
| 15 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-------|---|
| 11 | ERRE | B'0 | R/WC1 | Internal FIFO flow error bit Internal Error (Set when a flow error occurs in the internal FIFO. Perform a software reset whenever this bit is set.) |
| 10 | ERRP | B'0 | R/WC1 | Protocol error bit Protocol Error (Set when either one of the following conditions occurs): An illegal state transition occurs in the Link layer (same as the DIAGS bit) An unsupported FIS is received (same as the DIAGF bit). The size of the received FIS is found invalid. A transfer result notification is received at an invalid timing (same as the DIAGT bit). |
| 9 | ERRC | B'0 | R/WC1 | FIS transmission PhyRdy signal negated bit Non-recovered persistent communication or data integrity error (Set when the PhyRdy signal is negated during transmission of a FIS except Data FIS (including retries)) |
| 8 | ERRT | B'0 | R/WC1 | Data FIS transfer error bit Non-recovered transient data integrity error (Set when an error occurs during Data FIS transfer. Be sure to check this bit during PIO reads.) |
| 7 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | ERRM | B'0 | R/WC1 | Communication error recovery monitor bit Recovered communications error (The PhyRdy signal indicating the Phy layer's transfer ready state is negated but asserted again within approximately 200 μ s.) |
| 0 | ERRI | B'0 | R/WC1 | Retransmission result bit Recovered data integrity error (Set when retransmission of an FIS is successful) |

- Notes:
1. Refer to the Serial-ATA specification for further information about this register.
 2. There are cases in which the DIAGW and DIAGN bits of this register are already set before a first access is made since the initial data is automatically set in the register when it is provided with a clock signal after a power-on reset. Consequently, it is recommended that all bits of this register be cleared (writing H'FFFF_FFFF) at the beginning of the startup sequence.
 3. The operation of this register is irrelevant to the settings of the SATA INT status register (SATAINTSTAT) and SATA INT mask register (SATAINTMASK). The way in which the SCR SError register behaves remain unchanged even when bits b1, b2, and b3 of the SATA INT mask register are set (that is, interrupt sources are masked as the result of the update of the SCR SError result).

59.2.29 SCR SControl Register (SCRSCON)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|-----|-----|-----|------|-----|-----|-----|------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CSPM | | | | CIPM | | | | CSPD | | | | CDET | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W* | R/W* | R/W* | R/W* | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|------|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 12 | CSPM | B'0000 | R/W* | Power Management Mode transition request bits (SPM) CSPM is a set of bits that request for the state of the Power Management mode. Set as follows to have Power Management to transit to the required state: B'0000: No request B'0001: Request transition to the Partial Mode. B'0010: Request transition to the Slumber Mode. B'0100: Request to restoration to the Active Mode. Other: Setting prohibited. (* The transition request to the Partial/Slumber Mode is not retried. When using this field, unmask the SLUMMSK and PARTIMSK bits (bits 6 and 5) of the SATA INT mask register (SATAINTMASK) and enable "device rejects transition to Slumber Mode" and "device rejects transition to Partial Mode" interrupts.) |
| 11 to 8 | CIPM | B'0000 | R/W | Interface Power Management Mode enable bit (IPM) CIPM enables or disables the interface's Power Management mode. B'0000: No limitation. B'0001: Partial Mode Disable B'0010: Slumber Mode Disable B'0011: Partial/Slumber Mode Disable B'0100: DevSleep power management state is disabled. B'0101: Partial and DevSleep power management states are disabled. B'0110: Slumber and DevSleep power management states are disabled. B'0111: Partial, Slumber, and DevSleep power management states are disabled. Other: Setting prohibited. |

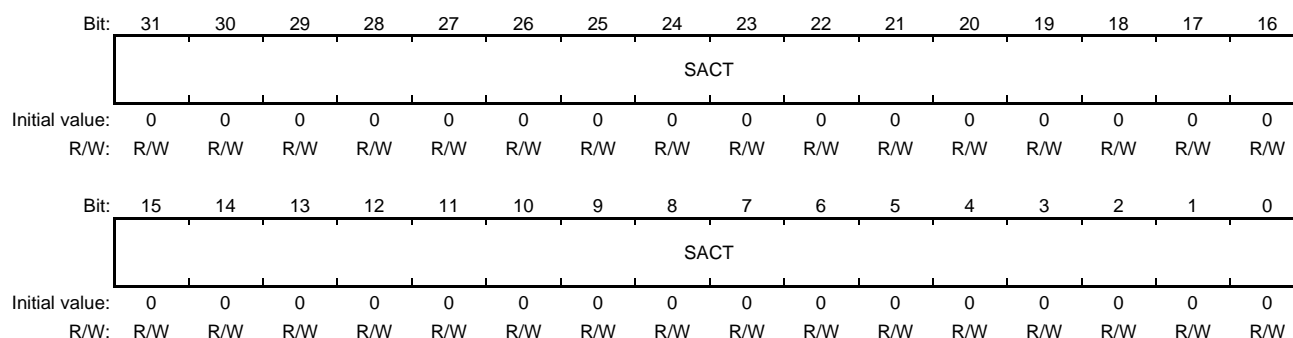
| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 4 | CSPD | B'0000 | R/W | <p>Communication speed bits (SPD)</p> <p>The CSPD bits define the limiting speed at which communication with the device is to be made.</p> <p>B'0000: No limitation (same as B'0010)</p> <p>B'0001: First generation (1.5 Gbps)</p> <p>B'0010: Second generation (3.0Gbps) or less</p> <p>B'0011: Third generation (6.0 Gbps) or less</p> <p>Other: Setting prohibited.</p> <p>(This field is not initialized by a hardware reset. It is initialized at power-on reset time.)</p> |
| 3 to 0 | CDET | B'0000 | R/W | <p>RESET/Offline mode bits (DET)</p> <p>The CDET bits specify the RESET/Offline mode.</p> <p>B'0000: No request</p> <p>B'0001: RESET. Handled as a hardware reset.</p> <p>B'0100: Places the PHY in offline mode.</p> <p>Other: Setting prohibited.</p> <p>(The SCR SControl register is initialized after the switching of the CDET field value from B'0001 to B'0000 is detected. Consequently, the values of the CSPM, CIPM, and CSPD fields are invalid when a write access such that the CDET field is set to B'0000 is made to this register.)</p> |

- Notes:
1. Refer to the Serial-ATA specification for further information about this register.
 2. It is impossible to check the value of this field since the field is reset to B'0000 as soon as setting of the value is internally detected.

59.2.30 SCR SActive Register (SCRSACT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The SCR SActive register is used to set up the TAG value to be used for Native Command Queuing.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | SACT | H'0000_0000 | R/W | b31: Corresponds to a TAG value of B'1_1111. b30: Corresponds to a TAG value of B'1_1110. b29: Corresponds to a TAG value of B'1_1101. b28: Corresponds to a TAG value of B'1_1100. b27: Corresponds to a TAG value of B'1_1011. b26: Corresponds to a TAG value of B'1_1010. b25: Corresponds to a TAG value of B'1_1001. b24: Corresponds to a TAG value of B'1_1000. b23: Corresponds to a TAG value of B'1_0111. b22: Corresponds to a TAG value of B'1_0110. b21: Corresponds to a TAG value of B'1_0101. b20: Corresponds to a TAG value of B'1_0100. b19: Corresponds to a TAG value of B'1_0011. b18: Corresponds to a TAG value of B'1_0010. b17: Corresponds to a TAG value of B'1_0001. b16: Corresponds to a TAG value of B'1_0000. b15: Corresponds to a TAG value of B'0_1111. b14: Corresponds to a TAG value of B'0_1110. b13: Corresponds to a TAG value of B'0_1101. b12: Corresponds to a TAG value of B'0_1100. b11: Corresponds to a TAG value of B'0_1011. b10: Corresponds to a TAG value of B'0_1010. b9: Corresponds to a TAG value of B'0_1001. b8: Corresponds to a TAG value of B'0_1000. b7: Corresponds to a TAG value of B'0_0111. b6: Corresponds to a TAG value of B'0_0110. b5: Corresponds to a TAG value of B'0_0101. b4: Corresponds to a TAG value of B'0_0100. b3: Corresponds to a TAG value of B'0_0011. b2: Corresponds to a TAG value of B'0_0010. b1: Corresponds to a TAG value of B'0_0001. b0: Corresponds to a TAG value of B'0_0000. |

- Notes:
1. Refer to the Serial-ATA specification for further information about this register.
 2. This register is write-only. To reset this register, it is necessary to reset the device (Software Reset will do).

59.2.31 SATA INT Status Register (SATAINTSTAT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

Bits 31 to 24 of the SATA INT status register are Shadow Status bits and mirror the value of the shadow status register. As with the shadow status register, accessing this register for read causes the ATA-sourced host_int_ata_intrq interrupt signal to be negated. (* The real entity of the register lies on the device side; the host side value is handled as a mirror value.)

Bits 11 to 0 of this register identify the interrupt source of the host_intrq signal. The bits that are masked by the SATA INT mask register (SATAINTMASK) do not cause the host_intrq signal to be asserted when the corresponding interrupt source event occurs. In this case, the pertinent bit is not set either.

| | | | | | | | | | | | | | | | | |
|----------------|---------|----------|----------|-------------|---------|-------|--------|---------|------|------|-------|-------|------|-----|---------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | INT BSY | INT DRDY | INT DFSE | INTSST ATUS | INT DRQ | — | — | INT ERR | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | SDBF | SLUMR | PARTIR | VEND | BIST | SLUM | PARTI | DMAST | SERR | ERR | ERR CRT | ATA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | RC | RC | RC | RC | RC | RC | RC | RC | RC | RC | RC | RC |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 | INTBSY | B'0 | R | BSY bit |
| 30 | INTDRDY | B'1 | R | DRDY bit |
| 29 | INTDFSE | B'1 | R | DF/SE bit |
| 28 | INTSSTATUS | B'1 | R | Definition varies with the command. |
| 27 | INTDRQ | B'1 | R | DRQ bit. |
| 26, 25 | — | B'11 | R | Reserved These bits are always read as 1. |
| 24 | INTERR | B'1 | R | ERR/CHK bit |
| 23 to 12 | — | All 0 | R | Reserved These bits are always read as 0. |
| 11 | SDBF | B'0 | RC | Set Device Bits FIS reception |
| 10 | SLUMR | B'0 | RC | A request for transition to the Slumber state has been received (for device-initiated power management, i.e. DIPM). |
| 9 | PARTIR | B'0 | RC | A request for transition to the Partial state has been received (for device-initiated power management, i.e. DIPM). |
| 8 | VEND | B'0 | RC | Vendor Specific FIS reception |
| 7 | BIST | B'0 | RC | BIST Active FIS reception |
| 6 | SLUM | B'0 | RC | Device rejects transition to Slumber Mode. Refer to notes 2. |
| 5 | PARTI | B'0 | RC | Device rejects transition to Partial Mode. Refer to notes 2. |
| 4 | DMAST | B'0 | RC | Same meaning as DMA Setup FIS reception (fpdma_req signal asserted) |
| 3 | SERR | B'0 | RC | SCR SERROR register update |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | ERR | B'0 | RC | SCR SERROR register (SCRSERR) ERRE, ERRP, ERRC, ERRT, ERRM, and ERRI bits update |
| 1 | ERRCRT | B'0 | RC | SCR SERROR register (SCRSERR) ERRE, ERRP, and ERRT bits update Refer to notes 1. |
| 0 | ATA | B'0 | RC | ATA source (equivalent to P-ATA's INTRQ) |

Notes: 1. When the "SCR SError register ERRE, ERRP, and ERRT bits update" interrupt bit is set, it indicates that the condition cannot be recovered by this host controller module. In such a case, take appropriate actions according to the operating state established before the error occurred and the value of the SCR SError register (SCRSERR).

Example:

- 1) When the ERRT bit of the SCR SError register (SCRSERR) is set on completion of a PIO Read (data-in) transfer:
The read data contains an error bit. Restart processing at the issuance of the command.
 - 2) When the ERRE bit of the SCR SError register (SCRSERR) is set during DMA transfer:
The data that is being transferred contains an error but since it cannot be detected by the device, it is necessary to perform a software reset and restart processing at the issuance of the command.
 - 3) When the ERRP bit of the SCR SError register (SCRSERR) is set:
Perform a hardware reset and re-execute the preceding operation.
2. When a "device rejects transition to Slumber Mode" or "device rejects transition to Partial Mode" interrupt occurs, no request is reissued (retried) to the device. If requests should be made again, reset the SCR SControl register with a correct value.

59.2.32 SATA INT Mask Register (SATAINTMASK)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The SATA INT mask register is used to mask the interrupt sources configured in the SATA INT status register (SATAINTSTAT). Interrupt sources corresponding to bits for which 1s have been specified are masked.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-------------|--------------|---------------|-------------|-------------|-------------|--------------|--------------|-------------|------------|---------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | SDBF MSK | SLUMR MSK | PARTIT MSK | VEND MSK | BIST MSK | SLUM MSK | PARTI MSK | DMAST MSK | SERR MSK | ERR MSK | ERRCRT MSK | ATA MSK |
| Initial value: | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11 | SDBFMSK | B'1 | R/W | Masks the "Set Device Bits FIS received" interrupt. |
| 10 | SLUMRMSK | B'0 | R/W | Mask for the "request for transition to Slumber mode" interrupt (for DIPM). |
| 9 | PARTITMSK | B'0 | R/W | Mask for the "request for transition to Partial mode" interrupt (for DIPM). |
| 8 | VENDMSK | B'1 | R/W | Masks the "Vendor Specific FIS received" interrupt. |
| 7 | BISTMSK | B'1 | R/W | Masks the "BIST Activate FIS received" interrupt. |
| 6 | SLUMMSK | B'1 | R/W | Masks the "device reject transition to Slumber mode" interrupt. |
| 5 | PARTIMSK | B'1 | R/W | Masks the "device reject transition to Partial mode" interrupt. |
| 4 | DMASTMSK | B'1 | R/W | Masks the "DMA Setup FIS received" interrupt. |
| 3 | SERRMSK | B'1 | R/W | Masks the "SCR SError register (SCRSERR) update" interrupt. |
| 2 | ERRMSK | B'1 | R/W | Masks the "SCR SError register (SCRSERR) ERRE, ERRP, ERRC, ERRT, ERRM, and ERRI bits update" interrupt. |
| 1 | ERRCRTMSK | B'0 | R/W | Masks the "SCR SError register (SCRSERR) ERRE, ERRP, and ERRT bits update" interrupt. |
| 0 | ATAMSK | B'0 | R/W | Masks the "ATA source (equivalent to P-ATA's INTRQ)" interrupt. The NIEN bit of the shadow device control register (SDEVCON) provides the same effect. |

59.2.33 PHY STOP Register (PHYSTOP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | PHY STOP |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved These bits are read as 0. The write value should be 0. |
| 0 | PHY STOP | B'0 | R/W | 0: Normal 1: PHY STOP |

Note: PHY STOP mode is used to stop the AFE. Apply a power-on reset to return the AFE from the PHY STOP mode (a software reset will not do this).

59.2.34 Rx DMA Setup FIS Dword0 Register (DMADW0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The Rx DMA setup FIS Dword0 Register contains the most recently received DMA Setup FIS information. It should be referenced when a DMA Setup FIS received interrupt occurs as indicated in the SATA INT status register (SATAINTSTAT).

| | | | | | | | | | | | | | | | | |
|----------------|-------------|------------|------------|----|--------|-----|-----|-----|---------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | AUTO ACT | DMA INT | DMA DIR | — | PMPORT | | | | FISTYPE | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

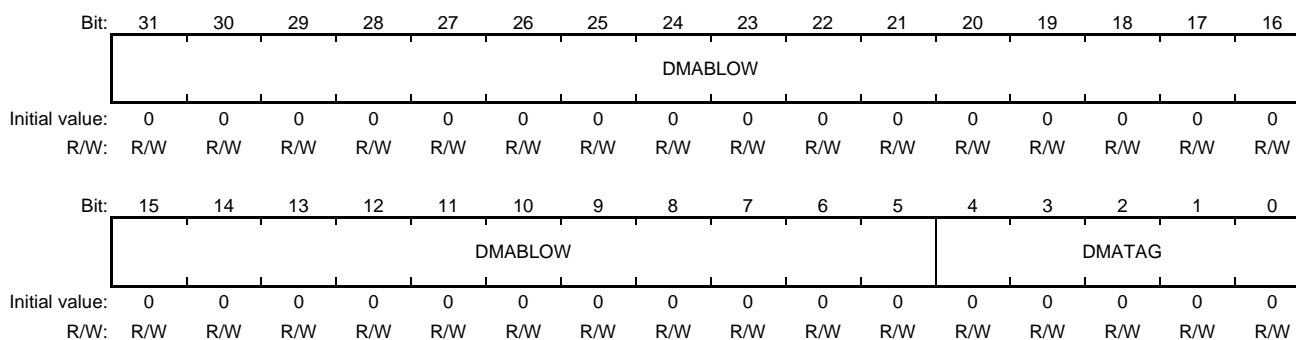
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | AUTOACT | B'0 | R/W | Auto-Activation bit 1 indicates Auto-Activation. |
| 14 | DMAINT | B'0 | R/W | Interrupt bit |
| 13 | DMADIR | B'0 | R/W | Data transfer direction bit The direction of the next data transfer. 0: Host → device 1: Device → host |
| 12 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 11 to 8 | PMPORT | All 0 | R/W | Source device ID Always set to B'0000 in NCQ mode. |
| 7 to 0 | FISTYPE | All 0 | R/W | FIS type bits FISTYPE indicates the type of a received FIS. The value is "41" when a DMA setup FIS is received. |

Note: Refer to the Serial-ATA specification for further information about this register.

59.2.35 Rx DMA Setup FIS Dword1 Register (DMADW1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The Rx DMA setup FIS Dword1 Register contains the most recently received DMA Setup FIS information. It should be referenced when a DMA Setup FIS received interrupt occurs as indicated in the SATA INT status register (SATAINTSTAT).



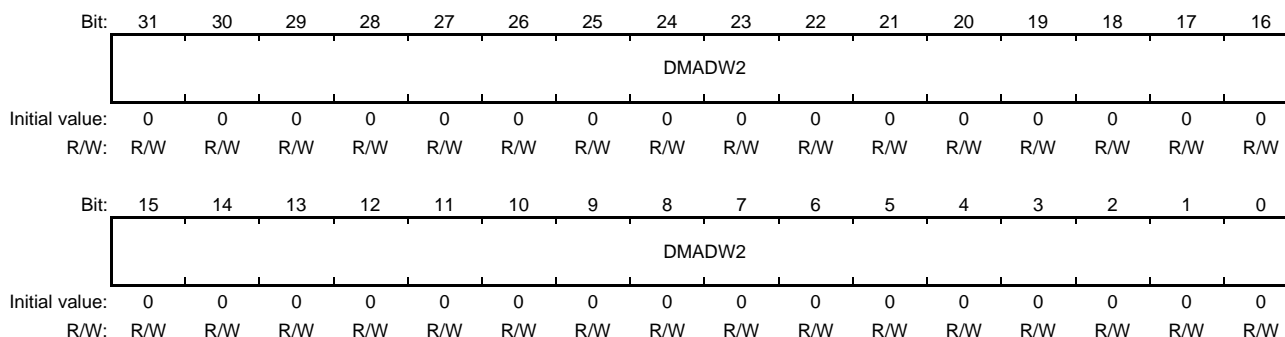
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|--|-----|---|
| 31 to 5 | DMABLOW | B'000_0000_0000_0000_0000_0000_0000_0000 | R/W | DMA Buffer Identifier Low field value in the DMA Setup FIS (set to H'0000_0000 in NCQ mode) |
| 4 to 0 | DMATAG | B'0_0000 | R/W | TAG field for NCQ |

Note: Refer to the Serial-ATA specification for further information about this register.

59.2.36 Rx DMA Setup FIS Dword2 Register (DMADW2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The Rx DMA setup FIS Dword2 Register contains the most recently received DMA Setup FIS information. It should be referenced when a DMA Setup FIS received interrupt occurs as indicated in the SATA INT status register (SATAINTSTAT).



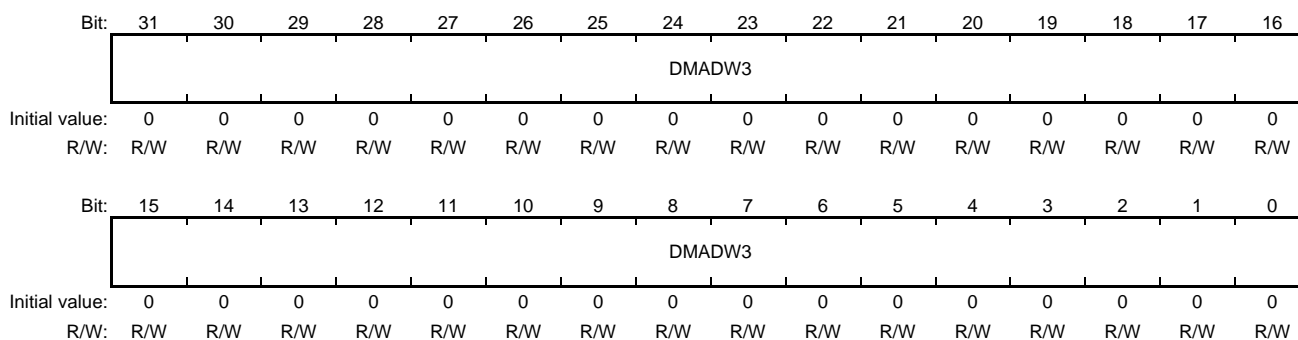
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | DMADW2 | H'0000_0000 | R/W | DMA Buffer Identifier High field value (Set to H'0000_0000 in NCQ mode) |

Note: Refer to the Serial-ATA specification for further information about this register.

59.2.37 Rx DMA Setup FIS Dword3 Register (DMADW3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The Rx DMA setup FIS Dword3 Register contains the most recently received DMA Setup FIS information. It should be referenced when a DMA Setup FIS received interrupt occurs as indicated in the SATA INT status register (SATAINTSTAT).



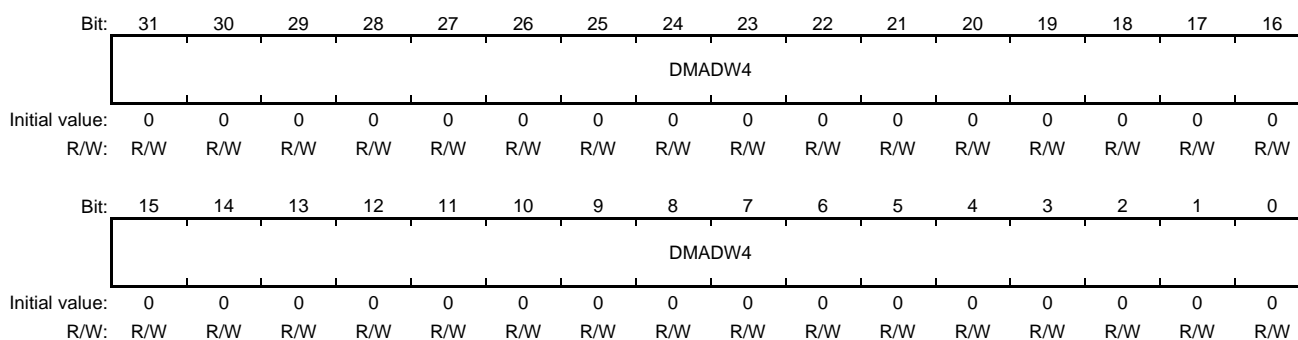
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | DMADW3 | H'0000_0000 | R/W | Dword3 value in the DMA Setup FIS (Normally set to H'0000_0000 as DMADW3 is handled as Reserved) |

Note: Refer to the Serial-ATA specification for further information about this register.

59.2.38 Rx DMA Setup FIS Dword4 Register (DMADW4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The Rx DMA setup FIS Dword4 Register contains the most recently received DMA Setup FIS information. It should be referenced when a DMA Setup FIS received interrupt occurs as indicated in the SATA INT status register (SATAINTSTAT).



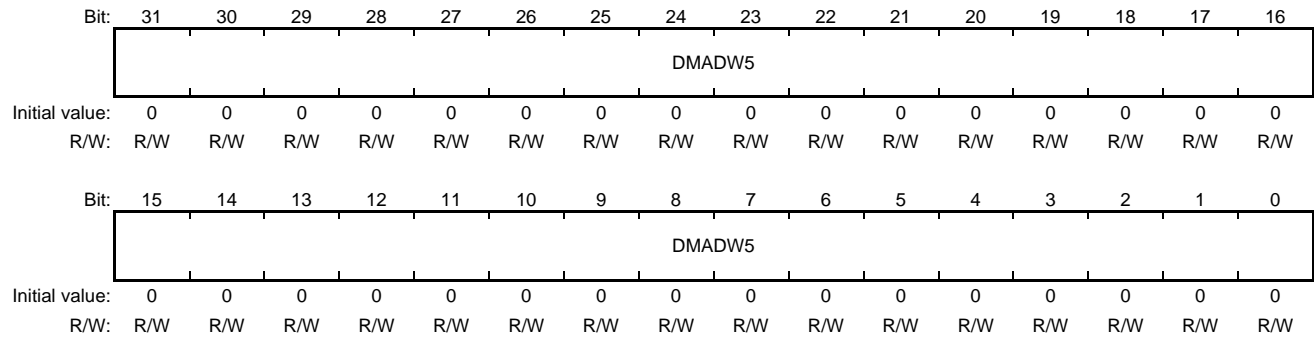
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | DMADW4 | H'0000_0000 | R/W | DMA Buffer Offset value in the DMA Setup FIS |

Note: Refer to the Serial-ATA specification for further information about this register.

59.2.39 Rx DMA Setup FIS Dword5 Register (DMADW5)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The Rx DMA setup FIS Dword5 Register contains the most recently received DMA Setup FIS information. It should be referenced when a DMA Setup FIS received interrupt occurs as indicated in the SATA INT status register (SATAINTSTAT).



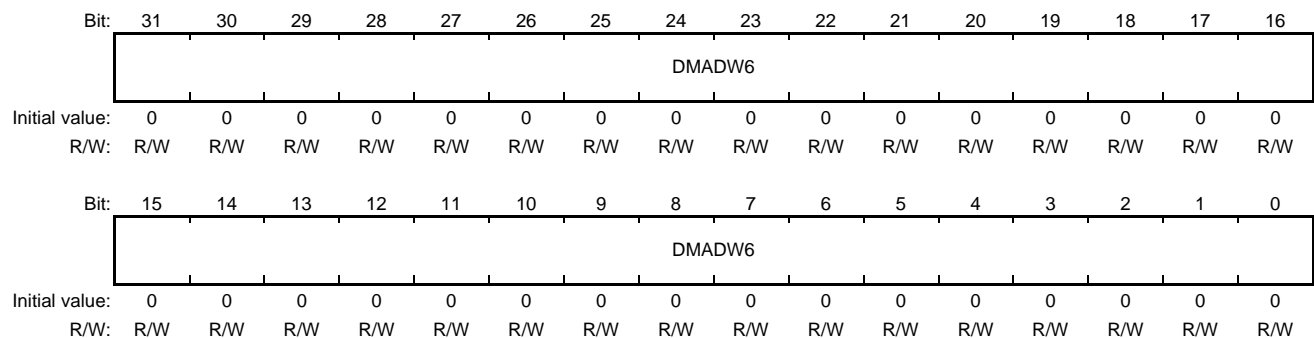
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | DMADW5 | H'0000_0000 | R/W | DMA Transfer Count value in the DMA Setup FIS |

Note: Refer to the Serial-ATA Specification for further information about this register.

59.2.40 Rx DMA Setup FIS Dword6 Register (DMADW6)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | — | √ | — |

The Rx DMA setup FIS Dword6 register contains the most recently received DMA Setup FIS information. It should be referenced when a DMA Setup FIS received interrupt occurs as indicated in the SATA INT status register (SATAINTSTAT).



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | DMADW6 | H'0000_0000 | R/W | Dword6 value in the DMA Setup FIS (Normally set to H'0000_0000 as DMADW6 is handled as Reserved) |

Note: Refer to the Serial-ATA specification for further information about this register.

59.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

59.3.1 Power Management Mode

The SATA interface can be switched into Partial, Slumber, or Active mode by setting up the CSPM field of the SCR SControl register (SCRSCON) for the required mode.

Note: Whenever restoring the SATA interface from Partial or Slumber mode to Active mode, be sure to verify with the SCR SStatus register (SCRSTS) that the SATA interface is in Partial or Slumber mode. Any request issued to the device to switch into Power Management Mode with this register setting is not retried when it is rejected. To reissue the request, it is necessary to set up the SCR SControl register (SCRSCON) again.

59.3.2 Device Sleep

The device sleep (DevSleep) mode is a low power consumption mode which has been added to revision 3.2 of the SATA standard. The DevSleep mode dramatically reduces power consumption for devices (HDD and SSD) while the devices are placed in the standby state. To support the DevSleep mode, a side-band signal (DEVSLP pin) must be added to the host chip. For this SATA host IP, we recommend use of bit 0 (DEVSLP_EN) of the SATA extend DEVSLP register to control the DEVSLP pin.

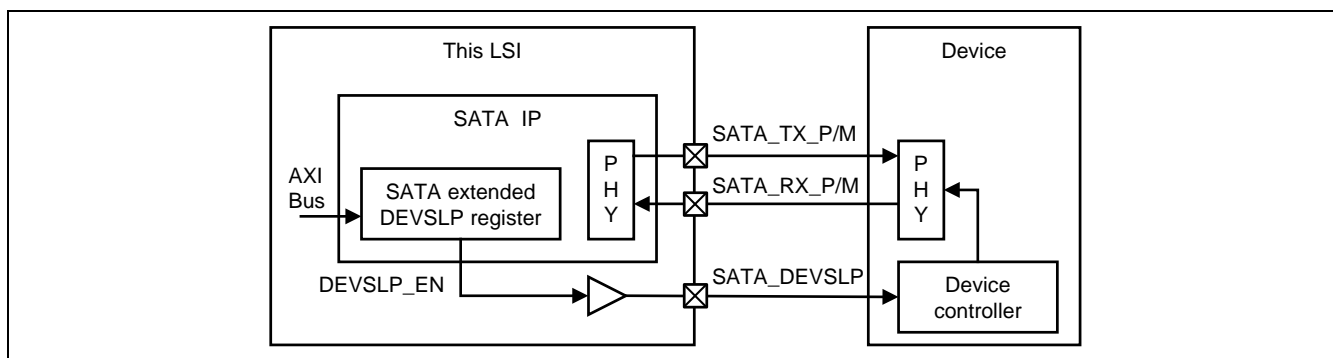


Figure 59.2 Example of DEVSLP Connection

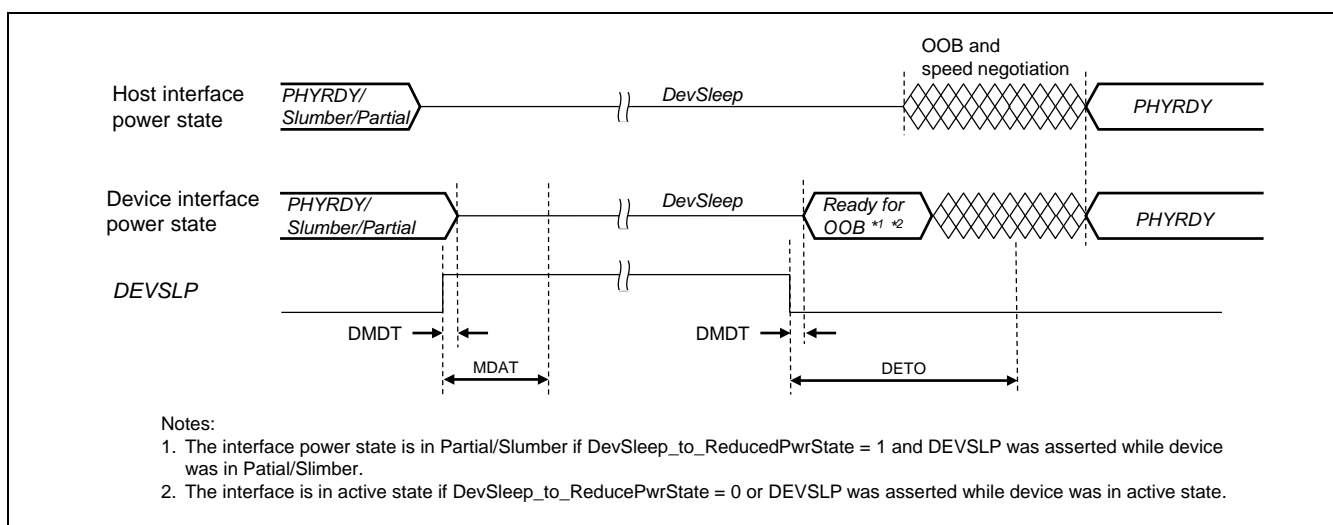


Figure 59.3 DEVSLP Protocol Overview (Source: Serial ATA Revision 3.2)

Table 59.5 DevSleep Timing Parameters (Source: Serial ATA Revision 3.2)

| Symbol | Parameter | Value |
|--------|---|---|
| MDAT | Minimum DEVSLP Assertion Time, minimum time that the host shall assert DEVSLP, once it has been asserted. | 10 ms unless otherwise specified in Identify Device Data Log. |
| DMDT | DEVSLP Minimum Detection Time, minimum time the device needs to de-bounce the DEVSLP signal after detecting it has been asserted. | 10 μ s |
| DETO | DevSleep Exit Timeout, maximum time from when DEVSLP is negated, to when the device shall be ready to detected OOB Signals. | 20 ms unless otherwise specified in Identify Device Data Log. |

59.3.3 Initialization

To perform communication using this module, it is necessary to set up the bridge function and establish a SATA connection.

Follow the procedure shown below to initialize the SATA.

(1) Release the Module from Standby

After resetting, release the module from the standby state. Refer to section 12, Module Standby, Software Reset, for details on module standby. The module enters the module standby state after release from the power-on reset state.

(2) Establishing OOB and Speed Negotiation

Start OOB and speed negotiation at the resetting of the standby state.

59.3.4 Interrupt Modes

The Link block supports two operating modes in addition to the ATA compliant registers.

This section describes the interrupt source masks and the specifications with respect to the clearing of the interrupt sources.

Figure 59.4 illustrates interrupt functions.

(1) IP Compatible Mode (Initial Value, when ISM = 0)

(a) Interrupt Source Mask

When an interrupt is masked, the corresponding interrupt flag is not set in the status register.

(b) Clearing Interrupt Sources

Interrupt sources are cleared by reading the SATA INT status register (SATAINTSTAT).

As with the shadow status register (SSTATUS), a read access causes the ATA-sourced (corresponding to INTRQ of P-ATA) host_intrq interrupt signal to be negated.

The ATA-specified interrupt source, SCR SError register (SCRSERR), is cleared by writing 1.

(2) CIS Interrupt Mode (when ISM = 1)

(a) Interrupt Source Mask

Interrupts are masked but corresponding flags are still set in the status register.

During negotiation, bit 3 in the Interrupt Status register is set. After negotiation ends, all bits in the Interrupt Status register must be cleared to 0.

(b) Clearing Interrupt Sources

Interrupt sources are cleared by writing 0.

A read access does not cause the ATA-sourced (corresponding to INTRQ of P-ATA) host_intrq interrupt signal to be negated.

The ATA-specified interrupt source, SCR SError register (SCRSERR), is cleared by writing 1; there is no change with this respect.

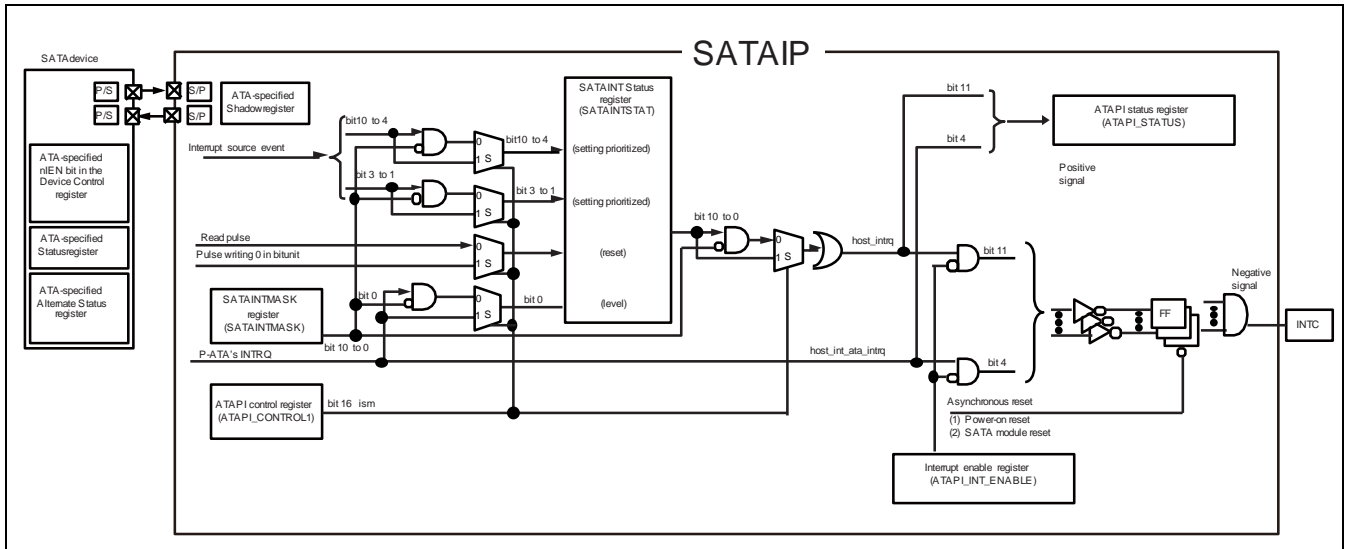


Figure 59.4 Interrupt Functions

59.3.5 Interrupt Processing Flow

The end of host device transfer interrupt and the ERR interrupt need to be considered.

The use of a timer in the wait processing is recommended to detect abnormal operations.

The device and host sources conflict. Figure 59.5 to 59.12 show the flow diagrams of interrupt processing.

The meanings of the words in the flow diagrams are given below.

| | |
|--------------------------|---|
| CMD: | The CPU requests data transfer by writing to the SATA host and device registers. |
| Data ready: | Data for writing to (DATA-OUT) the SATA device by the CPU are prepared in memory. |
| DATA: | Data to be transferred (data to be transferred by DMA) |
| Data reference: | The CPU refers to data to be read from (DATA-IN) the SATA device in memory. |
| DMA request: | Request for DMA transfer from the SATA device |
| DEVINT, SERR, SATAINT: | Interrupt request originating in the device, such as the request corresponding to DEVINT (bit 4) in the ATAPI status register (ATAPI_STATUS) |
| NEND: | Interrupt request for the end of normal transfer of the SATA host, such as the requests corresponding to NEND (bit 1) in the ATAPI status register (ATAPI_STATUS) |
| Response: | Response to completion of writing to memory of data read from the SATA device |
| 'Order may be reversed.' | The order of the error interrupt or the interrupt at the end of normal transfer to or from the device may be reversed before or after the end of normal transfer. |
| End of normal transfer: | The time of generation of the interrupt request corresponding to NEND (bit 1) or DNEND (bit 6) in the ATAPI status register (ATAPI_STATUS) as the source |

One of the two sets of events listed below will occur according to the response time of the device and the internal response time.

States (1) to (3) occur at the end of normal transfer of DATA-OUT:

- (1) Errors originating in the SATA device overlap.
- (2) Errors due to SATA communications and the host overlap.
- (3) The device's end interrupts will overlap if the device response is fast.

States (4) to (6) occur at the end of normal transfer of DATA-IN:

- (4) Errors originating in the SATA device overlap.
- (5) Errors due to SATA communications or the host overlap.
- (6) The device's end interrupts will overlap if the internal response is fast.

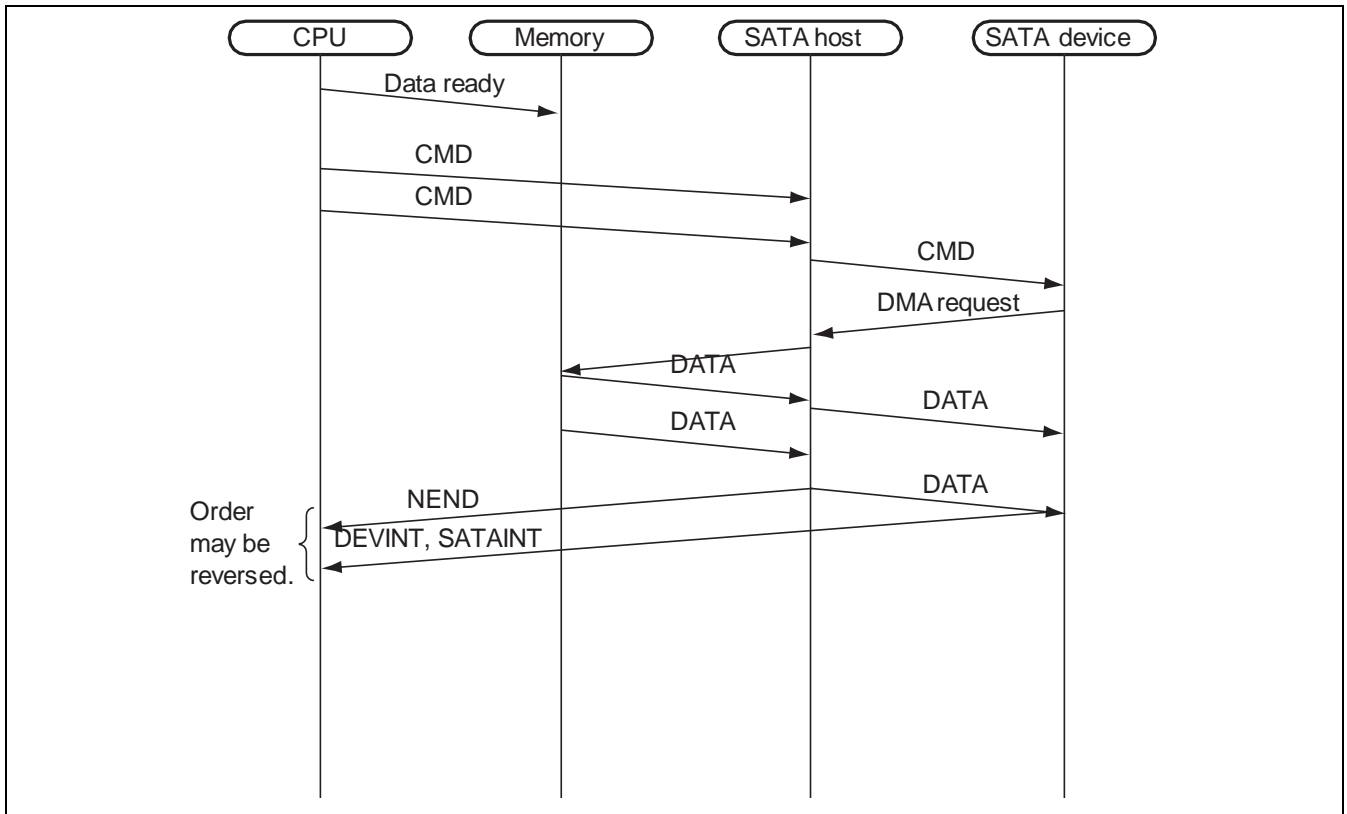


Figure 59.5 DATA-OUT (Normal Mode Interrupts)

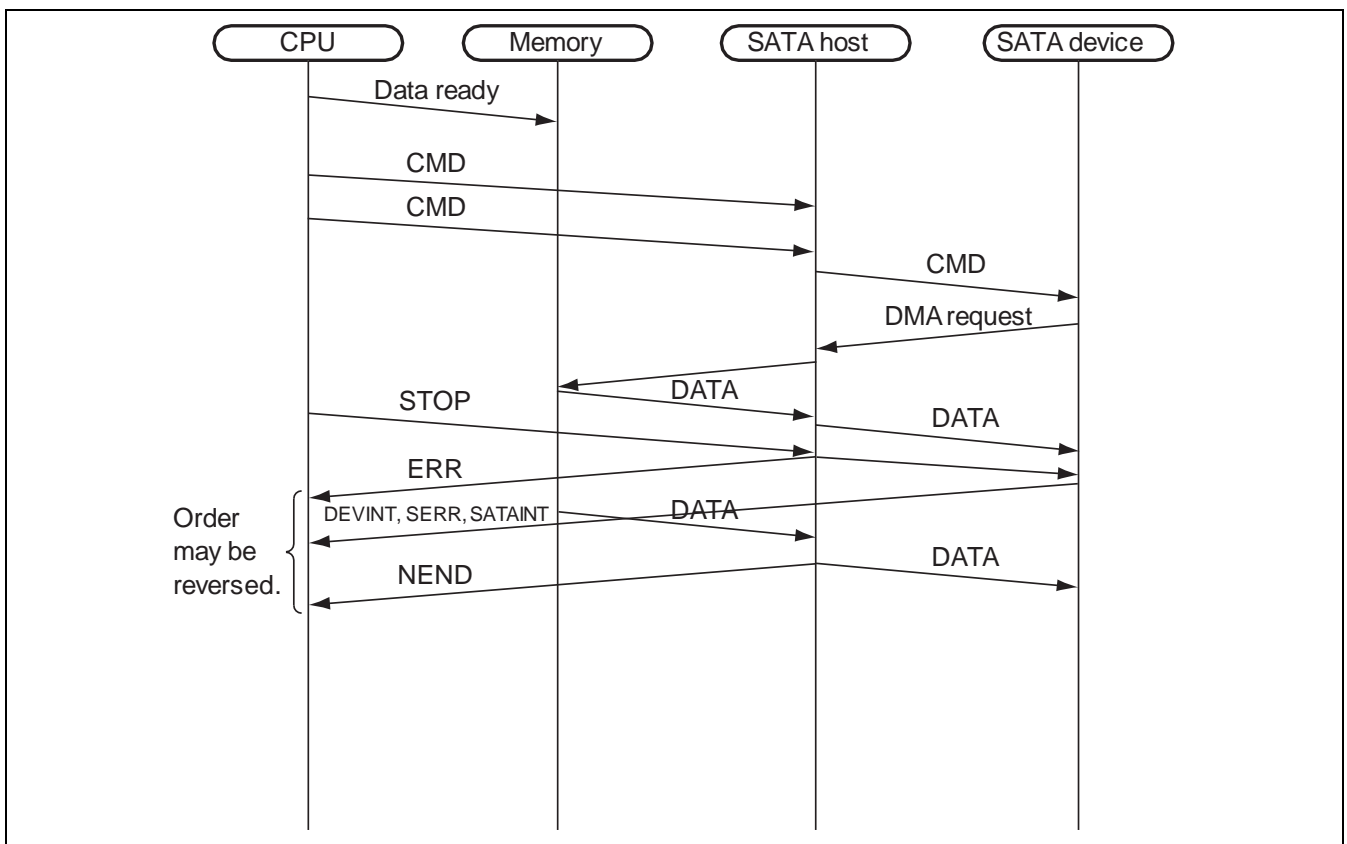


Figure 59.6 DATA-OUT (Internal Bus or CPU Sources)

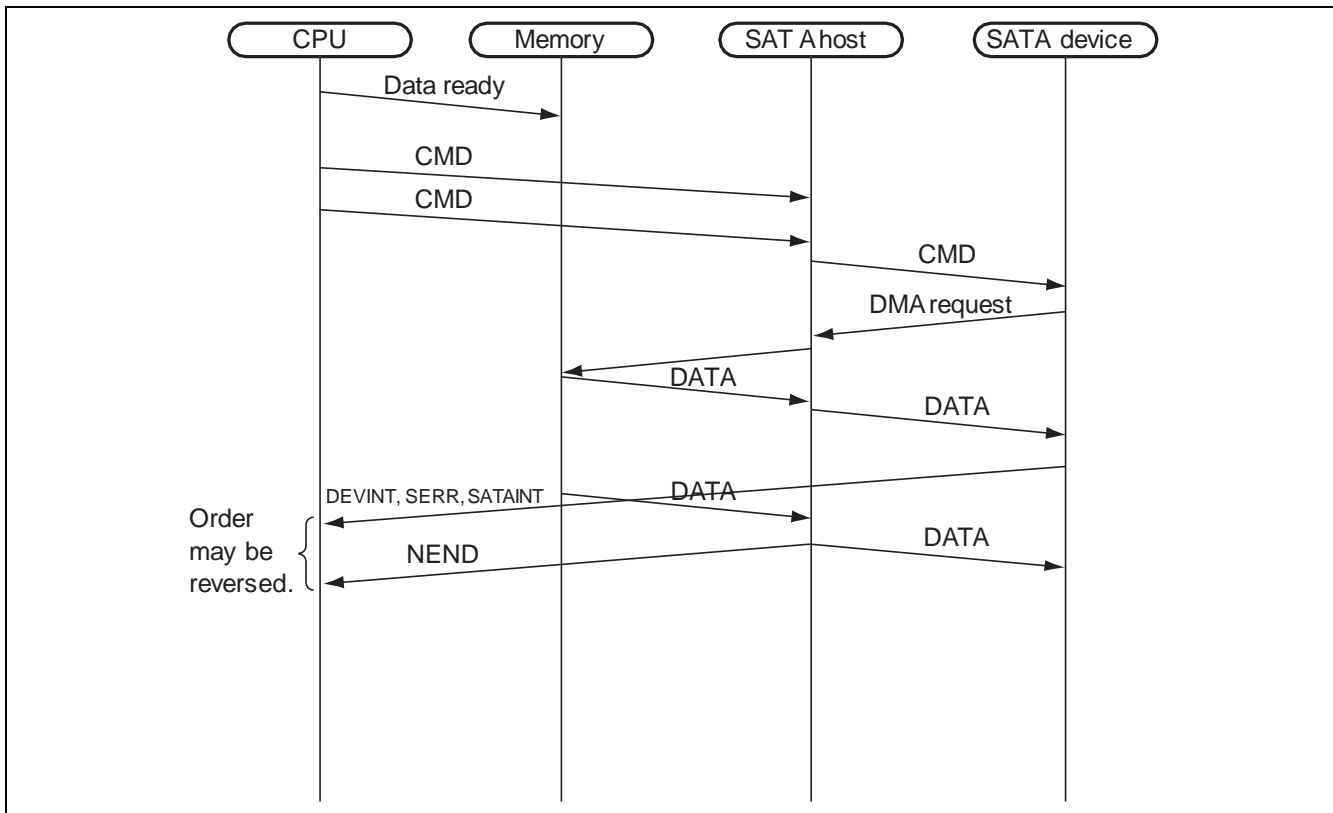


Figure 59.7 DATA-OUT (SATA Device Error)

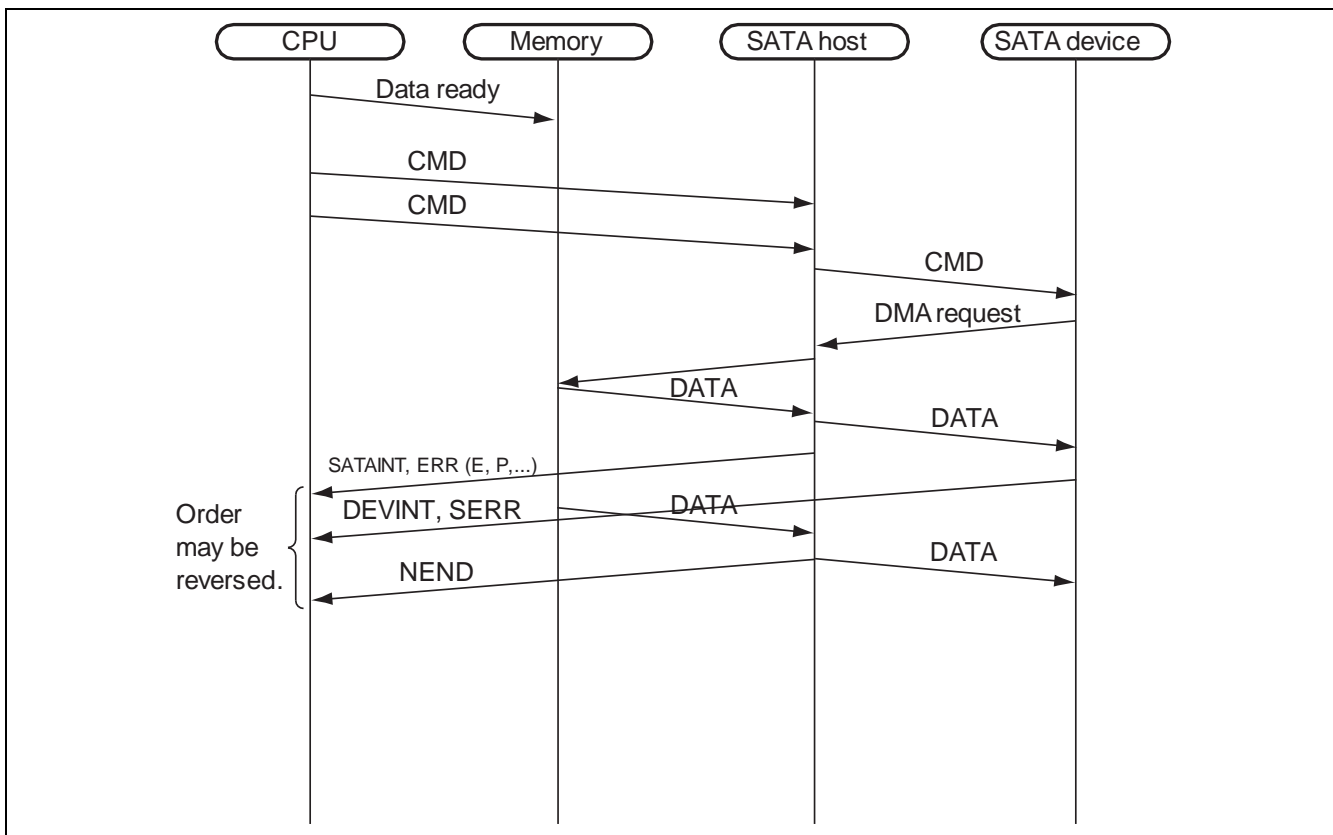


Figure 59.8 DATA-OUT (SATA Communication/Host-Sourced Error)

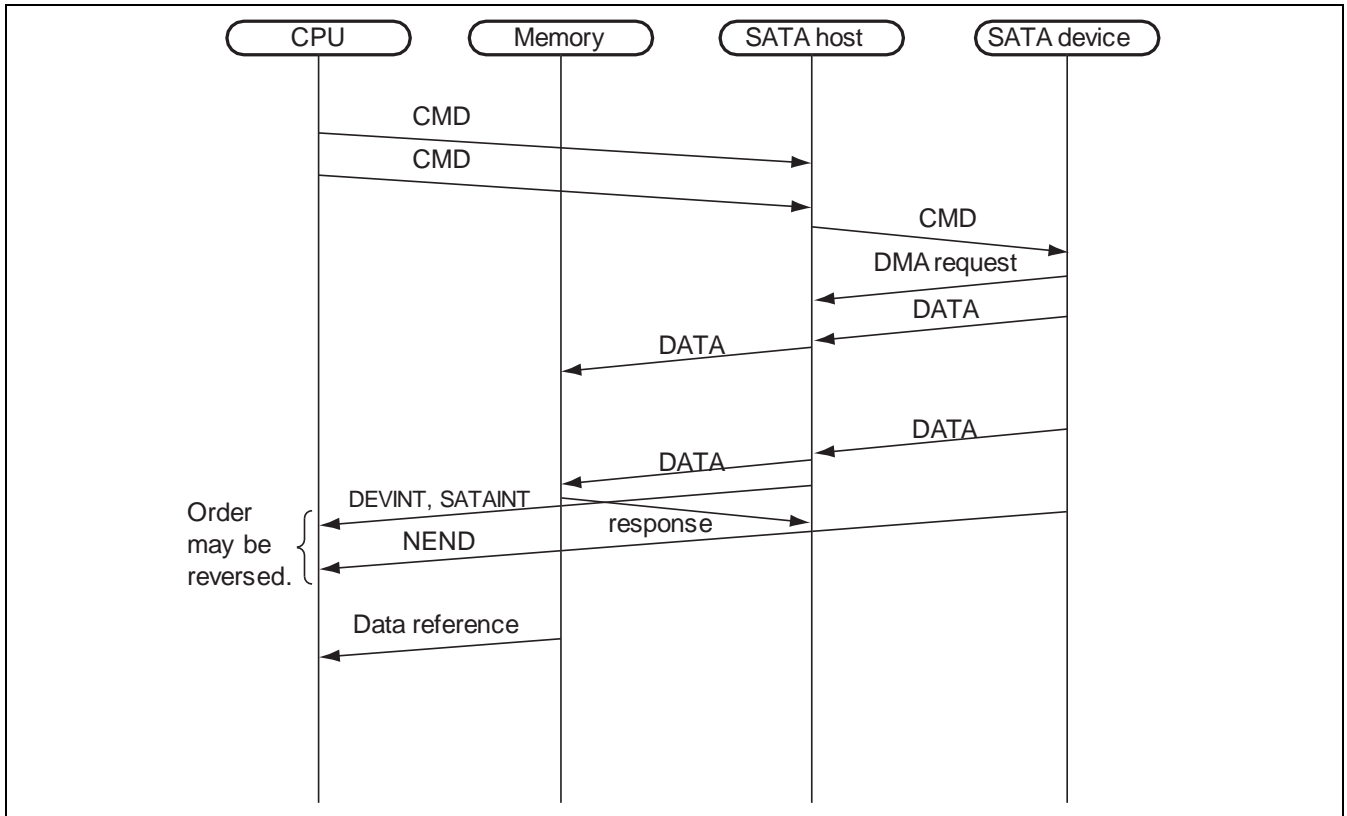


Figure 59.9 DATA-IN (Normal Mode Interrupts)

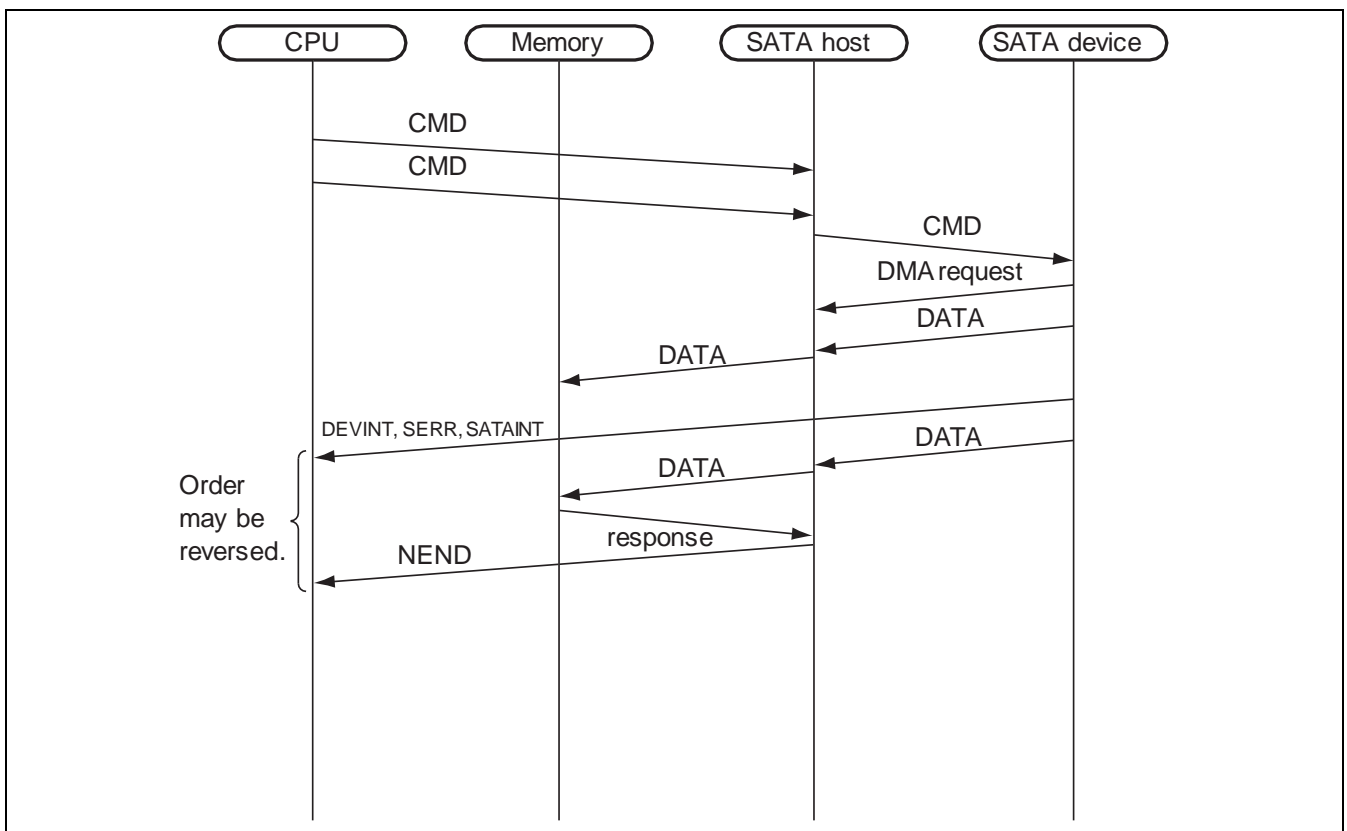


Figure 59.10 DATA-IN (SATA Device Error)

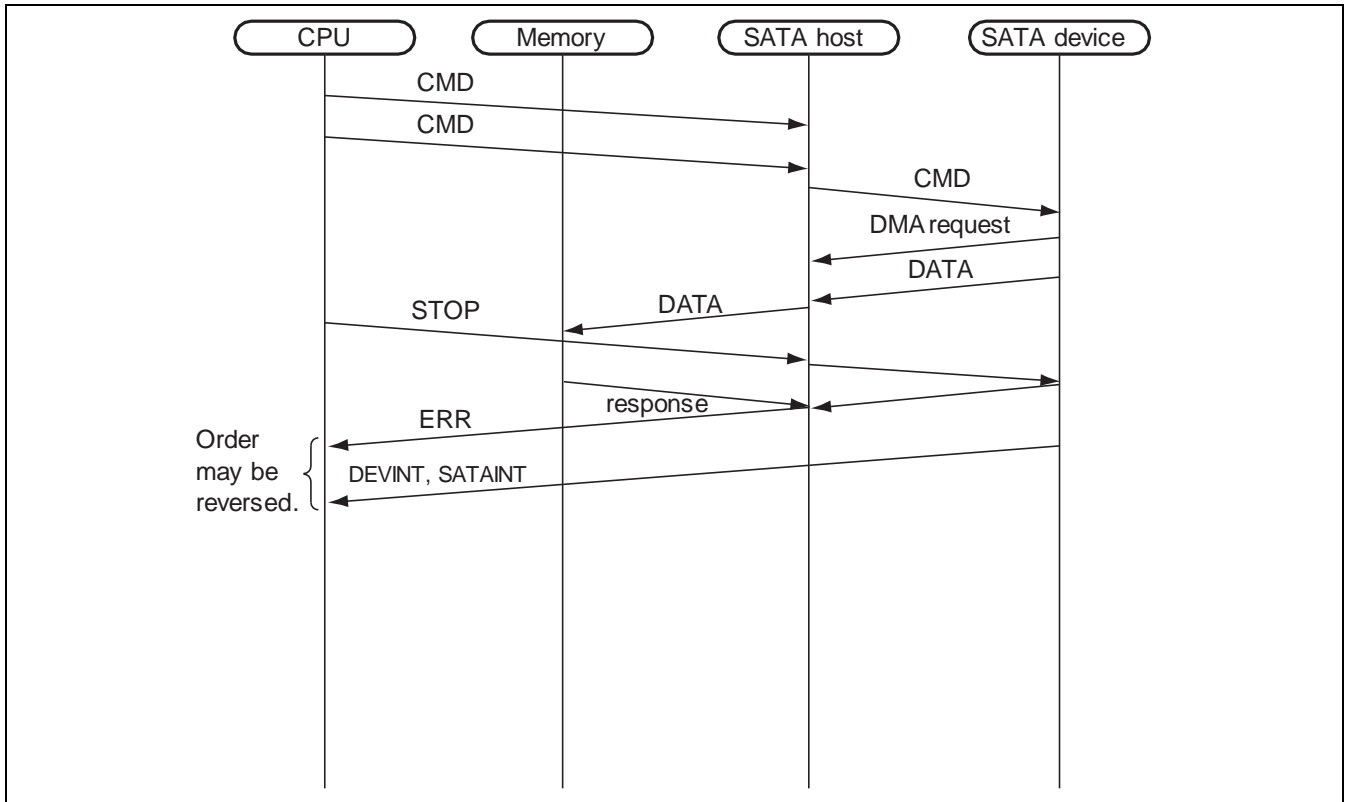


Figure 59.11 DATA-IN (Internal Bus or CPU Sources)

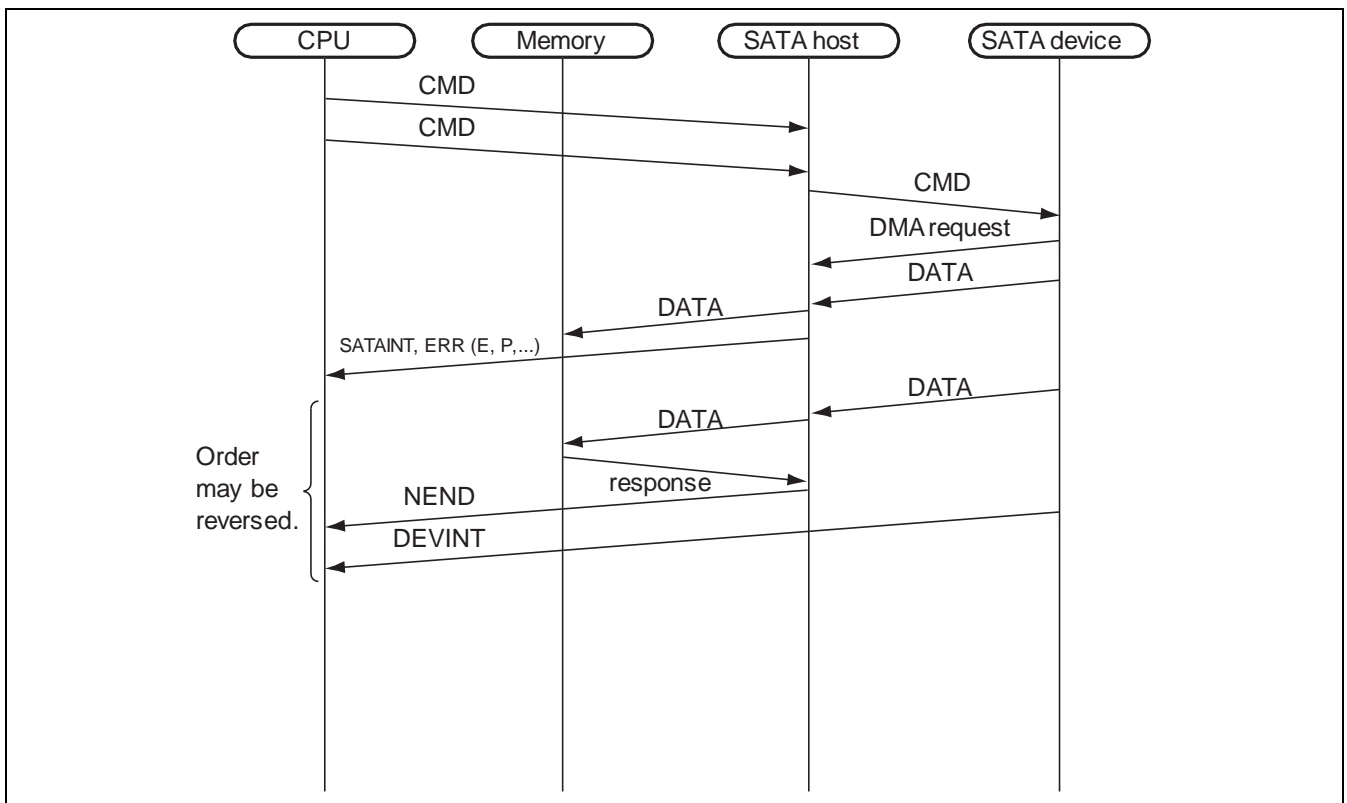


Figure 59.12 DATA-IN (SATA Communication/Host-Sourced Error)

59.3.6 Endian

This module adopts little endian. Consequently, data is represented in the same way as specified in the ATA/ATAPI-7 Specification and Serial-ATA Specification.

The bits of the registers are assigned in the order shown in Figure 59.13.

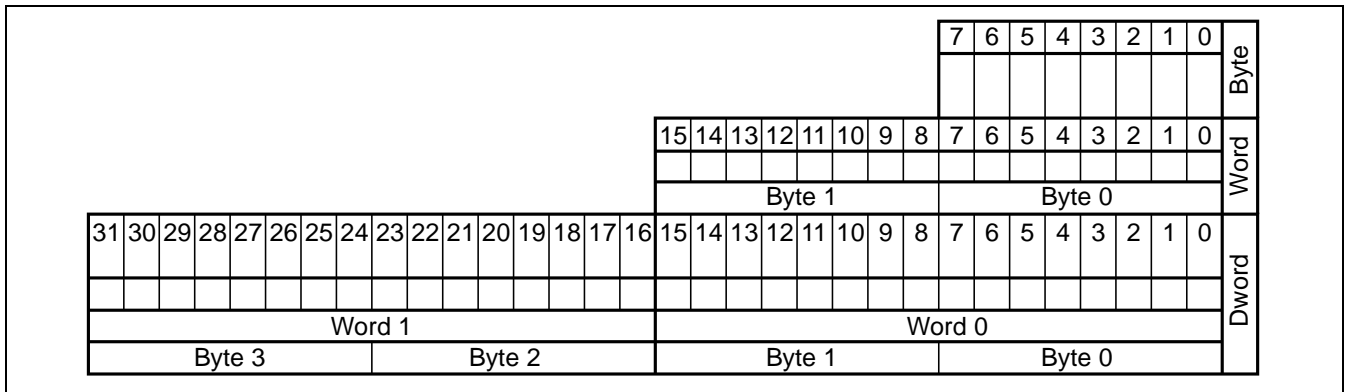


Figure 59.13 Endian and Bit Order

59.3.7 SATA BIST Modes

This module supports the Far-end Retimed Loopback and Far-end Analog Loopback modes that are stipulated in the Serial-ATA standard. The SATA BIST mode is the Loopback mode in which Loopback is formed on the host side. This mode can be configured and controlled for operation using the BIST config register (BISTCONF).

No match is performed between the transmit data from this module and the receive data from the device.

(1) Loopback Mode

In this mode, the data characters that are received are looped back to the sending side. The processing passes through the steps of input for reception → 10b/8b conversion → (Loopback) → 8b/10b conversion → output for transmission. This corresponds to the Serial-ATA Specification's Far-end Retimed Loopback (Figure 59.14).

There are cases in which some action is requested by the device side in this mode. More specifically, the device side will request the module to transmit the BIST Activate FIS. The module, however, can reject the request.

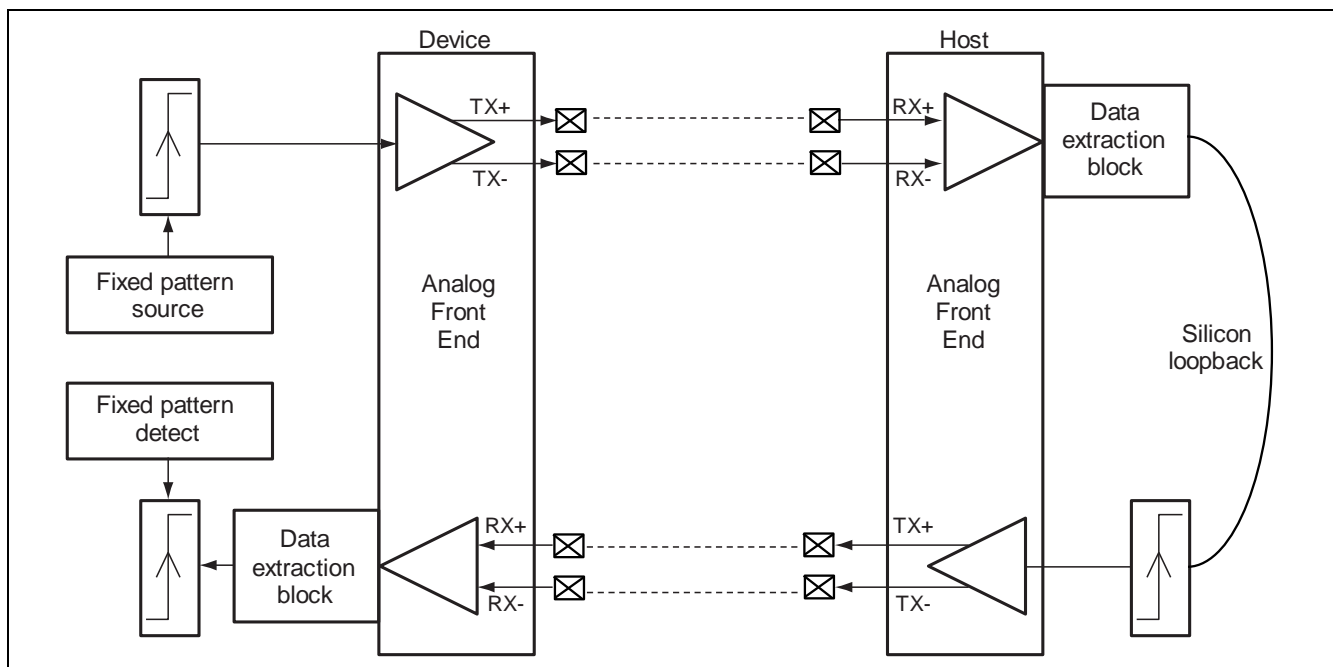


Figure 59.14 Far-End Retimed Loopback

(2) Analog Loopback Mode

This module does not support the Far-End Analog Loopback Mode (Figure 59.15) that is stipulated in the Serial-ATA Specification. In this mode, the received signal loops back through the AFE and output without modification as the result of which none of the Link and Transport layers get activated.

There are cases in which some action is requested by the device side in this mode. More specifically, the device side will request the module to transmit the BIST Activate FIS. The module, however, can reject the request.

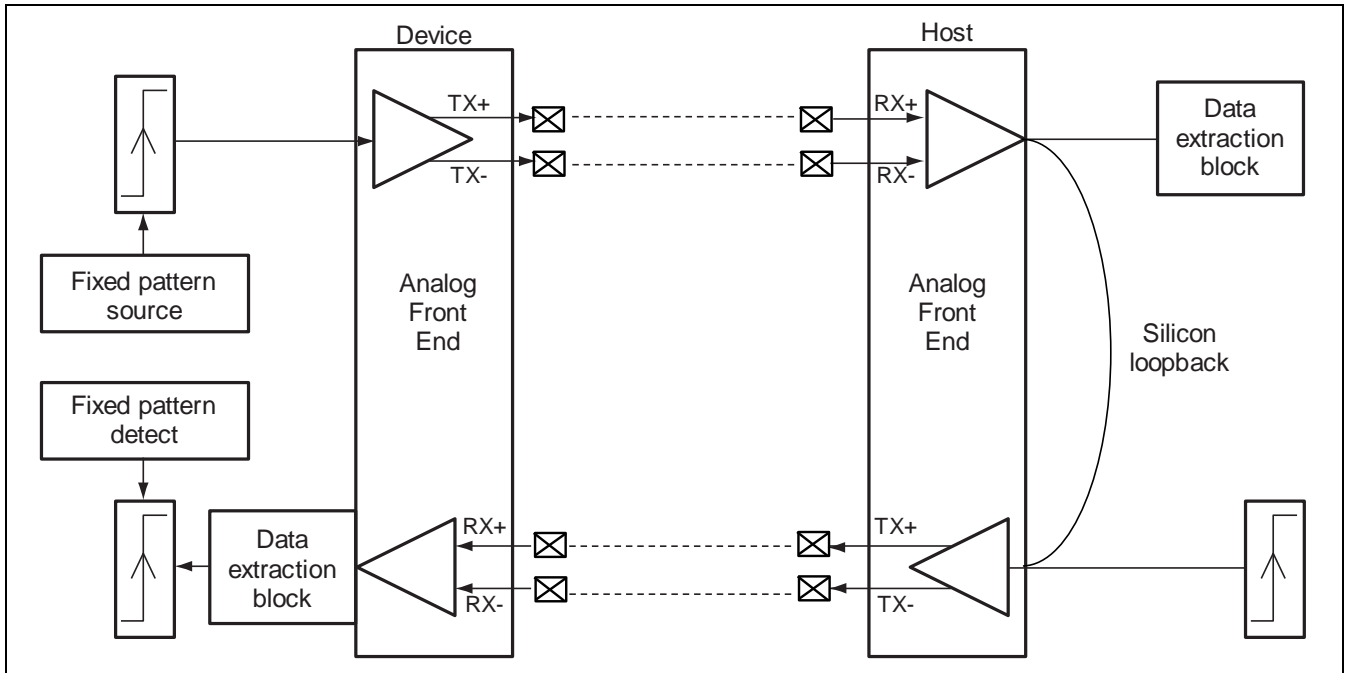


Figure 59.15 Far-End Analog Loopback

59.3.8 Operation Flow

Figure 59.16 shows the flow of operations for turning on the power, setting the interrupt mode, and initializing the PHY layer.

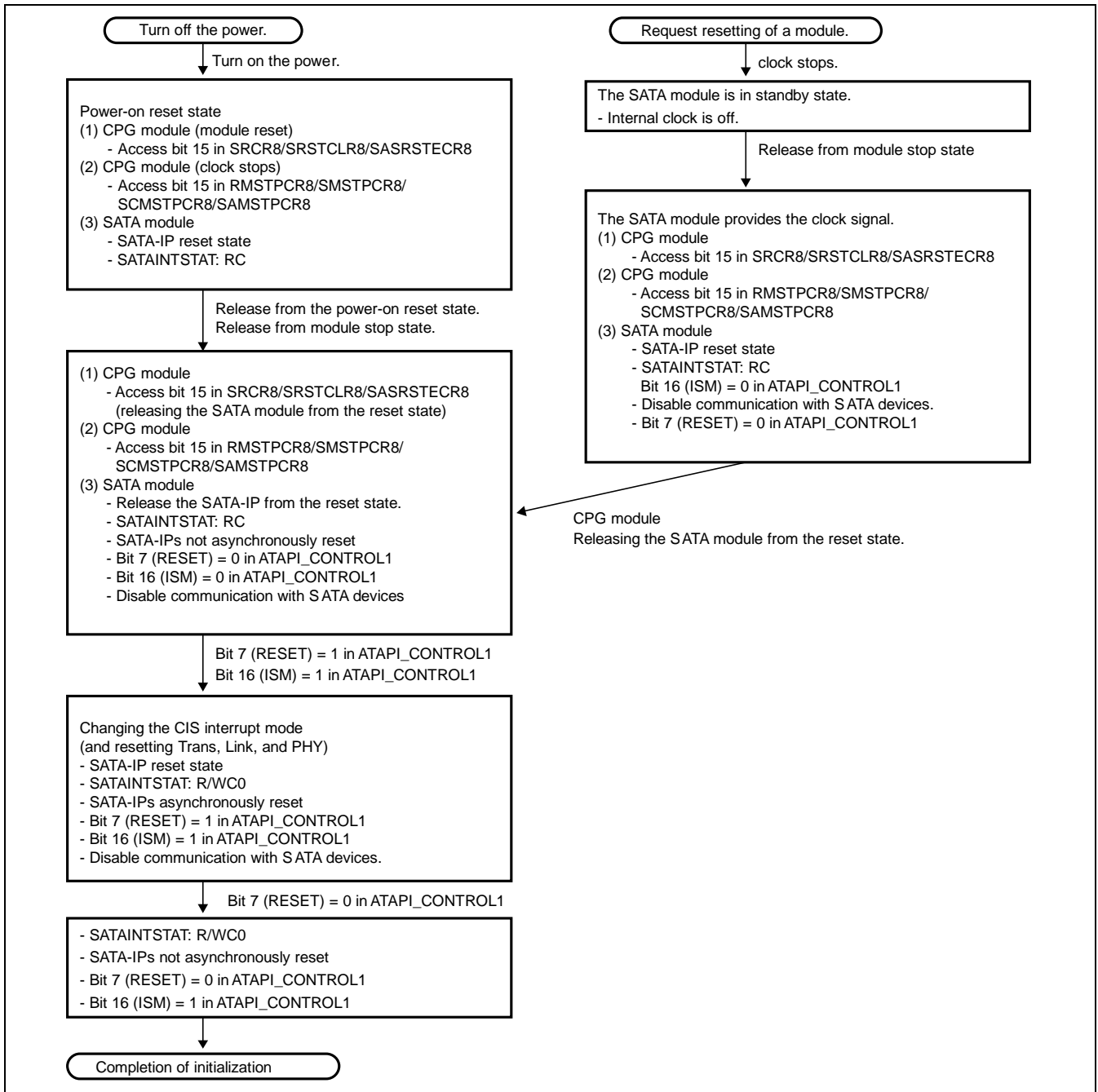


Figure 59.16 Operation Flow

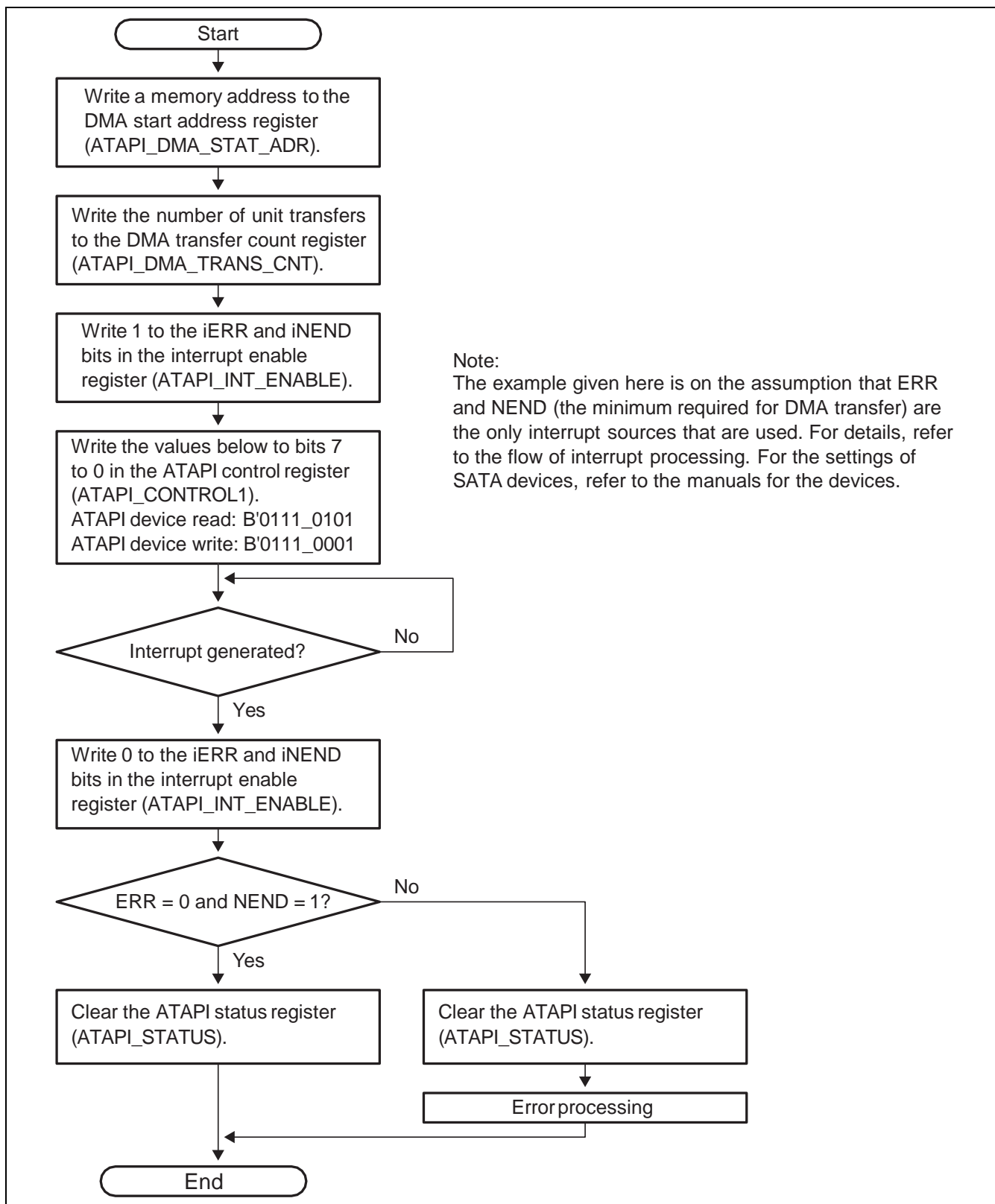


Figure 59.17 Flow of Data Transfer

59.3.9 Resets

(1) Power-on Reset

It is possible to reset this module into the initial state that is entered when the module is started by performing a power-on reset. Set and hold this line low for 25 μ s or longer before releasing the signal. When the reset sequence subsequently gets completed, the BSY bit of the shadow status register (SSTATUS) is set to 0 (the same sequence as in a hardware reset is carried out after the SATA interface is stabilized).

(2) Resetting the SATA Module

The CPG module has reset functions equivalent to those of a power-on reset. For resetting of the SATA module, refer to the descriptions of the CPG module.

(3) Hardware Reset

Changing the value of the CDET field of the SCR SControl register (SCRSCON) from B'0001 to B'0000 resets the device and the Phy, Link, and Transport layers' state machines. This handling corresponds to the Hardware Reset stipulated in the ATA standard. Reset the CDET field to B'0001 and, after the lapse of 25 μ s or longer, reset it to B'0000. When the reset sequence subsequently is completed, the BSY bit of the shadow status register (SSTATUS) is set to 0.

The hardware reset causes all of the shadow task registers (shadow data (DATA), shadow error (SERR), shadow features (SFEATURES), shadow sector CNT (SECCNT), shadow LBA low (LBALOW), shadow LBA mid (LBAMID), shadow LBA high (LBAHIGH), shadow device/head (DEVHEAD), shadow status (SSTATUS), shadow command (SCOM), shadow alternates status (SALTSTS), and shadow device control (SDEVCON) registers), SATA extend register and SCR registers (SCR SStatus (SCRSSSTS), SCR SError (SCRSEERR), SCR SControl (SCRSCON), and SCR SActive (SCRSACT) registers) to be reset.

Note: Refer also to the ATA/ATAPI-7 specification and Serial-ATA specification for further information about register update and hardware reset.

(4) Software Reset

The ATA specified software reset can be carried out by setting and resetting the SRST bit of the shadow device control register (DEVHEAD). Set the SRST bit to 1 and, after the lapse of 5 μ s or longer, reset the bit to 0. When the reset sequence subsequently is completed, the BSY bit of the shadow status register (SSTATUS) is set to 0.

Note: Refer also to the ATA/ATAPI-7 specification and Serial-ATA specification for further information about register update and software reset.

59.3.10 Extended ATA Registers

The five ATA stipulated registers, i.e., Features, sector count, LBA low, LBA mid, and LBA high registers, are provided with an 8-bit extended (exp) register. This register is used by the 48-bit Address feature set commands, WRITE FPDMA QUEUED, and READ FPDMA QUEUED commands.

This host controller module implements the exp registers for the above registers according to the ATA standard.

The following steps are required when writing data into the exp register of, for example, the Sector Count register:

- 1) Write a desired value to be set in sector count exp register to the sector count register.
- 2) Write a desired value to be set in sector count register to the sector count register.

Consequently, the value that is written in step 1) is pushed into the exp register by the access that is made in step 2).

This host controller module transmits the data corresponding to the exp register as contents of the (H2D) Register FIS irrespective of the command code specified. Whether the data corresponding to the exp register is used or not is at the discretion of the command interpreter on the device side and the host side makes no intervention.

59.3.11 Signature of the Shadow Register (INFORMATIVE)

The Serial-ATA Specification stipulates that the device shall transmit a Register FIS*¹ immediately after a power-on reset, hardware reset, or software reset*² for the purpose of loading a shadow register with a value called the signature. This is also stipulated in the ATA/ATAPI-7 standard in the same way. Even when this SATA module reads out the shadow register for the purpose of receiving this register FIS and having its value reflected in the shadow register, for example, following the completion of OOB processing at power-on time, it cannot read the original initial value of the register because its value has already been modified.

For details on the signature, refer to the section on "Device command layer protocol" in the Serial-ATA Specification.

- Notes:
1. There are commands such as EXECUTE DEVICE DIAGNOSTIC which take the signature as their return value. Their detailed description is omitted here. For details, refer to the Serial-ATA Specification.
 2. This refers to the PHYRDY state, that is, immediately after bits 3-0 of the SCR SStatus register (SCRSSSTS) are set to B'0011 for power-on reset and hardware reset, and to the time immediately after the reset processing is completed on the device side for software reset.

59.3.12 Reference Clock Source

Reference clock of SATA module is supplied from the external pin.

59.4 Usage Notes

RZ/G2H

RZ/G2M V1.3

RZ/G2M V3.0

RZ/G2N

RZ/G2E

59.4.1 Host Termination

(1) Restriction

Immediately after host termination is executed in this IP, conduct a power-on reset or reset the hardware immediately.

(2) Fractional Data in DOUT Transfer

For host termination in DOUT transfer, the last DWORD is copied, including the end of the data management unit for the device. Zero padding is applied where data do not take up 16 bits.

59.4.2 Clock Input in the Partial and Slumber States

Do not stop input of the reference clock signal while the interface is in the Partial or Slumber state.

60. USB2.0 Host (EHCI/OHCI)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

60.1 Overview

This module is a USB2.0 host controller conforming to the Enhanced Host Controller Interface (EHCI) Specification for USB Rev. 1.1.

Channel 0/3 can also be used as a dual-role device (DRD) or an OTG device in combination with a peripheral link module connected to the external UTMI+ interface.

60.1.1 Features

The descriptions in this section are compliant with the following USB 2.0 standards: "Universal Serial Bus Specification Revision 2.0" and "On-The-Go and Embedded Host Supplement to the USB Revision 2.0 Specification".

The following shows the features of this module.

| Function | Description |
|-------------------------|---|
| Host function | <ul style="list-style-type: none"> Supports High-speed (480 Mbps), Full-speed (12 Mbps), and Low-speed (1.5 Mbps) transfer. Conforms to the Open Host Controller Interface (OHCI) Specification for USB Rev. 1.0a. Conforms to the Enhanced Host Controller Interface (EHCI) Specification for USB Rev. 1.1.*¹ |
| Peripheral function | The peripheral function is available through a peripheral controller module connected to the external UTMI+ interface (only for channel 0). |
| Accessible memory space | 32 bits |
| USB port count | Downstream port: 1 port for each channel Upstream port: 1 port |
| Other functions | <ul style="list-style-type: none"> Battery charging function (supporting SDP and CDP conforming to Battery Charging Specification Rev. 1.2) Dual-role device function (static switching between the USB host and peripheral functions) OTG function*² (supporting HNP and SRP conforming to Specification Rev. 2.0) |

Notes: 1. Some functions (specifications) are not supported.
2. This function isn't supported RZ/G2E.

60.1.1.1 OTG Function

This module provides a function (registers) necessary to implement the OTG function using this module (host core) and the HS-USB module (peripheral). The registers necessary for OTG control are implemented in the UCOM module.

The OTG function can be used through appropriate control of these registers, EHCI registers, and HS-USB module registers by firmware according to the following policies.

[Basic policies of OTG control]

1. When this module is used for the OTG function, USB0SEL[1:0] in UGCTRL2 register of HS-USB should be set to B'11 and the host and peripheral modes should be switched through the UCOM registers.
2. The host function should be controlled through the EHCI registers and the peripheral function should be controlled through the HS-USB module registers in principle. The UCOM registers are used to control only the following.
 - Host and peripheral mode switching
 - VBUS supply control (A-device)
 - Interrupt control (ID interrupt, VBUS interrupt, etc.)
 - USB bus monitoring
 - 15-kΩ pulldown resistor control

60.1.1.2 Battery Charging Function

The UCOM module in this module provides registers for controlling battery charging. The battery charging function of the UTMI+ PHY connected to this module is controlled by these registers of UCOM.

60.1.1.3 Core Setting

60.1.1.4 Host/Peripheral Switching Specifications

This module controls whether to use channel 0 as a downstream port (host function) or an upstream port (peripheral function) through a register setting. Control the OTG_PERI bit according to the user system specifications.

Switching between the host and peripheral functions is controlled through bit 31 (OTG_PERI) in the common control register in the UCOM when USB0SEL[1:0] in UGCTRL2 register of HS-USB be set to B'11. This control assumes the case where this module is used for OTG.

| Channel | OTG_PERI (Register Bit) | Function |
|---------|----------------------------|---------------|
| 1 | — | Fixed to host |
| 0 | 0 | Host |
| | 1 | Peripheral |

60.1.2 External Pins

Table 60.1 External Pins of USB

| Name | Pin Name | I/O | Description | Second Generation RZ/G Series Products | | | | |
|---------------------------------|---|-----|---|---|----------------------------|--------|--------|--|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Clock input for USB | USB_EXTAL | I | Used as an external clock input pin (50MHz) | √ | √ | √ | — | |
| | USB_XTAL | O | Outputs amplified negative feedback of EXTAL (50MHz) | √ | √ | √ | — | |
| USB D+ data | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] DPn [RZ/G2E] USB0_DP0 | I/O | D+ Input/output of the on-chip transceiver Connect this pin to the D+ pin of the USB bus. | √ | √ | √ | √ | |
| USB D- data | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] DMn [RZ/G2E] USB0_DM0 | I/O | D- Input/output of the on-chip transceiver Connect this pin to the D- pin of the USB bus. | √ | √ | √ | √ | |
| USB Identification | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] IDn [RZ/G2E] USB0_ID | I/O | Connect to the pin on the Micro connectors that is used to differentiate a Micro-A plug from a Micro-B plug. | √ | √ | √ | √ | |
| USB VBUS | VBUSn | I/O | Connect to USB VBUS with external 30kΩ±1% series resistor. | √ | √ | √ | — | |
| USB external reference resistor | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] TXRTUNEn [RZ/G2E] USB0_RREF | I/O | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Connect to external 200Ω±1% resistor. [RZ/G2E] Connect to external 1.8kΩ±1% resistor. | √ | √ | √ | √ | |

- Notes:
1. The VBUS of USB connector must not connect directly to VBUSn pin.
 2. VBUSn pin must be isolated by an external 30kΩ±1% series resistor.
 3. 'n' in Pin Name column are 0 or 1 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, 0 for RZ/G2E.
 4. A charge pump external to the USB PHY must provide power to VBUSn pin.

60.1.3 Register Configuration

60.1.3.1 Register Attributes

Table 60.2 Register Bit-field types

| Register Attribute | Description |
|--------------------|--|
| R/W | Register bits can be read and written. |
| R/WC1 | Register bits can be read. A clear bit may be set by writing "1"; writing 0 to R/WC1 bits has no effect. |
| R | Register bits can only be read. |
| W | Register bits can only be written. |
| Reserved | Reserved bits are Read Only field. |

60.1.3.2 Address Mappings

Table 60.3 EHCI/OHCI Base Address

| EHCI/OHCI Base Address | Base Address | RZ/G2H | RZ/G2M | | | RZ/G2E |
|------------------------------|--------------|--------|--------|------|-----|--------|
| | | | V1.3 | V3.0 | V2N | |
| Channel 0 | H'EE08_0000 | √ | √ | √ | √ | √ |
| Channel 1 | H'EE0A_0000 | √ | √ | √ | — | — |

Note: √: Supported, —: Not Supported

| | |
|-------|--------------------------------------|
| H'FFC | Reserved |
| H'834 | |
| H'800 | UCOM registers |
| | Reserved |
| H'40C | |
| H'400 | Renesas UTMI + PHY setting registers |
| H'370 | Reserved |
| H'360 | Core defined registers |
| H'348 | Reserved |
| H'340 | Core defined registers |
| H'328 | Reserved |
| H'300 | Core defined registers |
| H'210 | Reserved |
| H'200 | AHB registers |
| | Reserved |
| H'168 | |
| H'160 | EHCI operational registers |
| H'13C | Reserved |
| H'120 | EHCI operational registers |
| H'110 | Reserved |
| H'100 | EHCI capability registers |
| H'064 | Reserved |
| H'000 | OHCI operational registers |

Figure 60.1 Address Map

60.1.3.3 Register Configuration

Table 60.4 Register Configuration (1)

| | | | | | Second Generation RZ/G Series Products | | | |
|--------------------|--------------------|-------------------|-------------|---------|---|----------------------------|--------|--------|
| Register Name | Abbreviation | Address Offset | Access Size | Remarks | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| | | | | | HcRevision | HcRevision | H'000 | 32 |
| HcControl | HcControl | H'004 | 32 | √ | √ | √ | √ | |
| HcCommandStatus | HcCommandStatus | H'008 | 32 | | √ | √ | √ | √ |
| HcInterruptStatus | HcInterruptStatus | H'00C | 32 | | √ | √ | √ | √ |
| HcInterruptEnable | HcInterruptEnable | H'010 | 32 | | √ | √ | √ | √ |
| HcInterruptDisable | HcInterruptDisable | H'014 | 32 | | √ | √ | √ | √ |
| HcHCCA | HcHCCA | H'018 | 32 | | √ | √ | √ | √ |
| HcPeriodCurrentED | HcPeriodCurrentED | H'01C | 32 | | √ | √ | √ | √ |
| HcControlHeadED | HcControlHeadED | H'020 | 32 | | √ | √ | √ | √ |
| HcControlCurrentED | HcControlCurrentED | H'024 | 32 | | √ | √ | √ | √ |
| HcBulkHeadED | HcBulkHeadED | H'028 | 32 | | √ | √ | √ | √ |
| HcBulkCurrentED | HcBulkCurrentED | H'02C | 32 | | √ | √ | √ | √ |
| HcDoneHead | HcDoneHead | H'030 | 32 | | √ | √ | √ | √ |
| HcFmInterval | HcFmInterval | H'034 | 32 | | √ | √ | √ | √ |
| HcFmRemaining | HcFmRemaining | H'038 | 32 | | √ | √ | √ | √ |
| HcFmNumber | HcFmNumber | H'03C | 32 | | √ | √ | √ | √ |
| HcPeriodicStart | HcPeriodicStart | H'040 | 32 | | √ | √ | √ | √ |
| HcLSThreshold | HcLSThreshold | H'044 | 32 | | √ | √ | √ | √ |
| HcRhDescriptorA | HcRhDescriptorA | H'048 | 32 | | √ | √ | √ | √ |
| HcRhDescriptorB | HcRhDescriptorB | H'04C | 32 | | √ | √ | √ | √ |
| HcRhStatus | HcRhStatus | H'050 | 32 | | √ | √ | √ | √ |
| HcRhPortStatus1 | HcRhPortStatus1 | H'054 | 32 | | √ | √ | √ | √ |
| Reserved | — | H'058 | — | | √ | √ | √ | √ |
| Reserved * | — | H'064 to H'0FC | — | | √ | √ | √ | √ |

Note: * This area is read as 0. The write value should be 0.

Table 60.5 Register Configuration (2)

| Register Name | Abbreviation | Address Offset | Access Size | Remarks | Second Generation RZ/G Series Products | | | |
|--------------------------------------|------------------|----------------|-------------|----------------------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| HCVERSION / CAPLENGTH | CAPL_VERSION | H'100 | 32 | EHCI capability registers | √ | √ | √ | √ |
| HCSPARAMS | HCSPARAMS | H'104 | 32 | | √ | √ | √ | √ |
| HCCPARAMS | HCCPARAMS | H'108 | 32 | | √ | √ | √ | √ |
| HCSP_PORTROUT E | HCSP_PORTROUT E | H'10C | 32 | | √ | √ | √ | √ |
| Reserved * | — | H'110 to H'11C | — | | √ | √ | √ | √ |
| USBCMD | USBCMD | H'120 | 32 | EHCI operational registers | √ | √ | √ | √ |
| USBSTS | USBSTS | H'124 | 32 | | √ | √ | √ | √ |
| USBINTR | USBINTR | H'128 | 32 | | √ | √ | √ | √ |
| FRINDEX | FRINDEX | H'12C | 32 | | √ | √ | √ | √ |
| CTRLDSSEGMENT | CTRLDSSEGMENT | H'130 | 32 | | √ | √ | √ | √ |
| PERIODICLISTBASE | PERIODICLISTBASE | H'134 | 32 | | √ | √ | √ | √ |
| ASYNCLISTADDR | ASYNCLISTADDR | H'138 | 32 | | √ | √ | √ | √ |
| Reserved * | — | H'13C to H'15C | — | | √ | √ | √ | √ |
| CONFIGFLAG | CONFIGFLAG | H'160 | 32 | | √ | √ | √ | √ |
| PORTSC1 | PORTSC1 | H'164 | 32 | | √ | √ | √ | √ |
| Reserved | — | H'168 | — | | √ | √ | √ | √ |
| Reserved * | — | H'16C to H'1FC | — | | √ | √ | √ | √ |
| INT_ENABLE | INT_ENABLE | H'200 | 32 | AHB bridge registers | √ | √ | √ | √ |
| INT_STATUS | INT_STATUS | H'204 | 32 | | √ | √ | √ | √ |
| AHB_BUS_CTR | AHB_BUS_CTR | H'208 | 32 | | √ | √ | √ | √ |
| USBCTR | USBCTR | H'20C | 32 | | √ | √ | √ | √ |
| Reserved * | — | H'210 to H'2FC | — | | √ | √ | √ | √ |
| REVID | REVID | H'300 | 32 | Core defined registers | √ | √ | √ | √ |
| Register Enable/Clock Gating Control | REGEN_CG_CTRL | H'304 | 32 | | √ | √ | √ | √ |
| Suspend Control | SPD_CTRL | H'308 | 32 | | √ | √ | √ | √ |
| Suspend/Resume Timer Setting | SPD_RSM_TIMSET | H'30C | 32 | | √ | √ | √ | √ |
| Overcurrent Detection Timer Setting | OC_TIMSET | H'310 | 32 | | √ | √ | √ | √ |
| SBRN, FLADJ, PORTWAKECAP | SBRN_FLADJ_PW | H'314 | 32 | | √ | √ | √ | √ |

| Register Name | Abbreviation | Address Offset | Access Size | Remarks | Second Generation RZ/G Series Products | | | | |
|-------------------------|--------------|----------------|-------------|--|--|----------------------------|--------|--------|--|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Reserved * | — | H'318 to H'7FC | — | | | | | | |
| Common Control | COMMCTRL | H'800 | 32 | OTG/BC module control register | √ | √ | √ | √ | |
| OTG-BC Interrupt Status | OBINTSTA | H'804 | 32 | OTG/BC interrupt status register | √ | √ | √ | √ | |
| OTG-BC Interrupt Enable | OBINTEN | H'808 | 32 | OTG/BC interrupt enable register | √ | √ | √ | √ | |
| VBUS Control | VBCTRL | H'80C | 32 | OTG VBUS control register | √ | √ | √ | — | |
| Line Control Port1 | LINECTRL1 | H'810 | 32 | OTG USB bus control register (port 1) | √ | √ | √ | √ | |
| Reserved * | — | H'814 to H'81C | — | | | | | | |
| BC Control Port1 | BCCTRL1 | H'820 | 32 | Battery charging control register (port 1) | √ | √ | √ | √ | |
| Reserved * | — | H'824 to H'82C | — | | | | | | |
| ADP Control | ADPCTRL | H'830 | 32 | ADP function control register | √ | √ | √ | √ | |
| Reserved * | — | H'834 to H'BFC | — | | | | | | |

Note: * An undefined value is read. When this area is written to, operation is not defined.

60.1.4 Connected Module

Table 60.6 The connected modules to the EHCI/OHCI

| Module name | Connected module name | Function of connected module |
|-------------|--------------------------------|--|
| EHCI/OHCI | AP-System Core | Access the Register |
| | CPG | Output Clocks |
| | PFC | Select External pins |
| | Module Standby, Software Reset | Control to stop clocks, Execute software reset |
| | INTC, INTC-AP | Control to interrupt |
| | HS-USB | HS-USB module (peripheral) for OTG function |

60.2 Register Description

60.2.1 OHCI Operational Registers

60.2.1.1 HcRevision Register (offset: H'000)

Register symbol: HcRevision

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | Revision | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | R/W | | Initial Value | Description |
|---------|----------|-----|----|---------------|---|
| | | HCD | HC | | |
| 31 to 8 | — | R | R | All 0 | Reserved |
| 7 to 0 | Revision | R | R | H'10 | This field contains the BCD representation of the version of the OHCI specification that is implemented by this host controller core. This core supports OHCI 1.0a and this field shows H'10. |

60.2.1.2 HcControl Register (offset: H'004)

Register symbol: HcControl

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcControl register defines the operating modes as the classic host controller. Most of the fields in this register are modified only by software, except HostControllerFunctionalState and RemoteWakeupConnected.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|-----|-----|-----|------|-----|-----|-----|-----|------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | RWE | RWC | IR | HCFS | BLE | CLE | IE | PLE | CBSR | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|----------|-----|----|---------------|---|
| | | HCD | HC | | |
| 31 to 11 | Reserved | R | R | All 0 | This field shows a BCD encoding of the EHCI revision number supported by this host controller core. This host controller core supports EHCI v.1.1. |
| 10 | RWE | R/W | R | B'0 | RemoteWakeupEnable This bit is used by software to enable or disable the remote wakeup feature upon the detection of upstream resume signaling. When this bit is set and the ResumeDetected bit in HcInterruptStatus is set, a remote wakeup is signaled to the host system. Setting this bit has no impact on the generation of hardware interrupt. 0: An interrupt due to remote wakeup is not generated. 1: An interrupt due to remote wakeup is generated. |
| 9 | RWC | R/W | RW | B'0 | RemoteWakeupConnected This bit indicates whether the host controller supports remote wakeup signaling. Software should set this bit to 1 in the initialization sequence if it is required to support remote wakeup. The host controller clears this bit to 0 upon hardware reset but does not alter it upon software reset. 0: The remote wakeup is not supported. 1: The remote wakeup is supported. |
| 8 | IR | R/W | R | B'0 | InterruptRouting This bit determines the routing of interrupts generated by events registered in HcInterruptStatus. If this bit is 0, all interrupts are routed to the normal host bus interrupt mechanism. If this bit is set to 1, interrupts are routed to the System Management Interrupt. Software clears this bit upon a hardware reset, but it does not alter this bit upon a software reset. |

| Bit | Bit Name | R/W | | Initial Value | Description |
|------|----------|-----|-----|---------------|--|
| | | HCD | HC | | |
| 7, 6 | HCFS | R/W | R/W | B'00 | <p>HostControllerFunctionalState</p> <p>B'00: USB Reset</p> <p>B'01: USB Resume</p> <p>B'10: USB Operational</p> <p>B'11: USB Suspend.</p> <p>This field defines the functional state of the host controller. The transition to USB Operational state from another state causes SOF generation to show 1-ms frame boundary. Software controls this bit basically; this field may be changed by the host controller only when in the USB Suspend state. The host controller may move from the USB Suspend state to the USB Resume state after detecting the resume signaling from a downstream port. The host controller enters USB Suspend state after software reset, whereas it enters USB Reset state after hardware reset.</p> |
| 5 | BLE | R/W | R | B'0 | <p>BulkListEnable</p> <p>This bit is set to 1 in order to enable the processing of the Bulk list in the next Frame. If this bit is 0, processing of the Bulk list does not occur after the next SOF. The host controller checks this bit whenever it determines to process the list.</p> <p>When software modifies the Bulk list, it should set this bit to 0 before modifying the list.</p> <p>0: The processing of the Bulk list is disabled.</p> <p>1: The processing of the Bulk list is enabled.</p> |
| 4 | CLE | R/W | R | B'0 | <p>ControlListEnable</p> <p>This bit is set to 1 in order to enable the processing of the Control list in the next Frame. If this bit is 0, processing of the Control list does not occur after the next SOF. The host controller checks this bit whenever it determines to process the list.</p> <p>When software modifies the Control list, it should set this bit to 0 before modifying the list.</p> <p>0: The processing of the Control list is disabled.</p> <p>1: The processing of the Control list is enabled.</p> |
| 3 | IE | R/W | R | B'0 | <p>IsochronousEnable</p> <p>This bit is set to 1 in order to enable the processing of isochronous EDs. While processing the periodic list in a Frame, the host controller checks the status of this bit when it finds an Isochronous ED (F = 1). If this bit is 1 (enabled), the host controller continues processing the EDs. If this bit is 0 (disabled), the host controller halts processing of the periodic list (which contains only isochronous EDs right now) and begins processing the Bulk/Control lists.</p> <p>Setting this bit is guaranteed to take effect in the next Frame (not the current Frame).</p> <p>0: The processing of the isochronous ED is disabled.</p> <p>1: The processing of the isochronous ED is enabled.</p> |

| Bit | Bit Name | R/W | | Initial Value | Description |
|------|----------|-----|----|---------------|--|
| | | HCD | HC | | |
| 2 | PLE | R/W | R | B'0 | <p>PeriodicListEnable</p> <p>This bit is set to enable the processing of the periodic list in the next Frame. If cleared to 0 by software, processing of the periodic list does not occur after the next SOF. The host controller checks this bit before it starts processing the periodic list.</p> <p>0: The processing of the periodic list is disabled. 1: The processing of the periodic list is enabled.</p> |
| 1, 0 | CBSR | R/W | R | B'00 | <p>ControlBulkServiceRatio</p> <p>This field specifies the service ratio between Control and Bulk EDs.</p> <p>In case of reset, software should restore the value in this field.</p> <p>CBSR No. of Control EDs Over Bulk EDs Served</p> <p>B'00: 1:1 B'01: 2:1 B'10: 3:1 B'11: 4:1</p> |

60.2.1.3 HcCommandStatus Register (offset: H'008)

Register symbol: HcCommandStatus

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcCommandStatus register is used by the host controller to receive commands issued by software, as well as reflecting the current status of the host controller.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SOC | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | OCR | BLF | CLF | HCR |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | W | R/W | R/W | W |

| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|----------|-----|-----|---------------|---|
| | | HCD | HC | | |
| 31 to 18 | — | R | R | All 0 | Reserved |
| 17, 16 | SOC | R | R/W | B'00 | SchedulingOverrunCount This field is incremented on each scheduling overrun error. It is initialized to B'00 and wraps around at B'11. This field is incremented when a scheduling overrun is detected even if SchedulingOverrun in HcInterruptStatus register has already been set. |
| 15 to 4 | — | R | R | All 0 | Reserved |
| 3 | OCR | W | R/W | B'0 | OwnershipChangeRequest This bit is set by software to request a change of control of the host controller. When this bit is set to 1, the host controller sets the OwnershipChange bit in HcInterruptStatus register. After the changeover, this bit is cleared to 0. 0: The change of control of the host controller is not requested. 1: The change of control of the host controller is requested. |

| Bit | Bit Name | R/W | | Initial Value | Description |
|-----|----------|-----|-----|---------------|---|
| | | HCD | HC | | |
| 2 | BLF | R/W | R/W | B'0 | <p>BulkListFilled</p> <p>This bit is used to indicate whether there are any TDs on the Bulk list. It is set by software whenever it adds a TD to an ED in the Bulk list.</p> <p>When the host controller begins to process the head of the Bulk list, it checks BLF bit. As long as BLF bit is 0, the host controller does not start processing the Bulk list. If BLF bit is 1, the host controller starts processing the Bulk list and sets BLF bit to 0. If the host controller finds a TD on the list, then the host controller set BLF bit to 1 causing the Bulk list processing to continue. If no TD is found on the Bulk list, and if software does not set BLF bit, then BLF bit remains at 0 when the host controller completes processing the Bulk list and Bulk list processing will stop.</p> <p>0: TD exists in the Bulk list. 1: TD does not exist in the Bulk list.</p> |
| 1 | CLF | R/W | R/W | B'0 | <p>ControlListFilled</p> <p>This bit is used to indicate whether there are any TDs on the Control list. It is set by software whenever it adds a TD to an ED in the Control list.</p> <p>When the host controller begins to process the head of the Control list, it checks CLF bit. As long as CLF bit is 0, the host controller does not start processing the Control list. If CLF bit is 1, the host controller starts processing the Control list and sets CLF bit to 0. If the host controller finds a TD on the list, then the host controller set CLF bit to 1 causing the Control list processing to continue. If no TD is found on the Control list, and if software does not set CLF bit, then CLF bit remains at 0 when the host controller completes processing the Control list and Control list processing will stop.</p> <p>0: TD exists in the Control list. 1: TD does not exist in the Control list.</p> |
| 0 | HCR | W | R/W | B'0 | <p>HostControllerReset</p> <p>This bit is set by software to initiate software reset of the host controller. Regardless of the functional state of the host controller, it moves to the USB Suspend state.</p> <p>This bit is cleared to 0 by the host controller upon the completion of the reset operation.</p> <p>0: The software reset of the host controller is not requested. 1: The software reset of the host controller is requested.</p> |

60.2.1.4 HcInterruptStatus Register (offset: H'00C)

Register symbol: HcInterruptStatus

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides status on various events that cause hardware interrupts. When an event occurs, the host controller sets the corresponding bit in this register. When a bit becomes set, an interrupt is generated if the interrupt is enabled in the HcInterruptEnable register and the MasterInterruptEnable bit is set.

| | | | | | | | | | | | | | | | | |
|----------------|----|-----|----|----|----|----|----|----|----|------|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | OC | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | RHSC | FNO | UE | RD | SF | WDH | SO |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | R/W | | | Initial Value | Description |
|---------|----------|-----|-----|-------|--|-------------|
| | | HCD | HC | | | |
| 31 | — | R | R | B'0 | Reserved | |
| 30 | OC | R/W | R/W | B'0 | OwnershipChange This bit is set by the host controller when software sets the OwnershipChangeRequest field in HcCommandStatus register. 0: The ownership change has not been requested. 1: The ownership change has been requested. | |
| 29 to 7 | — | R | R | All 0 | Reserved | |
| 6 | RHSC | R/W | R/W | B'0 | RootHubStatusChange This bit is set when the content of HcRhStatus or the content of any of HcRhPortStatus [NumberOfDownstreamPort] has changed. 0: The status of root hub has not changed. 1: The status of root hub has changed. | |
| 5 | FNO | R/W | R/W | B'0 | FrameNumberOverflow This bit is set when the MSB of HcFmNumber (bit 15) changes value, from 0 to 1 or from 1 to 0, and after HccaFrameNumber has been updated. 0: The overflow of frame number has not occurred. 1: The overflow of frame number has occurred. | |
| 4 | UE | R/W | R/W | B'0 | UnrecoverableError This bit is set when the host controller detects a system error not related to USB (system bus error, for example). The host controller should not proceed with any processing nor signaling before the system error has been corrected. Software clears this bit after the host controller has been reset in that case. 0: The unrecoverable error has not occurred. 1: The unrecoverable error has occurred. | |

| Bit | Bit Name | R/W | | Initial Value | Description |
|-----|----------|-----|-----|---------------|---|
| | | HCD | HC | | |
| 3 | RD | R/W | R/W | B'0 | <p>ResumeDetected</p> <p>This bit is set when the host controller detects that USB device is asserting resume signaling. This bit is not set when Software sets the USB Resume state.</p> <p>0: The host controller has not detected resume signaling. 1: The host controller has detected resume signaling.</p> |
| 2 | SF | R/W | R/W | B'0 | <p>StartofFrame</p> <p>This bit is set by the host controller at each start of a frame and after the update of HccaFrameNumber. The host controller also generates a SOF token at the same time.</p> <p>0: The host controller has not started new frame. 1: The host controller has started new frame.</p> |
| 1 | WDH | R/W | R/W | B'0 | <p>WritebackDoneHead</p> <p>This bit is set immediately after the host controller has written HcDoneHead to HccaDoneHead. Further updates of the HccaDoneHead will not occur until this bit has been cleared. Software should only clear this bit after it has saved the content of HccaDoneHead.</p> <p>0: The write back to HccaDoneHead has not occurred. 1: The write back to HccaDoneHead has occurred.</p> |
| 0 | SO | R/W | R/W | B'0 | <p>SchedulingOverrun</p> <p>This bit is set when the USB schedule for the current Frame overruns and after the update of HccaFrameNumber. A scheduling overrun also increments the SchedulingOverrunCount of HcCommandSttus register.</p> <p>0: The scheduling overrun has not occurred. 1: The scheduling overrun has occurred.</p> |

60.2.1.5 HcInterruptEnable Register (offset: H'010)

Register symbol: HcInterruptEnable

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Each enable bit in the HcInterruptEnable register corresponds to an associated interrupt bit in the HcInterruptStatus register. The HcInterruptEnable register is used to control which events generate a hardware interrupt.

A hardware interrupt is generated when all conditions below are satisfied.

1. A bit is set in HcInterruptStatus register
2. The corresponding bit in HcInterruptEnable register is set
3. MIE (MasterInterruptEnable) bit in HcInterruptEnable register is set

Writing 1 to a bit in this register sets the corresponding bit, whereas writing 0 to a bit in this register leaves the corresponding bit unchanged. On read, the current value of this register is returned.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|----|----|----|----|----|----|----|-------|------|-----|-----|-----|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MIE | OCE | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | RHSCE | FNOE | UEE | RDE | SFE | WDHE | SOE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | R/W | | Initial Value | Description |
|---------|----------|-----|----|---------------|--|
| | | HCD | HC | | |
| 31 | MIE | R/W | R | B'0 | <p>MasterInterruptEnable</p> <p>Writing 1 to this bit enables an interrupt generation due to events specified in the other bits of this register. This bit is used by software as a Master interrupt Enable. Writing 0 to this bit is ignored by the host controller.</p> <p>0: An interrupt generation due to events specified in the other bits of this register is disabled.</p> <p>1: An interrupt generation due to events specified in the other bits of this register is enabled.</p> |
| 30 | OCE | R/W | R | B'0 | <p>Writing 1 to this bit enables an interrupt generation due to the request of ownership change. Writing 0 to this bit is ignored by the host controller.</p> <p>0: An interrupt generation due to the request of ownership change is disabled.</p> <p>1: An interrupt generation due to the request of ownership change is enabled.</p> <p>Write 1 to OC bit in HcInterruptDisable register in order to clear this bit.</p> |
| 29 to 7 | — | R | R | All 0 | Reserved |

| Bit | Bit Name | R/W | | Initial Value | Description |
|-----|----------|-----|----|---------------|---|
| | | HCD | HC | | |
| 6 | RHSCE | R/W | R | B'0 | <p>Writing 1 to this bit enables an interrupt generation due to the change of the status of root hub. Writing 0 to this bit is ignored by the host controller.</p> <p>0: An interrupt generation due to the change of the status of root hub is disabled.</p> <p>1: An interrupt generation due to the change of the status of root hub is enabled.</p> <p>Write 1 to RHSC bit in HcInterruptDisable register in order to clear this bit.</p> |
| 5 | FNOE | R/W | R | B'0 | <p>Writing 1 to this bit enables an interrupt generation due to the overflow of frame number. Writing 0 to this bit is ignored by the host controller.</p> <p>0: An interrupt generation due to the overflow of frame number is disabled.</p> <p>1: An interrupt generation due to the overflow of frame number is enabled.</p> <p>Write 1 to FNO bit in HcInterruptDisable register in order to clear this bit.</p> |
| 4 | UEE | R/W | R | B'0 | <p>Writing 1 to this bit enables an interrupt generation due to the unrecoverable error. Writing 0 to this bit is ignored by the host controller.</p> <p>0: An interrupt generation due to the unrecoverable error is disabled.</p> <p>1: An interrupt generation due to the unrecoverable error is enabled.</p> <p>Write 1 to UE bit in HcInterruptDisable register in order to clear this bit.</p> |
| 3 | RDE | R/W | R | B'0 | <p>Writing 1 to this bit enables an interrupt generation due to the detection of resume signaling. Writing 0 to this bit is ignored by the host controller.</p> <p>0: An interrupt generation due to the detection of resume signaling is disabled.</p> <p>1: An interrupt generation due to the detection of resume signaling is enabled.</p> <p>Write 1 to RD bit in HcInterruptDisable register in order to clear this bit.</p> |
| 2 | SFE | R/W | R | B'0 | <p>Writing 1 to this bit enables an interrupt generation due to the start of frame. Writing 0 to this bit is ignored by the host controller.</p> <p>0: An interrupt generation due to the start of frame is disabled.</p> <p>1: An interrupt generation due to the start of frame is enabled.</p> <p>Write 1 to SF bit in HcInterruptDisable register in order to clear this bit.</p> |

| Bit | Bit Name | R/W | | Initial Value | Description |
|-----|----------|-----|----|---------------|---|
| | | HCD | HC | | |
| 1 | WDHE | R/W | R | B'0 | <p>Writing 1 to this bit enables an interrupt generation due to the write back to HccaDoneHead. Writing 0 to this bit is ignored by the host controller.</p> <p>0: An interrupt generation due to the write back to HccaDoneHead is disabled.</p> <p>1: An interrupt generation due to the write back to HccaDoneHead is enabled.</p> <p>Write 1 to WDH bit in HclInterruptDisable register in order to clear this bit.</p> |
| 0 | SOE | R/W | R | B'0 | <p>Writing 1 to this bit enables an interrupt generation due to the scheduling overrun. Writing 0 to this bit is ignored by the host controller.</p> <p>0: An interrupt generation due to the scheduling overrun is disabled.</p> <p>1: An interrupt generation due to the scheduling overrun is enabled.</p> <p>Write 1 to SO bit in HclInterruptDisable register in order to clear this bit.</p> |

60.2.1.6 HcInterruptDisable Register (offset: H'014)

Register symbol: HcInterruptDisable

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Each disable bit in the HcInterruptDisable register corresponds to an associated interrupt bit in the HcInterruptStatus register. The HcInterruptDisable register is coupled with the HcInterruptEnable register. Thus, writing 1 to a bit in this register clears the corresponding bit in the HcInterruptEnable register, whereas writing 0 to a bit in this register leaves the corresponding bit in the HcInterruptEnable register unchanged.

When this register is read, the current value of the HcInterruptEnable register is returned.

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|----|----|----|----|----|----|----|-------|------|-----|-----|-----|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MID | OCD | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | RHSCD | FNOD | UED | RDD | SFD | WDHD | SOD |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | R/W | | Initial Value | Description |
|---------|----------|-----|----|---------------|--|
| | | HWD | HC | | |
| 31 | MID | R/W | R | B'0 | <p>MasterInterruptDisable</p> <p>Writing 1 to this bit disables interrupt generation due to events specified in the other bits of this register.</p> <p>Writing 0 to this bit is ignored by the host controller.</p> <p>When read, the value of MIE bit in HcInterruptEnable register is returned.</p> |
| 30 | OCD | R/W | R | B'0 | <p>Writing 1 disables an interrupt generation due to the request of ownership change.</p> <p>Writing 0 is ignored by the host controller.</p> <p>When read, the value of OC bit in HcInterruptEnable register is returned.</p> |
| 29 to 7 | — | R | R | All 0 | Reserved |
| 6 | RHSCD | R/W | R | B'0 | <p>Writing 1 disables an interrupt generation due to the change of the status of root hub.</p> <p>Writing 0 is ignored by the host controller.</p> <p>When read, the value of RHSC bit in HcInterruptEnable register is returned.</p> |
| 5 | FNOD | R/W | R | B'0 | <p>Writing 1 disables an interrupt generation due to the overflow of frame number.</p> <p>Writing 0 is ignored by the host controller.</p> <p>When read, the value of FNO bit in HcInterruptEnable register is returned.</p> |

| Bit | Bit Name | R/W | | Initial Value | Description |
|-----|----------|-----|----|---------------|---|
| | | HWD | HC | | |
| 4 | UED | R/W | R | B'0 | Writing 1 disables an interrupt generation due to the unrecoverable error. Writing 0 is ignored by the host controller. When read, the value of UE bit in HcInterruptEnable register is returned. |
| 3 | RDD | R/W | R | B'0 | Writing 1 disables an interrupt generation due to the detection of resume signaling. Writing 0 is ignored by the host controller. When read, the value of RD bit in HcInterruptEnable register is returned. |
| 2 | SFD | R/W | R | B'0 | Writing 1 disables an interrupt generation due to the start of frame. Writing 0 is ignored by the host controller. When read, the value of SF bit in HcInterruptEnable register is returned. |
| 1 | WDHD | R/W | R | B'0 | Writing 1 disables an interrupt generation due to the write back to HccaDoneHead. Writing 0 is ignored by the host controller. When read, the value of WDH bit in HcInterruptEnable register is returned. |
| 0 | SOD | R/W | R | B'0 | Writing 1 disables an interrupt generation due to the scheduling overrun. Writing 0 is ignored by the host controller. When read, the value of SO bit in HcInterruptEnable register is returned. |

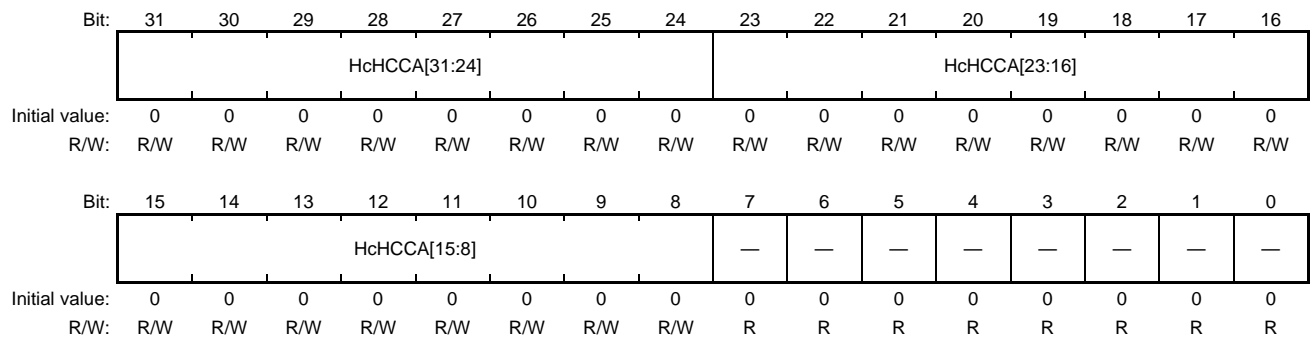
60.2.1.7 HcHCCA Register (offset: H'018)

Register symbol: HcHCCA

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcHCCA register contains the physical address of the Host Controller Communication Area.

As HCCA has 256-byte structure, this host controller core requires 256-byte alignment for HcHCCA register. Therefore, bit 0 through bit 7 always return 0 even after 1 is written to them.



| Bit | Bit Name | R/W | | Initial Value | Description |
|---------|----------|-----|----|---------------|---|
| | | HCD | HC | | |
| 31 to 8 | HcHCCA | R/W | R | H'00_0000 | This field contains the base address of the Host Controller Communication Area. This host controller requires 256-byte alignment for the base address. |
| 7 to 0 | — | R | R | All 0 | Reserved |

60.2.1.8 HcPeriodCurrentED Register (offset: H'01C)

Register symbol: HcPeriodCurrentED

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcPeriodCurrentED register contains the physical address of the current Isochronous or Interrupt Endpoint Descriptor.

| | | | | | | | | | | | | | | | | |
|----------------|-------------|----|----|----|----|----|----|----|-------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | PECD[31:24] | | | | | | | | PECD[23:16] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PECD[15:8] | | | | | | | | PECD[7:4] | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

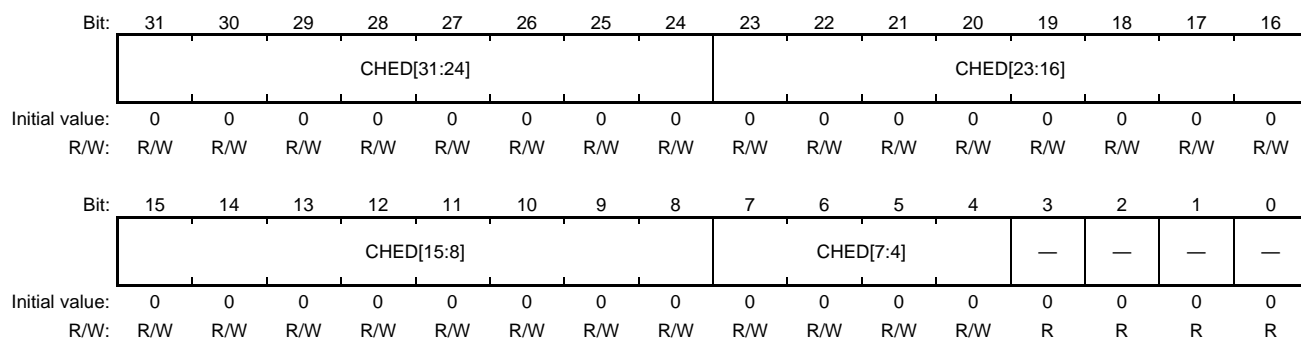
| Bit | Bit Name | R/W | | Initial Value | Description |
|---------|----------|-----|-----|---------------|---|
| | | HCD | HC | | |
| 31 to 4 | PECD | R | R/W | H'000_0000 | PeriodCurrentED This is used by the host controller to point to the head of one of the Periodic lists which will be processed in the current Frame. The content of this register is updated by the host controller after a periodic ED has been processed. |
| 3 to 0 | — | R | R | All 0 | Reserved |

60.2.1.9 HcControlHeadED Register (offset: H'020)

Register symbol: HcControlHeadED

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcControlHeadED register contains the physical address of the first Endpoint Descriptor of the Control list.



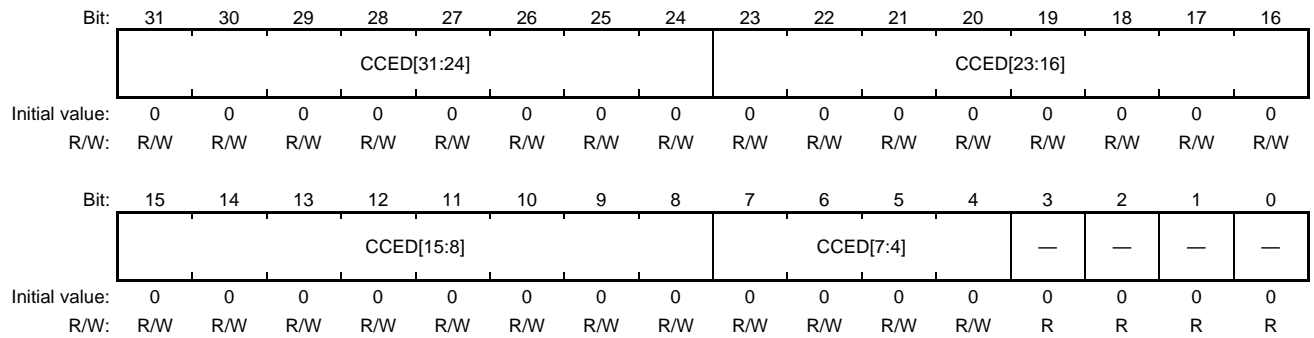
| Bit | Bit Name | R/W | | Initial Value | Description |
|---------|----------|-----|----|---------------|--|
| | | HCD | HC | | |
| 31 to 4 | CHED | R/W | R | H'000_0000 | ControlHeadED The host controller traverses the Control list starting with the HcControlHeadED pointer. The content is loaded from HCCA during the initialization of the host controller. |
| 3 to 0 | — | R | R | All 0 | Reserved |

60.2.1.10 HcControlCurrentED Register (offset: H'024)

Register symbol: HcControlCurrentED

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcControlCurrentED register contains the physical address of the current Endpoint Descriptor of the Control list.



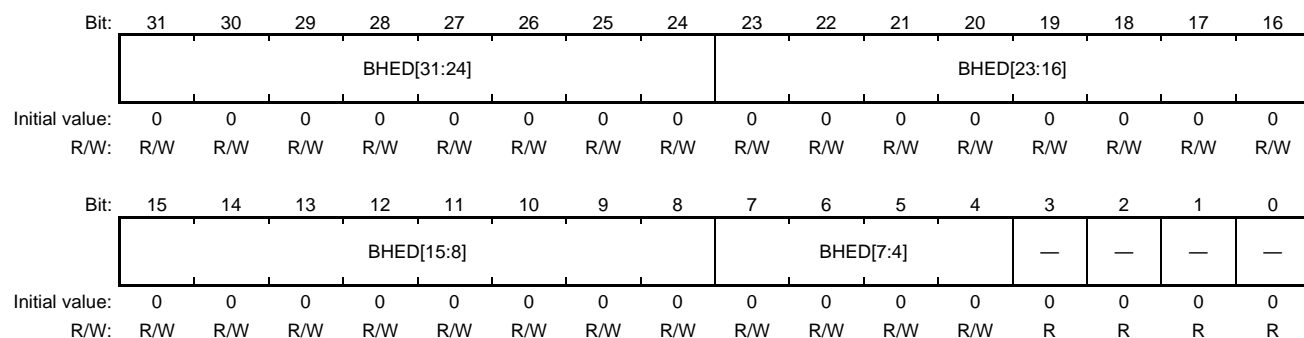
| Bit | Bit Name | R/W | | Initial Value | Description |
|---------|----------|-----|-----|---------------|--|
| | | HCD | HC | | |
| 31 to 4 | CCED | R/W | R/W | H'000_0000 | ControlCurrentED This pointer is advanced to the next ED after serving the present one. The host controller will continue processing the list from where it left off in the last Frame. When it reaches the end of the Control list, the host controller checks the ControlListFilled of in HcCommandStatus. If set, it copies the content of HcControlHeadED to HcControlCurrentED and clears the bit. If not set, it does nothing. Software is allowed to modify this register only when the ControlListEnable of HcControl is cleared. When set, software only reads the instantaneous value of this register. Initially, this is set to zero to indicate the end of the Control list. |
| 3 to 0 | — | R | R | All 0 | Reserved |

60.2.1.11 HcBulkHeadED Register (offset: H'028)

Register symbol: HcBulkHeadED

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcBulkHeadED register contains the physical address of the first Endpoint Descriptor of the Bulk list.



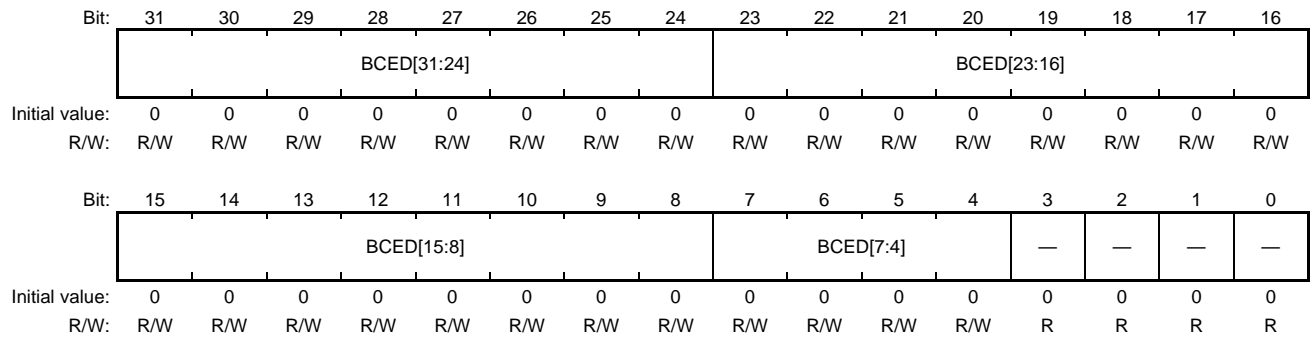
| Bit | Bit Name | R/W | | Initial Value | Description |
|---------|----------|-----|----|---------------|---|
| | | HCD | HC | | |
| 31 to 4 | BHED | R/W | R | H'000_0000 | BulkHeadED The host controller traverses the Bulk list starting with the HcBulkHeadED pointer. The content is loaded from HCCA during the initialization of the host controller. |
| 3 to 0 | — | R | R | All 0 | Reserved |

60.2.1.12 HcBulkCurrentED Register (offset: H'02C)

Register symbol: HcBulkCurrentED

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcBulkCurrentED register contains the physical address of the current endpoint of the Bulk list.



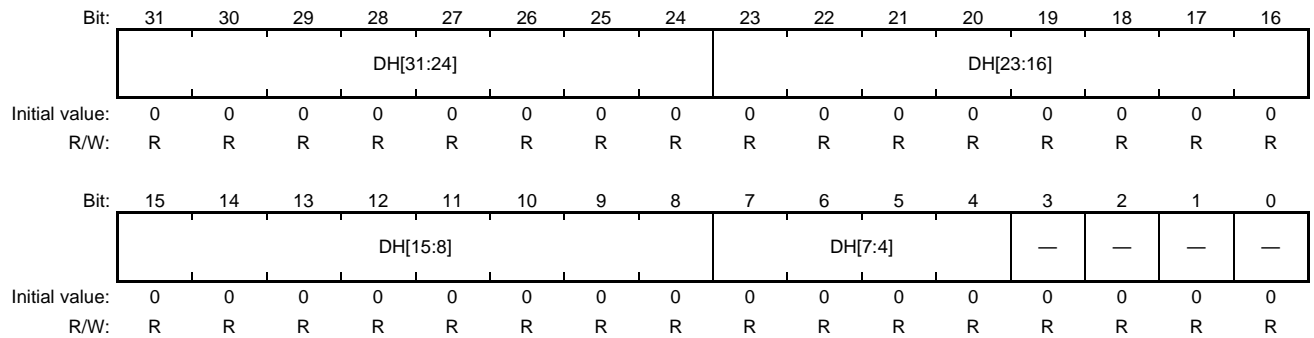
| Bit | Bit Name | R/W | | Initial Value | Description |
|---------|----------|-----|-----|---------------|---|
| | | HCD | HC | | |
| 31 to 4 | BCED | R/W | R/W | H'000_0000 | <p>BulkCurrentED</p> <p>This is advanced to the next ED after the host controller has served the present one. The host controller continues processing the list from where it left off in the last Frame. When it reaches the end of the Bulk list, the host controller checks the ControlListFilled of HcControl. If set, it copies the content of HcBulkHeadED to HcBulkCurrentED and clears the bit. If it is not set, it does nothing.</p> <p>Software is only allowed to modify this register when the BulkListEnable of HcControl is cleared. When set, software only reads the instantaneous value of this register.</p> <p>This is initially set to zero to indicate the end of the Bulk list.</p> |
| 3 to 0 | — | R | R | All 0 | Reserved |

60.2.1.13 HcDoneHead Register (offset: H'030)

Register symbol: HcDoneHead

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcDoneHead register contains the physical address of the last completed Transfer Descriptor that was added to the Done queue. In normal operation, software is not required to read this register as its content is periodically written to the HCCA.



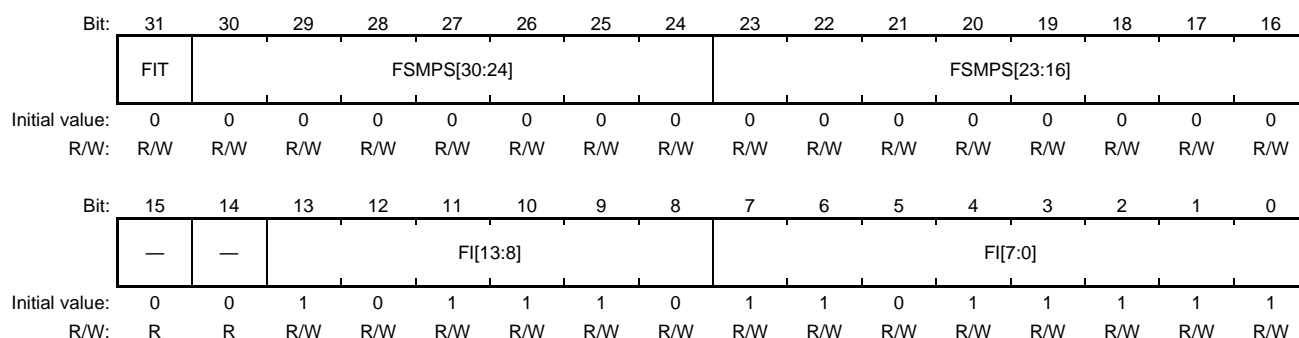
| Bit | Bit Name | R/W | | Initial Value | Description |
|---------|----------|-----|-----|---------------|--|
| | | HCD | HC | | |
| 31 to 4 | DH | R | R/W | H'000_0000 | DoneHead When a TD is completed, the host controller writes the content of HcDoneHead to the NextTD field of the TD. The host controller then overwrites the content of HcDoneHead with the address of this TD. This is set to zero whenever the host controller writes the content of this register to HCCA. It also sets the WritebackDoneHead of HcInterruptStatus. |
| 3 to 0 | — | R | R | All 0 | Reserved |

60.2.1.14 HcFmInterval Register (offset: H'034)

Register symbol: HcFmInterval

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcFmInterval register contains a 14-bit value which indicates the bit time interval in a Frame, (i.e., between two consecutive SOFs), and a 15-bit value indicating the Full Speed maximum packet size that the host controller may transmit or receive without causing scheduling overrun. Software can carry out minor adjustment on the FrameInterval by writing a new value over the present one at each SOF.



| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|----------|-----|----|---------------|---|
| | | HCD | HC | | |
| 31 | FIT | R/W | R | B'0 | <p>FrameIntervalToggle</p> <p>Software writes the toggled value to this bit whenever it writes a new value to FI field.</p> <p>The value in this bit is also loaded to FRT bit in HcFmRemaining register whenever the value in FI field is loaded to FR field of the same register.</p> <p>Software can confirm that the value in FI field is loaded to FR field in HcFmRemaining register by seeing the toggled value in FRT field in the same register.</p> |
| 30 to 16 | FSMPS | R/W | R | H'0000 | <p>FSLargestDataPacket</p> <p>The field specifies the largest amount of data in bits which can be sent or received by the HC in a single transaction at any given time without causing scheduling overrun. The host controller decides whether it can still continue the transfer by comparing the current position in frame with the value in this field.</p> <p>Software calculates the value and set it to this field.</p> |
| 15, 14 | — | R | R | All 0 | Reserved |
| 13 to 0 | FI | R/W | R | H'2EDF | <p>FrameInterval</p> <p>This specifies the interval between two consecutive SOFs in bit times for full speed mode of USB.</p> <p>Use with the value "H'2EDF" (=11,999) to satisfy 1 ms length of frame.</p> |

60.2.1.15 HcFmRemaining Register (offset: H'038)

Register symbol: HcFmRemaining

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcFmRemaining register is a 14-bit down counter showing the bit time remaining in the current Frame.

| | | | | | | | | | | | | | | | | |
|----------------|-----|----|----------|----|----|----|----|---------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | FRT | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | FR[13:8] | | | | | FR[7:0] | | | | | | | | |
| Initial value: | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|----------|-----|-----|---------------|---|
| | | HCD | HC | | |
| 31 | FRT | R | R/W | B'0 | <p>FrameRemainingToggle</p> <p>When FR field reaches 0, the value in FI field in HcFmInterval register is loaded to FR field. At the same time, the value in FIT bit in HcFmInterval register is loaded to this bit.</p> <p>Software can use this bit to confirm that the value in FI field in HcFmInterval register is exactly copied to FR field.</p> |
| 30 to 14 | — | R | R | All 0 | Reserved |
| 13 to 0 | FR | R | R/W | H'2EDF | <p>FrameRemaining</p> <p>This counter is decremented at each bit time. When it reaches 0, the value in FI field in HcFmInterval register is loaded to it at the next bit time boundary.</p> <p>When entering USB Operational state, the host controller re-loads the value in FI field in HcFmInterval register and uses the updated value from the next SOF.</p> |

60.2.1.16 HcFmNumber Register (offset: H'03C)

Register symbol: HcFmNumber

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcFmNumber register is a 16-bit counter. It provides a timing reference among events happening in the host controller and software.

| | | | | | | | | | | | | | | | | |
|----------------|----------|----|----|----|----|----|----|----|---------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | FN[15:8] | | | | | | | | FN[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

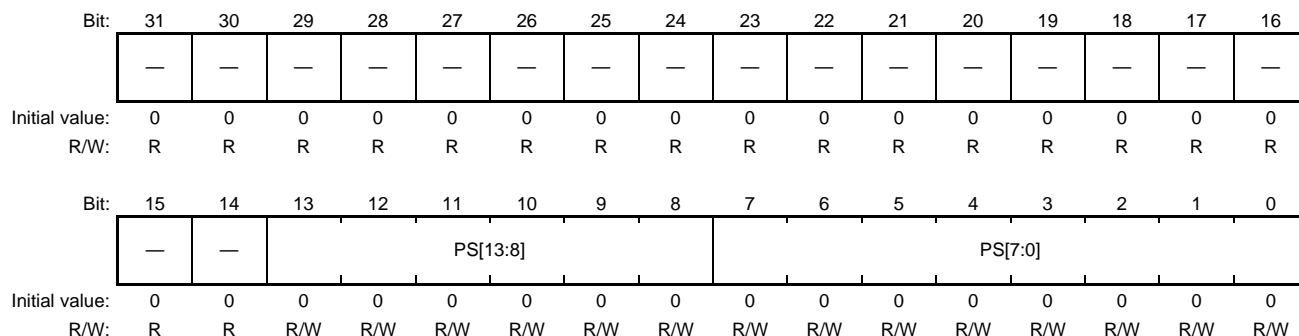
| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|----------|-----|-----|---------------|--|
| | | HCD | HC | | |
| 31 to 16 | — | R | R | All 0 | Reserved |
| 15 to 0 | FN | R | R/W | H'0000 | FrameNumber This field is incremented when HcFmRemaining register is re-loaded. It will be rolled over to H'0 after H'FFFF. When entering the USB Operational state, this will be incremented automatically. The content will be written to HCCA after the host controller has incremented the FrameNumber at each frame boundary and sent a SOF but before the host controller reads the first ED in that Frame. After writing to HCCA, the host controller will set SF bit in HcInterruptStatus register. |

60.2.1.17 HcPeriodicStart Register (offset: H'040)

Register symbol: HcPeriodicStart

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcPeriodicStart register has a 14-bit programmable value which determines when is the earliest time the host controller should start processing the periodic list.



| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|----------|-----|----|---------------|--|
| | | HCD | HC | | |
| 31 to 14 | — | R | R | All 0 | Reserved |
| 13 to 0 | PS | R/W | R | H'0000 | PeriodicStart This field specifies when host controller starts processing the periodic list. Software calculates the adequate value and set it to this field during initialization. The value is calculated roughly as 10% off from FI field in HcFmInterval register. When FR field in HcFmRemaining register reaches the value specified, processing of the periodic lists has priority over Control/Bulk processing. Therefore the host controller starts processing the Interrupt list after completing the current Control or Bulk transaction that is in progress. |

60.2.1.18 HcLSThreshold Register (offset: H'044)

Register symbol: HcLSThreshold

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcLSThreshold register contains an 12-bit value used by the host controller to determine whether to commit to the transfer of a maximum of 8-byte LS packet before EOF. Neither the host controller nor software is allowed to change this value.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-----------|-----|-----|----------|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | LST[11:8] | | | LST[7:0] | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|----------|-----|----|---------------|--|
| | | HCD | HC | | |
| 31 to 12 | — | R | R | All 0 | Reserved |
| 11 to 0 | LST | R/W | R | H'628 | LSThreshold This field specifies the threshold to decide whether the transfer can be still continued in the remaining time of the current frame for LS transfer. Only when the value in FR field of HcFrameRemaining register is larger than that in this field, the host controller can start LS transfer. |

60.2.1.19 HcRhDescriptorA Register (offset: H'048)

Register symbol: HcRhDescriptorA

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcRhDescriptorA register is the first register of two describing the characteristics of the Root Hub.

| | | | | | | | | | | | | | | | | |
|----------------|--------|-----|-----|------|------|-----|-----|-----|-----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | POTPGT | | | | | | | | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | NOCP | OCPM | DT | NPS | PSM | NDP | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R/W | R/W | R | R/W | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|----------|-----|----|---------------|--|
| | | HCD | HC | | |
| 31 to 24 | POTPGT | R/W | R | H'0F | PowerOnToPowerGoodTime This byte specifies the duration the host controller has to wait before accessing a powered-on port of the Root Hub. The unit of time is 2 ms. The duration is calculated as POTPGT * 2 ms. |
| 23 to 13 | — | R | R | All 0 | Reserved |
| 12 | NOCP | R/W | R | B'0 | NoOverCurrentProtection This bit defines how the overcurrent status for the Root Hub ports is reported. When this bit is cleared, OCPM bit specifies global or per-port reporting. 0: Over-current status is reported collectively for all downstream ports 1: No overcurrent protection supported |
| 11 | OCPM | R/W | R | B'1 | OverCurrentProtectionMode This bit defines how the overcurrent status for the Root Hub ports is reported. This bit shows the same mode as PSM bit when it is reset. This bit is valid only if NOCP bit is cleared. 0: over-current status is reported collectively for all downstream ports. 1: over-current status is reported on a per-port basis. |
| 10 | DT | R | R | B'0 | DeviceType This bit is always 0 in order to show that the root hub is not a compound device. |

| Bit | Bit Name | R/W | | Initial Value | Description |
|--------|----------|-----|----|---------------|---|
| | | HCD | HC | | |
| 9 | NPS | R/W | R | B'0 | <p>NoPowerSwitching</p> <p>This bit is used to specify whether power switching is supported or port is always powered.</p> <p>When this bit is cleared, PSM bit specifies global or per-port switching.</p> <p>0: Ports are power switched</p> <p>1: Ports are always powered on when the HC is powered on</p> |
| 8 | PSM | R/W | R | B'1 | <p>PowerSwitchingMode</p> <p>This bit is used to specify how the power switching of the Root Hub ports is controlled.</p> <p>This bit is only valid if the NPS bit is cleared.</p> <p>0: all ports are powered at the same time.</p> <p>1: each port is powered individually.</p> <p>This mode allows port power to be controlled by either the global switch or per-port switching. If the PPCM bit in HcRhDescriptorB register is set, the port responds only to port power commands (Set/ClearPortPower). If the port mask is cleared, then the port is controlled only by the global power switch (Set/ClearGlobalPower).</p> |
| 7 to 0 | NDP | R/W | R | H'01 | <p>NumberDownstreamPorts</p> <p>This field specifies the number of downstream ports supported by the Root Hub.</p> |

60.2.1.20 HcRhDescriptorB Register (offset: H'04C)

Register symbol: HcRhDescriptorB

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcRhDescriptorB register is the second register of two describing the characteristics of the Root Hub.

| | | | | | | | | | | | | | | | | |
|----------------|------------|----|----|----|----|----|----|----|-----------|----|----|----|----|----|-----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | PPCM[15:8] | | | | | | | | PPCM[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DR[15:8] | | | | | | | | DR[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R |

| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|----------|-----|----|---------------|--|
| | | HCD | HC | | |
| 31 to 16 | PPCM | R/W | R | H'0002 | PortPowerControlMask Each bit indicates if a port is affected by a global power control command when PSM bit in HcRhDescriptorA register is set. When set, the port's power state is only affected by per-port power control (Set/ClearPortPower). When cleared, the port is controlled by the global power switch (Set/ClearGlobalPower). If the device is configured to global switching mode (PSM bit = 0), this field is not valid. bit 0: Reserved bit 1: Ganged-power mask on Port #1 bit 15 to 2: Reserved |
| 15 to 0 | DR | R/W | R | H'0000 | DeviceRemovable Each bit is dedicated to a port of the Root Hub. When cleared, the attached device is removable. When set, the attached device is not removable. bit 0: Reserved bit 1: Device attached to Port #1 bit 15 to 2: Reserved |

60.2.1.21 HcRhStatus Register (offset: H'050)

Register symbol: HcRhStatus

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The HcRhStatus register is divided into two parts. The lower word of this dword represents the Hub Status field and the upper word represents the Hub Status Change field. Reserved bits should always be written 0.

| | | | | | | | | | | | | | | | | |
|----------------|------|----|----|----|----|----|----|----|----|----|----|----|----|----|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CRWE | — | — | — | — | — | — | — | — | — | — | — | — | — | OCIC | LPSC |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | W | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | DRWE | — | — | — | — | — | — | — | — | — | — | — | — | — | OCI | LPS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|----------|-----|-----|---------------|--|
| | | HCD | HC | | |
| 31 | CRWE | W | R | B'0 | ClearRemoteWakeupEnable Writing 1 clears DRWE bit. Writing 0 has no effect. |
| 30 to 18 | — | R | R | All 0 | Reserved |
| 17 | OCIC | R/W | R/W | B'0 | OverCurrentIndicatorChange This bit is set by hardware when a change has occurred to the OCI bit. Writing 1 clears this bit when it is set. Writing 0 has no effect. 0: No change has occurred in over-current status. 1: A change has occurred in over-current status. |

| Bit | Bit Name | R/W | | Initial Value | Description | | | | | | | | | | | | | | | | | | | |
|---------------------|----------|---------|----------------|---------------|--|---------------------|-----|---------|-------------|---|---|---|-----------|---|---|---|----------------|---|---|----------------|--|--|---|-----------|
| | | HCD | HC | | | | | | | | | | | | | | | | | | | | | |
| 16 | LPSC | R/W | R | B'0 | <p>This bit has different meanings for read and write.</p> <p>(read) LocalPowerStatusChange</p> <p>This bit always returns 0 for read, since the Root Hub does not support the local power status feature.</p> <p>(write) SetGlobalPower</p> <p>In global power mode (PSM bit = 0 in HcRhDescriptorA register), this bit is written to 1 in order to turn on power to all ports (PPS bit in HcRhPortStatus[N] register is cleared).</p> <p>In per-port power mode, it sets PPS bit in HcRhPortStatus[N] register only on port whose PPCM bit in HcRhDescriptorB register is not set.</p> <p>Writing 0 has no effect.</p> <table border="1"> <thead> <tr> <th>written to this bit</th> <th>PSM</th> <th>PPCM[N]</th> <th>description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>—</td> <td>—</td> <td>no effect</td> </tr> <tr> <td rowspan="2">1</td> <td>0</td> <td>—</td> <td>PPS bit is set</td> </tr> <tr> <td>1</td> <td>0</td> <td>PPS bit is set</td> </tr> <tr> <td></td> <td></td> <td>1</td> <td>no effect</td> </tr> </tbody> </table> | written to this bit | PSM | PPCM[N] | description | 0 | — | — | no effect | 1 | 0 | — | PPS bit is set | 1 | 0 | PPS bit is set | | | 1 | no effect |
| written to this bit | PSM | PPCM[N] | description | | | | | | | | | | | | | | | | | | | | | |
| 0 | — | — | no effect | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | — | PPS bit is set | | | | | | | | | | | | | | | | | | | | | |
| | 1 | 0 | PPS bit is set | | | | | | | | | | | | | | | | | | | | | |
| | | 1 | no effect | | | | | | | | | | | | | | | | | | | | | |
| 15 | DRWE | R/W | R | B'0 | <p>(read) DeviceRemoteWakeupEnable</p> <p>This bit enables a CCS bit in HcRhPortStatus[N] register as a resume event, causing the transition from USB Suspend state to USB Resume state and setting the ResumeDetected interrupt.</p> <p>0: CCS bit does not show a remote wakeup event.</p> <p>1: CCS bit shows a remote wakeup event.</p> <p>(write) SetRemoteWakeupEnable</p> <p>Writing 1 sets this bit.</p> <p>Writing 0 has no effect.</p> | | | | | | | | | | | | | | | | | | | |
| 14 to 2 | — | R | R | All 0 | Reserved | | | | | | | | | | | | | | | | | | | |
| 1 | OCI | R | R | B'0 | <p>OverCurrentIndicator</p> <p>This bit reports overcurrent conditions when the global reporting is implemented.</p> <p>When set, an overcurrent condition exists.</p> <p>When cleared, all power operations are normal.</p> <p>If per-port overcurrent protection is implemented this bit is always '0'</p> <p>0: all power operations are normal</p> <p>1: an overcurrent condition exists.</p> | | | | | | | | | | | | | | | | | | | |

| Bit | Bit Name | R/W | | Initial Value | Description | | | | | | | | | | | | | | | | | | | |
|---------------------|----------|---------|----------------|---------------|---|---------------------|-----|---------|-------------|---|---|---|-----------|---|---|---|----------------|---|---|----------------|--|--|---|-----------|
| | | HCD | HC | | | | | | | | | | | | | | | | | | | | | |
| 0 | LPS | R/W | R/W | B'0 | <p>This bit has different meanings for read and write.</p> <p>(read) LocalPowerStatus</p> <p>This bit always returns 0 for read, since the Root Hub does not support the local power status feature.</p> <p>(write) ClearGlobalPower</p> <p>In global power mode (PSM bit = 0 in HcRhDescriptorA register), this bit is written to 1 in order to turn off power to all ports (PPS bit in HcRhPortStatus[N] register is cleared).</p> <p>In per-port power mode, it clears PPS bit in HcRhPortStatus[N] register only on port whose PPCM bit in HcRhDescriptorB register is not set.</p> <p>Writing 0 has no effect.</p> <table border="1"> <thead> <tr> <th>written to this bit</th> <th>PSM</th> <th>PPCM[N]</th> <th>description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>—</td> <td>—</td> <td>no effect</td> </tr> <tr> <td rowspan="2">1</td> <td>0</td> <td>—</td> <td>PPS bit is set</td> </tr> <tr> <td>1</td> <td>0</td> <td>PPS bit is set</td> </tr> <tr> <td></td> <td></td> <td>1</td> <td>no effect</td> </tr> </tbody> </table> | written to this bit | PSM | PPCM[N] | description | 0 | — | — | no effect | 1 | 0 | — | PPS bit is set | 1 | 0 | PPS bit is set | | | 1 | no effect |
| written to this bit | PSM | PPCM[N] | description | | | | | | | | | | | | | | | | | | | | | |
| 0 | — | — | no effect | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | — | PPS bit is set | | | | | | | | | | | | | | | | | | | | | |
| | 1 | 0 | PPS bit is set | | | | | | | | | | | | | | | | | | | | | |
| | | 1 | no effect | | | | | | | | | | | | | | | | | | | | | |

60.2.1.22 HcRhPortStatus[1] Register (offset: H'054)

Register symbol: HcRhPortStatus1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|------|-----|----|----|----|------|------|------|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | PRSC | OCIC | PSSC | PESC | CSC |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | LSDA | PPS | — | — | — | PRS | POCI | PSS | PES | CCS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|----------|-----|-----|---------------|--|
| | | HCD | HC | | |
| 31 to 21 | — | R | R | All 0 | Reserved |
| 20 | PRSC | R/W | R/W | B'0 | PortResetStatusChange This bit shows the port reset has been completed. This bit is set at the end of the 10-ms port reset signal. Writing 1 clears this bit. Writing 0 has no effect. 0: port reset has not been completed. 1: port reset has been completed |
| 19 | OCIC | R/W | R/W | B'0 | PortOverCurrentIndicatorChange This bit shows that overcurrent condition has been detected, and is set when the host controller changes POCI bit. This bit is valid only if overcurrent conditions are reported on a per-port basis. Writing 1 clears this bit. Writing 0 has no effect. 0: no change has occurred in POCI bit. 1: change has occurred in POCI bit. |
| 18 | PSSC | R/W | R/W | B'0 | PortSuspendStatusChange This bit is set when the full resume sequence has been completed. This sequence includes the 20-s resume pulse, LS EOP, and 3-ms resynchronization delay. Writing 1 clears this bit. Writing 0 has no effect. This bit is also cleared when PRSC bit is set. 0: resume sequence has not been completed. 1: resume sequence has been completed. |
| 17 | PESC | R/W | R/W | B'0 | PortEnableStatusChange This bit is set when hardware events cause PES bit to be cleared. Writing 1 clears this bit. Writing 0 has no effect. 0: no change has occurred in PES bit 1: change has occurred in PES bit |

| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|----------|-----|-----|---------------|--|
| | | HCD | HC | | |
| 16 | CSC | R/W | R/W | B'0 | <p>ConnectStatusChange</p> <p>This bit is set whenever connect or disconnect event occurs.</p> <p>Writing 1 means to clear this bit. Writing 0 has no effect.</p> <p>If CCS bit is cleared and writing to SetPortReset, SetPortEnable, or SetPortSuspend occurs, this bit is set to force the driver to re-evaluate the connection status since these write should not occur if the port is disconnected.</p> <p>0: no change has occurred in CCS bit 1: change has occurred in CCS bit</p> |
| 15 to 10 | — | R | R | All 0 | Reserved |
| 9 | LSDA | R/W | R/W | B'0 | <p>This bit has different meanings for read and write.</p> <p>(read) LowSpeedDeviceAttached</p> <p>This bit indicates the speed of the device attached to this port.</p> <p>When set, a Low Speed device is attached to this port. When cleared, a Full Speed device is attached to this port. This field is valid only when the CCS bit is set (which means device is connected).</p> <p>0: full speed device attached 1: low speed device attached</p> <p>(write) ClearPortPower</p> <p>Writing 1 clears PPS bit. Writing 0 has no effect.</p> |
| 8 | PPS | R/W | R/W | B'0 | <p>This bit has different meanings for read and write.</p> <p>(read) PortPowerStatus</p> <p>This bit reflects the port's power status.</p> <p>This bit is cleared if an overcurrent condition is detected.</p> <p>0: port power is off. 1: port power is on.</p> <p>(write) SetPortPower</p> <p>Writing 1 sets PPS bit. Writing 0 has no effect.</p> <p>The HCD writes a 1 to set the PortPowerStatus bit. Writing 0 has no effect.</p> |
| 7 to 5 | — | R | R | All 0 | Reserved |

| Bit | Bit Name | R/W | | Initial Value | Description |
|-----|----------|-----|-----|---------------|---|
| | | HCD | HC | | |
| 4 | PRS | R/W | R/W | B'0 | <p>This bit has different meanings for read and write.</p> <p>(read) PortResetStatus</p> <p>When this bit is set by writing to SetPortReset, port reset signaling is asserted. When reset is completed, this bit is cleared when PRSC bit is set. This bit cannot be set if CCS bit is cleared (which means no device is connected)</p> <p>0: port reset signal is not active 1: port reset signal is active</p> <p>(write) SetPortReset</p> <p>Writing 1 asserts the port reset signaling. Writing 0 has no effect.</p> <p>When CCS bit is cleared (which means no device is connected), PRS bit is not set even with writing 1. But it causes CSC bit to be set instead.</p> <p>This informs that a disconnected port is attempted to be reset.</p> |
| 3 | POCI | R/W | R/W | B'0 | <p>This bit has different meanings for read and write.</p> <p>(read) PortOverCurrentIndicator</p> <p>This bit is only valid when the host controller is configured in such a way that overcurrent conditions are reported on a per-port basis. If per-port overcurrent reporting is disabled, this bit always returns 0 for read. If cleared, all power operations are normal for this port. If set, an overcurrent condition exists on this port.</p> <p>0: no overcurrent condition exists 1: overcurrent condition exists.</p> <p>(write) ClearSuspendStatus</p> <p>Writing 1 initiates the resume sequence. Writing 0 has no effect.</p> <p>A resume is initiated only if PSS bit is set.</p> |
| 2 | PSS | R/W | R/W | B'0 | <p>This bit has different meanings for read and write.</p> <p>(read) PortSuspendStatus</p> <p>This bit indicates the port is suspended or in the resume sequence.</p> <p>This bit cannot be set when CCS bit is cleared (which means no device is connected).</p> <p>This bit is cleared with following conditions.</p> <p>The resume sequence is completed and PSSC bit is set. The port reset is completed and PRSC bit asset. The host controller is placed in USB Resume state.</p> <p>0: Port is not suspended. 1: Port is suspended.</p> <p>(write) SetPortSuspend</p> <p>Writing 1 sets PSS bit. Writing 0 has no effect.</p> <p>When CCS bit is cleared (which means no device is connected), PSS bit is not set even with writing 1. But it causes CSC bit to be set instead.</p> <p>This informs that a disconnected port is attempted to be suspended.</p> |

| Bit | Bit Name | R/W | | Initial Value | Description |
|-----|----------|-----|-----|---------------|---|
| | | HCD | HC | | |
| 1 | PES | R/W | R/W | B'0 | <p>This bit has different meanings for read and write.</p> <p>(read) PortEnableStatus</p> <p>This bit indicates whether the port is enabled or disabled. This host controller core clears this bit in case of an overcurrent condition, disconnect event, switched-off power, or operational bus error such as babble is detected.</p> <p>This change also sets PESC bit.</p> <p>This bit cannot be set when CCS bit is cleared (which means no device is connected).</p> <p>This bit is also set when port reset is completed and PRSC bit is set or when port is suspended and PSSC bit is set.</p> <p>0: Port is disabled 1: Port is enabled.</p> <p>(write) SetPortEnable</p> <p>Writing 1 sets PES bit. Writing 0 has no effect.</p> <p>If CCS bit is cleared, this write does not set PES bit, but instead sets CSC bit. This informs the driver that it attempted to enable a disconnected port.</p> |
| 0 | CCS | R/W | R/W | B'0 | <p>This bit has different meanings for read and write.</p> <p>(read) CurrentConnectStatus</p> <p>This bit reflects the current state of the downstream port.</p> <p>0: no device is connected 1: device is connected</p> <p>(write) ClearPortEnable</p> <p>Writing 1 to this bit clears PES bit. Writing 0 has no effect.</p> |

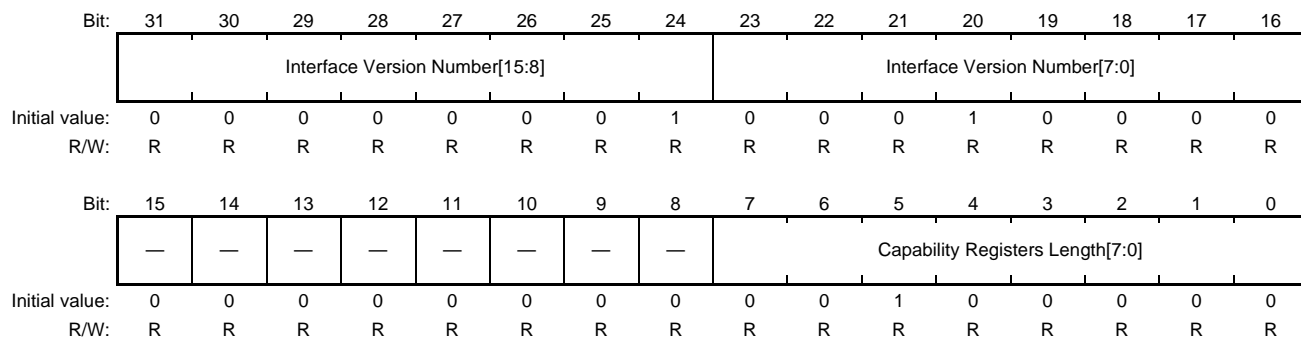
60.2.2 EHCI Controller Capability Registers

60.2.2.1 HCIVERSION/CAPLENGTH Register (offset: H'100)

Register symbol: CAPL_VERSION

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register is used as an offset to add to register base to find the beginning of the operational register space.



| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|-----------------------------|-----|----|---------------|---|
| | | HCD | HC | | |
| 31 to 16 | Interface Version Number | R | R | H'0110 | This field shows a BCD encoding of the EHCI revision number supported by this host controller core. This host controller core supports EHCI v.1.1. |
| 15 to 8 | — | R | R | All 0 | Reserved |
| 7 to 0 | Capability Registers Length | R | R | H'20 | This field is used as an offset to add to register base to find the beginning of the operational register space. This field is prepared considering the compatibility with EHCI specification. This field has the value of H'20 since the operational register starts from H'20 offset. |

60.2.2.2 HCSPARAMS Register (offset: H'104)

Register symbol: HCSPARAMS

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------|----|----|-------|----|----|----|--------------|------------------------|----|-----|--------------|----|----|-------------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | Debug Port Number[3:0] | | | — | — | — | P_INDICATOR | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | N_CC | | | N_PCC | | | | Port Routing | — | — | PPC | N_PORTS[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|--------------------|-----|----|---------------|--|
| | | HCD | HC | | |
| 31 to 24 | — | R | R | All 0 | Reserved |
| 23 to 20 | Debug Port Number | R | R | H'0 | This field always shows H'0 since this host controller core has no debug port. |
| 19 to 17 | — | R | R | All 0 | Reserved |
| 16 | P_INDICATOR | R | R | B'0 | Port Indicators (P_INDICATOR). This field always shows 0 since this host controller core does not have the function of port indicator. |
| 15 to 12 | N_CC | R | R | H'1 | Number of Companion Controller This field indicates the number of companion controllers. |
| 11 to 8 | N_PCC | R | R | H'1 | Number of Ports per Companion Controller (N_PCC). This field indicates the number of ports supported per companion host controller. It is used to indicate the port routing configuration to software. |
| 7 | Port Routing Rules | R | R | B'1 | Port Routing Rules. This field indicates the method used by this implementation for how all ports are mapped to companion controllers. 0: The first N_PCC ports are routed to the lowest numbered function companion host controller, the next N_PCC port are routed to the next lowest function companion controller, and so on. 1: The port routing is explicitly enumerated by the first N_PORTS elements of the HCSP-PORTROUTE array. |
| 6, 5 | — | R | R | All 0 | Reserved |
| 4 | PPC | R | R | B'1 | Port Power Control (PPC). This field indicates whether the host controller implementation includes port power control. |
| 3 to 0 | N_PORTS | R | R | H'1 | This field specifies the number of physical downstream ports implemented as host controller |

60.2.2.3 HCCPARAMS Register (offset: H'108)

Register symbol: HCCPARAMS

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------|----|----|----|----|----|----|----|----------------------------------|----|----|----|---|--|--|------------------------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | 32-Frame Periodic List Capability | Per-Port Change Event Capability | Link Power Management Capability | Hardware Prefetch Capability |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | EECP | | | | | | | | Isochronous Scheduling Threshold | | | | — | Asynchronous Schedule Park Capability | Programmable Frame List Flag | 64-bit Addressing Capability |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|-----------------------------------|-----|----|---------------|---|
| | | HCD | HC | | |
| 31 to 20 | — | R | R | All 0 | Reserved |
| 19 | 32-Frame Periodic List Capability | R | R | B'1 | This bit shows 1 since this host controller has 32-Frame Periodic List Capability. When set (1) this optional field indicates the host controller supports a 32 frame periodic schedule as specified by using the value B'11 in the USBCMD Frame List Size field. Software must treat a Flame List Size values (B'01 = 512 frames, B'10 = 256 frames) continues to be indicated through the Programmable Frame List Flag, where this bit only indicates programmability for a 32-frame list. |
| 18 | Per-Port Change Event Capability | R | R | B'1 | This bit shows 1 since this host controller has Per-Port Change Event Capability. When set (1) this optional field indicates host controller support for pre-port change events and associated USBCMD Pre-Port Change Event Enable, USBSTS Port-n Change Detect, and USBINT Port-n Change Interrupt Enable fields. Note that software should treat those fields as reserved when this bit is cleared (0). |
| 17 | Link Power Management Capability | R | R | B'0 | This bit shows 0 since this host controller doesn't have Link Power Management Capability. When set (1) this optional field indicates host controller support for this the Link Power Management L1 state and associated PORTSC Suspend using L1, Suspend Status, and Device Address field. Note that software should treat those fields as reserved then this bit is cleared (0). |
| 16 | Hardware Prefetch Capability | R | R | B'0 | This bit shows 0 since this host controller does not have Hardware Prefetch Capability. This host controller doesn't support this function. |
| 15 to 8 | EECP | R | R | H'00 | EHCI Extended Capabilities Pointer (EECP). |
| 7 to 4 | Isochronous Scheduling Threshold | R | R | H'0 | This field shows 0 since this host controller core does not support the function to cache the isochronous data structure. |
| 3 | — | R | R | B'0 | Reserved |

| Bit | Bit Name | R/W | | Initial Value | Description |
|-----|---------------------------------------|-----|----|---------------|---|
| | | HCD | HC | | |
| 2 | Asynchronous Schedule Park Capability | R | R | B'1 | This bit shows 1 since this host controller core has Asynchronous Schedule Park Capability. |
| 1 | Programmable Frame List Flag | R | R | B'1 | This bit shows 1 since this host controller core supports Programmable Frame List. |
| 0 | 64-bit addressing Capability | R | R | B'0 | This field shows 0 since this host controller core does not support 64-bit addressing. |

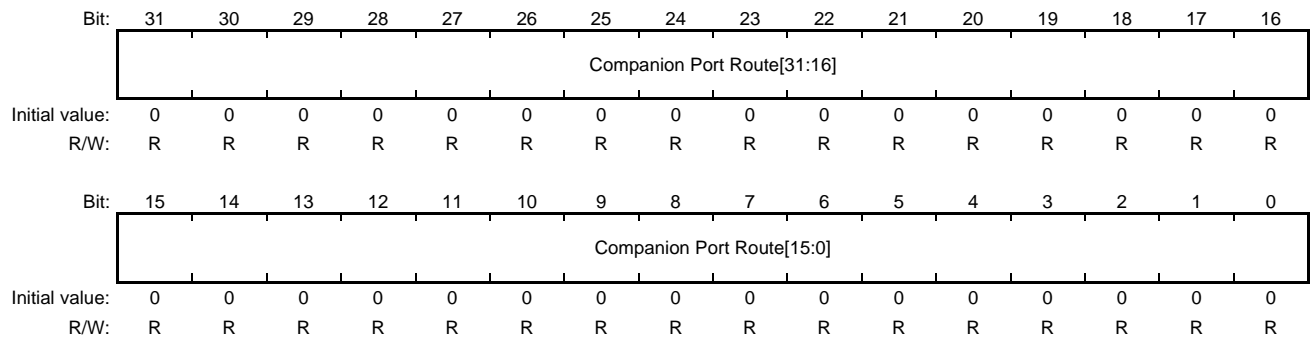
60.2.2.4 HCSP-PORTROUTE Register (offset: H'10C)

Register symbol: HCSP_PORTROUTE

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This operational register is valid only if Port Routing Rules field in HCSPARAMs register.

In that case, this register shows how OHCI controller is routed to each port.



| Bit | Bit Name | R/W | Initial Value | Description |
|---------|----------------------------|-----|---------------|---|
| 31 to 0 | Companion Port Route[31:0] | R | H'0000_0000 | These bits show the port controlled by the OHCI host controller. This module provides only one OHCI host controller and this bit is read as 0. |

60.2.3 EHCI Operational Registers

60.2.3.1 USBCMD Register (offset: H'120)

Register symbol: USBCMD

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The Command Register indicates the command to be executed by the serial bus host controller. Writing to the register causes a command to be executed.

| | | | | | | | | | | | | | | | | |
|----------------|-------------------------------|----------------------------|---------------------------------------|-----------------------------------|--|-----|---------------------------------------|-----------------------------|--------------------------------------|------------------------------|--------------------------|-----------------|-----|----------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | Host-Initiated Resume Duration | | | | Interrupt Threshold Control | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Per-Port Change Events Enable | Full Synchronized Prefetch | Asynchronous Schedule Prefetch Enable | Periodic Schedule Prefetch Enable | Asynchronous Schedule Park Mode Enable | — | Asynchronous Schedule Park Mode Count | Light Host Controller Reset | Interrupt on Async Advance Door-bell | Asynchronous Schedule Enable | Periodic Schedule Enable | Frame List Size | | HCRES ET | RS | |
| Initial value: | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R/W | R | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|--------------------------------|-----|----|---------------|---|
| | | HCD | HC | | |
| 31 to 28 | — | R | R | All 0 | Reserved |
| 27 to 24 | Host-Initiated Resume Duration | R/W | R | H'0 | <p>This field is used by software to specify the minimum amount of time the host controller will drive the K-state during a host-initiated resume from LPM (L1) state and is conveyed to each LPM-enabled device (via the HIRD bits within an LPM Token's bmAttributes field) upon entry into a low-power state.</p> <p>Encoding for this field is identical to the definition for the similarly named HIRD field within an LPM Token. Specifically, a value H'0 equals to 50 μs and each additional increment adds 75 μs. For example, the value H'1 equals 125 μs and H'F equals 1175 μs.</p> |
| 23 to 16 | Interrupt Threshold Control | R/W | R | H'08 | <p>This field is used by system software to select the maximum rate at which the host controller core issue interrupts. If software writes an invalid value to this field, the results are undefined.</p> <p>Value Maximum Interrupt Interval</p> <p>H'00: Reserved</p> <p>H'01: 1 micro-frame</p> <p>H'02: 2 micro-frames</p> <p>H'04: 4 micro-frames</p> <p>H'08: 8 micro-frames (default, equals to 1 MS)</p> <p>H'10: 16 micro-frames (2 MS)</p> <p>H'20: 32 micro-frames (4 MS)</p> <p>H'40: 64 micro-frames (8 MS)</p> <p>Software should not set this bit to 0 while Halted bit is equal to 0.</p> |

| Bit | Bit Name | R/W | | Initial Value | Description |
|------|--|-----|-----|---------------|---|
| | | HCD | HC | | |
| 15 | Per-Port Change Events Enable | R/W | R | B'0 | This field is used by software to enable the per-port change event capability as defined by the Port-n Change Detect field in the USBSTS register and Port-n Change Detect Enable field in the USBINTR register. When this bit is set to 1, the per-port change event is enabled. When this bit is set to 0, the per-port change event is disabled. |
| 14 | Full Synchronized Prefetch | R | R | B'0 | Since this host controller core does not have Hardware Prefetch Capability, this bit is fixed to 0. |
| 13 | Asynchronous Schedule Prefetch Enable | R | R | B'0 | Since this host controller core does not have Hardware Prefetch Capability, this bit is fixed to 0. |
| 12 | Periodic Schedule Prefetch Enable | R | R | B'0 | Since this host controller core does not have Hardware Prefetch Capability, this bit is fixed to 0. |
| 11 | Asynchronous Schedule Park Mode Enable | R/W | R | B'1 | Software uses to enable or disable Asynchronous Schedule Park Mode. When this bit is set to 1, Park mode is enabled. When this bit is set to 0, Park mode is disabled. |
| 10 | — | R | R | B'0 | Reserved |
| 9, 8 | Asynchronous Schedule Park Mode Count | R/W | R | B'11 | This field defines a count of the number of successive transactions the host controller core is allowed to execute from a high-speed queue head on the Asynchronous schedule before continuing traversal of the Asynchronous schedule. This function is enabled when Asynchronous Schedule Park Mode Enable is set to 1. Software should not write 0 to this bit. |
| 7 | Light Host Controller Reset | R | R | B'0 | Since this host controller core does not support the function of Light Host Controller Reset, this bit is fixed to 0. |
| 6 | Interrupt on Async Advance Doorbell | R/W | R/W | B'0 | This bit is used as a doorbell by software to tell the host controller to issue an interrupt the next time it advances asynchronous schedule. Software must write 1 to this bit to ring the doorbell. When the host controller has evicted all appropriate cached schedule state, it sets the Interrupt on Async Advance status bit in the USBSTS register. If the Interrupt on Async Advance Enable bit in the USBINTR register is 1 then the host controller will assert an interrupt at the next interrupt threshold. The host controller sets this bit to 0 after it has set the Interrupt on Async Advance status bit in the USBSTS register to 1. Software should not write 1 to this bit when the asynchronous schedule is disabled. |
| 5 | Asynchronous Schedule Enable | R/W | R | B'0 | This bit controls whether the host controller skips processing the Asynchronous Schedule. 0: Do not process the Asynchronous Schedule 1: Use the ASYNCLISTADDR register to access the Asynchronous Schedule. |

| Bit | Bit Name | R/W | | Initial Value | Description |
|------|--------------------------|-----|-----|---------------|---|
| | | HCD | HC | | |
| 4 | Periodic Schedule Enable | R/W | R | B'0 | <p>This bit controls whether the host controller skips processing the Periodic Schedule.</p> <p>0: Do not process the Periodic Schedule</p> <p>1: Use the PERIODICLISTBASE register to access the Periodic Schedule.</p> |
| 3, 2 | Frame List Size | R/W | R | B'00 | <p>This field specifies the size of the frame list. The size the frame list and controls which bits in the Frame Index Register should be used for the Frame List Current index. This host controller core has 32-Frame Periodic List Capability.</p> <p>Value the number of frames in Frame List</p> <p>B'00: 1024 frames (default)</p> <p>B'01: 512 frames</p> <p>B'10: 256 frames</p> <p>B'11: 32 frames</p> |
| 1 | HCRESET | R/W | R/W | B'0 | <p>Host Controller Reset (HCRESET).</p> <p>This control bit is used by software to reset the host controller.</p> <p>When software writes 1 to this bit, the Host Controller resets its internal pipelines, timers, counters, state machines, etc. to their initial value. Any transaction currently in progress on USB is immediately terminated. A USB reset is not driven on downstream ports.</p> <p>All registers except EHCI operational registers are not affected by this reset. All operational registers, including port registers and port state machines are set to their initial values. Port ownership reverts to the companion host controller(s). Software must reinitialize the host controller in order to return the host controller to an operational state.</p> <p>This bit is set to 0 by the Host Controller when the reset process is complete.</p> <p>Software cannot terminate the reset process early by writing 0 to this register.</p> <p>Software should not set this bit to 1 when the HCHalted bit in the USBSTS register is 0.</p> |
| 0 | RS | R/W | R/W | B'0 | <p>Run/Stop (RS).</p> <p>When this bit is set to 1, the host controller proceeds with execution of the schedule. The host controller continues execution as long as this bit is set to 1.</p> <p>When this bit is set to 0, the host controller completes the current and any actively pipelined transactions on the USB and then halts. The HCHalted bit in the status register indicates when the host controller has finished its pending pipelined transactions and has entered the stopped state.</p> <p>Software should not write 1 to this field unless the host controller is in the Halted state (i.e. HCHalted in the USBSTS register is 1).</p> |

60.2.3.2 USBSTS Register (offset: H'124)

Register symbol: USBSTS

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register indicates pending interrupts and various states of the host controller. The status resulting from a transaction on the serial bus is not indicated in this register.

| | | | | | | | | | | | | | | | | |
|----------------|------------------------------|--------------------------|-------------|----------|----|----|----|----|---------------------------|----|----------------------------|-------------------|---------------------|--------------------|------------|--------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Port-n Change Detect[15:8] | | | | | | | | Port-n Change Detect[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Asynchronous Schedule Status | Periodic Schedule Status | Reclamation | HCHalted | — | — | — | — | — | — | Interrupt on Async Advance | Host System Error | Frame List Rollover | Port Change Detect | USBERR INT | USBINT |
| Initial value: | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|------------------------------|-----|-----|---------------|--|
| | | HCD | HC | | |
| 31 to 16 | Port-n Change Detect | R/W | R/W | H'0000 | <p>The definition of each bit is identical to the Port Change Detect field (bit 2 of this register) except these bits are specific to a given port, where bit 16 = Port 1, 17 = Port 2, etc. For example, if bit 17 is set to 1 then a port change event was detected on Port 2, the N_PORTS field in HCSPARAMS specifies how many ports are exposed by the host controller and thus how many bits in this field are valid.</p> <p>The behavior of the Port Change Detect and related fields even when Per-Port Change Events are enabled.</p> <p>This field is only used by software when the Per-Port Change Events Enable bit within the USBCMD register is set to 1.</p> <p>bit 31 to 17: Reserved</p> |
| 15 | Asynchronous Schedule Status | R | R/W | B'0 | <p>This bit reports the current real status of the Asynchronous Schedule.</p> <p>If this bit is 0 then the status of the Asynchronous Schedule is disabled. If this bit is 1 then the status of the Asynchronous Schedule is enabled.</p> <p>When this bit and the Asynchronous Schedule Enable bit are the same value, the Asynchronous Schedule is either enabled (1) or disabled (0).</p> |
| 14 | Periodic Schedule Status | R | R/W | B'0 | <p>This bit reports the current real status of the Periodic Schedule.</p> <p>If this bit is 0 then the status of the Periodic Schedule is disabled. If this bit is 1 then the status of the Periodic Schedule is enabled.</p> <p>When this bit and the Periodic Schedule Enable bit are the same value, the Periodic Schedule is either enabled (1) or disabled (0).</p> |

| Bit | Bit Name | R/W | | Initial Value | Description |
|---------|----------------------------|-----|-----|---------------|--|
| | | HCD | HC | | |
| 13 | Reclamation | R | R/W | B'0 | This is a read-only status bit, which is used to detect an empty asynchronous schedule. When this bit is 1, asynchronous schedule is empty. |
| 12 | HCHalted | R | R/W | B'1 | This bit is 0 whenever the Run/Stop bit is 1. The host controller sets this bit to 1 after it has stopped executing as a result of the Run/Stop bit being set to 0, either by software or by the host controller hardware (e.g. internal error). |
| 11 to 6 | — | R | R/W | All 0 | Reserved |
| 5 | Interrupt on Async Advance | R/W | R/W | B'0 | System software can force the host controller to issue an interrupt the next time the host controller advances the asynchronous schedule by writing 1 to the Interrupt on Async Advance Doorbell bit in the USBCMD register. This status bit indicates the assertion of that interrupt source. This bit is cleared to 0 by writing 1 to it. Writing 0 occurs no effect. |
| 4 | Host System Error | R/W | R/W | B'0 | The host controller sets this bit to 1 when a serious error occurs. When this error occurs, the host controller clears the Run/Stop bit in the Command register to prevent further execution of the scheduled TDs. This bit is cleared to 0 by writing 1 to it. Writing 0 occurs no effect. |
| 3 | Frame List Rollover | R/W | R/W | B'0 | The host controller sets this bit to 1 when a frame list rollover event occurs. The exact Frame List Index value at which the rollover occurs depends on the frame list size, nothing this generally occurs when FRINDEX rolls over from its maximum value to 0. For example, if the frame list size (as programmed in the Frame List Size field of the USBCMD register) is 1024 then a frame list rollover would occur every time FRINDEX[13] toggles. Similarly, a frame list rollover event would occur every time FRINDEX[12] toggles for a 512 frame list, and when FRINDEX[11] toggles for a 256 frame list. Note this behavior is different for a 32 frame list where a rollover event occurs every time FRINDEX[13] toggles (same as 1024). This bit is cleared to 0 by writing 1 to it. Writing 0 occurs no effect. |
| 2 | Port Change Detect | R/W | R/W | B'0 | The host controller sets this bit to 1 when it detects any change of port status. The detailed conditions those set this bit to 1 are as follows. The connection or disconnection of device is detected and Connect Status Change bit (bit 1) of PORTSC[n] register is changed from 0 to 1. The enabled status of port is detected and Port Enable/Disable bit (bit 3) of PORTSC[n] register is changed from 0 to 1. The over current is detected and Over-current Change bit (bit 5) of PORTSC[n] register is changed from 0 to 1. J-K transition is detected on the suspended port and Force Port Resume bit (bit 6) of PORTSC[n] register is changed from 0 to 1. This bit is cleared to 0 by writing 1 to it. Writing 0 occurs no effect. |

| Bit | Bit Name | R/W | | Initial Value | Description |
|-----|-----------|-----|-----|---------------|--|
| | | HCD | HC | | |
| 1 | USBERRINT | R/W | R/W | B'0 | <p>USB Error Interrupt (USBERRINT).</p> <p>The host controller sets this bit to 1 when completion of a USB transaction results in an error condition. If the TD on which the error interrupt occurred also had its IOC bit set, both this bit and USBINT bit are set.</p> <p>This bit is cleared to 0 by writing 1 to it. Writing 0 occurs no effect.</p> |
| 0 | USBINT | R/W | R/W | B'0 | <p>USB Interrupt (USBINT).</p> <p>The host controller sets this bit to 1 on the completion of a USB transaction, which results in the retirement of a Transfer Descriptor that had its IOC bit set.</p> <p>The host controller also sets this bit to 1 when a short packet is detected (actual number of bytes received was less than the expected number of bytes).</p> <p>This bit is cleared to 0 by writing 1 to it. Writing 0 occurs no effect.</p> |

60.2.3.3 USBINTR Register (offset: H'128)

Register symbol: USBINTR

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register enables and disables reporting of the corresponding interrupt to the software. When a bit is set and the corresponding interrupt is active, an interrupt is generated to the host. Interrupt sources that are disabled in this register still appear in the USBSTS to allow the software to poll for events.

Each interrupt enable bit description indicates whether it is dependent on the interrupt threshold mechanism.

| | | | | | | | | | | | | | | | | |
|----------------|----------------------------------|----|----|----|----|----|----|----|---------------------------------|----|-----------------------------------|--------------------------|----------------------------|---------------------------|----------------------------|----------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Port-n Change Event Enable[15:8] | | | | | | | | Port-n Change Event Enable[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | Interrupt on Async Advance Enable | Host System Error Enable | Frame List Rollover Enable | Port Change Detect Enable | USB Error Interrupt Enable | USB Interrupt Enable |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|-----------------------------------|-----|----|---------------|---|
| | | HCD | HC | | |
| 31 to 16 | Port-n Change Event Enable | R/W | R | H'0000 | The definition for each bit in this field is identical to bit 2 of this register (Port Change Interrupt Enable bit) except these bits are specific to a given port, where bit 16 = Port 1, 17= Port 2, etc. For example, if bit 17 is set to 1 then a port change event was detected on Port 2. When a bit in this field is 1 and the corresponding Port-n Change Detect bit in the USBSTS register is 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Port-n Change Detect bit. bit 31 to 17: Reserved |
| 15 to 6 | — | R | R | All 0 | Reserved |
| 5 | Interrupt on Async Advance Enable | R/W | R | B'0 | When this bit is a one, and the Interrupt on Async Advance bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the Interrupt on Async Advance bit. |
| 4 | Host System Error Enable | R/W | R | B'0 | When this bit is 1, and the Host System Error Status bit in the USBSTS register is 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Host System Error bit. |
| 3 | Frame List Rollover Enable | R/W | R | B'0 | When this bit is 1, and the Frame List Rollover bit in the USBSTS register is 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Frame List Rollover bit. |
| 2 | Port Change Detect Enable | R/W | R | B'0 | When this bit is 1, and the Port Change Detect bit in the USBSTS register is 1, the host controller will issue an interrupt. The interrupt is acknowledged by software clearing the Port Change Detect bit. |

| Bit | Bit Name | R/W | | Initial Value | Description |
|-----|----------------------------|-----|----|---------------|---|
| | | HCD | HC | | |
| 1 | USB Error Interrupt Enable | R/W | R | B'0 | When this bit is 1, and the USBERRINT bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the USBERRINT bit. |
| 0 | USB Interrupt Enable | R/W | R | B'0 | When this bit is 1, and the USBINT bit in the USBSTS register is 1, the host controller will issue an interrupt at the next interrupt threshold. The interrupt is acknowledged by software clearing the USBINT bit. |

60.2.3.4 FRINDEX Register (offset: H'12C)

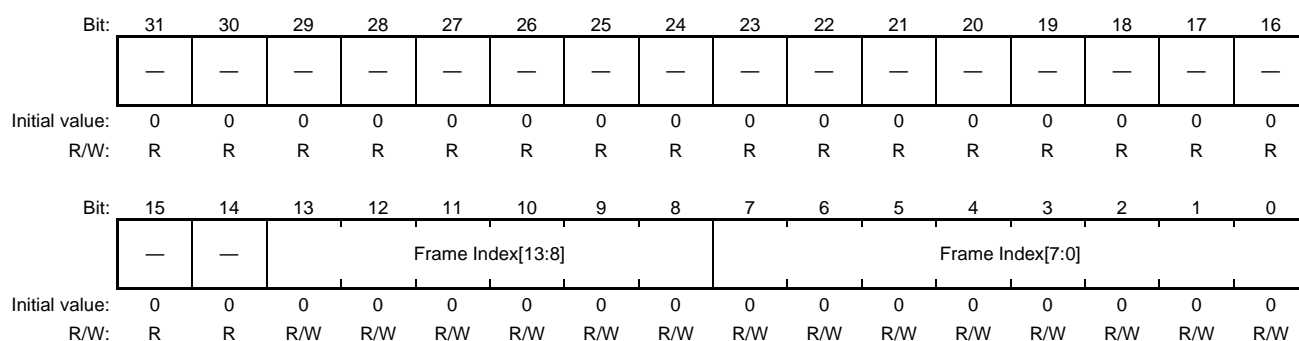
Register symbol: FRINDEX

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register is used by the host controller to index into the periodic frame list. The register updates every 125 microseconds (once each micro-frame). Bits [N:3] are used to select a particular entry in the Periodic Frame List during periodic schedule execution. The number of bits used for the index depends on the size of the frame list as set by system software in the Frame List Size field in the USBCMD register

This register can't be written unless the host controller is in the Halted state as indicated by HCHalted bit of USBSTS register. It is not allowed to write to this register while the Run/Stop bit is 1 either.

The host controller behaves slightly different when using a 32-frame Frame List Size, specifically it continues to count FRINDEX up to a full 1024 frames, transmit SOF values from 0 to 2047, and generate Frame List Rollover events (when enabled) every 1024 frames - exactly as if a 1024 frame list size was employed. The host controller will only reference 32 elements on the periodic schedule, however. This is accomplished by formulating the Periodic Frame List Element Address using {FRINDEX[N:3] modulo 32} rather than FRINDEX[N:3]. The same mapping should be done by system software whenever it needs to correlate the current FRINDEX value to a specific periodic schedule element. Note this new behavior only applies to a 32-frame list.



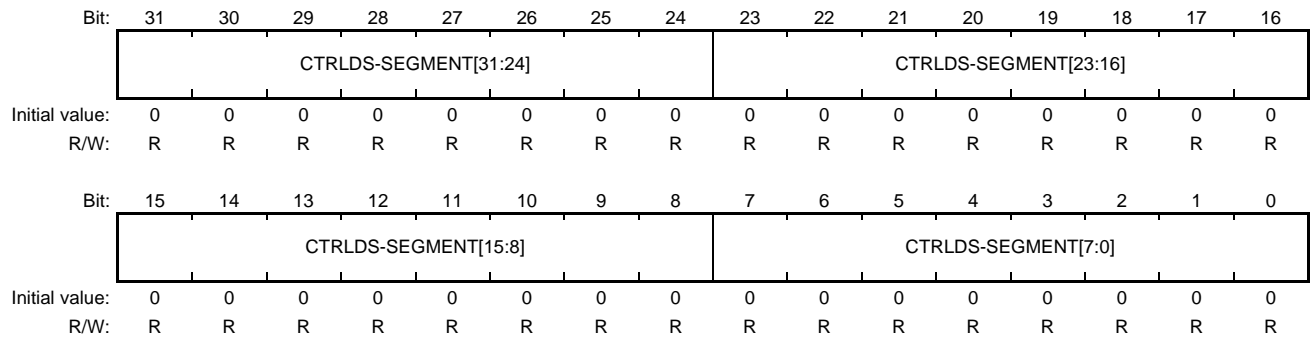
| Bit | Bit Name | R/W | | Initial Value | Description | | | | | | | | | | | | | | | |
|-----------------|-------------------|-----|-----|---------------|--|-----------------|------------------|---|------|------|----|------|-----|----|------|-----|----|------|----|----|
| | | HCD | HC | | | | | | | | | | | | | | | | | |
| 31 to 14 | — | R | R | All 0 | Reserved | | | | | | | | | | | | | | | |
| 13 to 0 | Frame Index[13:0] | R/W | R/W | H'0000 | <p>The value in this register increments at the end of each time frame (e.g. micro-frame). Bits [N:3] are used for the Frame List current index. This means that each location of the frame list is accessed 8 times (frames or micro-frames) before moving to the next index.</p> <p>The following illustrates values of N based on the value of the Frame List Size field in the USBCMD register.</p> <table border="1"> <thead> <tr> <th>Frame List Size</th> <th>Number of Frames</th> <th>N</th> </tr> </thead> <tbody> <tr> <td>B'00</td> <td>1024</td> <td>12</td> </tr> <tr> <td>B'01</td> <td>512</td> <td>11</td> </tr> <tr> <td>B'10</td> <td>256</td> <td>10</td> </tr> <tr> <td>B'11</td> <td>32</td> <td>12</td> </tr> </tbody> </table> | Frame List Size | Number of Frames | N | B'00 | 1024 | 12 | B'01 | 512 | 11 | B'10 | 256 | 10 | B'11 | 32 | 12 |
| Frame List Size | Number of Frames | N | | | | | | | | | | | | | | | | | | |
| B'00 | 1024 | 12 | | | | | | | | | | | | | | | | | | |
| B'01 | 512 | 11 | | | | | | | | | | | | | | | | | | |
| B'10 | 256 | 10 | | | | | | | | | | | | | | | | | | |
| B'11 | 32 | 12 | | | | | | | | | | | | | | | | | | |

60.2.3.5 CTRLDSSEGMENT Register (offset: H'130)

Register symbol: CTRLDSSEGMENT

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Since this host controller does not have the 64-bit Addressing Capability and the 64-bit Addressing Capability bit in HCCPARAMS is 0, CTRLDSSEGMENT register is not used and fixed to H'0000_0000.



| Bit | Bit Name | R/W | | Initial Value | Description |
|---------|----------------|-----|----|---------------|--|
| | | HCD | HC | | |
| 31 to 0 | CTRLDS-SEGMENT | R | R | H'0000_0000 | This host controller does not support 64-bit addressing. |

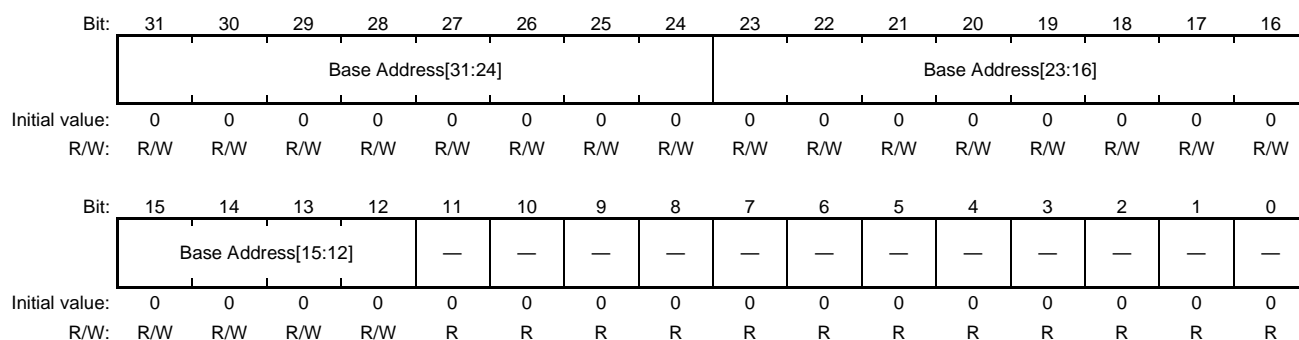
60.2.3.6 PERIODICLISTBASE Register (offset: H'134)

Register symbol: PERIODICLISTBASE

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register contains the beginning address of the Periodic Frame List in the system memory.

Software loads this register prior to starting the schedule execution by the host controller. The memory structure referenced by this physical memory pointer is assumed to be 4-Kbyte aligned. The contents of this register are combined with the Frame Index Register (FRINDEX) to enable the Host Controller to step through the Periodic Frame List in sequence.



| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|--------------|-----|----|---------------|---|
| | | HCD | HC | | |
| 31 to 12 | Base Address | R/W | R | H'0_0000 | This field defines the beginning address of the Periodic Frame List in the system memory. The pointer contained in this field should be 4-Kbyte aligned. |
| 11 to 0 | — | R | R | All 0 | Reserved |

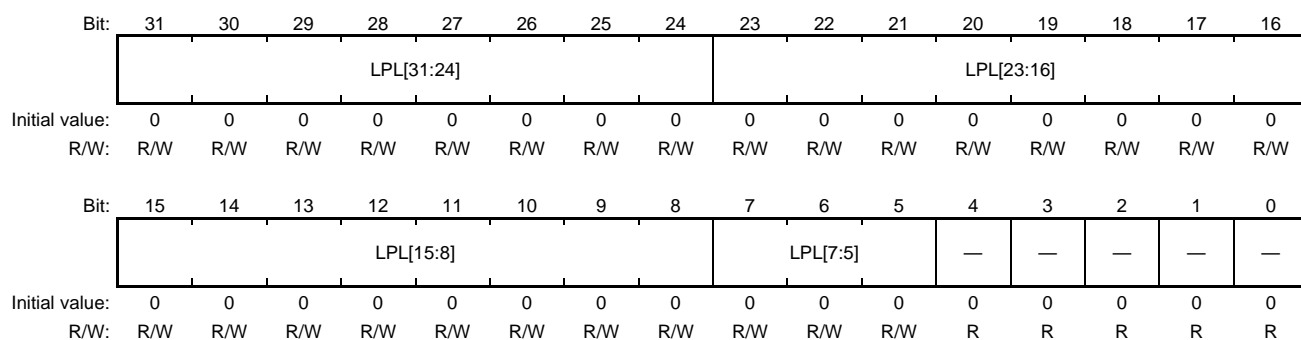
60.2.3.7 ASYNCLISTADDR Register (offset: H'138)

Register symbol: ASYNCLISTADDR

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register contains the address of the next asynchronous queue head to be executed.

Bits [4:0] of this register cannot be modified by software and will always return H'00 when read. The memory structure referenced by this physical memory pointer is assumed to be 32-byte (cache line) aligned.



| Bit | Bit Name | R/W | | Initial Value | Description |
|---------|----------|-----|-----|---------------|---|
| | | HCD | HC | | |
| 31 to 5 | LPL | R/W | R/W | H'000_0000 | Link Pointer Low (LPL). This field defines the address of the next asynchronous queue head to be executed. The pointer contained in this field should be 32-byte aligned. |
| 4 to 0 | — | R | R | All 0 | Reserved |

60.2.3.8 CONFIGFLAG Register (offset: H'160)

Register symbol: CONFIGFLAG

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

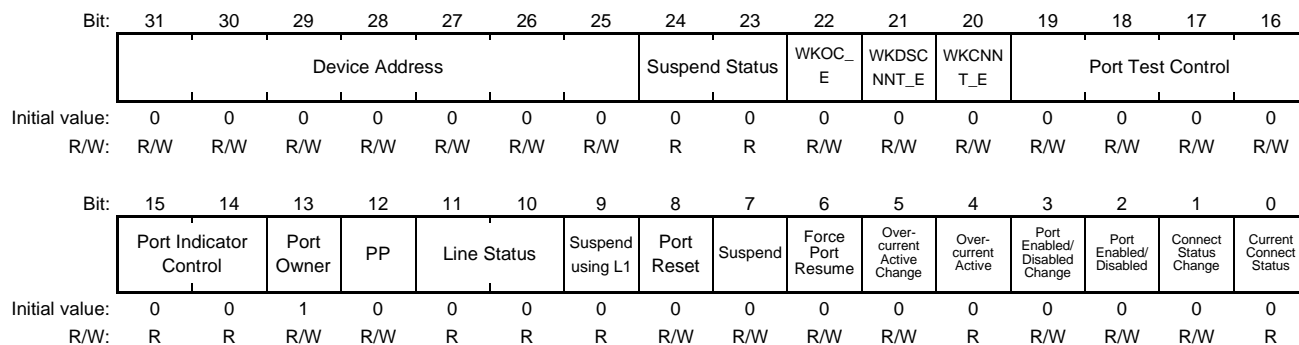
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | CF |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | R/W | | Initial Value | Description |
|---------|----------|-----|----|---------------|---|
| | | HCD | HC | | |
| 31 to 1 | — | R | R | All 0 | Reserved This field defines the address of the next asynchronous queue head to be executed. The pointer contained in this field should be 32-byte aligned. |
| 0 | CF | R/W | R | B'0 | Configuration Flag (CF). Software sets this bit as the last action in its process of configuring the host controller. This bit controls the default port-routing control logic. Bit values and side-effects are listed below. 0: Port routing control logic default-routes each port to OHCI host controller. 1: Port routing control logic default-routes all ports to this (EHCI) host controller. |

60.2.3.9 PORTSC[1] Register (offset: H'164)

Register symbol: PORTSC1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | R/W | | Initial Value | Description |
|----------|----------------|-----|-----|---------------|---|
| | | HCD | HC | | |
| 31 to 25 | Device Address | R/W | R/W | H'00 | The 7-bit USB device address for the device attached to and immediately downstream of the associated root port. A value of zero indicates no device is present or support for this feature is not present. |
| 24, 23 | Suspend Status | R | R/W | B'00 | These two bits are used by software to determine whether the most recent L1 suspend request was successful, specifically B'00: Success: State transition was successful (ACK) B'01: Not Yet: Device was unable to enter the L1 state at this time (NYET). B'10: Not Supported: Device does not support the L1 state (STALL) B'11: Timeout/Error: Device failed to respond or an error occurred This field is updated by hardware immediately following the completion of an L1 transition request (via an LPM token). To avoid any race conditions with hardware, software should only consume the contents of this field when Suspend = 0 (port no longer in L1). |
| 22 | WKOC_E | R/W | R/W | B'0 | Wake on Over-current Enable. Writing 1 to this bit enables the port to detect the over-current conditions as wake-up events. Writing 0 to this bit has no effect. This bit is zero if PP bit is zero. |
| 21 | WKDSCNNT_E | R/W | R/W | B'0 | Wake on Disconnect Enable. Writing 1 to this bit enables the port to detect the device disconnects as wake-up events. Writing 0 to this bit has no effect. This bit is zero if PP bit is zero. |

| Bit | Bit Name | R/W | | Initial Value | Description | | | | | | | | | |
|----------|------------------------|--|-----|---------------|--|-----|----|-----------|---|---|---|---|-----|--|
| | | HCD | HC | | | | | | | | | | | |
| 20 | WKCNTT_E | R/W | R/W | B'0 | <p>Wake on Connect Enable.</p> <p>Writing 1 to this bit enables the port to detect the device connects as wake-up events.</p> <p>Writing 0 to this bit has no effect.</p> <p>This field is zero if PP bit is zero.</p> | | | | | | | | | |
| 19 to 16 | Port Test Control | R/W | R | B'0000 | <p>Port Test Control.</p> <p>A non-zero value indicates that it is operating in test mode and the specific test mode is indicated by the specific value.</p> <p>When this field is zero, the port is NOT operating in a test mode.</p> <p>Value test mode</p> <p>B'0000: Test mode not enabled</p> <p>B'0001: Test J_STATE</p> <p>B'0010: Test K_STATE</p> <p>B'0011: Test SE0_NAK</p> <p>B'0100: Test Packet</p> <p>B'0101: Test FORCE_ENABLE</p> <p>Other: Reserved</p> | | | | | | | | | |
| 15, 14 | Port Indicator Control | R | R | B'00 | <p>This field always shows B'00 since this host controller does not support Port Indicator Control.</p> | | | | | | | | | |
| 13 | Port Owner | R/W | R/W | B'1 | <p>This bit unconditionally goes to 0 when the Configured bit in the CONFIGFLAG register makes a transition from 0 to 1.</p> <p>This bit unconditionally goes to 1 whenever the Configured bit is zero.</p> <p>Software uses this field to release ownership of the port to a selected host controller (in the event that the attached device is not a high-speed device).</p> <p>Software writes 1 to this bit when the attached device is not a high-speed device.</p> <p>0: EHCI controller owns the port.</p> <p>1: OHCI controller owns the port.</p> | | | | | | | | | |
| 12 | PP | R/W | R/W | B'0 | <p>Port Power (PP).</p> <p>The function of this bit depends on the value of PPC bit in HCSPARAMS register. The behavior is as follows:</p> <table border="1"> <thead> <tr> <th>PPC</th> <th>PP</th> <th>Operation</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1</td> <td>RO – The host controller does not have port power control switches. Each port is hard-wired to power.</td> </tr> <tr> <td>1</td> <td>1/0</td> <td>R/W – The host controller has port power control switches. This bit represents the current setting of the switch (0 = off, 1 = on). When power is not available on a port (i.e. PP equals to 0), the port is non-functional and will not report attaches, detaches, etc.</td> </tr> </tbody> </table> <p>When an over-current condition is detected on a powered port and PPC bit is 1, the PP bit in each affected port may be transitioned by the host controller from 1 to 0 (which means removing power from the port).</p> | PPC | PP | Operation | 0 | 1 | RO – The host controller does not have port power control switches. Each port is hard-wired to power. | 1 | 1/0 | R/W – The host controller has port power control switches. This bit represents the current setting of the switch (0 = off, 1 = on). When power is not available on a port (i.e. PP equals to 0), the port is non-functional and will not report attaches, detaches, etc. |
| PPC | PP | Operation | | | | | | | | | | | | |
| 0 | 1 | RO – The host controller does not have port power control switches. Each port is hard-wired to power. | | | | | | | | | | | | |
| 1 | 1/0 | R/W – The host controller has port power control switches. This bit represents the current setting of the switch (0 = off, 1 = on). When power is not available on a port (i.e. PP equals to 0), the port is non-functional and will not report attaches, detaches, etc. | | | | | | | | | | | | |

| Bit | Bit Name | R/W | | Initial Value | Description |
|--------|------------------|-----|-----|---------------|--|
| | | HCD | HC | | |
| 11, 10 | Line Status | R | R/W | B'00 | <p>This field reflects the current logical levels of the D+ (bit 11) and D- (bit 10) signal lines. This field is used for detection of low-speed USB devices prior to the port reset and enable sequence. This field is valid only when Port Enable/Disable bit is 0 and Current Connect Status bit is set to 1.</p> <p>{bit11, bit10} USB state description</p> <p>B'00 SE0 Not LS-device, perform EHCI reset.</p> <p>B'10 J-state Not LS-device, perform EHCI reset.</p> <p>B'01 K-state LS-device, change ownership from EHCI to OHCI.</p> <p>B'11 Undefined Not LS-device, perform EHCI reset.</p> <p>The value in this field is undefined if PP bit is 0.</p> |
| 9 | Suspend using L1 | R | R | B'0 | 0: Suspend using L2 |
| 8 | Port Reset | R/W | R/W | B'0 | <p>This bit shows the reset status of the port.</p> <p>When software writes 1 to this bit (from a zero), the bus reset is started. Software is required to write 0 to this bit to terminate the bus reset sequence. Software should keep this bit at 1 long enough to ensure the reset sequence completes as specified in the USB Specification.</p> <p>Do not write 1 to this bit when HCHalted bit in USBSTS register is 0.</p> <p>This bit is 0 if PP bit is 0.</p> <p>0: The port is not in reset.</p> <p>1: The port is in reset.</p> |

| Bit | Bit Name | R/W | | Initial Value | Description | | | | | | | | |
|-------------------------|-------------|-----|-----|---------------|---|-------------------------|-------------|------|---------|------|--------|------|---------|
| | | HCD | HC | | | | | | | | | | |
| 7 | Suspend | R/W | R/W | B'0 | <p><i>Port Enabled</i> Bit and <i>Suspend</i> bit of this register define the port states as follows:</p> <table border="0"> <tr> <td>{Port Enabled, Suspend}</td> <td>Port Status</td> </tr> <tr> <td>B'0x</td> <td>Disable</td> </tr> <tr> <td>B'10</td> <td>Enable</td> </tr> <tr> <td>B'11</td> <td>Suspend</td> </tr> </table> <p>Software writes a one to this bit to transition a port to either the L1 or L2 suspend state. Which suspend state the host controller attempts depends on the value of the Suspend Using L1 field.</p> <p>When in suspend state, downstream propagation of data is blocked on this port, except for port reset. If this bit is set to a one when a transaction is in progress then the blocking will not occur until the end of the current transaction. In the suspend state, the port is sensitive to resume detection. Note that the status of this bit does not change until the port is fully suspended and there may be a delay in suspending a port if a transaction is currently in progress on the USB. Addition status for L1-based transitions is provided to software via the Suspend Status field.</p> <p>Software writes 1 to this bit in order to set it. Writing 0 to this bit has no effect.</p> <p>Software can writes to this bit only when PP bit is 1 and Port Owner bit is 0 and Current Connect Status bit is 1.</p> <p>The host controller unconditionally sets this bit to 0 when: Software sets Force Port Resume bit to 0 (from 1). Software sets Port Reset bit to 1 (from 0). Whenever Port Power is 0 This field is 0 if PP bit is 0.</p> | {Port Enabled, Suspend} | Port Status | B'0x | Disable | B'10 | Enable | B'11 | Suspend |
| {Port Enabled, Suspend} | Port Status | | | | | | | | | | | | |
| B'0x | Disable | | | | | | | | | | | | |
| B'10 | Enable | | | | | | | | | | | | |
| B'11 | Suspend | | | | | | | | | | | | |

| Bit | Bit Name | R/W | | Initial Value | Description |
|-----|----------------------------|-----|-----|---------------|---|
| | | HCD | HC | | |
| 6 | Force Port Resume | R/W | R/W | B'0 | <p>This bit indicates that the resume on the port is detected.</p> <p>0: No resume signaling is detected or driven on the port.</p> <p>1: The resume signaling is detected or driven on the port.</p> <p>The host controller set this bit to 1 if a J-to-K transition is detected while the port is in the Suspend state. When this bit transitions to 1 because a J-to-K transition is detected, the Port Change Detect and/or Port-n Changes Detect bits in the USBSTS register are also set to 1.</p> <p>If software sets this bit to 1, the host controller must not set the Port Change Detect and/or Port-n Change Detect bits.</p> <p>This functionality defined for manipulating this bit depends on the value of the Suspend and Suspend Using L1 bits. For example, if the port is not suspended (Suspend and Enabled bits are a one) and software transitions this bit to a one, then the effects on the bus are undefined.</p> <p>The resume signaling (Full-speed 'K') is driven on the port as long as this bit remains a one. For legacy (L2) transitions, Software should set this bit to 0 when the appropriate amount of time has elapsed.</p> <p>Writing 0 (from 1) causes the port to return to high speed mode (forcing the bus below the port into a high-speed idle). This bit will remain 1 until the port has switched to the high-speed idle. The host controller must complete this transition within 2 milliseconds of software setting this bit to 0. Software does not need to time resume signaling for L1 transactions as host controller hardware will automatically enforce the necessary timing and clear this bit when the port has fully resumed. Software can influence the amount of time hardware will drive resume signaling during L1 exit via the Host-Initiated Resume Duration field within the USBCMD register.</p> <p>This bit is 0 if PP bit is 0</p> |
| 5 | Over-current Change | R/W | R/W | B'0 | <p>This bit indicates whether Over-current Active bit has been changed or not.</p> <p>Writing 1 to this bit clears it. Writing 0 to this bit has no effect.</p> <p>0: Over-current Active bit has not changed.</p> <p>1: Over-current Active bit has changed.</p> |
| 4 | Over-current Active | R | R/W | B'0 | <p>This bit shows the over-current status of the port.</p> <p>When the host controller detects the over-current condition, it disables the port and then sets this bit to 1.</p> <p>The host controller automatically clears this bit when the over-current condition has been removed.</p> <p>0: No over-current condition exists on the port.</p> <p>1: The over-current condition exists on the port.</p> |
| 3 | Port Enable/Disable Change | R/W | R/W | B'0 | <p>This bit indicates whether Port Enabled/Disabled bit has been changed or not.</p> <p>This bit is set 1 only when a port is disabled due to the appropriate conditions existing at EOF2 point.</p> <p>Writing 1 to this bit clears it. Writing 0 to this bit has no effect.</p> <p>This bit is 0 if PP bit is 0.</p> |

| Bit | Bit Name | R/W | | Initial Value | Description |
|-----|---------------------------|-----|-----|---------------|---|
| | | HCD | HC | | |
| 2 | Port Enabled/ Disabled | R/W | R/W | B'0 | <p>Ports can only be enabled by the host controller as a part of the reset and enable. Software cannot enable a port by writing 1 to this field. The host controller will only set this bit to 1 the reset sequence determines that the attached device is a high-speed device.</p> <p>Ports can be disabled by either a fault condition (disconnect event or other fault condition) or by software. When the port is disabled (0) downstream propagation of data is blocked on this port, except for reset.</p> <p>When Port Test Control is B'0101, the port is always enabled and this bit is set to 1 independent of the port status.</p> <p>This bit is 0 if PP bit is 0.</p> <p>0: The port is disabled 1: The port is enabled.</p> |
| 1 | Connect Status Change | R/W | R/W | B'0 | <p>This bit indicates a change has occurred in the port's Current Connect Status. The host controller sets this bit for all changes to the port device connect status, even if software has not cleared an existing connect status change.</p> <p>Writing 1 to this bit clears it. Writing 0 to this bit has no effect.</p> <p>This field is 0 if PP bit is 0.</p> |
| 0 | Current Connect Status | R | R/W | B'0 | <p>This bit reflects the current state of the connection of the port. The host controller sets this bit to 1 when it detects the connection of device to the port.</p> <p>It also sets 1 to this bit with Port Test Control = B'0101, even when device is not connected.</p> <p>If it detects the device is disconnected, this bit is reset to 0. This bit is 0 if PP bit is 0 or if Port Owner bit is 0.</p> <p>0: No device is present. 1: Device is present.</p> |

60.2.4 AHB Bridge Registers

60.2.4.1 INT_ENABLE Register (offset: H'200)

Register symbol: INT_ENABLE

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register enables or disables interrupt generation for each interrupt source bit in the INT_STATUS register. When a bit in this register is set to disabled, the interrupt signal is not asserted even when the interrupt condition for the corresponding source is satisfied and the corresponding source bit in the INT_STATUS register is set to 1.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|---------|--------|--------|--------|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | WAKEON_ | UCOM_I | USBH_I | USBH_I | AHB_ |
| | | | | | | | | | | | | INTEN | NTEN | NTBEN | NTAEN | INTEN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|--|
| 31 to 5 | — | All 0 | R | Reserved The write value should be 0. |
| 4 | WAKEON_INTEN | B'0 | R/W | Enables or disables interrupt generation due to WAKEON_INT (bit 4) in INT_STATUS. 0: Disabled 1: Enabled |
| 3 | UCOM_INTEN | B'0 | R/W | Enables or disables interrupt generation due to UCOM_INT (bit 3) in INT_STATUS. 0: Disabled 1: Enabled |
| 2 | USBH_INTBEN | B'0 | R/W | Enables or disables interrupt generation due to USBH_INTB (bit 2) in INT_STATUS. 0: Disabled 1: Enabled |
| 1 | USBH_INTAEN | B'0 | R/W | Enables or disables interrupt generation due to USBH_INTA (bit 1) in INT_STATUS. 0: Disabled 1: Enabled |
| 0 | AHB_INTEN | B'0 | R/W | Enables or disables interrupt generation due to AHB_INT (bit 0) in INT_STATUS. 0: Disabled 1: Enabled |

60.2.4.2 INT_STATUS Register (offset: H'204)

Register symbol: INT_STATUS

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register indicates the state of the interrupt source in the AHB bridge and the state of each signal from the OHCI, EHCI, and UCOM modules.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----------------|--------------|---------------|---------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | WAKEO N_INT | UCOM _INT | USBH_ INTB | USBH_ INTA | AHB_I NT |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 to 5 | — | All 0 | R | Reserved The write value should be 0. |
| 4 | WAKEON_INT | B'0 | R/W | This bit indicates the state of the WAKEON interrupt from the host core. Writing 1 clears this bit. 0: No WAKEON interrupt has occurred. 1: A WAKEON interrupt has occurred. |
| 3 | UCOM_INT | B'0 | R | This bit indicates the state of the interrupt from the UCOM registers. Use the UCOM2 registers to clear the interrupt. 0: No external register interrupt has occurred. 1: An external register interrupt has occurred. |
| 2 | USBH_INTB | B'0 | R | This bit indicates the state of the EHCI interrupt. Use the HcInterruptStatus register (an OHCI operational register) to clear the interrupt. 0: No INTB interrupt has occurred. 1: An INTB interrupt has occurred. |
| 1 | USBH_INTA | B'0 | R | This bit indicates the state of the OHCI interrupt. Use the USBSTS register (an EHCI operational register) to clear the interrupt. 0: No INTA interrupt has occurred. 1: An INTA interrupt has occurred. |
| 0 | AHB_INT | B'0 | R/W | This bit indicates occurrence of an AHB bus error. Writing 1 clears this bit. 0: No AHB bus error has occurred. 1: An AHB bus error has occurred. |

60.2.4.3 AHB_BUS_CTR Register (offset: H'208)

Register symbol: AHB_BUS_CTR

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register specifies the AHB master and slave functions.

| | | | | | | | | | | | | | | | | |
|----------------|-----------|-----|-----|-----|----|----|-----------|-----|----|---------------|-----|-----|----|---------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PROT_TYPE | | | — | — | — | PROT_MODE | — | — | ALIGN_ADDRESS | | — | — | MAX_BURST_LEN | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R | R | R/W | R | R | R/W | R/W | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved The write value should be 0. |
| 15 to 12 | PROT_TYPE | B'0000 | R/W | These bits specify the MHPROT[3:0] value for transfer from the AHB master interface. Bit [15] 0: Indicates a non-cacheable transfer. 1: Indicates a cacheable transfer. Bit [14] 0: Indicates a non-bufferable transfer. 1: Indicates a bufferable transfer. Bit [13] 0: Indicates user access. 1: Indicates privileged access. Bit [12] 0: Indicates an opcode. 1: Indicates data. For details of each bit meaning, refer to the AMBA AHB specifications |
| 11 to 9 | — | All 0 | R | Reserved The write value should be 0. |
| 8 | PROT_MODE | B'0 | R/W | This bit selects the MHPROT[3:0] mode for transfer from the AHB master interface. 0: The PROT_TYPE value is output as MHPROT[3:0]. 1: During DMA transfer, H'0000 is output as MHPROT[3:0] in the last burst and the PROT_TYPE value is output in other burst transfers. |
| 7, 6 | — | All 0 | R | Reserved The write value should be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|-------------------|---------------|-----|---|
| 5, 4 | ALIGN_ADD RESS | B'00 | R/W | <p>These bits specify the address alignment for transfer from the AHB master interface.</p> <p>B'00: Burst transfer is issued so as not to extend beyond a 1-Kbyte boundary.</p> <p>B'01: Burst transfer is issued so as not to extend beyond a 64-byte boundary.</p> <p>B'10: Burst transfer is issued so as not to extend beyond a 32-byte boundary. (The maximum burst length is limited to INCR8 because transfer goes beyond a 32-byte boundary if INCR16 is specified.)</p> <p>B'11: Burst transfer is issued so as not to extend beyond a 16-byte boundary. (The maximum burst length is limited to INCR4 because transfer goes beyond a 16-byte boundary if INCR8 or a longer length is specified.)</p> |
| 3, 2 | — | All 0 | R | <p>Reserved</p> <p>The write value should be 0.</p> |
| 1, 0 | MAX_BURST _LEN | B'00 | R/W | <p>These bits specify the maximum burst length for transfer from the AHB master interface.</p> <p>B'00: INCR16</p> <p>B'01: INCR8</p> <p>B'10: INCR4</p> <p>B'11: SINGLE</p> |

60.2.4.4 USBCTR Register (offset: H'20C)

Register symbol: USBCTR

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register makes settings for the host core.

| | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|-------------|----------------------------|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | DIRPD | PLL_R ST | <i>Renesas Private</i> | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------------------|---------------|-----|--|
| 31 to 3 | — | All 0 | R | Reserved The write value should be 0. |
| 2 | DIRPD | B'0 | R/W | This bit controls the direct power-down (DIRPD) pin of the UTMI+ PHY developed by Renesas. Setting this bit to 1 places the PHY in the power-down state. Writing 0 to this bit while it is 1 releases the PHY from the direct power-down state. For setting and cancellation of direct power-down state, refer to the manual of the Renesas UTMI+ core. 0: Normal operation 1: Renesas UTMI+ PHY is placed in the power-down state. |
| 1 | PLL_RST | B'0 | R/W | This bit controls the PLLRESET pin of the UTMI+ PHY developed by Renesas. 0: PLL reset is canceled. 1: PLL reset is issued. |
| 0 | <i>Renesas Private</i> | B'0 | W | Do not write 1 to this bit. |

60.2.5 Core Defined Registers

60.2.5.1 REVID Register (offset: H'300)

Register symbol: REVID

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------------|----|----|----|----|----|----|----|---------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Core ID | | | | | | | | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Major Version | | | | | | | | Minor Version | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 24 | Core ID | H'00 | R | These bits indicate the ID of the host core. |
| 23 to 16 | — | All 0 | R | Reserved The write value should be 0. |
| 15 to 8 | Major Version | H'02 | R | These bits indicate the major version number of the host core. |
| 7 to 0 | Minor Version | H'01 | R | These bits indicate the minor version number of the host core. |

60.2.5.2 Register Enable/Clock Gating Control Register (offset: H'304)

Register symbol: REGEN_CG_CTRL

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Some registers are read-only in the initial state.

When such registers should be written to, set the corresponding bits in this register.

In addition, the clock can be gated for the unused sections in the host and peripheral function circuits.

Set the corresponding bits as necessary.

| | | | | | | | | | | | | | | | | |
|----------------|----------------|----|--------------|--------------|----|----|----|-----------------|----|----|----|----|----|----|----|-----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | NONUSE_CLK_MSK | — | HOST_CLK_MSK | PERI_CLK_MSK | — | — | — | Renesas Private | — | — | — | — | — | — | — | Renesas Private |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R/W | R/W | R | R | R | R/W | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | Renesas Private |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 | NONUSE_CLK_MSK | B'0 | R/W | This bit enables masking of the clock for the unused one of the host and peripheral function circuits. 0: Not masked. 1: Masked. |
| 30 | — | B'0 | R | Reserved The write value should be 0. |
| 29 | HOST_CLK_MSK | B'0 | R/W | This bit forcibly masks the clock to some circuits in the host core. 0: Not masked. 1: Masked. |
| 28 | PERI_CLK_MSK | B'0 | R/W | This bit forcibly masks the clock to some circuits in the peripheral controller. 0: Not masked. 1: Masked. |
| 27 to 25 | — | All 0 | R | Reserved The write value should be 0. |
| 24 | Renesas Private | B'0 | R/W | Do not write 1 to this bit. |
| 23 to 17 | — | All 0 | R | Reserved The write value should be 0. |
| 16 | Renesas Private | B'0 | R/W | Do not write 1 to this bit. |
| 15 to 1 | — | All 0 | R | Reserved The write value should be 0. |
| 0 | Renesas Private | B'0 | R/W | Do not write 1 to this bit. |

60.2.5.3 Suspend Control Register (offset: H'308)

Register symbol: SPD_CTRL

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register controls the SUSPENDM pin in the UTMI+ interface.

Through this register setting, SUSPENDM can be asserted when the Suspend state is entered, and UTMI+ interface SUSPENDM can be forcibly asserted.

The equivalent control is also applied to the SLEEPM pin.

| | | | | | | | | | | | | | | | | |
|----------------|-----------------|------------------------|----|----|----|----|------------------------|------------------------|---------------|------------------------|----|----|----|----|------------------------|--------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SUSPENDM_ENABLE | <i>Renesas Private</i> | — | — | — | — | — | — | WKCNTT_ENABLE | <i>Renesas Private</i> | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | <i>Renesas Private</i> | <i>Renesas Private</i> | — | — | — | — | — | — | <i>Renesas Private</i> | GLOBAL_SUSPENDM_P1 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------------|---------------|-----|--|
| 31 | SUSPENDM_ENABLE | B'0 | R/W | This bit provides a function to assert the UTMI+ interface SUSPENDM signal when the Suspend bit in the related OHCI or EHCI operational register is asserted. Setting this bit to 1 asserts the UTMI+ interface SUSPENDM signal in the target port after the Suspend-related bit in the OHCI or EHCI operational register shown below is set. This function is intended to reduce the power consumption in the UTMI+ PHY when this module enters the Suspend state. [Target OHCI and EHCI operational registers of this function] OHCI: Bit [2] (PSS bit) in the HcRhPortStatus register EHCI: Bit [7] (Suspend bit) in the PORTSC register |
| 30 | <i>Renesas Private</i> | B'0 | R/W | Do not write 1 to this bit. |
| 29 to 24 | — | All 0 | R | Reserved The write value should be 0. |
| 23 | WKCNTT_ENABLE | B'0 | R/W | Setting this bit to 1 deasserts SUSPENDM and SLEEPM when a change in the connection state is detected. This setting is valid when SUSPENDM_ENABLE or SLEEP_ENABLE is 1. |
| 22 | <i>Renesas Private</i> | B'0 | R/W | Do not write 1 to this bit. |
| 21 to 10 | — | All 0 | R | Reserved The write value should be 0. |
| 9 | <i>Renesas Private</i> | B'0 | R/W | Do not write 1 to this bit. |
| 8 | <i>Renesas Private</i> | B'0 | R/W | Do not write 1 to this bit. |

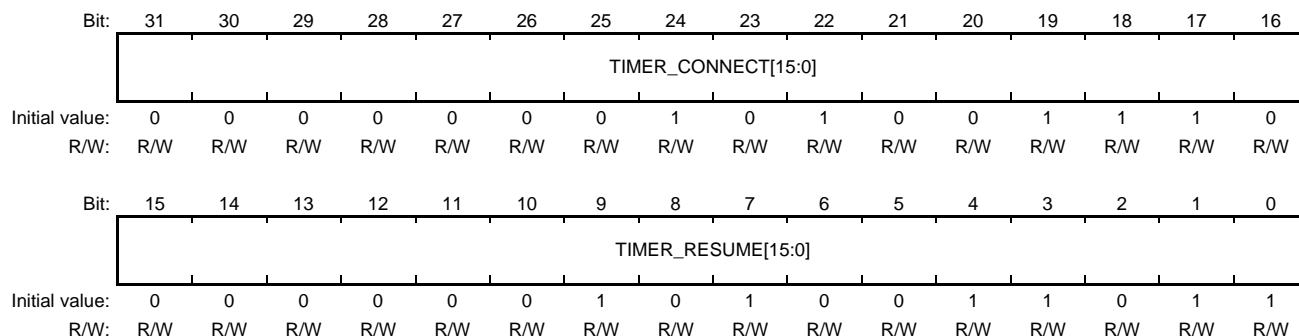
| Bit | Bit Name | Initial Value | R/W | Description |
|--------|--------------------|---------------|-----|--|
| 7 to 1 | — | All 0 | R | Reserved The write value should be 0. |
| 0 | GLOBAL_SUSPENDM_P1 | B'0 | R/W | <p>This bit provides a function to forcibly assert UTMI_P1SUSPENDM to 0.</p> <p>Setting this bit to 1 asserts the UTMI+ interface SUSPENDM pin (UTMI_P1SUSPENDM) for port 1.</p> <p>This function is intended to reduce the power consumption of the UTMI+ PHY by controlling this bit appropriately.</p> <p>[Note] Writing 1 to this bit is prohibited when the SUSPENDM_ENABLE bit is 1.</p> |

60.2.5.4 Suspend/Resume Timer Setting Register (offset: H'30C)

Register symbol: SPD_RSM_TIMSET

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register specifies the Connect/Disconnect detection time and Resume detection time when the clock from the UTMI+ PHY is stopped in the Suspend state.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|---|
| 31 to 16 | TIMER_CONNECT[15:0] | H'014E | R/W | <p>These bits specify the timer value used to detect Device Connect or Disconnect while the UTMI+ interface SUSPENDM is asserted. Assuming that the clock from the UTMI+ PHY stops while SUSPENDM is asserted, whether Connect or Disconnect has occurred is checked using the AHB clock (HCLK). *</p> <p>Set this timer value so that the detection time is 2.5 μs or longer according to the AHB clock frequency used. A value of 1 specifies one cycle (μs).</p> <p>Reference setting values: AHB clock is 133 MHz: H'014E or a greater value AHB clock is 200 MHz: H'1F4 or a greater value</p> |
| 15 to 0 | TIMER_RESUME[15:0] | H'029B | R/W | <p>These bits specify the timer value used to detect the RemoteWakeup signal from the device while the UTMI+ interface SUSPENDM is asserted. Assuming that the clock from the UTMI+ PHY stops while SUSPENDM is asserted, whether the RemoteWakeup signal has been received is checked using the AHB clock (HCLK).</p> <p>Set this timer value so that the detection time is 5 μs or longer according to the AHB clock frequency used. A value of 1 specifies one cycle (μs).</p> <p>Reference setting values: AHB clock is 133 MHz: H'029B or a greater value</p> |

Note: Frequency of AHB clock (HCLK) is 133MHz (S3D2 ϕ).

60.2.5.5 Overcurrent Detection Timer Setting Register (offset: H'310)

Register symbol: OC_TIMSET

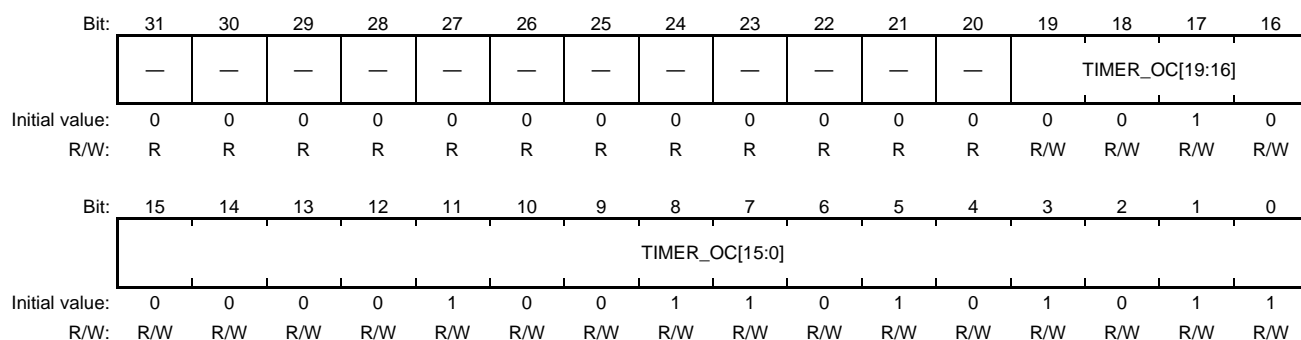
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register specifies the overcurrent detection time.

When the overcurrent detection input signal (EXT_P1OCI) is asserted to 0 continuously for the time specified in this register, occurrence of an overcurrent is detected.

[Note]

This register setting is ignored when the OCISEL bit (bit [5]) in the host COMMON/OHCI extended register (offset: H'360) is 1.



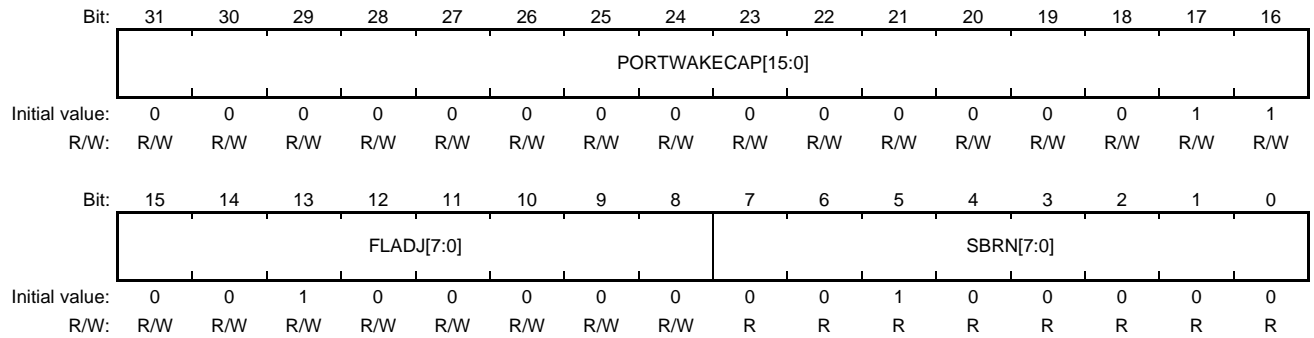
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 to 20 | — | All 0 | R | Reserved The write value should be 0. |
| 19 to 0 | TIMER_OC [19:0] | H'2_09AB | R/W | These bits specify the timer value used to detect an overcurrent. When the overcurrent detection input signal (EXT_P1OCI) is asserted to 0 continuously for the time specified in this field, this module determines that an overcurrent has occurred. Set this timer value so that the detection time is 1 to 2 ms or longer according to the AHB clock frequency used. A value of 1 specifies one cycle (μs). Note that occurrence of an overcurrent is detected when EXT_P1OCI is asserted to 0 continuously over a certain period. Specify the period in these bits according to the AHB clock frequency. Reference setting values: To specify 1 ms when AHB clock (HCLK) is 133 MHz: H'2 09AB or a greater value. |

Note: Frequency of AHB clock (HCLK) is 133MHz (S3D2φ).

60.2.5.6 SBRN, FLADJ, PORTWAKECAP Register (offset: H'314)

Register symbol: SBRN_FLADJ_PW

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------|---------------|-----|---|
| 31 to 16 | PORTWAKE CAP[15:0] | H'0003 | R/W | The intended use of these bits are to establish a policy about which ports are to be used for wake events. This bit setting does not affect the operation of the host core. |
| 15 to 8 | FLADJ[7:0] | H'20 | R/W | These bits adjust the length of one micro-frame in units of 16-HS-bit time. The initial value is H'20 (60000 HS-bit time). |
| 7 to 0 | SBRN[7:0] | H'20 | R | These bits indicate the serial bus release number. This value is fixed to H'20. |

60.2.6 UCOM Registers

60.2.6.1 Common Control Register (offset: H'800)

Register symbol: COMMCTRL

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register selects either the host or peripheral mode for each of the OTG and battery charging functions.

| | | | | | | | | | | | | | | | | |
|----------------|--------------|-------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | OTG_P ERI | BC_PE RI | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 | OTG_PERI | B'1 | R/W | This bit specifies whether to place this module in the host mode or peripheral mode when using this module as an OTG device. 0: Host mode 1: Peripheral mode |
| 30 | BC_PERI | B'0 | R/W | This bit specifies whether to control the battery charging function through host core registers or peripheral core registers. 0: Controlled by the host core. 1: Controlled by the peripheral core. |
| 29 to 0 | — | All 0 | R | Reserved |

60.2.6.2 OTG-BC Interrupt Status Register (offset: H'804)

Register symbol: OBINTSTA

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register indicates the states of the OTG and BC-related interrupts. An OBINT interrupt occurs when any of the bits in this register is asserted and occurrence of the interrupt is enabled in the OBINTEN register.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|--------------------|------------------|------------------|------------------|------------------|----|----|-------------------|-------------------|------------------|-------------------|------------------|------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | VGPICH G_STA3 | VGPICH G_STA2 | VGPICH G_STA1 | VGPICH G_STA0 | — | — | — | — | — | DPMSTAI NT_STA | DPMONC HG_STA | DMMONC HG_STA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R | R | R | R | R | R/WC1 | R/WC1 | R/WC1 |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | SESSVLD CHG_STA | IDDIGC HG_STA | VVLDCG G_STA | APRBCH G_STA | ASNSCH G_STA | — | — | PDDETCH G2_STA | PDDETCH G1_STA | VBSTAI NT_STA | VBSTAC HG_STA | OCINT_ STA | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R | R | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R/WC1 | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------|---------------|-------|---|
| 31 to 28 | — | All 0 | R | Reserved |
| 27 | VGPICHG_ STA3* | B'0 | R/WC1 | This bit is set when the input to the EXT_VGPI[3] pin changes. 0: EXT_VGPI[3] input has not changed. 1: EXT_VGPI[3] input has changed. |
| 26 | VGPICHG_ STA2* | B'0 | R/WC1 | This bit is set when the input to the EXT_VGPI[2] pin changes. 0: EXT_VGPI[2] input has not changed. 1: EXT_VGPI[2] input has changed. |
| 25 | VGPICHG_ STA1* | B'0 | R/WC1 | This bit is set when the input to the EXT_VGPI[1] pin changes. 0: EXT_VGPI[1] input has not changed. 1: EXT_VGPI[1] input has changed. |
| 24 | VGPICHG_ STA0* | B'0 | R/WC1 | This bit is set when the input to the EXT_VGPI[0] pin changes. 0: EXT_VGPI[0] input has not changed. 1: EXT_VGPI[0] input has changed. |
| 23 to 19 | — | All 0 | R | Reserved |
| 18 | DPMSTAI T_STA | B'0 | R/WC1 | This bit is set when D+ and D- in port 1 become equal to the value specified in LINECTRL1.DPMLVL[1:0]. However, if this bit was set when LINECTRL1.DMMON and DPMON became equal to LINECTRL1.DMPLVL[1:0] and then this bit was cleared but LINECTRL1.DMMON and DPMON have not changed, this bit is not set even if LINECTRL1.DPMON and DMMON are equal to LINECTRL1.DPMLVL[1:0]. In this case, this bit is set only when LINECTRL1.DPMON or DMMON changes (a change in either of them will suffice in this case) and then LINECTRL1.DPMON and DMMON become equal to LINECTRL1.DPMLVL[1:0] again. 0: D+ and D- have not become equal to the value specified in LINECTRL1.DPMLVL[1:0]. 1: D+ and D- have become equal to the value specified in LINECTRL1.DPMLVL[1:0]. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-------|---|
| 17 | DPMONCH_G_STA | B'0 | R/WC1 | This bit is set when LINECTRL1.DPMON changes — that is, when the input to D+ in port 1 changes. 0: LINECTRL1.DPMON has not changed. 1: LINECTRL1.DPMON has changed. |
| 16 | DMMONCH_G_STA | B'0 | R/WC1 | This bit is set when LINECTRL1.DMMON changes — that is, when the input to D- in port 1 changes. 0: LINECTRL1.DMMON has not changed. 1: LINECTRL1.DMMON has changed. |
| 15 to 13 | — | All 0 | R | Reserved |
| 12 | SESSVLDCHG_STA* | B'0 | R/WC1 | This bit is set when ADPCTRL.OTGSESSVLD changes. 0: ADPCTRL.OTGSESSVLD has not changed. 1: ADPCTRL.OTGSESSVLD has changed. |
| 11 | IDDIGCHG_STA | B'0 | R/WC1 | This bit is set when ADPCTRL.IDDIG changes. 0: ADPCTRL.IDDIG has not changed. 1: ADPCTRL.IDDIG has changed. |
| 10 | VVLDCHG_STA* | B'0 | R/WC1 | This bit is set when ADPCTRL.VBUSVALID changes. 0: ADPCTRL.VBUSVALID has not changed. 1: ADPCTRL.VBUSVALID has changed. |
| 9 | APRBCHG_STA | B'0 | R/WC1 | This bit is set when ADPCTRL.ADPPRB changes. 0: ADPCTRL.ADPPRB has not changed. 1: ADPCTRL.ADPPRB has changed. |
| 8 | ASNSCHG_STA | B'0 | R/WC1 | This bit is set when ADPCTRL.ADPSNS changes. 0: ADPCTRL.ADPSNS has not changed. 1: ADPCTRL.ADPSNS has changed. |
| 7, 6 | — | All 0 | R | Reserved |
| 5 | PDDETCH_G2_STA | B'0 | R/WC1 | This bit is set when BCCTRL2.PDDETSTS changes from 0 to 1. 0: BCCTRL2.PDDETSTS has not changed from 0 to 1. 1: BCCTRL2.PDDETSTS has changed from 0 to 1. |
| 4 | PDDETCH_G1_STA | B'0 | R/WC1 | This bit is set when BCCTRL1.PDDETSTS changes from 0 to 1. 0: BCCTRL1.PDDETSTS has not changed from 0 to 1. 1: BCCTRL1.PDDETSTS has changed from 0 to 1. |
| 3 | VBSTAINCHG_STA* | B'0 | R/WC1 | This bit is set when VBCTRL.VBSTA[3:0] becomes equal to the value specified in VBCTRL.VBLVL[3:0]. However, if this bit was set when VBCTRL.VBSTA[3:0] became equal to VBCTRL.VBLVL[3:0] and then this bit was cleared but VBCTRL.VBSTA[3:0] has not changed, this bit is not set even if VBCTRL.VBSTA[3:0] is equal to VBCTRL.VBLVL[3:0]. In this case, this bit is set only when VBCTRL.VBSTA[3:0] changes and then VBCTRL.VBSTA[3:0] becomes equal to VBCTRL.VBLVL[3:0] again. 0: VBCTRL.VBSTA[3:0] has not become equal to the value specified in VBCTRL.VBLVL[3:0]. 1: VBCTRL.VBSTA[3:0] has become equal to the value specified in VBCTRL.VBLVL[3:0]. |
| 2 | VBSTACHG_STA* | B'0 | R/WC1 | This bit is set when the VBCTRL.VBSTA[3:0] state changes. 0: VBCTRL.VBSTA[3:0] has not changed. 1: VBCTRL.VBSTA[3:0] has changed. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|---------------|---------------|-------|--|
| 1 | OCINT_ST A | B'0 | R/WC1 | This bit is set when the input to the EXT_P10CI pin is asserted. 0: EXT_P10CI has not been asserted (remains at 1). 1: EXT_P10CI has been asserted (has become 0). |
| 0 | — | B'0 | R | Reserved |

Note: * These bits are not supported RZ/G2E

60.2.6.3 OTG-BC Interrupt Enable Register (offset: H'808)

Register symbol: OBINTEN

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

This register enables or disables interrupts due to the status bits in the OBINTSTA register. When any of the bits in OBINTSTA is asserted and occurrence of the interrupt is enabled through the corresponding bit in the OBINTEN register, an OBINT interrupt occurs.

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|----|----|----|----|-----------------|-----------------|-----------------|-----------------|----|----|----|----|----|------------------|-----------------|-----------------|
| | — | — | — | — | VGPICH G_EN3 | VGPICH G_EN2 | VGPICH G_EN1 | VGPICH G_EN0 | — | — | — | — | — | DPMSTA INT_EN | DPMON CHG_EN | DMMON CHG_EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R | R/W | R/W | R/W |

| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------------|----|----|----|-------------------|-----------------|----------------|----------------|----------------|---|---|-----------------|-----------------|-----------------|-----------------|--------------|--------------|
| | — | — | — | SESSVLD CHG_EN | IDDIGC HG_EN | VVLDCH G_EN | APRBCH G_EN | ASNSCH G_EN | — | — | PDETC HG2_EN | PDETC HG1_EN | VBSTAI NT_EN | VBSTAC HG_EN | OCINT_ EN | IDCHG_ EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|--|
| 31 to 28 | — | All 0 | R | Reserved |
| 27 | VGPICHG_E N3* | B'0 | R/W | This bit enables VGPICHG_STA3 interrupts. 0: Interrupts due to VGPICHG_STA3 are disabled. 1: Interrupts due to VGPICHG_STA3 are enabled. |
| 26 | VGPICHG_E N2* | B'0 | R/W | This bit enables VGPICHG_STA2 interrupts. 0: Interrupts due to VGPICHG_STA2 are disabled. 1: Interrupts due to VGPICHG_STA2 are enabled. |
| 25 | VGPICHG_E N1 | B'0 | R/W | This bit enables VGPICHG_STA1 interrupts. 0: Interrupts due to VGPICHG_STA1 are disabled. 1: Interrupts due to VGPICHG_STA1 are enabled. |
| 24 | VGPICHG_E N0* | B'0 | R/W | This bit enables VGPICHG_STA0 interrupts. 0: Interrupts due to VGPICHG_STA0 are disabled. 1: Interrupts due to VGPICHG_STA0 are enabled. |
| 23 to 19 | — | All 0 | R | Reserved |
| 18 | DPMSTAI _EN | B'0 | R/W | This bit enables DPMSTAI_STA interrupts. 0: Interrupts due to DPMSTAI_STA are disabled. 1: Interrupts due to DPMSTAI_STA are enabled. |
| 17 | DPMONCH _EN | B'0 | R/W | This bit enables DPMONCHG_STA interrupts. 0: Interrupts due to DPMONCHG_STA are disabled. 1: Interrupts due to DPMONCHG_STA are enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 16 | DMMONCHG_EN | B'0 | R/W | This bit enables DMMONCHG_STA interrupts. 0: Interrupts due to DMMONCHG_STA are disabled. 1: Interrupts due to DMMONCHG_STA are enabled. |
| 15 to 13 | — | All 0 | R | Reserved |
| 12 | SESSVLDCHG_EN* | B'0 | R/W | This bit enables SESSVLDCHG_STA interrupts. 0: Interrupts due to SESSVLDCHG_STA are disabled. 1: Interrupts due to SESSVLDCHG_STA are enabled. |
| 11 | IDDIGCHG_EN | B'0 | R/W | IDDIGCHG_STA interrupt enable 0: Interrupts due to IDDIGCHG_STA are disabled. 1: Interrupts due to IDDIGCHG_STA are enabled. |
| 10 | VVLDCHG_EN* | B'0 | R/W | This bit enables VVLDCHG_STA interrupts. 0: Interrupts due to VVLDCHG_STA are disabled. 1: Interrupts due to VVLDCHG_STA are enabled. |
| 9 | APRBCHG_EN* | B'0 | R/W | This bit enables APRBCHG_STA interrupts. 0: Interrupts due to APRBCHG_STA are disabled. 1: Interrupts due to APRBCHG_STA are enabled. |
| 8 | ASNSCHG_EN* | B'0 | R/W | This bit enables ASNSCHG_STA interrupts. 0: Interrupts due to ASNSCHG_STA are disabled. 1: Interrupts due to ASNSCHG_STA are enabled. |
| 7, 6 | — | All 0 | R | Reserved |
| 5 | PDDETCHG2_EN | B'0 | R/W | This bit enables PDDETCHG2_STA interrupts. 0: Interrupts due to PDDETCHG2_STA are disabled. 1: Interrupts due to PDDETCHG2_STA are enabled. |
| 4 | PDDETCHG1_EN | B'0 | R/W | This bit enables PDDETCHG1_STA interrupts. 0: Interrupts due to PDDETCHG1_STA are disabled. 1: Interrupts due to PDDETCHG1_STA are enabled. |
| 3 | VBSTAINTE_N* | B'0 | R/W | This bit enables VBSTAINTE_STA interrupts. 0: Interrupts due to VBSTAINTE_STA are disabled. 1: Interrupts due to VBSTAINTE_STA are enabled. |
| 2 | VBSTACHG_EN* | B'0 | R/W | This bit enables VBSTACHG_STA interrupts. 0: Interrupts due to VBSTACHG_STA are disabled. 1: Interrupts due to VBSTACHG_STA are enabled. |
| 1 | OCINT_EN | B'0 | R/W | This bit enables OCINT_STA interrupts. 0: Interrupts due to OCINT_STA are disabled. 1: Interrupts due to OCINT_STA are enabled. |
| 0 | IDCHG_EN* | B'0 | R/W | This bit enables IDCHG_STA interrupts. 0: Interrupts due to IDCHG_STA are disabled. 1: Interrupts due to IDCHG_STA are enabled. |

Note: * These bits are not supported RZ/G2E

60.2.6.4 VBUS Control Register (offset: H'80C)

Register symbol: VBCTRL

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

This register controls VBUS when this LSI works as an OTG device.

| | | | | | | | | | | | | | | | | |
|----------------|------------|----|----|----|----|----|----|------------|------------|-----|-----|------------|-----|-----|------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | VBSTA[3:0] | | | | — | — | — | — | VBLVL[3:0] | | | | — | — | — | OCCLR EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | DRVVBUSSEL | VGPUO[3:0] | | | VGPLO[2:0] | | | VBOU | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 28 | VBSTA[3:0] | H'0 | R | These bits indicate the VBUS voltage level. The input to the EXT_VBSTA_IN[3:0] core pins is shown without change. |
| 27 to 24 | — | H'0 | R | Reserved |
| 23 to 20 | VBLVL[3:0] | H'0 | R/W | These bits are used to detect through an interrupt that VBUS reaches a specified level. When the value of the EXT_VBSTA_IN[3:0] core pins reaches the value specified in these bits, OBINTSTA.VBSTAINT_STA is set and a corresponding interrupt occurs if its occurrence is not masked. |
| 19 to 17 | — | All 0 | R | Reserved |
| 16 | OCCLR EN | B'0 | R/W | This bit specifies whether to automatically clear VBCTRL.VBOU and ADPCTRL.DRVVBUS when an overcurrent occurs. When an overcurrent occurs, OBINTSTA.OCINT_STA is set. 0: VBCTRL.VBOU and ADPCTRL.DRVVBUS are not automatically cleared when an overcurrent occurs. 1: VBCTRL.VBOU and ADPCTRL.DRVVBUS are automatically cleared when an overcurrent occurs. |
| 15 to 9 | — | All 0 | R | Reserved |
| 8 | DRVVBUSSEL | B'0 | R/W | This bit specifies whether to use VBCTRL.VBOU or ADPCTRL.DRVVBUS to control the VBUS output control pin (EXT_VBOU). 0: VBCTRL.VBOU is used. 1: ADPCTRL.DRVVBUS is used. The output logic level can be specified through the EXT_VGPLO_INV core pin. EXT_VGPLO_INV = 0: The value is output without change. EXT_VGPLO_INV = 1: The inverted value is output. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|---|
| 7 to 4 | VGPUO[3:0] | H'0 | R/W | <p>The value in these bits is output through the EXT_VGPUO[3:0] core pins.</p> <p>This value is used to control the external power-supply IC. The output logic level can be specified through the EXT_VGPUO_INV core pin.</p> <p>EXT_VGPUO_INV = 0: The value is output without change. EXT_VGPUO_INV = 1: The inverted value is output.</p> |
| 3 to 1 | VGPLO[3:1] | B'000 | R/W | <p>The value in these bits is output through the EXT_VGPLO[3:0] core pins.</p> <p>This value is used to control the external power-supply IC. The output logic level can be specified through the EXT_VGPLO_INV core pin.</p> <p>EXT_VGPLO_INV = 0: The value is output without change. EXT_VGPLO_INV = 1: The inverted value is output.</p> |
| 0 | VBOUT | B'0 | R/W | <p>The VBOUT pin is one of the VBUS output control pins. It is used to control the external power-supply IC and assert VBUS.</p> <p>When VBCTRL.DRVVBUSSEL = 0, the EXT_VBOUT core pin reflects the value specified in this bit.</p> <p>0: VBUS output is disabled. 1: VBUS output is enabled.</p> <p>Whether to output this bit value to the EXT_VBOUT core pin without change or invert it before output can be selected through the EXT_VGPLO_INV core pin.</p> <p>EXT_VGPLO_INV = 0: The value is output without change. EXT_VGPLO_INV = 1: The inverted value is output.</p> <p>Whether to automatically clear this bit when an overcurrent occurs can be specified in OCCLREN (bit 16).</p> |

60.2.6.5 Line Control Port1 Register (offset: H'810)

Register symbol: LINECTRL1

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register mainly controls D+ and D- in port 1.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-------------|------|----|--------------|--------|----------|--------|---------|--------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | DPMLVL[1:0] | — | — | DPRPU_EN | DP_RPU | DPRPD_EN | DP_RPD | DMPD_EN | DM_RPD | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | CNEN_EN | CNEN | — | OPMODE_NODRV | — | — | DPMON | DMMON | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R/W | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 26 | — | All 0 | R | Reserved |
| 25, 24 | DPMLVL[1:0] | B'00 | R/W | These bits are used to detect through an interrupt that D+ and D- in port 1 change to specified states. Specify the state of D+ in port 1 in DPMLVL[1] and the state of D- in DPMLVL[0]; when both D+ and D- in port 1 become equal to the specified values (not when only either of the two is equal to the specified value), OBINTSTA.DPMINTSTA is set and a corresponding interrupt occurs if its occurrence is not masked. This function is used to detect a specified bus state such as J-state, K-state, or SE0. |
| 23, 22 | — | All 0 | R | Reserved |
| 21 | DPRPU_EN | B'0 | R/W | This bit enables the DP_RPU (bit 20) setting for pullup control of D+ in port 1. 0: DP_RPU (bit 20) setting for pullup control of D+ in port 1 is disabled. 1: DP_RPU (bit 20) setting for pullup control of D+ in port 1 is enabled. |
| 20 | DP_RPU | B'0 | R/W | This bit controls pullup of D+ in port 1. This control is enabled only when DPRPU_EN (bit 21) = 1. Note that if this bit is cleared to 0 when DPRPU_EN (bit 21) = 1, D+ is in the floating state while no device drives the USB bus. Note also that setting DPRPD_EN (bit 19) and DP_RPD (bit 18) to enabled is prohibited while this bit is set to enabled — that is, enabling both pullup and pulldown of D+ together is prohibited. 0: Pullup of D+ in port 1 is disabled. 1: Pullup of D+ in port 1 is enabled. |
| 19 | DPRPD_EN | B'0 | R/W | This bit enables the DP_RPD (bit 18) setting for pulldown control of D+ in port 1. 0: DP_RPD (bit 18) setting for pulldown control of D+ in port 1 is disabled. 1: DP_RPD (bit 18) setting for pulldown control of D+ in port 1 is enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|---|
| 18 | DP_RPD | B'0 | R/W | <p>This bit controls pulldown of D+ in port 1.</p> <p>This control is enabled only when DPRPD_EN (bit 19) = 1.</p> <p>Note that if this bit is cleared to 0 when DPRPD_EN (bit 19) = 1, D+ is in the floating state while no device drives the USB bus.</p> <p>Note also that setting DPRPU_EN (bit 21) and DP_RPU (bit 20) to enabled is prohibited while this bit is set to enabled — that is, enabling both pullup and pulldown of D+ together is prohibited.</p> <p>0: Pulldown of D+ in port 1 is turned off.</p> <p>1: Pulldown of D+ in port 1 is turned on.</p> |
| 17 | DMRPD_EN | B'0 | R/W | <p>This bit enables the DM_RPD (bit 16) setting for pulldown control of D- in port 1.</p> <p>0: DM_RPD (bit 16) setting for pulldown control of D- in port 1 is disabled.</p> <p>1: DM_RPD (bit 16) setting for pulldown control of D- in port 1 is enabled.</p> |
| 16 | DM_RPD | B'0 | R/W | <p>This bit controls pulldown of D- in port 1.</p> <p>This control is enabled only when DMRPD_EN (bit 17) = 1.</p> <p>Note that if this bit is cleared to 0 when DMRPD_EN (bit 17) = 1, D- is in the floating state while no device drives the USB bus.</p> <p>0: Pulldown of D- in port 1 is disabled.</p> <p>1: Pulldown of D- in port 1 is enabled</p> |
| 15 to 10 | — | All 0 | R | Reserved |
| 9 | CNEN_EN | B'0 | R/W | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>Not used.</p> <p>[RZ/G2E]</p> <p>Enable UTMI_P1CONNECT pin control by CNEN (bit 8).</p> <p>0: UTMI_P1CONNECT pin control by CNEN is disable.</p> <p>1: UTMI_P1CONNECT pin control by CNEN is enable.</p> |
| 8 | CNEN | B'0 | R/W | <p>[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>Not used.</p> <p>[RZ/G2E]</p> <p>This bit controls the single-end receiver in port 1.</p> <p>This control is valid only when CNEN_EN (bit 9) = 1.</p> <p>When this module is used in the peripheral mode, this function is used to prevent the USB signals from becoming Hi-Z and entering the floating state if nothing is connected through USB.</p> <p>Set this bit to 0 when nothing is connected in the peripheral mode.</p> <p>0: Single-end receiver in port 1 is disabled. (The value of the UTMI_P1CONNECT core pin becomes 0.)</p> <p>1: Single-end receiver in port 1 is enabled. (The value of the UTMI_P1CONNECT core pin becomes 1.)</p> |
| 7 | — | B'0 | R | Reserved |
| 6 | OPMODE_N ODRV | B'0 | R/W | <p>This pin controls OPMODE[1:0] in port 1 (UTMI_P1OPMODE) in the UTMI+ interface.</p> <p>Set this bit to 1 when OPMODE should be set to the non-driving state (B'01).</p> <p>0: UTMI_P1OPMODE[1:0] is controlled by the host or peripheral device.</p> <p>1: UTMI_P1OPMODE[1:0] is fixed to B'01 (non-driving state)</p> |
| 5, 4 | — | All 0 | R | Reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 3 | DPMON | B'0 | R | This bit indicates the state of the D+ signal in port 1. The value of the UTMI_P1LINESTATE[0] core pin is shown without change. |
| 2 | DMMON | B'0 | R | This bit indicates the state of the D- signal in port 1. The value of the UTMI_P1LINESTATE[1] core pin is shown without change. |
| 1 | — | B'0 | R | Reserved |
| 0 | — | B'0 | R | Reserved |

60.2.6.6 BC Control Port1 Register (offset: H'820)

Register symbol: BCCTRL1

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

This register controls battery charging for port 1.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---------------|---------------|----|----|-------------|-------------|--------------|-------------|--------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | BCPWR DWNB | PWRD WNB | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | PDDET STS | CHGD ETSTS | — | — | DCPM ODE | VDMS RCE | IDPSIN KE | VDPSR CE | IDMSIN KE | IDPSR CE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 26 | — | All 0 | R | Reserved |
| 25 | BCPWRDWNB | B'1 | R/W | Not used. This value should be left at 1. |
| 24 | PWRDWNB | B'1 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Not used. [RZ/G2E] This bit controls the PHY_P1PWRDWNB pin in port 1. 0: Power down state 1: Power on state |
| 23 to 10 | — | All 0 | R | Reserved |
| 9 | PDDETSTS | B'0 | R | This bit indicates the result of portable device detection in port 1. The value input to the PHY_P1PRTBLDET core pin is shown. (The CHGET value output from the PHY is shown.) |
| 8 | CHGDETSTS | B'0 | R | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Not used. [RZ/G2E] This bit indicates the result of Charging Port detection in port 1. The value input to the PHY_P1CHGDET core pin is shown. (The CHGDET value output from PHY is shown) |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|--------------------------|---------------|-----|---|
| 7, 6 | — | All 0 | R | Reserved |
| 5 | DCPMODE | B'0 | R/W | Not used. This value should be left at 0. |
| 4 | VDMSRCE (VDATSRCEENB) | B'0 | R/W | This bit controls the PHY VDATSRCEENB pin in port 1. This setting is valid when COMMCTRL.BC_PERI (bit 30) = 0. When BC_PERI = 0, this bit value is output to PHY_P1VDMSRCEENB to control VDATSRCEENB in the PHY. 0: VDAT_SRC is disabled. 1: VDAT_SRC is enabled. |
| 3 | IDPSINKE (VDATDETENB) | B'0 | R/W | This bit enables detection of Attach or Connect for port 1 in PHY battery charging. This setting is valid when COMMCTRL.BC_PERI (bit 30) = 0. When BC_PERI = 0, this bit value is output to PHY_P1PRTBLDETENB to control whether to enable or disable detection by CGDETSTS in the PHY. 0: Detection by CHDGETSTS is disabled. 1: Detection by CHDGETSTS is enabled. |
| 2 | VDPSRCE (DCDENB) | B'0 | R/W | This bit enables or disables the data contact detection for port 1 in the PHY. This setting is valid when COMMCTRL.BC_PERI (bit 30) = 0. When BC_PERI = 0, this bit value is output to PHY_P1VDPSRCEENB to control the DCDENB pin in the PHY. 0: Data contact detection is disabled. 1: Data contact detection is enabled. |
| 1 | IDMSINKE (VDATDETENB) | B'0 | R/W | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Not used. [RZ/G2E] This bit enables detection of Attach or Connect for port 1 in PHY battery charging when this module is Portable Device mode. This setting is valid when COMMCTRL.BC_PERI (bit 30) = 0. When BC_PERI = 0, this bit value is output to PHY_P1CHGDETENB to control whether to enable or disable detection by CHGDETSTS in the PHY. 0: Detection by CHGDETSTS is disabled. 1: Detection by CHGDETSTS is enabled. |
| 0 | IDPSRCE (CHARGSEL) | B'0 | R/W | This bit selects the battery charging source for port 1 in the PHY. This setting is valid when COMMCTRL.BC_PERI (bit 30) = 0. This bit selects whether to apply the VDAT_SRC voltage to D+ or D-. In the CDP mode, set this bit to 1 because the voltage should be applied to D-. 0: VDAT_SRC is applied to D+ and SINK for D+ is enabled. 1: VDAT_SRC is applied to D- and SINK for D- is enabled. |

60.2.6.7 ADP Control Register (offset: H'830)

Register symbol: ADPCTRL

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register controls the ADP function.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----------------|-------------------|----|----|--------------|----------------|----------------|---------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | OTGSE SSVLD | IDDIG | VBUSV ALID | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | VBUSV LDEXT | VBUSVLD EXTSEL | — | — | IDPULL UP | DRVVB US | OTGDI SABLE | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 21 | — | All 0 | R | Reserved |
| 20 | OTGSESSVLD | B'0 | R | This bit indicates the result from the OTG device session valid comparator. 0: The voltage on VBUS is below the OTG Device Session Valid threshold. 1: The voltage on VBUS is above the OTG Device Session Valid threshold. |
| 19 | IDDIG | B'0 | R | This bit indicates the result of ID pin sampling. 0: A mini-A plug is connected. 1: A mini-B plug is connected. |
| 18 | VBUSVALID | B'0 | R | This bit indicates the result from the VBUS valid comparator in the PHY. 0: VBUS is not valid. 1: VBUS is valid. |
| 17 to 10 | — | All 0 | R | Reserved |
| 9 | VBUSVLDEXT | B'0 | R/W | This bit controls the PHY_P1VBUSVLDEXT pin of this module and the VBUSVLDEXT pin of the PHY. The VBUSVLDEXT signal is valid and used for DP pullup control when VBUSVLDEXTSEL = 1. 0: DP pullup resistor is disabled when VBUSVDLEXTSEL = 1. 1: DP pullup resistor is enabled when VBUSVDLEXTSEL = 1. When this module is used for OTG, the VBUS supply is turned off. Therefore, to execute SRP as a B-device, set both this bit and VBUSVLDEXTSEL to 1. DP can be pulled up only when the UTMI+ interface is controlled (XCVRSELECT == 01, TERMSELECT == 1, and OPMODE != 01). |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------|---------------|-----|--|
| 8 | VBUSVLDEXTSEL | B'0 | R/W | <p>This bit controls the PHY_P1VBUSVLDEXTSEL pin of this module and the VBUSVLDEXTSEL pin of the PHY.</p> <p>The VBUSVLDEXTSEL signal is used to detect VBUS in the PHY and to select whether to control DP pullup by the session valid comparator in the PHY or through an external pin (PHY_P1 VBUSVLDEXT) of the core.</p> <p>0: Controlled by the session valid comparator in the PHY. 1: Controlled through the PHY_P1 VBUSVLDEXT core pin (VBUSVLDEXT pin of the PHY).</p> <p>When this module is used for OTG, the VBUS supply is turned off. Therefore, to execute SRP as a B-device, set both this bit and VBUSVLDEXT to 1.</p> <p>DP can be pulled up only when the UTMI+ interface is controlled (XCVRSELECT == 01, TERMSELECT == 1, and OPMODE != 01).</p> |
| 7, 6 | — | All 0 | R | Reserved |
| 5 | IDPULLUP | B'0 | R/W | <p>This bit controls the PHY_P1IDPULLUP pin of this module and the IDPULLUP pin of the PHY.</p> <p>Set this bit to 1 when the ID signal of a mini-AB plug is connected to ID0 of the PHY and sampling of the ID value should be enabled.</p> <p>The sampling result is indicated in ADPCTRL.IDDIG.</p> <p>0: ID sampling is disabled. 1: ID sampling is enabled.</p> |
| 4 | DRVVBUS | B'0 | R/W | <p>This bit controls the PHY_P1DRVVBUS pin of this module and the DRVVBUS pin of the PHY.</p> <p>This signal is input to the PHY and is also used to enable or disable VBUS control in the external charge pump.</p> <p>For the PHY, this signal enables or disables the VBUS valid comparator.</p> <p>0: DRVVBUS is set to 0. 1: DRVVBUS is set to 1.</p> |
| 3 | OTGDISABLE | B'0 | R/W | <p>This bit works as the VBUS valid comparator disable signal and controls the PHY_P1OTGDISABLE pin of this module and the OTGDISABLE pin of the PHY.</p> <p>Setting this bit to 1 turns off the power to the VBUS valid comparator. However, this bit does not affect the session valid comparator, ADP probing or sensing, or ID detection.</p> <p>When not using the OTG function, set this bit to 1 to reduce the power consumed by the VBUS valid comparator.</p> <p>0: VBUS valid comparator is enabled. 1: VBUS valid comparator is disabled.</p> |
| 2 to 0 | — | All 0 | R/W | Reserved |

60.3 Clock Specifications

60.3.1 Clock Gating Specifications

60.3.1.1 Overview

This module has a function for switching between the host and peripheral modes. In some case, no clock supply is needed for the unused circuits (stopping the clock causes no problem).

To reduce the power consumption through clock gating for the circuits that do not need clock supply, this module provides three clock control gating bits in the register enable/clock gating control register.

Control these bits of REGEN_CG_CTRL appropriately to gate the clock supply to some circuits in the host core and peripheral controller.

| Bit | Symbol |
|------|----------------|
| [31] | NONUSE_CLK_MSK |
| [29] | HOST_CLK_MSK |
| [28] | PERI_CLK_MSK |

It is the outside targeted of clock gating for the common function part (UCOM register and so on) in this core which has no dependence of core operation.

60.3.1.2 NONUSE_CLK_MSK Function Specifications

This bit is intended to reduce the power consumption when the function of this module is fixed to either the host or the peripheral and no clock needs to be supplied to the unused function or when no clock needs to be supplied to a function that is not assigned as the role of this module in OTG operation. After this bit is set, the clock is automatically gated for the host or peripheral function that is not assigned as the role.

The clock is gated as follows according to the target USB channel, USB0SEL[1:0] in UGCTRL2 register of HS-USB and the OTG_PERI bit setting.

Table 60.7 NONUSE_CLK_MSK Function

| Clock Gating Register Bit Settings | | | Host/Peripheral Switch Setting | Register Bit | | | |
|------------------------------------|--------------|--------------|--------------------------------|----------------|----------|-----------|------------|
| NONUSE_CLK_MSK | HOST_CLK_MSK | PERI_CLK_MSK | Channel | USB0SEL [1:0]* | OTG_PERI | Host | Peripheral |
| B'1 | B'0 | B'0 | Channel 1/2 | — | — | Gated | — |
| | | | Channel 0/3 | 2'b00 | — | Gated | Gated |
| | | | | 2'b01 | — | Not Gated | Gated |
| | | | | 2'b10 | — | Not Gated | Not Gated |
| | | | | 2'b11 | B'0 | Not Gated | Gated |
| | | | | | B'1 | Gated | Not Gated |

[Note on Using the NONUSE_CLK_MSK Bit]

When using the NONUSE_CLK_MSK bit, the HOST_CLK_MSK and PERI_CLK_MSK bits should be cleared to 0 in principle. If either of them is set to 1, the effect of clock gating is determined by a logical OR of all these bit settings.

Note: USB0SEL[1:0] is in UGCTRL2 register of HS-USB

60.3.1.3 PERI_CLK_MSK and HOST_CLK_MSK Function Specifications

The PERI_CLK_MSK and HOST_CLK_MSK bits respectively provide functions for forcibly gating the 60-MHz clock to the peripheral controller and the internal clock for the host core.

Their purposes and prerequisites are the same as those of NONUSE_CLK_MSK, but they provide manual gating functions that do not depend on the settings through the USB0SEL[1:0] in UGCTRL2 register of HS-USB and the OTG_PERI bit.

Table 60.8 PERI_CLK_MSK and HOST_CLK_MSK Function

| Clock Gating Register Bit Settings | | | Host/Peripheral Switch Setting | | Clock to be Gated | | |
|------------------------------------|--------------|--------------|--------------------------------|----------------|-------------------|-------|------------|
| NONUSE_CLK_MSK | HOST_CLK_MSK | PERI_CLK_MSK | Channel | Register Bit | | Host | Peripheral |
| | | | | USB0SEL [1:0]* | OTG_PERI | | |
| B'0 | B'1 | B'0 | Channel 1/2 | — | — | Gated | — |
| | | | Channel 0/3 | — | — | Gated | — |
| | B'0 | B'1 | Channel 1/2 | — | — | — | — |
| | | | Channel 0/3 | — | — | — | Gated |
| | B'1 | B'1 | Channel 1/2 | — | — | Gated | — |
| | | | Channel 0/3 | — | — | Gated | Gated |

[Note on Using the HOST_CLK_MSK and PERI_CLK_MSK Bits]

When using the HOST_CLK_MSK and PERI_CLK_MSK bits, the NONUSE_CLK_MSK bit should be cleared to 0 in principle. If it is set to 1, the effect of clock gating is determined by a logical OR of all these bit settings.

Note: USB0SEL[1:0] is in UGCTRL2 register of HS-USB

60.4 Reset Specifications

60.4.1 Input Reset Specifications

This module has a software reset input from CPG module. Asserting this reset signal initializes all FFs in this module.

60.5 Interrupt Specifications

60.5.1 Interrupt Signal List

This module has the following six interrupt signal lines.

Use all of them as level interrupts.

| Interrupt Pin Name | Interrupt Type | Pulse/ Level | Minimum Pulse Width | Active Level | Synchro- nized with |
|--------------------|--|-----------------|---------------------------|-----------------|---------------------------|
| U2H_INT | An interrupt signal output from the AHB_IF module. This is generated when a bus error occurs in the AHB master. It is controlled through the related AHB bridge register. | Level | — | H | AHB clock (HCLK)* |
| U2H_OHCI_INT | An interrupt signal output from OHCI. This is generated when data transfer is completed or a change in the USB bus state is detected in the FS or LS transfer. It is controlled through the related OHCI operational register. | Level | — | H | AHB clock (HCLK)* |
| U2H_EHCI_INT | An interrupt signal output from EHCI. This is generated when data transfer is completed or a change in the USB bus state is detected in the HS transfer. It is controlled through the related EHCI operational register. | Level | — | H | AHB clock (HCLK)* |
| U2H_WAKEON_INT | An interrupt signal output from EHCI. This is generated due to an EHCI wakeup event. It is controlled through the related EHCI operational register. | Level | — | H | AHB clock (HCLK)* |
| U2H_OBINT | An interrupt signal output from the UCOM2 module. This is generated due to an event related to OTG or battery charging. It is controlled through the related UCOM2 register. | Level | — | H | AHB clock (HCLK)* |
| U2H_BIND_INT | Logical OR of the interrupt signals. (U2H_INT + U2H_OHCI_INT + U2H_EHCI_INT + U2H_WAKEON_INT + U2H_OBINT) | Level | — | H | AHB clock (HCLK)* |

Note: Frequency of AHB clock (HCLK) is 133MHz (S3D2φ).

60.5.2 Interrupt Sources and Control

60.5.2.1 U2H_INT Assertion Source and Control

Interrupt Enable Control in Register:

This interrupt signal is asserted when an interrupt source event occurs while the interrupt enable bit in the following register is set to enabled (1).

AHB bridge register: INT_ENABLE register (offset: H'200) Bit [0] (AHB_INTEN)

Interrupt Source:

A bus error occurs (MHRESP = 1) in the AHB bridge.

Clearing Interrupt:

To clear the interrupt, write 1 to the following register bit.

AHB bridge register: INT_STATUS register (offset: H'204) Bit [0] (AHB_INT)

60.5.2.2 U2H_OHCI_INT Assertion Sources and Control

Interrupt Enable Control in Registers:

This interrupt signal is asserted when an interrupt source event occurs while the interrupt enable bits in the following registers are set to enabled (1).

AHB bridge register: INT_ENABLE register (offset: H'200) Bit [1] (USBH_INTAEN)

OHCI operational register: HcInterruptEnable Register (offset: H'010) Bit [31] and Bits [6:0]*

Note: * Set the necessary bits enabled as interrupt assertion source bits.

Interrupt Sources:

| Interrupt Source | Necessary Enable Settings in Register Bits | | | | | | | |
|---|--|-------------------|--------|--------|--------|--------|--------|--------|
| | USBH_INTAEN | HcInterruptEnable | | | | | | |
| | | Bit[31] | Bit[6] | Bit[5] | Bit[3] | Bit[2] | Bit[1] | Bit[0] |
| | MIE | RHSCE | FNOE | ROE | SFE | WDHE | SOE | |
| 1 Device connection is detected. | √ | √ | √ | — | — | — | — | — |
| 2 Device disconnection is detected. | √ | √ | √ | — | — | — | — | — |
| 3 Port power is turned off (overcurrent detection is not included). | √ | √ | √ | — | — | — | — | — |
| 4 Babble error is detected during USB transfer. | √ | √ | √ | — | — | — | — | — |
| 5 Resume process is completed. | √ | √ | √ | — | — | — | — | — |
| 6 Overcurrent is detected. | √ | √ | √ | — | — | — | — | — |
| 7 Bus reset process is completed. | √ | √ | √ | — | — | — | — | — |
| 8 OHCI enters the Operational state (HCFS[1:0] = B'10) or Suspend state (HCFS[1:0] = B'11) while the DR bit in the HcRhDescriptorB register is 1. | √ | √ | √ | — | — | — | — | — |
| 9 1 is written to the OHCI HcRhPort status bit [0] (ClearPortEnable) while no device is connected (CCS bit = 0). | √ | √ | √ | — | — | — | — | — |

| Interrupt Source | | Necessary Enable Settings in Register Bits | | | | | | | |
|------------------|---|--|-------------------|-----------------|----------------|---------------|---------------|----------------|---------------|
| | | USBH_ INTAEN | HcInterruptEnable | | | | | | |
| | | | Bit[31] MIE | Bit[6] RHSCE | Bit[5] FNOE | Bit[3] ROE | Bit[2] SFE | Bit[1] WDHE | Bit[0] SOE |
| 10 | 1 is written to the OHCI HcRhPort status bit [1] (SetPortEnable) while no device is connected (CCS bit = 0). | √ | √ | √ | — | — | — | — | — |
| 11 | 1 is written to the OHCI HcRhPort status bit [2] (SetPortSuspend) while no device is connected (CCS bit = 0). | √ | √ | √ | — | — | — | — | — |
| 12 | 1 is written to the OHCI HcRhPort status bit [3] (ClearSuspendStatus) while no device is connected (CCS bit = 0). | √ | √ | √ | — | — | — | — | — |
| 13 | 1 is written to the OHCI HcRhPort status bit [4] (SetPortReset) while no device is connected (CCS bit = 0). | √ | √ | √ | — | — | — | — | — |
| 14 | Port power is turned off while no device is connected (CCS bit = 0). | √ | √ | √ | — | — | — | — | — |
| 15 | MSB of bits [15:0] (FrameNumber) in the HcFmNumber register changes. | √ | √ | — | √ | — | — | — | — |
| 16 | RemoteWakeup signal (Resume signal) from a device is detected. | √ | √ | — | — | √ | — | — | — |
| 17 | HccaFrameNumber is updated (this means almost the same as an SOF transmission). | √ | √ | — | — | — | √ | — | — |
| 18 | Transfer (including an error) ends and the host core updates HccaDoneHead. | √ | √ | — | — | — | — | √ | — |
| 19 | USB schedule overrun occurs for a frame. | √ | √ | — | — | — | — | — | √ |

√: Necessary

Clearing Interrupt:

To clear the interrupt, write 1 to one of the following register bits that corresponds to the interrupt source.

OHCI operational register: HcInterruptStatus register (offset: H'00C) Bits [6:0]

60.5.2.3 U2H_EHCI_INT Assertion Sources and Control

Interrupt Enable Control in Registers:

This interrupt signal is asserted when an interrupt source event occurs while the interrupt enable bits in the following registers are set to enabled (1).

AHB bridge register: INT_ENABLE register (offset: H'200) Bit [2] (USBH_INTBEN)

EHCI operational register: USBINTR register (offset: H'128) Bits [17:16] and Bits [5:0]*

Note: * Set the necessary bits enabled as interrupt assertion source bits.

In addition, control the following register bits as necessary.

EHCI operational register: USBCMD register (offset: H'120) Bit [15] and Bit [6]

Interrupt Sources:

Necessary Enable Settings in Register Bits

| Interrupt Source | USBH_INTBEN | USBCMD | | USBINTR | | | | | | |
|---|-------------|-----------------------|----------|--------------|---------|---------------|-----------------|-------------|------------|---------|
| | | Bit[15] | Bit[6] | Bit[17] | Bit[16] | Bit[5] | Bit[3] | Bit[2] | Bit[1] | Bit[0] |
| | | Per-Port Change Event | Doorbell | Port-n Event | Change | Async Advance | Frame Roll-over | Port Change | USB ERRINT | USB INT |
| 1 Device connection is detected. | √ | — | — | — | — | — | — | √ | — | — |
| 2 Device disconnection is detected. | √ | — | — | — | — | — | — | √ | — | — |
| 3 Overcurrent is detected. | √ | — | — | — | — | — | — | √ | — | — |
| 4 RemoteWakeup signal (Resume signal) from a device is detected. | √ | — | — | — | — | — | — | √ | — | — |
| 5 Babble state is detected on the USB bus. | √ | — | — | — | — | — | — | √ | — | — |
| 6 USB transfer with qTD IOC = 1 is successfully completed. | √ | — | — | — | — | — | — | — | — | √ |
| 7 Short packet is received. | √ | — | — | — | — | — | — | — | — | √ |
| 8 USB transfer ends with an error. (Retry fails three times, a babble error is detected, or STALL is received.) | √ | — | — | — | — | — | — | — | √ | — |
| 9 QH processing with bit [6] (Interrupt on Async Advance Doorbell) in the USBCMD register set to 1 is successfully completed. | √ | — | √ | — | — | √ | — | — | — | — |
| 10 FrameIndex bit in the FRINDEX register returns from the maximum value to H'000 (rollover is detected). | √ | — | — | — | — | — | √ | — | — | — |
| 11 Port Change Detect event (interrupt sources 1 to 5) is detected in USB port 1. | √ | √ | — | — | √ | — | — | — | — | — |

√: Necessary

Clearing Interrupt:

To clear the interrupt, write 1 to one of the following register bits that corresponds to the interrupt source.

EHCI operational register: USBSTS register (offset: H'124) Bits [17:16] and Bits [5:0]

60.5.2.4 U2H_WAKEON_INT Assertion Sources and Control

Interrupt Enable Control in Registers:

This interrupt signal is asserted when an interrupt source event occurs while the interrupt enable bits in the following registers are set to enabled (1).

AHB bridge register: INT_ENABLE register (offset: H'200) Bit [4] (WAKEON_INTEN)
EHCI operational register: PORTSC[1:2] registers (offset: H'164 and H'168) Bits [22:20]*

Note: * Set the necessary bits enabled as interrupt assertion source bits.

Interrupt Sources:

| Interrupt Source | WAKEON_INTEN | Necessary Enable Settings in Register Bits | | |
|--|--------------|--|---------|---------|
| | | PORTSC | | |
| | | Bit[22] | Bit[21] | Bit[20] |
| 1 Device connection is detected. | √ | — | — | √ |
| 2 Device disconnection is detected. | √ | — | √ | — |
| 3 Overcurrent is detected. | √ | √ | — | — |
| 4 RemoteWakeup signal (Resume signal) from a device is detected. | √ | — | — | — |

√: Necessary

Clearing Interrupt:

To clear the interrupt, write 1 to the following register bit.

AHB bridge register: INT_STATUS register (offset: H'204) Bit [4] (WAKEON_INT)

60.5.2.5 U2H_OBINT Assertion Sources and Control

Interrupt Enable Control in Registers:

This interrupt signal is asserted when an interrupt source event occurs while the interrupt enable bits in the following registers are set to enabled (1).

AHB bridge register: INT_ENABLE register (offset: H'200) Bit [3] (UCOM_INTEN)

UCOM register*¹: OTG-BC interrupt enable register (offset: H'808) Bits [27:24], Bits [18:16], Bits [12:8], and Bits [3:0]*²

Notes: 1. Bits [27:24], Bits [12:8], and Bits [3:2] are not supported RZ/G2E

2. Set the necessary bits enabled as interrupt assertion source bits.

Interrupt Sources:

| Interrupt Source | | Necessary Enable Settings in Register Bits | | | | |
|------------------|---|--|-------------------------|-------------|-------------|---------|
| | | UCOM_INTEN | OTG-BC Interrupt Enable | | | |
| | | | Bit[27] | Bit[26] | Bit[25] | Bit[24] |
| | | VGPICHG_EN3 | VGPICHG_EN2 | VGPICHG_EN1 | VGPICHG_EN0 | |
| 1 | Bit 3 of the EXT_VGPI[3:0] input pins changes. * ¹ | √ | √ | — | — | |
| 2 | Bit 2 of the EXT_VGPI[3:0] input pins changes. * ¹ | √ | — | √ | — | |
| 3 | Bit 1 of the EXT_VGPI[3:0] input pins changes. * ¹ | √ | — | — | √ | |
| 4 | Bit 0 of the EXT_VGPI[3:0] input pins changes. * ¹ | √ | — | — | √ | |

| Interrupt Source | | Necessary Enable Settings in Register Bits | | | |
|------------------|---|--|-------------------------|-------------|---------|
| | | UCOM_INTEN | OTG-BC Interrupt Enable | | |
| | | | Bit[18] | Bit[17] | Bit[16] |
| | | DPMSTAIN_TEN | DPMONCHG_EN | DMMONCHG_EN | |
| 1 | Change of the USB bus state in port 1 into the same value as that specified in bits [25:24] (DPMLVL[1:0]) in the line control port1 register is detected. | √ | √ | — | |
| 2 | DP1 in the USB bus changes. | √ | — | √ | |
| 3 | DM1 in the USB bus changes. | √ | — | — | |

| Interrupt Source | | Necessary Enable Settings in Register Bits | | | | | |
|------------------|---|--|-------------------------|------------|------------|------------|--------|
| | | UCOM_INTEN | OTG-BC Interrupt Enable | | | | |
| | | | Bit[12] | Bit[11] | Bit[10] | Bit[9] | Bit[8] |
| | | SESSVLD_CHG_EN | IDDIGCHG_EN | VVLDCHG_NE | APRBCHG_EN | ASNSCHG_EN | |
| 1 | Value of the PHY_P1OTGSESSVLD input pin changes. * ¹ | √ | √ | — | — | — | |
| 2 | Value of the PHY_P1IDDIG input pin changes. * ¹ | √ | — | √ | — | — | |
| 3 | Value of the PHY_P1VBUSVALID input pin changes. * ¹ | √ | — | — | √ | — | |
| 4 | Value of the PHY_P1ADPPRB input pin changes. * ¹ | √ | — | — | — | √ | |
| 5 | Value of the PHY_P1ADPSNS input pin changes. * ¹ | √ | — | — | — | √ | |

| Interrupt Source | | Necessary Enable Settings in Register Bits | | |
|------------------|--|--|------------------|------------------|
| | | OTG-BC Interrupt Enable | | |
| | | UCOM_INT EN | PDDETCHG2_ EN | PDDETCHG1_ EN |
| 1 | Portable device detection signal (PHY_P1PRTBLDET) changes in port 1. | √ | — | √ |

| Interrupt Source | | Necessary Enable Settings in Register Bits | | | | |
|------------------|---|--|-----------------|-----------------|----------|----------|
| | | OTG-BC Interrupt Enable | | | | |
| | | UCOM_IN TEN | VBSTAIN T_EN | VBSTACHG _EN | OCINT_EN | IDCHG_EN |
| 1 | Change of the EXT_VBSTA_IN[3:0] pins into the same value as that specified in bits [23:20] (VBLVL[3:0]) in the VBUS control register is detected. | √ | √ | — | — | — |
| 2 | Value of the EXT_VBSTA_IN[3:0] input pins changes. *1 | √ | — | √ | — | — |
| 3 | Overcurrent is detected (change of EXT_P1OCI from 1 to 0 is detected). | √ | — | — | √ | — |
| 4 | Value of the EXT_OTG_ID input pin changes. *1 | √ | — | — | — | √ |

√: Necessary

Note: These bits are not supported RZ/G2E

Clearing Interrupt:

To clear the interrupt, write 1 to one of the following register bits that corresponds to the interrupt source.

UCOM register*1: OTG-BC interrupt status register (offset: H'804) Bits [27:24], Bits [18:16], Bits [12:8], and Bits [3:0]

Note: Bits [27:24], Bits [12:8], and Bits [3:2] are not supported RZ/G2E

60.5.3 Interrupt Signal Deassertion Timing

After a register is accessed to clear an interrupt source, it may take a certain period of time until the corresponding interrupt is actually cleared. Therefore, a measure should be taken to prevent misdetection of the previous interrupt between register access for clearing and correct detection of the next interrupt.

60.5.3.1 Connection of Interrupt Signal Lines

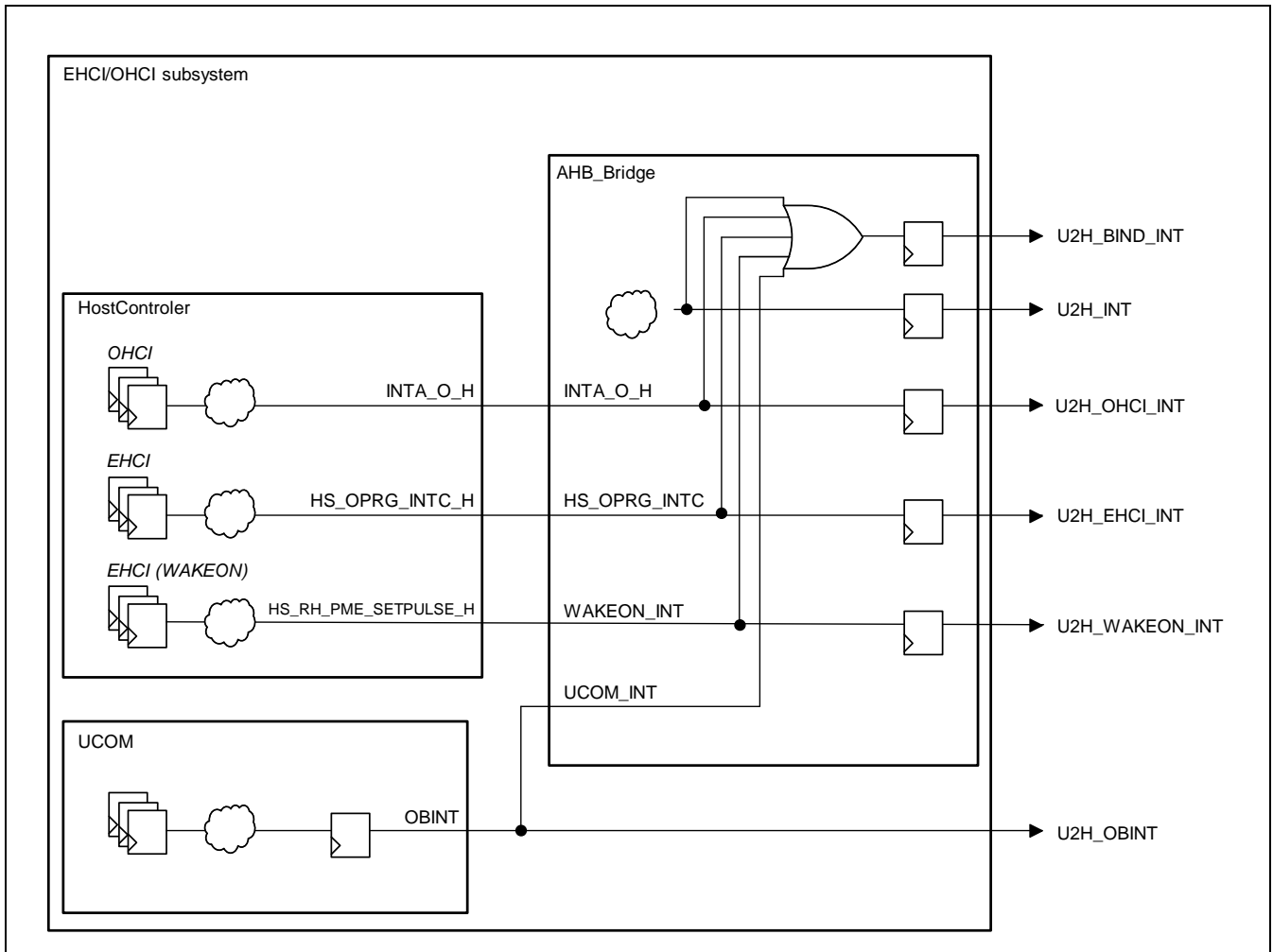


Figure 60.2 Connection diagram of Interrupt Signal Lines

60.6 Power Management

The system implementing this module provides the following two types of low-power control for the USB 2.0 core (including the UTMI+ PHY).

1. Controls the SUSPENDM and SLEEPM pins of the UTMI+ PHY.
2. Stops unnecessary clock supply to the host core internal circuits and the peripheral core through clock gating.

60.6.1 Controlling SUSPENDM and SLEEPM Pins of UTMI+ PHY

The following power-reducing effects are expected from assertion of the SUSPENDM and SLEEPM pins of the UTMI+ PHY.

- The power consumption in the UTMI+ PHY can be reduced.
- The power consumption in the host core can be reduced because the clock from the PHY is stopped.

The SUSPENDM and SLEEPM pins of the UTMI+ PHY are not asserted in the default settings even when the EHCI or OHCI is placed in the Suspend state as described earlier.

See section 60.2.5.3, Suspend Control Register (offset: H'308), and set up the register appropriately.

60.6.2 Controlling Clock Gating Function

The clock supply to the unused one of the host and peripheral functions can be gated through register settings.

See section 60.2.5.2, Register Enable/Clock Gating Control Register (offset: H'304), and set up the register appropriately.

60.7 OTG Control

60.7.1 Conditions for Using This Module for OTG

60.7.1.1 Core Pin and define Item Settings

Make the following settings and define items shown below when using this module for OTG.

1. Set the USB0SEL[1:0] in UGCTRL2 register of HS-USB module to B'11.

60.7.1.2 ID Pin Connections

This module has an ID detection pin; use this pin to detect ID.

60.7.1.3 Port Power (VBUS) Control and Connection with External Power-Supply IC

The following describes how to control the port power (VBUS) when using this module for OTG.

1. Control the port power through the related UCOM register.
An example for connection is described later.
2. Connect the external power-supply IC and the corresponding pins of this module as follows.

| LSI Pin Name | I/O | Connection |
|--|-----|---|
| [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] USB*_PWEN [RZ/G2E] USB0_PWEN_A/B | O | Connect this as output to the power-supply IC. |
| [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] USB*_OVC [RZ/G2E] USB0_OVC_A/B | I | Connect this as input from the power-supply IC. |

Notes: OTG function isn't supported RZ/G2E

** in LSI Pin Name column are 0 or 1 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N.

60.7.1.4 OTG System Configuration Example

The following shows an example of connections to support the OTG function[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N].

This configuration includes an external power-supply IC and a peripheral controller (HS-USB module).

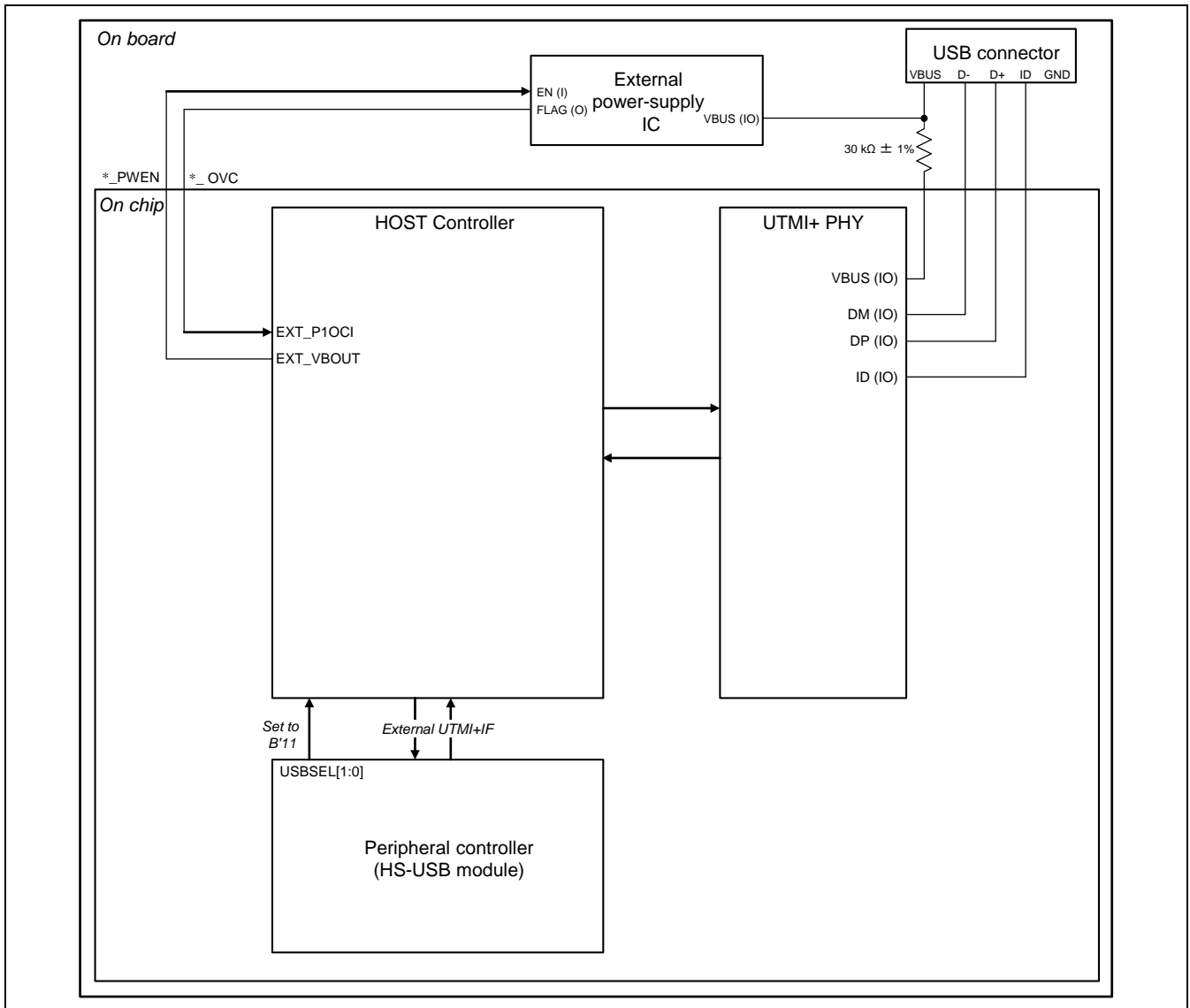


Figure 60.3 OTG System Configuration Example

60.7.2 Initial Setting Procedure

The following shows the initial setting procedure when this module is used for OTG.

This shows the steps before detection of A-plug or B-plug connection through the ID or VBUS pin.

60.7.2.1 OTG Initial Setting Procedure

Assumed control of ID and VBUS: VBUS control: ADPCTRL.DRVVBUS
 ID detection: PHY

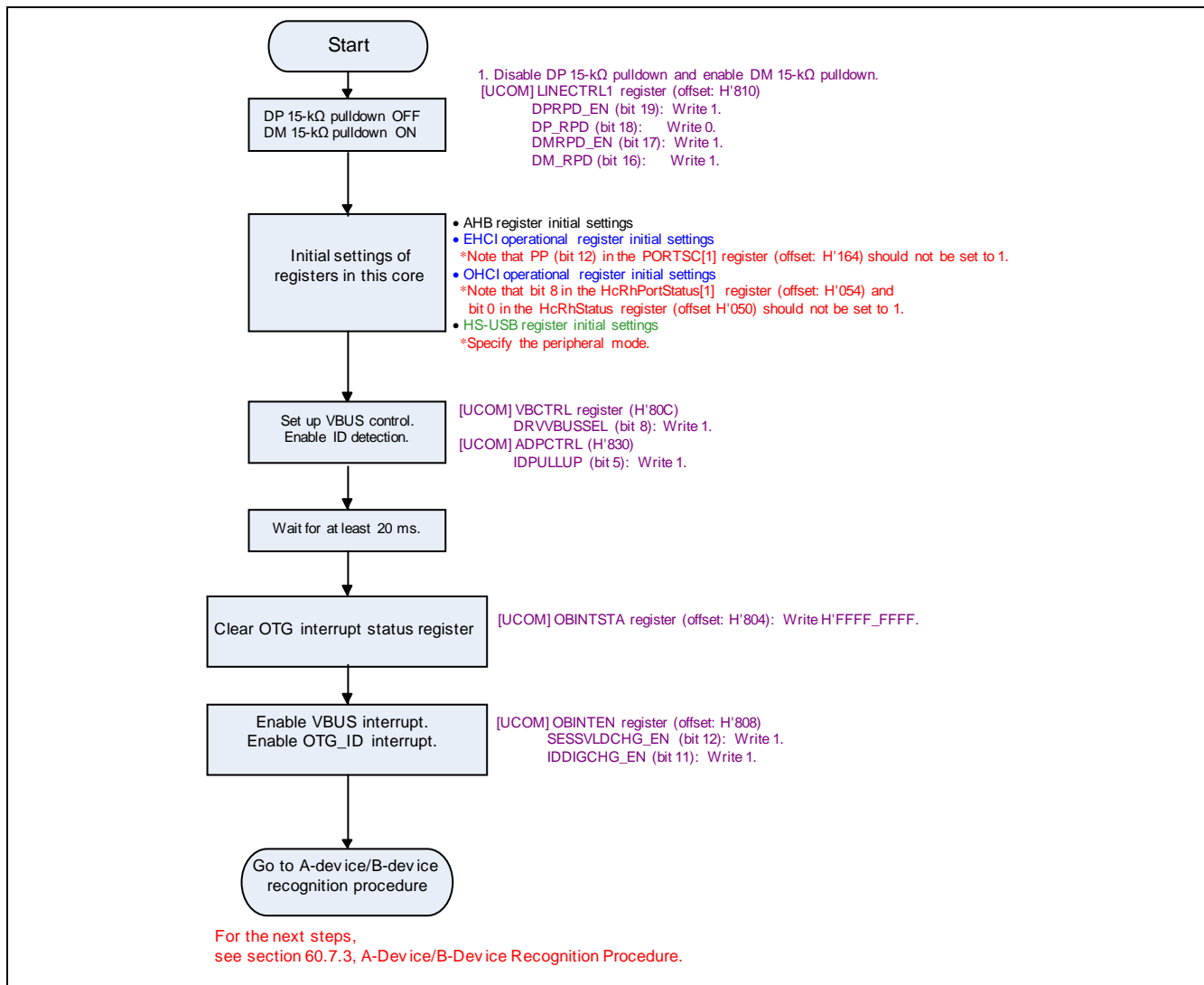


Figure 60.4 OTG Initial Setting Procedure

60.7.3 A-Device/B-Device Recognition Procedure

The following shows the control procedure for operating this module as an A-device or a B-device through A-plug or B-plug connection after the steps described in section 60.7.2, Initial Setting Procedure, or section 60.7.4, Disconnect Detection Procedure.

60.7.3.1 A-Device/B-Device Recognition Procedure

Assumed control of ID and VBUS:

VBUS control: ADPCTRL.DRVVBUS

ID detection: PHY

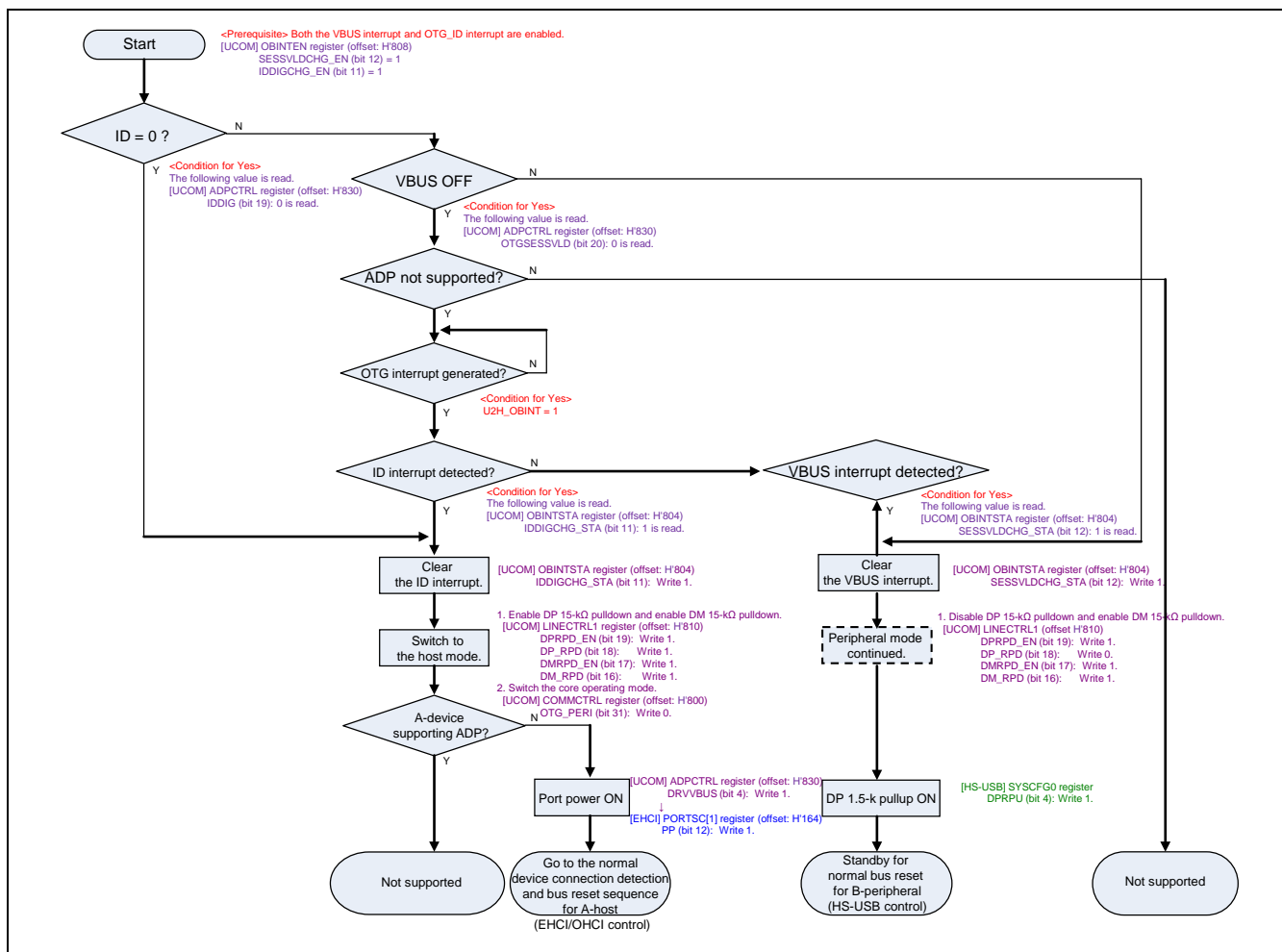


Figure 60.5 A-Device/B-Device Recognition Procedure

60.7.4 Disconnect Detection Procedure

The following shows the Disconnect detection procedure.

Here, "Disconnect detection" means detection of physical disconnection of an A-plug or B-plug from a USB port.

60.7.4.1 Disconnect Detection Procedure

Assumed control of ID and VBUS:

VBUS control: ADPCTRL.DRVVBUS

ID detection: PHY

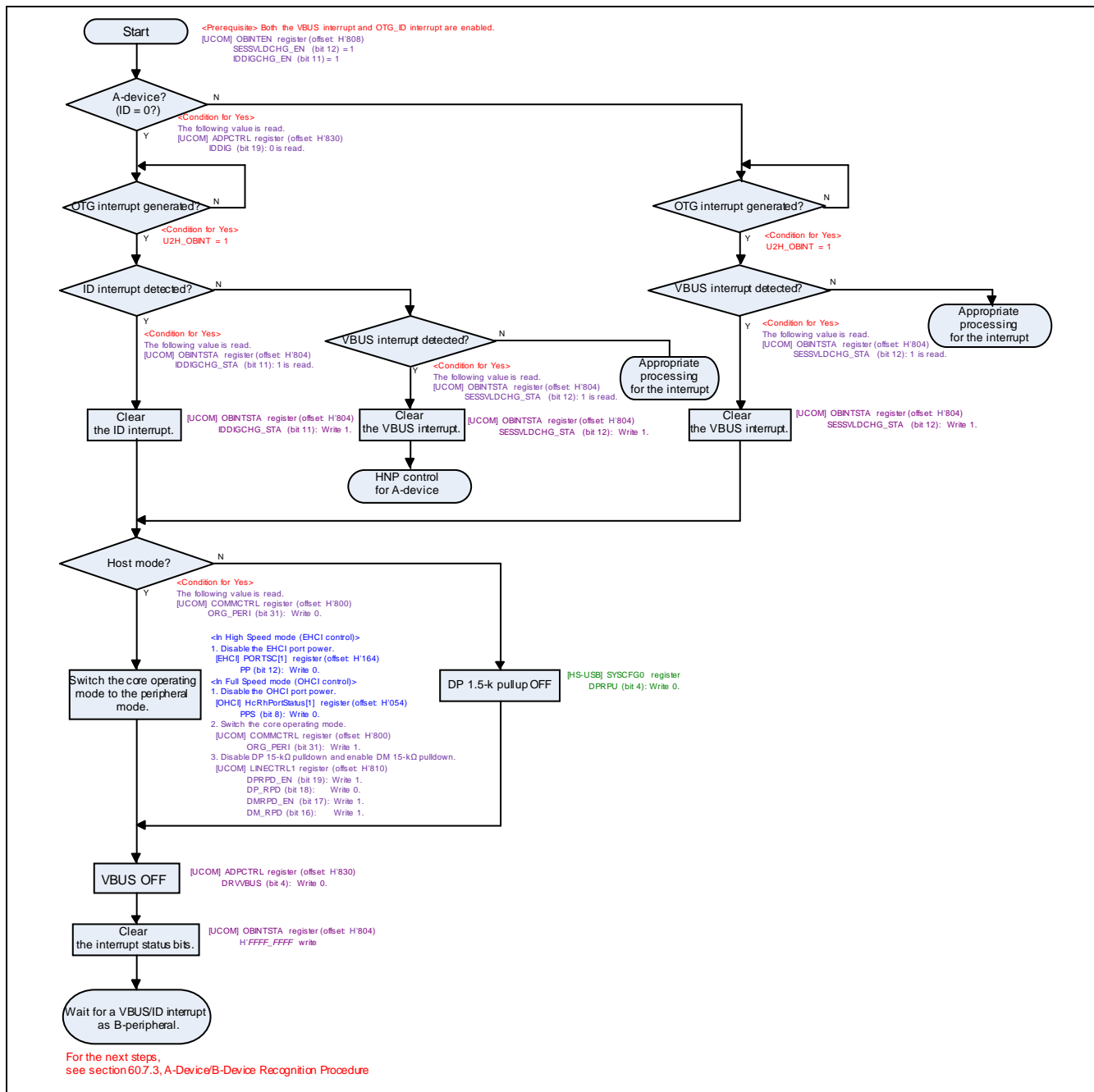


Figure 60.6 Disconnect Detection Procedure

60.7.5 HNP Control Procedure (Conforming to OTG Revision 2.0)

60.7.5.1 HNP Control Procedure for A-Device (Switching from A-Host to A-Peripheral)

The following shows the control procedure after a B-device's request for the host role is detected from the result of HNP polling for the B-device.

For HNP Control in High Speed Mode (Controlled by EHCI):

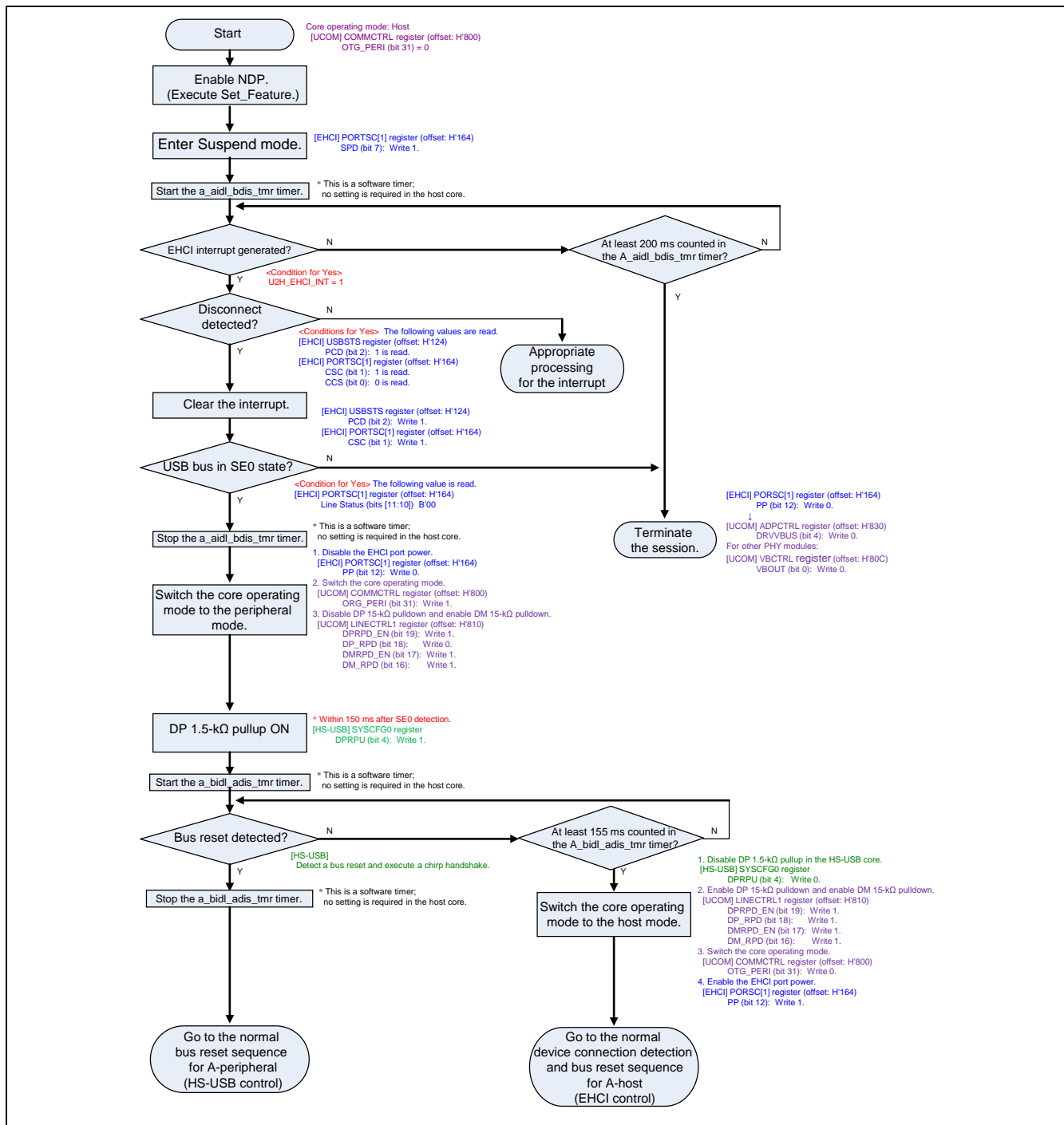


Figure 60.7 HNP Control Procedure for A-Device (Switching from A-Host to A-Peripheral in High Speed Mode)

For HNP Control in Full Speed Mode (Controlled by OHCI):

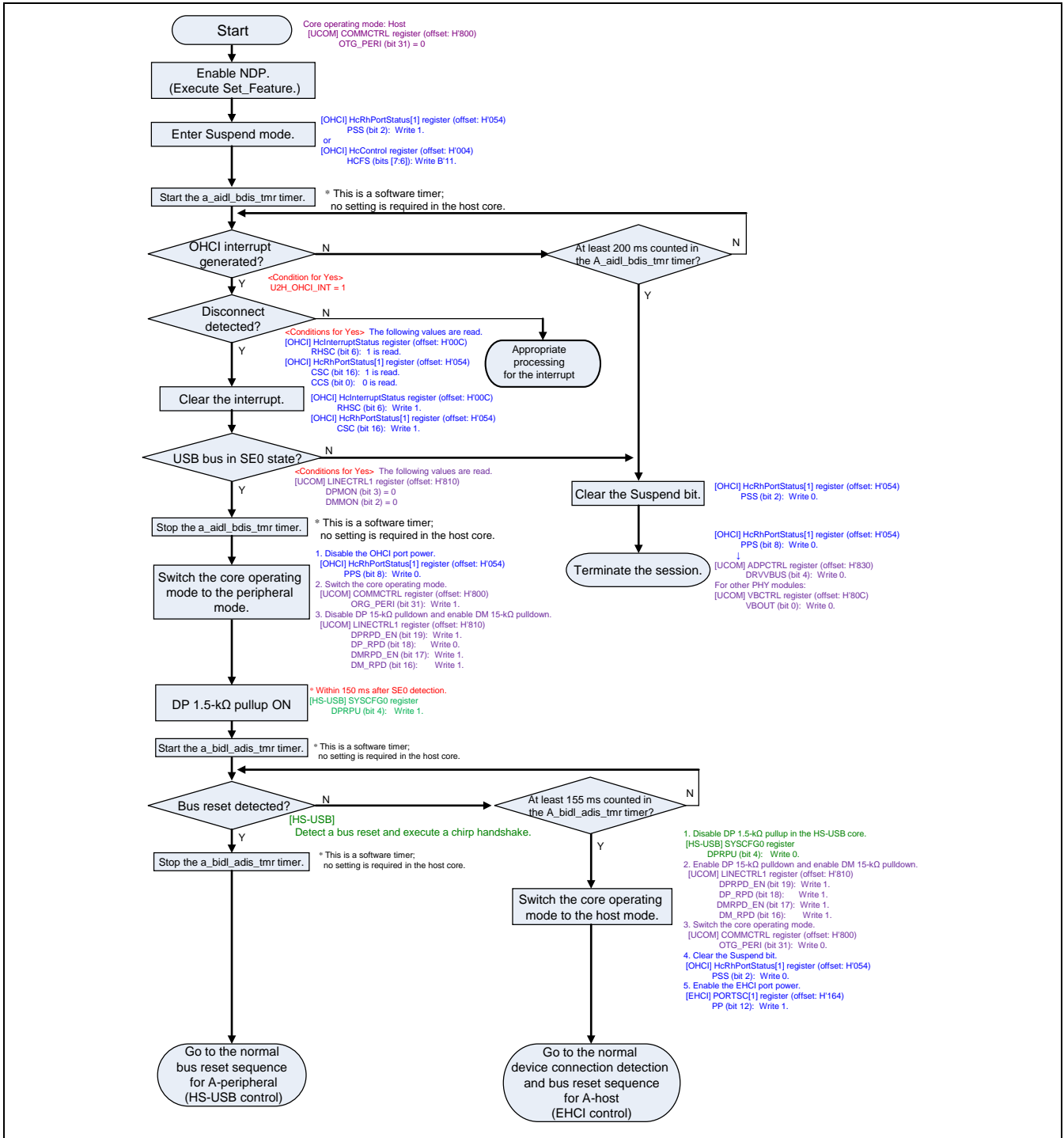


Figure 60.8 HNP Control Procedure for A-Device (Switching from A-Host to A-Peripheral in Full Speed Mode)

60.7.5.2 HNP Control Procedure for A-Device (Switching from A-Peripheral to A-Host)

<Assumed Conditions>

Even for FS OTG, EHCI is assumed to be the port owner after switching of the role from the peripheral to the host. Therefore, the procedure for switching from the A-peripheral to the A-host is not separately prepared for each of OHCI and EHCI.

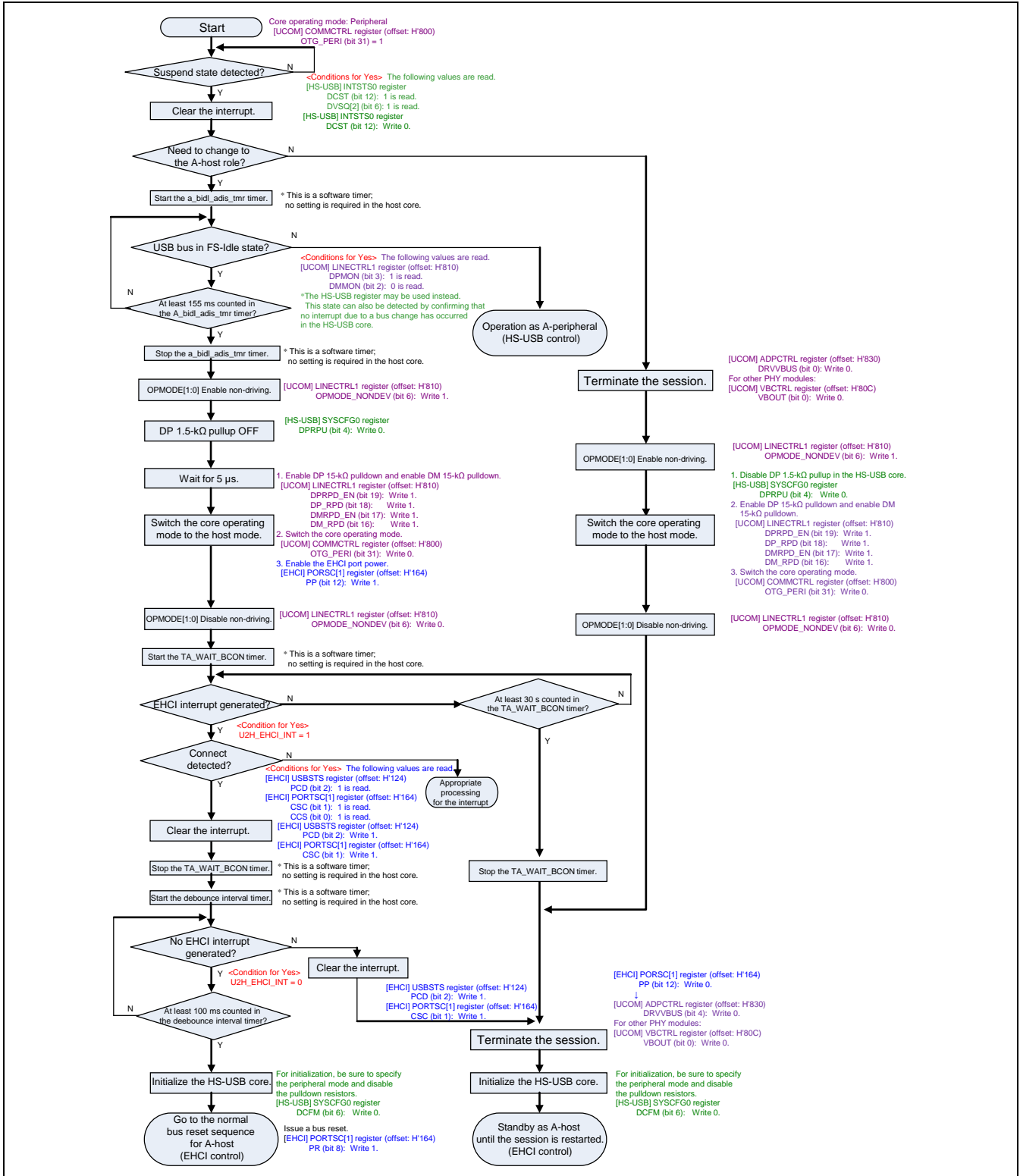


Figure 60.9 HNP Control Procedure for A-Device (Switching from A-Peripheral to A-Host)

60.7.5.3 HNP Control Procedure for B-Device (Switching from B-Peripheral to B-Host)

The following shows the procedure after the A-device issues Set Feature.

<Assumed conditions>

Even for FS OTG, EHCI is assumed to be the port owner after switching of the role from the peripheral to the host. Therefore, the procedure for switching from the B-peripheral to the B-host is not separately prepared for each of OHCI and EHCI.

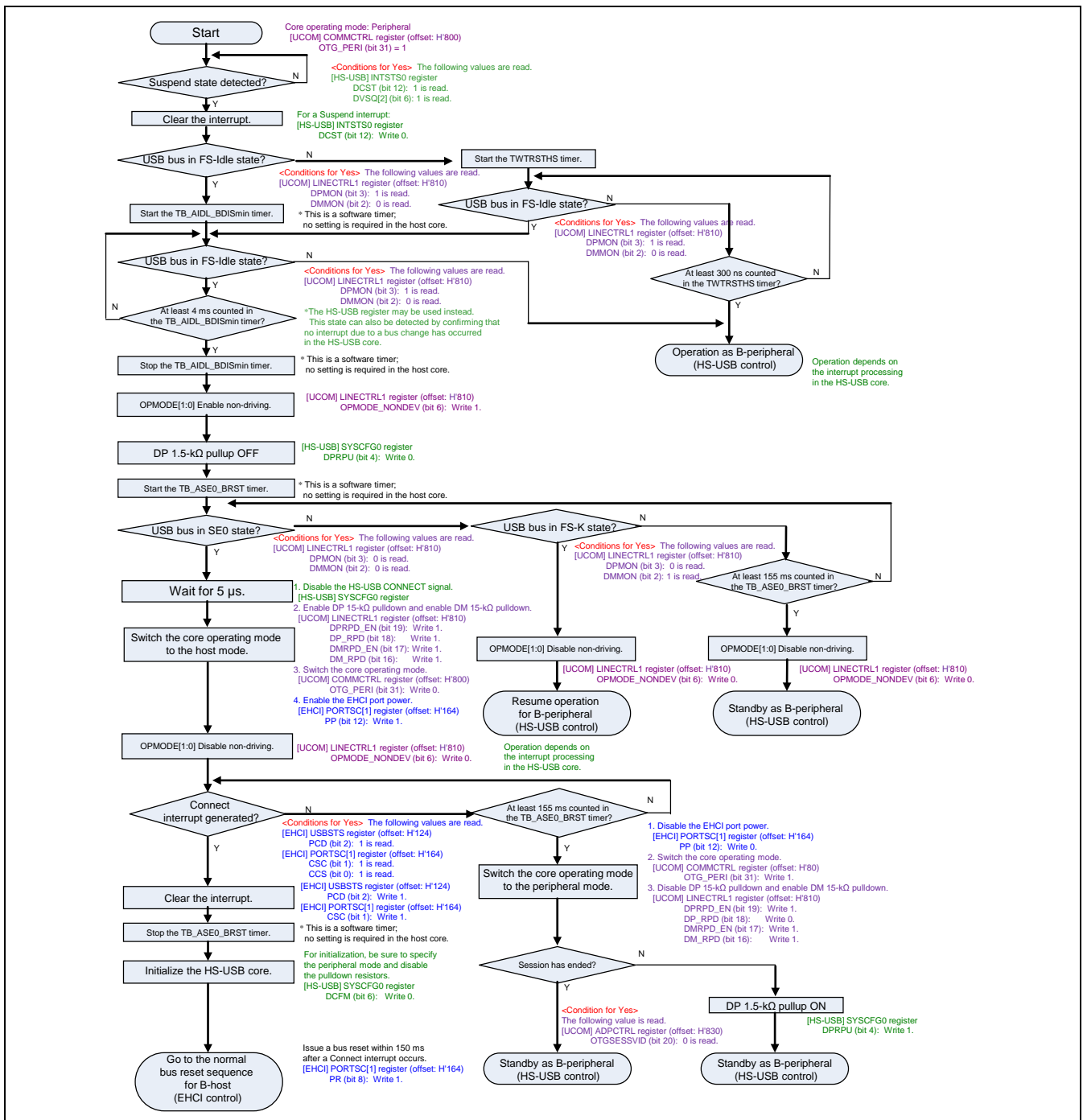


Figure 60.10 HNP Control Procedure for B-Device (Switching from B-Peripheral to B-Host)

60.7.5.4 HNP Control Procedure for B-Device (Switching from B-Host to B-Peripheral)

The following shows the procedure after no transfer is done by the B-Host.

<VBUS control: ADPCTRL.DRVVBUS; ID detection: PHY>

For HNP Control in High Speed Mode (Controlled by EHCI):

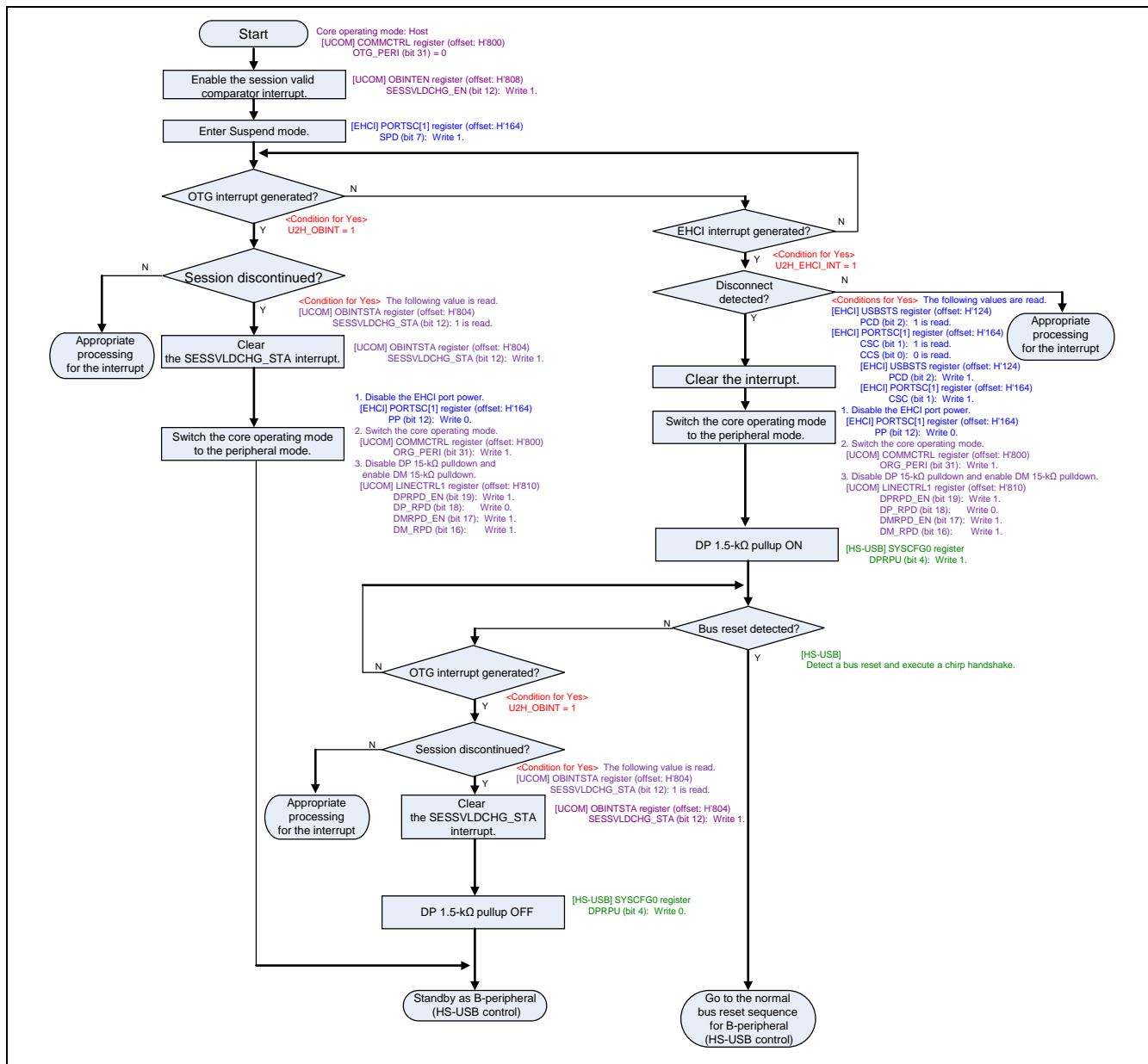


Figure 60.11 HNP Control Procedure for B-Device (Switching from B-Host to B-Peripheral in High Speed Mode)

For HNP Control in Full Speed Mode (Controlled by OHCI):

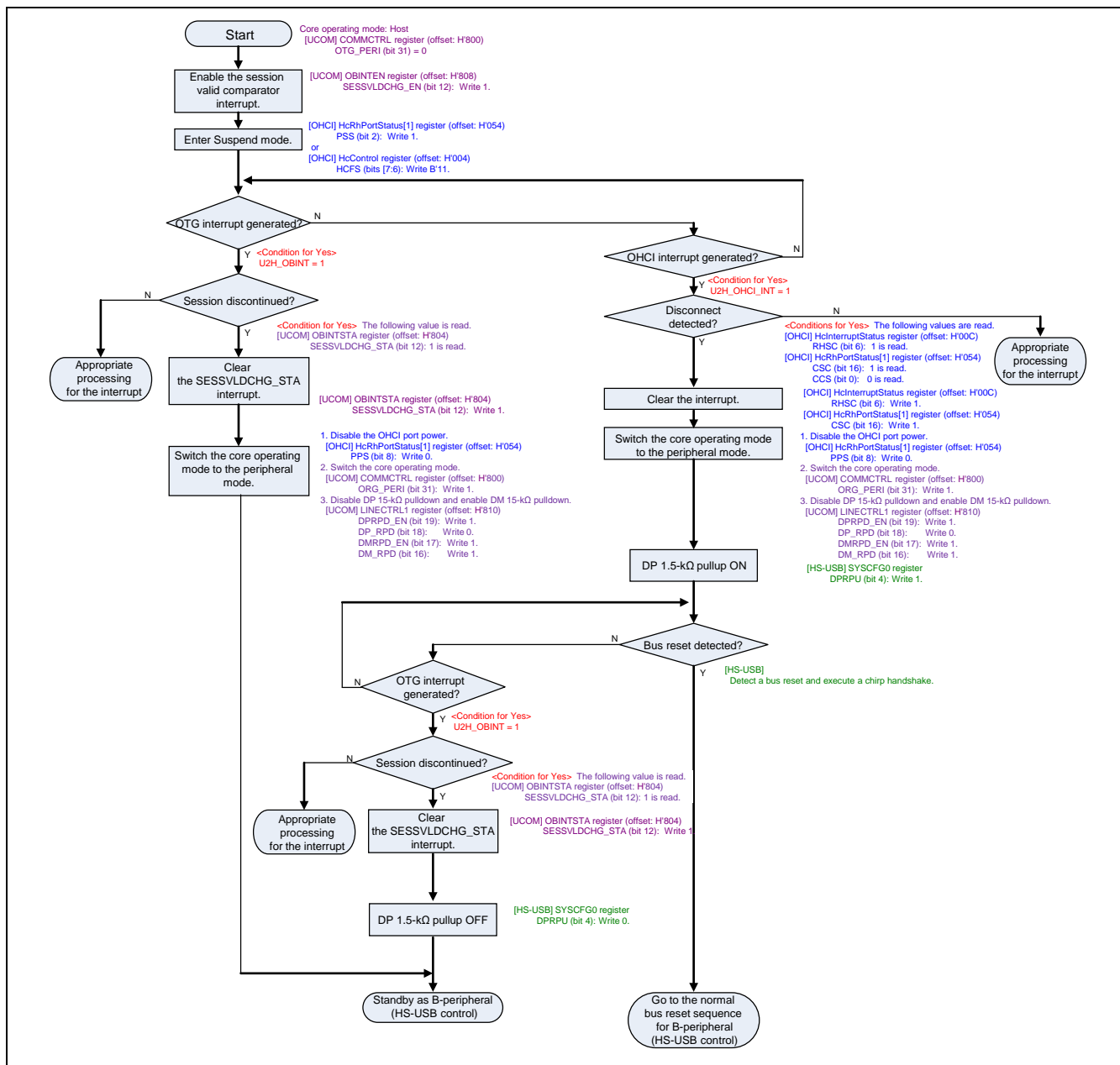


Figure 60.12 HNP Control Procedure for B-Device (Switching from B-Host to B-Peripheral in Full Speed Mode)

60.7.6 SRP Control Procedure (Conforming to OTG Revision 2.0)

60.7.6.1 SRP Control Procedure for A-Device

The following shows the procedure starting with the step for terminating a session because the B-device issues no request for the host role and the A-device does not need to transfer data.

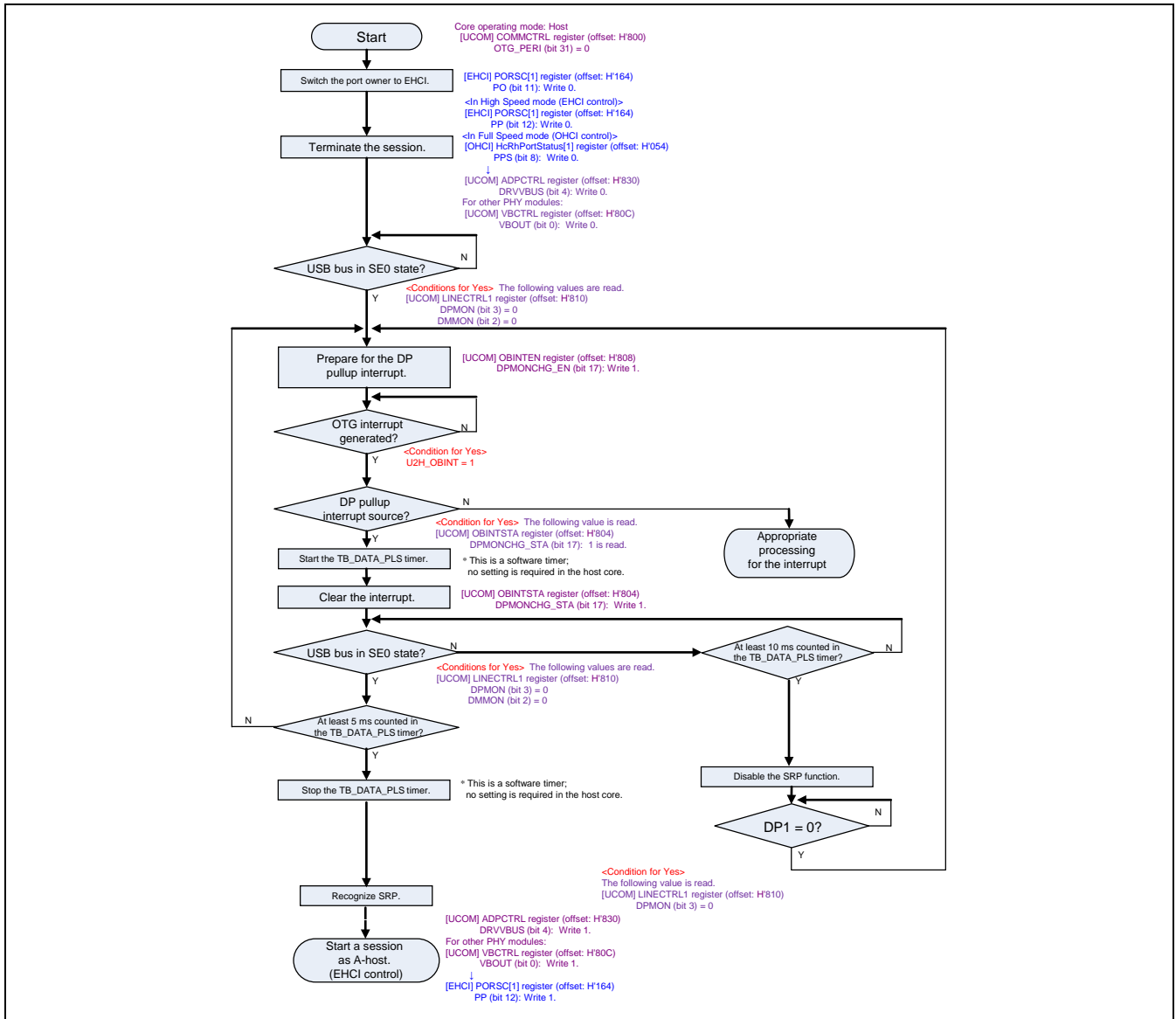


Figure 60.13 SRP Control Procedure for A-Device

60.7.6.2 SRP Control Procedure for B-Device

The following shows the procedure starting from the state of standby for a session termination request from the A-device.

<VBUS control: ADPCTRL.DRVVBUS; ID detection: PHY>

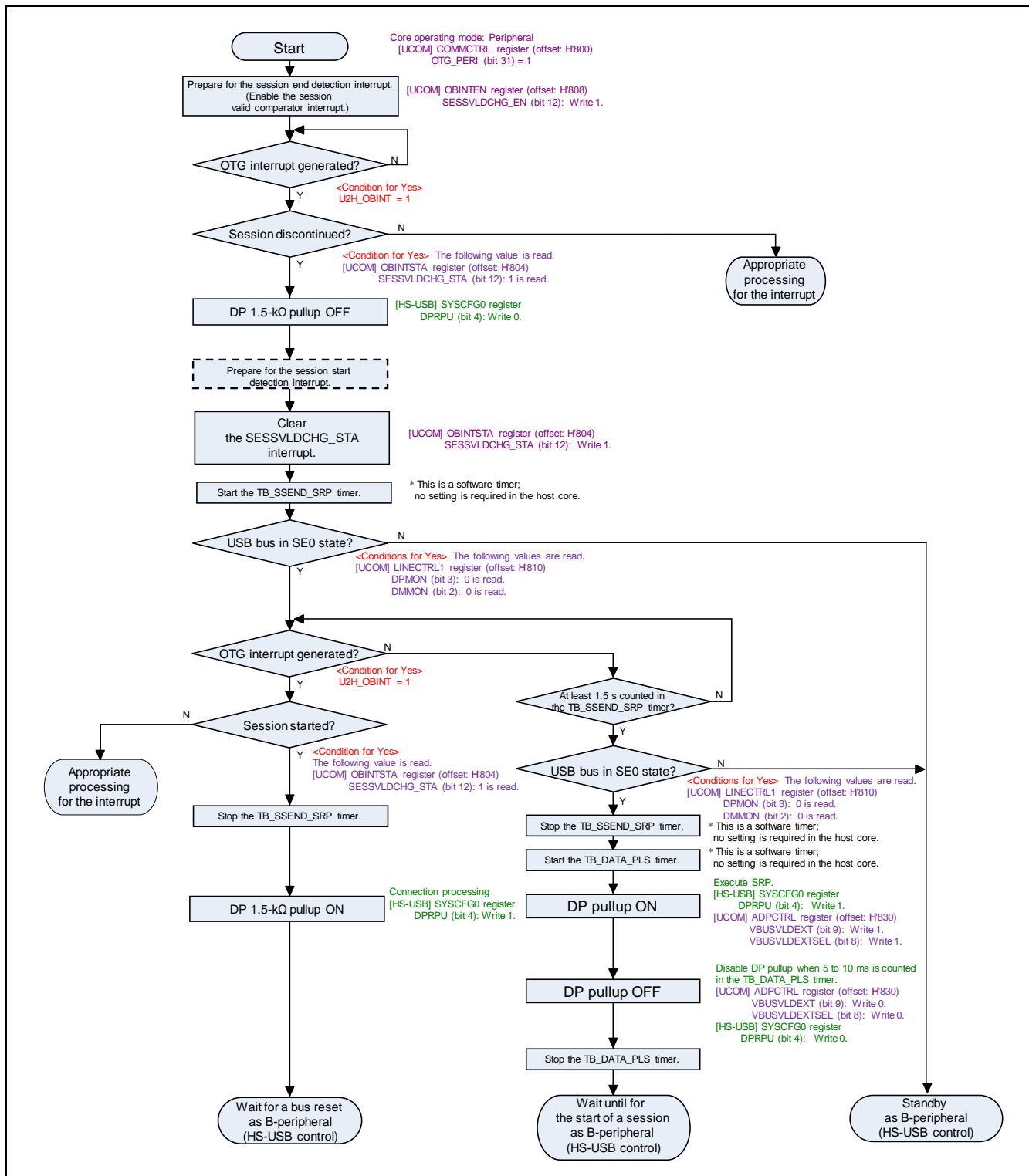


Figure 60.14 SRP Control Procedure for B-Device

60.8 Battery Charging Control

60.8.1 Conditions for Using Battery Charging Function in this Module

60.8.1.1 Limitations on Support of Charging Ports

A port that supplies power conforming to the Battery Charging Specification is called a charging port, and it is implemented in the host side in most cases.

Charging Ports are classified into the following types according to the function.

| Type | Function | Support in This Module |
|--------------------------------|--|------------------------|
| CDP (Charging Downstream Port) | This is a downstream port that supplies power conforming to the Battery Charging Specification. After a portable device is detected and a handshake for battery charging is completed, the normal device connect sequence is entered (the port works as a normal host). | Supported |
| DCP (Dedicated Charging Port) | This is a port dedicated to the power supply function conforming to the Battery Charging Specification. This port does not perform normal host operation. | — |
| SDP (Standard Downstream Port) | This is a normal downstream port that does not conform to the Battery Charging specification. It supplies power within the conventional power range prescribed in the USB 2.0 specification. It works as a normal host. | Supported |

60.8.1.2 Portable Device Control

A device that receives (requests) power supply conforming to the Battery Charging Specification is called a portable device, and it is implemented in the peripheral side in most cases.

This module assumes the Renesas HS-USB module as the peripheral controller.

A portable device can be controlled through either the HS-USB module or the registers of this module (UCOM registers).

Select a controller in accordance with the target system.

60.8.2 Controlling Method

The following describes control procedure for the Battery Charging function.

60.8.2.1 How to Control

(1) Notes on CDP Mode

Note the following two points in the CDP mode.

1. Use the following UCOM register bits to control the OPMODE[1:0] in the UTMI+ interface and 15-kΩ pulldown resistors for DP and DM as shown below.

| Register | Setting | Function |
|---------------------------------------|---------|--|
| [UCOM] LINECTRLx.OPMODE_NODRV (bit 6) | B'1 | Set UTMI_PxOPMODE[1:0] to output B'01 and set the USB bus control in the PHY to the non-driving state (actually, to the weak pulldown state). This bit is not supported RZ/G2E. |
| LINECTRLx.DPRPD_EN (bit 19) | B'1 | Disable the 1.5-kΩ pulldown resistor for DP. |
| LINECTRLx.DP_RPD (bit 18) | B'0 | |
| LINECTRLx.DMRPD_EN (bit 17) | B'1 | Disable the 1.5-kΩ pulldown resistor for DM. |
| LINECTRLx.DM_RPD (bit 16) | B'0 | |

2. Use the Global Suspend function to forcibly assert UTMI_PxSUSPENDM to 0.

| Register | Setting |
|--|---------|
| [Core Defined] Port 1: SPD_CTRL.GLOBAL_SUSPENDM_P1 (bit 0) | B'1 |

(2) CDP Mode Control

The following shows control on the port 1 side.

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Port 1: UTMI+ PHY SUSPENDM is Asserted in CDP Mode:

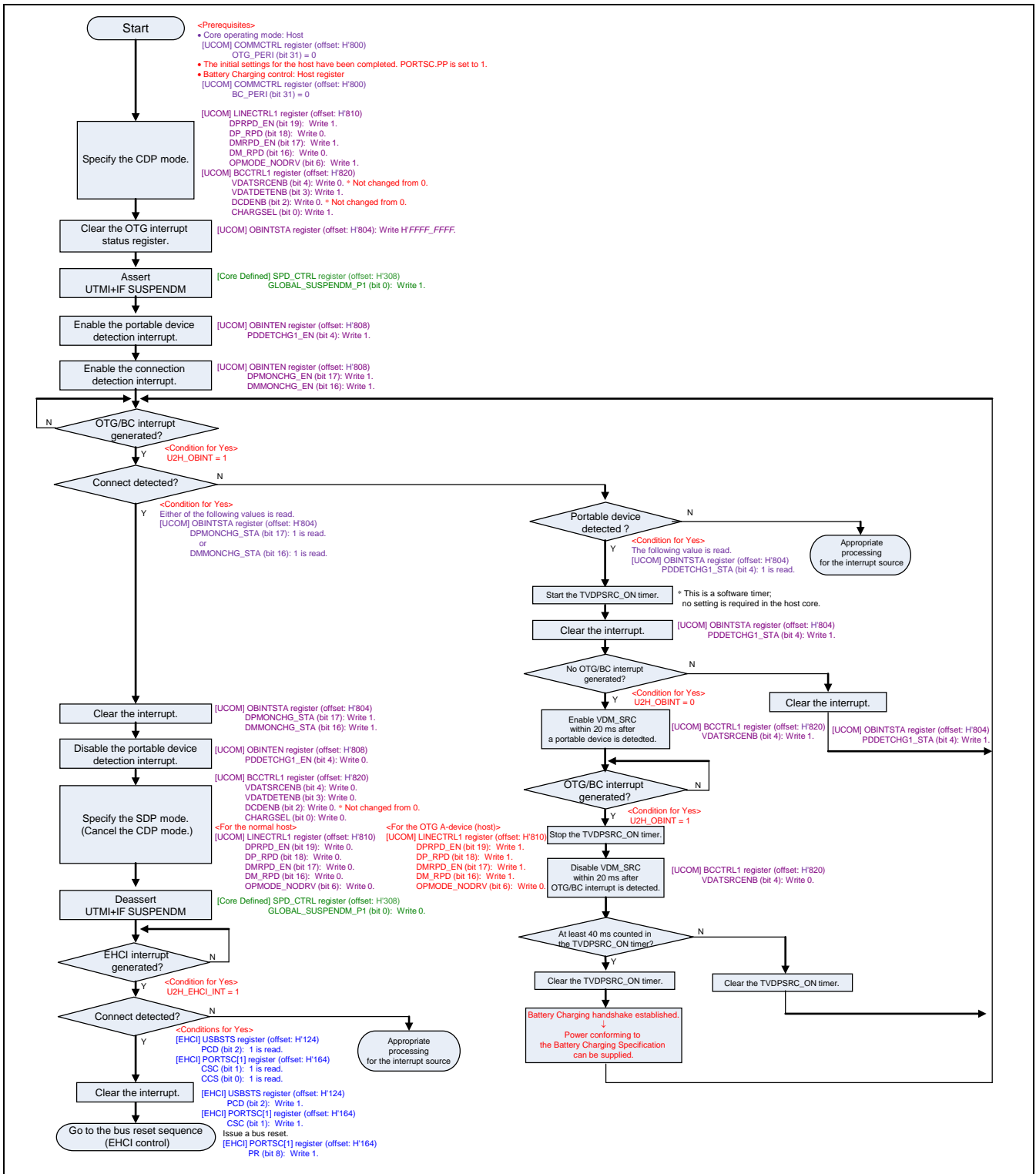


Figure 60.15 CDP Mode Control (UTMI+ PHY SUSPENDM is Asserted in CDP Mode)

Port 1: UTMI+ PHY SUSPENDM is Not Asserted in CDP Mode:

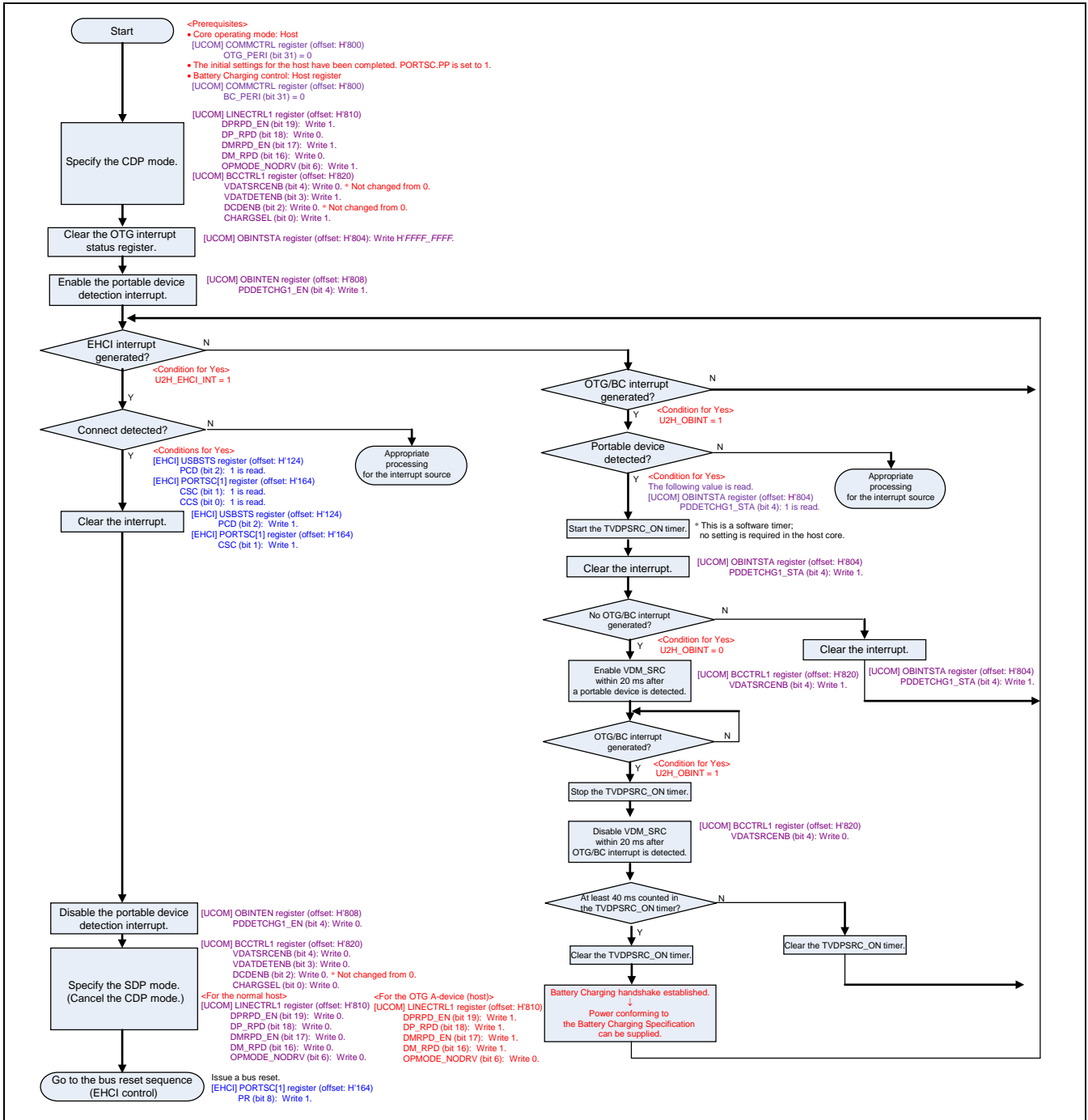


Figure 60.16 CDP Mode Control (UTMI+ PHY SUSPENDM is Not Asserted in CDP Mode)

[RZ/G2E]

Port 1: UTMI+ PHY SUSPENDM is Asserted in CDP Mode:

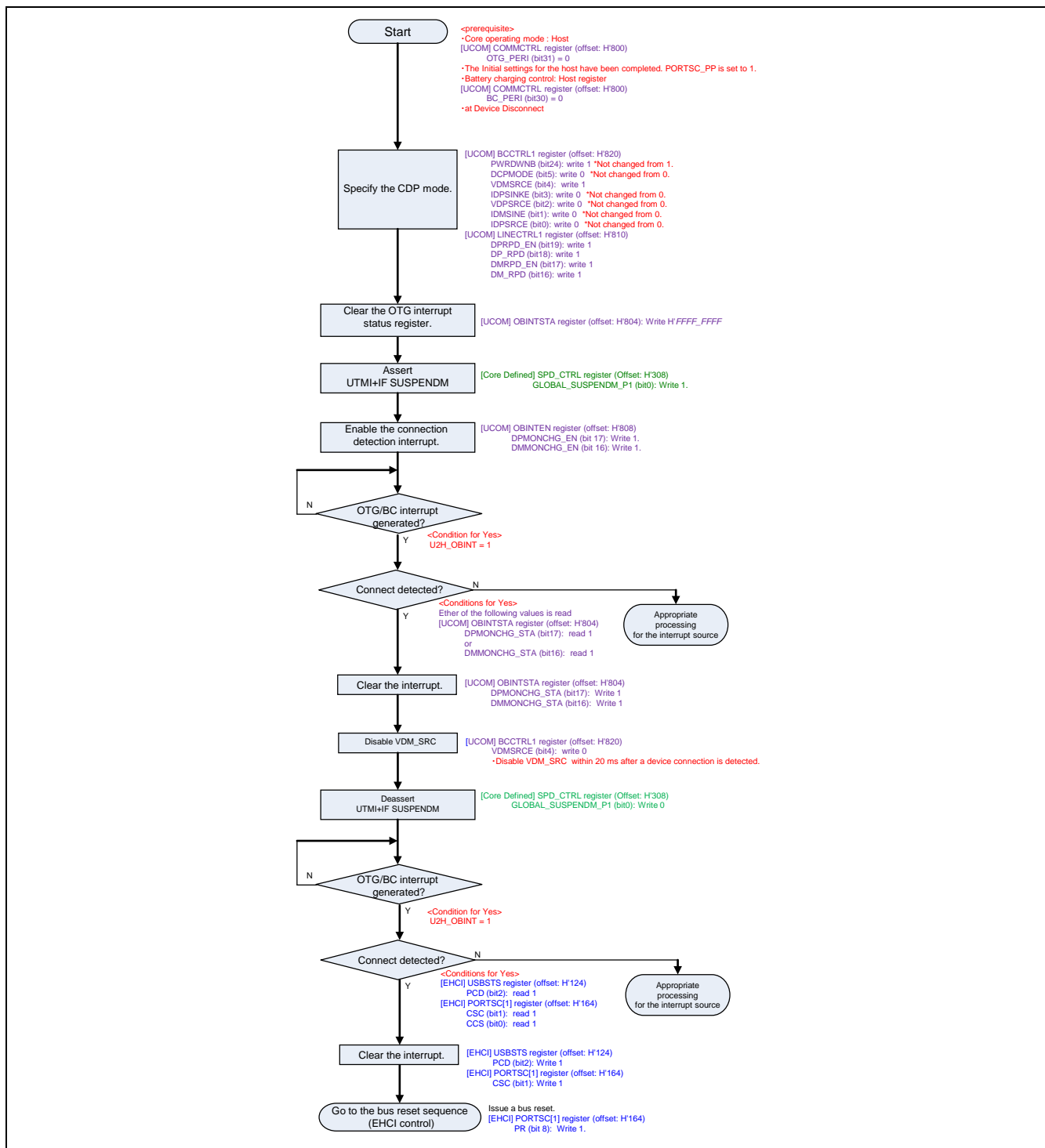


Figure 60.17 CDP Mode Control (UTMI+ PHY SUSPENDM is Asserted in CDP Mode)

Port 1: UTMI+ PHY SUSPENDM is Not Asserted in CDP Mode:

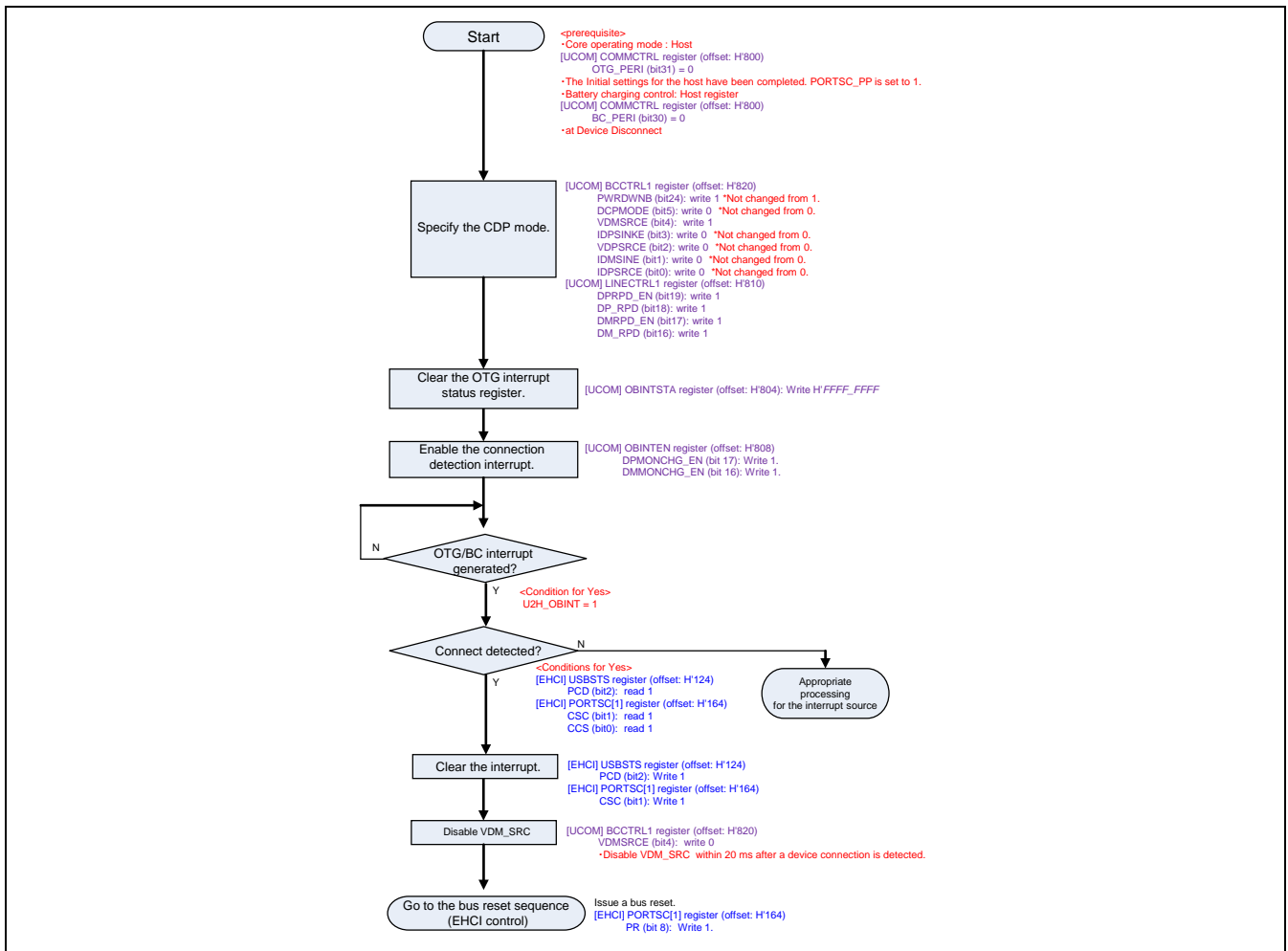


Figure 60.18 CDP Mode Control (UTMI+ PHY SUSPENDM is Asserted in CDP Mode)

(3) Portable Device Control

The following figures respectively show the procedures when a portable device is controlled through the register in this module (UCOM register) and when it is controlled through the peripheral controller (HS-USB module).

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Portable Device Control through UCOM Register:

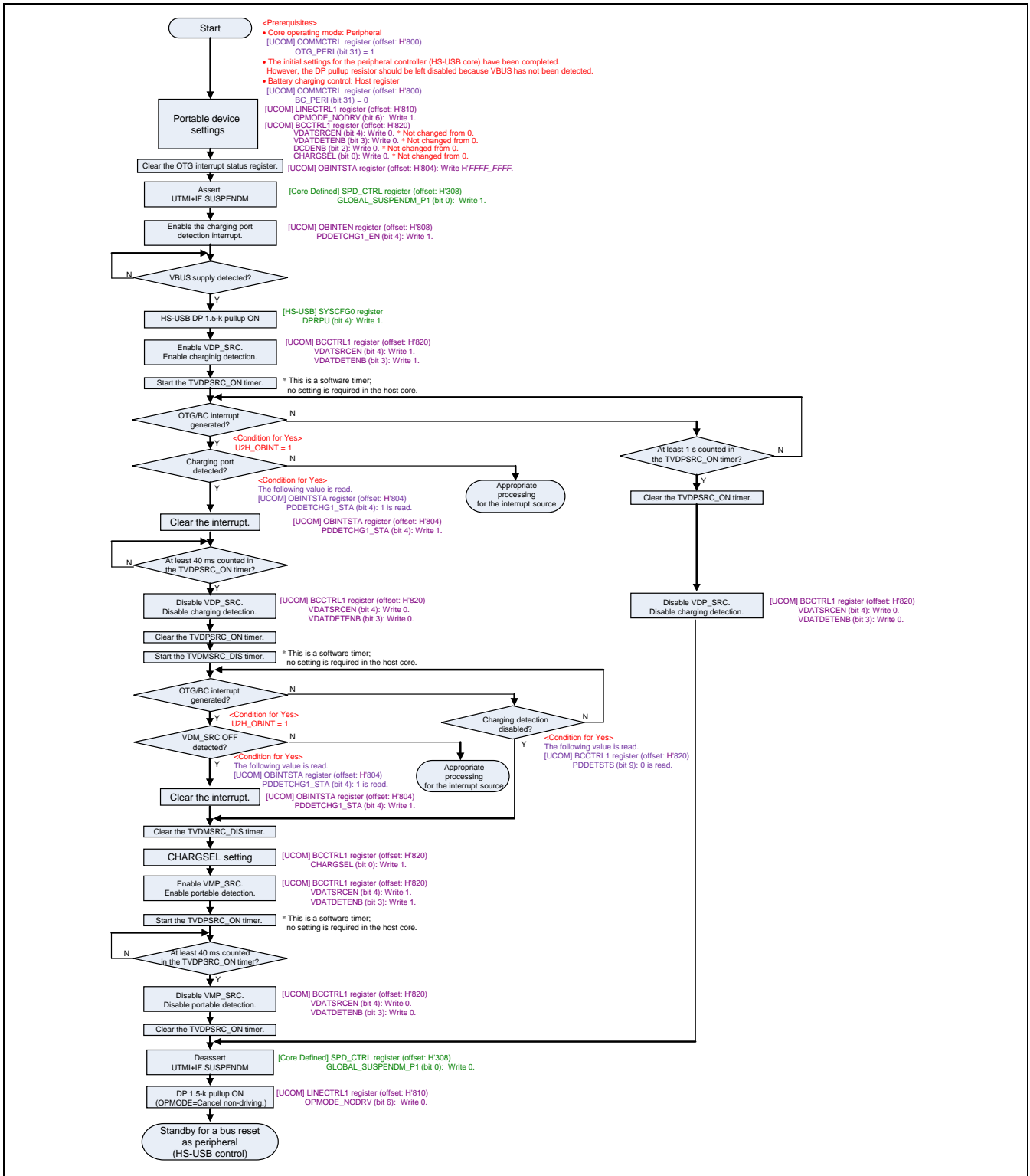


Figure 60.19 Portable Device Control through UCOM Register

Portable Device Control through Peripheral Controller (HS-USB module):

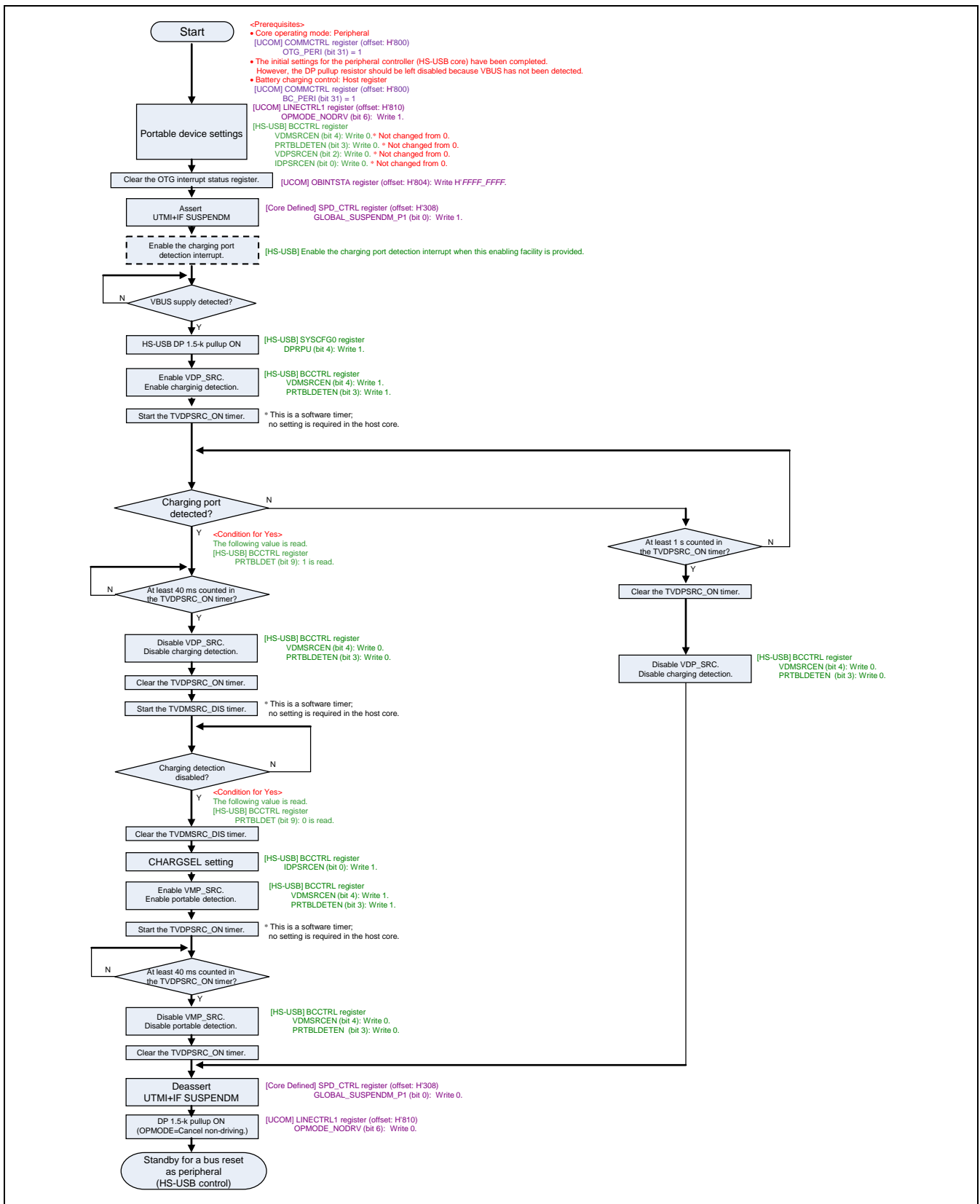


Figure 60.20 Portable Device Control through Peripheral Controller (HS-USB module)

[RZ/G2E]

Portable Device Control through UCOM Register:

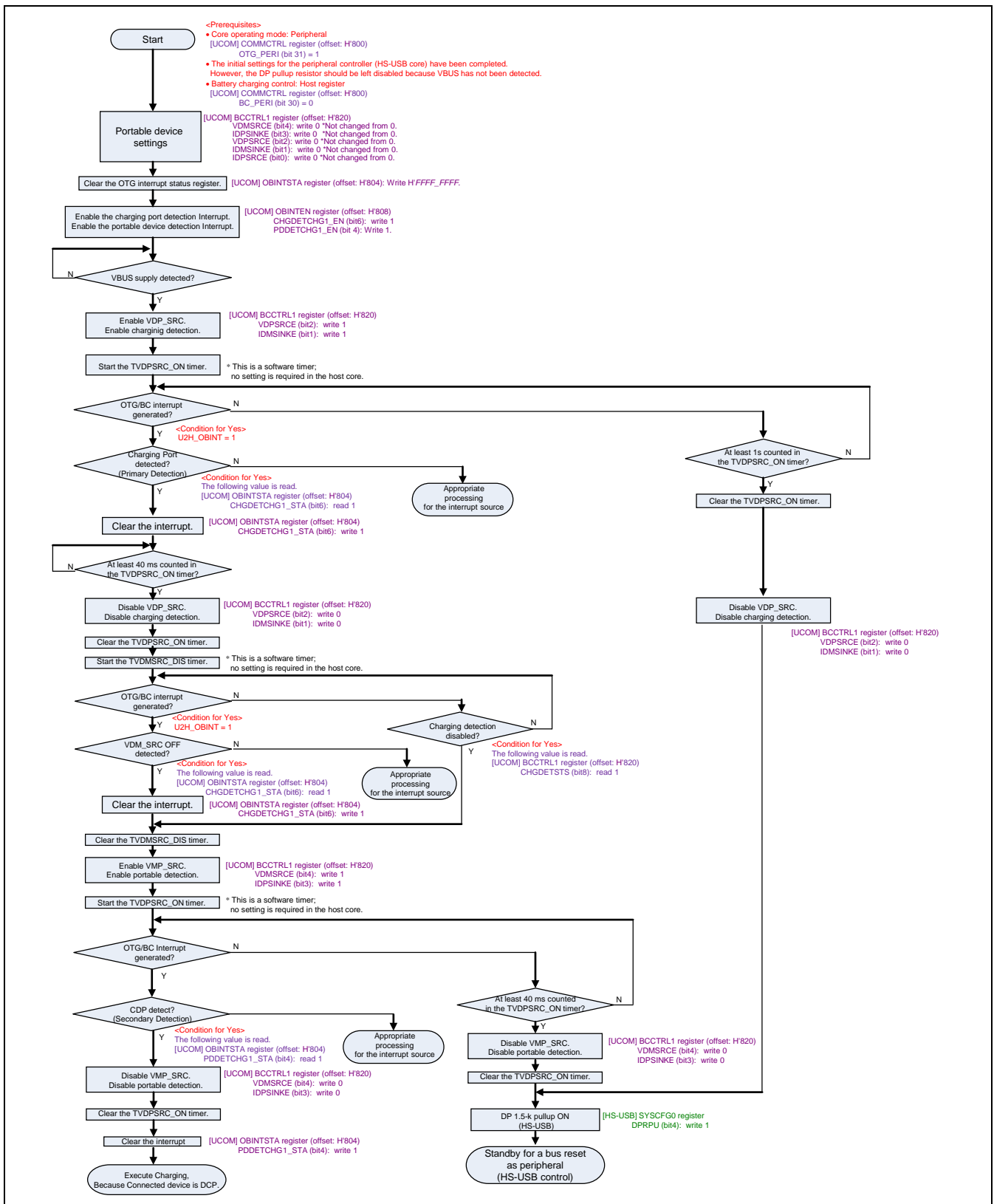


Figure 60.21 Portable Device Control through UCOM Register

Portable Device Control through Peripheral Controller (HS-USB module):

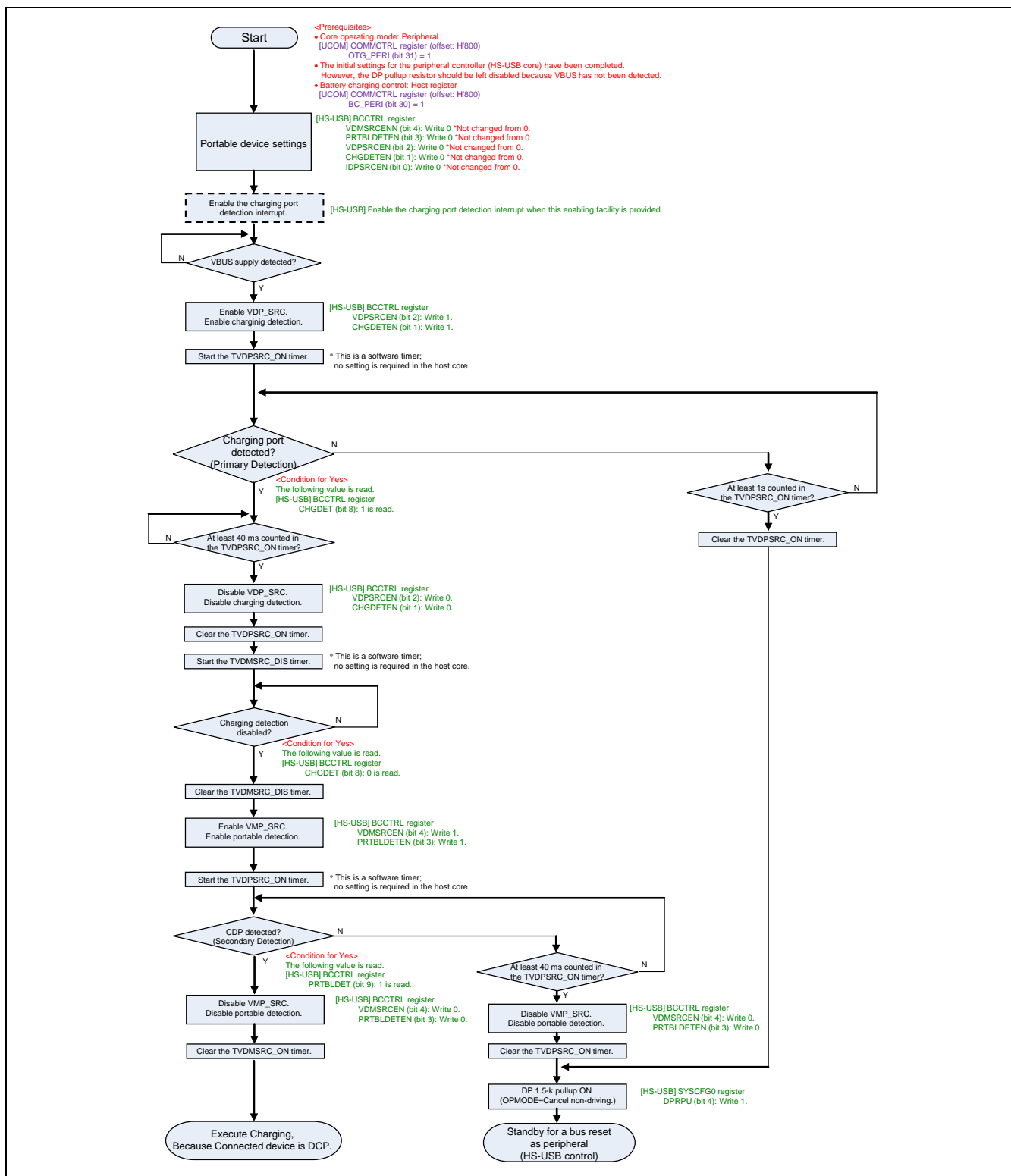


Figure 60.22 Portable Device Control through Peripheral Controller (HS-USB module)

60.9 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

This section describes register operations required for the initial settings of this module and the registers required for the power consumption control.

60.9.1 Power control and Initialization

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

1. EHCI/OHCI and HS-USB module stop release and module reset assert.
2. USB0SEL[1:0] in UGCTRL2 register of HS-USB should be set to B'01 for EHCI/OHCI (Host function). In case of OTG function, it should be set to B'11.
3. Starting normal operation.

[RZ/G2E]

1. EHCI/OHCI and HS-USB module stop release and module reset assert.
2. PLLRESET release.
3. USB0SEL[1:0] in UGCTRL2 register of HS-USB should be set to B'01 for EHCI/OHCI (Host function).
4. LPSTS.SUSPM set to normal mode.
5. Check PLL Lock status by UGSTS.PLLRESET.
6. Starting normal operation after PLL Lock status enabled.
7. UGCTRL.CONNECT bit set to 1

60.9.2 Transfer Overview

Transfer by the host controller must be performed in compliance with the following OHCI and EHCI specifications.

Open Host Controller Interface Specification for USB Rev 1.0a

Enhanced Host Controller Interface Specification for Universal Serial Bus Revision 1.0

61. USB High-Speed Module (HS-USB)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

61.1 Overview

This module is a USB controller provided with the function controller function, and high-speed transfer and full-speed transfer are available. Provided with an on-chip USB transceiver, this module also supports all transfer types defined by the USB Specification.

This module can use up to twelve pipes when the function controller is selected. Furthermore, for pipes 1 to F, arbitrary endpoint numbers can be allocated according to peripheral devices to be communicated and user systems.

61.1.1 Features

(1) On-chip USB transceiver

(2) Achieves space-saving mount with less external elements

- On-chip D+ pull-up resistor (function operation mode)
- On-chip D+/D- terminating resistors (high-speed operation mode)
- On-chip D+/D- output resistors (full-speed operation mode)

(3) All Types of USB Transfers Supported

- Control transfer
- Bulk transfer
- Interrupt transfer (except for high bandwidth)
- Isochronous transfer (except for high bandwidth)

(4) Dedicated DMA interface

- On-chip 4-channel DMA interface

(5) Pipe configuration

- Up to sixteen pipes are selectable (including the default control pipe)
- Programmable pipe configuration
- Arbitrary endpoint numbers can be allocated to pipes 1 to F.
- Settable transfer conditions for each pipe are as follows:

Pipe 0: A pipe only for control transfer with a 64-byte (fixed) single buffer

Pipes 1 and 2: Bulk transfers/isochronous transfer, continuous transfer mode, programmable buffer size (up to 2-Kbytes: double buffer can be specified)

Pipes 3 to 5 and B to F: Bulk transfer, continuous transfer mode, programmable buffer size (up to 2-Kbytes: double buffer can be specified)

Pipes 6 to 8: Interrupt transfer, 64-byte fixed single buffer

Pipes 9: Bulk transfer, programmable buffer size (up to 512-bytes: double buffer can be specified)

Pipes A: Bulk transfer, continuous transfer mode, programmable buffer size (up to 64-bytes: double buffer can be specified)/Interrupt transfer, 64-byte fixed single buffer

(6) Other functions

- Transfer ending function using transaction count
- DMA transfer ending function using external triggers (TEND or WREND signal)
- SOF pulse output function
- BRDY interrupt event notification timing change function (BFRE)
- Function that automatically clears the buffer memory after the data for the pipe specified at the DnFIFO (n = 0 to 3) port has been read (DCLRM)
- NAK setting function (SHTNAK) for the response PID due to the transfer end
- The suspend and resume of whole LSI can be done by using USBHS.

(7) Embedded USB PHY

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

The embedded USB PHY consists of the following features.

- Contains the USB functions drivers and receivers for D+ and D- signaling
- Supports battery charger detection (Battery Charging Specification revision 1.2)
- Supports On-The-Go function (On-The-Go Specification revision 2.0)

This USB subsystem support external device interface for external device control. These signals inform the vbus status, vbus valid and the OTG SRP, HNP

[RZ/G2E]

The embedded USB PHY consists of the following features.

- Contains the USB functions drivers and receivers for D+ and D- signaling
- Supports battery charger detection (Battery Charging Specification revision 1.2)

The embedded USB PHY does not support ID pin detection and VBUS detection. This function is exported to the external device (for example, MAX3355E). This product support only OVC signals for vbus detection.

This USB subsystem support external device interface for external device control. These signals inform the vbus status, vbus valid and the OTG SRP

61.1.2 External Pins

Table 61.1 lists the input and output pins of the USB.

Table 61.1 Pin Configuration of USB

| Name | Pin Name | I/O | Description | Second Generation RZ/G Series Products | | | |
|--|--|-----|---|---|----------------------------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Clock input for USB | USB_EXTAL | I | Used as an external clock input pin(50MHz) | √ | √ | √ | — |
| | USB_XTAL | O | Outputs amplified negative feedback of EXTAL (50MHz) | √ | √ | √ | — |
| USB D+ data | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] DPn/USB2_CH3_DP [RZ/G2E] USB0_DP0 | I/O | D+ Input/output of the on-chip transceiver Connect this pin to the D+ pin of the USB bus. | √ | √ | √ | √ |
| USB D- data | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] DMn/USB2_CH3_DM [RZ/G2E] USB0_DM0 | I/O | D- Input/output of the on-chip transceiver Connect this pin to the D- pin of the USB bus. | √ | √ | √ | √ |
| USB Identification | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] IDn/USB2_CH3_ID [RZ/G2E] USB0_ID | I/O | Connect to the pin on the Micro connectors that is used to differentiate a Micro-A plug from a Micro-B plug. | √ | √ | √ | √ |
| USB VBUS | VBUSn | I/O | Connect to USB VBUS with external 30kΩ±1% series resistor. | √ | √ | √ | — |
| USB external reference resistor | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] TXRTUNEn [RZ/G2E] USB0_RREF | I/O | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Connect to external 200Ω±1% resistor. [RZ/G2E] Connect to external 1.8kΩ±1% resistor. | √ | √ | √ | √ |

- Notes: 1. The VBUS of USB connector must not connect directly to VBUSn pin.
 2. VBUSn pin must be isolated by an external 30kΩ±1% series resistor.
 3. Suffix "n" of pin names is 0 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E.

61.1.3 Register Configuration

Table 61.2 HS-USB Base Address

| HS-USB Base Address | Base Address | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|---------------------|--------------|--------|----------------------------|--------|--------|
| Channel 0 | H'E659_0000 | √ | √ | √ | √ |

Table 61.3 lists the registers used in this module. Table 61.4 lists the register states in each processing mode.

Table 61.3 Register Configuration

| Register Name | Abbreviation | R/W | Address Offset | Access Size | Second Generation RZ/G Series Products | | | |
|---------------------------------------|--------------|-----|----------------|-------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| System configuration control register | SYSCFG | R/W | H'000 | 16 | √ | √ | √ | √ |
| CPU bus wait register | BUSWAIT | R/W | H'002 | 16 | √ | √ | √ | √ |
| System configuration status register | SYSSTS | R | H'004 | 16 | √ | √ | √ | √ |
| Device state control register | DVSTCTR | R/W | H'008 | 16 | √ | √ | √ | √ |
| Test mode register | TESTMODE | R/W | H'00C | 16 | √ | √ | √ | √ |
| CFIFO port register | CFIFO | R/W | H'014 | 32 | √ | √ | √ | √ |
| CFIFO port select register | CFIFOSEL | R/W | H'020 | 16 | √ | √ | √ | √ |
| CFIFO port control register | CFIFOCTR | R/W | H'022 | 16 | √ | √ | √ | √ |
| D0FIFO port select register | D0FIFOSEL | R/W | H'028 | 16 | √ | √ | √ | √ |
| D0FIFO port control register | D0FIFOCTR | R/W | H'02A | 16 | √ | √ | √ | √ |
| D1FIFO port select register | D1FIFOSEL | R/W | H'02C | 16 | √ | √ | √ | √ |
| D1FIFO port control register | D1FIFOCTR | R/W | H'02E | 16 | √ | √ | √ | √ |
| Interrupt enable register 0 | INTENB0 | R/W | H'030 | 16 | √ | √ | √ | √ |
| BRDY interrupt enable register | BRDYENB | R/W | H'036 | 16 | √ | √ | √ | √ |
| NRDY interrupt enable register | NRDYENB | R/W | H'038 | 16 | √ | √ | √ | √ |
| BEMP interrupt enable register | BEMPENB | R/W | H'03A | 16 | √ | √ | √ | √ |
| SOF output configuration register | SOFCFG | R/W | H'03C | 16 | √ | √ | √ | √ |
| Interrupt status register 0 | INTSTS0 | R/W | H'040 | 16 | √ | √ | √ | √ |
| BRDY interrupt status register | BRDYSTS | R/W | H'046 | 16 | √ | √ | √ | √ |
| NRDY interrupt status register | NRDYSTS | R/W | H'048 | 16 | √ | √ | √ | √ |
| BEMP interrupt status register | BEMPSTS | R/W | H'04A | 16 | √ | √ | √ | √ |
| Frame number register | FRMNUM | R/W | H'04C | 16 | √ | √ | √ | √ |
| μ frame number register | UFRMNUM | R/W | H'04E | 16 | √ | √ | √ | √ |
| USB address register | USBADDR | R | H'050 | 16 | √ | √ | √ | √ |

| Register Name | Abbreviation | R/W | Address Offset | Access Size | Second Generation RZ/G Series Products | | | |
|--|--------------|-----|-------------------|----------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| USB request type register | USBREQ | R | H'054 | 16 | √ | √ | √ | √ |
| USB request value register | USBVAL | R | H'056 | 16 | √ | √ | √ | √ |
| USB request index register | USBINDX | R | H'058 | 16 | √ | √ | √ | √ |
| USB request length register | USBLENG | R | H'05A | 16 | √ | √ | √ | √ |
| DCP maximum packet size register | DCPMAXP | R/W | H'05E | 16 | √ | √ | √ | √ |
| DCP control register | DCPCTR | R/W | H'060 | 16 | √ | √ | √ | √ |
| Pipe window select register | PIPESEL | R/W | H'064 | 16 | √ | √ | √ | √ |
| Pipe configuration register | PIPECFG | R/W | H'068 | 16 | √ | √ | √ | √ |
| Pipe buffer register | PIPEBUF | R/W | H'06A | 16 | √ | √ | √ | √ |
| Pipe maximum packet size register | PIPEMAXP | R/W | H'06C | 16 | √ | √ | √ | √ |
| Pipe cycle control register | PIPEPERI | R/W | H'06E | 16 | √ | √ | √ | √ |
| Pipe 1 control register | PIPE1CTR | R/W | H'070 | 16 | √ | √ | √ | √ |
| Pipe 2 control register | PIPE2CTR | R/W | H'072 | 16 | √ | √ | √ | √ |
| Pipe 3 control register | PIPE3CTR | R/W | H'074 | 16 | √ | √ | √ | √ |
| Pipe 4 control register | PIPE4CTR | R/W | H'076 | 16 | √ | √ | √ | √ |
| Pipe 5 control register | PIPE5CTR | R/W | H'078 | 16 | √ | √ | √ | √ |
| Pipe 6 control register | PIPE6CTR | R/W | H'07A | 16 | √ | √ | √ | √ |
| Pipe 7 control register | PIPE7CTR | R/W | H'07C | 16 | √ | √ | √ | √ |
| Pipe 8 control register | PIPE8CTR | R/W | H'07E | 16 | √ | √ | √ | √ |
| Pipe 9 control register | PIPE9CTR | R/W | H'080 | 16 | √ | √ | √ | √ |
| Pipe A control register | PIPEACTR | R/W | H'082 | 16 | √ | √ | √ | √ |
| Pipe B control register | PIPEBCTR | R/W | H'084 | 16 | √ | √ | √ | √ |
| Pipe C control register | PIPECCTR | R/W | H'086 | 16 | √ | √ | √ | √ |
| Pipe D control register | PIPEDCTR | R/W | H'088 | 16 | √ | √ | √ | √ |
| Pipe E control register | PIPEECTR | R/W | H'08A | 16 | √ | √ | √ | √ |
| Pipe F control register | PIPEFCTR | R/W | H'08C | 16 | √ | √ | √ | √ |
| Pipe 1 transaction counter enable register | PIPE1TRE | R/W | H'090 | 16 | √ | √ | √ | √ |
| Pipe 1 transaction counter register | PIPE1TRN | R/W | H'092 | 16 | √ | √ | √ | √ |
| Pipe 2 transaction counter enable register | PIPE2TRE | R/W | H'094 | 16 | √ | √ | √ | √ |
| Pipe 2 transaction counter register | PIPE2TRN | R/W | H'096 | 16 | √ | √ | √ | √ |
| Pipe 3 transaction counter enable register | PIPE3TRE | R/W | H'098 | 16 | √ | √ | √ | √ |
| Pipe 3 transaction counter register | PIPE3TRN | R/W | H'09A | 16 | √ | √ | √ | √ |

| Register Name | Abbreviation | R/W | Address Offset | Access Size | Second Generation RZ/G Series Products | | | |
|--|--------------|-----|----------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Pipe 4 transaction counter enable register | PIPE4TRE | R/W | H'09C | 16 | √ | √ | √ | √ |
| Pipe 4 transaction counter register | PIPE4TRN | R/W | H'09E | 16 | √ | √ | √ | √ |
| Pipe 5 transaction counter enable register | PIPE5TRE | R/W | H'0A0 | 16 | √ | √ | √ | √ |
| Pipe 5 transaction counter register | PIPE5TRN | R/W | H'0A2 | 16 | √ | √ | √ | √ |
| Pipe B transaction counter enable register | PIPEBTRE | R/W | H'0A4 | 16 | √ | √ | √ | √ |
| Pipe B transaction counter register | PIPEBTRN | R/W | H'0A6 | 16 | √ | √ | √ | √ |
| Pipe C transaction counter enable register | PIPECTRE | R/W | H'0A8 | 16 | √ | √ | √ | √ |
| Pipe C transaction counter register | PIPECTRN | R/W | H'0AA | 16 | √ | √ | √ | √ |
| Pipe D transaction counter enable register | PIPEDTRE | R/W | H'0AC | 16 | √ | √ | √ | √ |
| Pipe D transaction counter register | PIPEDTRN | R/W | H'0AE | 16 | √ | √ | √ | √ |
| Pipe E transaction counter enable register | PIPEETRE | R/W | H'0B0 | 16 | √ | √ | √ | √ |
| Pipe E transaction counter register | PIPEETRN | R/W | H'0B2 | 16 | √ | √ | √ | √ |
| Pipe F transaction counter enable register | PIPEFTRE | R/W | H'0B4 | 16 | √ | √ | √ | √ |
| Pipe F transaction counter register | PIPEFTRN | R/W | H'0B6 | 16 | √ | √ | √ | √ |
| Pipe 9 transaction counter enable register | PIPE9TRE | R/W | H'0B8 | 16 | √ | √ | √ | √ |
| Pipe 9 transaction counter register | PIPE9TRN | R/W | H'0BA | 16 | √ | √ | √ | √ |
| Pipe A transaction counter enable register | PIPEATRE | R/W | H'0BC | 16 | √ | √ | √ | √ |
| Pipe A transaction counter register | PIPEATRAN | R/W | H'0BE | 16 | √ | √ | √ | √ |
| D2FIFO port select register | D2FIFOSEL | R/W | H'0F0 | 16 | √ | √ | √ | √ |
| D2FIFO port control register | D2FIFOCTR | R/W | H'0F2 | 16 | √ | √ | √ | √ |
| D3FIFO port select register | D3FIFOSEL | R/W | H'0F4 | 16 | √ | √ | √ | √ |
| D3FIFO port control register | D3FIFOCTR | R/W | H'0F6 | 16 | √ | √ | √ | √ |
| Low power status register | LPSTS | R/W | H'102 | 16 | √ | √ | √ | √ |
| Battery charging control register | BCCTRL | R/W | H'140 | 16 | √ | √ | √ | √ |
| USB general control register | UGCTRL | R/W | H'180 | 32 | — | — | — | √ |

| Register Name | Abbreviation | R/W | Address Offset | Access Size | Second Generation RZ/G Series Products | | | |
|------------------------------------|-------------------|-----|-------------------|----------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| USB general control register 2 | UGCTRL2 | R/W | H'184 | 32 | √ | √ | √ | √ |
| USB General Status Register | UGSTS | R | H'188 | 32 | — | — | — | √ |
| USB XTAL/EXTTAL control register * | USB20_CLKS ETO | R/W | H'630 | 16 | √ | √ | √ | — |

Note: * USB XTAL/EXTTAL control register is only in Channel 0.

Table 61.4 Register States in Each Processing Mode

| Abbreviation | Power-On Reset | Module Standby |
|--------------|----------------|----------------|
| SYSCFG | Initialized | Retained |
| BUSWAIT | Initialized | Retained |
| SYSSTS | Initialized | Retained |
| DVSTCTR | Initialized | Retained |
| TESTMODE | Initialized | Retained |
| CFIFO | Initialized | Retained |
| CFIFOSEL | Initialized | Retained |
| CFIFOCTR | Initialized | Retained |
| D0FIFOSEL | Initialized | Retained |
| D0FIFOCTR | Initialized | Retained |
| D1FIFOSEL | Initialized | Retained |
| D1FIFOCTR | Initialized | Retained |
| INTENB0 | Initialized | Retained |
| BRDYENB | Initialized | Retained |
| NRDYENB | Initialized | Retained |
| BEMPENB | Initialized | Retained |
| SOFCFG | Initialized | Retained |
| INTSTS0 | Initialized | Retained |
| BRDYSTS | Initialized | Retained |
| NRDYSTS | Initialized | Retained |
| BEMPSTS | Initialized | Retained |
| FRMNUM | Initialized | Retained |
| UFRMNUM | Initialized | Retained |
| USBADDR | Initialized | Retained |
| USBREQ | Initialized | Retained |
| USBVAL | Initialized | Retained |
| USBINDX | Initialized | Retained |
| USBLENG | Initialized | Retained |
| DCPMAXP | Initialized | Retained |
| DCPCTR | Initialized | Retained |
| PIPESEL | Initialized | Retained |

| Abbreviation | Power-On Reset | Module Standby |
|---------------------|-----------------------|-----------------------|
| PIPECFG | Initialized | Retained |
| PIPEBUF | Initialized | Retained |
| PIPEMAXP | Initialized | Retained |
| PIPEPERI | Initialized | Retained |
| PIPE1CTR | Initialized | Retained |
| PIPE2CTR | Initialized | Retained |
| PIPE3CTR | Initialized | Retained |
| PIPE4CTR | Initialized | Retained |
| PIPE5CTR | Initialized | Retained |
| PIPE6CTR | Initialized | Retained |
| PIPE7CTR | Initialized | Retained |
| PIPE8CTR | Initialized | Retained |
| PIPE9CTR | Initialized | Retained |
| PIPEACTR | Initialized | Retained |
| PIPEBCTR | Initialized | Retained |
| PIPECCTR | Initialized | Retained |
| PIPEDCTR | Initialized | Retained |
| PIPEECTR | Initialized | Retained |
| PIPEFCTR | Initialized | Retained |
| PIPE1TRE | Initialized | Retained |
| PIPE1TRN | Initialized | Retained |
| PIPE2TRE | Initialized | Retained |
| PIPE2TRN | Initialized | Retained |
| PIPE3TRE | Initialized | Retained |
| PIPE3TRN | Initialized | Retained |
| PIPE4TRE | Initialized | Retained |
| PIPE4TRN | Initialized | Retained |
| PIPE5TRE | Initialized | Retained |
| PIPE5TRN | Initialized | Retained |
| PIPEBTRE | Initialized | Retained |
| PIPEBTRN | Initialized | Retained |
| PIPECTRE | Initialized | Retained |
| PIPECTRN | Initialized | Retained |
| PIPEDTRE | Initialized | Retained |
| PIPEDTRN | Initialized | Retained |
| PIPEETRE | Initialized | Retained |
| PIPEETRN | Initialized | Retained |
| PIPEFTRE | Initialized | Retained |
| PIPEFTRN | Initialized | Retained |
| PIPE9TRE | Initialized | Retained |
| PIPE9TRN | Initialized | Retained |
| PIPEATRE | Initialized | Retained |
| PIPEATRN | Initialized | Retained |
| D2FIFOSEL | Initialized | Retained |

| Abbreviation | Power-On Reset | Module Standby |
|-----------------------------|----------------|----------------|
| D2FIFOCTR | Initialized | Retained |
| D3FIFOSEL | Initialized | Retained |
| D3FIFOCTR | Initialized | Retained |
| LPSTS | Initialized | Retained |
| BCCTRL | Initialized | Retained |
| UGCTRL* ¹ | Initialized | Retained |
| UGCTRL2 | Initialized | Retained |
| UGSTS* ¹ | Initialized | Retained |
| USB20_CLKSET0* ² | Initialized | Retained |

Notes: 1. UGCTRL and UGSTS registers are implemented RZ/G2E

2. USB20_CLKSET0 is only in Channel 0

61.1.4 Connected Module

Table 61.5 The connected modules to the HS-USB

| Module name | Connected module name | Function of connected module |
|-------------|--------------------------------|--|
| HS-USB | AP-System Core | Access the Register |
| | CPG | Output Clocks |
| | PFC | Select External pins |
| | Module Standby, Software Reset | Control to stop clocks, Execute software reset |
| | INTC, INTC-AP | Control to interrupt |
| | EHCI/OHCI | USB2.0 Host (EHCI/OHCI) |

61.2 Register Description

[Legend for the Register Description]

A bit assignment figure is shown for each register. The initial value and R/W attribute are indicated for each bit.

Bit: Bit number or bit range

Bit Name: Bit name or field name

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

RW0C: Readable/writable. The bit is initialized when 0 is written. Writing 1 is ignored.

RW1C: Readable/writable. The bit is initialized when 1 is written. Writing 0 is ignored.

R: Read-only. The write value should always be 0. (If there is a direction on a read or write value in the Description column, set the bit as directed.)

61.2.1 System Configuration Control Register (SYSCFG)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SYSCFG enables or disables high-speed operation, controls the DPn/USB_DP pin, and enables or disables the operation of this module.

Note: "n" of pin names is 0 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|-----|---|---|-------|---|---|---|------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | HSE | — | — | DPRPU | — | — | — | USBE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 15 to 11 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 | — | B'0 | R | Reserved This bit is always read as 0. Writes have no effect. |
| 9, 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 7 | HSE | B'0 | R/W | High-Speed Operation Enable 0: High-speed operation is disabled. Full-speed operation only 1: High-speed operation is enabled. (This module detects the transmission rate.) When HSE is 0, this module performs full-speed operation. When HSE is 1, this module executes the reset handshake protocol, and according to the result, this module automatically performs high-speed or full-speed operation. Change this bit while the DPRPU bit is 0. |
| 6, 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | DPRPU | B'0 | R/W | D+ Line Resistance Control Enables or disables pull-up of the D+ line. 0: Pull-up is disabled. 1: Pull-up is enabled. When this bit is set to 1, this module pulls up the D+ line at 3.3V, and informs the USB host of ATTACH. Clearing this bit to 0 cancels the pull-up of the D+ line, and informs the USB host of DETACH. Set this bit to 1 before using this module. Set this bit to 0 when this module is not used. (when EHCI/OHCI Host function is used or USB is not used). |
| 3 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | USBE | B'0 | R/W | <p>USB Module Operation Enable</p> <p>Enables or disables the USB module operation.</p> <p>0: The USB module operation is disabled.</p> <p>1: The USB module operation is enabled.</p> <p>Table 61.6 list the registers and bit names initialized when this bit is cleared to 0.</p> |

Table 61.6 Registers Initialized by Writing 0 in the USBE Bit

| Register Name | Bit Name |
|---------------|-------------------------|
| SYSSTS | LNST |
| DVSTCTR | RHST |
| INTSTS0 | DVSQ |
| USBADDR | USBADDR |
| USBREQ | BRequest, bmRequestType |
| USBVAL | wValue |
| USBINDX | wIndex |
| USBLENG | wLength |

61.2.2 CPU Bus Wait Register (BUSWAIT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

BUSWAIT specifies the number of access waits from the CPU to this module.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|---|---|------------|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | BWAIT[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | BWAIT[3:0] | B'1111 | R/W | <p>CPU Bus Wait</p> <p>Specify the number of access waits to this module.</p> <p>0000: 0 waits (2 access cycles)</p> <p>0001: 1 wait (3 access cycles)</p> <p>0010 to 1110: 2 to 14 waits (4 to 16 access cycles)</p> <p>1111: 15 waits (17 access cycles) initial value</p> <p>The following restriction is provided for access cycles to the registers of addresses beginning with H'04 of this controller.</p> <p>Restriction for wait: The cycle of continuous accesses to the registers of this controller must be 80 ns or more.</p> <p>To satisfy this restriction, control waits using the frequency of system clock S3D2φ. Choose the best value within the initial value of 17 clock cycles (maximum).</p> <p>This setting is the same as the waits in accesses to the FIFO port register. The maximum speed of accesses to the FIFO port is as follows:</p> <p>MBW = B'10 (32-bit width): Max. 48 Mbytes/s</p> <p>MBW = B'01 (16-bit width): Max. 24 Mbytes/s</p> <p>MBW = B'00 (8-bit width) : Max. 12 Mbytes/s</p> |

61.2.3 System Configuration Status Register (SYSSTS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SYSSTS monitors the USB data bus line status (D+ and D- lines).

This register is initialized by a power-on reset or a USB reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|-----------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | LNST[1:0] |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | — | — |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 15 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1, 0 | LNST[1:0] | Undefined* | R | USB Data Line Status Monitor Indicate the USB data bus line (D+ and D- lines) status. Table 61.7 shows the USB data bus line status. Read these bits after the attach processing (DPRPU = 1). |

Note: * These bits depend on the state of the DPn/USB_DP and DMn/USB_DM pins.
"n" of pin name is 0 for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E

Table 61.7 USB Data Bus Line Status

| LNST[1] | LNST[0] | Full-Speed Operation | High-Speed Operation | Chirp Operation |
|---------|---------|----------------------|----------------------|-----------------|
| 0 | 0 | SE0 | Squelch | Squelch |
| 0 | 1 | J-State | UnSquelch | Chirp J |
| 1 | 0 | K-State | Invalid | Chirp K |
| 1 | 1 | SE1 | Invalid | Invalid |

Legend:

| | |
|------------|---|
| Chirp: | The reset handshake protocol (RHSP) is being executed with high-speed operation is enabled (HSE = 1 in SYSCFG). |
| Squelch: | SE0 or idle state |
| UnSquelch: | High-speed J-State or high-speed K-State |
| Chirp J: | Chirp J-State |
| Chirp K: | Chirp K-State |
| Invalid: | Invalid |

61.2.4 Device State Control Register (DVSTCTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DVSTCTR controls and monitors the USB data bus state.

This register is initialized by a power-on reset. The WKUP bit is initialized and the RESUME bit becomes undefined by a USB bus reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|------|---|---|---|---|---|-----------|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | WKUP | — | — | — | — | — | RHST[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 15 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | WKUP | B'0 | R/W | <p>Wakeup Output</p> <p>Enables or disables the remote wakeup (resume signal output) to the USB bus.</p> <p>0: The remote wakeup signal is not output. 1: The remote wakeup signal is output.</p> <p>This module controls the remote wakeup signal output time. When this bit is set to 1, this module outputs 10 ms K-State and then clears this bit. The USB Specification requires the USB bus idle state to be maintained for at least 5 ms before transmitting the remote wakeup signal. For this reason, this module outputs K-State after waiting for 2 ms even if 1 is written in this bit immediately after the suspended state is detected.</p> <p>Write 1 in this bit only when the device is in the suspended state (DVSQ = 1xx in INTSTS0) and remote wakeup is enabled by the USB host. When setting this bit to 1, do not stop the internal clock even in the suspended state (Write 1 in this bit with SUSPM set to 1).</p> |
| 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 to 0 | RHST[2:0] | B'000 | R | <p>Reset Handshake</p> <p>Indicate the reset handshake state.</p> <p>B'000: Transmission rate is undefined. B'100: The reset handshake processing is in progress. B'010: Full-speed connection B'011: High-speed connection</p> <p>When this module detects a USB bus reset with the HSE bit set to 1 for the port, these bits become B'100. After that, when this module outputs ChirpK and detects ChirpJK from the USB host three times, these bits become B'011. Unless high-speed mode is fixed within 2.5 ms after the ChirpK output, these bits become B'010.</p> <p>When this module detects a USB bus reset with the HSE bit set to 0 for the port, these bits become B'010.</p> <p>When the RHST bits are fixed to B'010 or B'011 after this module detected a USB bus reset, the DVST interrupt occurs.</p> |

61.2.5 Test Mode Register (TESTMODE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TESTMODE controls the USB test signal output in high-speed operation mode.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|---|---|-----------|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | UTST[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | UTST[3:0] | B'0000 | R/W | Test Mode Table 61.8 shows the test mode operation of this module. Control the USB test signal output in high-speed operation mode. These bits are valid only in high-speed operation mode. Use this test mode when the RHST bits in DVSTCTR are B'011. After the test with these bits, cancel this test mode by a power-on reset. Set these bits according to the SetFeature request from the USB host during high-speed communication. While these bits are B'0001 to B'0100, this module does not enter the suspended state. |

Table 61.8 Test Mode Operation

| Test Mode | UTST Bits Setting |
|-------------------|-------------------|
| Normal operation | B'0000 |
| Test_J | B'0001 |
| Test_K | B'0010 |
| Test_SE0_NAK | B'0011 |
| Test_Packet | B'0100 |
| Test_Force_Enable | — |
| Reserved | B'0101 to B'0111 |

61.2.6 CFIFO Port Register (CFIFO)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

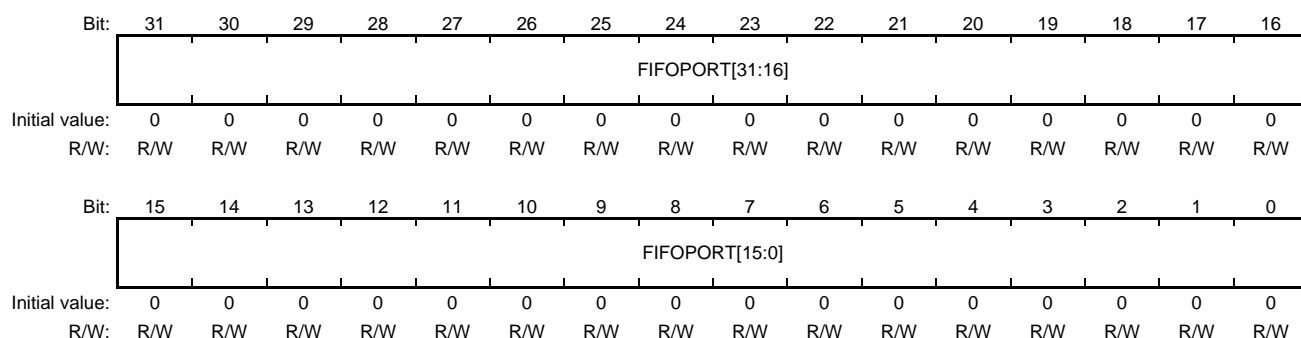
CFIFO is a port register to read data from or write data to the FIFO buffer memory.

Each FIFO port consists of this register (CFIFO) for data read/write from/to the FIFO buffer memory, the select registers (CFIFOSEL, D0FIFOSEL, D1FIFOSEL, D2FIFOSEL and D3FIFOSEL) to select pipes to be allocated to a FIFO port, and the control registers (CFIFOCTR, D0FIFOCTR, D1FIFOCTR, D2FIFOCTR and D3FIFOCTR).

Each FIFO port provides the following features.

- Make accesses to the FIFO buffer for DCP through the CFIFO port.
- Make accesses to the FIFO buffer with DMA transfers through the DMAC module for USB high-speed only.
- When using a function specific to FIFO ports (DMA transfer function, etc.), the pipe number (selected pipe) specified by the CURPIPE bits cannot be changed.
- Register groups of a FIFO port do not affect other FIFO ports.
- Do not allocate the same pipe number to different FIFO ports.
- There are two FIFO buffer states where the access mastership is on the CPU side and on the SIE side. When the FIFO buffer access mastership is on the SIE side, the CPU cannot make an access to the FIFO buffer.

This register is initialized by a power-on reset.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 31 to 0 | FIFOPORT [31:0] | H'0000_0000 | R/W | <p>FIFO Port</p> <p>Read receive data from the FIFO buffer or write transmit data to the FIFO buffer using these bits.</p> <p>Access to this register is enabled only when the FRDY bit in each control register (CFIFOCTR, D0FIFOCTR, D1FIFOCTR, D2FIFOCTR and D3FIFOCTR) is set to 1.</p> <p>Valid bits in this register vary with the value of the MBW bits. Table 61.9 shows the valid bits in this register.</p> |

Table 61.9 Operation of This Register when Accessed

| Access Size | Bits 31 to 24 | Bits 23 to 16 | Bits 15 to 8 | Bits 7 to 0 |
|-------------|-----------------------------------|---------------|---------------|---------------|
| 32 bits | N + 3 address | N + 2 address | N + 1 address | N + 0 address |
| 16 bits | Write: Invalid, Read: Prohibited* | | N + 1 address | N + 0 address |
| 8 bits | Write: Invalid, Read: Prohibited* | | | N + 0 address |

Note: * Reading an invalid register with word access or byte access is prohibited.

61.2.7 FIFO Port Select Registers (CFIFOSEL, D0FIFOSEL, D1FIFOSEL, D2FIFOSEL, D3FIFOSEL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CFIFOSEL, D0FIFOSEL, D1FIFOSEL, D2FIFOSEL and D3FIFOSEL select pipes to be allocated to FIFO ports and control accesses to each FIFO port.

Do not specify the same pipe number for the CURPIPE bits in CFIFOSEL, D0FIFOSEL, D1FIFOSEL, D2FIFOSEL and D3FIFOSEL. When the CURPIPE bits in D0FIFOSEL, D1FIFOSEL, D2FIFOSEL and D3FIFOSEL are set to B'000, no pipe is specified.

Do not change any pipe number when DMA transfer is enabled.

This register is initialized by a power-on reset.

(1) CFIFOSEL

| | | | | | | | | | | | | | | | | |
|----------------|------|-----|----|----|----------|-----|---|---|---|------|-----|--------------|-----|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RCNT | REW | — | — | MBW[1:0] | — | — | — | — | ISEL | — | CURPIPE[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R/W | R/W | R | R | R | R | R/W | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|------|---|
| 15 | RCNT | B'0 | R/W | Read Count Mode Specifies read mode of the DTLN bits in CFIFOCTR. 0: Clears the DTLN bits after reading all receive data in the CFIFO (or after reading receive data of one side of the double buffer). 1: Counts down the DTLN bits at each reading of CFIFO receive data. |
| 14 | REW | B'0 | R/W* | Buffer Pointer Rewind Specifies whether to rewind the buffer pointer or not. 0: The buffer pointer is not rewind. 1: The buffer pointer is rewind. When the selected pipe is in the receive direction and this bit is set to 1 during FIFO buffer reading, the FIFO buffer can be read from the first data. (In the case of a double buffer, the first data of one side that is being read can be read again.) Do not set this bit to 1 concurrently with the CURPIPE setting change. Be sure to confirm that the FRDY bit is 1 when setting this bit to 1. To rewrite the FIFO buffer data from the first data for a pipe in the transmit direction, use the BCLR bit. |
| 13, 12 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 11, 10 | MBW[1:0] | B'00 | R/W | <p>CFIFO Port Access Bit Width</p> <p>Specify the bit width for accesses to the CFIFO port.</p> <p>B'00: 8 bits B'01: 16 bits B'10: 32 bits B'11: Setting prohibited</p> <p>When the selected pipe is in the receive direction and data read is started after these bits are set, do not change these bits until all data is completely read.</p> <p>When the selected pipe is in the receive direction, set the CURPIPE and MBW bits at the same time.</p> <p>When the selected pipe is in the transmit direction, no bit width can be changed (from 8 bits to 16/32 bits or from 16 bits to 32 bits) during data write to the buffer memory.</p> <p>When 8-bit width or 16-bit width is selected, data of odd bytes can also be written by controlling the byte access.</p> |
| 9 to 6 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 5 | ISEL | B'0 | R/W | <p>CFIFO Port Access Direction when DCP is Selected</p> <p>0: Buffer memory read is selected. 1: Buffer memory write is selected.</p> <p>To change this bit when the selected pipe is DCP, write 1 or 0 in this bit, read the value of this bit, confirm that the write value equals the read value, and then proceed with the next processing.</p> <p>If this bit is changed in the middle of access to the FIFO buffer, accesses made before that time are retained, the value of this bit is rewritten, and then further access is enabled.</p> <p>Set this bit concurrently with the CURPIPE bits.</p> |
| 4 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------------|---------------|-----|---|
| 3 to 0 | CURPIPE [3:0] | B'0000 | R/W | <p>CFIFO Port Access Pipe Select</p> <p>Set the number of pipe to be used for data read or write through the CFIFO port.</p> <p>B'0000: DCP B'0001: Pipe 1 B'0010: Pipe 2 B'0011: Pipe 3 B'0100: Pipe 4 B'0101: Pipe 5 B'0110: Pipe 6 B'0111: Pipe 7 B'1000: Pipe 8 B'1001: Pipe 9 B'1010: Pipe A B'1011: Pipe B B'1100: Pipe C B'1101: Pipe D B'1110: Pipe E B'1111: Pipe F</p> <p>To change these bits, write a value in these bits, read the value of these bits, confirm that the write value equals the read value, and then proceed with the next processing.</p> <p>Do not write the same pipe number in the CURPIPE bits in CFIFOSEL, D0FIFOSEL, D1FIFOSEL, D2FIFOSEL and D3FIFOSEL.</p> <p>If these bits are changed in the middle of access to the FIFO buffer, accesses made before that time are retained, the value of these bits is rewritten, and then further access is enabled.</p> |

Note: * Only reading 0 is enabled.

(2) D0FIFOSEL, D1FIFOSEL, D2FIFOSEL, D3FIFOSEL

| | | | | | | | | | | | | | | | | |
|----------------|------|-----|-------|-------|----------|-----|---|-------|-----|---|---|--------------|-----|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | RCNT | REW | DCLRM | DREQE | MBW[1:0] | — | — | DEZPM | — | — | — | CURPIPE[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 15 | RCNT | B'0 | R/W | <p>Read Count Mode</p> <p>Specifies read mode of the DTLN bits in DnFIFOCTR.</p> <p>0: Clears the DTLN bits after reading all receive data in the DnFIFO (or after reading receive data of one side of the double buffer).</p> <p>1: Counts down the DTLN bits at each reading of DnFIFO receive data. When making an access to the DnFIFO with the BFRE bit set to 1, set this bit to 0.</p> |
| 14 | REW | B'0 | R/W* | <p>Buffer Pointer Rewind</p> <p>Specifies whether to rewind the buffer pointer or not.</p> <p>0: The buffer pointer is not rewound.</p> <p>1: The buffer pointer is rewound.</p> <p>When the selected pipe is in the receive direction and this bit is set to 1 during FIFO buffer reading, the FIFO buffer can be read from the first data. (In the case of a double buffer, the first data of one side that is being read can be read again.)</p> <p>Do not set this bit to 1 concurrently with the CURPIPE setting change. Be sure to confirm that the FRDY bit is 1 when setting this bit to 1.</p> <p>To rewrite the FIFO buffer data from the first data for a pipe in the transmit direction, use the BCLR bit.</p> |
| 13 | DCLRM | B'0 | R/W | <p>Buffer Memory Auto-Clear Mode after Reading Data of Selected Pipe</p> <p>Enables or disables buffer memory auto-clear mode after reading data of the selected pipe.</p> <p>0: Buffer memory auto-clear mode is disabled.</p> <p>1: Buffer memory auto-clear mode is enabled.</p> <p>When this bit is 1 and a Zero-Length packet is received while the FIFO buffer allocated to the selected pipe is empty or when a short packet is received with the BFRE bit set to 1 and the packet data is completely read, this module performs buffer clear (BCLR = 1 processing) of the FIFO buffer.</p> <p>When using this module with the BRDYM bit set to 1, be sure to set this bit to 0.</p> |
| 12 | DREQE | B'0 | R/W | <p>DMA Transfer Request Enable</p> <p>Enables or disables making a DMA transfer request.</p> <p>0: DMA transfer request is disabled.</p> <p>1: DMA transfer request is enabled.</p> <p>To enable a DMA transfer request, set the CURPIPE bits and then set this bit to 1.</p> <p>Set this bit to 0 before changing the CURPIPE setting.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 11, 10 | MBW[1:0] | B'00 | R/W | <p>FIFO Port Access Bit Width</p> <p>Specify the bit width for accesses to the DnFIFO port.</p> <p>B'00: 8 bits B'01: 16 bits B'10: 32 bits B'11: Setting prohibited</p> <p>When the selected pipe is in the receive direction and data read is started after these bits are set, do not change these bits until all data is completely read.</p> <p>When the selected pipe is in the receive direction, set the CURPIPE and MBW bits at the same time.</p> <p>When the selected pipe is in the transmit direction, no bit width can be changed (from 8 bits to 16/32 bits or from 16 bits to 32 bits) during data write to the buffer memory.</p> <p>When 8-bit width or 16-bit width is selected, data of odd bytes can also be written by controlling the byte access.</p> |
| 9, 8 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 7 | DEZPM | B'0 | R/W | <p>DEZPM: Zero-Length Packet Addition Mode</p> <p>If the DMA transfer size that is set for the USB-DMAC modules is an integral multiple of the maximum packet size, a null packet is added after all of the data has been sent.</p> <p>0: Disables addition of packets. 1: Enables addition of packets.</p> <p>This bit is applied to pipes used to write to the FIFO buffer.</p> <p>DEZPM must be set before starting DMA transfers.</p> <p>If the DEZPM bit is set after DMA transfers complete, no null packets are added.</p> <p>Note: This mode depends on the buffer size (PIPEBUF.BUFSIZE[4:0]).</p> <p>A null packet is added after transferring data up to buffer size which is defined by PIPEBUF.BUFSIZE[4:0].</p> <p>When transfer data size is over 2Kbytes, this mode must not be used because max size of buffer is 2Kbytes.</p> <p>In that case, add the null packet by SW with BVAL bit of FIFO Port Control Registers.</p> |
| 6 to 4 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------------|---------------|-----|---|
| 3 to 0 | CURPIPE [3:0] | B'0000 | R/W | <p>FIFO Port Access Pipe Select</p> <p>Set the number of pipe to be used for data read or write through the DnFIFO ports.</p> <p>B'0000: No pipe B'0001: Pipe 1 B'0010: Pipe 2 B'0011: Pipe 3 B'0100: Pipe 4 B'0101: Pipe 5 B'0110: Pipe 6 B'0111: Pipe 7 B'1000: Pipe 8 B'1001: Pipe 9 B'1010: Pipe A B'1011: Pipe B B'1100: Pipe C B'1101: Pipe D B'1110: Pipe E B'1111: Pipe F</p> <p>To change these bits, write a value in these bits, read the value of these bits, confirm that the write value equals the read value, and then proceed with the next processing.</p> <p>Do not write the same pipe number in the CURPIPE bits in CFIFOSEL, DnFIFOSEL (n = 0 to 3).</p> <p>If these bits are changed in the middle of access to the FIFO buffer, accesses made before that time are retained, the value of these bits is rewritten, and then further access is enabled.</p> |

Note: * Only reading 0 is enabled.

61.2.8 FIFO Port Control Registers (CFIFOCTR, D0FIFOCTR, D1FIFOCTR, D2FIFOCTR, D3FIFOCTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CFIFOCTR and D0FIFOCTR, D1FIFOCTR, D2FIFOCTR and D3FIFOCTR specify buffer memory write end and CPU buffer clear, as well as indicate whether the FIFO port is accessible or not. These registers correspond to each FIFO port.

These registers are initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|----|------------|----|---|---|---|---|---|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | BVAL | BCLR | FRDY | — | DTLN[11:0] | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 15 | BVAL | B'0 | R/W*1 | <p>Buffer Memory Enable Flag</p> <p>This flag is set to 1 upon completion of data write in the FIFO buffer of the CPU of the pipe selected by the CURPIPE bits.</p> <p>0: Invalid 1: Write end</p> <p>When the selected pipe is in the transmit direction, set this bit to 1 in the following cases. This module sets the FIFO buffer of the CPU to the SIE side to enable data transmission.</p> <p>To send a short packet, set this bit to 1 upon completion of data write.</p> <p>To send a Zero-Length packet, set this bit to 1 before writing data to the FIFO buffer.</p> <p>For pipes of continuous transfer mode, set this bit to 1 after writing data that is a positive integer multiple of MaxPacketSize and less than the BufferSize value.</p> <p>When data with the size specified by MaxPacketSize is written to a pipe of continuous transfer mode, this module writes 1 in this bit and sets the FIFO buffer of the CPU to the SIE side to enable data transmission.</p> <p>Set this bit to 1 when this module shows FRDY = 1.</p> <p>When the selected pipe is in the receive direction, do not set this bit to 1.</p> |
| 14 | BCLR | B'0 | R/W*2 | <p>CPU buffer clear</p> <p>Clears the CPU side FIFO buffer of the selected pipe.</p> <p>0: Invalid 1: Clears the CPU side FIFO buffer.</p> <p>When the FIFO buffer assigned to the selected pipe is set to a double buffer, and even when both sides of the buffer are readable, this module clears only one side of the FIFO buffer.</p> <p>When the selected pipe is DCP, this module clears the FIFO buffer when this bit is set to 1 irrespective of whether the FIFO buffer is on the CPU side or SIE side. When clearing the SIE side FIFO buffer, be sure to set the PID bits in DCP to NAK and then set the BCLR bit to 1.</p> <p>When the selected pipe is in the transmit direction, and if 1 is written in the BVAL and BCLR bits at the same time, this module clears the previously written data, and makes a Zero-Length packet transmittable.</p> <p>When the selected pipe is not DCP, write 1 in this bit while the FRDY flag indicates 1.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 13 | FRDY | B'0 | R | <p>FIFO Port Ready</p> <p>Indicates whether the CPU (DMAC) can access the FIFO port.</p> <p>0: The FIFO port is not accessible.</p> <p>1: The FIFO port is accessible.</p> <p>This module indicates FRDY = 1 in the following cases. However, since there is no data to read, no data can be read from the FIFO port. In these cases, set the BCLR bit to 1 to clear the FIFO buffer to make transmission/reception of the next data available.</p> <p>When a Zero-Length packet is received with the FIFO buffer assigned to the selected pipe empty</p> <p>When a short packet is received with BFRE set to 1 and the packet data is completely read</p> |
| 12 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 11 to 0 | DTLN[11:0] | H'000 | R | <p>Receive Data Length</p> <p>Indicate the receive data length.</p> <p>The value of these bits during FIFO buffer reading varies as follows depending on the setting of the RCNT bit.</p> <p>When RCNT = 0:</p> <p>This module indicates the receive data length with these bits until the CPU (DMAC) finishes reading one side of the FIFO buffer.</p> <p>When BFRE is 1, this module retains the receive data length until the BCLR bit is set to 1 after the FIFO buffer read is completed.</p> <p>When RCNT = 1:</p> <p>This module decrements the value of the DTLN bits in each read cycle (-1 when MBW = 0, and -2 when MBW = 1)</p> <p>When reading out from a single FIFO buffer is complete, DTLN = 0 for this module. However, the behavior is different when double-buffering has been specified. If reading out of data from one side is not finished before the other side is full of received data, when all data have been read out from the first side, the DTLN bits reflect the amount of data received in the other side.</p> <p>When reading these bits during FIFO buffer reading while the RCNT bit is 1, note that this module updates the value in these bits within 150 ns after each FIFO port read cycle.</p> |

- Notes: 1. Only writing 1 is enabled.
 2. Only reading 0 and writing 1 are enabled.

61.2.9 Interrupt Enable Register 0 (INTENB0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

INTENB0 specifies masking of each interrupt when selection for function controller. When this module detects an interrupt for which 1 is written in the corresponding bit in this register by the software, this module generates a USB interrupt.

When interrupt source detection conditions are satisfied irrespective of the register setting (interrupt enable/disable), this module sets the corresponding status bit in INTSTS0.

When the software changes an interrupt enable bit (0 to 1) whose interrupt source status bit in INTSTS0 is set to 1, this module generates a USB interrupt.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|-------|-------|-------|---|---|---|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | VBSE | RSME | SOFE | DVSE | CTRE | BEMPE | NRDYE | BRDYE | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 15 | VBSE | B'0 | R/W | VBUSn [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] /OVC signal status [RZ/G2E] Interrupt Enable Enables or disables USB interrupt outputs when the VBINT interrupt is detected. 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 14 | RSME | B'0 | R/W | Resume Interrupt Enable Enables or disables USB interrupt outputs when the RESM interrupt is detected. 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 13 | SOFE | B'0 | R/W | Frame Number Update Interrupt Enable Enables or disables USB interrupt outputs when the SOFR interrupt is detected. 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 12 | DVSE | B'0 | R/W | Device State Transition Interrupt Enable Enables or disables USB interrupt outputs when the DVST interrupt is detected. 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 11 | CTRE | B'0 | R/W | Control Transfer Stage Transition Interrupt Enable Enables or disables USB interrupt outputs when the CTRT interrupt is detected. 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 10 | BEMPE | B'0 | R/W | <p>Buffer Empty Interrupt Enable</p> <p>Enables or disables USB interrupt outputs when the BEMP interrupt is detected.</p> <p>0: Interrupt outputs are disabled.</p> <p>1: Interrupt outputs are enabled.</p> |
| 9 | NRDYE | B'0 | R/W | <p>Buffer Not Ready Response Interrupt Enable</p> <p>Enables or disables USB interrupt outputs when the NRDY interrupt is detected.</p> <p>0: Interrupt outputs are disabled.</p> <p>1: Interrupt outputs are enabled.</p> |
| 8 | BRDYE | B'0 | R/W | <p>Buffer Ready Interrupt Enable</p> <p>Enables or disables USB interrupt outputs when the BRDY interrupt is detected.</p> <p>0: Interrupt outputs are disabled.</p> <p>1: Interrupt outputs are enabled.</p> |
| 7 to 0 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

61.2.10 BRDY Interrupt Enable Register (BRDYENB)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

BRDYENB enables or disables the BRDY bit to be set to 1 when the BRDY interrupt of a pipe is detected.

When this module detects the BRDY interrupt of a pipe for which the software sets the corresponding bit in this register to 1, this module sets the PIPEBRDY bit of the pipe in BRDYSTS and also sets the BRDY bit in INTSTS0 and generates the BRDY interrupt.

When one or more PIPEBRDY bits in BRDYSTS are 1 and the software changes the corresponding interrupt enable bit(s) in this register from 0 to 1, this module generates the BRDY interrupt.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PIPEF BRDYE | PIPEE BRDYE | PIPED BRDYE | PIPEC BRDYE | PIPEB BRDYE | PIPEA BRDYE | PIPE9 BRDYE | PIPE8 BRDYE | PIPE7 BRDYE | PIPE6 BRDYE | PIPE5 BRDYE | PIPE4 BRDYE | PIPE3 BRDYE | PIPE2 BRDYE | PIPE1 BRDYE | PIPE0 BRDYE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------------|---------------|-----|---|
| 15 | PIPEF BRDYE | B'0 | R/W | Pipe F BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 14 | PIPEE BRDYE | B'0 | R/W | Pipe E BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 13 | PIPED BRDYE | B'0 | R/W | Pipe D BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 12 | PIPEC BRDYE | B'0 | R/W | Pipe C BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 11 | PIPEB BRDYE | B'0 | R/W | Pipe B BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 10 | PIPEA BRDYE | B'0 | R/W | Pipe A BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 9 | PIPE9 BRDYE | B'0 | R/W | Pipe 9 BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 8 | PIPE8 BRDYE | B'0 | R/W | Pipe 8 BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 7 | PIPE7 BRDYE | B'0 | R/W | Pipe 7 BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------------|---------------|-----|---|
| 6 | PIPE6 BRDYE | B'0 | R/W | Pipe 6 BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 5 | PIPE5 BRDYE | B'0 | R/W | Pipe 5 BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 4 | PIPE4 BRDYE | B'0 | R/W | Pipe 4 BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 3 | PIPE3 BRDYE | B'0 | R/W | Pipe 3 BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 2 | PIPE2 BRDYE | B'0 | R/W | Pipe 2 BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 1 | PIPE1 BRDYE | B'0 | R/W | Pipe 1 BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 0 | PIPE0 BRDYE | B'0 | R/W | Pipe 0 BRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

61.2.11 NRDY Interrupt Enable Register (NRDYENB)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

NRDYENB enables or disables the NRDY bit to be set to 1 when the NRDY interrupt of a pipe is detected.

When this module detects the NRDY interrupt of a pipe for which the software sets the corresponding bit in this register to 1, this module sets the PIPENRDY bit of the pipe in NRDYSTS and also sets the NRDY bit in INTSTS0 and generates an NRDY interrupt.

When one or more PIPENRDY bits in NRDYSTS are 1 and the software changes the corresponding interrupt enable bit(s) in this register from 0 to 1, this module generates an NRDY interrupt.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PIPEF NRDYE | PIPEE NRDYE | PIPED NRDYE | PIPEC NRDYE | PIPEB NRDYE | PIPEA NRDYE | PIPE9 NRDYE | PIPE8 NRDYE | PIPE7 NRDYE | PIPE6 NRDYE | PIPE5 NRDYE | PIPE4 NRDYE | PIPE3 NRDYE | PIPE2 NRDYE | PIPE1 NRDYE | PIPE0 NRDYE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------------|---------------|-----|---|
| 15 | PIPEF NRDYE | B'0 | R/W | Pipe F NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 14 | PIPEE NRDYE | B'0 | R/W | Pipe E NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 13 | PIPED NRDYE | B'0 | R/W | Pipe D NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 12 | PIPEC NRDYE | B'0 | R/W | Pipe C NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 11 | PIPEB NRDYE | B'0 | R/W | Pipe B NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 10 | PIPEA NRDYE | B'0 | R/W | Pipe A NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 9 | PIPE9 NRDYE | B'0 | R/W | Pipe 9 NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 8 | PIPE8 NRDYE | B'0 | R/W | Pipe 8 NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 7 | PIPE7 NRDYE | B'0 | R/W | Pipe 7 NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------------|---------------|-----|---|
| 6 | PIPE6 NRDYE | B'0 | R/W | Pipe 6 NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 5 | PIPE5 NRDYE | B'0 | R/W | Pipe 5 NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 4 | PIPE4 NRDYE | B'0 | R/W | Pipe 4 NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 3 | PIPE3 NRDYE | B'0 | R/W | Pipe 3 NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 2 | PIPE2 NRDYE | B'0 | R/W | Pipe 2 NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 1 | PIPE1 NRDYE | B'0 | R/W | Pipe 1 NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 0 | PIPE0 NRDYE | B'0 | R/W | Pipe 0 NRDY Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

61.2.12 BEMP Interrupt Enable Register (BEMPENB)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

BEMPENB enables or disables the BEMP bit to be set to 1 when the BEMP interrupt of a pipe is detected.

When this module detects the BEMP interrupt of a pipe for which the software sets the corresponding bit in this register to 1, this module sets the PIPEBEMP bit of the pipe in BEMPSTS and the BEMP bit in INTSTS0 and generates the BEMP interrupt.

When one or more PIPEBEMP bits in BEMPSTS are 1 and the software changes the corresponding interrupt enable bit(s) in this register from 0 to 1, this module generates the BEMP interrupt.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PIPEF BEMPE | PIPEE BEMPE | PIPED BEMPE | PIPEC BEMPE | PIPEB BEMPE | PIPEA BEMPE | PIPE9 BEMPE | PIPE8 BEMPE | PIPE7 BEMPE | PIPE6 BEMPE | PIPE5 BEMPE | PIPE4 BEMPE | PIPE3 BEMPE | PIPE2 BEMPE | PIPE1 BEMPE | PIPE0 BEMPE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------------|---------------|-----|---|
| 15 | PIPEF BEMPE | B'0 | R/W | Pipe F BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 14 | PIPEE BEMPE | B'0 | R/W | Pipe E BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 13 | PIPED BEMPE | B'0 | R/W | Pipe D BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 12 | PIPEC BEMPE | B'0 | R/W | Pipe C BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 11 | PIPEB BEMPE | B'0 | R/W | Pipe B BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 10 | PIPEA BEMPE | B'0 | R/W | Pipe A BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 9 | PIPE9 BEMPE | B'0 | R/W | Pipe 9 BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 8 | PIPE8 BEMPE | B'0 | R/W | Pipe 8 BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 7 | PIPE7 BEMPE | B'0 | R/W | Pipe 7 BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------------|---------------|-----|---|
| 6 | PIPE6 BEMPE | B'0 | R/W | Pipe 6 BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 5 | PIPE5 BEMPE | B'0 | R/W | Pipe 5 BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 4 | PIPE4 BEMPE | B'0 | R/W | Pipe 4 BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 3 | PIPE3 BEMPE | B'0 | R/W | Pipe 3 BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 2 | PIPE2 BEMPE | B'0 | R/W | Pipe 2 BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 1 | PIPE1 BEMPE | B'0 | R/W | Pipe 1 BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |
| 0 | PIPE0 BEMPE | B'0 | R/W | Pipe 0 BEMP Interrupt Enable 0: Interrupt outputs are disabled. 1: Interrupt outputs are enabled. |

61.2.13 SOF Output Configuration Register (SOFCFG)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SOFCFG specifies an effective period of transactions, the BRDY interrupt status clear timing, and others.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|-----------|---|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | BRDY M | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 15 to 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 | BRDYM | B'0 | R/W | Status Clear Timing of Each Pipe BRDY Interrupt Specifies the timing to clear the BRDY interrupt status of each pipe. 0: The software clears the status. 1: This module clears the status by reading or writing the FIFO buffer. |
| 5 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

61.2.14 Interrupt Status Register 0 (INTSTS0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

INTSTS0 can know the interrupt generation by referring to the register when the function controller is selected.

Permit interrupt by the status change that each bit of this register shows only when you select the function.

This register is initialized by a power-on reset. The DVSQ2 to DVSQ0 bits are also initialized by a USB bus reset.

| | | | | | | | | | | | | | | | | |
|----------------|-------|------|------|------|-------|------|------|------|-------|-----------|---|-----|-------|-----------|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | VBINT | RESM | SOFR | DVST | CTRTR | BEMP | NRDY | BRDY | VBSTS | DVSQ[2:0] | | | VALID | CTSQ[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0/1 | 0 | 0 | 0 | 0 | 0/1 | 0 | 0 | 0/1 | 0 | 0 | 0 | 0 |
| R/W: | RW0C | RW0C | RW0C | RW0C | RW0C | R | R | R | R | R | R | R | RW0C | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|---|
| 15 | VBINT | B'0 | RW0C | VBUSn [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] / OVC signal status [RZ/G2E] Interrupt Status*4*5 0: No VBUS interrupt is generated. 1: A VBUS interrupt is generated. This bit indicates 1 when this module detects a change in the VBUSn [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] /OVC signal status [RZ/G2E] pin input level (high to low or low to high level). This module indicates the VBUSn [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] /OVC signal status [RZ/G2E] pin input value using the VBSTS bit. When a VBINT interrupt occurs, read the VBSTS bit several times by the software and confirm that the read value is equal each time to eliminate chattering. |
| 14 | RESM | B'0 | RW0C | Resume Interrupt Status*4*5 0: No resume interrupt is generated. 1: A resume interrupt is generated. This bit indicates 1 when this module detects a falling of the DP_0 pin level in the suspended state (DVSQ = 1XX). |
| 13 | SOFR | B'0 | RW0C | Frame Number Update Interrupt Status*4 0: No SOF interrupt is generated. 1: An SOF interrupt is generated. This module indicates SOFR = 1 at the frame number update timing. (This interrupt is detected every millisecond.) Even when an SOF packet from the USB host is corrupted, this module detects an SOFR interrupt using the internal interpolation. |
| 12 | DVST | B'0/B'1 *1 | RW0C | Device State Transition Interrupt Status*4 0: No device state transition interrupt is generated. 1: A device state transition interrupt is generated. When this module detects a change in device state, this module updates the DVSQ value and set this bit to 1. When this interrupt occurred, clear the status before this module detects the next device state transition. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|---|
| 11 | CTRT | B'0 | RW0C | <p>Control Transfer Stage Transition Interrupt Status*4</p> <p>0: No control transfer stage transition interrupt is generated. 1: A control transfer stage transition interrupt is generated.</p> <p>When this module detects a control transfer stage transition, this module updates the CTSQ value and set this bit to 1.</p> <p>When this interrupt occurred, clear the status before this module detects the next stage transition in control transfers.</p> |
| 10 | BEMP | B'0 | R | <p>Buffer Empty Interrupt Status</p> <p>0: No BEMP interrupt is generated. 1: A BEMP interrupt is generated.</p> <p>This module indicates BEMP = 1 when at least one of the PIPEBEMP bits in BEMPSTS corresponding to the pipes for which 1 is set in the PIPEBEMPE bits in BEMPENB is set to 1 (when this module detects a BEMP interrupt of one or more pipes for which BEMP interrupts are enabled by the software).</p> <p>For PIPEBEMP status assertion conditions, see section 61.3.2 (3), BEMP Interrupt.</p> <p>When the software writes 0 to all PIPEBEMP bits corresponding to the pipes for which interrupts are enabled by the PIPEBEMPE bits, this module clears this bit to 0.</p> <p>This bit cannot be cleared to 0 by writing 0 by the software.</p> |
| 9 | NRDY | B'0 | R | <p>Buffer Not Ready Interrupt Status</p> <p>0: No NRDY interrupt is generated. 1: An NRDY interrupt is generated.</p> <p>This module indicates NRDY = 1 when at least one of the PIPENRDY bits in NRDYSTS corresponding to the pipes for which 1 is set in the PIPENRDYE bits in NRDYENB is set to 1 (when this module detects an NRDY interrupt of one or more pipes for which NRDY interrupts are enabled by the software).</p> <p>For PIPENRDY status assertion conditions, see section 61.3.2 (2), NRDY Interrupt.</p> <p>When the software writes 0 to all PIPENRDY bits corresponding to the pipes for which interrupts are enabled by the PIPENRDYE bits, this module clears this bit to 0.</p> <p>This bit cannot be cleared to 0 by writing 0 by the software.</p> |
| 8 | BRDY | B'0 | R | <p>Buffer Ready Interrupt Status</p> <p>Indicates the BRDY interrupt status.</p> <p>0: No BRDY interrupt is generated. 1: A BRDY interrupt is generated.</p> <p>This module indicates BRDY = 1 when at least one of the PIPEBRDY bits in BRDYSTS corresponding to the pipes for which 1 is set in the PIPEBRDYE bits in BRDYENB is set to 1 (when this module detects a BRDY interrupt of one or more pipes for which BRDY interrupts are enabled by the software).</p> <p>For PIPEBRDY status assertion conditions, see section 61.3.2 (1), BRDY Interrupt.</p> <p>When the software writes 0 to all PIPEBRDY bits corresponding to the pipes for which interrupts are enabled by the PIPEBRDYE bits, this module clears this bit to 0.</p> <p>This bit cannot be cleared to 0 by writing 0 by the software.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|-------------------|------|---|
| 7 | VBSTS | B'0/B'1 *3 | R | VBUS Input Status 0: The VBUS pin is low level. 1: The VBUS pin is high level. [RZ/G2E] This value is 1, when USB0_OVC function is not selected by PFC. |
| 6 to 4 | DVSQ[2:0] | B'000/B '001*2 | R | Device State B'000: Powered state B'001: Default state B'010: Address state B'011: Configuration state B'1xx: Suspended state |
| 3 | VALID | B'0 | RW0C | USB Request Receive 0: Not detected 1: A setup packet is received. |
| 2 to 0 | CTSQ[2:0] | B'000 | R | Control Transfer Stage B'000: Idle stage or setup stage B'001: Control read data stage B'010: Control read status stage B'011: Control write data stage B'100: Control write status stage B'101: Control write (NoData) status stage B'110: Control transfer sequence error B'111: Setting prohibited |

- Notes:
1. The initial value is B'0 after a power-on reset, and B'1 after a USB bus reset.
 2. The initial value is B'000 after a power-on reset, and B'001 after a USB bus reset.
 3. The initial value is 1 when the VBUS pin is high level, and is 0 when the VBUS pin is low level.
 4. To clear the VBINT, RESM, SOFR, DVST, or CTRT bit, write 0 only to bits to be cleared and write 1 to the other bits. Do not write 0 to status bits that indicate 0.
 5. When a status change shown in the VBINT or RESM bit is detected while the clock is halted (SUSPM = 0) and the corresponding interrupt is enabled, the interrupt is output. Clear the interrupt status of these bits by the software after the clock is enabled.

61.2.15 BRDY Interrupt Status Register (BRDYSTS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

BRDYSTS indicates the BRDY interrupt status of each pipe.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PIPEF BRDY | PIPEE BRDY | PIPED BRDY | PIPEC BRDY | PIPEB BRDY | PIPEA BRDY | PIPE9 BRDY | PIPE8 BRDY | PIPE7 BRDY | PIPE6 BRDY | PIPE5 BRDY | PIPE4 BRDY | PIPE3 BRDY | PIPE2 BRDY | PIPE1 BRDY | PIPE0 BRDY |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------|---------------|-------|--|
| 15 | PIPEFBRDY | B'0 | R/W*1 | Pipe F BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |
| 14 | PIPEEBRDY | B'0 | R/W*1 | Pipe E BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |
| 13 | PIPEDBRDY | B'0 | R/W*1 | Pipe D BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |
| 12 | PIPECBRDY | B'0 | R/W*1 | Pipe C BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |
| 11 | PIPEBBRDY | B'0 | R/W*1 | Pipe B BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |
| 10 | PIPEABRDY | B'0 | R/W*1 | Pipe A BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |
| 9 | PIPE9BRDY | B'0 | R/W*1 | Pipe 9 BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |
| 8 | PIPE8BRDY | B'0 | R/W*1 | Pipe 8 BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |
| 7 | PIPE7BRDY | B'0 | R/W*1 | Pipe 7 BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |
| 6 | PIPE6BRDY | B'0 | R/W*1 | Pipe 6 BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |
| 5 | PIPE5BRDY | B'0 | R/W*1 | Pipe 5 BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------|---------------|-------|--|
| 4 | PIPE4BRDY | B'0 | R/W*1 | Pipe 4 BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |
| 3 | PIPE3BRDY | B'0 | R/W*1 | Pipe 3 BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |
| 2 | PIPE2BRDY | B'0 | R/W*1 | Pipe 2 BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |
| 1 | PIPE1BRDY | B'0 | R/W*1 | Pipe 1 BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |
| 0 | PIPE0BRDY | B'0 | R/W*1 | Pipe 0 BRDY Interrupt Status*2 0: No interrupt is generated. 1: An interrupt is generated. |

Notes: 1. To clear the status of each bit in this register when BRDYM is 0, write 0 only to bits to be cleared and write 1 to the other bits.

2. When BRDYM is 0, be sure to clear this interrupt before making an access to the FIFO.

61.2.16 NRDY Interrupt Status Register (NRDYSTS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

NRDYSTS indicates the NRDY interrupt status of each pipe.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PIPEF NRDY | PIPEE NRDY | PIPED NRDY | PIPEC NRDY | PIPEB NRDY | PIPEA NRDY | PIPE9 NRDY | PIPE8 NRDY | PIPE7 NRDY | PIPE6 NRDY | PIPE5 NRDY | PIPE4 NRDY | PIPE3 NRDY | PIPE2 NRDY | PIPE1 NRDY | PIPE0 NRDY |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------|---------------|------|--|
| 15 | PIPEFNRDY | B'0 | R/W* | Pipe F NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 14 | PIPEENRDY | B'0 | R/W* | Pipe E NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 13 | PIPEDNRDY | B'0 | R/W* | Pipe D NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 12 | PIPECNRDY | B'0 | R/W* | Pipe C NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 11 | PIPEBNRDY | B'0 | R/W* | Pipe B NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 10 | PIPEANRDY | B'0 | R/W* | Pipe A NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 9 | PIPE9NRDY | B'0 | R/W* | Pipe 9 NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 8 | PIPE8NRDY | B'0 | R/W* | Pipe 8 NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 7 | PIPE7NRDY | B'0 | R/W* | Pipe 7 NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 6 | PIPE6NRDY | B'0 | R/W* | Pipe 6 NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 5 | PIPE5NRDY | B'0 | R/W* | Pipe 5 NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------|---------------|------|--|
| 4 | PIPE4NRDY | B'0 | R/W* | Pipe 4 NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 3 | PIPE3NRDY | B'0 | R/W* | Pipe 3 NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 2 | PIPE2NRDY | B'0 | R/W* | Pipe 2 NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 1 | PIPE1NRDY | B'0 | R/W* | Pipe 1 NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 0 | PIPE0NRDY | B'0 | R/W* | Pipe 0 NRDY Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |

Note: * To clear the status of each bit in this register, write 0 only to bits to be cleared and write 1 to the other bits.

61.2.17 BEMP Interrupt Status Register (BEMPSTS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

BEMPSTS indicates the BEMP interrupt status of each pipe.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PIPEF BEMP | PIPEE BEMP | PIPED BEMP | PIPEC BEMP | PIPEB BEMP | PIPEA BEMP | PIPE9 BEMP | PIPE8 BEMP | PIPE7 BEMP | PIPE6 BEMP | PIPE5 BEMP | PIPE4 BEMP | PIPE3 BEMP | PIPE2 BEMP | PIPE1 BEMP | PIPE0 BEMP |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------|---------------|------|--|
| 15 | PIPEFBEMP | B'0 | R/W* | Pipe F BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 14 | PIPEEBEMP | B'0 | R/W* | Pipe E BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 13 | PIPEDBEMP | B'0 | R/W* | Pipe D BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 12 | PIPECBEMP | B'0 | R/W* | Pipe C BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 11 | PIPEBBEMP | B'0 | R/W* | Pipe B BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 10 | PIPEABEMP | B'0 | R/W* | Pipe A BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 9 | PIPE9BEMP | B'0 | R/W* | Pipe 9 BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 8 | PIPE8BEMP | B'0 | R/W* | Pipe 8 BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 7 | PIPE7BEMP | B'0 | R/W* | Pipe 7 BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 6 | PIPE6BEMP | B'0 | R/W* | Pipe 6 BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 5 | PIPE5BEMP | B'0 | R/W* | Pipe 5 BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------|---------------|------|--|
| 4 | PIPE4BEMP | B'0 | R/W* | Pipe 4 BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 3 | PIPE3BEMP | B'0 | R/W* | Pipe 3 BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 2 | PIPE2BEMP | B'0 | R/W* | Pipe 2 BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 1 | PIPE1BEMP | B'0 | R/W* | Pipe 1 BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |
| 0 | PIPE0BEMP | B'0 | R/W* | Pipe 0 BEMP Interrupt Status 0: No interrupt is generated. 1: An interrupt is generated. |

Note: * To clear the status of each bit in this register, write 0 only to bits to be cleared and write 1 to the other bits.

61.2.18 Frame Number Register (FRMNUM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

FRMNUM indicates the sources of isochronous errors and a frame number.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|------|------|----|----|----|------------|---|---|---|---|---|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | OVRN | CRCE | — | — | — | FRNM[10:0] | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | RW0C | RW0C | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|------|--|
| 15 | OVRN | B'0 | RW0C | <p>Overrun/Underrun Detect Status</p> <p>Indicates whether an overrun or underrun error is detected or not in a pipe that is performing isochronous transfer.</p> <p>0: No error is detected. 1: An error is detected.</p> <p>Writing 0 to this bit by the software clears this bit to 0. At this time, write 1 to the other bits in this register.</p> <p>This module indicates OVRN =1 in any of the following cases.</p> <p>When the IN token is received when transmit data write to the FIFO buffer is not completed in a transmit direction pipe of isochronous transfer type</p> <p>When the OUT token is received with no side of the FIFO buffer empty in a receive direction pipe of isochronous transfer type</p> |
| 14 | CRCE | B'0 | RW0C | <p>Receive Data Error</p> <p>Indicates whether a CRC error or bit stuffing error is detected or not in a pipe that is performing isochronous transfer.</p> <p>0: No error is detected. 1: An error is detected.</p> <p>Writing 0 to this bit by the software clears this bit to 0. At this time, write 1 to the other bits in this register.</p> <p>When a CRC error is detected, this module does not generate an internal NRDY interrupt request.</p> |
| 13 to 11 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 10 to 0 | FRNM [10:0] | H'000 | R | <p>Frame Number</p> <p>This module modifies this bit when an SOF is received and indicates the latest frame number.</p> <p>Read these bits twice and confirm that the read value is equal each time.</p> |

61.2.19 μ Frame Number Register (UFRMNUM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

UFRMNUM indicates a μ frame number.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|---|---|---|------------|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | UFRNM[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

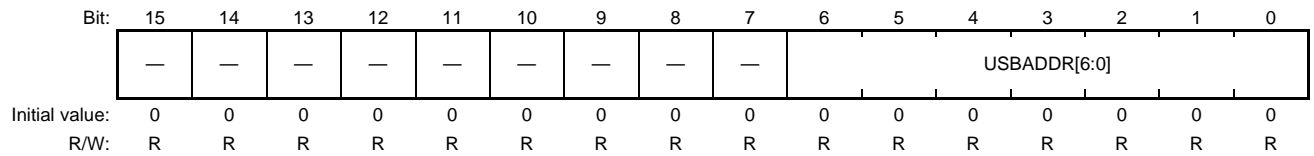
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 15 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 to 0 | UFRNM[2:0] | B'000 | R | μ Frame Indicate a μ frame number. In high-speed operating mode, these bits indicate a μ frame number. In other operating modes, these bits indicate B'000. Read these bits twice and confirm that the read value is equal each time. |

61.2.20 USB Address Register (USBADDR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

USBADDR indicates a USB address.

This register is initialized by a power-on reset or USB bus reset.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|--|
| 15 to 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 to 0 | USBADDR[6:0] | H'00 | R | USB Address Indicate the USB address allocated by the host when the SET_ADDRESS request is successfully processed. When this module detects a USB reset, these bits indicate H'00. |

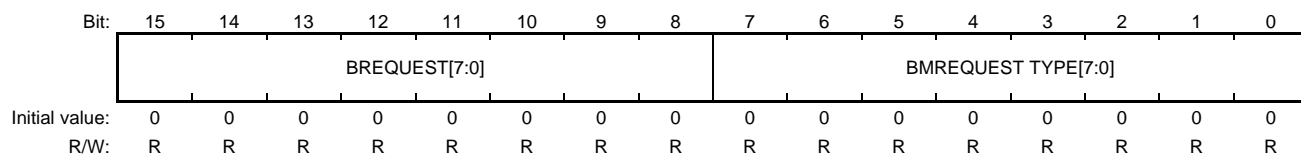
61.2.21 USB Request Type Register (USBREQ)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

USBREQ stores the setup request for control transfers.

USBREQ stores the values of received bRequest and bmRequestType.

This register is initialized by a power-on reset or USB bus reset.



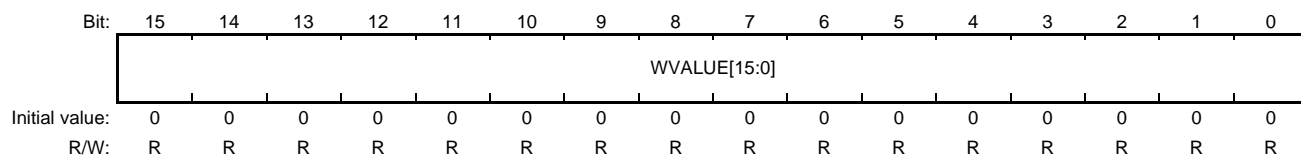
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------|---------------|-----|--|
| 15 to 8 | BREQUEST [7:0] | H'00 | R | Request Store the value of USB request bRequest. Indicate the USB request data received in the SETUP transaction. These bits cannot be modified. |
| 7 to 0 | BMREQUEST TYPE[7:0] | H'00 | R | Request Type Store the value of USB request bmRequestType. Indicate the USB request data received in the SETUP transaction. These bits cannot be modified. |

61.2.22 USB Request Value Register (USBVAL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

USBVAL stores the value of received wValue.

This register is initialized by a power-on reset or USB bus reset.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 15 to 0 | WVALUE [15:0] | H'0000 | R | Value Store the value of USB request wValue. Indicate the value of USB request wValue received in the SETUP transaction. These bits cannot be modified. |

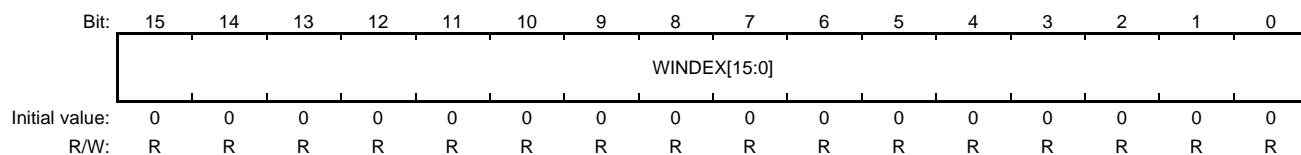
61.2.23 USB Request Index Register (USBINDX)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

USBINDX stores the setup request for control transfers.

USBINDX stores the value of received wIndex.

This register is initialized by a power-on reset or USB bus reset.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 15 to 0 | WINDEX [15:0] | H'0000 | R | Index Store the value of USB request wIndex. Indicate the value of USB request wIndex received in the SETUP transaction. These bits cannot be modified. |

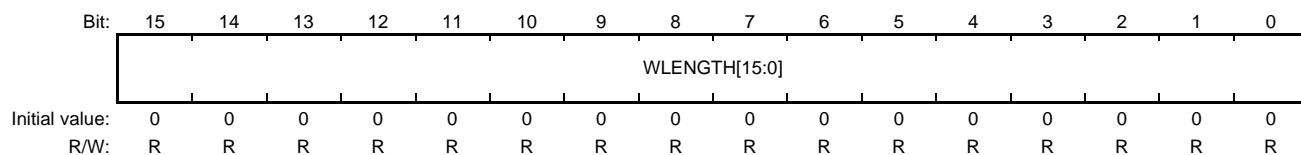
61.2.24 USB Request Length Register (USBLENG)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

USBLENG stores the setup request of control transfers.

USBLENG stores the value of received wLength.

This register is initialized by a power-on reset or USB bus reset.



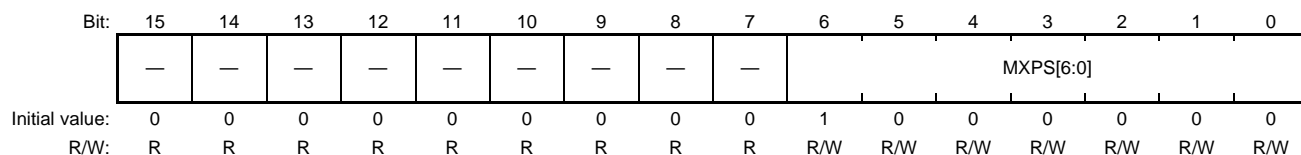
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|--|
| 15 to 0 | WLENGTH [15:0] | H'0000 | R | Length Store the value of USB request wLength. Indicate the value of USB request wLength received in the SETUP transaction. These bits cannot be modified. |

61.2.25 DCP Maximum Packet Size Register (DCPMAXP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DCPMAXP specifies the maximum packet size of the DCP.

This register is initialized by a power-on reset.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 15 to 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 6 to 0 | MXPS [6:0] | H'40 | R/W | Maximum Packet Size Set the payload of DCP data (maximum DCP packet size) in these bits. The initial value of these bits is H'40 (64 bytes). Set a value in the MXPS bits based on the USB Specification. Set the MXPS bits when PID = NAK, and the CURPIPE bits are not set. When setting these bits to 1 after changing the PID bits for the pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK. Do not write the FIFO buffer when the MXPS bits are 0 or set the PID bits to BUF. |

61.2.26 DCP Control Register (DCPCTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

DCPCTR is used to monitor the buffer memory status, change/check the data PID sequence bit, and set the response PID for a DCP.

This register is initialized by a power-on reset. The PID2 to PID0 bits in CCPL are also initialized by a USB bus reset.

| | | | | | | | | | | | | | | | | |
|----------------|------|----|----|----|----|----|---|-------|-------|-------|-------|---|---|------|----------|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | BSTS | — | — | — | — | — | — | SQCLR | SQSET | SQMON | PBUSY | — | — | CCPL | PID[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|------|--|
| 15 | BSTS | B'0 | R | <p>Buffer Status</p> <p>Indicates whether the DCP FIFO buffer is accessible or not.</p> <p>0: The buffer is not accessible.</p> <p>1: The buffer is accessible.</p> <p>This bit indicates as follows depending on the ISEL value.</p> <p>When ISEL = 0, this bit indicates whether the buffer is ready to read receive data.</p> <p>When ISEL = 1, this bit indicates whether the buffer is ready to write transmit data.</p> |
| 14 to 9 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 8 | SQCLR | B'0 | R/W* | <p>Toggle Bit Clear</p> <p>The expected value of the sequence toggle bit for the next transaction in DCP transfer can be set in DATA0.</p> <p>0: Invalid</p> <p>1: The expected value is set in DATA0.</p> <p>This bit always indicates 0.</p> <p>Do not set the SQCLR and SQSET bits to 1 at the same time.</p> <p>Set this bit to 1 when PID = NAK, and the CURPIPE bits are not set.</p> <p>When setting this bit to 1 after changing the PID bits for the pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> |
| 7 | SQSET | B'0 | R/W* | <p>Toggle Bit Set</p> <p>The expected value of the sequence toggle bit for the next transaction in DCP transfer can be set in DATA1.</p> <p>0: Invalid</p> <p>1: The expected value is set in DATA1.</p> <p>Do not set the SQCLR and SQSET bits to 1 at the same time.</p> <p>Set this bit to 1 when PID = NAK, and the CURPIPE bits are not set.</p> <p>When setting this bit to 1 after changing the PID bits for the pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|------|---|
| 6 | SQMON | B'1 | R | <p>Sequence Toggle Bit Monitor</p> <p>Indicates the expected value of the sequence toggle bit for the next transaction in DCP transfer.</p> <p>0: DATA0 1: DATA1</p> <p>When the transaction is successfully completed, this module toggles this bit. However, when a DATA-PID mismatch occurs in the receive direction transfer, this bit is not toggled.</p> <p>This module sets this bit to 1 (sets the expected value in DATA1) when the setup packet is successfully received.</p> <p>This module does not read this bit in IN transactions or OUT transactions in the status stage.</p> |
| 5 | PBUSY | B'0 | R | <p>Pipe Busy</p> <p>Indicates whether the DCP communication state is shifted to the NAK state or not when the DCP changed the PID bits from BUF to NAK.</p> <p>0: Transition to the NAC state is not completed. 1: Transition to the NAC state is completed.</p> <p>When this module starts the USB transaction for the corresponding pipe, this module sets this bit to 1. This module clears this bit to 0 upon completion of each transaction.</p> <p>After the software sets PID = NAK, read this bit to check whether changing the pipe setting is enabled or not.</p> |
| 4, 3 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 2 | CCPL | B'0 | R/W* | <p>Control Transfer End Enable</p> <p>0: Invalid 1: The control transfer end is enabled.</p> <p>When the software sets this bit to 1 while the corresponding PID bits are set to BUF, this module finishes the control transfer stage.</p> <p>That is, this module sends an ACK handshake in response to OUT transactions from the USB host in the control read transfer, and sends a Zero-Length packet in response to IN transactions from the USB host in control write and no-data control transfers. However, a SET_ADDRESS request is detected, this module automatically sends responses from the SETUP stage until the end of the status stage irrespective of the setting of this bit.</p> <p>When this module receives the next setup packet, this module clears this bit to 0.</p> <p>When VALID = 1, the software cannot write 1 to this bit.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|--|
| 1, 0 | PID[1:0] | B'00 | R/W | <p>Response PID</p> <p>Control responses of this module in control transfers.</p> <p>B'00: NAK response</p> <p>B'01: BUF response (depending on buffer state)</p> <p>B'10: STALL response</p> <p>B'11: STALL response</p> <p>This module modifies these bits in the following cases.</p> <p>This module changes these bits to NAK upon receiving a SETUP packet. At this time, this module indicates VALID = 1. The software cannot modify these bits until the software sets VALID = 0.</p> <p>When this module receives data with a size exceeding the MaxPacketSize value while these bits are set to BUF by the software, this module indicates PID = STALL (11).</p> <p>When this module detects a control transfer sequence error, this module indicates PID = STALL (1x).</p> <p>When this module detects a USB bus reset, this module indicates PID = NAK.</p> <p>This module does not read the value of these bits during the SET_ADDRESS request processing (auto processing).</p> |

Note: * These bits are always read as 0. Only writing 1 is enabled.

61.2.27 Pipe Window Select Register (PIPESEL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Make settings for pipes 1 to F using PIPESEL, PIPECFG, PIPEBUF, PIPEMAXP, PIPEPERI, PIPEnCTR, PIPEnTRE, and PIPEnTRN.

Specify a pipe to be used with PIPESEL, and then make function settings for each pipe in PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI. The PIPEnCTR, PIPEnTRE, and PIPEnTRN registers can be set independently from pipe selection using PIPESEL.

Not only the selected pipe but also the corresponding bits in registers for all pipes are initialized by a power-on reset or USB bus reset.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|---|---|--------------|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | PIPESEL[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 3 to 0 | PIPESEL [3:0] | B'0000 | R/W | Pipe Window Select Specify a pipe number corresponding to PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI to be read or written. B'0000: No pipe is selected. B'0001: Pipe 1 B'0010: Pipe 2 B'0011: Pipe 3 B'0100: Pipe 4 B'0101: Pipe 5 B'0110: Pipe 6 B'0111: Pipe 7 B'1000: Pipe 8 B'1001: Pipe 9 B'1010: Pipe A B'1011: Pipe B B'1100: Pipe C B'1101: Pipe D B'1110: Pipe E B'1111: Pipe F PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI corresponding to the pipe number specified by these bits can be read and written. When these bits are set to B'0000, all bits in PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI are read as 0 and cannot be modified. |

61.2.28 Pipe Configuration Register (PIPECFG)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

PIPECFG specifies the transfer type, buffer memory access direction, and endpoint number for pipes 1 to F, selects continuous transfer mode or discontinuous transfer mode, and single buffer or double buffer, and also specifies whether to disable the operation of each pipe when data transfer finishes.

This register is initialized by a power-on reset. The TYPE1 and TYPE0 bits are also initialized by a USB bus reset.

| | | | | | | | | | | | | | | | | |
|----------------|-----------|-----|----|----|----|------|------|-----------|------------|---|---|-----|------------|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TYPE[1:0] | | — | — | — | BFRE | DBLB | CNTM D | SHTNA K | — | — | DIR | EPNUM[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 15, 14 | TYPE[1:0] | B'00 | R/W | <p>Transfer Type</p> <p>Select the transfer type of the pipe specified by the PIPESEL bits (selected pipe).</p> <p>Pipes 1 and 2</p> <p>B'00: No pipe is used. B'01: Bulk transfer B'10: Setting prohibited B'11: Isochronous transfer</p> <p>Pipes 3 to 5, 9 to F</p> <p>B'00: No pipe is used. B'01: Bulk transfer B'10: Setting prohibited B'11: Setting prohibited</p> <p>Pipes 6 to 8</p> <p>B'00: No pipe is used. B'01: Setting prohibited B'10: Interrupt transfer B'11: Setting prohibited</p> <p>Before setting the selected pipe for PID = BUF (before starting USB communication using the selected pipe), be sure to set these bits to a value other than B'00.</p> <p>Modify these bits when the PID bits of the selected pipe are set to NAK. When changing the setting of these bits after changing the PID bits for the selected pipe from BUF to NAK, check that and PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> |
| 13 to 11 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 10 | BFRE | B'0 | R/W | <p>BRDY Interrupt Operation</p> <p>Specifies the BRDY interrupt output timing from this module to the CPU for the selected pipe.</p> <p>0: At a BRDY interrupt timing during data transmission or reception 1: At a BRDY interrupt timing after reading data</p> <p>When this bit is set to 1 by the software and the selected pipe is used in the receive direction, this module detects the transfer end and outputs a BRDY interrupt when the packet is completely read.</p> <p>When a BRDY interrupt occurs with this setting, the software must set BCLR = 1. Unless BCLR = 1 is set, the FIFO buffer allocated to the selected pipe does not enter the receive ready state.</p> <p>When this bit is set to 1 by the software and the selected pipe is used in the transmit direction, this module generates no BRDY interrupt.</p> <p>For details, see section 61.3.2 (1), BRDY Interrupt.</p> <p>Modify this bit when PID = NAK, and no pipe number is specified by the CURPIPE bits.</p> <p>Furthermore, when changing the setting of this bit after USB communication using the selected pipe, set the ACLRM bit to 1 and clear it to 0 continuously by the software to clear the FIFO buffer allocated to the selected pipe, in addition to the three register states above.</p> <p>When changing the setting of this bit after changing the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> |
| 9 | DBLB | B'0 | R/W | <p>Double Buffer Mode</p> <p>Specifies a single buffer or double buffer for the FIFO buffer used in the selected pipe.</p> <p>0: Single buffer 1: Double buffer</p> <p>This bit is valid when pipes 1 to 5, 9 to F are selected.</p> <p>When this bit is set to 1 by the software, this module allocates the FIFO buffer size specified by the BUFSIZE bits in PIPEBUF to the selected pipe for two sides.</p> <p>Thus the size of the FIFO buffer that this module allocates to the selected pipe is as follows: (BUFSIZE + 1) × 64 × (DBLB + 1) [bytes]</p> <p>When this bit is set to 1 by the software and the selected pipe is used in the transmit direction, this module generates no BRDY interrupt.</p> <p>For details, see section 61.3.2 (1), BRDY Interrupt.</p> <p>Modify this bit when PID = NAK, and no pipe number is specified by the CURPIPE bits.</p> <p>Furthermore, when changing the setting of this bit after USB communication using the selected pipe, set the ACLRM bit to 1 and clear it to 0 continuously by the software to clear the FIFO buffer allocated to the selected pipe, in addition to the three register states above.</p> <p>When changing the setting of this bit after changing the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|--|
| 8 | CNTMD | B'0 | R/W | <p>Continuous Transfer Mode</p> <p>Selects continuous transfer mode or discontinuous transfer mode to be used in communication for the selected pipe.</p> <p>0: Discontinuous transfer mode 1: Continuous transfer mode</p> <p>This bit is valid when pipes 1 to 5, 9 to F are selected by the PIPESEL bits and bulk transfer is selected (TYPE = 01).</p> <p>Modify this bit when PID = NAK, and no pipe number is specified by the CURPIPE bits.</p> <p>Furthermore, when changing the setting of this bit after USB communication using the selected pipe, set the ACLRM bit to 1 and clear it to 0 continuously by the software to clear the FIFO buffer allocated to the selected pipe, in addition to the three register states above.</p> <p>When changing the setting of this bit after changing the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> |
| 7 | SHTNAK | B'0 | R/W | <p>Pipe Disable at the Time of Transfer End</p> <p>Specifies whether to change the setting of the PID bits to NAK at the end of transfer when the selected pipe is in the receive direction.</p> <p>0: The pipe is continued at the end of transfer. 1: The pipe is disabled at the end of transfer.</p> <p>This bit is valid when the selected pipe is either of pipes 1 to 5, 9 to F and is in the receive direction.</p> <p>When the software sets this bit to 1 for a receive direction pipe, this module changes the setting of the PID bits corresponding to the selected pipe to NAK for the selected pipe when this module determines the transfer end. When the following conditions are satisfied, this module determines the transfer end.</p> <p>When short packet data (including a Zero-Length packet) is correctly received.</p> <p>When the transaction counter is used and packets with a size of the transaction counter are correctly received.</p> <p>Modify this bit when PID = NAK.</p> <p>When changing the setting of this bit after changing the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> <p>For transmit direction pipes, set this bit to 0.</p> |
| 6, 5 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

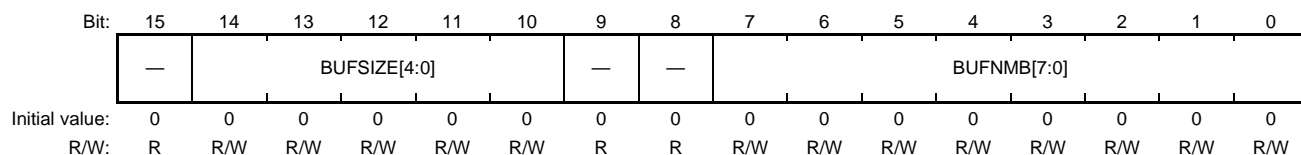
| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-------------|---------------|-----|--|
| 4 | DIR | B'0 | R/W | <p>Transfer Direction</p> <p>Selects the transfer direction of the selected pipe.</p> <p>0: Receive direction 1: Transmit direction</p> <p>When this bit is set to 0 by the software, this module uses the selected pipe in the receive direction. When this bit is set to 1, this module uses the selected pipe in the transmit direction.</p> <p>Modify this bit when PID = NAK, and no pipe number is specified by the CURPIPE bits.</p> <p>Furthermore, when changing the setting of this bit after USB communication using the selected pipe, set the ACLRM bit to 1 and clear it to 0 continuously by the software to clear the FIFO buffer allocated to the selected pipe, in addition to the three register states above.</p> <p>When changing the setting of this bit after changing the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> |
| 3 to 0 | EPNUM [3:0] | B'0000 | R/W | <p>Endpoint Number</p> <p>Specify the endpoint number of the selected pipe.</p> <p>The setting of B'0000 means an unused pipe.</p> <p>Modify these bits when PID = NAK.</p> <p>When changing the setting of these bits after changing the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> <p>Set these bits so that the combination of the DIR setting and the EPNUM setting does not overlap the setting of other pipes (EPNUM = B'0000 is allowed in this case.)</p> |

61.2.29 Pipe Buffer Register (PIPEBUF)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

PIPEBUF specifies the buffer size and buffer number for pipes 1 to F.

This register is initialized by a power-on reset.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 15 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 14 to 10 | BUFSIZE [4:0] | H'00 | R/W | Buffer Size Specify the buffer size of the pipe specified by the PIPESEL bits (selected pipe) in units of blocks. One block is 64 bytes. B'0_0000 (H'00): 64 bytes B'0_0001 (H'01): 128 bytes : B'1_1111 (H'1F): 2 Kbytes When DBLB = 1 is set by the software, this module allocates the FIFO buffer size specified by these bits to the selected pipe for two sides. Thus the size of the FIFO buffer that this module allocates to the selected pipe is as follows: $(BUFSIZE + 1) \times 64 \times (DBLB + 1)$ [bytes] A value that can be set in these bits varies depending on pipes selected. For pipes 1 to 5, 9 to F: Set BUFSIZE to H'00 to H'1F. For pipes 6 to 8: Set BUFSIZE to H'00. When using the buffer with CNTMD = 1, set a value in multiples of MaxPacketSize in these bits. Modify these bits when PID = NAK, and no pipe number is specified by the CURPIPE bits. When changing the setting of this bit after changing the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK. |
| 9, 8 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

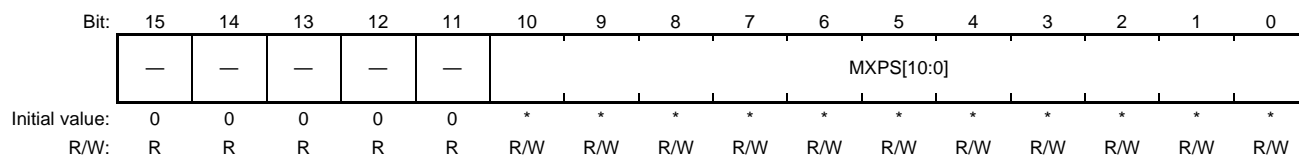
| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------------|---------------|-----|--|
| 7 to 0 | BUFNMB [7:0] | H'00 | R/W | <p>Buffer Number</p> <p>Specify the FIFO buffer number of the selected pipe with a value from H'04 to H'FF.</p> <p>When the selected pipe is either of pipes 1 to 5, 9 to F, these bits can be set to a value according to user systems. Set BUFNMB to H'06 to H'4F. BUFNMB = 0 to 3 are for DCP only. BUFNMB = 4 is for pipe 6 only. However, when pipe 6 is not used, this is available for other pipes. When pipe 6 is selected, these bits cannot be modified and this module automatically allocates BUFNMB = 4. BUFNMB = 5 is for pipe 7 only. However, when pipe 7 is not used, this is available for other pipes. When pipe 7 is selected, these bits cannot be modified and this module automatically allocates BUFNMB = 5. BUFNMB = 6 is for pipe 8 only. However, when pipe 8 is not used, this is available for other pipes. When pipe 8 is selected, these bits cannot be modified and this module automatically allocates BUFNMB = 6.</p> <p>Modify these bits when PID = NAK, and no pipe number is set in the CURPIPE bits.</p> <p>When changing the setting of these bits after changing the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> |

61.2.30 Pipe Maximum Packet Size Register (PIPEMAXP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

PIPEMAXP specifies the maximum packet size for pipes 1 to F.

This register is initialized by a power-on reset.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 15 to 11 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 to 0 | MXPS[10:0] | * | R/W | <p>Maximum Packet Size</p> <p>Specify the payload of data (maximum packet size) of the selected pipe. The settable value range for each pipe is as follows: Pipes 1 and 2: 1 byte (H'001) to 1024 bytes (H'400) Pipes 3 to 5, 9 to F: 8 (H'008), 16 (H'010), 32 (H'020), 64 (H'040), or 512 bytes (H'200) (MXPS2 to MXPS0 bits are not provided.) Pipes 6 to 8: 1 byte (H'001) to 64 bytes (H'040)</p> <p>Set a value in the MXPS bits based on the USB Specification for each transfer type.</p> <p>To transmit an isochronous pipe in split transactions, set a value of 188 bytes or less in the MXPS bits.</p> <p>When changing the setting of this bit after changing the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> <p>Do not write the FIFO buffer when the MXPS = 0 or set the PID bits to BUF.</p> |

Note: * The initial value is H'000 (when no pipe is selected by the PIPESEL bits in PIPESEL) or H'040 (when a pipe is selected).

61.2.31 Pipe Cycle Control Register (PIPEPERI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

PIPEPERI specifies whether to activate the buffer flush function or not at the time of an interval error during the isochronous IN transfer, and also specifies the interval error detection interval for pipes 1 to F.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|------|----|----|---|---|---|---|---|---|---|-----------|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | IFIS | — | — | — | — | — | — | — | — | — | IITV[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 15 to 13 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 12 | IFIS | B'0 | R/W | <p>Isochronous IN Buffer Flush</p> <p>Specifies whether to flush the FIFO buffer when the pipe specified by the PIPESEL bits (selected pipe) is isochronous IN transfer type.</p> <p>0: The FIFO buffer is not flushed. 1: The FIFO buffer is flushed.</p> <p>Buffer flush is a function that this module automatically clears the FIFO buffer when this module does not receive IN-Token from the USB host in a (μ) frame for each interval specified in the IITV bits when the selected pipe is isochronous IN transfer type.</p> <p>When double buffer is selected (DBLB = 1), this module clears data of only the older side.</p> <p>The FIFO buffer is cleared when an SOF packet is received immediately after the (μ) frame where IN-Token is to be received. Even if the SOF packet is corrupted, the FIFO buffer is cleared at the time when the SOF packet is to be received using the internal interpolation.</p> <p>When the selected pipe is not isochronous transfer type, set this bit to 0.</p> |
| 11 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 to 0 | IITV[2:0] | B'000 | R/W | <p>Interval Error Detection Interval</p> <p>Specify the interval error detection interval of the selected pipe with a value of 2's n-th power of the frame timing.</p> <p>Set these bits when PID = NAK, and no pipe number is specified by the CURPIPE bits.</p> <p>When changing the setting of these bits after changing the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> <p>To change the setting of these bits after USB communication, set PID = NAK and then set ACLRM = 1 to initialize the interval timer.</p> <p>These bits are not provided for pipes 3 to F. Set B'000 in the position of these bits corresponding to pipes 3 to F.</p> |

61.2.32 Pipe n Control Register (PIPEnCTR) (n = 1 to F)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

PIPEnCTR monitors the buffer memory status, changes/checks the data PID sequence bits, selects whether to use auto-response mode and buffer auto-clear mode, and specifies a response PID for pipes 1 to F. This register can be set independently of the pipe selection using PIPESEL.

This register is initialized by a power-on reset. The PID1 and PID0 bits are also initialized by a USB bus reset.

(1) PIPEnCTR (n = 1 to 5, 9 to F)

| | | | | | | | | | | | | | | | | |
|----------------|------|--------|----|----|----|--------|--------|-------|-------|-------|-------|---|---|---|----------|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | BSTS | INBUFM | — | — | — | ATREPM | ACLARM | SQCLR | SQSET | SQMON | PBUSY | — | — | — | PID[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 15 | BSTS | B'0 | R | <p>Buffer Status</p> <p>Indicates the FIFO buffer status of the pipe.</p> <p>0: The buffer is not accessible from the CPU.</p> <p>1: The buffer is accessible from the CPU.</p> <p>The meaning of this bit differs depending on the settings of the DIR, BFRE, and DCLRM bits as shown in Table 61.10.</p> |
| 14 | INBUFM | B'0 | R | <p>Transmit Buffer Monitor</p> <p>Indicates the FIFO buffer status of the selected pipe in the transmit direction.</p> <p>0: The buffer memory contains no transmittable data.</p> <p>1: The buffer memory contains transmittable data.</p> <p>When transmit direction (DIR = 1) is set for the pipe, this module sets this bit to 1 when the software (or DMAC) finishes writing data to the FIFO buffer for at least one register set.</p> <p>When this module finishes sending all data on the register set (writing is completed) of the FIFO buffer, this bit indicates 0. In the case of a double buffer (DBLB = 1), when this module finishes sending all data on both register sets and the software (or DMAC) has not completed writing of data for one register set, this bit indicates 0.</p> <p>When receive direction (DIR = 0) is set for the selected pipe, this bit indicates the same value as the BSTS bit.</p> |
| 13 to 11 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 10 | ATREPM | B'0 | R/W | <p>Auto-Response Mode</p> <p>Enables or disables auto-response for the selected pipe.</p> <p>0: Auto-response is disabled.</p> <p>1: Auto-response is enabled.</p> <p>When the transfer type of the selected pipe is set to bulk transfer, this bit can be set to 1.</p> <p>When this bit is 1, this module sends responses to tokens from the USB host as follows:</p> <p>(1) When the selected pipe is set for Bulk-IN transfer (TYPE = B'01 and DIR = 1)</p> <p>When ATREPM = 1 and PID = BUF, this module sends a Zero-Length packet in response to the IN-Token.</p> <p>Each time this module receives ACK from the USB host (flow of one transaction: receiving IN-Token -> sending Zero Length packet -> receiving ACK), this module updates the sequence toggle bit (DATA-PID) by toggling it.</p> <p>No BRDY interrupt or BEMP interrupt is generated.</p> <p>(2) When the selected pipe is set for Bulk-OUT transfer (TYPE = B'01 and DIR = 0)</p> <p>When ATREPM = 1 and PID = BUF, this module sends NAK in response to the OUT-Token (or PING-Token) and generates an NRDY interrupt.</p> <p>Modify this bit when PID = NAK.</p> <p>When setting this bit to 1 after changing the PID bits for the pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> <p>Be sure to set this bit to 1 for USB communication when the FIFO buffer is empty. Do not write data to the FIFO buffer during USB communication with this bit set to 1.</p> <p>When the selected pipe is set for isochronous transfer, be sure to set this bit to 0.</p> |
| 9 | ACLRM | B'0 | R/W | <p>Buffer Auto-Clear Mode</p> <p>Enables or disables buffer auto-clear mode for the selected pipe.</p> <p>0: Buffer auto-clear mode is disabled.</p> <p>1: Buffer auto-clear mode is enabled. (All buffers are initialized.)</p> <p>To completely delete the data in the FIFO buffer allocated to the pipe, write 1 and then 0 continuously to the ACLRM bit.</p> <p>Table 61.11 shows data cleared by this module when this bit is set to 1 and then 0 and cases where such data is cleared.</p> <p>Modify this bit when PID = NAK.</p> <p>When setting this bit to 1 after changing the PID bits for the pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|------|--|
| 8 | SQCLR | B'0 | R/W* | <p>Toggle Bit Clear</p> <p>Set this bit to 1 to clear the expected value (in DATA0) of the sequence toggle bit for the next transaction of the selected pipe.</p> <p>0: Invalid 1: The expected value is cleared in DATA0.</p> <p>When the software sets this bit to 1, this module sets the expected value of the sequence toggle bit of the pipe in DATA0. This module always indicates SQCLR = 0.</p> <p>Set the SQCLR bit to 1 when PID = NAK.</p> <p>When setting this bit to 1 after changing the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> |
| 7 | SQSET | B'0 | R/W* | <p>Toggle Bit Set</p> <p>Set this bit to 1 to set the expected value of the sequence toggle bit for the next transaction of the selected pipe.</p> <p>0: Invalid 1: The expected value is set in DATA1.</p> <p>When the software sets this bit to 1, this module sets the expected value of the sequence toggle bit of the pipe in DATA1. This module always indicates SQSET = 0.</p> <p>Set the SQSET bit to 1 when PID = NAK.</p> <p>When setting this bit to 1 after changing the PID bits for the pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> |
| 6 | SQMON | B'0 | R | <p>Toggle Bit Check</p> <p>Indicates the expected value of the sequence toggle bit for the next transaction of the selected pipe.</p> <p>0: DATA0 1: DATA1</p> <p>If the pipe is not the isochronous transfer type, when a transaction is successfully processed, this module toggles this bit. However, when a DATA-PID mismatch occurs in the receive direction transfer, this bit is not toggled.</p> |
| 5 | PBUSY | B'0 | R | <p>Pipe Busy</p> <p>Indicates whether the pipe is currently used in the USB bus or not.</p> <p>0: The selected pipe is not used in the USB bus. 1: The selected pipe is being used in the USB bus.</p> <p>When this module starts the USB transaction for the selected pipe, this module sets this bit to 1. This module clears this bit to 0 upon completion of each transaction.</p> <p>After the software sets PID = NAK, read this bit to see if changing the pipe setting is enabled or not.</p> |
| 4 to 2 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 1, 0 | PID[1:0] | B'00 | R/W | <p>Response PID</p> <p>Specify a response used in the next transaction for the selected pipe.</p> <p>B'00: NAK response</p> <p>B'01: BUF response (depending on buffer state)</p> <p>B'10: STALL response</p> <p>B'11: STALL response</p> <p>These bits are set to NAK by default. To perform USB transfer with the selected pipe, change the setting of these bits to BUF. Table 61.12 shows the basic operation (with no error in packets) of this module in each setting of the PID bits.</p> <p>When changing these bits from BUF to NAK by the software while the pipe is in USB communication, write 00 (NAK) to these bits and then check PBUSY = 1 to confirm that the USB transfer of the pipe entered the NAK state. However, when this module changed these bits to NAK, the software does not have to check the PBUSY bit.</p> <p>This module modifies these bits in the following cases.</p> <ul style="list-style-type: none"> When the pipe is in the receive direction and the software sets the SHTNAK bit of the selected pipe to 1, this module indicates PID = NAK when this module recognizes the transfer end. When this module receives data with a payload exceeding the MaxPacketSize value for the pipe, this module indicates PID = STALL (11). When this module detects a USB bus reset, this module indicates PID = NAK. <p>Set these bits as follows:</p> <ul style="list-style-type: none"> To change the state from NAK (00) to STALL, write B'10. To change the state from BUF (01) to STALL, write B'11. To change the state from STALL (11) to NAK, write B'10 then B'00. To change the state from STALL to BUF, set these bits to NAK and then to BUF. |

Note: * Only reading 0 and writing 1 are enabled.

Table 61.10 Operations of BSTS Bit

| DIR Bit | BFRE Bit | DCLRM Bit | Meaning of BSTS Bit |
|---------|----------|-----------|--|
| B'0 | B'0 | B'0 | Indicates 1 when the FIFO buffer becomes ready to read receive data from the buffer, and indicates 0 upon completion of the data read. |
| | | B'1 | Setting prohibited |
| | B'1 | B'0 | Indicates 1 when the FIFO buffer becomes ready to read receive data from the buffer, and indicates 0 when the software sets BCLR = 1 after the buffer data read. |
| | | B'1 | Indicates 1 when the FIFO buffer becomes ready to read receive data from the buffer, and indicates 0 upon completion of the data read. |
| B'1 | B'0 | B'0 | Indicates 1 when the FIFO buffer becomes ready to write transmit data to the buffer, and indicates 0 upon completion of the data write. |
| | | B'1 | Setting prohibited |
| | B'1 | B'0 | Setting prohibited |
| | | B'1 | Setting prohibited |

Table 61.11 Data Cleared by This Module when ACLRM = 1

| No. | Data Cleared by the ACLRM Bit | Clearing Timing |
|-----|---|--|
| 1 | All data in the FIFO buffer (both sides for double FIFO buffer) allocated to the pipe | — |
| 2 | Interval count value when the pipe is isochronous transfer type | When resetting the interval count value |
| 3 | Internal flags related to the BFRE bit | When changing the BFRE setting |
| 4 | FIFO buffer toggle control | When changing the DBLB setting |
| 5 | Internal flags related to the transaction count | When forcibly terminating the transaction count function |

Table 61.12 Operations of This Module in Each Setting of PID Bits

| PID Bits | Transfer Type | Transfer Direction (DIR Bit) | Operation of This Module |
|------------------------------------|-------------------|--|---|
| B'00 (NAK) | Bulk or interrupt | Independent of the setting | Sends a NAK response to a token from the USB host. |
| | Isochronous | Independent of the setting | Sends no response to a token from the USB host. |
| B'01 (BUF) | Bulk | Receive direction (DIR = 0) | Receives data and sends an ACK response to the OUT token from the USB host when the FIFO buffer for the pipe is ready to receive. Otherwise, sends a NAK response. Sends an ACK response to the PING token from the USB host when the FIFO buffer for the selected pipe is ready to receive. Otherwise, sends a NYET response. |
| | | Interrupt | Receive direction (DIR = 0) |
| | Bulk or interrupt | Transmit direction (DIR = 1) | Sends data in response to a token from the USB host when the FIFO buffer for the pipe is ready to transmit. Otherwise, sends a NAK response. |
| | Isochronous | Receive direction (DIR = 0) | Receives data in response to the OUT token from the USB host when the FIFO buffer for the pipe is ready to receive. Otherwise, ignores the data. |
| Transmit direction (DIR = 1) | | Sends data in response to a token from the USB host when the FIFO buffer for the pipe is ready to transmit. Otherwise, sends a Zero-Length packet. | |
| B'10 (STALL) or B'11 (STALL) | Bulk or interrupt | Independent of the setting | Sends a STALL response to a token from the USB host. |
| | Isochronous | Independent of the setting | Sends no response to a token from the USB host. |

(2) PIPE_nCTR (n = 6 to 8)

| | | | | | | | | | | | | | | | | |
|----------------|------|----|----|----|----|----|-------|-------|-------|-------|-------|---|---|---|----------|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | BSTS | — | — | — | — | — | ACLRM | SQCLR | SQSET | SQMON | PBUSY | — | — | — | PID[1:0] | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|---|
| 15 | BSTS | B'0 | R | <p>Buffer Status</p> <p>Indicates the state of the FIFO buffer of the selected pipe.</p> <p>0: The buffer is not accessible.</p> <p>1: The buffer is accessible.</p> <p>The meaning of this bit differs as shown in Table 61.10 depending on the settings of the DIR, BFRE, and DCLRM bits.</p> |
| 14 to 10 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 9 | ACLRM | B'0 | R/W | <p>Buffer Auto-Clear Mode*2*3</p> <p>Enables or disables buffer auto-clear mode for the selected pipe.</p> <p>0: Buffer auto-clear mode is disabled.</p> <p>1: Buffer auto-clear mode is enabled. (All buffers are initialized.)</p> <p>To completely delete the data in the FIFO buffer allocated to the pipe, write 1 and then 0 continuously to the ACLRM bit.</p> <p>Table 61.13 shows data cleared by this module when this bit is set to 1 and then 0 and cases where such data is cleared.</p> <p>Modify this bit when PID = NAK, and the pipe is not set in the CURPIPE bits.</p> <p>When setting this bit to 1 after changing the PID bits for the pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> |
| 8 | SQCLR | B'0 | R/W*1 | <p>Toggle Bit Clear*2*3</p> <p>Set this bit to 1 to clear the expected value (in DATA0) of the sequence toggle bit for the next transaction of the selected pipe.</p> <p>0: Invalid</p> <p>1: The expected value is cleared in DATA0.</p> <p>When the software sets this bit to 1, this module sets the expected value of the sequence toggle bit of the pipe in DATA0. This module always indicates SQCLR = 0.</p> <p>When this bit is set to 1 for a Bulk-Out transfer pipe with the host controller function selected, this module starts the next transfer with the PING-token for the pipe.</p> <p>Set the SQCLR bit to 1 when PID = NAK.</p> <p>When setting this bit to 1 after changing the PID bits for the pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-------|--|
| 7 | SQSET | B'0 | R/W*1 | <p>Toggle Bit Set*2*3</p> <p>Set this bit to 1 to set the expected value of the sequence toggle bit for the next transaction of the selected pipe.</p> <p>0: Invalid 1: The expected value is set in DATA1.</p> <p>When the software sets this bit to 1, this module sets the expected value of the sequence toggle bit of the selected pipe in DATA1. This module always indicates SQSET = 0.</p> <p>Set the SQSET bit to 1 when PID = NAK.</p> <p>When setting this bit to 1 after changing the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> |
| 6 | SQMON | B'0 | R | <p>Toggle Bit Check</p> <p>Indicates the expected value of the sequence toggle bit for the next transaction of the pipe.</p> <p>0: DATA0 1: DATA1</p> <p>If the selected pipe that is not the isochronous transfer type, when a transaction is successfully processed, this module toggles this bit. However, when a DATA-PID mismatch occurs in the receive direction transfer, this bit is not toggled.</p> |
| 5 | PBUSY | B'0 | R | <p>Pipe Busy</p> <p>Indicates whether the selected pipe is currently used in the USB bus or not.</p> <p>0: The selected pipe is not used in the USB bus. 1: The selected pipe is being used in the USB bus.</p> <p>When this module starts the USB transaction for the selected pipe, this module sets this bit to 1. This module clears this bit to 0 upon completion of each transaction.</p> <p>After the software sets PID = NAK, read this bit to see if changing the pipe setting is enabled or not.</p> |
| 4 to 2 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|--|
| 1, 0 | PID[1:0] | B'00 | R/W | <p>Response PID</p> <p>Specify a response used in the next transaction for the selected pipe.</p> <p>B'00: NAK response</p> <p>B'01: BUF response (depending on buffer state)</p> <p>B'10: STALL response</p> <p>B'11: STALL response</p> <p>These bits are set to NAK by default. To perform USB transfer with the selected pipe, change the setting of these bits to BUF. Table 61.12 shows the basic operation (with no error in packets) of this module in each setting of the PID bits.</p> <p>When changing these bits from BUF to NAK by the software while the pipe is in USB communication, write 00 (NAK) to these bits and then check PBUSY = 1 to confirm that the USB transfer of the selected pipe entered the NAK state. However, when this module changed these bits to NAK, the software needs not check the PBUSY bit.</p> <p>This module modifies these bits in the following cases.</p> <ul style="list-style-type: none"> • When the selected pipe is in the receive direction and the software sets the SHTNAK bit of the selected pipe to 1, this module indicates PID = NAK when this module recognizes the transfer end. • When this module receives data with a payload exceeding the MaxPacketSize value for the pipe, this module indicates PID = STALL (11). • When this module detects a USB bus reset, this module indicates PID = NAK. <p>Set these bits as follows:</p> <ul style="list-style-type: none"> • To change the state from NAK (00) to STALL, write B'10. • To change the state from BUF (01) to STALL, write B'11. • To change the state from STALL (11) to NAK, write B'10 then B'00. • To change the state from STALL to BUF, set these bits to NAK and then to BUF. |

- Notes:
1. Only reading 0 and writing 1 are enabled.
 2. Set the ACLRM, SQCLR, or SQSET bit when PID = NAK, and the selected pipe is not set in the CURPIPE bits.
 3. When setting the ACLRM, SQCLR, or SQSET bit after changing the PID bits from BUF to NAK, check that PBUSY = 0 for the pipe. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.

Table 61.13 Data Cleared by This Module when ACLRM = 1

| No. | Data Cleared by the ACLRM Bit | Clearing Timing |
|-----|--|--|
| 1 | All data in the FIFO buffer allocated to the selected pipe | — |
| 2 | Not Support | — |
| 3 | Internal flags related to the BFRE bit | When changing the BFRE setting |
| 4 | Internal flags related to the transaction count | When forcibly terminating the transaction count function |

61.2.33 Pipe n Transaction Counter Enable Register (PIPEnTRE) (n = 1 to 5, 9 to F)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

PIPEnTRE enables or disables the transaction counter function and clears the counter for pipes 1 to 5, 9 to F.

This register is initialized by a power-on reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-------|-------|---|---|---|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | TRENb | TRCLR | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | TRENb | B'0 | R/W | Transaction Counter Enable Enables or disables the transaction counter function. 0: The transaction counter function is disabled. 1: The transaction counter function is enabled. When the software sets the number of total packets in the TRNCNT bits in PIPEnTRN for the receive pipe and then sets this bit to 1, this module controls the following when it received the same number of packets as the setting of the TRNCNT bits. When continuous communication mode is used (CNTMD = 1), this module toggles the received data to the CPU even if the FIFO buffer is not full at the end of reception. When SHTNAK = 1, this module changes the PID bits for the corresponding pipe to NAK when this module received the same number of packets as the setting of the TRNCNT bits. When BFRE = 1, this module asserts the BRDY interrupt when this module received the same number of packets as the setting of the TRNCNT bits and read the receive data completely. For transmit pipes, set this bit to 0. When the transaction counter function is not used, set this bit to 0. When using the transaction counter function, set the TRNCNT bits and then set this bit to 1. Furthermore, set this bit to 1 before receiving the first packet that is included in the transaction count range. |
| 8 | TRCLR | B'0 | R/W | Transaction Counter Clear Clears the current count value of the transaction counter for the pipe and indicates TRCLR = 0. 0: Invalid 1: The current count value is cleared. |
| 7 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

Note: Change the bits in this register when PID = NAK.

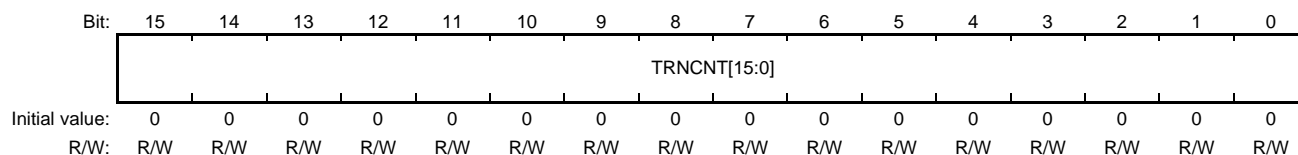
When changing the bits in this register after changing the PID bits for the pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.

61.2.34 Pipe n Transaction Counter Register (PIPEnTRN) (n = 1 to 5, 9 to F)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

PIPEnTRN is a transaction counter for pipes 1 to 5, 9 to F.

This register is initialized by a power-on reset, but the setting of this register is retained through a USB bus reset.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 15 to 0 | TRNCNT [15:0] | All 0 | R/W | <p>Transaction Counter</p> <p>Writing:</p> <p style="padding-left: 20px;">Specify the number of transactions of DMA transfer.</p> <p>Reading:</p> <p style="padding-left: 20px;">When TRENb = 0, the set number of transactions is indicated.</p> <p style="padding-left: 20px;">When TRENb = 1, the number of current transactions is indicated.</p> <p>This module increments (+1) the value of these bits when the following conditions are all satisfied.</p> <p>TRENb = 1</p> <p>TRCNT value ≠ current count value +1 when a packet is received.</p> <p>The payload of the received packet equals the value of the MXPS bits.</p> <p>This module clears these bits to 0 when any of the following is satisfied.</p> <p>When the following conditions are all satisfied</p> <ul style="list-style-type: none"> — TRENb = 1 — TRCNT value = current count value +1 when a packet is received. — The payload of the received packet equals the value of the MXPS bits. <p>When the following conditions are all satisfied</p> <ul style="list-style-type: none"> — TRENb = 1 — A short packet is received. <p>When the following conditions are all satisfied</p> <ul style="list-style-type: none"> — TRENb = 1 — The software sets the TRCLR bit to 1. <p>For transmit pipes, set these bits to all 0.</p> <p>When the transaction counter function is not used, set these bits to all 0.</p> <p>Change these bits when PID = NAK and TRENb = 0.</p> <p>When setting these bits to 1 after changing the PID bits for the pipe from BUF to NAK, check that PBUSY = 0. However, the software does not have to check the PBUSY bit when this module changed the PID bits to NAK.</p> <p>When changing these bits, set TRCNT = 1 and then set TRENb = 1.</p> |

61.2.35 Low Power Status register (LPSTS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides low power management control.

| | | | | | | | | | | | | | | | | |
|----------------|----|-------|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | SUSPM | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R/W | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 15 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 14 | SUSPM | B'0 | R/W | SuspendM control 0: UTMI suspend mode 1: UTMI normal mode This bit should set to 1 when normal operating. UTMI clock is halted if this bit set to 0. Note: This controller deny register access without as follow registers if this bit set to 0. SYSCFG0 BUSWAIT INTENB0 LPSTS BCCTRL UGCTRL * UGCTRL2 UGSTS * |
| 13 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

Note: * UGCTRL and UGSTS are implemented RZ/G2E.

61.2.36 Battery Charging Control Register (BCCTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides battery charging control signals and status signals for embedded USB PHY with battery charging.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|--------------|------------|---|---|---|--------------|----------------|--------------|--------------|--------------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | PRTBL DET | CHG DET | — | — | — | VDM SRCEN | PRTBL DETEN | VDP SRCEN | CHG DETEN | IDP SRCEN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | PRTBLDET | B'0 | R/W | PRTBLDET status 0: PRTBLDET is low level 1: PRTBLDET is high level |
| 8 | CHGDET | B'0 | R/W | CHGDET status 0: CHGDET is low level 1: CHGDET is high level |
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 4 | VDMSRCEN | B'0 | R/W | VDM_SRC control Peripheral mode: Set this bit to 1 in secondary detection. Note: SYSSFG.DRPD should be set to 0 before setting this bit. Host mode: Set this bit to 1 if connection of the portable device is detected. 0: VDM_SRC disabled 1: VDM_SRC enabled |
| 3 | PRTBLDETE N | B'0 | R/W | PRTBLDET control Peripheral mode: Set this bit to 1 in secondary detection. Note: SYSSFG.DRPD should be set to 0 before setting this bit. Host mode: Set this bit to 1 if connection of the portable device is detected. 0: PRTBLDET disabled 1: PRTBLDET enabled |
| 2 | VDPSRCEN | B'0 | R/W | VDP_SRC control Peripheral mode: Set this bit to 1 in primary detection. Note: SYSSFG.DRPD should be set to 0 before setting this bit. 0: VDP_SRC disabled 1: VDP_SRC enabled |
| 1 | CHGDETEN | B'0 | R/W | CHGDET control Peripheral mode: Set this bit to 1 in primary detection. Note: SYSSFG.DRPD should be set to 0 before setting this bit. 0: CHGDET enabled 1: CHGDET disabled |

| Bit | Bit Name | Initial Value | R/W | Description |
|------------|-----------------|----------------------|------------|--|
| 0 | IDPSRCEN | B'0 | R/W | IDP_SRC control 0: IDP_SRC disabled 1: IDP_SRC enabled |

61.2.37 USB General Control Register (UGCTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

This register provides embedded USB PHY control.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|----|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | CONNECT | — | PLL RESET |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | CONNECT | B'0 | R/W | USB connect control 0: PHY receiver halted 1: PHY receiver enabled |
| 1 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 0 | PLLRESET | B'1 | R/W | PLL Reset 0: PLL reset release 1: PLL reset assert |

61.2.38 USB General Control Register 2 (UGCTRL2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register provides embedded USB PHY control.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|---------|----|----|----|----|---------|-----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | VBUSSEL | — | — | — | — | USB0SEL | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 1*2 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R/W | R | R | R | R | R/W | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 11 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 10 | VBUSSEL | B'1*2 | R/W | VBUS drive control select 1: VBUS drive is controlled by VBCTRL.VBOUT or ADPCTRL/DRVVBUS 0: VBUS drive is controlled by PORTSC1.PP of EHCI/OHCI |
| 9 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5, 4 | USB0SEL | B'11*1 | R/W | USB2.0 Ch.0 Selection B'01: Select EHCI/OHCI host module for USB2.0 ch0 B'10: Select HS USB module for USB2.0 ch0 B'11: Select USB OTG function for USB2.0 ch0 |
| 3 to 0 | — | B'0001 | R | Reserved These bits are always read as B'0001. The write value should always be B'0001. |

Notes: 1. RZ/G2E: B'01
2. RZ/G2E: B'0

61.2.39 USB General Status Register (UGSTS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

This register provides embedded USB PHY status.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|------|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | LOCK | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 9 | — | All 0 | R | Reserved |
| 8 | LOCK | B'0 | R | Embedded USB PHY PLL Lock status 1: Embedded USB PHY Lock completed 0: Embedded USB PHY Lock halted |
| 7 to 0 | — | H'01 | R | Reserved The write value should always be same value. |

61.2.40 USB XTAL/EXTTAL control register (USB20_CLKSET0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

This register provides USB_XTAL/USB_EXTAL control.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-----------|----|---|---|---|---|---|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | INCLK_SEL | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R/W | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 15 to 12 | — | All 0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 11 | INCLK_SEL | B'0 | R/W | USB_XTAL/USB_EXTAL to USB3 path control 0: Connects the crystal oscillator to the USB_EXTAL and USB_XTAL 1: Inputs an external oscillator to the USB_EXTAL |
| 10 to 0 | — | H'001 | R | Reserved The write value should always be same value. |

Note: This register is only in Channel 0.

61.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

61.3.1 System Control and Oscillation Control

This section describes register operations required for the initial settings of this module.

(1) Power control and Initialization

The following is the initialize power on procedure of USB subsystem.

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

1. Supply 3.3 V
2. Release HS-USB and EHCI/OHCI module stop and assert HS-USB and EHCI/OHCI module reset.
3. Set LPSTS.SUSPM to normal mode.
4. Set UGCTRL2.USB0SEL bits to B'10 to select HS USB module.
5. Wait 45 μ s for USB PHY to stabilize, and then start normal operation.

[RZ/G2E]

1. Supply 1.8 V
2. Supply 3.3 V (Note: Do not supply 3.3 V when 1.8 V is not supplied.)
3. Wait 100 μ s for USB PHY power stable.
4. Release HS-USB and EHCI/OHCI module stop and assert HS-USB and EHCI/OHCI module reset.
5. UGCTRL.PLLRESET release
6. UGCTRL2.USB0SEL[1:0] should be set to B'10 for HS-USB.
7. LPSTS.SUSPM set to normal mode.
8. Check PLL Lock status by USB General status register (UGSTS)
9. Starting normal operation after PLL Lock status enabled.
10. UGCTRL.CONNECT bit set to 1

The following is the power off procedure of USB subsystem

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

1. Normal operation finished
2. Module stop enable.
3. Shut off powers.

[RZ/G2E]

1. Normal operation finished
2. UGCTRL.CONNECT bit set to 0
3. PLL disable by UGCTRL.PLLRESET assert
4. Shut off 3.3V (VD331)
5. Shut off 1.8V (VD181) (Note: Do not shut off 1.8 V while 3.3 V is being supplied.)
6. Module stop enable.

(2) Enabling High-Speed Operation

This module can set the USB transmission rate (communication bit rate) by the software.

High-speed operation or full-speed operation is selectable. To enable high-speed operation with this module, set the HSE bit in SYSCFG to 1. When high-speed operation is enabled, this module executes the reset handshake protocol and automatically sets the USB transmission rate. The reset handshake result can be checked by the RHST bits in DVSTCTR.

When high-speed operation is disabled, this module operates only in full speed when the function controller function is selected.

(3) USB Data Bus Resistor Control

Figure 61.1 shows the connection between this module and the USB connector.

This module incorporates a pull-up resistor of the D+ signal. Specify pull-up using the DPRPU bits in SYSCFG.

Furthermore, this module controls the terminating resistors of the D+ and D- signals in high-speed operation, and the output resistors in full-speed operation. This module automatically switches the on-chip resistors after connection to the host controller detecting a reset handshake, suspended state, or resume.

When the DPRPU bit in SYSCFG is set to 0 during communication with the host controller, this module disables the pull-up resistors (or terminating resistors) of the USB data line. Therefore, the USB host can be notified of a disconnection from the device.

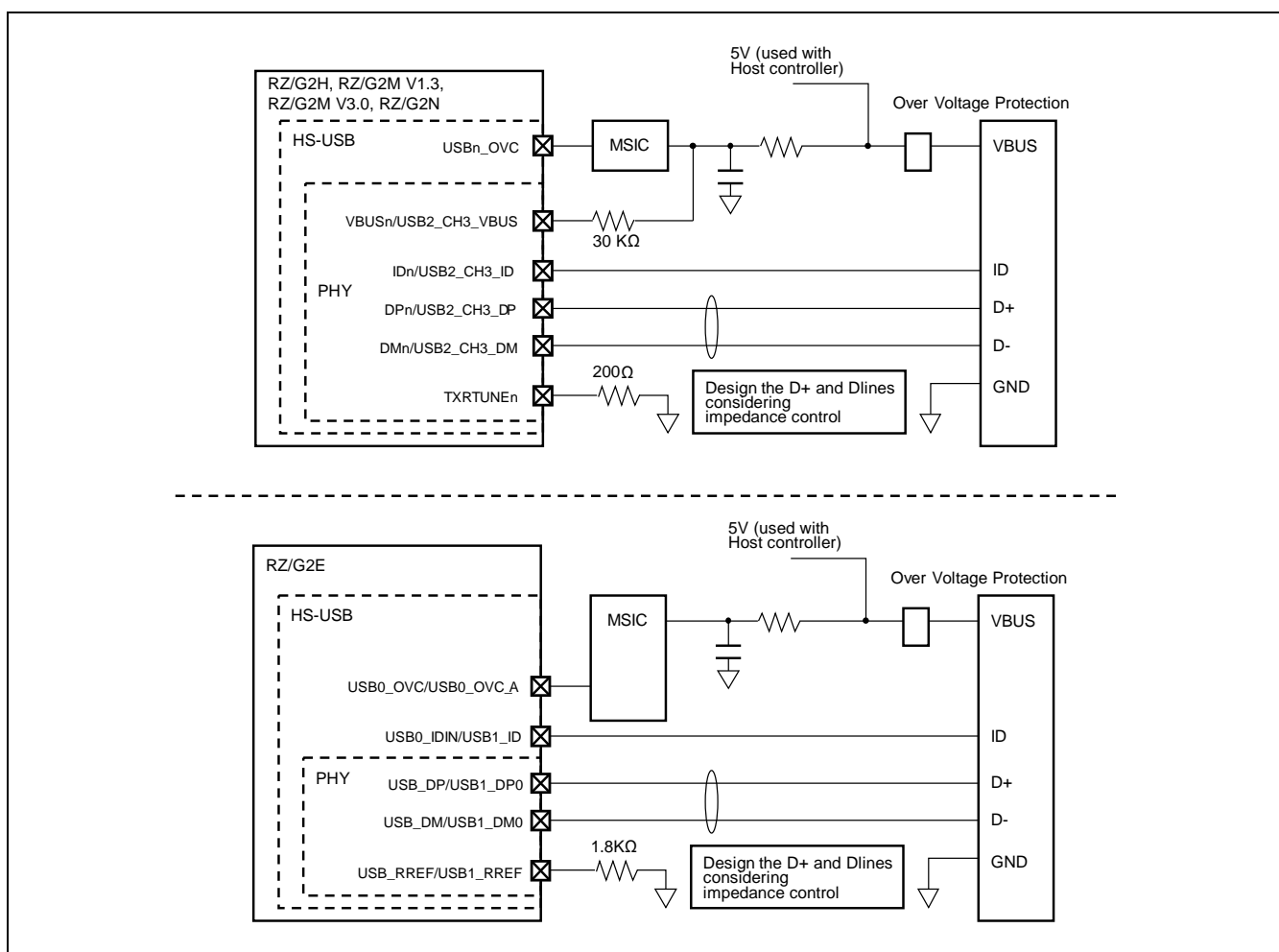


Figure 61.1 Connection to the USB Connector

(4) Software reset when the USB disconnection is detected

Issue a software reset of the USBHS module when the USB disconnection is detected. If the DMA interface is being used, issue a software reset of the USB-DMAC module as well. A software reset can be issued via a register in the CPG

Whether or not the USB disconnection is detected can be determined by the VBSTS in the INTSTS0 register.

Note: USB-PHY might become inoperable to use by the instantaneous interruption of VBUS. As a result S/W on the LSI side (driver) becomes a state of the connection, and USB-PHY becomes a state of power cutoff. It's possible to cancel this state by putting the next way into effect by software.

- 1) When detecting a bus reset or cutoff of VBUS, refer to the state bit of USB - PHY (USB_OFF bit of USBCR2 register in GPIO) and in case of USB_OFF = 1, set USB_START bit.
- 2) When 1) is performed and USB - PHY is started, USB_PHY_ON interrupt occurs, so perform a soft reset to USBHS,USB-DMAC and do the setting by which USBHS,USB-DMAC is initialization and DP pull up --, etc. by this timing.

61.3.2 Interrupt Function

Table 61.14 lists the interrupt generating conditions of this module.

When any of the following interrupt generating conditions are satisfied and the interrupt output is enabled by the corresponding interrupt enable register, this module outputs a USB interrupt request to the interrupt controller (INTC).

Table 61.14 Interrupt Generating Conditions

| Bit | Interrupt Name | Interrupt Generating Conditions | Related Status |
|-------|---|---|------------------|
| VBINT | VBUS interrupt | When VBUS input pin state change is detected (both L to H and H to L) | VBSTS |
| RESM | Resume interrupt | When a USB bus state change is detected in the suspended state (J-State to K-State or J-State to SE0) | — |
| SOFR | Frame number update interrupt | SOFRM = 0: When an SOF packet with a different frame number is received SOFRM = 1: When an SOF packet with a μ frame number of 0 cannot be received due to damage, etc. | — |
| DVST | Device state transition interrupt | When a device state transition is detected USB bus reset detected Suspended state detected SET_ADDRESS request received SET_CONFIGURATION request received | DVSQ |
| CTRT | Control transfer stage transition interrupt | When a control transfer stage transition is detected Setup stage completed Control write transfer status stage shifted Control read transfer status stage shifted Control transfer completed Control transfer sequence error occurred | CTSQ |
| BEMP | Buffer empty interrupt | When the buffer becomes empty after sending all buffer memory data When a packet with a size exceeding the maximum packet size is received | BEMPSTS.PIPEBEMP |
| NRDY | Buffer not ready interrupt | When NAK is sent in response to IN token, OUT token, or PING token When a CRC error or bit stuffing error occurred during data reception in isochronous transfer When an overrun or underrun error occurred during data reception in isochronous transfer | NRDYSTS.PIPENRDY |
| BRDY | Buffer ready interrupt | When the buffer becomes ready for reading or writing | BRDYSTS.PIPEBRDY |
| OVRCR | OVRCR interrupt | When an OVC input pin state change is detected | OVCMON |

Note: These bits in this table are those of INTSTS0 if any register's name is not indicated.

Figure 61.2 shows a block diagram of the interrupt circuit of this module.

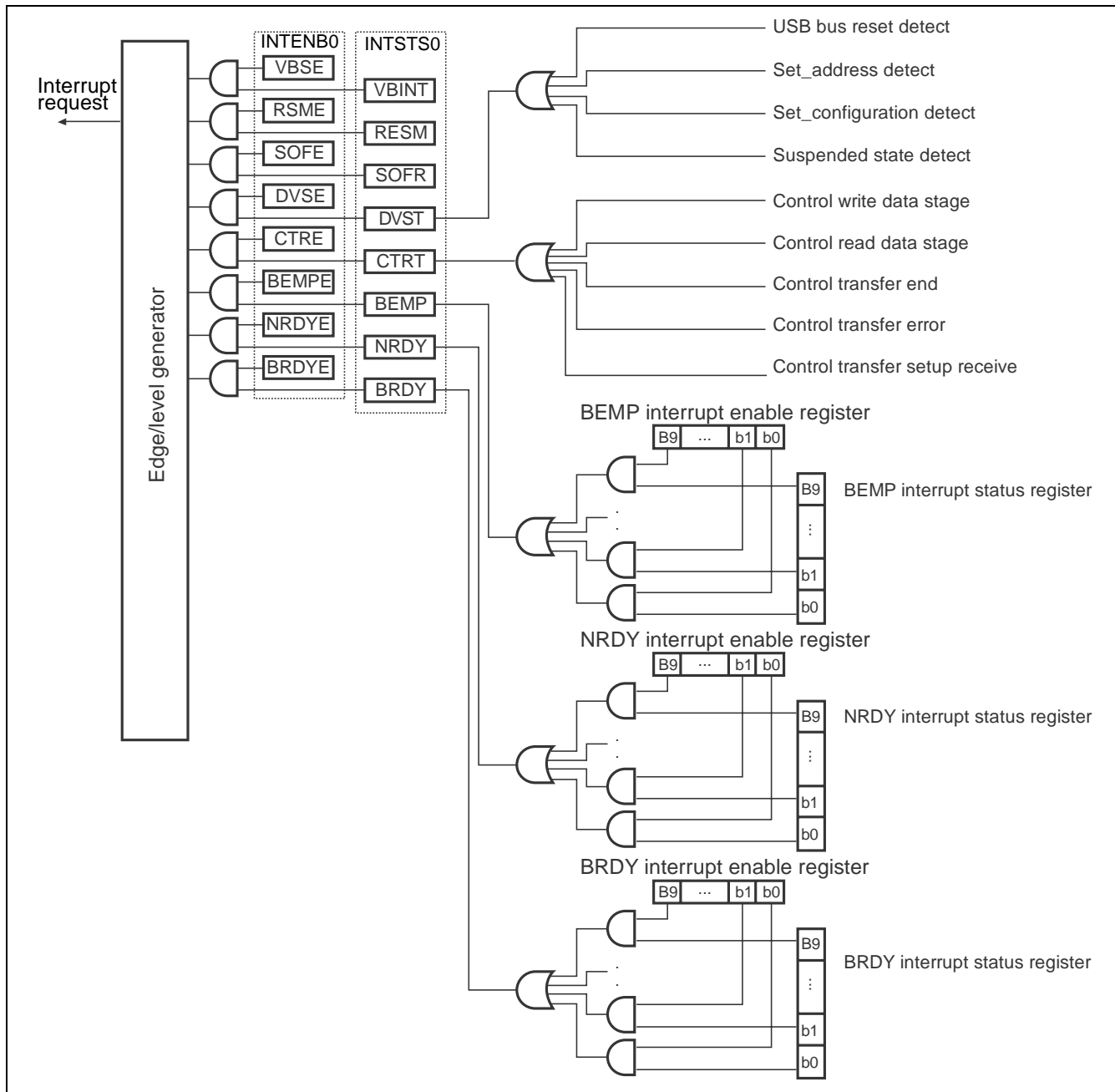


Figure 61.2 Block Diagram of Interrupt Circuit

(1) BRDY Interrupt

A BRDY interrupt can be generated. When the following conditions in each pipe are satisfied, this module sets the corresponding bit in BRDYSTS to 1. At this time, when the PIPEBRDYE bit for the pipe in BRDYENB is set to 1 and the BRDYE bit in INTENB0 is set to 1 by the software, this module generates a BRDY interrupt.

BRDY interrupt generating conditions and clearing method depend on the settings of the BRDYM bit and the BFRE bit for each pipe.

(a) When BRDY interrupt disable (BRDYM = 0) and BRDY interrupt output enable (BFRE = 0) at data reception/transmission are specified

With this setting, a BRDY interrupt indicates that the FIFO port is accessible.

This module generates an internal BRDY interrupt request trigger when the following conditions are satisfied, and sets the PIPEBRDY bit corresponding to the request trigger pipe to 1.

1. Pipes set for the transmit direction

- When the software changes the DIR bit from 0 to 1
- When this module finishes sending a packet of the pipe while data write from the CPU to the FIFO buffer allocated to the pipe is disabled (BSTS = 0)
- When continuous communication mode is selected, this module generates a BRDY interrupt request trigger upon completion of sending data for one side of the FIFO buffer.
- When a double FIFO buffer is specified and one side of the FIFO buffer is empty when data write to the other side of the FIFO buffer is completed.
Even if one side of the FIFO buffer becomes empty while the other side is in data write processing, no BRDY interrupt is generated until the ongoing data write finishes.
- When this module generates a buffer flush for a pipe of isochronous transfer type
- When the FIFO buffer becomes write enable state by writing 1 to the ACLRM bit

No BRDY interrupt request trigger is generated for the DCP (in data transmission of control transfer).

2. Pipes set for the receive direction

- When this module receives a packet successfully and the FIFO buffer becomes ready for reading while data read access from the CPU to the FIFO buffer allocated to the pipe is disabled (BSTS = 0)
- No BRDY interrupt request trigger is generated in transactions with incorrect data PID.
- When continuous communication mode is selected, no BRDY interrupt request trigger is generated for packets with a MaxPacketSize data size and when the FIFO buffer still has available space.
- When this module receives a short packet, this module generates a BRDY interrupt request trigger even if the buffer has available space.
- When a transaction counter is used, this module generates a BRDY interrupt request trigger when this module receives a packet with a size of the set value even if the FIFO buffer still has available space.
- When a double FIFO buffer is specified and one side of the FIFO buffer is ready for reading when data read from the other side of the FIFO buffer is completed
Even if data read for one side of the FIFO buffer is completed while the other side is undergoing data read, no BRDY interrupt request trigger is generated until the ongoing data read finishes.

This interrupt is not generated in communication in the control transfer status stage.

The software can clear the PIPEBRDY interrupt status of the pipe by writing 0 to the PIPEBRDY bit for the pipe in BRDYSTS. At this time, write 1 to the bits corresponding to other pipes.

Be sure to clear the BRDY interrupt status before accessing the FIFO buffer.

(b) When BRDY interrupt disable (BRDYM = 0) and BRDY interrupt output enable (BFRE = 1) at the end of data read are specified

With this setting, this module determines that a BRDY interrupt is generated upon completion of reading all data of one transfer in a receive pipe, and sets the bit for the pipe in this register to 1.

This module determines that the last data of a transfer was received in either of the following cases.

- A Zero-Length packet or another short packet is received.
- A transaction counter (TRNCNT bits) is used and a packet with a size of the value of the TRNCNT bits is received.

When either of these conditions is satisfied and the data read is completed, this module determines that reading all data for one transfer is completed.

When this module receives a Zero-Length packet with the FIFO buffer empty, this module determines that reading all data for one transfer is completed when the Zero-Length packet data is toggled to the CPU. In this case, write 1 to the BCLR bit in the corresponding FIFOCTR register by the software to start the next transfer.

With this setting, this module generates no BRDY interrupt for transmit pipes.

The software can clear the PIPEBRDY interrupt status of the pipe to 0 by writing 0 to the PIPEBRDY bit for the pipe. At this time, write 1 to the bits corresponding to other pipes.

When using this mode, do not change the BFRE setting until the processing for one transfer is completed.

To change the BFRE bit during processing, clear all FIFO buffers for the pipe by the ACLRM bit.

(c) When BRDY interrupt enable (BRDYM = 1) and BRDY interrupt output enable (BFRE = 0) at data reception/transmission are specified

With this setting, the value of the PIPEBRDY bit varies with the value of the BSTS bit for each pipe. That is, this module indicates 1 or 0 of the BRDY interrupt status according to the FIFO buffer status.

1. Pipes set for the transmit direction

When the FIFO port is ready to write data, the PIPEBRDY bit indicates 1. When the FIFO port is not ready to write data, the PIPEBRDY bit indicates 0.

However, even if the transmit pipe of the DCP is ready to write data, no BRDY interrupt is asserted.

2. Pipes set for the receive direction

When the FIFO port is ready to read data, the PIPEBRDY bit indicates 1. When all data is read from the FIFO port (reading disabled), the PIPEBRDY bit indicates 0.

When this module receives a Zero-Length packet with the FIFO buffer empty, this module indicates 1 with the corresponding bit until 1 is written to BCLR by the software, and continues to assert the BRDY interrupt.

With this setting, the software cannot clear the PIPEBRDY bit to 0.

When BRDYM = 1, set the BFRE bit to 0 for all pipes.

Figure 61.3 shows the BRDY interrupt generation timing.

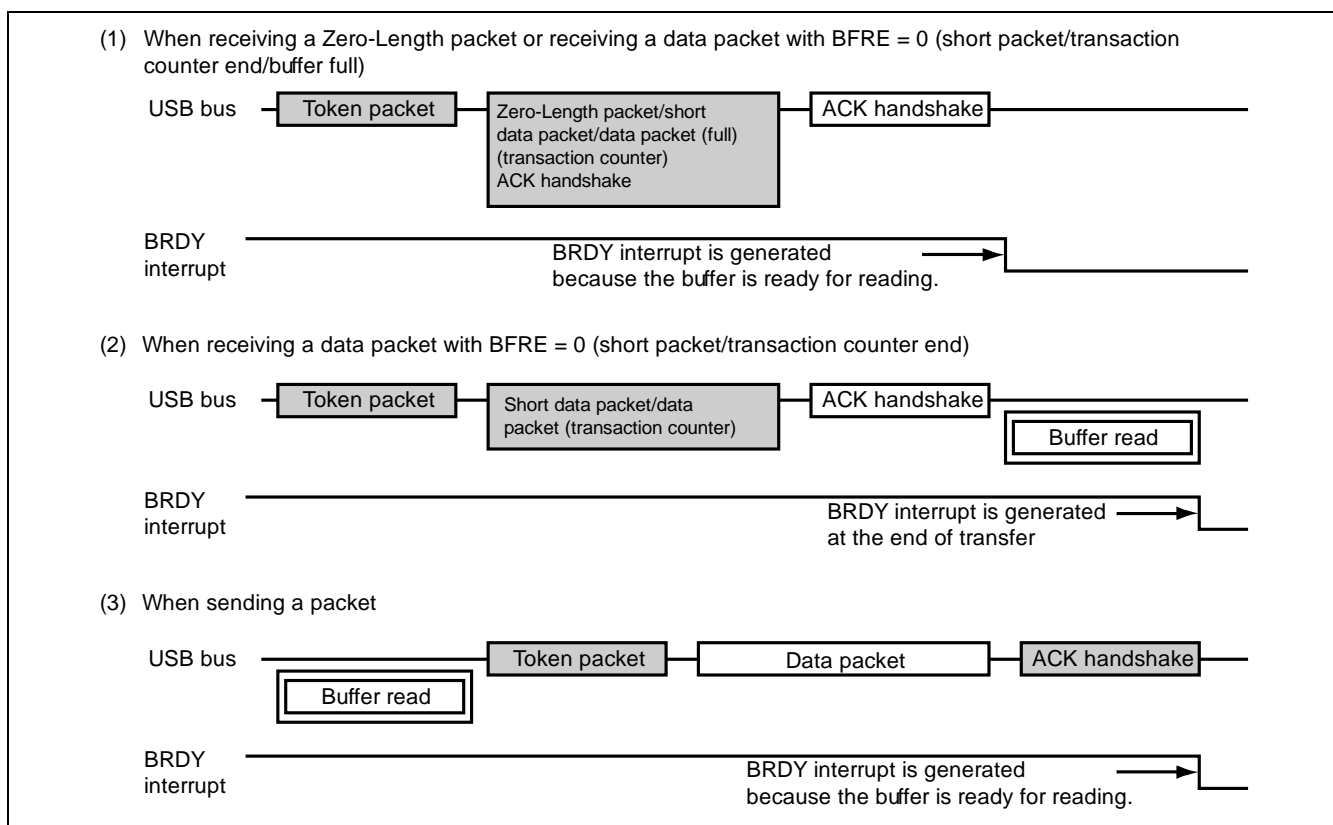


Figure 61.3 BRDY Interrupt Generation Timing

(2) NRDY Interrupt

When this module generates an internal NRDY interrupt request for the pipe in which PID = BUF is set by the software, this module indicates PIPENRDY = 1 for the pipe in NRDYSTS. At this time, when 1 is set in the corresponding bit in NRDYENB by the software, this module indicates NRDY = 1 in INTSTS0 and generates a USB interrupt.

The following shows the conditions for this module to generate an internal NRDY interrupt request for a pipe.

Furthermore, when executing the control transfer status stage, this module generates no interrupt request.

1. Pipes in the transmit direction

- When this module receives an IN token with no transmit data in the FIFO buffer
This module generates an NRDY interrupt request and indicates PIPENRDY = 1.
- When the pipe in which the interrupt is generated is isochronous transfer type, this module sends a Zero-Length packet and indicates OVRN = 1.

2. Pipes in the receive direction pipe

- When this module receives an OUT token with the FIFO buffer occupied
- When the pipe in which the interrupt is generated is isochronous transfer type, this module generates an NRDY interrupt request when this module receives an OUT token, and indicates PIPENRDY = 1 and OVRN = 1.
- When the pipe in which the interrupt is generated is not isochronous transfer type, this module generates an NRDY interrupt request when this module sends an NAK handshake response after receiving data following the OUT token, and indicates PIPENRDY = 1.
However, this module generates no NRDY interrupt when resending data (in the case of DATA-PID mismatch) or when an error occurs in a data packet.
- When this module receives a PING token with the FIFO buffer occupied
This module generates an NRDY interrupt request when this module receives a PING token and indicates PIPENRDY = 1.
- When a packet is not received correctly within the interval frame in an isochronous transfer-type pipe
This module generates an NRDY interrupt request when this module receives SOF and indicates PIPENRDY = 1.

Figure 61.4 shows the NRDY interrupt generation timing.

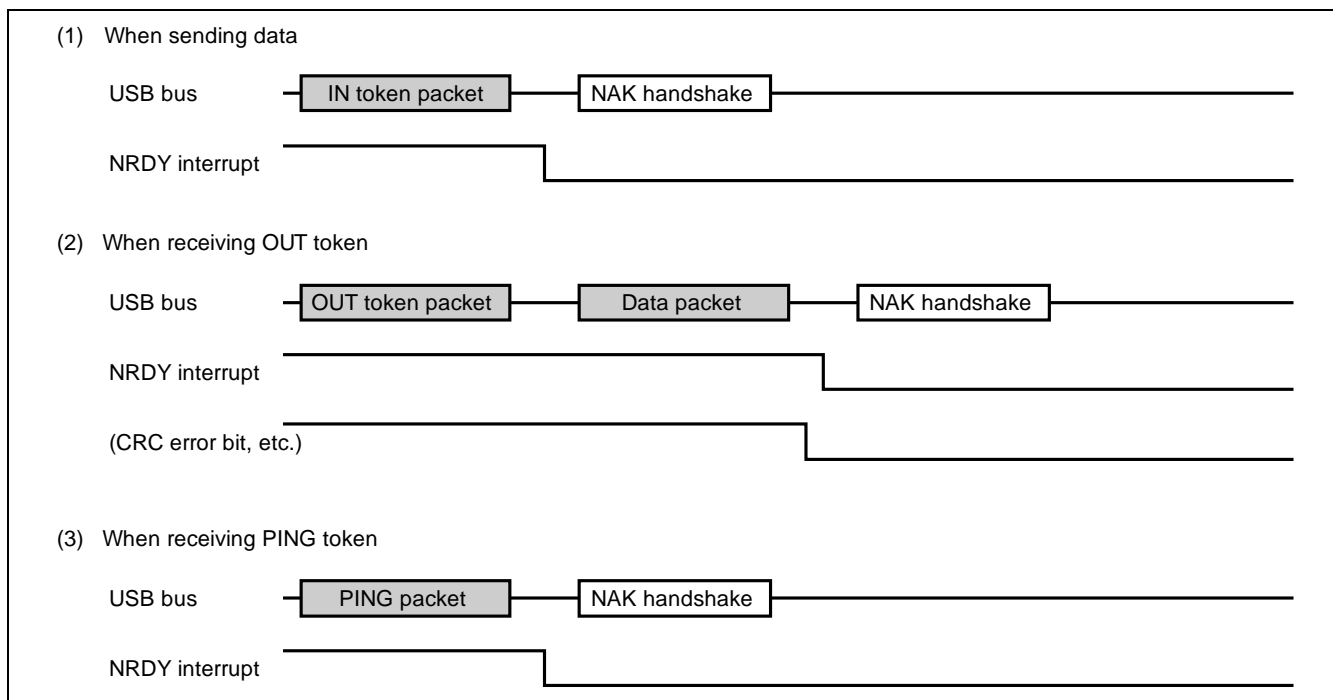


Figure 61.4 NRDY Interrupt Generation Timing

(3) BEMP Interrupt

When this module detects a BEMP interrupt for the pipe in which PID = BUF is set by the software, this module indicates PIPEBEMP = 1 for the pipe in BEMPSTS. At this time, when 1 is set in the corresponding bit in BEMPENB by the software, this module indicates BEMP = 1 in INTSTS0 and generates a USB interrupt.

This module generates an internal BEMP interrupt request in the following cases.

1. Transmit direction pipes

- When the FIFO buffer for the pipe is empty at the end of transmission (including transmission of a Zero-Length packet)
- When single buffer is set for the FIFO buffer, this module generates an internal BEMP interrupt request concurrently with a BRDY interrupt for pipes other than DCP.

However, this module generates no internal BEMP interrupt request in the following cases.

- In the case of double buffer, when the software (DMAC) already started writing data to the FIFO buffer of the CPU at the end of transmission for one-side data
- When the buffer is cleared (empty) by writing 1 to the ACLRM or BCLR bit.
- When sending a Zero-Length packet (IN transfer) of the control transfer status stage

2. Receive direction pipes

- When this module receives a packet with a data size exceeding the MaxPacketSize value
- In this case, this module generates a BEMP interrupt request and indicates the PIPEBEMP = 1 for the pipe, ignores the received data, and changes the PID bits for the pipe to STALL (11).

At this time, this module sends a STALL response.

However, this module generates no BEMP interrupt request in the following cases.

- When this module detects a CRC error or bit stuffing error in the receive data
- When executing the SETUP transaction

Writing 0 to the PIPEBEMP bits clears the status.

Writing 0 to the PIPEBEMP bits has no effect on the operation of this module.

Figure 61.5 shows the BEMP interrupt generation timing.

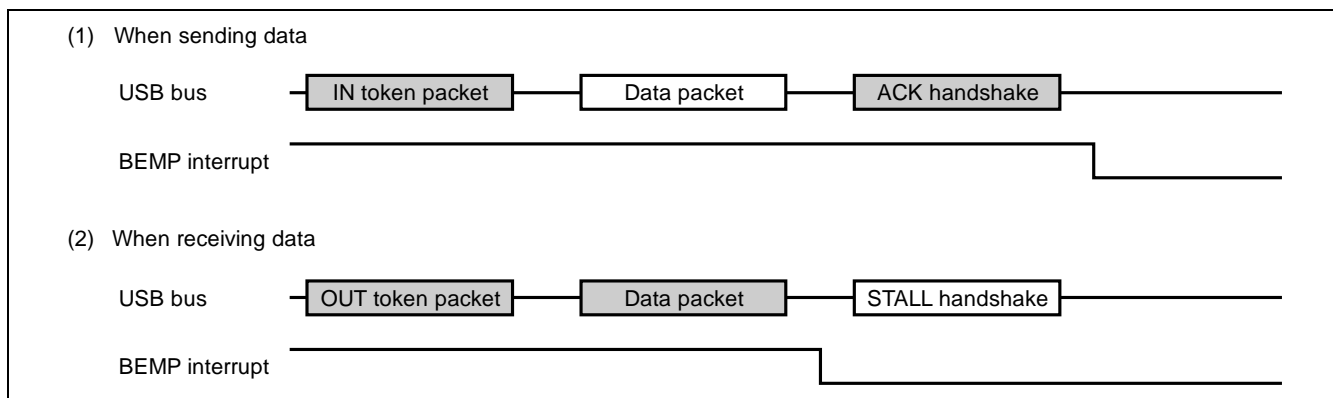


Figure 61.5 BEMP Interrupt Generation Timing

(4) Device State Transition Interrupt

Figure 61.6 shows the device state transitions of this module. This module controls device states and generates device state transition interrupts. However, when returning from the suspended state (detecting the resume signal), this module detects the transition with a resume interrupt. The device state transition interrupt is enabled or disabled by setting INTENB0. The current device state can be monitored using the DVSQ bits in INTSTS0.

When this module shifts to the default state, a device state transition interrupt occurs after executing the reset handshake protocol.

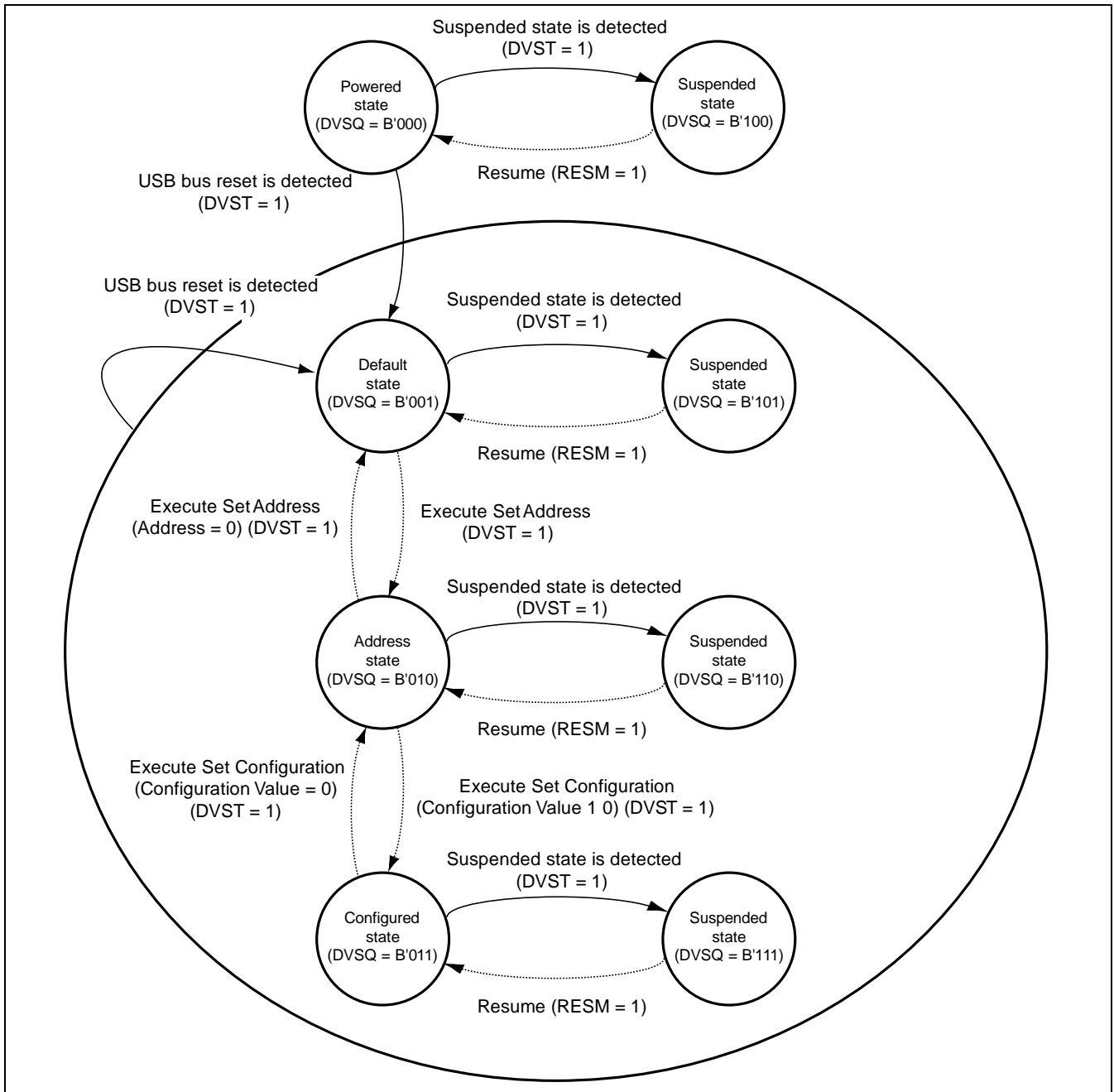


Figure 61.6 Device State Transitions

(5) Control Transfer Stage Transition Interrupt

Figure 61.7 shows control transfer stage transitions of this module. This module manages the sequence of control transfers and generates control transfer stage transition interrupts. Enabling or disabling each control transfer stage transition interrupt can be set by INTENB0. The current transfer stage after transition can be checked by the CTSQ bits in INTSTS0.

The following describes control transfer sequence errors. When an error occurs, the PID bits in DCPCTR indicate B'1x (STALL response).

1. Control read transfer
 - This module receives an OUT or PING token in response to the IN token in the data stage before starting data transfer.
 - This module receives an IN token in the status stage.
 - This module receives a packet with a data packet of DATAPID = DATA0 in the status stage.
2. Control write transfer
 - This module receives an IN token in response to the OUT token in the data stage before sending an ACK response.
 - This module receives a packet with the first data packet of DATAPID = DATA0.
 - This module receives an OUT or PING token in the status stage.
3. Control write no-data control transfer
 - This module receives an OUT or PING token in the status stage.

If the size of the receive data exceeds the wLength value of the USB request in the control write transfer data stage, this is not treated as a control transfer sequence error. This module sends ACK for normal end in response to a packet other than Zero-Length packet in the control read transfer status stage.

When a CTRT interrupt is generated (SERR = 1) due to a sequence error, CTSQ = B'110 is retained until the interrupt status is cleared (CTRT = 0) by the system. For this reason, while CTSQ = B'110, even if a new USB request is received, a CTRT interrupt at the end of the setup stage is not generated. (This module retains the setup stage end status and generates a setup stage end interrupt after the interrupt status is cleared by the software.)

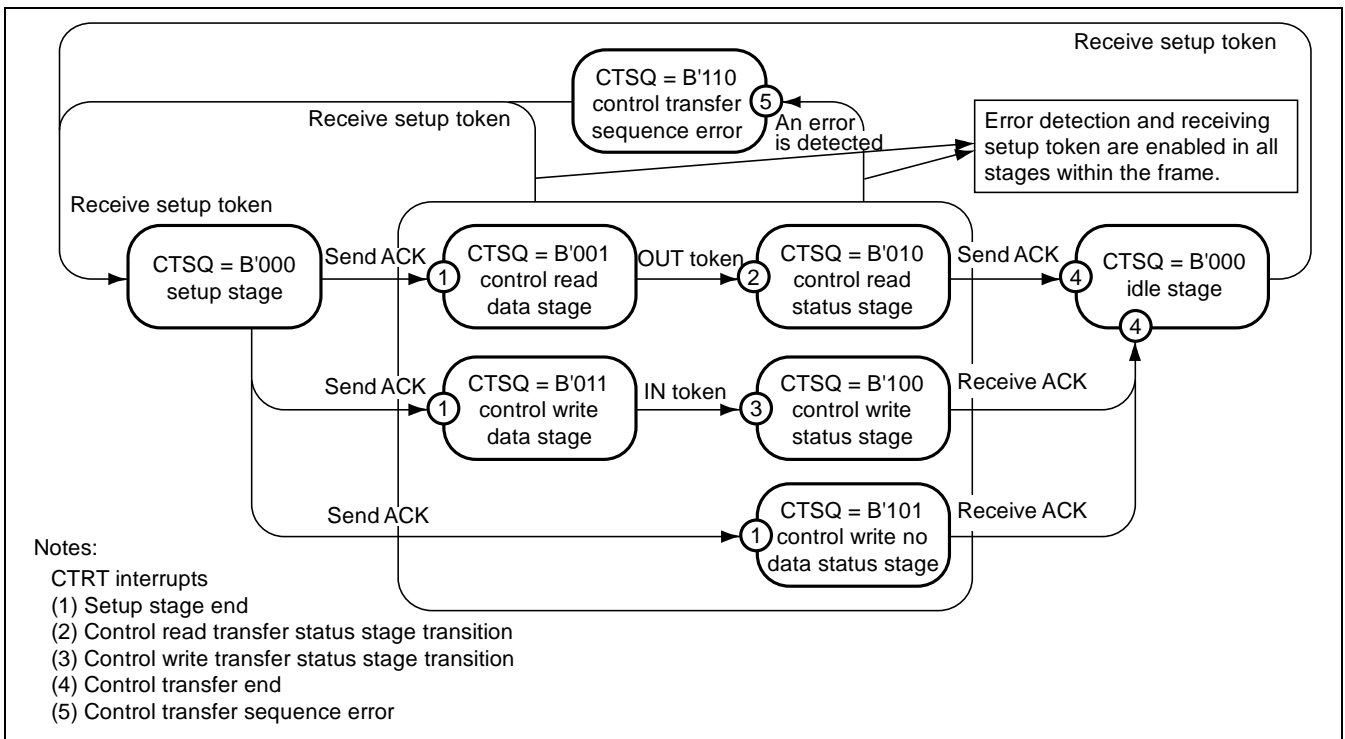


Figure 61.7 Control Transfer Stage Transitions

(6) Frame Update Interrupt

Figure 61.8 shows an example of the SOFR interrupt output timing of this module.

An SOFR interrupt is generated immediately after the frame number is updated.

When this module detects a new SOF packet during the full-speed operation, this module updates the frame number and generates an SOFR interrupt. In the high-speed operation, however, the frame number is not updated or no SOFR interrupt is generated unless the module enters the μ SOF lock state. Furthermore, the SOF interpolation function is not activated. The μ SOF lock state occurs when this module receives a μ SOF packet twice in a row with different frame numbers without an error.

The μ SOF lock monitoring start and stop conditions are as follows:

1. μ SOF lock monitoring start conditions
 USBE = 1
2. μ SOF lock monitoring stop conditions
 USBE = 0, receiving USB bus reset, or detection of suspended state

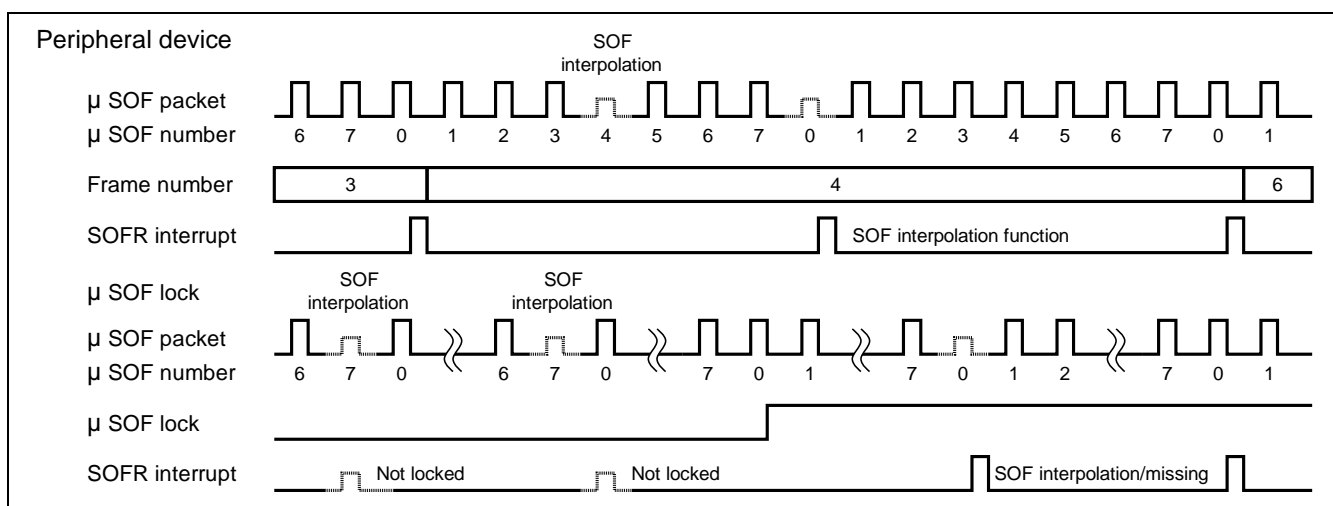


Figure 61.8 Example of SOFR Interrupt Output Timing

(7) VBUS Interrupt

When the VBUSn [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] /OVC signal status [RZ/G2E] pin state changes, a VBUS interrupt is generated. The VBUSn [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] /OVC signal status [RZ/G2E] pin level can be checked by the VBSTS bit in INTSTS0. The VBUS interrupt is used to check connection or disconnection to/from the host controller. However, when the system starts up with the host controller connected, the VBUSn [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] /OVC signal status [RZ/G2E] pin state remains unchanged, and therefore the first VBUS interrupt is not generated.

(8) Resume Interrupt

When the USB bus state changes (J-State to K-State or J-State to SE0) with the device state suspended, a resume interrupt is generated. The resume interrupt is used to detect the return from the suspended state.

When the host controller function is selected, no resume interrupt is generated. Use the BCHG interrupt to detect a change in the USB bus state.

61.3.3 Pipe Control

Table 61.15 lists the pipe settings of this module. In the USB data transfer, data must be transmitted or received using virtual pipes called “endpoint.” This module is provided with ten pipes for data transfer. Set each pipe according to system specifications.

Table 61.15 Pipe Settings

| Register Name | Bit Name | Setting | Remarks |
|---------------------------------|----------|---|--|
| DCPCFG | TYPE | Transfer type | Pipes 1 to F: Setting allowed |
| PIPECFG | BFRE | BRDY interrupt mode | Pipes 1 to 5, B to F: Setting allowed |
| | DBLB | Double buffer | Pipes 1 to 5, B to F: Setting allowed Pipes 9, A: Setting allowed only when bulk transfer is selected |
| | CNTMD | Continuous transfer or discontinuous transfer | Pipes 1 to 2, 9 to A: Setting allowed only when bulk transfer is selected Pipes 3 to 5, B to F: Setting allowed |
| | DIR | Transfer direction | IN or OUT selectable |
| | EPNUM | Endpoint number | Pipes 1 to F: Setting allowed Set a value other than 0000 when using pipes |
| | SHTNAK | Disabling pipes at the end of transfer | Pipes 1 to 2, 9 to A: Setting allowed only when bulk transfer is selected Pipes 3 to 5, B to F: Setting allowed |
| | PIPEBUF | BUFSIZE | Buffer memory size |
| BUFNMB | | Buffer memory number | DCP: Setting disallowed (H'0 to H'3 fixed) Pipes 1 to 5, 9 to F: Setting allowed (H'8 to H'87) Pipes 6 to 8: Setting disallowed (H'4 to H'7 fixed) |
| DCPMAXP PIPEMAXP | MXPS | Maximum packet size | Setting conforming to the USB Specification |
| PIPEPERI | IFIS | Buffer flush | Pipes 1, 2: Setting allowed only when isochronous transfer is selected Pipes 3 to F: Setting disallowed |
| | IITV | Interval counter | Pipes 1, 2: Setting allowed only when isochronous transfer is selected Pipes 3 to F: Setting disallowed |
| DCPCTR PIPE _n CTR | BSTS | Buffer status | For DCP, receive buffer status and transmit buffer status are switched by the ISEL bit. |
| | INBUFM | IN buffer monitor | Provided only for pipes 3 to 5, 9 to F. |
| | ATREPM | Auto-response mode | Pipes 1 to 5, 9 to F Setting allowed. |
| | ACLRM | Buffer auto-clear | Pipes 1 to F Setting allowed |
| | SQCLR | Sequence clear | Data toggle bit clear |
| | SQSET | Sequence setting | Data toggle bit setting |
| | SQMON | Sequence check | Data toggle bit check |
| | PBUSY | Pipe busy check | |
| | PID | Response PID | See section 61.3.3 (6), Response PID. |
| PIPE _n TRE | TRENB | Transaction count enable | Pipes 1 to 5, 9 to F: Setting allowed |
| | TRCLR | Current transaction counter clear | Pipes 1 to 5, 9 to F: Setting allowed |
| PIPE _n TRN | TRNCNT | Transaction counter | Pipes 1 to 5, 9 to F: Setting allowed |

(1) Pipe Control Registers Switching Procedure

The following bits in the pipe control registers can be modified only when USB communication is set to disabled (PID = NAK).

Bits/registers that cannot be modified when USB communication is set to enabled (PID = BUF)

- SQCLR and SQSET bits in DCPCTR
- Each bit in PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI
- ATREPM, ACLRM, SQCLR, and SQSET bits in PIPEXCTR
- Each bit in PIPEXTRE and PIPEXTRN

When changing these bits from the state where USB communication is enabled (PID = BUF), follow the steps below.

1. Generate a pipe control register bit change request.
2. Change the PID bits for the pipe to NAK.
3. Wait until the PBUSY bit for the pipe is cleared to 0.
4. Change the bits in the pipe control registers.

The following bits in the pipe control registers can be modified only for pipe information that is not set in any of the CURPIPE bits in CFIFOSEL, D0FIFOSEL, D1FIFOSEL, D2FIFOSEL and D3FIFOSEL.

Bits/registers that cannot be modified while the CURPIPE bits in FIFO-PORT are being set

- Each bit in DCPCFG and DCPMAXP
- Each bit in PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI

Before changing the pipe information, set the CURPIPE bits to a pipe other than that to be changed. For the DCP, clear the buffer by the BCLR bit after modifying the pipe information.

(2) Transfer Type

Specify the transfer type of each pipe by the TYPE bits in PIPEPCFG as follows:

- DCP: Setting is not required (fixed to control transfer).
- Pipes 1, 2: Set bulk transfer or isochronous transfer.
- Pipes 3 to 5, 9 to F: Set bulk transfer.
- Pipes 6 to 8: Set interrupt transfer.

(3) Endpoint Number

Specify the endpoint number of each pipe by the EPNUM bits in PIPEPCFG. The endpoint number of the DCP is always 0. For other pipes, endpoint 1 to endpoint 15 are selectable.

- DCP: Setting is not required (fixed to endpoint 0).
- Pipes 1 to F Set an endpoint number by selecting a number 1 to 15.

However, set an endpoint number so that combinations of the DIR bit and the EPNUM bits do not overlap.

(4) Maximum Packet Size

Specify the maximum packet size of each pipe by the MXPS bits in DCPMAXP and PIPEMAXP. For the DCP and pipes 1 to 5, 9 to F, any of the maximum packet sizes defined in the USB Specification can be set. For pipes 6 to 8 a maximum packet size up to 64 bytes can be set. Specify a maximum packet size as follows before starting transfer (PID = BUF):

- DCP: Set 64 for high-speed operation.
- DCP: Set 8, 16, 32, or 64 for full-speed operation.
- Pipes 1 to 5, B to F: Set 512 for high-speed bulk transfer.
- Pipes 1 to 5, B to F: Set 8, 16, 32, or 64 for full-speed bulk transfer.
- Pipes 1, 2: Set a value from 1 to 1024 for high-speed isochronous transfer.
- Pipes 1, 2: Set a value from 1 to 1023 for full-speed isochronous transfer.
- Pipes 6 to 8: Set a value from 1 to 64.
- Pipe 9: Set 512 for high-speed bulk transfer.
- Pipe 9: Set 8, 16, 32, or 64 for full-speed bulk transfer.
- Pipe A: Set a value from 1 to 64.

High Bandwidth in interrupt and isochronous transfers is not supported.

(5) Transaction Counter (Pipes 1 to 5, 9 to F: Read Direction)

When the specified number of transactions with data packet in the receive direction are completed, this module recognizes this as a transfer end. The transaction counter works when the pipe selected for the D0FIFOSEL, D1FIFOSEL, D2FIFOSEL and D3FIFOSEL ports is set for the data read (from the buffer memory) direction. The transaction counter is provided with the TRNCNT register to specify the number of transactions and the current counter to count transactions internally. When the current counter reaches the specified value, the buffer memory becomes ready for reading. The current counter of the transaction counter function can be initialized by the TRCLR bit to count transactions again from the beginning. The TRNCNT read data varies depending on the TRENb setting.

- TRENb = 0: The set transaction counter value is read.
- TRENb = 1: The internally counted current counter value is read.
- Change the CURPIPE bits according to the following.
- Do not change the CURPIPE bits until the transaction for the specified pipe finishes.
- Change the CURPIPE bits when the current counter is cleared.
- Control the TRCLR bit according to the following.
- Do not clear the current counter during transaction count and PID = BUF.
- Do not clear the current counter while data is remaining in the buffer.

(6) Response PID

Specify the response PID for each pipe by the PID bits in DCPCTR and PIPEnCTR.

This module operates as follows according to the response PID setting.

The response PID specifies a response to transactions from the host.

- NAK setting: NAK is always sent in response to a transaction.
- BUF setting: A response is sent to a transaction according to the buffer memory status.
- STALL setting: STALL is always sent in response to a transaction.

Note: This module always sends ACK in response to a setup transaction irrespective of the PID setting, and stores a USB request in the register.

This module modifies the PID bits in the following cases depending on transaction results.

- When the response PID is set by the hardware
 - NAK setting: This module sets PID to NAK in the following cases and always sends NAK in response to a transaction.
- When a SETUP token is successfully received (DCP only)
- When the SHTNAK bit in PIPECFG is set to 1 in a bulk transfer and the transaction counter stops counting or this module receives a short packet
 - BUF setting: This module does not set the PID bits to BUF.
 - STALL setting: This module sets PID to STALL in the following cases and always sends STALL in response to a transaction.
- When a maximum packet size exceeded error is detected in a receive data packet
- When a control transfer sequence error is detected (DCP only)

(7) Data PID Sequence Bit

When data is successfully transferred in the control transfer data stage, bulk transfer, or interrupt transfer, this module automatically toggles the data PID sequence bit. The sequence bit of the data PID transmitted next can be monitored by the SQMON bit in DCPCTR and PIPEnCTR. In a data transmission period, the sequence bit switches at the ACK handshake receive timing. In a data reception period, the sequence bit switches at the ACK handshake transmit timing. The data PID sequence bit can also be changed by the SQCLR and SQSET bits in DCPCTR and PIPEnCTR.

In control transfers when the function controller function is selected, this module automatically sets the sequence bit during a stage transition. DATA0 is indicated at the end of the setup stage, and DATA1 is used for responses in the status stage. This does not require the setting of the sequence bit by the software.

Note that, the software must also set the data PID sequence bit when sending or receiving a ClearFeature request.

The sequence bit cannot be modified by the SQSET bit for isochronous transfer pipes.

(8) Response PID = NAK Function

This module has a function to disable the pipe operation (Response PID = NAK) at the last data packet receive timing in a transfer (this module automatically identifies the last data packet receive timing at the short packet receive timing or using the transaction counter) by setting the SHTNAK bit in PIPECFG to 1.

This function allows reception of data packets in transfer units when the buffer memory is used as a double buffer. Furthermore, when the pipe operation is disabled, the software must set the response PID to BUF to enable the pipe operation.

This function is available only for bulk transfers.

(9) Auto-Response Mode

When the ATREPM bit in PIPEnCTR is set to 1 in a bulk transfer pipe (pipes 1 to 5, 9 to F), this module enters auto-response mode: OUT-NAK mode in OUT transfers (DIR = 0) or null auto-response mode in IN transfers (DIR = 1).

1. OUT-NAK mode

When the ATREPM bit is set to 1 in a bulk OUT transfer pipe, this module sends NAK in response to an OUT token or PING token and outputs an NRDY interrupt. To change response mode from normal mode to OUT-NAK mode, specify OUT-NAK mode with the pipe operation disabled (Response PID = NAK), and then enable the pipe operation (Response PID = BUF). After that, OUT-NAK mode is enabled. However, if an OUT token is received immediately before the pipe operation is disabled, this module receives the data of the token correctly and sends ACK to the host.

To change response mode from OUT-NAK mode to normal mode, reset OUT-NAK mode with the pipe operation disabled (Response PID = NAK), and then enable the pipe operation (Response PID = BUF). In normal mode, this module can receive OUT data and sends ACK in response to a PING token when the buffer is ready for receiving.

2. Null auto-response mode

When the ATREPM bit is set to 1 in a bulk IN transfer pipe, this module continues to send Zero-Length packets. To change response mode from normal mode to null auto-response mode, specify null auto-response mode with the pipe operation disabled (Response PID = NAK), and then enable the pipe operation (Response PID = BUF). After that, null auto-response mode is enabled. However, specify null auto-response mode while the buffer is empty. Verify this by checking INBUFM = 0. Since INBUFM = 1 indicates that the buffer contains data, clear the buffer by setting the ACLRM bit. Do not write data to the buffer from the FIFO port during setting to null auto-response mode.

To change response mode from null auto-response mode to normal mode, delay the pipe operation disabled state (Response PID = NAK) for a time period for sending a Zero-Length packet (full-speed: 10 μ s, high-speed: 3 μ s), and then reset null auto-response mode. Normal mode allows writing from the FIFO port, and also allows sending packets to the host by enabling the pipe operation (Response PID = BUF).

61.3.4 FIFO Buffer Memory

(1) Allocating FIFO Buffer Memory

Figure 61.9 shows an example of the FIFO buffer memory map of this module. The FIFO buffer memory is an area shared by the CPU and this module. There are two FIFO buffer memory states where the access right is given to the system (CPU) or to this module (SIE).

The FIFO buffer memory provides independent areas for each pipe. A memory area is set with the block start number and the number of blocks (BUFNMB and BUFSIZE bits in PIPEBUF) in 64-byte units as one block.

When continuous transfer mode is selected by the CNTMD bit in PIPECFG, be sure to set the BUFSIZE bits to a value in integral multiples of the maximum packet size. When double buffer is selected by the DBLB bit in PIPECFG, two sides of the memory area specified by the BUFSIZE bits in PIPEBUF are provided for the same pipe.

Furthermore, three FIFO ports are used for accesses (data read/write) to the buffer memory. The number of a pipe to be allocated to the FIFO ports is specified by the CURPIPE bits in CFIFOSEL and DnFIFOSEL.

The buffer status of each pipe can be monitored by the BSTS and INBUFM bits in DCPCTR and PIPECTR. The access right to the FIFO ports can be checked by the FRDY bit in CFIFOCTR and DnFIFOCTR.

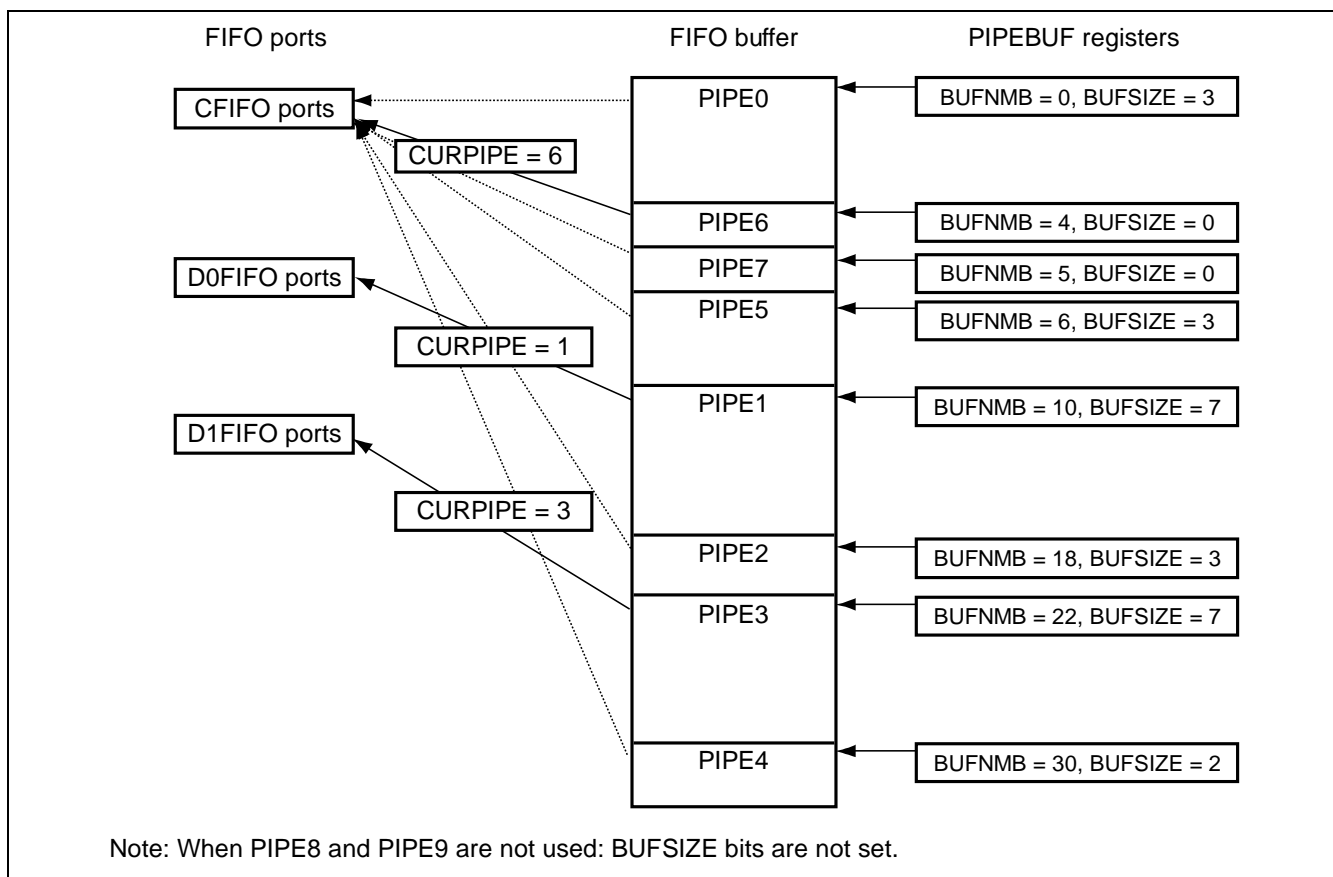


Figure 61.9 Example of Buffer Memory Map

- Buffer status

Tables 61.16 and 61.17 show the buffer memory status of this module. The buffer memory status can be monitored by the BSTS bit in DCPCTR and the INBUFM bit in PIPEnCTR. The buffer memory access direction is specified by the DIR bit in PIPEnCFG or the ISEL bit in CFIFOSEL (when DCP is selected).

The INBUFM bit is valid only for pipes 1 to 5, 9 to F in the transmit direction.

When a transfer pipe in the transmit direction is set to double buffer, the BSTS bit is used to monitor the buffer status on the CPU side, and the INBUFM bit is used to monitor the buffer status on the SIE side. When data write to the FIFO port by the CPU (DMAC) takes time and buffer empty cannot be checked by the BEMP interrupt, the INBUFM bit is available to check completion of transmission.

Table 61.16 Buffer Memory Status That Varies Depending on the BSTS Bit

| ISEL or DIR | BSTS | Buffer Memory Status |
|------------------------|------|--|
| 0 (receive direction) | 0 | No receive data in the buffer memory or data is being received. Data cannot be read from the FIFO port. |
| 0 (receive direction) | 1 | Receive data in the buffer memory or a Zero-Length packet is received. Data can be read from the FIFO port. However, the buffer must be cleared because the FIFO port cannot be read when this module receives a Zero-Length packet. |
| 1 (transmit direction) | 0 | Transmission of data is not completed. Data cannot be written to the FIFO port. |
| 1 (transmit direction) | 1 | Transmission of data is completed. The CPU can write data to the buffer memory. |

Table 61.17 Buffer Memory Status That Varies Depending on the INBUFM Bit

| DIR | INBUFM | Buffer Memory Status |
|------------------------|---------|---|
| 0 (receive direction) | Invalid | Invalid |
| 1 (transmit direction) | 0 | Transmittable data is completely sent. No transmittable data is in the buffer memory. |
| 1 (transmit direction) | 1 | Transmittable data is written from the FIFO port. The buffer memory contains transmittable data. |

- FIFO buffer clear

Table 61.18 shows clearing of the FIFO buffer memory of this module. The buffer memory can be cleared by the BCLR, DCLRM, and ACLRM bits.

Table 61.18 Buffer Clear

| Bit Name | BCLR | DCLRM | ACLRM |
|-----------------|---|--|--|
| Register | CFIFOCTR DnFIFOCTR | DnFIFOSEL | PIPEnCTR |
| Function | Clears the buffer memory of the CPU. | Automatically clears the buffer memory after reading data of the specified pipe. | Automatically clears the buffer memory by ignoring all received packets. |
| Clearing method | Writing 1 to this bit clears the buffer memory. | 1: This mode is enabled. 0: This mode is disabled. | 1: This mode is enabled. 0: This mode is disabled. |

- Buffer area

Table 61.19 shows a buffer memory map of this module. The buffer memory includes a dedicated areas reserved beforehand for pipes and user areas provided for user settings.

The buffer for DCP is a dedicated area commonly used in control read transfers and control write transfers.

Areas for pipes 6 to 8 are allocated respectively in advance. However, when none of pipes 6 to 9 is used, the areas are available for pipes 1 to 5, 9 to F as user areas.

Allocate different areas of the buffer memory to each pipe. Note that, when double buffer is specified, the size of an area is twice of the set value.

Do not set the buffer size to a value smaller than the maximum packet size.

Table 61.19 Buffer Memory Map

| Buffer Memory Number | Buffer Size | Pipe Setting | Remarks |
|----------------------|-----------------|------------------------------------|--|
| H'0 | 64 bytes | Fixed only for DCP | Single buffer |
| H'1 to H'3 | — | Unavailable | — |
| H'4 | 64 bytes | Fixed only for pipe 6 | Single buffer |
| H'5 | 64 bytes | Fixed only for pipe 7 | Single buffer |
| H'6 | 64 bytes | Fixed only for pipe 8 | Single buffer |
| H'7 to H'7F | Max. 7616 bytes | User area for pipes 1 to 5, 9 to F | Double buffer and continuous transfer can be set |

- Buffer auto-clear mode function

When the ACLRM bit PIPEnCTR is set to 1, this module discards all data packets received. However, when this module receives a data packet successfully, it sends ACK to the host controller. This function is available only for the buffer memory read direction.

Writing 1 and then 0 continuously to the ACLRM bit clears the buffer memory of the selected pipe irrespective of the access direction.

Take at least 100 ns between writing 1 and writing 0 for the sequence execution time of the hardware.

- Buffer memory specifications (single buffer/double buffer setting)

Single buffer or double buffer is selectable for pipes 1 to 5, 9 to F with the DBLB bit in PIPEnCFG. Double buffer is a function to provide two sides of the memory area specified by the BUFSIZE bits in PIPEBUF for the same pipe.

Figure 61.10 shows an example of setting for the buffer memory of this module.

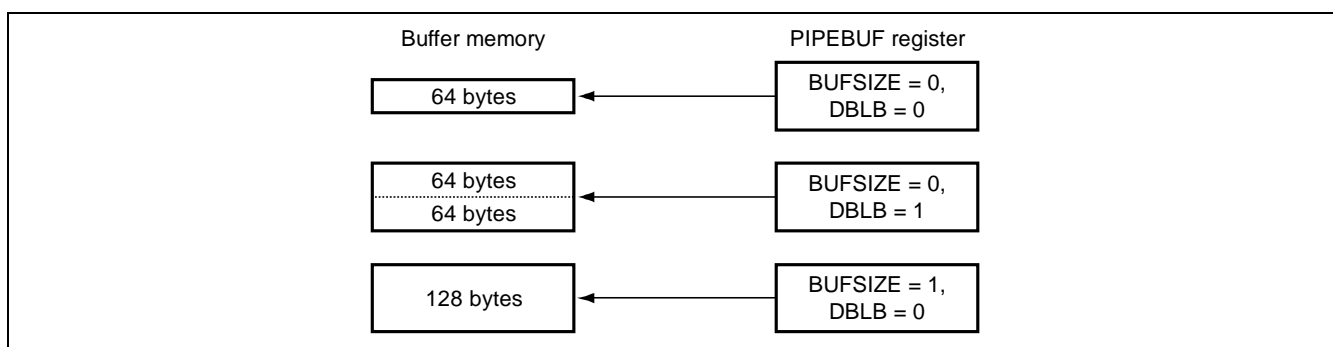


Figure 61.10 Example of Buffer Memory Setting

- Buffer memory operation (continuous transfer setting)

Continuous transfer mode or discontinuous transfer mode is selectable with the CNTMD bit in PIPECFG. This selection is enabled for pipes 1 to 5, 9 to F.

The continuous transfer mode function allows continuous transmission/reception of multiple transactions. When continuous transfer mode is selected, data up to the buffer size allocated to each pipe can be transferred without outputting an interrupt to the CPU.

In continuous transmission mode, data in the buffer is divided by the maximum packet size and is then transmitted. To send data less than the buffer size (a short packet or a packet with a size of integral multiples of the maximum packet size and less than the buffer size), set BVAL = 1 after writing transmit data to the buffer.

In continuous reception mode, no interrupt is generated until this module receives packets to the buffer size, the transaction count ends, or this module receives a short packet.

Table 61.20 shows the relationship between the CNTMD setting and determination of transmission/reception end of the FIFO buffer.

Table 61.20 Relationship between CNTMD Setting and Determination of Transmission/Reception End of FIFO Buffer

| Transfer Mode | Determination of Buffer Readable State and Data Transmittable State |
|---------------------------------------|---|
| Discontinuous transfer (CNTMD = 0) | Conditions for FIFO buffer readable state when data receive direction is selected (DIR = 0): When this module receives one packet |
| | Conditions for FIFO buffer data transmittable state when data transmit direction is selected (DIR = 1): Any of the following cases: The software (or DMAC) writes data of the maximum packet size to the FIFO buffer. The software (or DMAC) writes data of a short packet (including 0 bytes) to the FIFO buffer, and then sets BVAL = 1. |
| Continuous transfer (CNTMD = 1) | Conditions for FIFO buffer readable state when data receive direction is selected (DIR = 0): When the bytes of data received in the FIFO buffer for the selected pipe equals the allocated bytes $((BUFSIZE + 1) \times 64)$ When this module receives a short packet other than a Zero-Length packet When this module receives a Zero-Length packet when data is already stored in the FIFO buffer for the selected pipe When this module receives packets as large as the transaction counter value specified for the selected pipe by the software |
| | Conditions for FIFO buffer data transmittable state when data transmit direction is selected (DIR = 1): Any of the following cases: The volume of data written by the software (or DMAC) equals the size of one side of the FIFO buffer for the selected pipe. The software (or DMAC) writes data less than the size of one side of the FIFO buffer for the selected pipe to the FIFO buffer, and then sets BVAL = 1. |

Figure 61.11 shows an example of buffer memory operation of this module.

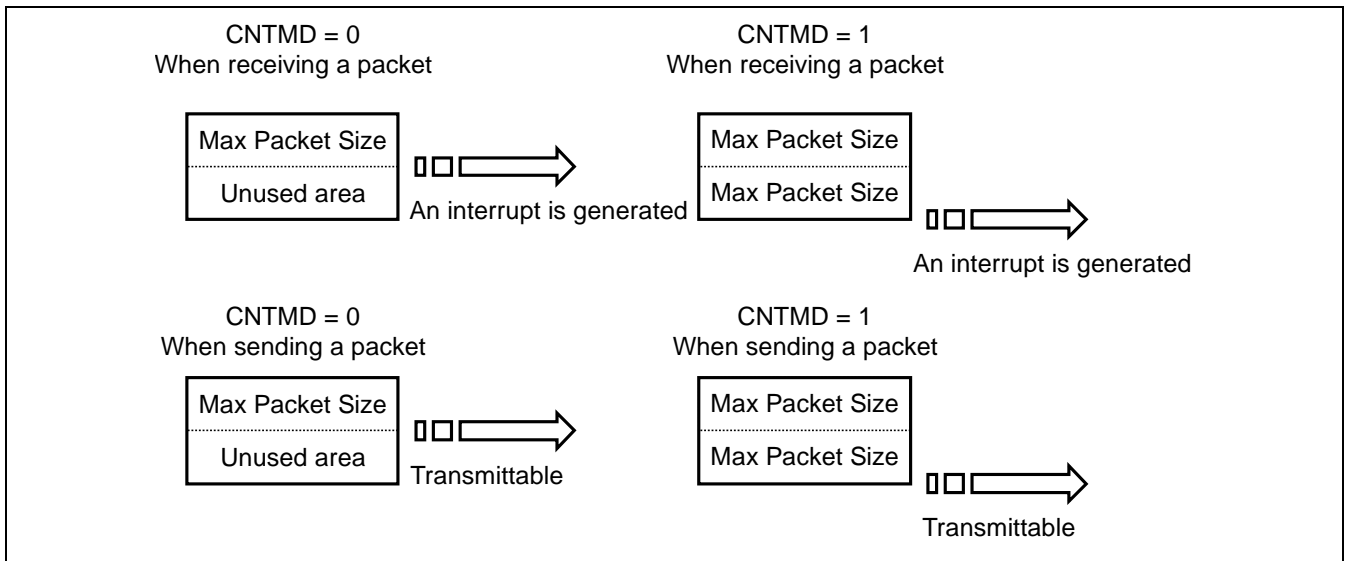


Figure 61.11 Example of Buffer Memory Operation

(2) FIFO Port Function

Table 61.21 lists the settings for the FIFO port functions of this module. During a data write access, when data up to buffer full (or up to the maximum packet size in discontinuous transfer) is written, the FIFO port automatically enters the data transmittable state. To make data less than buffer full (or the maximum packet size in discontinuous transfer) transmittable, set the BVAL bit in CFIFOCTR and DnFIFOCTR to 1 to set write end (DMA transfer: **TEND** signal). For transmission of a Zero-Length packet, clear the buffer using the BCLR bit in these registers and then set the BVAL bit to 1.

During a data read access, when all data is read, the FIFO port automatically enters the new packet receivable state. However, when receiving a Zero-Length packet ($DTLN = 0$), no data can be read. Therefore, clear the buffer using the BCLR bit in these registers. The receive data length can be checked by the DTLN bits in CFIFOCTR and DnFIFOCTR.

Table 61.21 FIFO Port Function Settings

| Register Name | Bit Name | Function | Remarks |
|---------------|----------|--|-------------|
| CFIFOSEL | RCNT | Selects DTLN read mode. | |
| DnFIFOSEL | REW | Rewinds the buffer memory (reread/rewrite). | |
| | DCLRM | Reads and clears receive data of the specified pipe. | DnFIFO only |
| | DREQE | Enables DMA transfer. | DnFIFO only |
| | MBW | Specify FIFO port access bit width. | |
| | BIGEND | Selects FIFO port endian. | |
| | ISEL | Selects FIFO port access direction. | DCP only |
| | CURPIPE | Select the current pipe. | |
| CFIFOCTR | BVAL | Specifies buffer memory write end. | |
| DnFIFOCTR | BCLR | Clears the buffer memory of the CPU. | |
| | DTLN | Check receive data length. | |

(a) FIFO Port Selection

Table 61.22 shows pipes that can be selected in each FIFO port. Select a pipe to be accessed by the CURPIPE bits in CFIFOSEL and DnFIFOSEL. After selecting a pipe, confirm that the set CURPIPE value is read correctly, check FRDY = 1, and then make an access to the FIFO port. (If the previous pipe number is read, this controller is changing a pipe.)

Also select the bus width to be accessed by the MBW bits. The buffer memory access direction is specified by the DIR bit in PIPECFG. However, only the access direction of DCP is determined by the ISEL bit.

Table 61.22 FIFO Port Access

| Pipe | Access | Available port |
|--------------|------------|--|
| DCP | CPU access | CFIFO port register |
| Pipes 1 to F | CPU access | CFIFO port register |
| | DMA access | D0FIFO, D1FIFO, D2FIFO and D3FIFO port |

(b) REW Bit

The current access to a pipe can be suspended temporarily, and it is also possible to make an access to another pipe to continue the ongoing pipe processing. This processing is allowed by using the REW bit in CFIFOSEL and DnFIFOSEL.

When the REW bit is set to 1 together with the CURPIPE bits in CFIFOSEL and DnFIFOSEL and a pipe is selected, the reading or writing pointer of the buffer memory is reset, which allows reading or writing data from the first byte. Furthermore, when a pipe is selected with the REW bit set to 0, data can be read or written following the data at the previous selection without resetting the reading or writing pointer of the buffer memory.

To make accesses to the FIFO port, select a pipe and check FRDY = 1.

(3) DMA Transfer (Dedicated DMA Interface)**(a) Overview of DMA Transfer**

The FIFO port can be accessed by the DMAC for pipes 1 to F. When the buffer of a pipe specified for DMA transfer becomes accessible, this module outputs a DMA transfer request.

Specify the data transfer unit to the FIFO port by the MBW bits in DnFIFOSEL, and select a pipe used for DMA transfer by the CURPIPE bits in DnFIFOSEL. Do not change the selected pipe during a DMA transfer.

(b) DMA Transfer End Auto-Recognition

This module can terminate writing of data by DMA transfer to the FIFO buffer by controlling the DMA transfer end signal input. After sampling the transfer end signal, this module drives the buffer memory into the transmittable state (same state as BVAL = 1).

(c) DnFIFO Auto-Clear Mode (D0FIFO, D1FIFO, D2FIFO and D3FIFO Port Read Direction)

By setting the DCLRM bit in DnFIFOSEL to 1, this module automatically clears the buffer memory for the selected pipe upon completion of data read from the buffer memory.

Table 61.23 shows the relationship between packet reception and buffer memory clear by the software. As shown in the table, buffer clearing conditions vary with the BFRE setting. However, in any state that needs clearing, using the DCLRM bit makes buffer clearing by the software unnecessary. Thus DMA transfer is enabled without requiring any software control.

- This function is available only for the buffer memory read direction.

Table 61.23 Relationship between Packet Reception and Buffer Memory Clear by the Software

| Buffer Status When Receiving a Packet | Register Setting | | | |
|---------------------------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| | DCLRM = 0 | | DCLRM = 1 | |
| | BFRE = 0 | BFRE = 1 | BFRE = 0 | BFRE = 1 |
| Buffer full | Clearing not required | Clearing not required | Clearing not required | Clearing not required |
| Zero-Length packet reception | Clearing required | Clearing required | Clearing not required | Clearing not required |
| Normal short packet reception | Clearing not required | Clearing required | Clearing not required | Clearing not required |
| Transaction count end | Clearing not required | Clearing required | Clearing not required | Clearing not required |

61.3.5 Control Transfer (DCP)

The default control pipe (DCP) is used for data transfer in the control transfer data stage. The buffer memory for the DCP is a 64-byte single buffer of a fixed area used for both control read and control write. The buffer memory can be accessed only by the CFIFO ports.

(1) Setup Stage

1. This module always sends ACK in response to a normal setup packet to this module. The following describes the operation of this module in the setup stage.

When this module receives a new setup packet, this module sets the following bits.

- The VALID bit in INTSTS0 to 1
- The PID bits in DCPCTR to NAK
- The CCPL bit in DCPCTR to 0

2. When this module receives a data packet following a setup packet, this module stores the USB request parameters in USBREQ, USBVAL, USBINDX, and USBLENG.

Be sure to process responses to control transfers while VALID = 0. The PID bits cannot be set to BUF when VALID = 1, and therefore the data stage cannot be completed.

With the VALID bit function, this module can suspend the ongoing request processing if this module receives a new USB request during a control transfer, and send a response to the latest request.

This module also determines the direction bit (bit 8 of bmRequestType) of the received USB request and the request data length (wLength), identifies a control read transfer, control write transfer, and control write no-data transfer, and controls stage transitions. For an incorrect sequence, a sequence error of the control transfer stage transition interrupt occurs, and the error is reported to the software. For the stage control of this module, see Figure 61.7.

(2) Data Stage

Transfer data in response to the received USB request using the DCP. Before making an access to the DCP buffer memory, specify the access direction by the ISEL bit in CFIFOSEL.

When the transfer data size is larger than the DCP buffer memory size, use a BRDY interrupt for control write transfers and a BEMP interrupt for control read transfers.

In the high-speed control write transfer, this module sends responses using the NYET handshake according to the buffer memory status.

(3) Status Stage

Setting the CCPL bit to 1 with the PID bits in DCPCTR set to BUF finishes a control transfer.

After these settings, this module executes the status stage as follows according to the data transfer direction determined in the setup stage:

- Control read transfer
This module sends a Zero-Length packet and receives an ACK response from the USB host.
- Control write transfer, no-data control transfer
This module receives a Zero-Length packet from the USB host and sends an ACK response.

(4) Control Transfer Auto-Response Function

This module automatically responds to a correct SET_ADDRESS request. If any of the following errors occurs in the SET_ADDRESS request, the software must respond to the error.

- Other than control read transfer: bmRequestType ≠ H'00
- Request errors: wIndex ≠ H'00
- Other than no-data control transfer: wLength ≠ H'00
- Request errors: wValue > H'7F
- Control transfer of device state errors: DVSQ = B'011 (configured)

All requests other than SET_ADDRESS require a response made by the software.

61.3.6 Bulk Transfer (Pipes 1 to 5, 9 to F)

The buffer memory usage (single buffer/double buffer setting or continuous/discontinuous transfer mode setting) can be selected for bulk transfers. The buffer memory size can be set to up to 2 Kbytes. This module manages the buffer memory status and automatically responds to a PING packet and an NYET handshake.

(1) NYET Handshake Control

Table 61.24 lists NYET handshake responses of this module. This module sends a NYET response under the following conditions. However, when this module receives a short packet, this module sends ACK instead of NYET. The same is true of the control write transfer data stage.

Table 61.24 NYET Handshake Responses

| Setting of PID Bits in DCPCTR | Buffer Memory Status | Token | Response | Remarks |
|-------------------------------|----------------------|-------------|-----------|--|
| NAK/STALL | — | SETUP | ACK | — |
| | — | IN/OUT/PING | NAK/STALL | — |
| BUF | — | SETUP | ACK | — |
| | RCV-BRDY1 | OUT/PING | ACK | Receives a data packet when receiving an OUT token. |
| | RCV-BRDY2 | OUT | NYET | Receives a data packet. Sends a “reception disable” response. |
| | RCV-BRDY2 | OUT (Short) | ACK | Receives a data packet. Sends a “reception enable” response. |
| | RCV-BRDY2 | PING | ACK | Sends a “reception enable” response. |
| | RCV-NRDY | OUT/PING | NAK | Sends a “reception disable” response. |
| | TRN-BRDY | IN | DATA0/1 | Sends a data packet. |
| TRN-NRDY | IN | NAK | TRN-NRDY | |

[Legend]

- RCV-BRDY1: The buffer memory has available space for two packets or more when receiving an OUT or PING token.
- RCV-BRDY2: The buffer memory has available space for only one packet when receiving an OUT token.
- RCV-NRDY: The buffer memory is occupied when receiving an PING token.
- TRN-BRDY: The buffer memory contains transmit data when receiving an IN token.
- TRN-NRDY: The buffer memory contains no transmit data when receiving an IN token.

61.3.7 Interrupt Transfer (Pipes 6 to 8)

This module performs interrupt transfer following the cycles controlled by the host controller. In interrupt transfers, this module ignores PING packets (no response), sends no NYET handshake, and sends ACK, NAK, and STALL responses.

This module does not support the High-Bandwidth interrupt transfer.

61.3.8 Isochronous Transfer (Pipes 1, 2)

This module is provided with the following functions for isochronous transfers.

- Reporting isochronous transfer error information
- Interval counter (IITV bits)
- Isochronous IN transfer data setup control (IDLY function)
- Isochronous IN transfer buffer flush function (IFIS bit)

This module does not support the isochronous High-Bandwidth transfer.

(1) Isochronous Transfer Error Detection

This module is provided with a function to detect the following error information so that the software can manage isochronous transfer errors. Table 61.25 and 61.26 show the error detection priority of this module and interrupts generated as a result of error detection.

1. PID error
 - When the PID in a receive packet is invalid
2. CRC error, bit stuffing error
 - When a CRC error is found in a receive packet or bit stuffing is invalid
3. Maximum packet size exceeded error
 - When the data size of a receive packet is over the maximum packet size setting
4. Overrun/underrun error
 - When the buffer memory contains no data when this module receives an IN token in the IN direction (transmission) transfer
 - When the buffer memory is occupied when this module receives an OUT token in the OUT direction (reception) transfer
5. Interval error

This module generates an interval error in the following cases.

 - When no IN token can be received in interval frames in the isochronous IN transfer
 - When this module receives an OUT token other than interval frames in the isochronous OUT transfer

Table 61.25 Error Detection when Receiving Tokens

| Priority for Error Detection | Error | Interrupt and Status |
|------------------------------|-------------------------------|--|
| 1 | PID error | This module generates no interrupt (ignores as a damaged packet). |
| 2 | CRC error, bit stuffing error | This module generates no interrupt (ignores as a damaged packet). |
| 3 | Overrun error, underrun error | This module generates an NRDY interrupt and sets the OVRN bit. This module sends a Zero-Length packet in response to the IN token. This module receives no data packet in response to the OUT token. |
| 4 | Interval error | This module generates an NRDY interrupt. |

Table 61.266 Error Detection when Receiving Data Packets

| Priority for Error Detection | Error | Interrupt and Status |
|------------------------------|------------------------------------|--|
| 1 | PID error | This module generates no interrupt (ignores as a damaged packet). |
| 2 | CRC error, bit stuffing error | This module generates an NRDY interrupt and sets the CRCE bit. |
| 3 | Maximum packet size exceeded error | This module generates a BEMP interrupt and sets the PID bits to STALL. |

(2) DATA-PID

This module does not support the High-Bandwidth transfer. The following lists responses to received PIDs when the function controller function is selected.

1. IN direction
 - DATA0: Sent correctly as the PID of data packet.
 - DATA1: Not sent
 - DATA2: Not sent
 - mData: Not sent
2. OUT direction (full-speed operation)
 - DATA0: Received correctly as the PID of a data packet.
 - DATA1: Received correctly as the PID of a data packet.
 - DATA2: The packet is ignored.
 - mData: The packet is ignored.
3. OUT direction (high-speed operation)
 - DATA0: Received correctly as the PID of a data packet.
 - DATA1: Received correctly as the PID of a data packet.
 - DATA2: Received correctly as the PID of a data packet.
 - mData: Received correctly as the PID of a data packet.

(3) Interval Counter

Intervals for isochronous transfers can be set by the IITV bits in PIPEPERI. The functions shown in Table 61.27 are made available with the interval counter.

Table 61.27 Interval Counter Functions

| Transfer Direction | Function | Detection Conditions |
|--------------------|---------------------------------|--|
| IN | Transmit buffer flush | No IN token can be successfully received in the interval frame during the isochronous IN transfer. |
| OUT | Reporting reception of no token | No OUT token can be successfully received in the interval frame during the isochronous OUT transfer. |

Since the interval counter is counted upon receiving an SOF or using the interpolated SOF, the isochronism can be maintained even if the SOF is damaged. The specifiable frame interval is 2^{IITV} frames or $2^{\text{IITV}} \mu$ frames.

(a) Counter initialization

This module initializes the interval counter as follows:

- Power-on reset
The IITV bits are initialized.
- Initializing buffer memory by ACLRM
The IITV bits are not initialized but the interval counter is initialized. Clearing the ACLRM bit to 0 causes the counter to start counting from the set IITV value.
- USB bus reset.

After the interval counter is initialized, the interval counter starts counting in either of the following cases after a packet is correctly transferred.

1. When this module receives an SOF after sending data in response to the IN token with PID = BUF
2. When this module receives an SOF after receiving the OUT token data with PID = BUF

The interval counter is not initialized under the following conditions.

1. When PID = NAK or STALL
The interval timer does not stop. Try to execute the transaction in the next interval.
2. USB bus reset and USB suspended
The IITV bits are not initialized. When this module receives an SOF, the interval counter starts counting from the value before receiving the SOF.

(b) Interval count and transfer control

1. When the selected pipe is Isochronous-OUT transfer pipe
When this module receives no data packet in (μ) frames at the intervals specified by the IITV bits, this controller generates an NRDY interrupt.
When this module cannot receive data due to a CRC error in the data packet, FIFO buffer full, or other reasons, this module also generates an NRDY interrupt.
An NRDY interrupt is generated at the SOF packet receive timing. Even if the SOF packet is damaged, this module generates an NRDY interrupt at the SOF packet receive timing using the internal interpolation function.
When IITV \neq 0, however, this module generates an NRDY interrupt at the SOF packet receive timing in each interval after the interval count starts.

When PID = NAK is set by the software after the interval timer is activated, this module generates no NRDY interrupt even when this module receives an SOF packet.

The interval count start conditions vary with the set IITV value.

- When IITV = 0: The interval count starts from the next (μ) frame where the PID bits for the selected pipe is changed to BUF.

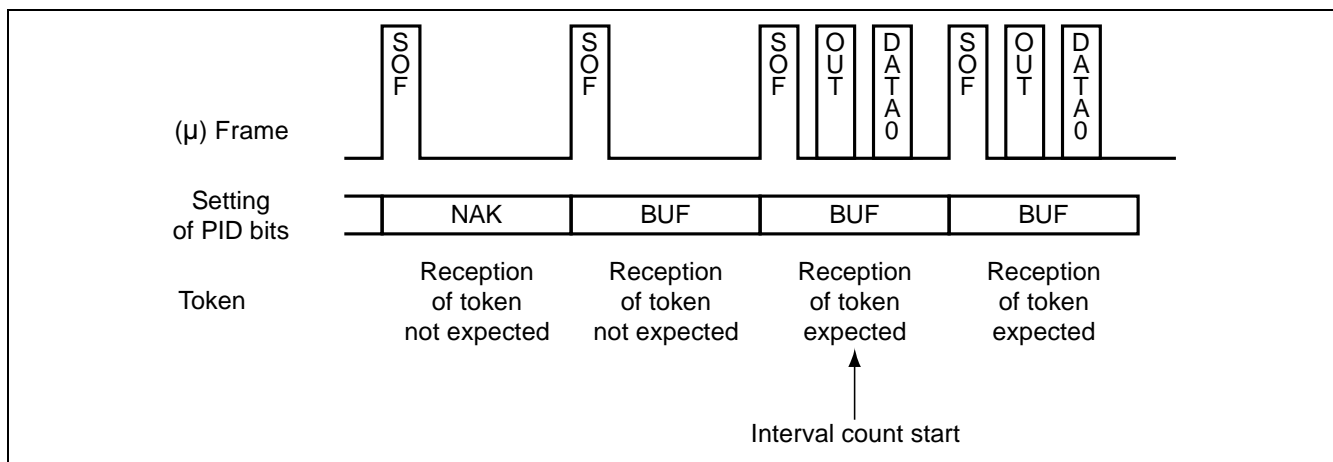


Figure 61.12 (μ) Frames and Expectation of Receiving a Token when IITV = 0

- When IITV \neq 0: The interval count starts from the time when this module successfully receives the first data packet after the PID bits for the selected pipe is changed to BUF.

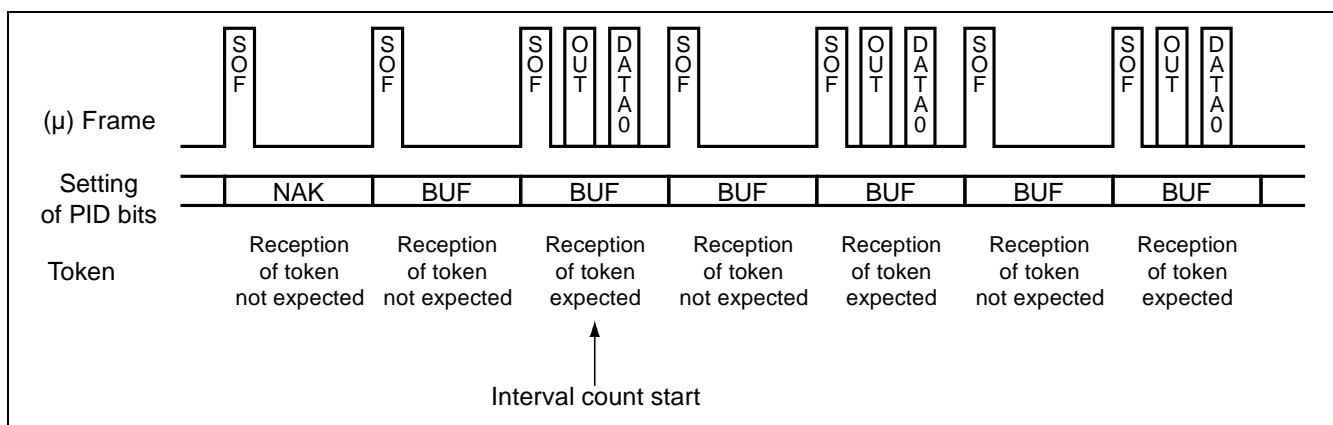


Figure 61.13 (μ) Frames and Expectation of Receiving a Token when IITV \neq 0

2. When the selected pipe is Isochronous-IN transfer pipe

Use the selected pipe by combining with the setting of IFIS = 1. When IFIS = 0, this module sends data packets in response to received tokens irrespective of the IITV value.

When this module receives no IN-Token in (μ) frames at the intervals specified by the IITV bits with transmittable data stored in the FIFO buffer with IFIS = 1, this module clears the FIFO buffer.

When this module cannot receive data due to a CRC error or another bus error in the IN-Token, this module also clears the FIFO buffer.

The FIFO buffer is cleared at the SOF packet receive timing. Even if the SOF packet is damaged, this module clears the FIFO buffer at the SOF packet receive timing using the internal interpolation function.

The interval count start conditions vary with the set IITV value (same as Isochronous-OUT transfer).

When the function controller function is selected, the interval count start conditions are as follows:

- When this module is reset by the hardware (The set IITV value is also cleared to 0.)
- When ACLRM is set to 1 by the software
- When this module detects a USB reset

(4) Transmit Data Setup in Isochronous Transfer

In isochronous data transmission is selected, this module can send a data packet in the next frame after this module detects an SOF packet after writing data to the buffer. This function is referred to as the isochronous transfer transmit data setup function. With this function, the frame that started data transmission can be identified.

When the buffer memory is used as a double buffer, even if writing data to both buffers is completed, only one side that finished writing earlier becomes transferable. For this reason, even if multiple IN tokens are received in the same frame, only one packet of the buffer memory is sent.

When the buffer memory is ready for transmission in the IN token reception cycle, this module sends data and an ACK response. However, when the buffer memory is not ready for transmission, this module sends a Zero-Length packet and generates an underrun error.

Figure 61.14 shows an example of transmission using the isochronous transfer transmit data setup function of this module when IITV = 0 (each frame). Transmission of a Zero-Length packet is indicated as hatched “Null” in the figure.

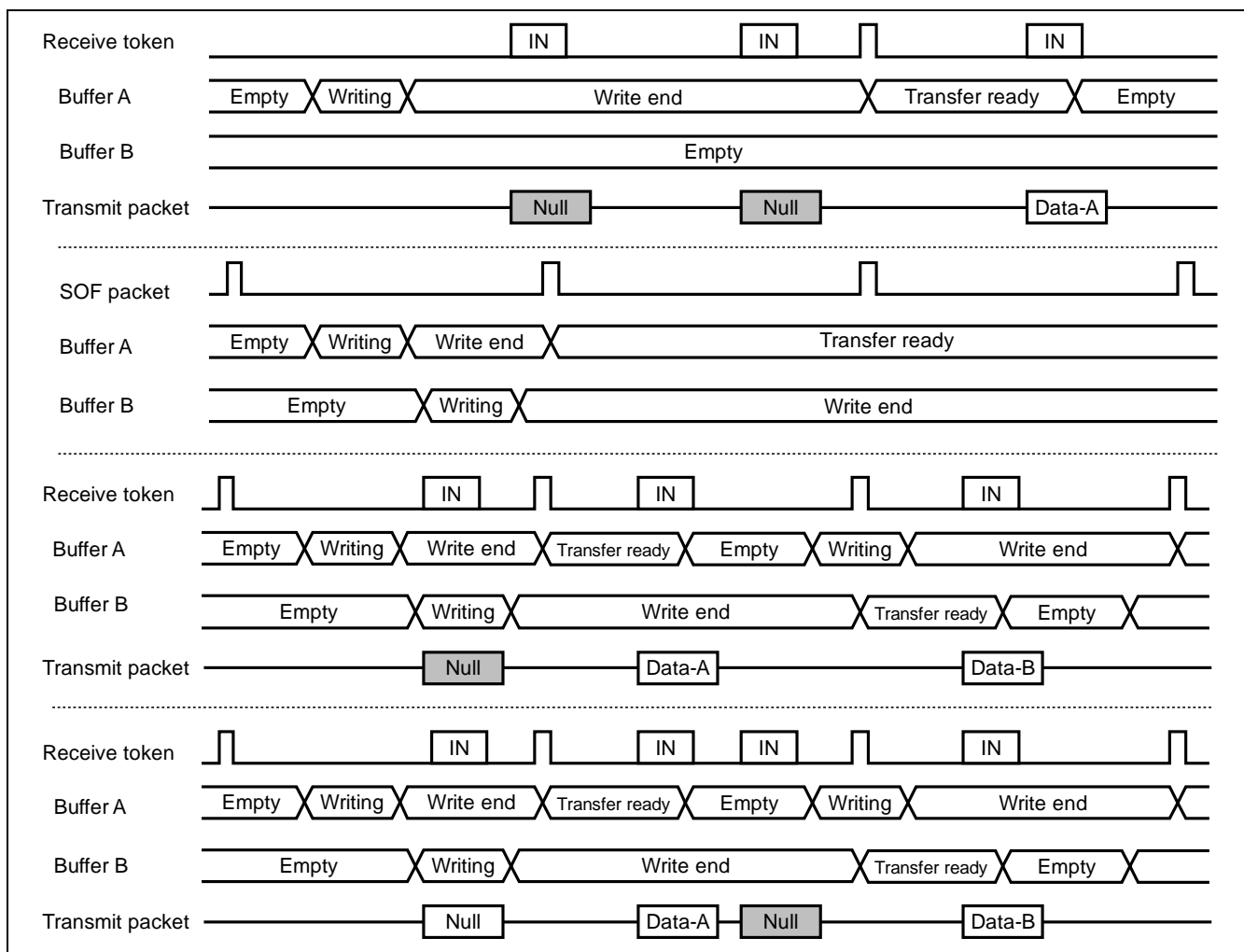


Figure 61.14 Example of Transmit Data Setup Function

(5) Transmit Buffer Flush in Isochronous Transfer

In isochronous data transmission is selected, this module receives no IN token in the interval frame. When this module receives the SOF or μ SOF packet of the next frame, this module regards the IN token as damaged and clears the buffer that is ready for transmission to make it writable.

At this time, when the buffer memory is used as a double buffer and writing data to both buffers is completed, this module regards the discarded buffer memory data as transmitted in the same interval frame, and drives the buffer memory that is not discarded due to the reception of an SOF or μ SOF packet into the transmission ready state.

The buffer flush function activation timing varies depending on the IITV value.

- When IITV = 0
This module activates the buffer flush function from the next frame after the pipe is enabled.
- When IITV \neq 0
This module activates the buffer flush function from the first normal transaction.

Figure 61.15 shows an example of the buffer flush function of this module. For tokens before the interval frame (outside the specified interval), however, this module sends the data stored in the buffer according to the data setup state or sends a Zero-Length packet as an underrun error.

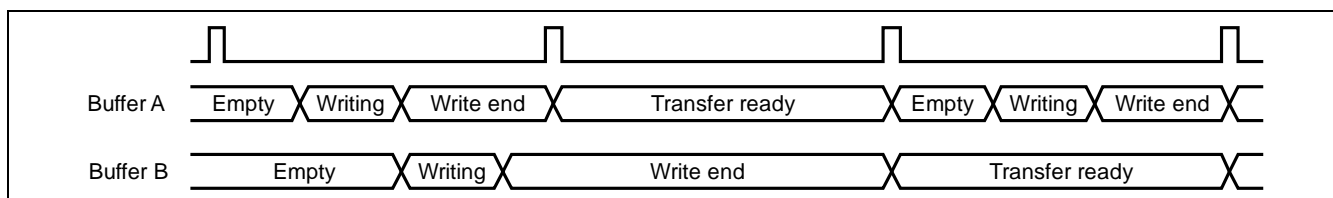


Figure 61.15 Example of Buffer Flush Function

Figure 61.16 shows an example of an interval error of this module. Interval errors are the following five. An interval error occurs at timing [1] in the figure, and the buffer flush function is activated.

When an interval error occurs, the buffer flush function is activated in the IN transfer and an NRDY interrupt occurs in the OUT transfer. Discriminate an NRDY interrupt (including a receive packet error) from an overrun error with the OVRN bit.

This module sends a response to the hatched tokens in the figure according to the buffer memory status.

1. IN direction
 - When the buffer is ready to transmit data, this module sends the data and an ACK response.
 - When the buffer is not ready to transmit data, this module sends a Zero-Length packet and generates an underrun error.
2. OUT direction
 - When the buffer is ready to receive data, this module receives data and sends an ACK response.
 - When the buffer is not ready to receive data, this module discards the data and generates an overrun error.

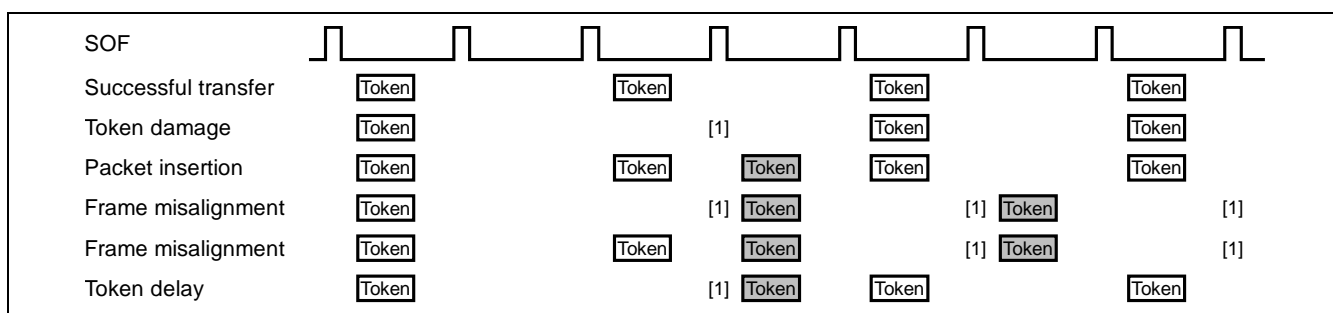


Figure 61.16 Example of Interval Error when IITV = 1

61.3.9 SOF Interpolation Function

In case this module cannot receive SOF packets at intervals of 1 ms (full-speed operation) or 125 μ s (high-speed operation) due to damage or missing of SOF packets, this module interpolates the SOF. The SOF interpolation starts functioning when SYSCFG.USBE = 1, LPSTS.SUSPM= 1, and at the SOF packet receive timing. Furthermore, the interpolation function is initialized in the following cases.

- Power-on reset
- USB bus reset
- When the suspended state is detected

The SOF interpolation functions with the following specifications.

- The frame intervals (125 μ s or 1 ms) conform to the result of the reset handshake protocol.
- The SOF interpolation does not function until this module receives an SOF packet.
- After receiving the first SOF packet, 125 μ s or 1 ms is counted by the internal 48-MHz clock to interpolate the SOF.
- After receiving the second SOF packet or a following packet, the SOF is interpolated using the previous receive intervals.
- The SOF interpolation does not function in the suspended state and during a USB bus reset.
(When this module enters the suspended state in high-speed operation, the SOF interpolation continues for 3 ms from the final packet.)

This module activates the following functions based on the reception of SOF packets. When one or more SOF packets are lost, normal operation can be continued to perform the SOF interpolation.

- Updating frame numbers and μ frame numbers
- SOFR interrupt timing and μ SOF lock
- SOF pulse output
- Isochronous transfer interval count

When one or more SOF packets are lost in full-speed operation, the FRNM bits in FRMNUM are not updated. When one or more μ SOF packets are lost in high-speed operation, the UFRNM bits in the UFRMNUM are updated. However, when a μ SOF packet with UFRNM = B'000 is lost, the FRNM bits are not updated even if following μ SOF packets with UFRNM \neq B'000 are successfully received.

61.4 Guidelines for Designing USB2.0 Hi-Speed Boards

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

61.4.1 USB Transmission Line

The USB transmission line indicates the pattern that connects the USB connector and the USB transceiver with built-in this LSI.

The USB2.0 has three communication modes: Hi-Speed, Full-Speed and Low-Speed modes. The transmission speed in the Hi-Speed mode is 480 Mbps. Therefore, the USB transmission line must be designed as a high-frequency circuit. Impedance control is required for the USB transmission line.

- The characteristic impedance required for the USB Hi-Speed transmission line is differential impedance $90\ \Omega$ ($\pm 15\%$).
- The pattern width and pattern pitch for impedance control vary depending on the board thickness, material and layer configuration. For details, consult the board manufacturer.
- A maximum delay of 1 ns is allowed from the USB connector to this LSI. Therefore it is recommended that the pattern length from the USB device to the USB connector is less than 50 mm and the difference between the pattern lengths for D+ and D- is less than 2.5 mm for a generic PCB.
- The lower layer of the USB transmission line must be a solid ground plane. The ground must be wider than the USB transmission line by 2 mm or more. The power supply for a solid ground plane is DG33.
- Do not allocate other signal lines near to the USB transmission line. In particular, lines carrying widely fluctuating signals, such as clock and data bus lines, must be assigned far from the USB transmission line. Furthermore, layout of the USB transmission lines must be such that they do not cross with other signal lines.
- Locate the USB transmission line on the same layer without passing it through a through-hole. Do not create stubs.
- Place the USB transmission lines so that they are equally spaced from each other.
- Locate the USB transmission line as far as possible from the oscillator, power supply circuit, and other I/O connectors.
- Try to avoid bending the USB transmission line as far as possible. If it is absolutely necessary to bend it, do not bend it at acute or right angles. Bend it moderately, at not more than 45 degrees, or bend it into an arc.
- It is recommended to guard ring the clock, reset, read, write and chip select signals with grounds.

When a resistance is connected to the USB transmission line, locate it near to the USB transmission line. The connecting wire must be as short as possible.

61.4.2 Power Supply and Ground Pattern

Refer to serial interface PCB design guideline.

61.4.3 Oscillation

- USB clock is supplied by the internal PLL.
- Provide an oscillation circuit near the clock input pin for USB (EXTAL). It is recommended to guard ring EXTAL with a ground.
- When using a crystal resonator, consult its manufacturer to determine the circuit constant.

Note: RZ/G2E supply 48MHz reference clock from EXTAL/XTAL instead of USB_EXTAL/USB_XTAL.

61.4.4 VBUS

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

- This HS-USB must not connect directly to the 5V VBUS. The VBUSn pin must be isolated by an $30\text{k}\Omega \pm 1\%$ external resistor.

[RZ/G2E]

- This HS-USB subsystem cannot connect 5 V signals so that vbus status is informed by OVC signal status.

61.4.5 TXRTUNEn/USB_RREF Pin

- Provide a resistor (hereafter called reference resistor)*¹, between the TXRTUNEn/USB_RREF pin and AVSS, PVSS.

Note: 1. RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: $200\Omega \pm 1\%$, RZ/G2E : $1.8\text{k}\Omega \pm 1\%$

- Locate the reference resistor as close as possible to this LSI.
- The RREF pin, the reference resistor, PVSS and AVSS should be wired on wide areas and the shortest length.
- Use a wiring pattern for and only for the reference resistor, PVSS and AVSS. The pattern should be connected to the analog ground. The pattern should be designed avoiding the possibility for common impedance between RREF and other signals.
- To prevent cross talk, very-frequently switched signals such as DP, DM, clocks, and control signals for addresses and data, should not be placed near to the reference resistor, and their patterns should neither get across nor go parallel to the wiring pattern between the reference resistor and the RREF pin.

It is recommended to guard ring the reference resistor and its wiring pattern with ground.

61.5 Battery Charging Control

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Figure 61.17 shows embedded USB PHY battery charging control. BCCTRL, UGCTRL, and UGSTS provide battery charging control. The embedded USB PHY supports only D+ and D- signals.

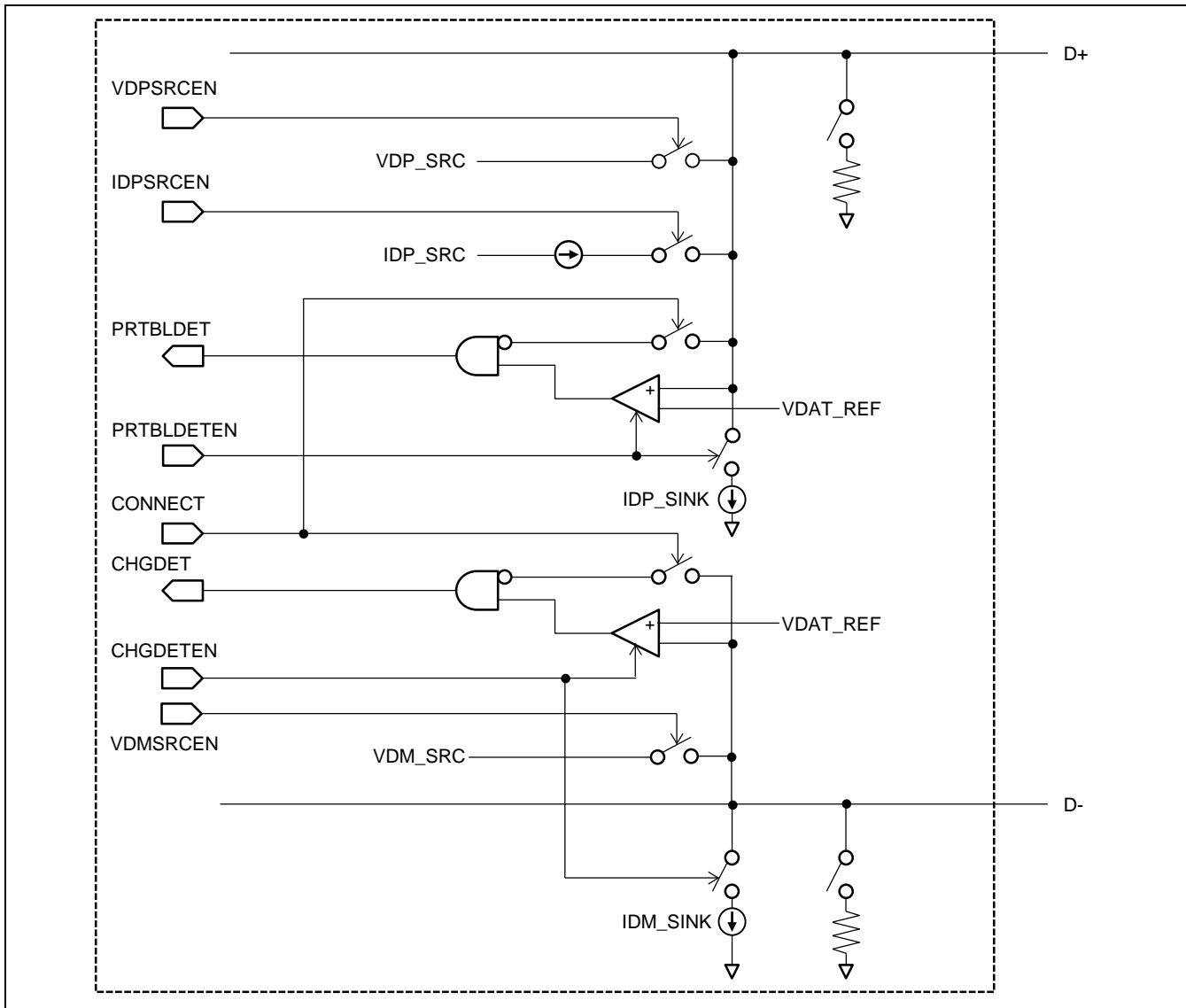


Figure 61.17 Circuit of Battery Charging Control

62. USB High-Speed DMAC (USB-DMAC)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

62.1 Overview

62.1.1 Features

- Three types of clock: S3D1 ϕ , S3D2 ϕ and HS-USB local bus clock (S3D4 ϕ : 66 MHz), which are asynchronously absorbed in the local bus bridge and AXI bus bridge.
- The AXI bus interface functions are implemented.
The AXI bus interface handles data transfer with the memory and register settings.
- Data communication is performed using the local DMA protocol.
- Supports two channels that can work concurrently.
- Interrupts
 - Setting count end interrupt
 - USB-DMAC is composed of USB-DMAC0 and USB-DMAC1.
 - Transfer end interrupt upon receiving a short packet from the HS-USB
This function allows generation of an interrupt even in the case where DMA transfer ends after receiving data less than the transfer size set in the DMAC.
 - Timeout interrupt
A timeout interrupt can be generated after the specified cycles after the last HS-USB transfer request. (This interrupt is used for the timeout when the final packet received is Max packet.)
 - Interrupts can also be generated when a NULL packet is received or an address error is detected.

Table 62.1 Main Functions of This Module

| Item | Description |
|---|---|
| Application | Transfer between HS-USB (FIFO) and AXI bus (4-Gbyte arbitrary space) |
| Purpose of DMA transfer | High-speed data transfer between HS-USB and AXI bus |
| DMA transfer end notification to the software | Interrupt signals (transfer count end, HS-USB short packet reception, timeout, etc.) |
| Transfer space/transfer address direction | 4-Gbyte arbitrary space (forward direction) to 4-Gbyte arbitrary space (forward direction) |
| Maximum transfer count | 16,777,216 |
| Transfer unit | 8, 16, or 32 bytes |
| Control signals generated | Address/data read/data write conforming to the HS-USB local bus interface and AXI bus protocols |
| Number of channels | 2 (parallel operation enabled) |
| Bus specification | AXI bus, local bus (HS-USB local bus) |
| Clock | AXI bus clock (S3D1 ϕ : 266.66 MHz), APB bus clock (S3D2 ϕ :133.33 MHz) HS-USB local bus clock (S3D4 ϕ : 66 MHz) Note: The restrictions of ratio of the clock $S3D1\phi \geq S3D2\phi \geq S3D4\phi$ Do not change the clock frequency during USB-DMAC0 or USB-DMAC1 transferring. |
| Maximum transfer rate | IN transfer: About 350 Mbps (USB-DMAC0) OUT transfer: About 340 Mbps (USB-DMAC0) IN transfer: About 175 Mbps (USB-DMAC1) OUT transfer: About 170 Mbps (USB-DMAC1) |

Table 62.2 Bus Access Type

| | Supported Access Type |
|----------------------------------|---|
| AXI bus access (USB-DMAC master) | 8-byte read/write 16-byte read/write 32-byte read/write |
| USB-DMAC register setting | 4-byte read/write 8-byte read/write |

The configuration of this module is shown below.

This module is provided with buffers and bus bridges and controls transfers between the AXI bus and USB High-Speed module (HS-USB).

62.1.2 Block Diagram

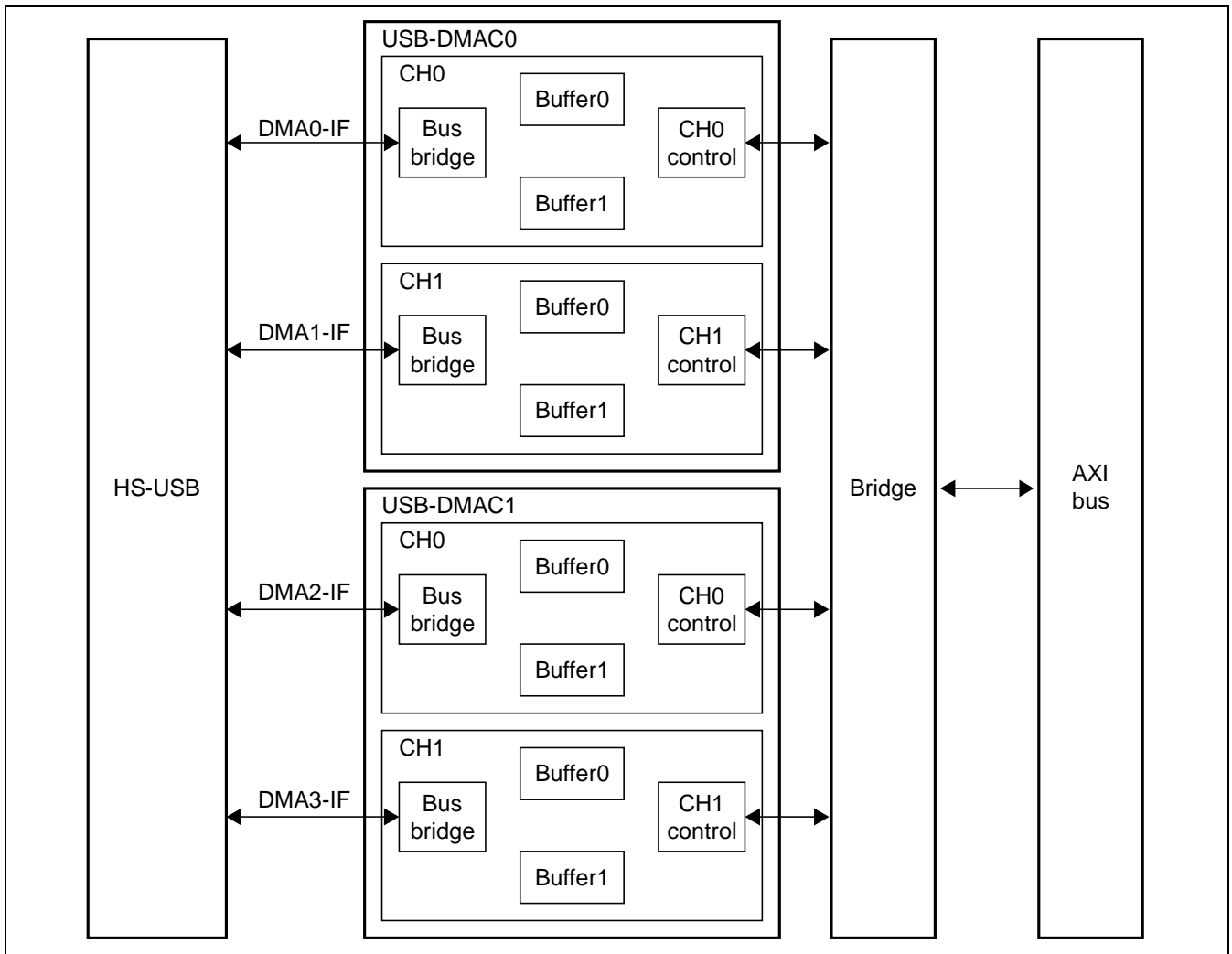


Figure 62.1 USB-DMAC Block Diagram

62.1.3 Register Configuration

Table 62.3 USB-DMAC Base Address

| EHCI/OHCI Base Address | Base Address | RZ/G2H | RZ/G2M V1.3 | | RZ/G2N | RZ/G2E |
|------------------------|--------------|--------|-------------|--|--------|--------|
| | | | RZ/G2M V3.0 | | | |
| USB-DMAC0 | H'E65A_0000 | √ | √ | | √ | √ |
| USB-DMAC1 | H'E65B_0000 | √ | √ | | √ | √ |

Table 62.4 shows the register configuration. Table 62.5 shows the register states in each processing mode.

Table 62.4 Register Configuration

| Register Name | Abbreviation | R/W | Address Offset | Access Size | ThirdGeneration RZ/G Series Products | | | | |
|--|------------------|-----|----------------|-------------|---|----------------------------|--------|--------|--|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| DMA _n VCR register | USBDMAN_VCR | R/W | H'0000 | 32 | √ | √ | √ | √ | |
| DMA _n Software reset register | USBDMAN_SWR | R/W | H'0008 | 32 | √ | √ | √ | √ | |
| DMA _n interrupt source register | USBDMAN_DMICR | R | H'0010 | 32 | √ | √ | √ | √ | |
| DMA _n source address register_0 | USBDMAN_SAR_0 | R/W | H'0020 | 32 | √ | √ | √ | √ | |
| DMA _n destination address register_0 | USBDMAN_DAR_0 | R/W | H'0024 | 32 | √ | √ | √ | √ | |
| DMA _n transfer count register_0 | USBDMAN_TCR_0 | R/W | H'0028 | 32 | √ | √ | √ | √ | |
| DMA _n timeout count register_0 | USBDMAN_TOCNTR_0 | R | H'002C | 32 | √ | √ | √ | √ | |
| DMA _n timeout constant register_0 | USBDMAN_TOCSTR_0 | R/W | H'0030 | 32 | √ | √ | √ | √ | |
| DMA _n channel control register_0 | USBDMAN_CHCR_0 | R/W | H'0034 | 32 | √ | √ | √ | √ | |
| DMA _n final transaction valid data transfer enable register_0 | USBDMAN_TEND_0 | R/W | H'0038 | 32 | √ | √ | √ | √ | |
| DMA _n source address register_1 | USBDMAN_SAR_1 | R/W | H'0040 | 32 | √ | √ | √ | √ | |
| DMA _n destination address register_1 | USBDMAN_DAR_1 | R/W | H'0044 | 32 | √ | √ | √ | √ | |
| DMA _n transfer count register_1 | USBDMAN_TCR_1 | R/W | H'0048 | 32 | √ | √ | √ | √ | |
| DMA _n timeout count register_1 | USBDMAN_TOCNTR_1 | R | H'004C | 32 | √ | √ | √ | √ | |
| DMA _n timeout constant register_1 | USBDMAN_TOCSTR_1 | R/W | H'0050 | 32 | √ | √ | √ | √ | |
| DMA _n channel control register_1 | USBDMAN_CHCR_1 | R/W | H'0054 | 32 | √ | √ | √ | √ | |
| DMA _n final transaction valid data transfer enable register_1 | USBDMAN_TEND_1 | R/W | H'0058 | 32 | √ | √ | √ | √ | |
| DMA _n operation register | USBDMAN_DMAOR | R/W | H'0060 | 32 | √ | √ | √ | √ | |

Note: 'n' in Register Name column is 0 or 1.

Table 62.5 Register States in Each Processing Mode

| Abbreviation | Power-On Reset | Module Standby |
|---------------------|-----------------------|-----------------------|
| USBDMAn_VCR | Initialized | Retained |
| USBDMAn_SWR | Initialized | Retained |
| USBDMAn_DMICR | Initialized | Retained |
| USBDMAn_SAR_0 | Initialized | Retained |
| USBDMAn_DAR_0 | Initialized | Retained |
| USBDMAn_TCR_0 | Initialized | Retained |
| USBDMAn_TOCNTR_0 | Initialized | Retained |
| USBDMAn_TOCSTR_0 | Initialized | Retained |
| USBDMAn_CHCR_0 | Initialized | Retained |
| USBDMAn_TEND_0 | Initialized | Retained |
| USBDMAn_SAR_1 | Initialized | Retained |
| USBDMAn_DAR_1 | Initialized | Retained |
| USBDMAn_TCR_1 | Initialized | Retained |
| USBDMAn_TOCNTR_1 | Initialized | Retained |
| USBDMAn_TOCSTR_1 | Initialized | Retained |
| USBDMAn_CHCR_1 | Initialized | Retained |
| USBDMAn_TEND_1 | Initialized | Retained |
| USBDMAn_DMAOR | Initialized | Retained |

Note: 'n' in Abbreviation column is 0 or 1.

62.2 Register Description

62.2.1 DMA_n VCR Register (USBDMA_n_VCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---------|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ERR_SNT | ERR_RCV |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | ERR_SNT | B'0 | R/W | Send Error Response 0: No error is generated. [Clearing condition] Writing 0 after reading 1. 1: An error is generated. |
| 0 | ERR_RCV | B'0 | R/W | Receive Error Response 0: No error is generated. [Clearing condition] Writing 0 after reading 1. 1: An error is generated. |

Note: 'n' in Register Name is 0 or 1.

62.2.2 DMA Software Reset Register (USBDMAn_SWR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | SWR |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | SWR | B'0 | R/W | Software Reset 0: Cancels reset. 1: Resets entire system. (No effect for register setting) Note: On software reset, this bit should be held high for long enough until the internal circuit stabilizes. When this bit is set to 1, it is recommended to allow several tens of clocks if operating at 260 MHz for a reset period before clearing the reset (setting this bit to 0) as a precaution. |

Note: 'n' in Register Name is 0 or 1.

62.2.3 DMA Interrupt Source Register (USBDMA_n_DMICR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----|------|-----|-------|-----|-----|-----|------------|-----|------|-----|-------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SHBSY 1 | — | — | — | — | — | — | — | SHBSY 0 | — | — | — | — | — | — | AE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | TR1 | BUF1 | RW1 | NULL1 | TO1 | SP1 | TE1 | — | TR0 | BUF0 | RW0 | NULL0 | TO0 | SP0 | TE0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

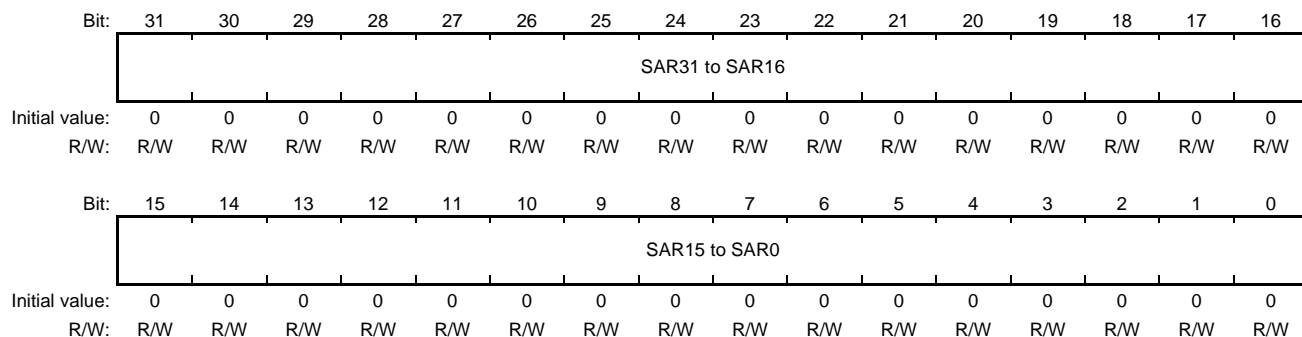
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 | SHBSY1 | B'0 | R | CH1 AXI Bus Busy Flag Monitor 0: The AXI bus is idle. 1: The AXI bus is busy. |
| 30 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 | SHBSY0 | B'0 | R | CH0 AXI Bus Busy Flag Monitor 0: The AXI bus is idle. 1: The AXI bus is busy. |
| 22 to 17 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 16 | AE | B'0 | R | Address Error Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (AE in the DMA operation register is reflected.) |
| 15 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 14 | TR1 | B'0 | R | CH1 Transaction End: Receive Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (TR in the channel 1 control register is reflected.) |
| 13 | BUF1 | B'0 | R | CH1 Buffer End Detect Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (BUF in the channel 1 control register is reflected.) |
| 12 | RW1 | B'0 | R | CH1 Final Buffer Access Detect Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (RW in the channel 1 control register is reflected.) |
| 11 | NULL1 | B'0 | R | CH1 NULL Packet Receive Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (NULL in the channel 1 control register is reflected.) |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 10 | TO1 | B'0 | R | CH1 Timeout Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (TO in the channel 1 control register is reflected.) |
| 9 | SP1 | B'0 | R | CH1 Short Packet Receive Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (SP in the channel 1 control register is reflected.) |
| 8 | TE1 | B'0 | R | CH1 Transfer End Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (TE in the channel 1 control register is reflected.) |
| 7 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 6 | TR0 | B'0 | R | CH0 Transaction End Detect Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (TR in the channel 0 control register is reflected.) |
| 5 | BUF0 | B'0 | R | CH0 Buffer End Detect Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (BUF in the channel 0 control register is reflected.) |
| 4 | RW0 | B'0 | R | CH0 RWEND Receive Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (RW in the channel 0 control register is reflected.) |
| 3 | NULL0 | B'0 | R | CH0 NULL Receive Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (NULL in the channel 0 control register is reflected.) |
| 2 | TO0 | B'0 | R | CH0 Timeout Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (TO in the channel 0 control register is reflected.) |
| 1 | SP0 | B'0 | R | CH0 Short Packet Receive Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (SP in the channel 0 control register is reflected.) |
| 0 | TE0 | B'0 | R | CH0 Transfer End Interrupt Source 0: No interrupt is generated. 1: An interrupt is generated. (TE in the channel 0 control register is reflected.) |

Note: 'n' in Register Name is 0 or 1.

62.2.4 DMA Source Address Registers 0 and 1 (USBDMAN_SAR_0/1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

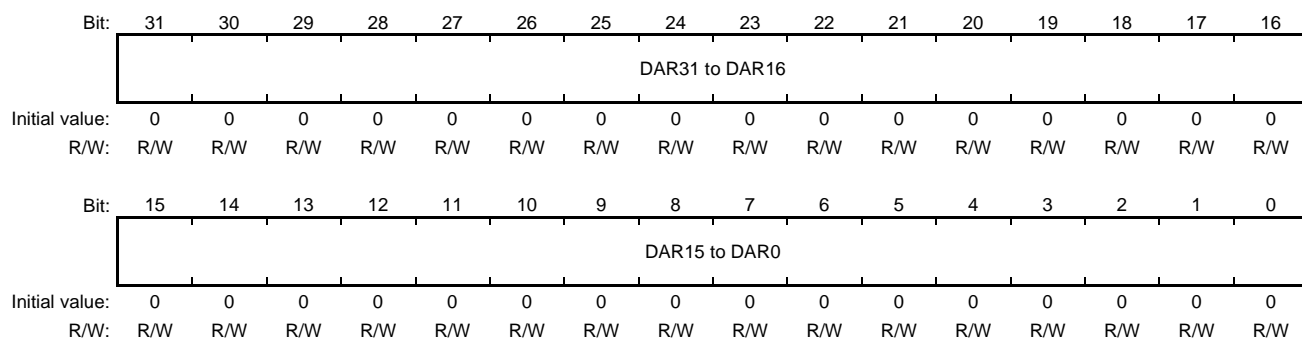


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 0 | SAR31 to SAR0 | H'0000_0000 | R/W | Source Address (H'0000_0000 to H'FFFF_FFFF (4 Gbytes)) Set a value of 8-byte, 16-byte, or 32-byte boundary. Indicate the next transfer source address during a DMA transfer. |

Note: 'n' in Register Name is 0 or 1.

62.2.5 DMA Destination Address Registers 0 and 1 (USBDMAN_DAR_0/1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

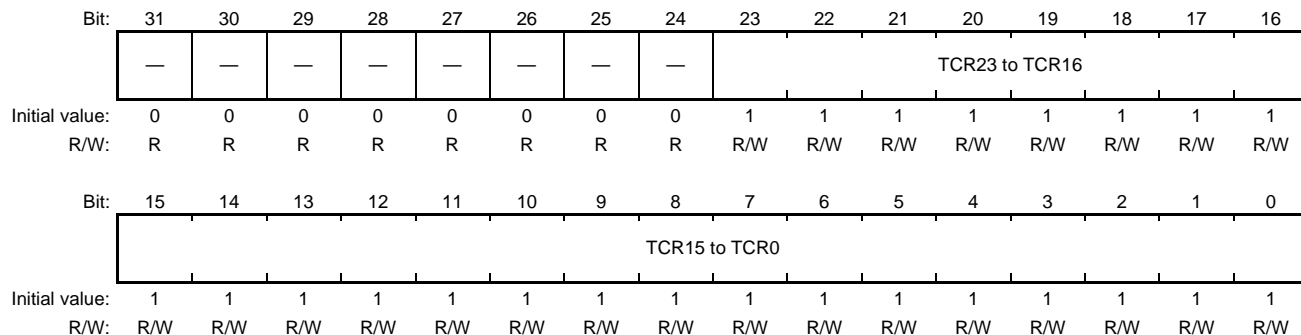


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 0 | DAR31 to DAR0 | H'0000_0000 | R/W | Destination Address (H'0000_0000 to H'FFFF_FFFF (4 Gbytes)) Set a value of 8-byte, 16-byte, or 32-byte boundary. Indicate the next transfer destination address during a DMA transfer. |

Note: 'n' in Register Name is 0 or 1.

62.2.6 DMA Transfer Count Registers 0 and 1 (USBDMAn_TCR_0/1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 to 0 | TCR23 to TCR0 | H'FF_ FFFF* | R/W | Transfer Count <ul style="list-style-type: none"> When H'00_0001 is set, total transfer volume is as follows: Transfer size = 32 bytes: 32 × 1 bytes Transfer size = 16 bytes: 16 × 1 bytes Transfer size = 8 bytes: 8 × 1 bytes When H'FF_FFFF is set, 16,777,215 times When H'00_0000 is set, 16,777,216 times (maximum) Indicate 0 when a DMA transfer completes successfully. |

Note: * To read this register, read the internal transfer counter value (the TCR counter value) directly. This register is read as all 1 after a reset is cleared because the initial value of the TCR counter is all 1 (H'FF_FFFF). When DMA is enabled (DE in USBDMAn_CHCR is asserted), the TCR counter is loaded with the value set in this register and decrements during DMA transfers. This means that the remaining transfer count is read back. Note that the read value may be different from the one set in this register due to this reason.

Note: 'n' in Register Name is 0 or 1.

62.2.7 DMA_n Timeout Count Registers 0 and 1 (USBDMAN_TOCNTR_0/1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

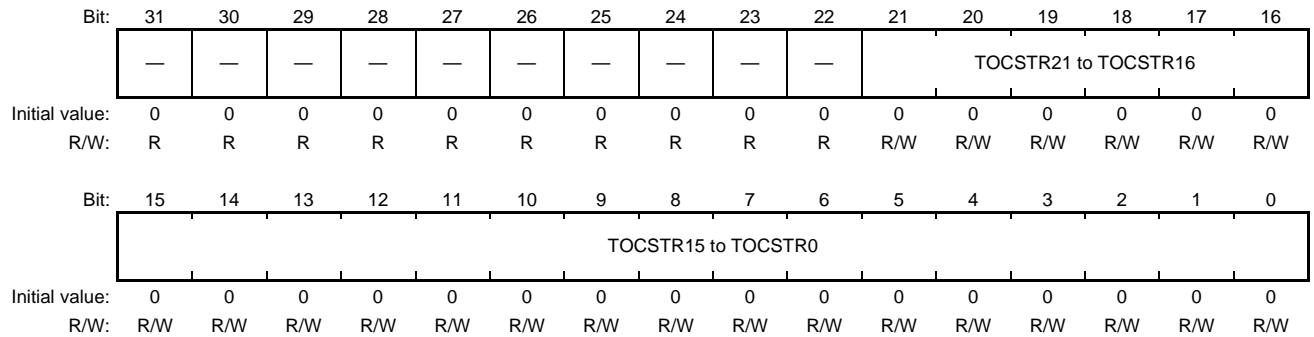
| | | | | | | | | | | | | | | | | |
|----------------|---------------------|----|----|----|----|----|----|----|----|----|----------------------|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | TOCNTR21 to TOCNTR16 | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TOCNTR15 to TOCNTR0 | | | | | | | | | | | | | | | |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|---|
| 31 to 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 to 0 | TOCNTR21 to TOCNTR0 | H'3F_FFFF | R | Timeout Counter Value (H'00_0000 to H'27_AC40) Indicate the remaining timeout count during timeout counting. (Countable up to H'3F_FFFF (approximately 21 ms) by the hardware. See the descriptions of the USBDMAN_TOCSTR register.) |

Note: 'n' in Register Name is 0 or 1.

62.2.8 DMA_n Timeout Constant Registers 0 and 1 (USBDMAN_TOCSTR_0/1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|--|
| 31 to 22 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 21 to 0 | TOCSTR21 to TOCSTR0 | H'00_0000 | R/W | Timeout Constant Value (H'00_0000 to H'27_AC40) Set a timeout value (at 200 MHz cycle). The maximum timeout time is approximately 13ms. (Setting up to H'3F_FFFF (approximately 21 ms) is allowed by the hardware.) |

Note: 'n' in Register Name is 0 or 1.

62.2.9 DMA Channel Control Registers 0 and 1 (USBDMAN_CHCR_0/1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|------|----|----|----|-----|-----|-----|-----|------|-----|------|-----|-------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | FTE | — | — | — | SPIM | TRE | BUFE | RWE | NULLE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TR | BUF | RW | NULL | — | — | — | — | TS1 | TS0 | IE | TOE | TO | SP | TE | DE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 25 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 24 | FTE | B'0 | R/W | Forced TE Set Register When a NULL packet is received, the TE bit is forced to 1 to terminate the transfer by setting 1 in this register. (The DREQE bit of the HS-USB should also be controlled accordingly. For details on the control procedure and restrictions, see section 62.3.2 (3), DMA transfer flow.) This bit is always read as 0. 0: No effect (nothing happens). 1: Forces the TE bit to 1. |
| 23 to 21 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 20 | SPIM | B'0 | R/W | Short Packet Receive Interrupt Mask Enables or disables the reflection of the short packet (SP) receive flag in the interrupt signal. If this bit is set to 1, no interrupts are generated when an SP is received. However, the SP receive flag (the SP bit) is set regardless of this register setting. 0: The short packet receive interrupt is enabled (default). 1: The short packet receive interrupt is disabled. |
| 19 | TRE | B'0 | R/W | Transaction End Detect Interrupt Flag Enable Enables or disables the reflection of TR in the interrupt signal and the DMA stop/restart control. (When this bit is set to 0, the TR bit is always 0.) 0: The TR flag is disabled. 1: The TR flag is enabled. |
| 18 | BUFE | B'0 | R/W | Buffer End Detect Interrupt Flag Enable Enables or disables the reflection of BUF in the interrupt signal and the DMA stop/restart control. (When this bit is set to 0, the BUF bit is always 0.) 0: The BUF flag is disabled. 1: The BUF flag is enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|------|--|
| 17 | RWE | B'0 | R/W | Final Buffer Access Detect Interrupt Flag Enable Enables or disables the reflection of RW in the interrupt signal and the DMA stop/restart control. (When this bit is set to 0, the RW bit is always 0.) 0: The RW flag is disabled. 1: The RW flag is enabled. |
| 16 | NULLE | B'0 | R/W | NULL Receive Interrupt Flag Enable Enables or disables the reflection of the NULL receive interrupt flag (NULL) in the interrupt signal. 0: The NULL receive interrupt flag is disabled. 1: The NULL receive interrupt flag is enabled. |
| 15 | TR | B'0 | R/W* | Transaction End Detect Interrupt Flag When the transaction end is detected, an interrupt is generated. 0: DMA transfer is in progress or suspended. [Clearing condition] Writing 0 after reading 1. 1: A transaction end receive interrupt is generated. |
| 14 | BUF | B'0 | R/W* | Buffer End Detect Interrupt Flag When the end of the buffer is detected, an interrupt is generated. 0: DMA transfer is in progress or suspended. [Clearing condition] Writing 0 after reading 1. 1: A buffer end detect interrupt is generated. |
| 13 | RW | B'0 | R/W* | Final Buffer Access Detect Interrupt Flag When the final access to the buffer is detected, an interrupt is generated. 0: DMA transfer is in progress or suspended. [Clearing condition] Writing 0 after reading 1. 1: A final buffer access detect interrupt is generated. |
| 12 | NULL | B'0 | R/W* | NULL Receive Interrupt Flag When NULL packet is received, an interrupt is generated. 0: DMA transfer is in progress or suspended. [Clearing condition] Writing 0 after reading 1. 1: A NULL packet receive interrupt is generated. |
| 11 to 8 | — | All 0 | R | Reserved The bits should always be written as 0. |
| 7, 6 | TS1 to TS0 | B'00 | R/W | DMA Transfer Size B'00: 8 bytes B'01: 16 bytes B'10: 32 bytes B'11: Setting prohibited Specify the same transfer size and address boundary when setting addresses in USBDMAN_SAR and USBDMAN_DAR as a transfer source and transfer destination. If TS = 11 is set, an address error is generated and DMA transfer is disabled. |

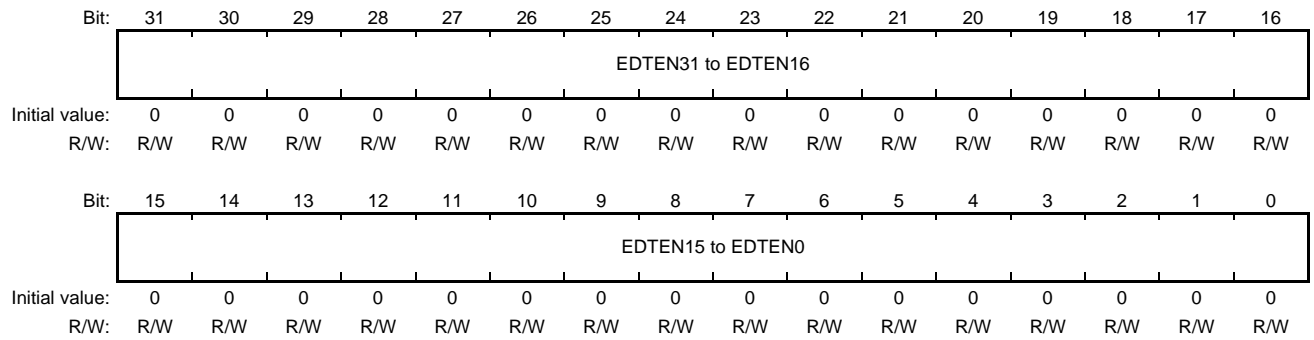
| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 5 | IE | B'0 | R/W | <p>Interrupt Enable</p> <p>Enables or disables an interrupt request to the CPU when DMA transfer ends. When AE, TE, SP, TO, NULL, RW, BUF, or TR bit is set with this bit set to 1, an interrupt request is output to the CPU. However, when the TRE, BUFE, RWE, NULLE, or TOE bit is set to disabled, no interrupt due to TR, BUF, RW, NULL, or TO is generated.</p> <p>0: An interrupt request is disabled. 1: An interrupt request is enabled.</p> |
| 4 | TOE | B'0 | R/W | <p>Timeout Enable</p> <p>Enables or disables the reflection of the timeout flag (TO) in the interrupt.</p> <p>0: Timeout flag is disabled. 1: Timeout flag is enabled.</p> |
| 3 | TO | B'0 | R/W* | <p>Timeout Flag</p> <p>After a transfer request REQ is negated, a timeout interrupt is generated after the specified cycles. (Used for timeout when the final packet is Max packet.)</p> <p>0: DMA transfer is in progress or suspended. [Clearing condition] Writing 0 after reading 1. 1: Timeout is generated.</p> |
| 2 | SP | B'0 | R/W* | <p>Short Packet Receive Flag</p> <p>A transfer end interrupt is generated upon receiving a short packet from the HS-USB. For example, if DMA transfer size is set to 1 MB but actual data transfer is 900 KB, this interrupt is generated when transfer of 900 KB finishes.</p> <p>0: DMA transfer is in progress or suspended. [Clearing condition] Writing 0 after reading 1. 1: A short packet is received.</p> |
| 1 | TE | B'0 | R/W* | <p>Transfer End Flag</p> <p>When a DMA transfer ends with USBDMAN_TCR = 0, the TE bit is set to 1. When a DMA address error occurs before USBDMAN_TCR is cleared to 0 and the transfer is suspended by clearing the DE bit and the DME bit in USBDMAN_DMAOR, the TE bit is not set to 1. To clear the TE bit, write 0 after reading 1. When TE = 1, DMA transfer is disabled even if the DE bit is set to 1.</p> <p>0: DMA transfer is in progress or suspended. [Clearing condition] Writing 0 after reading 1. 1: DMA transfer end (TCR = 0)</p> |
| 0 | DE | B'0 | R/W | <p>DMA Enable</p> <p>Enables or disables DMA transfer. Setting the DME bit in USBDMAN_DMAOR and the DE bit for the corresponding channel to 1 starts a DMA transfer, provided that all interrupt flags are 0. Clearing the DE bit to 0 suspends the ongoing transfer after the transfer of one transaction is completed.</p> <p>0: DMA transfer is disabled. 1: DMA transfer is enabled.</p> |

Note: * Only writing 0 to clear the flag is enabled.

Notes: 'n' in Register Name is 0 or 1.

62.2.10 DMA_n Final Transaction Valid Data Transfer Enable Registers 0 and 1 (USBDMA_n_TEND_0/1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------------|---------------|-----|--|
| 31 to 0 | EDTEN31 to EDTEN0 | H'0000_0000 | R/W | <p>Final Transaction Valid Data Transfer Enable (valid only for IN transfers)</p> <p>Specify valid data size in units of bytes in the final transaction of data transfer (see section 62.3.2, DMA Transfer Function).</p> <p>H'8000_0000: 32 bytes × (n-1) + 1 byte</p> <p>H'C000_0000: 32 bytes × (n-1) + 2 bytes</p> <p style="text-align: center;">:</p> <p>H'FFFF_FFFF: 32 bytes × (n-1) + 32 bytes</p> <p>Notes: 1. n: Transfer count setting (TCR)</p> <p style="margin-left: 20px;">2. For the 8-byte transfer, set EDTEN23 to EDTEN0 to 0. For the 16-byte transfer, set EDTEN15 to EDTEN0 to 0.</p> |

Note: 'n' in Register Name is 0 or 1.

62.2.11 DMA Operation Register (USBDMAn_DMAOR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|---------------|------|-------------|-----|-----|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | TID1 and TID0 | RM | PR1 and PR0 | AE | DME | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W* | R/W* | R/W | R/W | R/W | R/W* | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|------|---|
| 31 to 7 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 6, 5 | TID1 and TID0 | B'00 | R/W* | Response Error Channel Identity Information TID1 indicates the channel 1 error information, and TID0 indicates the channel 0 error information. TID1 0: No error 1: Channel 1 response packet error TID0 0: No error 1: Channel 0 response packet error [Setting condition] When a CH0 (1) response error occurs [Clearing condition] Clearing ERR_RCV in USBDMAn_VCR (Channel 0/channel 1 concurrent clear) |
| 4 | RM | B'0 | R/W | Response Error Mask Mode 0: A response error is output. 1: A response error is masked. |
| 3, 2 | PR1 and PR0 | B'00 | R/W | Priority Mode Specify the priority of channels to execute transactions when transfer requests are made simultaneously in both channel 0 and channel 1 as follows: B'00: Channel 0 > Channel 1 B'01: Channel 1 > Channel 0 B'10: Setting prohibited (If PR = 10 is set, channel 0 > channel 1) B'11: Setting prohibit |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|---|
| 1 | AE | B'0 | R/W* | <p>Address Error Flag</p> <p>Indicates that an address error interrupt occurred when setting USBDMAN_SAR or USBDMAN_DAR. This flag is set to 1 when the value of USBDMAN_SAR or USBDMAN_DAR differs from the transfer size (TS) boundary, and when the TS bits are set to 11 (setting prohibited).</p> <p>When the AE bit is set to 1, DMA transfer is not enabled even if the DE bit in DMACHCR and the DME bit in USBDMAN_DMAOR are set to 1.</p> <p>0: No address error interrupt is generated by the DMAC.</p> <p>[Clearing condition]</p> <p>Setting USBDMAN_SAR and USBDMAN_DAR to values that match the transfer size boundary.</p> <p>1: An address error interrupt is generated.</p> |
| 0 | DME | B'0 | R/W | <p>DMA Master Enable</p> <p>Enables or disables DMA transfers for all channels. When the DME bit and the DE bit in USBDMAN_CHCR are set to 1, DMA transfer is enabled, provided that all interrupt flags in USBDMAN_CHCR of the transfer channel are 0. Clearing the DME bit suspends the DMA transfer for all channels.</p> <p>0: DMA transfer for all channels is disabled.</p> <p>1: DMA transfer for all channels is enabled.</p> |

Note: * Only writing 0 to clear the flag is enabled.

Note: 'n' in Register Name is 0 or 1.

62.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

62.3.1 Interrupt/Error Detection Function

This section describes the interrupt/error detection function that is implemented in this module.

Table 62.6 lists the mask, source, and flag registers for each interrupt type.

Table 62.6 Registers by Interrupts/Errors

| | Interrupt Generation | Interrupt Mask Register | Interrupt Mask Register (Individual Mask) | Source Register | Flag Register |
|----------------------------|----------------------|-------------------------|---|----------------------|--|
| Transfer count end | √ | USBDMAn_CHCR_x.IE | — | USBDMAn_DMICR.TEx | USBDMAn_CHCR_x.TE |
| Short packet receive | √ | USBDMAn_CHCR_x.IE | USBDMAn_CHCR_x.SPI M | USBDMAn_DMICR.SPx | USBDMAn_CHCR_x.SP |
| NULL packet receive | √ | USBDMAn_CHCR_x.IE | USBDMAn_CHCR_x.NUL LE | USBDMAn_DMICR.NUL Lx | USBDMAn_CHCR_x.N ULL |
| Timeout | √ | USBDMAn_CHCR_x.IE | USBDMAn_CHCR_x.TOE | USBDMAn_DMICR.TOx | USBDMAn_CHCR_x.T O |
| Transaction end detect | √ | USBDMAn_CHCR_x.IE | USBDMAn_CHCR_x.TRE * | USBDMAn_DMICR.TRx | USBDMAn_CHCR_x.TR |
| Buffer end detect | √ | USBDMAn_CHCR_x.IE | USBDMAn_CHCR_x.BUF E* | USBDMAn_DMICR.BUF x | USBDMAn_CHCR_x.B UF |
| Final buffer access detect | √ | USBDMAn_CHCR_x.IE | USBDMAn_CHCR_x.RWE * | USBDMAn_DMICR.RW x | USBDMAn_CHCR_x.R W |
| Address error | √ | USBDMAn_CHCR_x.IE | — | USBDMAn_DMICR.AE | USBDMAn_DMAOR.AE |
| Response packet error | — | — | — | — | USBDMAn_VCR.ERR_SNT USBDMAn_VCR.ERR_RCV |

Note: These registers not only mask interrupt generation but also disable error processing (flag sources and flag clearing).

Note: 'n' in Register Name is 0 or 1.

Table 62.7 shows the processing to be executed when different interrupts occur. DMA transfer control is not affected when the NULL packet receive, timeout, or response packet error interrupt occurs. When the transfer count end interrupt occurs, processing is restarted. Processing can be resumed or restarted after interrupts occur from other sources.

Table 62.7 Processing to be Executed When Interrupts Occur (Suspend/Resume/Restart)

| | Suspend/Resume | Terminate/Restart | Suspend/Resume/Restart Control Mask Register |
|----------------------------|----------------|-------------------|--|
| Transfer count end | — | √ | — |
| Short packet receive | √ | √ | — |
| NULL packet receive | — | √* | — |
| Timeout | — | √ | — |
| Transaction end detect | √ | √ | USBDMAn_CHCR_x.TRE |
| Buffer end detect | √ | √ | USBDMAn_CHCR_x.BUFE |
| Final buffer access detect | √ | √ | USBDMAn_CHCR_x.RWE |
| Address error | √ | √ | — |
| Response packet error | — | — | — |

Notes: * When a NULL packet is received, processing is forced to be terminated and restarted by the FTE bit (see section 62.3.2 (3), DMA transfer flow.).
 'n' in Register Name is 0 or 1.

The internal circuit is reloaded register values when restarted, but not when resumed.

Table 62.8 lists the interrupt generating conditions.

Table 62.8 Interrupt Generating Conditions

| Interrupt Name | Interrupt Conditions |
|--------------------------------------|---|
| Transfer count end interrupt | When the final data transfer of the specified transfer count (in the USBDMAn_TCR register) completes IN transfer: When the final access to HS-USB completes OUT transfer: When the data transmission to the AXI completes |
| Short packet receive interrupt | When the data transfer at the time the end of a transaction is detected completes |
| NULL packet receive interrupt | When a NULL packet is detected or the forced TE register is set |
| Timeout interrupt | When the specified time elapses after the last transfer request |
| Transaction end detect interrupt | When the end of a transaction is detected |
| Buffer end detect interrupt | When the end of the buffer is detected |
| Final buffer access detect interrupt | When the final access to the buffer is detected |
| Address error interrupt | When invalid access to the DMAC register is attempted (With invalid USBDMAn_SAR/USBDMAn_DAR values or invalid transfer size (TS) specified) |
| Response packet error | Error Transmission (ERR_SNT) When accessed by a transaction that is not supported Error Reception (ERR_RCV) When received a response packet error from the system |

Note: 'n' in Register Name is 0 or 1.

Figure 62.2 shows a block diagram of the interrupt circuit.

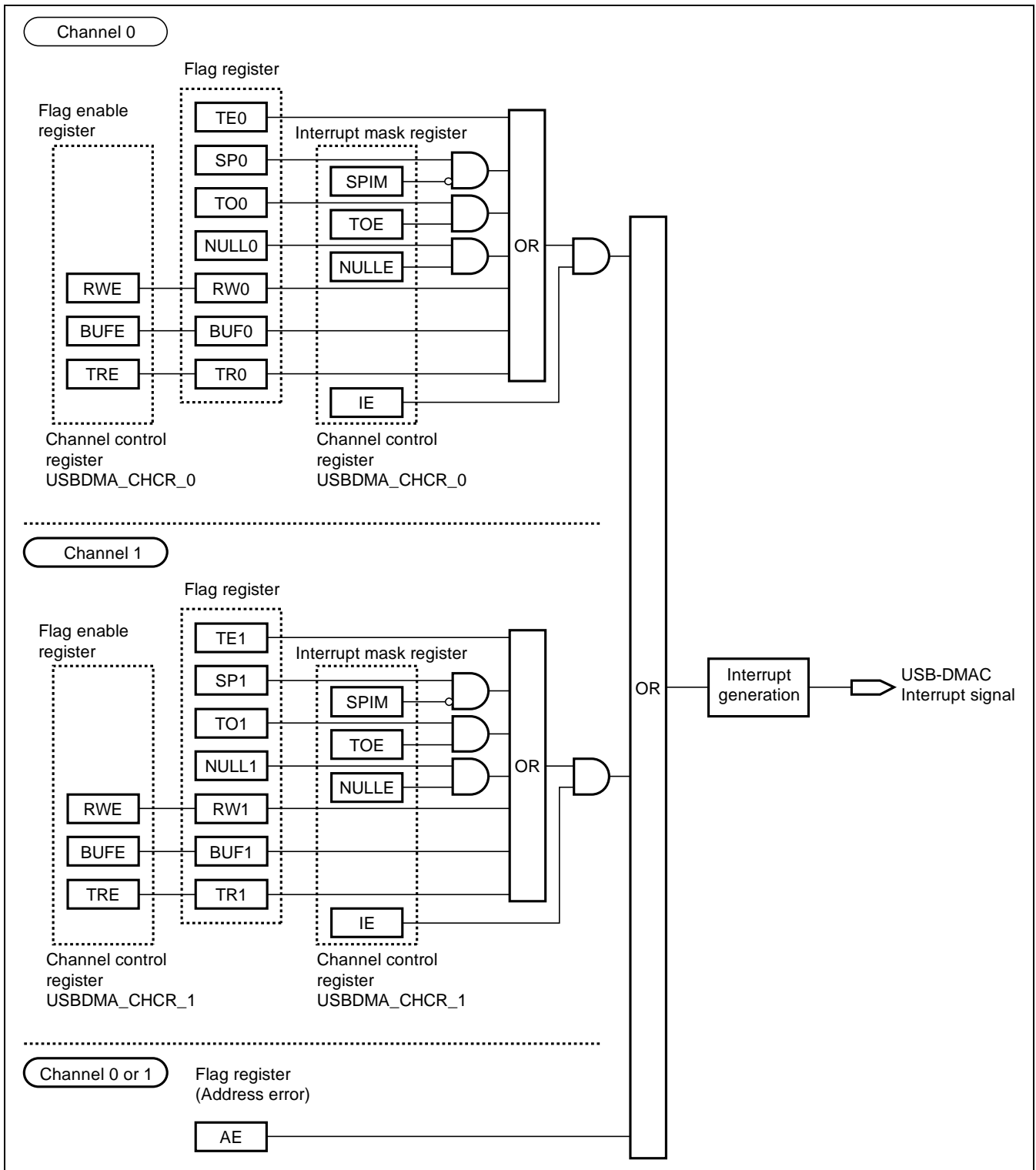


Figure 62.2 Block Diagram of Interrupt Circuit

(1) Transfer count end interrupt (TE flag)

Can be generated when the specified number of transfers complete or the FTE bit in the channel control register (USBDMA_n_CHCR_0 to USBDMA_n_CHCR_1) is set to 1. When the IE bit in the USBDMA_n_CHCR_0 to USBDMA_n_CHCR_1 is set to 0, no interrupts are generated. However, the TE bit in USBDMA_n_CHCR_0 to USBDMA_n_CHCR_1 and the TE0 or TE1 bit in the interrupt source register (USBDMA_n_DMICR) are set to 1 regardless of the IE bit setting.

The internal transfer counter is loaded with the value set in the USBDMA_n_TCR_x register and decrements based on the AXI clock every time one transaction (8-/16-/32-byte) completes. When the transfer count value reaches 0 (the end of a transfer count), the counter sets the TE (transfer end) flag from 0 to 1 (when the FTE bit is set, it is reflected on the TE flag immediately). The TE bit is cleared by writing 0 to it after reading it as 1 by software.

Note: 'n' in Register Name is 0, 1.

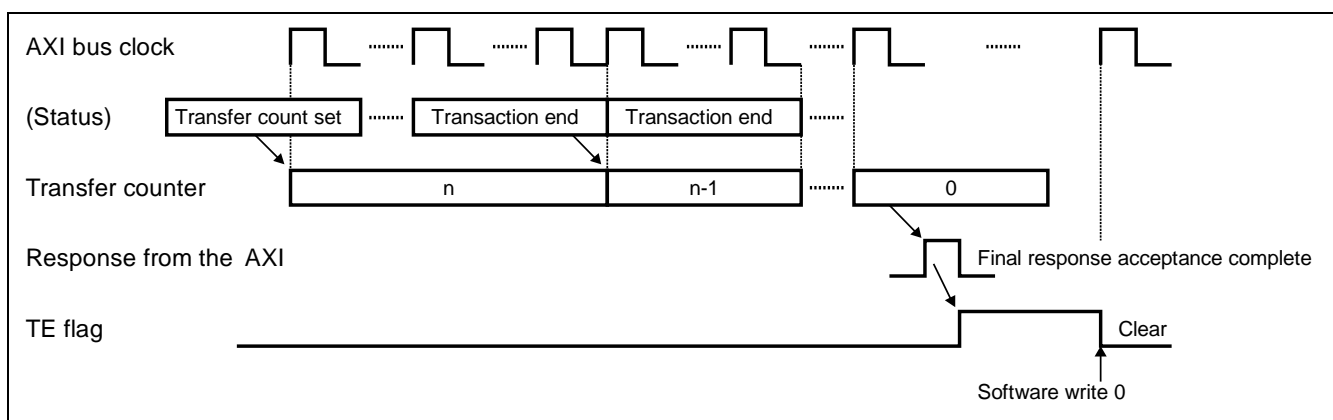


Figure 62.3 Transfer Count End Interrupt (TE) Generation Timing

(2) Short packet receive interrupt (SP flag)

Can be generated when a short packet is detected. When the SPIM bit in the USBDMA_n_CHCR_x register is set to 1, no interrupts are generated. However, the SP bit (SP flag) in the USBDMA_n_CHCR_x register and the SP0 or SP1 bit in the USBDMA_n_DMICR register are set to 1, regardless of the SPIM bit setting. The SP bit is cleared by writing 0 to it after reading it as 1 by software.

Note: 'n' in Register Name is 0,1.

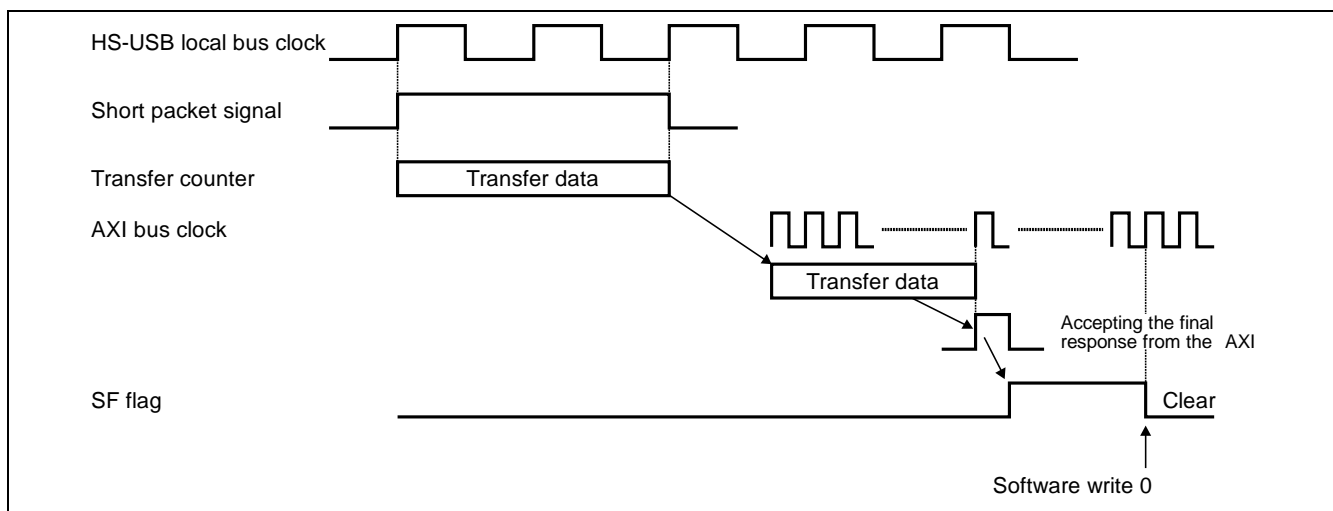


Figure 62.4 Short Packet Receive Interrupt (SP) Generation Timing

(3) NULL packet receive interrupt (NULL flag)

Can be generated when a NULL packet is detected. When the NULLE bit in the USBDMAN_CHCR_x register is set to 0, no interrupts are generated. However, the NULL bit (NULL flag) in the USBDMAN_CHCR_x register and the NULL0 or NULL1 bit in the USBDMAN_DMICR register are set to 1, regardless of the NULLE bit setting. Note also that the DMA transfer control in this module, such as stop, suspend, or resume, is not affected when a NULL packet is received. The software is responsible for implementing special processing if necessary since in this module this interrupt is designed only to signal the software that a NULL packet is received.

Note: 'n' in Register Name is 0 or 1.

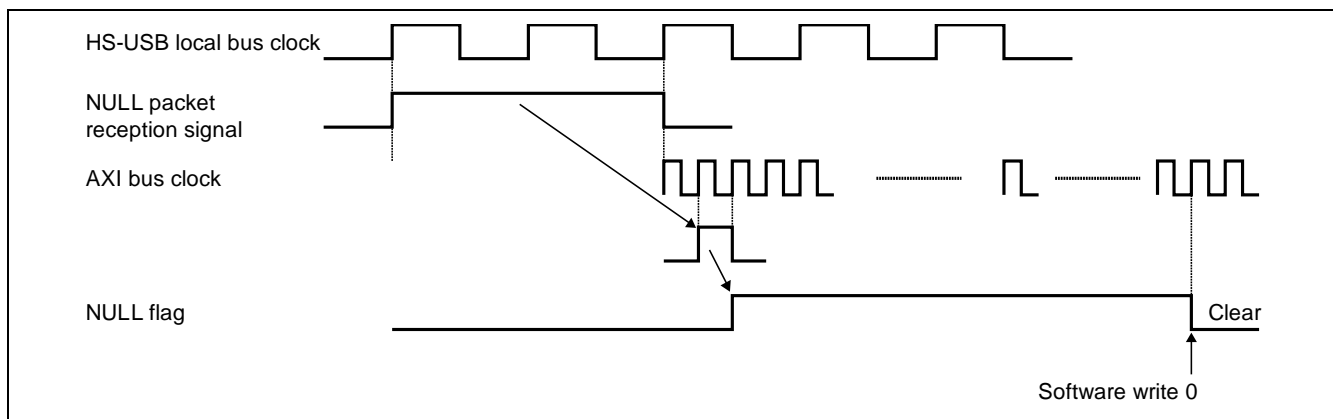


Figure 62.5 NULL Packet Receive Interrupt (NULL) Generation Timing

(4) Timeout interrupt (TO flag)

Can be generated when the time period that is specified in the timeout constant register (USBDMAN_TOCSR_0 or USBDMAN_TOCSR_1) elapses after the final transfer request is received from the HS-USB. On detecting the final transfer request, the internal counter is loaded with the value set in the USBDMAN_TOCSR_x register and starts counting down. When the counter reaches 0, it generates a timeout interrupt. Note that the DMA transfer control in this module, such as stop, suspend, or resume, is not affected when a timeout interrupt occurs. The TO bit is cleared by writing 0 to it after reading it as 1 by software. Figure 62.6 and 62.7 show the timeout interrupt generation timing.

Note: 'n' in Register Name is 0 or 1.

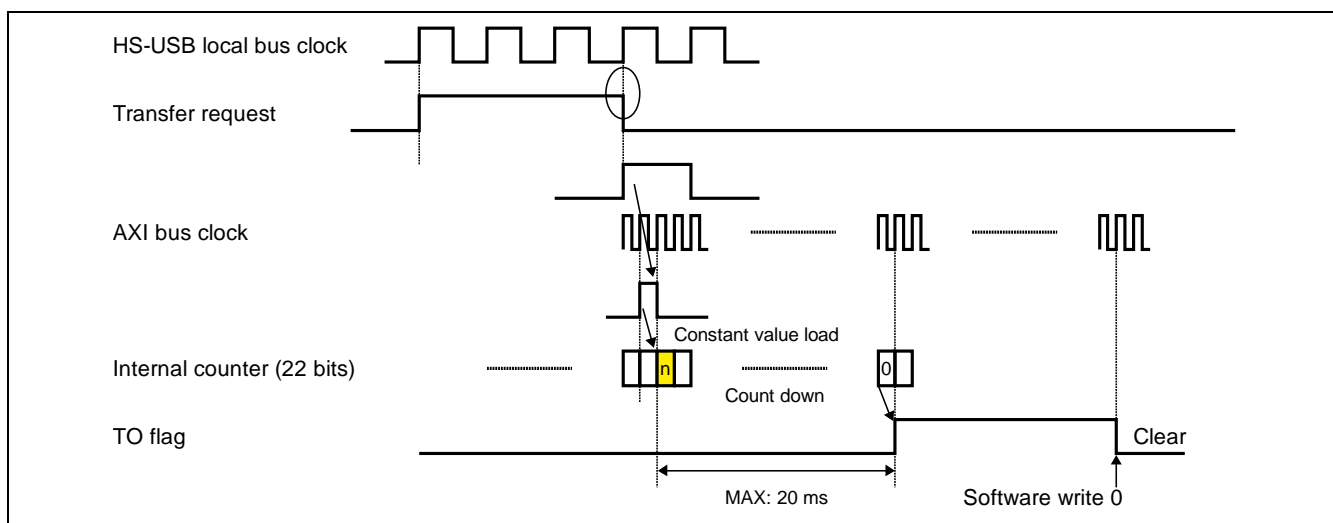


Figure 62.6 Timeout Interrupt (TO) Generation Timing

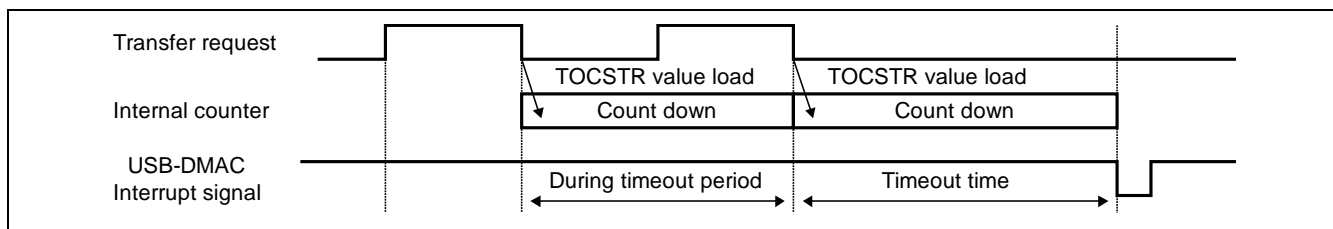


Figure 62.7 Timeout Interrupt Timing

The functions described in section 62.3.1 (5) to 62.3.1 (9) are added to detect and notify the transitions on an individual signal line to the software, but no use is defined. They can be used as required.

(5) Transaction end detect interrupt (TR flag)

Can be generated when the end of a transaction is detected within the HS-USB. Note that the TR bit in the USBDMAN_CHCR_x register and the TR0 or TR1 bit in the USBDMAN_DMICR register are not reflected in the interrupt signal when the TRE bit in the USBDMAN_CHCR_x register is set to 0.

Note: 'n' in Register Name is 0 or 1.

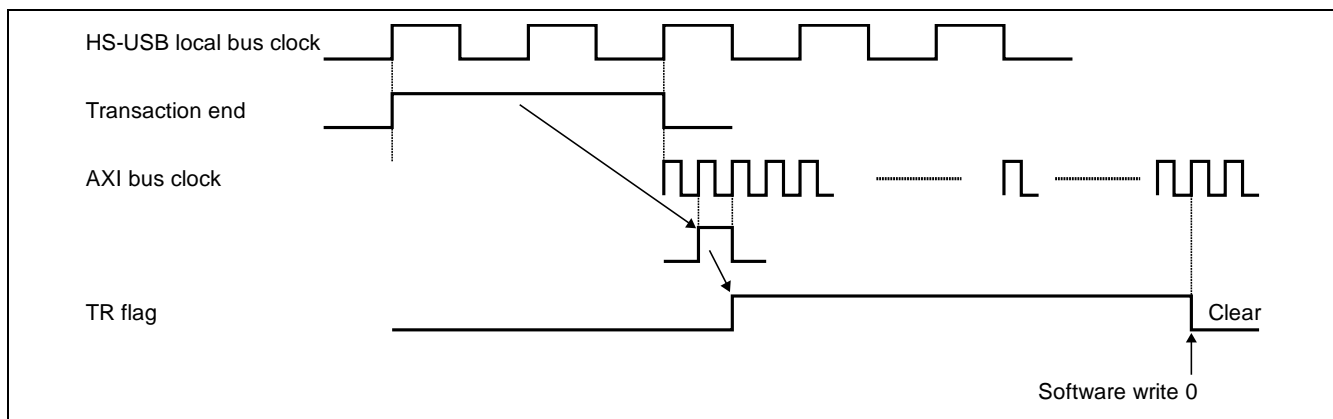


Figure 62.8 Transaction End Detect Interrupt Timing

(6) Buffer end detect interrupt (BUF flag)

Can be generated when the end of the buffer in the HS-USB is detected. Note that the BUF bit in the USBDMAN_CHCR_x register and the BUF0 or BUF1 bit in the USBDMAN_DMICR register are not reflected in the interrupt signal when the BUFE bit in the USBDMAN_CHCR_x register is set to 0.

Note: 'n' in Register Name is 0 or 1.

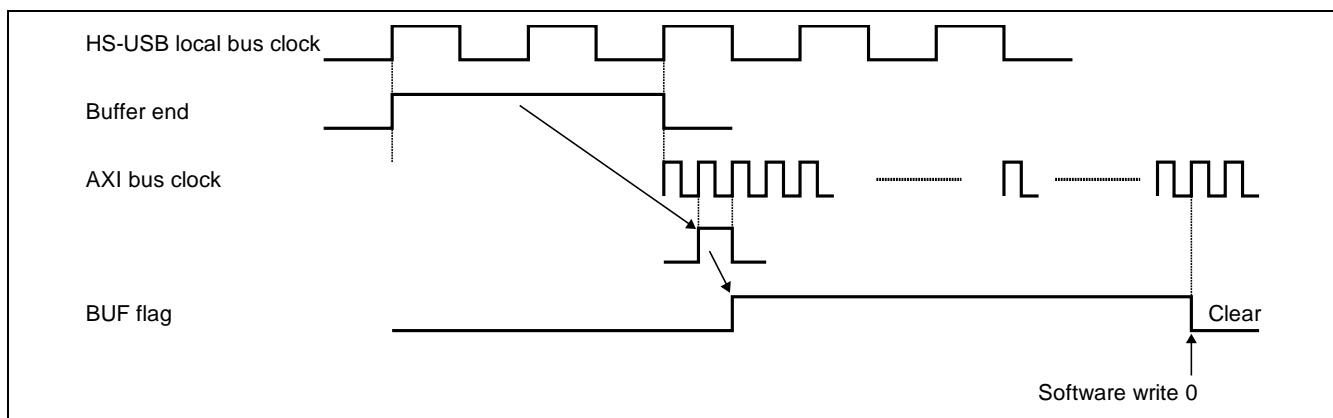


Figure 62.9 Buffer End Detect Interrupt Timing

(7) Final buffer access detect interrupt (RW flag)

Can be generated when the final access to the buffer in the HS-USB is detected. Note that the RW bit in the USBDMAN_CHCR_x register and the RW0 or RW1 bit in the USBDMAN_DMICR register are not reflected in the interrupt signal when the RWE bit in the USBDMAN_CHCR_x register is set to 0.

Note: 'n' in Register Name is 0 or 1.

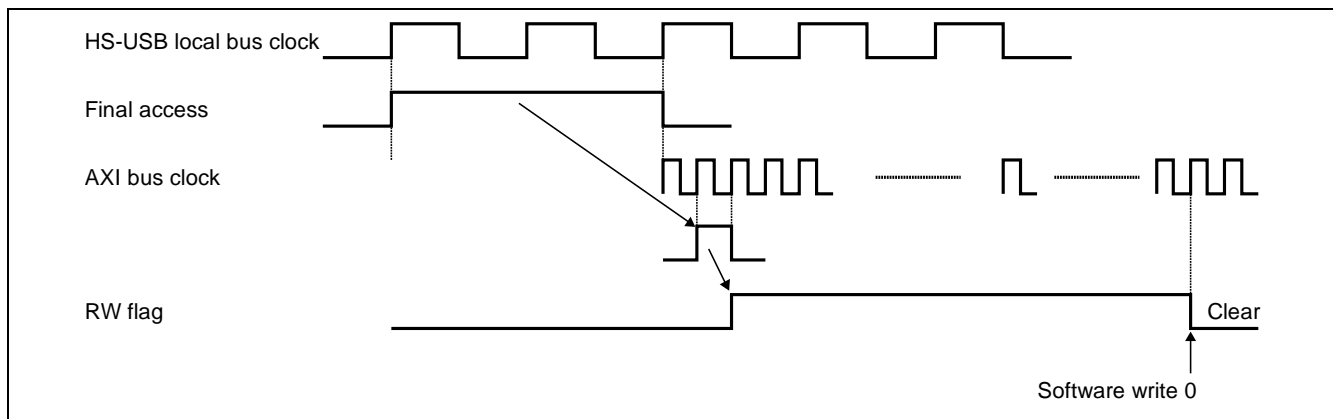


Figure 62.10 Final Buffer Access Detect Interrupt Timing

(8) Address error detection

This is a flag that indicates an address error interrupts has occurred when setting the source address register (USBDMAN_SAR_0 to USBDMAN_SAR_1) and the destination address register (USBDMAN_DAR_0 to USBDMAN_DAR_1). When the value written to the USBDMAN_SAR or USBDMAN_DAR register differs from the transfer size (TS) boundary, and when the TS bits are set to 11 (setting prohibited), the AE bit in the USBDMAN_DMAOR register is set to 1. There is no distinction between CH0 and CH1 for an address error (the AE bit is set to 1 when an address error occurs on CH0 or CH1).

While the AE is set, no DMA transfer is allowed even if the DE bit in the USBDMAN_CHCR_x register and the DME bit in the USBDMAN_DMAOR register are set to 1. The address error interrupt is cleared by setting correct addresses in USBDMAN_SAR_x and USBDMAN_DAR_x.

Note: 'n' in Register Name is 0 or 1.

Table 62.9 Address Error Detection Conditions When Setting Transfer Sizes

| Transfer Size | Address Error Detection Conditions |
|---------------------------------|--|
| TS = B'00 (8-byte transfer) | When data that is not aligned to 8-byte boundaries is written to the USBDMAN_SAR/ USBDMAN_DAR registers (Writes to the USBDMAN_SAR/ USBDMAN_DAR registers are valid.) |
| TS = B'01 (16-byte transfer) | When data that is not aligned to 16-byte boundaries is written to the USBDMAN_SAR/ USBDMAN_DAR registers (Writes to the USBDMAN_SAR/ USBDMAN_DAR registers are valid.) |
| TS = B'10 (32-byte transfer) | When data that is not aligned to 32-byte boundaries is written to the USBDMAN_SAR/ USBDMAN_DAR registers (Writes to the USBDMAN_SAR/ USBDMAN_DAR registers are valid.) |
| — | When B'11 (setting prohibited) is written to the TS bit (Writes to the TS bit are valid.) |

Note: 'n' in Register Name is 0 or 1.

(9) Response packet error detection

This function reports response packet errors when the interface between the AXI and this module is accessed in ways that are not listed in Table 62.2.

(a) Transmitting response packet errors

When a response packet error occurs during accessing from the AXI to this module, the ERR_SNT bit in the USBDMAN_VCR register is set to 1.* The error signal is also sent to the AXI bus if the RM bit in the USBDMAN_DMAOR register is 0. If the RM bit is set to 1, the error signal is masked (0 is signaled).

Note: * When a response packet error occurs, the ERR_SNT bit is set to 1 regardless of the RM bit setting. The ERR_SNT bit is cleared by writing 0 after reading 1.

Note: 'n' in Register Name is 0 or 1.

Table 62.10 Response Packet Error Generation

| USBDMA0_DMAOR (RM bit) Setting | 1 (Mask Mode) | 0 (Normal Mode) |
|--|---|--|
| Response packet error signal notification to AXI | Signals the fixed value (0) to the AXI bus without determining if there is a response packet error. | Reports a response packet error to the AXI bus when it is accessed with a transfer size other than those supported in this module (4-byte read/write and 8-byte read). |

(b) Receiving response packet errors

When a response packet error occurs during accessing from this module to the AXI, the ERR_RCV bit in the USBDMAN_VCR register is set to 1. The TID bit in the USBDMAN_DMAOR register indicates on which channel of this module the error has occurred (TID[1] = 1 means it has occurred on CH1 and TID[0] = 1 on CH0).

The ERR_RCV bit is cleared by writing 0 after reading 1.

Note: 'n' in Register Name is 0 or 1.

62.3.2 DMA Transfer Function

(1) Operating mode

The following operating modes (a) to (b) are supported according to the setting of the PR bit in the DMA operation register (USBDMAn_DMAOR).

Note: 'n' in Register Name is 0 or 1.

(a) CH0 first control

Forces CH1 to wait to start a DMA transfer until a DMA transfer completes on CH0.

When there is a timing conflict in starting a DMA transfer between CH0 and CH1, CH0 takes precedence over CH1. If a DMA transfer is in progress on CH1 when CH0 tries to start a DMA transfer, CH0 waits to start it until the ongoing DMA transfer completes on CH1.

(b) CH1 first control

Forces CH0 to wait to start a DMA transfer until a DMA transfer completes on CH1.

When there is a timing conflict in starting a DMA transfer between CH1 and CH0, CH1 takes precedence over CH0. If a DMA transfer is in progress on CH0 when CH1 tries to start a DMA transfer, CH1 waits to start it until the ongoing DMA transfer completes on CH0.

Figure 62.11 shows an overview of the processing of the AXI bridge in each operating mode.

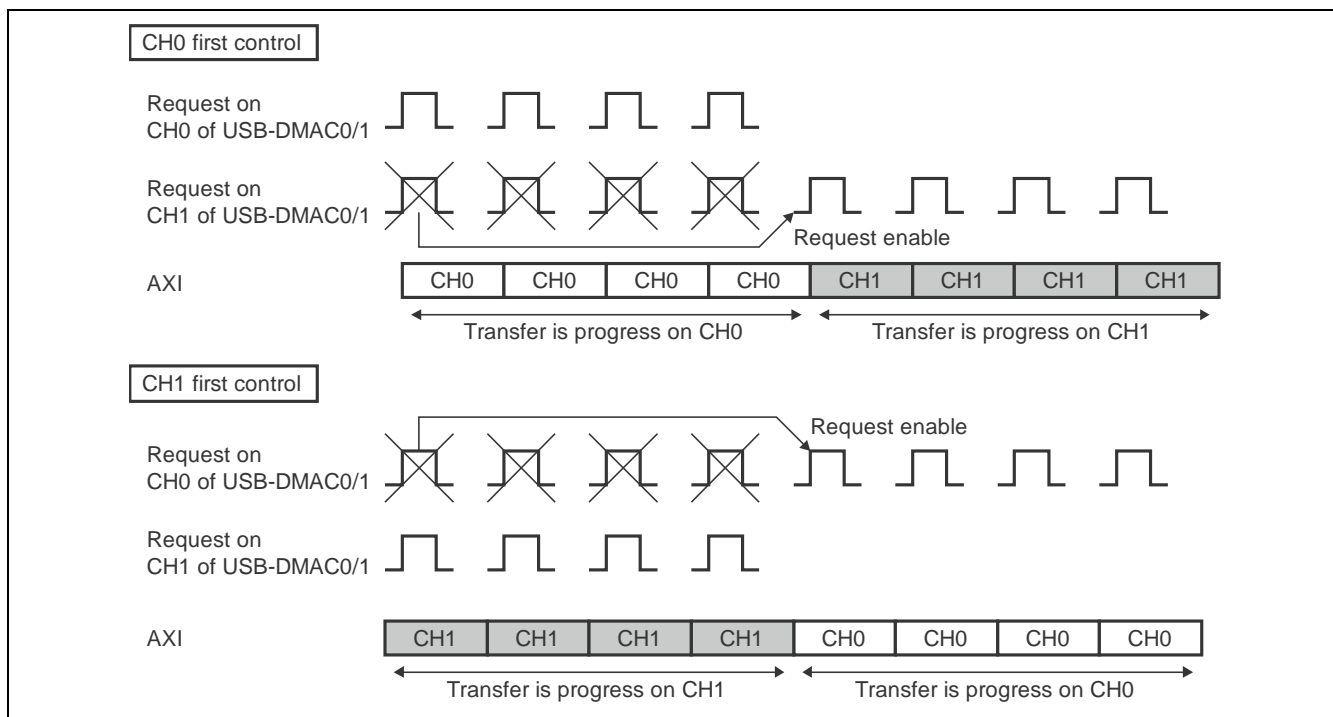


Figure 62.11 Overview of AXI Bridge Processing in Each Mode

(2) Transfer size

The transfer size on the AXI can be selected by setting the channel control register (the TS bit).

8-, 16-, and 32-byte can be set for each channel independently.

The following consideration should be observed during IN transfers.

The minimum transfer size that can be set to the TS bit is 8 bytes while the HS-USB can control transfers in 1-byte units. For example, if valid transfer data that is $8 \times (n - 1) + \alpha$ (α is less than 7 bytes) in size is to be transferred to the HS-USB with the transfer size set to 8 bytes and the transfer count set to n , α is transferred as a valid transfer unit by controlling byte enables according to the DMA final transaction valid data transfer enable register (the EDTEN bits) setting. Likewise, data is transferred in "transfer size x transfer count" (the final transaction is controlled by EDTEN) for 16- and 32-byte transfer sizes. The EDTEN setting is valid only for IN transfer and is don't care for OUT transfer.

Figure 62.12 shows an example of EDTEN setting. When only EDTEN31 in EDTEN31 to EDTEN0 is 1, one byte is enabled. When only EDTEN31 and EDTEN30 in EDTEN31 to EDTEN0 are 1, two bytes are enabled. Therefore, the total transfer volumes in the example shown above are calculated as "(32 bytes x (n - 1)) bytes + 10 bytes" for 32-byte transfer, "(16 bytes x (n - 1)) bytes + 7 bytes" for 16-byte transfer, and "(8 bytes x (n - 1)) bytes + 2 bytes" for 8-byte transfer. EDTEN15 to EDTEN0 in 16-byte transfer, EDTEN23 to EDTEN0 in 8-byte transfer are invalid (their values are ignored).

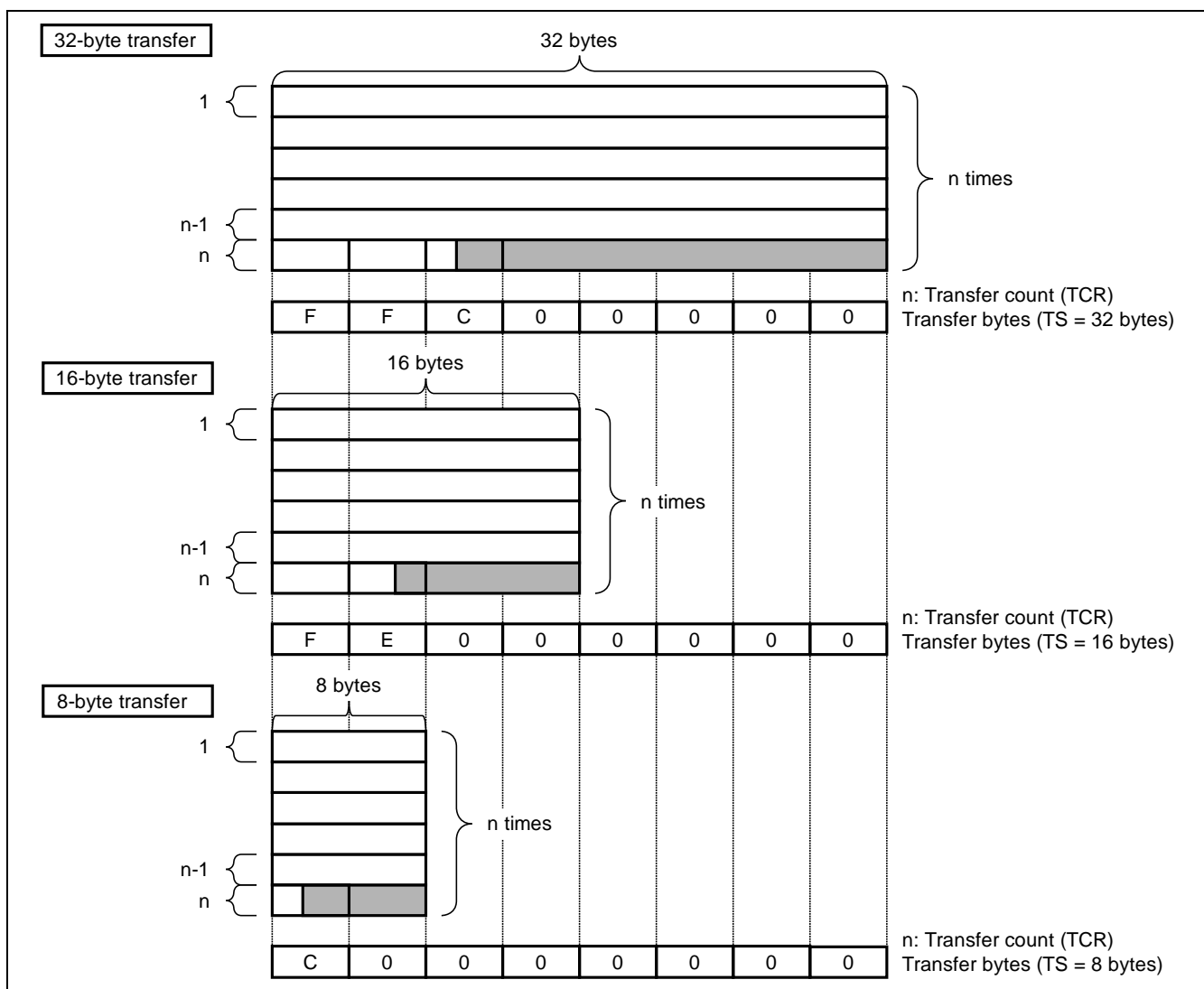


Figure 62.12 Example of Final Transaction Valid Data Transfer Enable (EDTEN) Setting

(3) DMA transfer flow

After the DMA source address register (USBDMAn_SAR), DMA destination address register (USBDMAn_DAR), DMA transfer count register (USBDMAn_TCR), DMA channel control register (USBDMAn_CHCR), final valid data transfer enable register (USBDMAn_TEND), and DMA operation register (USBDMAn_DMAOR) are set appropriately, the DMAC starts data transfer in the following sequence. The DMA timeout constant register (USBDMAn_TOCSTR) may also need to be set.

If the transfer enabling conditions are set as DE = 1, DME = 1, TE = 0, AE = 0, SP = 0, and (TR = 0, BUF = 0, RW = 0), transfers are performed according to the transfer request enables and transfer requests from the HS-USB. The transfer direction (IN/OUT transfer) is determined by the direction signal from the HS-USB. The transfer size is determined by the setting of TS1 to TS0.

Every time one AXI transaction (in the transfer size specified in the TS bit) completes, the internal counter of this module decrements the USBDMAn_TCR value by one. When the specified transfer count ends (the USBDMAn_TCR value reaches 0), it means all transfers complete. At this time, an USB-DMAC interrupt is generated if the IE bit in the USBDMAn_CHCR register is set to 1.

When a NULL packet is received, transfer can be stopped or restarted by setting FTE = 1 after negating the DREQE bit of the HS-USB module. Follow the steps described later in "Control when a NULL packet is received".

Note: 'n' in Register Name is 0 or 1.

Table 62.11 shows the conditions for each DMA transfer.

Table 62.11 DMA Transfer Conditions

| | |
|---|--|
| Transfer enabling condition | TE = 0, SP = 0, (*TR = 0, BUF = 0, RW = 0) |
| Transfer (starting) condition | When a transfer request is enabled by the HS-USB When a transfer request is issued by the HS-USB |
| Transfer completing (then restarting) condition | When the transfer count ends (the TE flag is set to 1), (can be restarted by setting DE and DME to 1.) When FTE = 1 (when DREQE = 1 in the HS-USB module is also set), and the steps described later should be followed. When reset (can enter into standby state by releasing the reset.) |
| Transfer suspending (then resuming) condition | When DE = 0 or DME = 0 (can be resumed by setting DE = 1 or DME = 1.) When SP = 1 (can be resumed by clearing the SP flag.) When RW = 1, BUF = 1, and TR = 1 (can be resumed by clearing the RW, BUF, and TR flag.)* When the module stops (can be resumed by resuming the module.) |

Note: * Can be ignored when the RWE, BUFE, and TRE bits are disabled (set to 0).

Figure 62.13 illustrates the high-level DMA transfer flow.

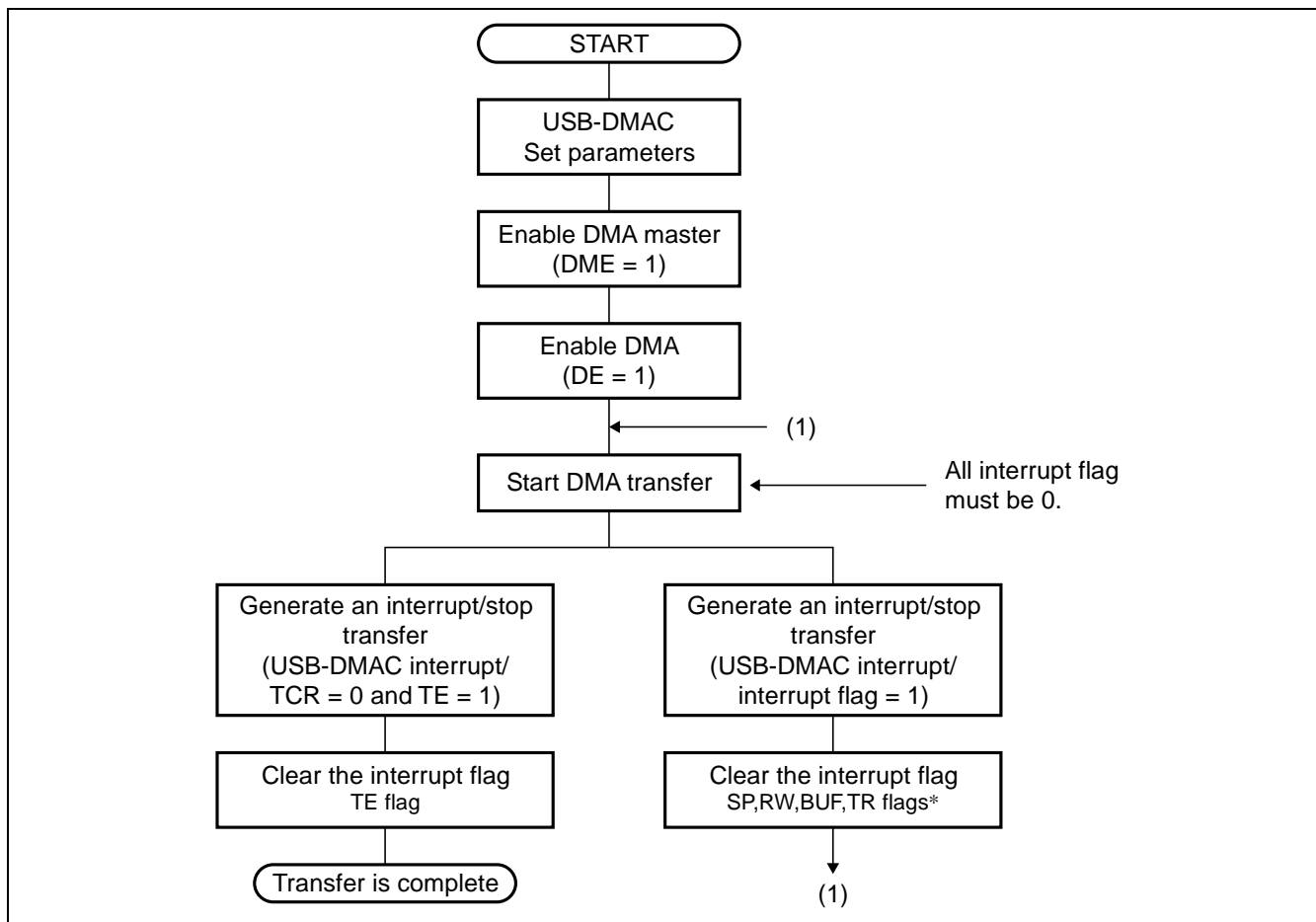


Figure 62.13 DMA Transfer Flow Overview

Note: * RW, BUF, and TR are not set when RWE, BUFE, and TRE are disabled

When to change USB-DMAC parameters

USB-DMAC parameters (such as destination address or transfer count) should be changed while all interrupt flags are cleared and DE = 0 and DME = 0 are set (a transfer is started with the latest parameters when DME = 1 and DE = 1 are set). Furthermore, USB-DMAC parameters should not be changed while the DMA is enabled (DME = 1 and DE = 1).

What to do if freeze

If a transfer should freeze during operation, issue a software reset after setting DE = 0 and DME = 0 and clearing all interrupt flags. All hardware except configuration registers is initialized. Restart by setting DME and DE to 1 after updating USB-DMAC parameters if necessary.

Control when a timeout occurs

Since this module assumes bus access in the transfer units listed in Table 62.2, it cannot be restarted when a timeout occurs in an irregular case where it is being accessed with any size that are not in the table. This irregular case should be handled by software, such as by issuing a software reset (to both CH0 and CH1).

Control when a NULL packet is received

When a NULL packet is received during a DMA transfer, the DMA transfer can be suspended and resumed on each channel independently by controlling the FTE bit of this module and the DREQE bit of the HS-USB module in the following sequence.

[Control sequence]

During a DMA transfer (A)

- (1) A NULL packet is received (an interrupt is generated and the NULL flag is detected).
- (2) The DREQE bit of the HS-USB module is negated (= 0).
- (3) Wait for the internal bus to be stabilized.
- (4) Perform three actions at the same time; setting the FTE bit, clearing the NULL bit and negating the DE bit.

An interrupt is generated due to TE.

- (5) Clear the interrupt caused by TE.
- (6) Set the parameters (such as TCR or DAR) for the next transfer.
- (7) Assert the DE bit (= 1). The DMA is restarted.
- (8) Assert the DREQE bit (= 1).

Start a DMA transfer (B).

[Restrictions]

1) FTE bit set timing

If this module is accessing the AXI (SHBSYx bit = 1) when setting the FTE bit, you need to wait to set the FTE bit until this module's internal buffer is empty and the AXI bus access is complete (SHBSYx bit = 0). Note that if you accidentally set the FTE bit while an AXI bus access is in progress, the transfer fails and transfer data may be lost or the transfer may be unable to be restarted.

2) DREQE bit assert (negate clear) timing

It is necessary to allow at least $20\text{clk}@S3D1\phi$ after the DE bit is asserted and before the DREQE is asserted. This is because DREQE needs to be asserted after transfers enabled by the DE bit are ready internally to be started. Asserting DREQE immediately after DE may cause malfunction such as initiating unnecessary bus access.

3) Fractional byte and NULL packet reception

This module assumes bus access in the transfer units listed in Table 62.2 and goes out of control when it receives a NULL packet in an unsupported transfer unit. Therefore, data transfer in any transfer unit that is not in the table is prohibited.

4) IN/OUT switching after NULL packet reception

After a NULL is received, transfers cannot be restarted with the transfer direction switched from OUT to IN. This is because this module still considers the terminated transfer as an OUT transfer until the transfer count reaches its end. Note that if you switch the transfer direction to IN transfer accidentally after a NULL packet is received, a conflict occurs between an IN and an OUT transfers, which may cause illegal bus access.

Use CH0 and CH1 with their transfer direction fixed, or if the transfer direction needs to be switched on the same channel, make sure to do it every time a transfer completes successfully due to the end of transfer count or is terminated by software reset.

Figure 62.14 shows the state transitions for DMA transfers.

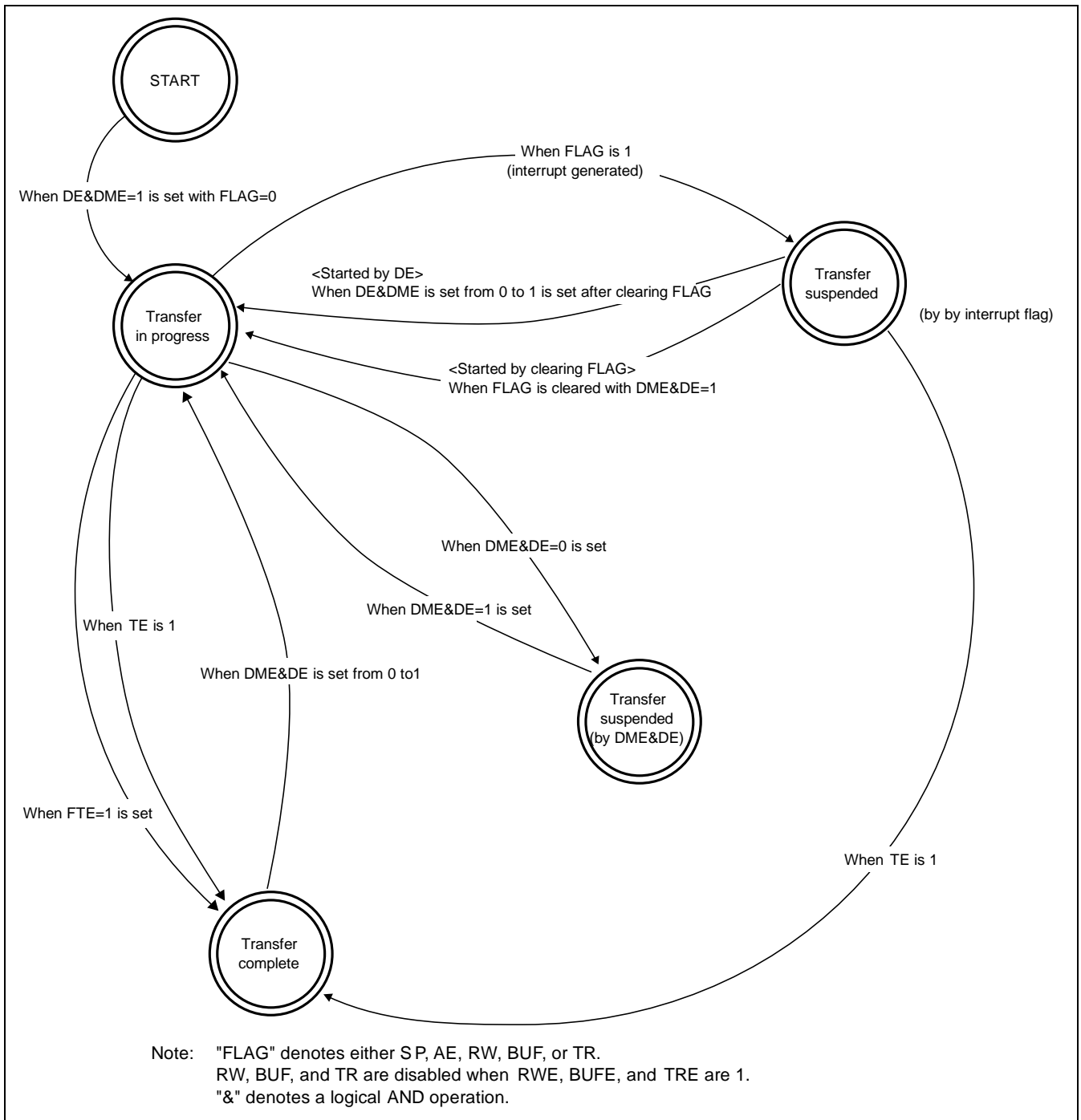


Figure 62.14 State Transitions for DMA Transfer

(4) Control sequence

Figure 62.15 shows an outline of the control sequence of this module.

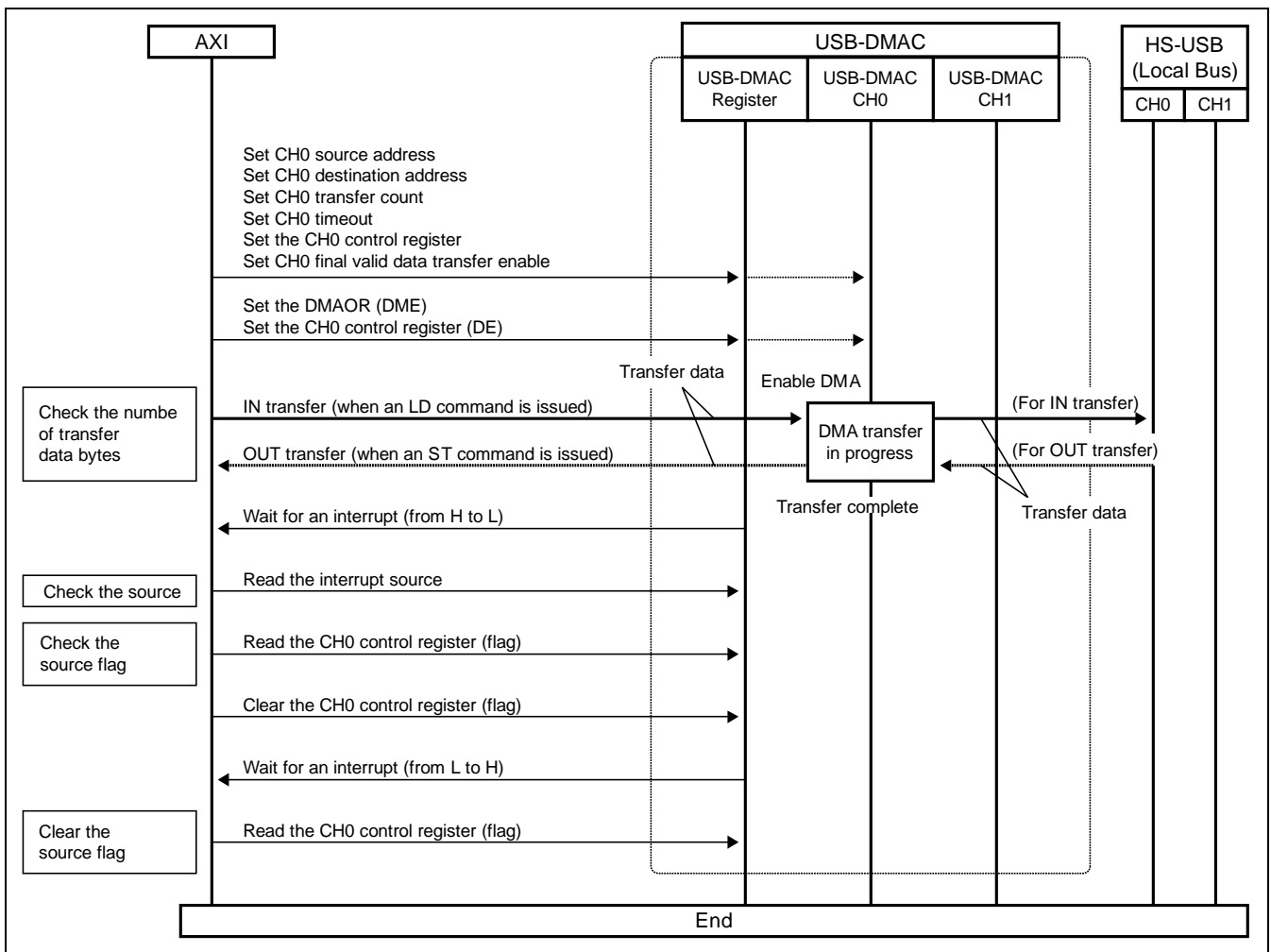


Figure 62.15 Control Sequence Overview

(5) Internal buffer configuration

This module has 64 bytes (64 bits × 8 words) of internal buffer for each of two channels. Thus, the total capacity of this module's internal buffers is 128 bytes (1,024 bits).

The internal buffers are composed of two planes each of which is organized as 64 bits × 4 words and are readable /writable on per-plane basis from the HS-USB local bus or AXI bus (plane 1 is writable when plane 0 is readable or vice versa). For 16-byte transfers, only 2 words in plane 0 and 2 words in plane 1 are used. For 8-byte transfers, only 1 word in plane 0 and 1 word in plane 1 are used.

Note that these internal buffers cannot be accessed directly by software.

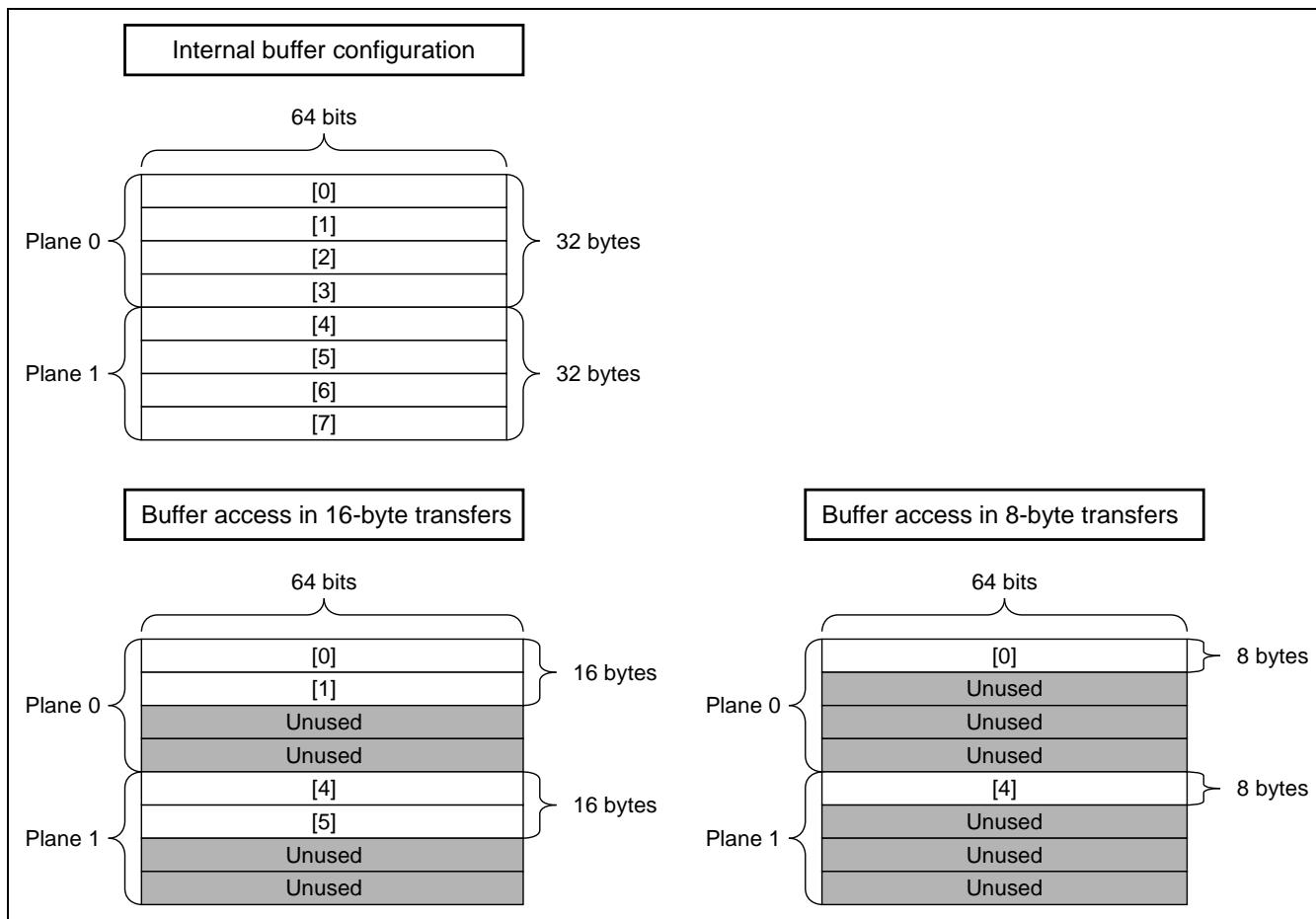


Figure 62.16 Internal Buffer Configuration and Used Area for Each Transfer Size

(6) Internal buffer control (hardware control, for reference)

(a) IN transfer (from AXI to USB)

Retrieves transfer data from a source on the AXI bus and writes it to this module's internal buffer. Reads the written data from the internal buffer and writes it to the HS-USB.

Figure 62.17 shows how the planes of this module's internal buffers are controlled during an IN transfer. In this figure, "read" and "write" indicate read/write operations to this module's internal buffers.

When a write to plane 0 or plane 1 is complete, the writable plane flag is toggled (every time 2 words* and 1 word* are written for 16-byte and 8-byte transfers, respectively). When a read from plane 0 or plane 1 and a transfer the read data to the HS-USB are complete, the readable plane flag is toggled (every time 2 words* and 1 word* are read for 16-byte and 8-byte transfers, respectively). Access to the AXI is prohibited during the period when both the writable plane flag and readable plane flag are set on the same plane as a read/write access conflict may occur.

Note: * 1 word = 64 bits

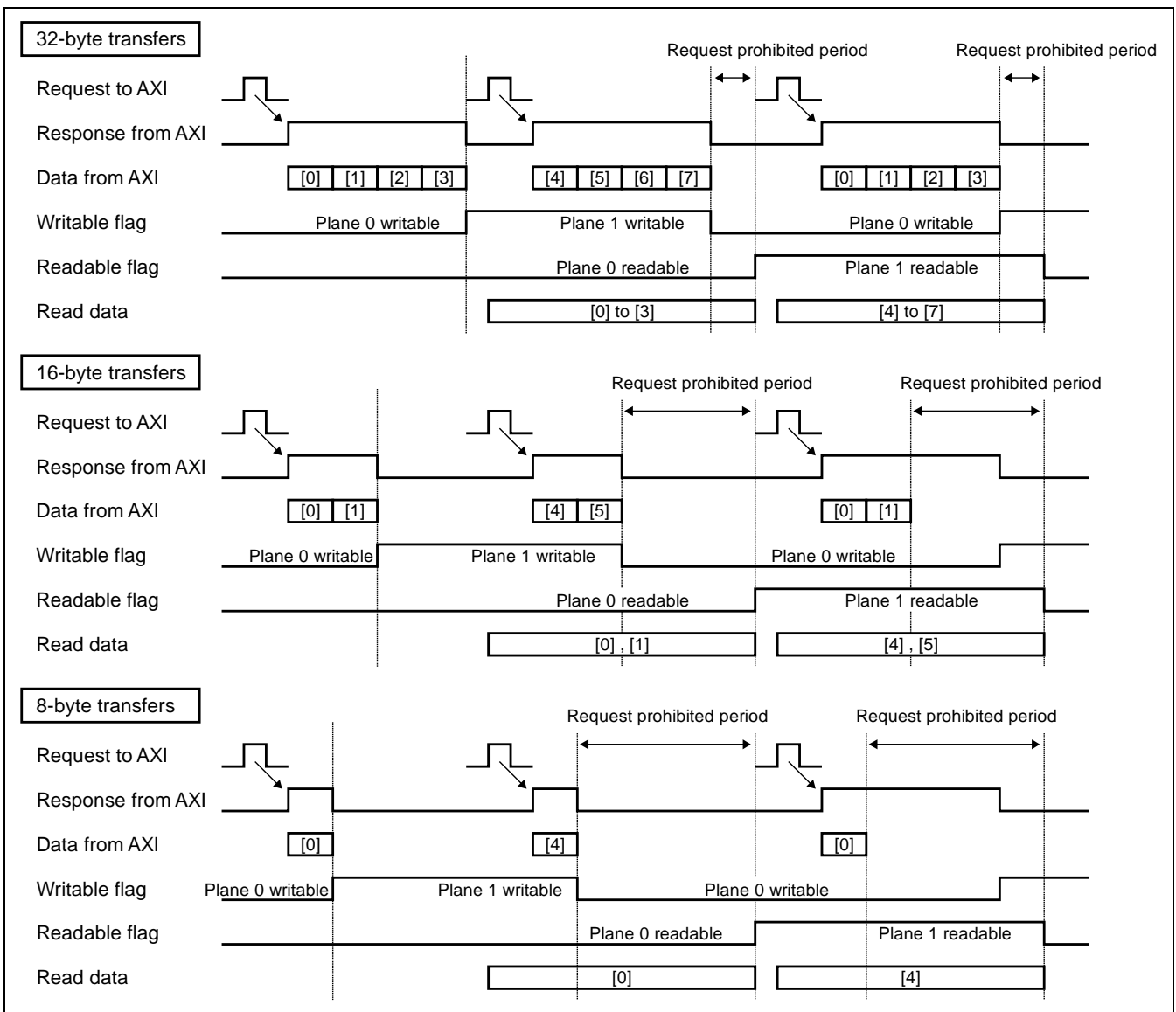


Figure 62.17 Read/Write Control for Internal Buffer: From AXI to USB (IN Transfer)

(b) OUT transfer (from USB to AXI)

Writes data read from the HS-USB to the internal buffers. Reads the written data from this module's internal buffers and writes it to the destination on the AXI bus (OUT transfer).

Figure 62.18 shows how the planes of this module's internal buffers are controlled during an OUT transfer. In this figure, "read" and "write" indicate read/write operations to this module's internal buffers.

When a write to plane 0 or plane 1 is complete, the writable plane flag is toggled (every time 2 words* and 1 word* are written for 16-byte and 8-byte transfers, respectively). When a read from plane 0 or plane 1 is complete, the readable plane flag is toggled (every time 2 words* and 1 word* are read for 16-byte and 8-byte transfers, respectively). Access to the AXI is prohibited during the period when both the writable plane flag and readable plane flag are set the same plane as a read/write access conflict may occur.

Note: * 1 word = 64 bits

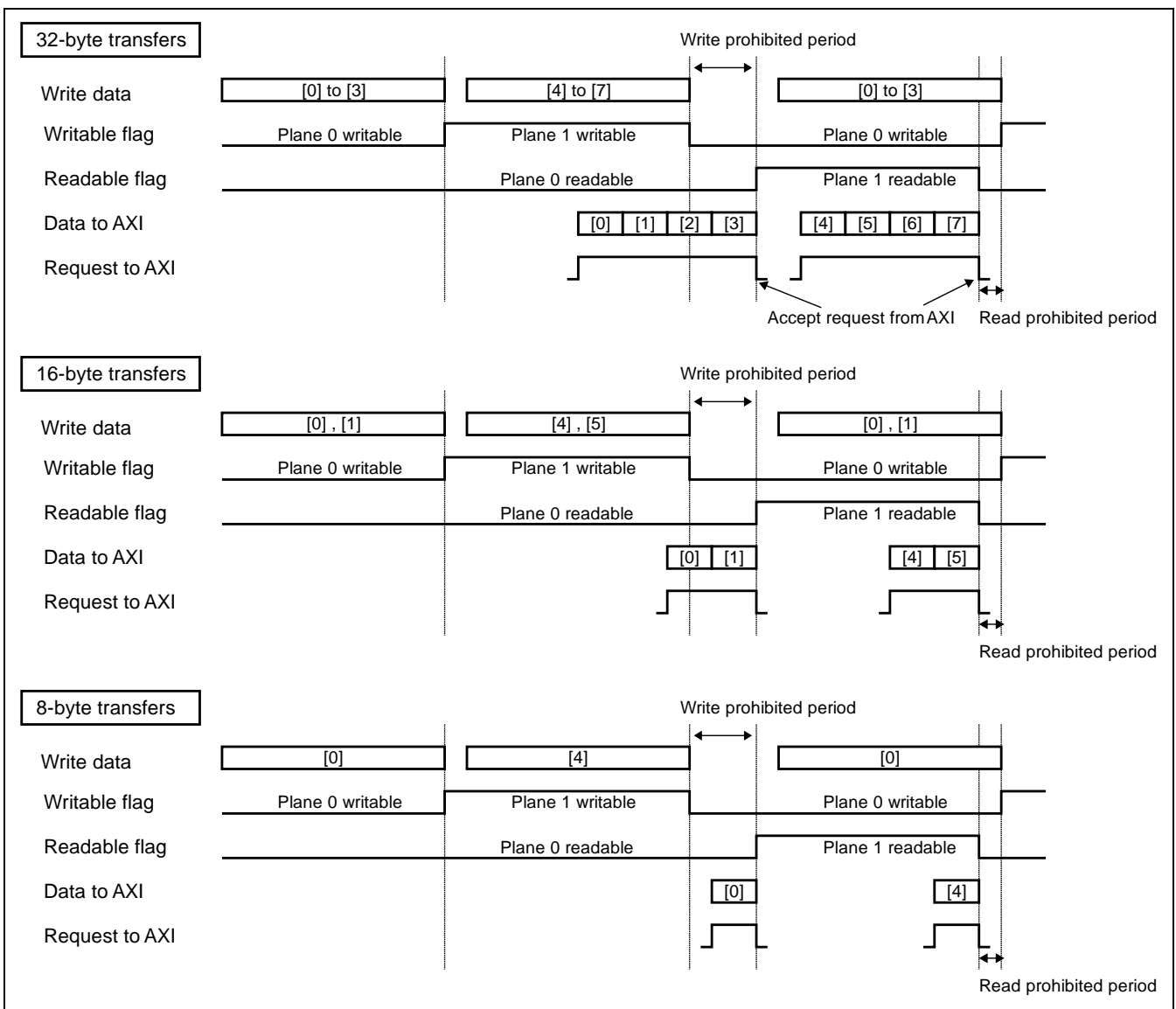


Figure 62.18 Read/Write Control for Internal Buffer: From USB to AXI (OUT Transfer)

62.4 DDM (Descriptor DMAC)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The DDM is a module that reconfigures the USB-DMAC. The OS on the main CPU is highly sophisticated and the overhead from interrupt notification to reconfiguration increases. To solve this problem, the DDM reconfigures the USB-DMAC in a fast response time on behalf of the CPU.

Figure 62.19 shows the relationship diagram of DDM USB-DMAC connection.

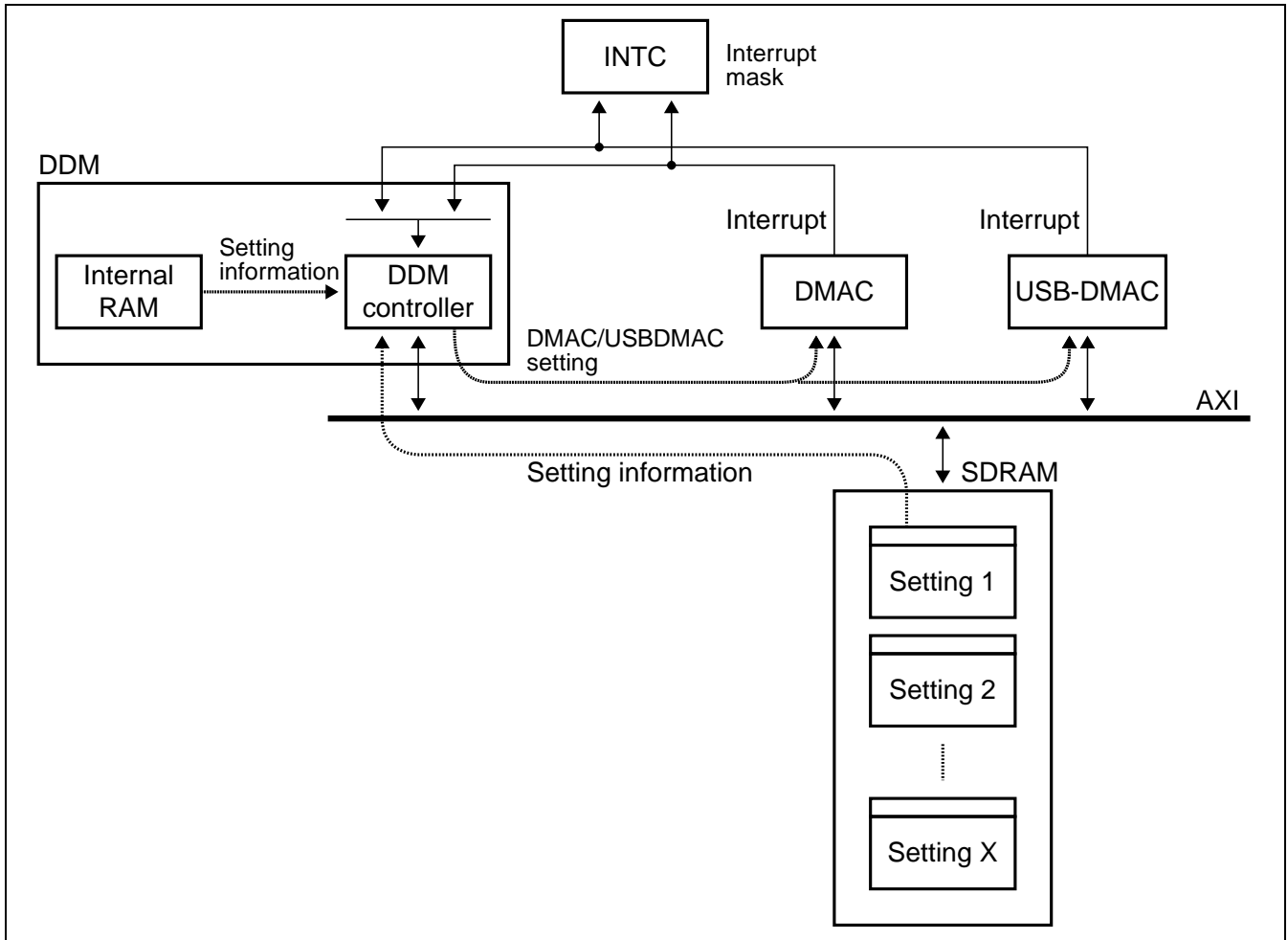


Figure 62.19 Relationship Diagram of DDM Connection

62.4.1 DDM Register Configuration

Table 62.12 shows the DDM register configuration. Table 62.13 shows the register states in each operating mode.

Table 62.12 DDM Control Registers

| Name | Register Abbreviation | R/W | Initial Value | Address | Access Size (Bits) | Second Generation RZ/G Series Products | | | |
|--------------------------------------|-----------------------|-----|---------------|-------------|--------------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DDM interrupt source mask register | DDIREQMSK | R/W | H'0000_0000 | H'E65C_0008 | 32 | √ | √ | √ | √ |
| DDM interrupt source register | DDIREQSTA | R | H'0000_0000 | H'E65C_000C | 32 | √ | √ | √ | √ |
| DDM CH1 start source mask register 1 | DDINTMSK11 | R/W | H'0000_0000 | H'E65C_0038 | 32 | √ | √ | √ | √ |
| DDM CH1 descriptor pointer | DDPTR1 | R/W | H'0000_0000 | H'E65C_0050 | 32 | √ | √ | √ | √ |
| DDM CH1 control register | DDCTRL1 | R/W | H'8000_0000 | H'E65C_0054 | 32 | √ | √ | √ | √ |
| DDM CH2 start source mask register 1 | DDINTMSK21 | R/W | H'0000_0000 | H'E65C_0068 | 32 | √ | √ | √ | √ |
| DDM CH2 descriptor pointer | DDPTR2 | R/W | H'0000_0000 | H'E65C_0080 | 32 | √ | √ | √ | √ |
| DDM CH2 control register | DDCTRL2 | R/W | H'8000_0000 | H'E65C_0084 | 32 | √ | √ | √ | √ |
| DDM C start source mask register 1 | DDINTMSK31 | R/W | H'0000_0000 | H'E65C_0098 | 32 | √ | √ | √ | √ |
| DDM C descriptor pointer | DDPTR3 | R/W | H'0000_0000 | H'E65C_00B0 | 32 | √ | √ | √ | √ |
| DDM C control register | DDCTRL3 | R/W | H'8000_0000 | H'E65C_00B4 | 32 | √ | √ | √ | √ |
| DDM CH4 start source mask register 1 | DDINTMSK41 | R/W | H'0000_0000 | H'E65C_00C8 | 32 | √ | √ | √ | √ |
| DDM CH4 descriptor pointer | DDPTR4 | R/W | H'0000_0000 | H'E65C_00E0 | 32 | √ | √ | √ | √ |
| DDM CH4 control register | DDCTRL4 | R/W | H'8000_0000 | H'E65C_00E4 | 32 | √ | √ | √ | √ |
| DDM CH5 start source mask register 1 | DDINTMSK51 | R/W | H'0000_0000 | H'E65C_00F8 | 32 | √ | √ | √ | √ |
| DDM CH5 descriptor pointer | DDPTR5 | R/W | H'0000_0000 | H'E65C_0110 | 32 | √ | √ | √ | √ |
| DDM CH5 control register | DDCTRL5 | R/W | H'8000_0000 | H'E65C_0114 | 32 | √ | √ | √ | √ |
| DDM CH6 start source mask register 1 | DDINTMSK61 | R/W | H'0000_0000 | H'E65C_0128 | 32 | √ | √ | √ | √ |
| DDM CH6 descriptor pointer | DDPTR6 | R/W | H'0000_0000 | H'E65C_0140 | 32 | √ | √ | √ | √ |
| DDM CH6 control register | DDCTRL6 | R/W | H'8000_0000 | H'E65C_0144 | 32 | √ | √ | √ | √ |

| Name | Register Abbreviation | R/W | Initial Value | Address | Access Size (Bits) | Second Generation RZ/G Series Products | | | |
|--------------------------------------|-----------------------|-----|---------------|-------------|--------------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DDM CH7 start source mask register 1 | DDINTMSK71 | R/W | H'0000_0000 | H'E65C_0158 | 32 | √ | √ | √ | √ |
| DDM CH7 descriptor pointer | DDPTR7 | R/W | H'0000_0000 | H'E65C_0170 | 32 | √ | √ | √ | √ |
| DDM CH7 control register | DDCTRL7 | R/W | H'8000_0000 | H'E65C_0174 | 32 | √ | √ | √ | √ |
| DDM CH8 start source mask register 1 | DDINTMSK81 | R/W | H'0000_0000 | H'E65C_0188 | 32 | √ | √ | √ | √ |
| DDM CH8 descriptor pointer | DDPTR8 | R/W | H'0000_0000 | H'E65C_01A0 | 32 | √ | √ | √ | √ |
| DDM CH8 control register | DDCTRL8 | R/W | H'8000_0000 | H'E65C_01A4 | 32 | √ | √ | √ | √ |

Table 62.13 Register States in Each Operating Mode

| Abbreviation | Power-On Reset | Module Standby |
|---------------------|-----------------------|-----------------------|
| DDIREQMSK | Initialized | Retained |
| DDIREQSTA | Initialized | Retained |
| DDINTMSK11 | Initialized | Retained |
| DDPTR1 | Initialized | Retained |
| DDCTRL1 | Initialized | Retained |
| DDINTMSK21 | Initialized | Retained |
| DDPTR2 | Initialized | Retained |
| DDCTRL2 | Initialized | Retained |
| DDINTMSK31 | Initialized | Retained |
| DDPTR3 | Initialized | Retained |
| DDCTRL3 | Initialized | Retained |
| DDINTMSK41 | Initialized | Retained |
| DDPTR4 | Initialized | Retained |
| DDCTRL4 | Initialized | Retained |
| DDINTMSK51 | Initialized | Retained |
| DDPTR5 | Initialized | Retained |
| DDCTRL5 | Initialized | Retained |
| DDINTMSK61 | Initialized | Retained |
| DDPTR6 | Initialized | Retained |
| DDCTRL6 | Initialized | Retained |
| DDINTMSK71 | Initialized | Retained |
| DDPTR7 | Initialized | Retained |
| DDCTRL7 | Initialized | Retained |
| DDINTMSK81 | Initialized | Retained |
| DDPTR8 | Initialized | Retained |
| DDCTRL8 | Initialized | Retained |

62.4.2 DDM Register Description

62.4.2.1 DDM Interrupt Source Mask Register (DDIREQMSK)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The DDM Interrupt Source Mask Register enables and disables interrupt sources from the DDM.

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | CM8 | CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | ERR8 | END8 | ERR7 | END7 | ERR6 | END6 | ERR5 | END5 | ERR4 | END4 | ERR3 | END3 | ERR2 | END2 | ERR1 | ENS1 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|---|
| 31 to 24 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 | CM8 | B'0 | R/W | Interrupt Output by CH8 Counter consistent 0: Disables interrupt output. 1: Enables interrupt output. |
| 22 | CM7 | B'0 | R/W | Interrupt Output by CH7 Counter consistent 0: Disables interrupt output. 1: Enables interrupt output. |
| 21 | CM6 | B'0 | R/W | Interrupt Output by CH6 Counter consistent 0: Disables interrupt output. 1: Enables interrupt output. |
| 20 | CM5 | B'0 | R/W | Interrupt Output by CH5 Counter consistent 0: Disables interrupt output. 1: Enables interrupt output. |
| 19 | CM4 | B'0 | R/W | Interrupt Output by CH4 Counter consistent 0: Disables interrupt output. 1: Enables interrupt output. |
| 18 | CM3 | B'0 | R/W | Interrupt Output by CH3 Counter consistent 0: Disables interrupt output. 1: Enables interrupt output. |
| 17 | CM2 | B'0 | R/W | Interrupt Output by CH2 Counter consistent 0: Disables interrupt output. 1: Enables interrupt output. |
| 16 | CM1 | B'0 | R/W | Interrupt Output by CH1 Counter consistent 0: Disables interrupt output. 1: Enables interrupt output. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|--|
| 15 | ERR8 | B'0 | R/W | Interrupt Output by CH8 DDM_ERR 0: Disables interrupt output. 1: Enables interrupt output. |
| 14 | END8 | B'0 | R/W | Interrupt Output by CH8 DDM_END 0: Disables interrupt output. 1: Enables interrupt output. |
| 13 | ERR7 | B'0 | R/W | Interrupt Output by CH7 DDM_ERR 0: Disables interrupt output. 1: Enables interrupt output. |
| 12 | END7 | B'0 | R/W | Interrupt Output by CH7 DDM_END 0: Disables interrupt output. 1: Enables interrupt output. |
| 11 | ERR6 | B'0 | R/W | Interrupt Output by CH6 DDM_ERR 0: Disables interrupt output. 1: Enables interrupt output. |
| 10 | END6 | B'0 | R/W | Interrupt Output by CH6 DDM_END 0: Disables interrupt output. 1: Enables interrupt output. |
| 9 | ERR5 | B'0 | R/W | Interrupt Output by CH5 DDM_ERR 0: Disables interrupt output. 1: Enables interrupt output. |
| 8 | END5 | B'0 | R/W | Interrupt Output by CH5 DDM_END 0: Disables interrupt output. 1: Enables interrupt output. |
| 7 | ERR4 | B'0 | R/W | Interrupt Output by CH4 DDM_ERR 0: Disables interrupt output. 1: Enables interrupt output. |
| 6 | END4 | B'0 | R/W | Interrupt Output by CH4 DDM_END 0: Disables interrupt output. 1: Enables interrupt output. |
| 5 | ERR3 | B'0 | R/W | Interrupt Output by CH3 DDM_ERR 0: Disables interrupt output. 1: Enables interrupt output. |
| 4 | END3 | B'0 | R/W | Interrupt Output by CH3 DDM_END 0: Disables interrupt output. 1: Enables interrupt output. |
| 3 | ERR2 | B'0 | R/W | Interrupt Output by CH2 DDM_ERR 0: Disables interrupt output. 1: Enables interrupt output. |
| 2 | END2 | B'0 | R/W | Interrupt Output by CH2 DDM_END 0: Disables interrupt output. 1: Enables interrupt output. |
| 1 | ERR1 | B'0 | R/W | Interrupt Output by CH1 DDM_ERR 0: Disables interrupt output. 1: Enables interrupt output. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|------------|-----------------|----------------------|------------|--|
| 0 | END1 | B'0 | R/W | Interrupt Output by CH1 DDM_END 0: Disables interrupt output. 1: Enables interrupt output. |

62.4.2.2 DDM Interrupt Source Register (DDIREQSTA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The DDM Interrupt Source Register indicates the currently output interrupt source. This register is read-only. The interrupt source isn't cleared by this register. Clear the interrupt source by using the control register of each channel.

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | CM8 | CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | ERR8 | END8 | ERR7 | END7 | ERR6 | END6 | ERR5 | END5 | ERR4 | END4 | ERR3 | END3 | ERR2 | END2 | ERR1 | ENS1 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|----------|----------|---------------|-----|--|
| 31 to 24 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 23 | CM8 | B'0 | R | Interrupt output by CH8 Counter consistent 0: The interrupt is not output. 1: The interrupt is output. |
| 22 | CM7 | B'0 | R | Interrupt output by CH7 Counter consistent 0: The interrupt is not output. 1: The interrupt is output. |
| 21 | CM6 | B'0 | R | Interrupt output by CH6 Counter consistent 0: The interrupt is not output. 1: The interrupt is output. |
| 20 | CM5 | B'0 | R | Interrupt output by CH5 Counter consistent 0: The interrupt is not output. 1: The interrupt is output. |
| 19 | CM4 | B'0 | R | Interrupt output by CH4 Counter consistent 0: The interrupt is not output. 1: The interrupt is output. |
| 18 | CM3 | B'0 | R | Interrupt output by CH3 Counter consistent 0: The interrupt is not output. 1: The interrupt is output. |
| 17 | CM2 | B'0 | R | Interrupt output by CH2 Counter consistent 0: The interrupt is not output. 1: The interrupt is output. |
| 16 | CM1 | B'0 | R | Interrupt output by CH1 Counter consistent 0: The interrupt is not output. 1: The interrupt is output. |

| Bit | Bit Name | Initial Value | R/W | Descriptions |
|-----|----------|---------------|-----|---|
| 15 | ERR8 | B'0 | R | Interrupt output by CH8 DDM_ERR 0: The interrupt is not output. 1: The interrupt is output. |
| 14 | END8 | B'0 | R | Interrupt output by CH8 DDM_END 0: The interrupt is not output. 1: The interrupt is output. |
| 13 | ERR7 | B'0 | R | Interrupt output by CH7 DDM_ERR 0: The interrupt is not output. 1: The interrupt is output. |
| 12 | END7 | B'0 | R | Interrupt output by CH7 DDM_END 0: The interrupt is not output. 1: The interrupt is output. |
| 11 | ERR6 | B'0 | R | Interrupt output by CH6 DDM_ERR 0: The interrupt is not output. 1: The interrupt is output. |
| 10 | END6 | B'0 | R | Interrupt output by CH6 DDM_END 0: The interrupt is not output. 1: The interrupt is output. |
| 9 | ERR5 | B'0 | R | Interrupt output by CH5 DDM_ERR 0: The interrupt is not output. 1: The interrupt is output. |
| 8 | END5 | B'0 | R | Interrupt output by CH5 DDM_END 0: The interrupt is not output. 1: The interrupt is output. |
| 7 | ERR4 | B'0 | R | Interrupt output by CH4 DDM_ERR 0: The interrupt is not output. 1: The interrupt is output. |
| 6 | END4 | B'0 | R | Interrupt output by CH4 DDM_END 0: The interrupt is not output. 1: The interrupt is output. |
| 5 | ERR3 | B'0 | R | Interrupt output by CH3 DDM_ERR 0: The interrupt is not output. 1: The interrupt is output. |
| 4 | END3 | B'0 | R | Interrupt output by CH3 DDM_END 0: The interrupt is not output. 1: The interrupt is output. |
| 3 | ERR2 | B'0 | R | Interrupt output by CH2 DDM_ERR 0: The interrupt is not output. 1: The interrupt is output. |
| 2 | END2 | B'0 | R | Interrupt output by CH2 DDM_END 0: The interrupt is not output. 1: The interrupt is output. |
| 1 | ERR1 | B'0 | R | Interrupt output by CH1 DDM_ERR 0: The interrupt is not output. 1: The interrupt is output. |

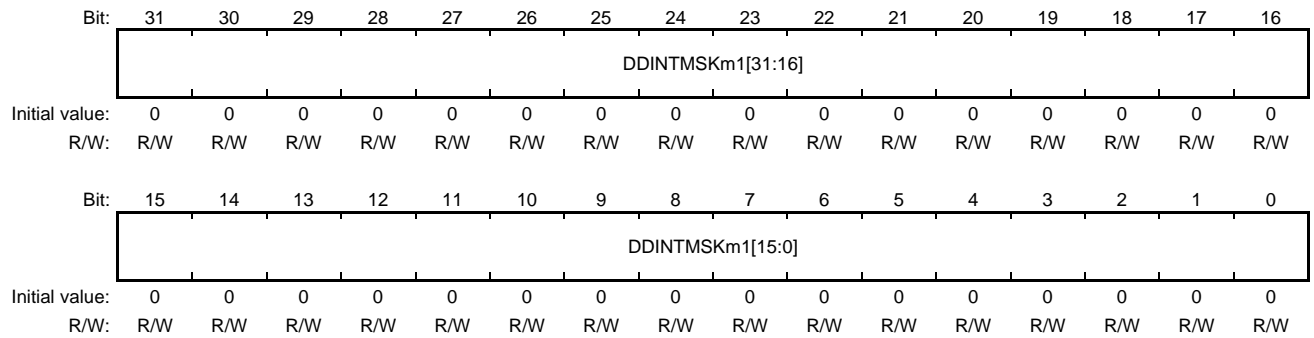
| Bit | Bit Name | Initial Value | R/W | Descriptions |
|------------|-----------------|----------------------|------------|---|
| 0 | END1 | B'0 | R | Interrupt output by CH1 DDM_END 0: The interrupt is not output. 1: The interrupt is output. |

62.4.2.3 DDM CHm Start Source Mask Registers (DDINTMSK_m1)

| | | | |
|-------------|-------------------------------|-------------|-------------|
| RZ/G2H √ | RZ/G2M V1.3, RZ/G2M V3.0 √ | RZ/G2N √ | RZ/G2E √ |
|-------------|-------------------------------|-------------|-------------|

DDM CHm Start Source Mask Registers 1 mask DDM start sources.

Note: “m” of Channels are 1 to 8.



| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|-------------------------------|---------------|-----|---|
| 31 to 0 | DDINTMSK _m 1[31:0] | H'0000_0000 | R/W | Start Source Mask 0: Does not mask the start source. 1: Masks the start source. |

The current source assignment is as follows:

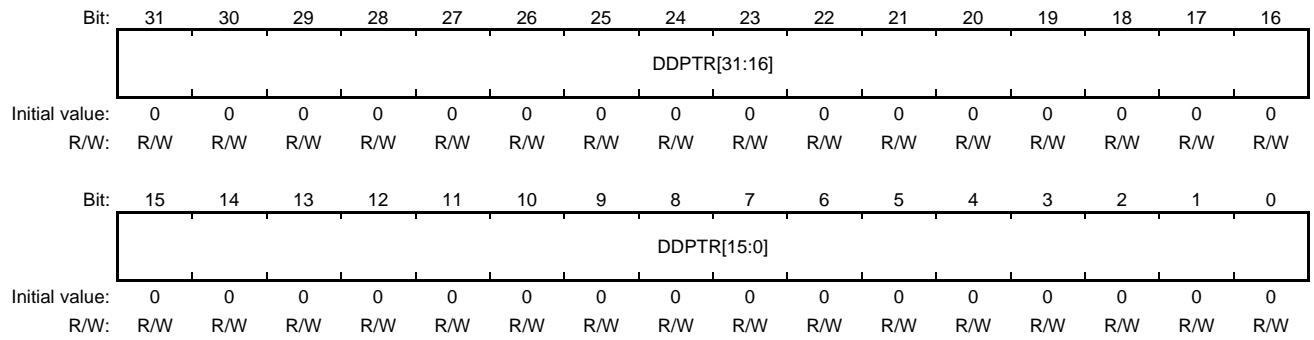
| Bit Name | Bit | Assigned Start Source |
|-------------------------|---------|-----------------------|
| DDINTMSK _m 1 | 31 to 2 | Reserved |
| | 1 | USB-DMAC1 Interrupt |
| | 0 | USB-DMAC0 Interrupt |

62.4.2.4 DDM CHm Descriptor Pointer (DDPTRm)

| | | | |
|-------------|-------------------------------|-------------|-------------|
| RZ/G2H √ | RZ/G2M V1.3, RZ/G2M V3.0 √ | RZ/G2N √ | RZ/G2E √ |
|-------------|-------------------------------|-------------|-------------|

The DDM CHm Descriptor Pointer indicates the start address of setting code allocated onto memory.

Note: “m” of Channels are 1 to 8.



| Bit | Bit Name | Initial Value | R/W | Descriptions |
|---------|-------------|---------------|-----|---------------------|
| 31 to 0 | DDPTR[31:0] | H'0000_0000 | R/W | Descriptor pointer. |

62.4.2.5 DDM CHm Control Register (DDCTRLm)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

The DDM CHm Control Register controls start and stop of DDM.

Note: “m” of Channels are 1 to 8.

| | | | | | | | | | | | | | | | | |
|----------------|--------------|-----|-----|-----|-----|-----|-----------|--------------|---------|-----|------|------|-------------|-------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | PRI3 to PRI0 | | | | — | — | ENDC M | ENDC MMSK | END_NUM | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | END_CNT | | | | | | | — | — | — | STOP | KICK | DDM_ ERR | DDM_ END | DDE | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|---|
| 31 to 28 | PRI3 to PRI0 | H'0 | R/W | PRI value on the AXI bus. |
| 27, 26 | — | B'00 | R | Reserved |
| 25 | ENDCM | B'0 | R/W | Descriptor end Counter Consistent Detection Writing 0 is possible to clear the flag. 0: Counter consistent is not detected. 1: Counter consistent is detected. |
| 24 | ENDCMMSK | B'0 | R/W | Descriptor end Counter Consistent Detection Mask 0: Masks the ENDCM. (END_CNT is cleared to 0) 1: Does not mask the ENDCM. |
| 23 to 16 | END_NUM | All 0 | R/W | Number of descriptor ends |
| 15 to 8 | END_CNT | All 0 | R/W | Descriptor end Counter |
| 7 to 5 | — | B'000 | R | Reserved |
| 4 | STOP | B'0 | R/W | Forced Stop Function 0: Does not perform forced stop. 1: Performs forced stop. |
| 3 | KICK | B'0 | R/W | Forced Start Function without Start Source 0: Does not perform forced start. 1: Performs forced start. This bit is automatically cleared to 0 after 1 is written. |
| 2 | DDM_ERR | B'0 | R | Error Occurrence. The error is cleared by setting the DDE bit to 0. 0: No error occurred. 1: An error occurred. |
| 1 | DDM_END | B'0 | R | DDM End. DDM end is cleared by setting the DDE bit to 0. 0: The DDM did not end. 1: The DDM ended. |
| 0 | DDE | B'0 | R/W | Start Enable 0: Does not enable DDM start. 1: Enables DDM start according to the start source. |

62.4.3 Internal RAM

| Address | Area |
|-------------|--------------------|
| H'E65C_0000 | Register_area |
| H'E65C_8000 | Code RAM 1 KB |
| H'E65C_C000 | Data RAM 1.5 KB |

Figure 62.20 Internal RAM

DDM builds in RAM. It is the 2nd page composition of Code RAM and Data RAM. DDM can access without AXI. Access size is arbitrary. In addition, a functional difference is not in both depending on the way of calling depending on which Code RAM and Data RAM were conscious of the use.

62.4.4 Principles of Operation

To reconfigure the DMAC by using the DDM, use the following procedure:

1. Describing setting codes
2. Starting and ending the DDM

(1) Describing setting codes

(a) Data structure

A setting code is fixed-length data consisting of four longwords.

Setting code data consists of the following fields, starting from the beginning:

| Field | bit |
|-----------------|---------|
| Operation field | 32 bits |
| Address field | 32 bits |
| Data field | 32 bits |
| Bit mask field | 32 bits |

Note: Arrange a setting code in 16 bytes boundary.

The following describes the operation field.

(b) Operation field

The structure of the operation field is as follows:

| Bit | Bit Name | Descriptions |
|----------|----------|---|
| 31 to 16 | HEADER | Indicates DDM setting data. H'DDDD: Normal DDM setting code Usually, set this value. H'EEEE: DDM end code Set this value when ending the channel without starting the DDM the next time. If the HEADER bits have other values, a DDM_ERR interrupt occurs. The channel of the DDM stops immediately. |
| 15 to 8 | OPERAND | Operand attribute B'00xyyzz B'00: immediately B'01: register B'10: pointer B'11: pointer (Register value) xx: Bit mask field yy: Data field zz: Address field |
| 7 to 4 | SIZE | Specify the data access size to be set. B'0000: Bytes B'0001: Words B'0010: Longwords |
| 3 | ENDFLG | End Flag of Setting Processing Based on This Start Source 0: Continuation 1: Setting end. The DDM waits for the next start source. |
| 2 to 0 | OPC | Operation Code B'000: read (cmp/eq) (read) B'001: write (write) B'010: rmodw (read modify write) B'011: add (addition) B'100: jmp/eq (branch) B'101: ror (rotate right) Others: DDM_ERR interrupt occurs |

The address field, data field, and bit mask field each consist of 32 bits. The data field and bit mask field use the LSB side for word- and byte-size access.

Four of register R0-R3 of longword size can be used for every Ch.

(c) read (cmp/eq)

The read (cmp/eq) code reads the target address and compares the address with the expected values written in the data field and bit mask field.

When the comparison result becomes true, the TFLG internal variable is set to 1. TFLG is cleared to 0 when jmp/eq is executed or operation with ENDFLG is performed.

Description example1:

The read (cmp/eq) code in this example reads the H'FE00_802C address in longwords and checks if bit 1 is set to 1.

```
H'DDDD_0020
H'FE00_802C
H'0000_0002
H'0000_0002
```

Description example2:

The read (cmp/eq) code in this example reads the R1 in longwords and checks if bit 1 is set to 1.

```
H'DDDD_0120
H'0000_0001
H'0000_0002
H'0000_0002
```

(d) write

The write code writes data to the target address.

Description example1:

The write code in this example accesses the H'FE00_802C address in longwords and writes H'0010_5400.

```
H'DDDD_0021
H'FE00_802C
H'0010_5400
H'FFFF_FFFF (don't care)
```

Description example2:

The write code in this example accesses the H'FE00_802C address in longwords and writes the value of H'0000_0001 address to H'FE00_802C address.

```
H'DDDD_0821
H'FE00_802C
H'0000_0001
H'FFFF_FFFF (don't care)
```

(2) Description example3:

The write code in this example accesses the H'FE00_802C address in longwords and writes the value of pointer of R1 to the H'FE00_802C address.

```
H'DDDD_0C21
H'FE00_802C
H'0000_0001
H'FFFF_FFFF (don't care)
```

(a) rmodw

The rmodw code reads the target address and performs modify-write according to the values of the bit mask field.

Description example:

The rmodw code in this example reads the H'FE00_802C address in longwords and applies H'54 modify-write to bits 11 to 4.

```
H'DDDD_0022
H'FE00_802C
H'0000_0540
H'0000_0FF0
```

(b) add

The add code reads the target address and performs add-modify-write according to the values of the bit mask field.

Description example:

The add code in this example reads the H'FE00_802C address in longwords and applies H'200 add-modify-write to bits 11 to 0.

```
H'DDDD_0023
H'FE00_802C
H'0000_0200
H'0000_0FFF
```

Use 0 as the addition data corresponding to the area where the bit mask is 0.

(c) jmp/eq

If TFLG is set to 1 by the cmp/eq code, jmp/eq changes the descriptor pointer to the values of the data field. Descriptor pointer change by jmp/eq is not reflected in execution based on the next start source. Changing the descriptor pointer for the next start requires that the CHm descriptor pointer be reconfigured.

Note: “m” of Channels are 1 to 8.

Description example:

The jmp/eq code in this example changes the descriptor pointer when TFLG is 1 to H'E558_000E.

```
H'DDDD_0024
H'0000_0000 (don't care)
H'E558_000E
H'0000_0000 (don't care)
```


(d) ror

The ror code reads the target address and performs rotate-right-write according to the values of the bit mask field. The shifted lower bits are rotated to higher bits. The ror command shifts according to the value of the data field.

Description example:

The ror code in this example reads the H'FE00_802C address in longwords and applies 8-bits right-shift.

```
H'DDDD_0025
H'FE00_802C
H'0000_0008
H'00FF_FFFF
```

(e) Settings description order

Describe settings in the following order:

- (1) Clearing the start source (source output side, not the DDM start source mask register)
- (2) Changing the DDM source mask setting as required
- (3) Reconfiguring registers such as DMAC
- (4) Reconfiguring the next descriptor pointer

If several start sources are set, the last write may be unable to catch up with the next start. In such a case, end next descriptor pointer setting in (4) in read after write (do not end it in write).

(f) Example of setting DMAC restart

Use DDM CH1. The following shows an example of describing setting codes for reconfiguration when the TE bit of the DMAC1 CH0 channel control register is HI.

```
//TE bit read
H'DDDD_0020
H'FE00_802C
H'0000_0002
H'0000_0002
//TE bit clear
H'DDDD_0021
H'FE00_802C
H'0010_5400
H'FFFF_FFFF
//SAR setting
H'DDDD_0021
H'FE00_8020
H'5800_0000
H'FFFF_FFFF
//DAR setting
H'DDDD_0021
H'FE00_8024
H'0000_0000
H'FFFF_FFFF
//TCR setting
H'DDDD_0021
```

```

H'FE00_8028
H'0000_0010
H'FFFF_FFFF
//DE bit write. DMA start
H'DDDD_0021
H'FE00_802C
H'0010_5401
H'FFFF_FFFF
//Next descriptor pointer write.
H'DDDD_0021
H'E65C_0050
H'5800_02D0
H'FFFF_FFFF
// Setting end by this start. (dummy read)
H'DDDD_0028
H'E65C_0050
H'0000_0000
H'0000_0000

```

(3) Start and end

(a) Starting the DDM

Make necessary settings in the DDM.

- Enabling interrupt output as required
 - Setting a DDM start source
 - Setting a descriptor pointer

After setting the above, set the DDM Enable bit of the DDM control register to 1 to enable DDM start.

Setting example:

```

Enable the interrupt by DDM CH1 end.
  Address: H'E65C_0008 Write data: H'0000_0001
Enable DDM CH1 start when the DMAC1 TE bit is changed.
  Address: H'E65C_0030 Write data: H'0000_1000
Set the DDM CH1 descriptor pointer to H'5800_0210.
  Address: H'E65C_0050 Write data: H'5800_0210
Enable DDM CH1 start.
  Address: H'E65C_0054 Write data: H'0000_0001

```

Set the KICK bit to forced start after setting the DDM Enable bit to 1. Do not set these bits at the same time.

(b) Ending the DDM

When the DDM ends, the DDM_END bit is set to 1. DDM end is cleared by setting the DDE bit to 0. To restart the DDM, you must clear the DDM_END bit by setting the DDE bit to 0.

Procedure for stopping the DDM by the STOP bit

```

Set the STOP bit to 1. Keep the DDE bit at 1.
Check that the DDM_END bit is 1.
Set the DDE bit to 0.

```

63. USB3.0 Host Controller

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

63.1 Overview

USB3.0 DRD (Dual-role Device) controller is capable of both USB Host and USB Peripheral operation, and is compliant with the Universal Serial Bus (USB) 3.0 Specifications.

The USB3.0 controller supports super speed (5 Gbps), high speed (480 Mbps), full speed (12 Mbps) and low speed (1.5 Mbps). The USB3.0 Peripheral Core supports Super Speed (5 Gbps), High Speed (480 Mbps) and Full Speed (12 Mbps).

Table 63.1 Supported Speed Type

| | Speed Type | | | |
|-------------------|-------------|------------|------------|-----------|
| | Super Speed | High Speed | Full Speed | Low Speed |
| Host | √ | √ | √ | √ |
| Peripheral | √ | √ | √ | — |

Also, the USB3.0 controller is compliant with battery charging.

It is based on the eXtensible Host Controller Interface for the Universal Serial Bus (xHCI) Specification as a host controller interface.

The USB3.0 controller requires to download FW to operate as a host (see section 63.4.1, FW Download). Contact Renesas to obtain the FW.

The USB3.0 module supports an input reference clock from both external and on-chip clock sources. The external clock source is supplied through USB3S0_CLK_P/USB3S0_CLK_M pins as a dedicated clock to one channel of USB. The on-chip clock source is supplied through USB_XTAL/USB_EXTAL to share the reference clocks with other USB2.0 channels. See “Design Guidelines for Serial Interface Printed Circuit Boards with Second Generation RZ/G Series Products” for more details.

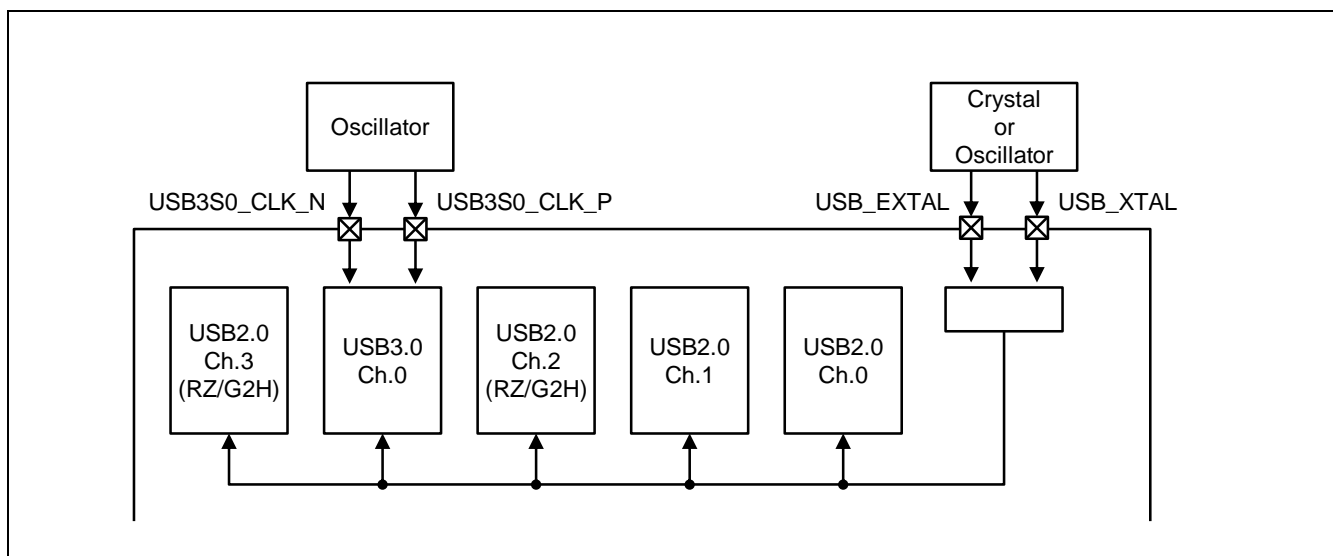


Figure 63.1 Clock Distribution [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

63.1.1 Features

The functions of the USB3.0 DRD controller are listed below.

(1) USB3.0 Host Controller Function

Compliant with Universal Serial Bus 3.0 Specification Revision 1.0

- Supports one downstream port
- Supports control, bulk, interrupt, and isochronous transfers.
- Supports all USB bus speeds, USB 3.0 super speed (5 Gbps), high speed (480 Mbps), full speed (12 Mbps) and low speed (1.5 Mbps).
- Supports power control and overcurrent detection
- Supports BIOS USB.
 - Pre-OS and OS hand-off
 - xHCI legacy support control
- Implements the USB2.0ECN USB 2.0 Link Power Management Addendum and LPM function that is compliant with the xHCI specification.
- Supports polarity inversion of USB3.0 SS differential data pin pair.

Has the following capabilities that are compliant with the eXtensible host controller interface for Universal Serial Bus Specification Revision 1.0.

- 64-bit addressing capability
- BW negotiation capability
- Port power control
- Latency tolerance message capability
- xHCI host capability
 - Number of device slots (MaxSlots): 32
 - Number of total endpoints: 128
 - Number of stream IDs (MaxPSASize): 64 per endpoint (1 EP for reserved bit)
 - Number of event ring segment tables: 2 (max. 512 events/ring) (ERST Max = 1)
 - Supported page size: 8 Kbytes
 - Context size: 64 bytes
- Debug Port
 - Supports debug port compliant with xHCI debug capability

(2) USB3.0 Peripheral Controller Function

- Compliant with Universal Serial Bus 3.0 Specification Revision 1.0
- Supports USB2.0 Speed types, High and Full Speed.
 - **Restriction**
Peripheral function doesn't support Low Speed.
- The maximum PIPE is 5 except PIPE0 and EP0.
- Able to be set to the following Configurations for each PIPE with USB3.0 Super Speed.
 - Endpoint Number: 0 to 15
 - Transfer Type: Control (PIPE0 only)/Bulk/Interrupt
 - Max Burst Size: 1 (value fixed. PIPE0 (Control)), 1 to 16(Bulk), 1 to 3(Interrupt)
 - Max Sequence Number: 31 (fixed)
- **Restriction**
Peripheral function doesn't support Isochronous transfer.
- Able to be set to the following Configurations for each PIPE with USB2.0 Speeds.
 - Endpoint Number: 0 to 15
 - Transfer Type: Control (PIPE0 only)/Bulk/Interrupt
- Supports Stream ID Control (Compliant with UASP specification)
 - It can be assigned both StreamIDs, one is in transfer process and the other will be used after that process.
- Suspend and Resume
 - Supports Power Management by Link Command transaction
 - Supports sending of Remote Wakeup Command.
- Supports sending of Arbitrary LMP, TP, and DP.
- Supports Link Power Management(LPM)

[Battery Charging]

Supports the following modes that are compliant with Battery Charging Specification Revision 1.2

Charging port device

- Standard downstream port mode (SDP)
- Charging downstream port mode (CDP)
- DCP is not supported

63.1.2 External Pins

Table 63.2 External Pins of USB

| | | | | Second Generation RZ/G Series Products | | | |
|--|--|-----|--|---|----------------------------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| | Pin Name | I/O | Function | | | | |
| USB SS receive | USB3S0_RX_P | I | Super Speed differential receive pair | √ | √ | √ | √ |
| | USB3S0_RX_M | | | | | | |
| USB SS transmit | USB3S0_TX_P | O | Super Speed differential transmit pair | √ | √ | √ | √ |
| | USB3S0_TX_M | | | | | | |
| Reference clock | USB3S0_CLK_P | I | Differential input clock pair Clocks must not be asserted before power become stable ON. | √ | √ | √ | √ |
| | USB3S0_CLK_M | | | | | | |
| USB external reference resistor | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] USB3S0_RESREF | I/O | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] Connect to external 200Ω±1% resistor. | √ | √ | √ | √ |
| | [RZ/G2E] USB3S0_RREF | | [RZ/G2E] Connect to external 1.8kΩ±1% resistor. | | | | |
| USB D+/ D- signal | USB3HS0_DP USB3HS0_DM | I/O | D+ / D- Input/output of the HS/FS/LS transceiver. | √ | √ | √ | √ |
| USB VBUS | USB3HS0_VBUS | I/O | Connect to USB VBUS with external 30kΩ±1% series resistor. | √ | √ | √ | — |
| USB Identification | USB3HS0_ID | I | Connect to the pin on the Micro connectors that is used to differentiate a Micro-A plug from a Micro-B plug. | √ | √ | √ | √ |
| Power IC Interface | USB30 PWEN | O | Power Enable | √ | √ | √ | √ |
| | USB30 OVC | I | Over Current Flag | √ | √ | √ | √ |

63.1.3 Register Configuration

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

63.1.3.1 Register Attributes

Table 63.3 Register Bit-field types

| Register Attribute | Description |
|--------------------|---|
| R/W | Register bits can be read and written. |
| RW1C | Register bits can be read. A clear bit may be set by writing "1"; writing 0 to RW1C bits has no effect. |
| RWT | Register bits can be read. An inversion bit may be set by writing "1"; writing 0 to RWT bits has no effect. |
| RW1S | Register bits can be read and written. A bit may be set to 1 by writing "1"; writing 0 to RW1S bits has no effect. |
| R | Register bits can only be read. |
| W | Register bits can only be written. |
| Reserved | Reserved bits are Read Only field. |

63.1.3.2 Register Overview

Register groups in host cores are consisted of following registers.

- PHY Register
- AXI Register
- xHCI Register

CLK60 (from USB PHY) shall be supplied to host core when user reads or writes valid value at host register.

The Peripheral's register groups are assigned as follows.

- PHY Register
- AXI Register
- EPC & USB Common Register
- USB3.0 Super Speed Interface Register

63.1.3.3 USB3.0 Host Register Configuration

Base address is H'EE00_0000 for ch.0. Following sections describe only offset address from the base address.

| Name | Symbol | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|---|------------|-----|---------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| xHCI Register | | | | | | | | | |
| Host Controller Capability Register | HCSPARAMS0 | R | H'000 | H'0100_0020 | 32 | √ | √ | √ | √ |
| Host Controller Capability Register | HCSPARAMS1 | R | H'004 | H'0200_0120 | 32 | √ | √ | √ | √ |
| Host Controller Capability Register | HCSPARAMS2 | R | H'008 | H'2400_0011 | 32 | √ | √ | √ | √ |
| Host Controller Capability Register | HCSPARAMS3 | R | H'00C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Capability Register | HCCPARAMS | R | H'010 | H'0140_51CF | 32 | √ | √ | √ | √ |
| Host Controller Capability Register | DBOFF | R | H'014 | H'0000_0800 | 32 | √ | √ | √ | √ |
| Host Controller Capability Register | RTSOFF | R | H'018 | H'0000_0600 | 32 | √ | √ | √ | √ |
| Host Controller Operational Register | USBCMD | R/W | H'020 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Operational Register | USBSTS | R/W | H'024 | H'0000_0801 | 32 | √ | √ | √ | √ |
| Host Controller Operational Register | PAGESIZE | R | H'028 | H'0000_0001 | 32 | √ | √ | √ | √ |
| Host Controller Operational Register | DNCTRL | R/W | H'034 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Operational Register | CRCRL | R/W | H'038 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Operational Register | CRCRH | R/W | H'03C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Operational Register | DCBAAPL | R/W | H'050 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Operational Register | DCBAAPH | R/W | H'054 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Operational Register | CONFIG | R/W | H'058 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AXI Register for Host Controller | | | | | | | | | |
| AXI Host Control Register | AXH_CON | R/W | H'100 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AXI Host Control Status Register | AXH_STA | R | H'104 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Pseudo PCI Class Code Register | CLASS_CODE | R | H'204 | H'0C03_30xx | 32 | √ | √ | √ | √ |
| Frame Length Adjustment Register | FLADJ | R/W | H'210 | H'0000_2030 | 32 | √ | √ | √ | √ |

| Name | Symbol | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|--|----------------|-----|---------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Host Interrupt Register | HOST_INT | R/W | H'220 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Interrupt Enable Register | INT_ENA | R/W | H'224 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Latency Tolerance Reporting Register | LTR | R | H'228 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Battery Charging Register | BTR_CHRG | R/W | H'230 | H'0100_0000 | 32 | √ | √ | √ | √ |
| FW Version Register | FW_VER | R | H'240 | H'0000_0000 | 32 | √ | √ | √ | √ |
| FW Download Control & Status Register | DL_CTRL | R/W | H'250 | H'0000_0000 | 32 | √ | √ | √ | √ |
| FW Data0 Register | FW_DATA0 | R/W | H'258 | H'0000_0000 | 32 | √ | √ | √ | √ |
| FW Data1 Register | FW_DATA1 | R/W | H'25C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Core Control 3 Register | CORE_CTRL3 | R/W | H'290 | H'0101_0000 | 32 | √ | √ | √ | √ |
| UTMI Plus Vendor Control Interface Set Register | VENDOR_CON_SET | R/W | H'298 | H'0000_0000 | 32 | √ | √ | √ | √ |
| UTMI Plus Vendor Control Interface Status Register | VENDOR_CON_STA | R | H'29C | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Register | | | | | | | | | |
| Port1 Status and Control Register | PORTSC1 | R/W | H'420 | H'0000_02A0 | 32 | √ | √ | √ | √ |
| Port1 Status and Control Register | PORTPMSC1 | R/W | H'424 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Port1 Status and Control Register | PORTPLI | R | H'428 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Port2 Status and Control Register | PORTSC2 | R/W | H'430 | H'0000_02A0 | 32 | √ | √ | √ | √ |
| Port2 Status and Control Register | PORTPMSC2 | R/W | H'434 | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Capabilities Register | USBLEGSUP | R/W | H'500 | H'0000_0401 | 32 | √ | √ | √ | √ |
| xHCI Extended Capabilities Register | USBLEGCTLST S | R/W | H'504 | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Capabilities Register | HCEXTDCAP1 | R | H'510 | H'0300_0602 | 32 | √ | √ | √ | √ |
| xHCI Extended Capabilities Register | HCEXTDCAP2 | R | H'514 | H'2042_5355 | 32 | √ | √ | √ | √ |
| xHCI Extended Capabilities Register | HCEXTDCAP3 | R | H'518 | H'0000_0101 | 32 | √ | √ | √ | √ |
| xHCI Extended Capabilities Register | HCEXTDCAP4 | R | H'520 | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Capabilities Register | HCEXTDCAP5 | R | H'528 | H'0200_0802 | 32 | √ | √ | √ | √ |
| xHCI Extended Capabilities Register | HCEXTDCAP6 | R | H'52C | H'2042_5355 | 32 | √ | √ | √ | √ |
| xHCI Extended Capabilities Register | HCEXTDCAP7 | R | H'530 | H'0000_0102 | 32 | √ | √ | √ | √ |

| Name | Symbol | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|--|---------------------|-----|---------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| xHCI Extended Capabilities Register | HCEXTDCAP8 | R | H'538 | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Capabilities Register | HCEXTDCAP9 | R | H'53C | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Capabilities Register | HCEXTDCAP10 | R | H'540 | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | HCEXTDPWMN GCAP1 | R | H'548 | H'4803_0203 | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | HCEXTDPWMN GCAP2 | R/W | H'54C | H'0000_0008 | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | HCEXTDPWMN GCAP3 | R | H'550 | H'0000_04C0 | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | HCEXTDPWMN GCAP4 | R/W | H'554 | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | DCID | R | H'560 | H'0000_000A | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | DCDB | R/W | H'564 | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | DCERSTSZ | R/W | H'568 | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | DCERSTBA0 | R/W | H'570 | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | DCERSTBA1 | R/W | H'574 | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | DCERDP0 | R/W | H'578 | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | DCERDP1 | R/W | H'57C | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | DCCTRL | R/W | H'580 | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | DCST | R | H'584 | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | DCPORTSC | R/W | H'588 | H'0000_0080 | 32 | √ | √ | √ | √ |

| Name | Symbol | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|--|------------|-----|---------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| xHCI Extended Power Management Capability Register | DCCP0 | R/W | H'590 | H'0000_0000 | 32 | √ | √ | √ | √ |
| xHCI Extended Power Management Capability Register | DCCP1 | R/W | H'594 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Runtime Register | MFINDEX | R | H'600 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Runtime Register | IMAN | R/W | H'620 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Runtime Register | IMOD | R/W | H'624 | H'0000_0FA0 | 32 | √ | √ | √ | √ |
| Host Controller Runtime Register | ERSTSZ | R/W | H'628 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Runtime Register | ERSTBA_L | R/W | H'630 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Runtime Register | ERSTBA_H | R/W | H'634 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Runtime Register | ERDP_L | R/W | H'638 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Runtime Register | ERDP_H | R/W | H'63C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL0 | R/W | H'800 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL1 | R/W | H'804 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL2 | R/W | H'808 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL3 | R/W | H'80C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL4 | R/W | H'810 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL5 | R/W | H'814 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL6 | R/W | H'818 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL7 | R/W | H'81C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL8 | R/W | H'820 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL9 | R/W | H'824 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL10 | R/W | H'828 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL11 | R/W | H'82C | H'0000_0000 | 32 | √ | √ | √ | √ |

| Name | Symbol | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|-----------------------------------|------------|-----|---------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Host Controller Doorbell Register | DOORBELL12 | R/W | H'830 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL13 | R/W | H'834 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL14 | R/W | H'838 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL15 | R/W | H'83C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL16 | R/W | H'840 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL17 | R/W | H'844 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL18 | R/W | H'848 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL19 | R/W | H'84C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL20 | R/W | H'850 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL21 | R/W | H'854 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL22 | R/W | H'858 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL23 | R/W | H'85C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL24 | R/W | H'860 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL25 | R/W | H'864 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL26 | R/W | H'868 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL27 | R/W | H'86C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL28 | R/W | H'870 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL29 | R/W | H'874 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL30 | R/W | H'878 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL31 | R/W | H'87C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Host Controller Doorbell Register | DOORBELL32 | R/W | H'880 | H'0000_0000 | 32 | √ | √ | √ | √ |

63.1.3.4 USB3.0 Peripheral Register Configuration

Base address is H'EE02 0000 for ch.0. Following sections describe only offset address from the base address.

| Name | Symbol | R/W | Address | Initial Value | Access Size (Available Bit Size) | Second Generation RZ/G Series Products | | | |
|--------------------------------------|----------------------|-----|---------|---------------|----------------------------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| AXI Register | | | | | | | | | |
| AXI Control Register | AXI_CON | R/W | H'000 | H'2003_2200 | 32 | √ | √ | √ | √ |
| AXI Status Register | AXI_STA | R | H'004 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AXI Interrupt Status Register | AXI_INT_STA | R/W | H'008 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AXI Interrupt Enable Register | AXI_INT_ENA | R/W | H'00C | H'0000_0000 | 32 | √ | √ | √ | √ |
| DMA Interrupt Status Register | DMA_INT_STA | R/W | H'010 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DMA Interrupt Enable Register | DMA_INT_ENA | R/W | H'014 | H'7FFF_FFFE | 32 | √ | √ | √ | √ |
| Data Enable Status Register | DATAEN_STA | R | H'018 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AXI Common Input Register | AXI_COM_IN | R | H'020 | H'0000_0000 | 32 | √ | √ | √ | √ |
| AXI Common Output Register | AXI_COM_OUT | R/W | H'024 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DMA Ch0 Control Register1 | DMA_Ch0_CON 1 | R/W | H'030 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DMA Ch0 PRD Address Register1 | DMA_Ch0_PRD _ADR1 | R/W | H'034 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DMA Ch0 Control Register2 | DMA_Ch0_CON 2 | R/W | H'040 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DMA Ch0 PRD Address Register2 | DMA_Ch0_PRD _ADR2 | R/W | H'044 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DMA Ch0 Control Register3 | DMA_Ch0_CON 3 | R/W | H'050 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DMA Ch0 PRD Address Register3 | DMA_Ch0_PRD _ADR3 | R/W | H'054 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DMA Ch0 Control Register4 | DMA_Ch0_CON 4 | R/W | H'060 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DMA Ch0 PRD Address Register4 | DMA_Ch0_PRD _ADR4 | R/W | H'064 | H'0000_0000 | 32 | √ | √ | √ | √ |
| EPC & USB Common Register | | | | | | | | | |
| USB Common Control Register | USB_COM_ CON | R/W | H'200 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB20 Control Register | USB20_CON | R/W | H'204 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB30 Control Register | USB30_CON | R/W | H'208 | H'1000_0000 | 32 | √ | √ | √ | √ |
| USB Status Register | USB_STA | R/W | H'210 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB20 Frame No. Register | USB20_FRAME | R | H'214 | H'0000_0000 | 32 | √ | √ | √ | √ |

| Name | Symbol | R/W | Address | Initial Value | Access Size (Available Bit Size) | Second Generation RZ/G Series Products | | | |
|---|-----------------|-----|---------|---------------|----------------------------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| DRD Common Control Register | DRD_CON | R/W | H'218 | H'0000_008x | 32 | √ | √ | √ | √ |
| RD Common Control 2 Register | DRD_CON2 | R/W | H'21C | H'0020_2001 | 32 | √ | √ | √ | √ |
| USB Interrupt Status 1 Register | USB_INT_STA_1 | R/W | H'220 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB Interrupt Status 2 Register | USB_INT_STA_2 | R | H'224 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB Interrupt Enable 1 Register | USB_INT_ENA_1 | R/W | H'228 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB Interrupt Enable 2 Register | USB_INT_ENA_2 | R | H'22C | H'0000_0000 | 32 | √ | √ | √ | √ |
| Setup Data 0 Register | STUP_DAT_0 | R | H'230 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Setup Data 1 Register | STUP_DAT_1 | R | H'234 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB30 TP Send Data 0 Register | USB3_TPDAT_0 | R/W | H'240 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB30 TP Send Data 1 Register | USB3_TPDAT_1 | R/W | H'244 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB30 TP Send Data 2 Register | USB3_TPDAT_2 | R/W | H'248 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB20 LPM Status Register | USB20_LPM_STA | R | H'250 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB30 Vendor Device Test Register | USB30_VND_DEV | R | H'254 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB30 Vendor Defined Data 0 Register | USB30_VND_DAT_0 | R | H'258 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB30 Vendor Defined Data 1 Register | USB30_VND_DAT_1 | R | H'25C | H'0000_0000 | 32 | √ | √ | √ | √ |
| EPC Version Register | USB_VER | R | H'260 | H'041E_0202 | 32 | √ | √ | √ | √ |
| Scratch Pad Register | USB_SCRATCH | R/W | H'264 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB OTG Status Register | USB_OTG_STA | R | H'268 | H'00x0_xxxx | 32 | √ | √ | √ | √ |
| USB OTG Interrupt Status Register | USB_OTG_INT_STA | R/W | H'26C | H'0000_xxxx | 32 | √ | √ | √ | √ |
| USB OTG Interrupt Enable Register | USB_OTG_INT_ENA | R/W | H'270 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPE0 Mode Setting Register | P0_MOD | R | H'280 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPE0 Control Register | P0_CON | R/W | H'288 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPE0 Status Register | P0_STA | R | H'28C | H'0000_0000 | 32 | √ | √ | √ | √ |
| EPO Interrupt Status Register | P0_INT_STA | R/W | H'290 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPE0 Interrupt Enable Register | P0_INT_ENA | R/W | H'294 | H'0000_0000 | 32 | √ | √ | √ | √ |
| UTMI Plus Vendor Control Interface Set Register | VENDOR_CON_SET | R/W | H'298 | H'0000_0000 | 32 | √ | √ | √ | √ |

| Name | Symbol | R/W | Address | Initial Value | Access Size (Available Bit Size) | Second Generation RZ/G Series Products | | | |
|--|----------------|-----|---------|---------------|----------------------------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| UTMI Plus Vendor Control Interface Status Register | VENDOR_CON_STA | R | H'29C | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPE0 Length Register | P0_LNG | R | H'2A0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPE0 Read Register | P0_READ | R | H'2A4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPE0 Write Register | P0_WRITE | W | H'2A8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| DRD Power Control Register | DRD_PW_CON | R/W | H'2AC | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPE Common Setting Register | PIPE_COM | R/W | H'2B0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPEn Mode Setting Register | Pn_MOD | R/W | H'2C0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPEn RAM Mapping Register | Pn_RAMMAP | R/W | H'2C4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPEn Control Register | Pn_CON | R/W | H'2C8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPEn Status Register | Pn_STA | R | H'2CC | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPEn Interrupt Status Register | Pn_INT_STA | R/W | H'2D0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPEn Interrupt Enable Register | Pn_INT_ENA | R/W | H'2D4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPEn Reserved Packet Register | Pn_RSVPKT | R | H'2DC | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPEn Length Register | Pn_LNG | R | H'2E0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPEn Read Register | Pn_READ | R | H'2E4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPEn Write Register | Pn_WRITE | W | H'2E8 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPEn Stream ID Register | Pn_STREAM | R/W | H'2F0 | H'0000_0000 | 32 | √ | √ | √ | √ |
| PIPEn NRDY Stream ID Register | Pn_NRDYSTRM | R/W | H'2F4 | H'0000_0000 | 32 | √ | √ | √ | √ |
| USB3.0 Super Speed Interface Register | | | | | | | | | |
| Port Status and Control Register | PORTSC | R/W | H'300 | H' 0000_0080 | 32 | √ | √ | √ | √ |
| Port PM Status and Control Register | PORTPMSC | R/W | H'304 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Port Link Information Register | PORTLI | R | H'308 | H'0000_0000 | 32 | √ | √ | √ | √ |
| SSIF Command Register | SSIFCMD | R/W | H'340 | H'0000_00C0 | 32 | √ | √ | √ | √ |
| SSIF Link Setting 2 Register | SSIFLINKSET2 | R/W | H'378 | H'D337_00B0 | 32 | √ | √ | √ | √ |
| SSIF Link Setting 3 Register | SSIFLINKSET3 | R/W | H'37C | H'0760_0AC8 | 32 | √ | √ | √ | √ |
| SSIF Link Setting 4 Register | SSIFLINKSET4 | R/W | H'380 | H'0000_FF00 | 32 | √ | √ | √ | √ |

63.1.3.5 PHY Register Configuration

Base address is H'E65E E000 for ch.0. Following sections describe only offset address from the base address.

| Name | Symbol | R/W | Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|------------------------------|------------------|-----|---------|---------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| PHY Register | | | | | | | | | |
| USB30 Clock Setting Register | USB30_CLKSET0 | R/W | H'034 | H'05E7 | 16 | √ | √ | √ | — |
| USB30 Clock Setting Register | USB30_CLKSET1 | R/W | H'036 | H'0003 | 16 | √ | √ | √ | — |
| USB30 SSC Setting Register | USB30_SSC_SET | R/W | H'038 | H'0000 | 16 | √ | √ | √ | — |
| USB30 PHY Enable Register | USB30_PHY_ENABLE | R/W | H'060 | H'0000 | 16 | √ | √ | √ | — |
| USB30 VBUS Enable Register | USB30_VBUS_EN | R/W | H'064 | H'0000 | 16 | √ | √ | √ | — |

63.1.4 Connected Module

Table 63.4 The connected modules to USB3.0 Controller

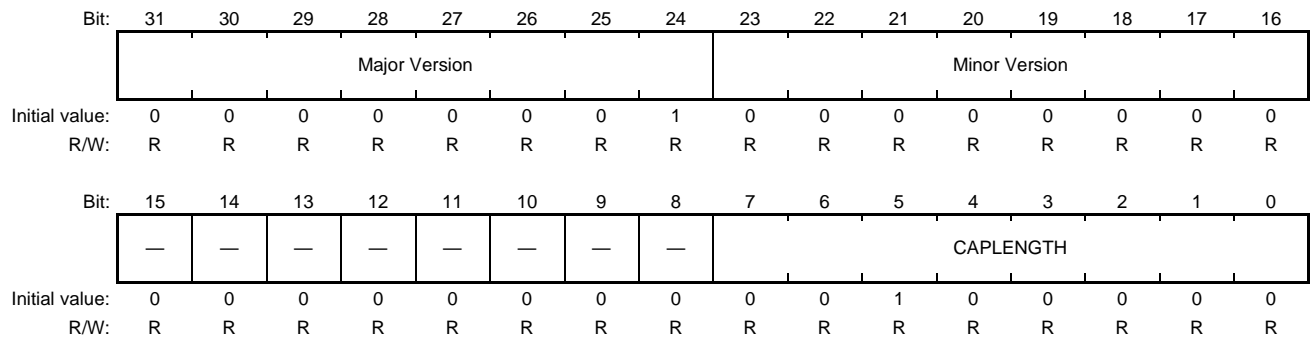
| Module name | Connected module name | Function of connected module |
|--------------------|------------------------------|-------------------------------------|
| USB3.0 Controller | AP-System Core | Access the Register |
| | CPG | Output Clocks |
| | PFC | Select External pins |
| | Module Standby | Control to stop clocks |
| | Software Reset | Execute software reset |
| | INTC | Control to interrupt |

63.2 Register Description

63.2.1 USB3.0 Host Register

63.2.1.1 Host Controller Capability Register (HCSPARAMS0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

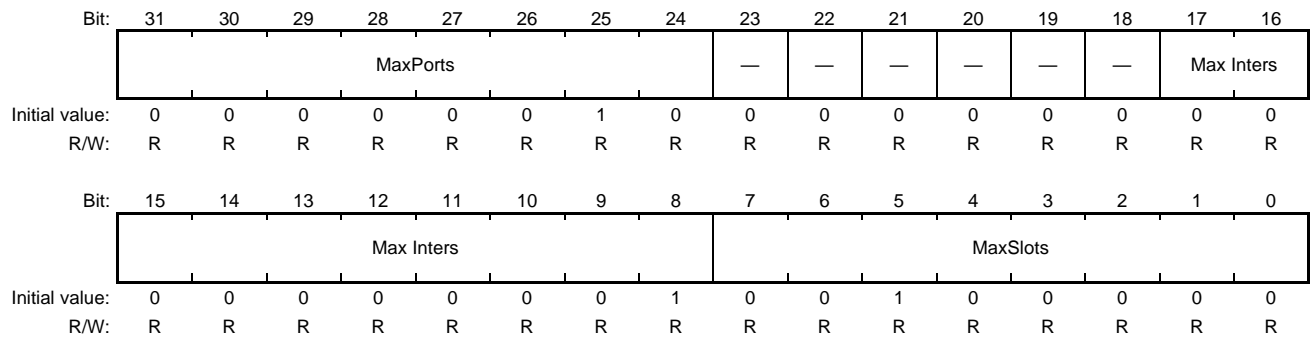


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 24 | Major Version | H'01 | R | Displays Major version of xHCI Specification |
| 23 to 16 | Minor Version | H'00 | R | Displays Minor version of xHCI Specification |
| 15 to 8 | — | All 0 | R | Reserved |
| 7 to 0 | CAPLENGTH | H'20 | R | Displays Capability Register Size |

63.2.1.2 Host Controller Capability Register (HCSPARAMS1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

For MaxPorts, the total number of ports for USB3 port and USB2 port is set. The host core has 1 port for each of USB3 and USB2, so the maxports value is 2. For MaxInters, the total number of Interrupters supported by the host controller is set. The host core has 1 interrupter, so bit is set to 1.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 24 | MaxPorts | H'02 | R | Number of Ports Total number of downstream ports implemented by host controller |
| 23 to 18 | — | All 0 | R | Reserved |
| 17 to 8 | Max Inters | H'001 | R | Number of Interrupters Total number of Interrupters implemented by host controller |
| 7 to 0 | MaxSlots | H'20 | R | Number of Device Slots Maximum number of device context structures and doorbell array entries that can be supported |

63.2.1.3 Host Controller Capability Register (HCSPARAMS2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

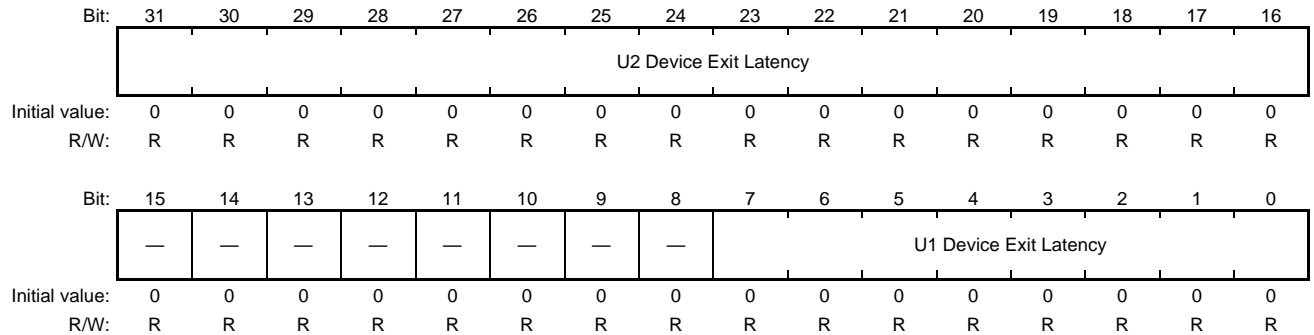
| | | | | | | | | | | | | | | | | |
|----------------|------------------------|----|----|----|----|-----|----|----|----------|----|----|----|-----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Max Scratchpad Buffers | | | | | SPR | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | ERST Max | | | | IST | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------------|---------------|-----|---|
| 31 to 27 | Max Scratchpad Buffers | H'04 | R | Displays the number of scratchpad buffers that the system software reserves for xHC. Valid: 0 to 31 |
| 26 | SPR | B'1 | R | Scratchpad Restore Indicates whether xHC uses a scratchpad buffer or not to save the state when max scratchpad buffers > 0 and save and restore operations are performed. If Max Scratchpad Buffer = 0, this bit is set to 0. |
| 25 to 8 | — | All 0 | R | Reserved |
| 7 to 4 | ERST Max | H'1 | R | Event Ring Segment Table Max Determines the maximum supported value for the event ring segment table base size register. The maximum number of event ring segment table entries = $2^{ERST\ Max}$ |
| 3 to 0 | IST | H'1 | R | Isochronous Scheduling Threshold Specifies the number of frames/microframes. Until bit[3] 0: IST[2:0] microframes, TRB can be added. Until bit[3] 1: IST[2:0] frames, TRB can be added. |

Note: TRB indicates “Transfer Request Block” in this chapter.

63.2.1.4 Host Controller Capability Register (HCSPARAMS3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------------|---------------|-----|--|
| 31 to 16 | U2 Device Exit Latency | H'0000 | R | Worst case latency during transition from U2 to U0. H'0000: zero H'0001: less than 1 μs. H'0002: less than 2 μs ... H'07FF: less than 2,047 μs. H'0800 to H'FFFF: Setting prohibited |
| 15 to 8 | — | All 0 | R | Reserved |
| 7 to 0 | U1 Device Exit Latency | H'00 | R | Worst case latency during transition from U1 to U0. H'00: zero H'01: less than 1 μs. H'02: less than 2 μs ... H'0A: less than 10 μs. H'0B to H'FF: Setting prohibited |

63.2.1.5 Host Controller Capability Register (HCCPARAMS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

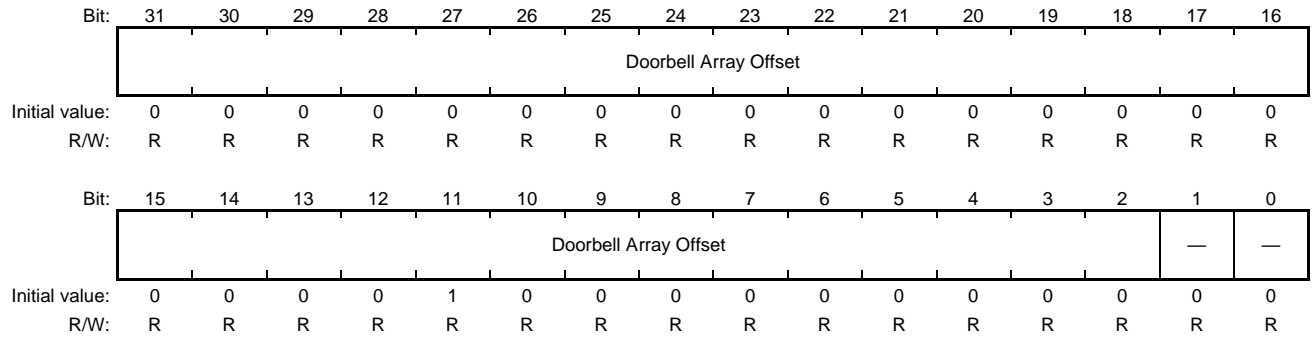
| | | | | | | | | | | | | | | | | |
|----------------|------------|----|----|----|----|----|-----|-----|-----|------|------|-----|-----|-----|------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | xECP | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MaxPSASize | | | — | — | — | PAE | NSS | LTC | LHRC | PIND | PPC | CSZ | BNC | AC64 | |
| Initial value: | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 16 | xECP | H'0140 | R | xHCI Extended Capabilities Pointer Presence or absence of capabilities list H'0000: No extended capabilities Other: Indicates offset from BASE to extended capability. USB Legacy Supported Protocol Capability is at offset H'500. |
| 15 to 12 | MaxPSASize | H'5 | R | Maximum Primary Stream Array Size Maximum size of primary stream array supported by host. Primary Stream Array size = $2^{\text{MaxPSASize}+1}$ Valid: 1 to 15 Default value is 5 as stream number HSP supports 64. |
| 11 to 9 | — | All 0 | R | Reserved |
| 8 | PAE | B'1 | R | Parse All Event Data Indicates whether host controller parses event data TRB that is found while skipping next TD after receiving short packet in aTD. 0: Just parse first 1: Parse all |
| 7 | NSS | B'1 | R | No Secondary SID Support Indicates whether host controller supports secondary stream IDs. 0: Supported 1: Not supported |
| 6 | LTC | B'1 | R | Latency Tolerance Messaging Capability Indicates whether host controller supports latency tolerance messaging. 0: Not supported 1: Supported |
| 5 | LHRC | B'0 | R | Light HC Reset Capability Indicates whether host controller supports light host controller reset. 0: Not supported 1: Supported |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 4 | PIND | B'0 | R | <p>Port Indicators</p> <p>Indicates whether xHCI root hub port supports port indicator control.</p> <p>0: Not contained</p> <p>1: Port status/control register contains a field that can be read and written to control port indicator condition.</p> |
| 3 | PPC | B'1 | R | <p>Port Power Control</p> <p>Indicates whether host controller contains port power control.</p> <p>0: Port doesn't contain port power switch.</p> <p>1: Port contains port power switch</p> |
| 2 | CSZ | B'1 | R | <p>Context Size</p> <p>0: xHC uses 32-byte Context data structure</p> <p>1: xHC uses 64-byte Context data structure</p> |
| 1 | BNC | B'1 | R | <p>BW Negotiation Capability</p> <p>Indicates whether xHC supports bandwidth negotiation</p> <p>0: Does not support BW negotiation</p> <p>1: Supports BW negotiation</p> |
| 0 | AC64 | B'1 | R | <p>64-bit Addressing Capability</p> <p>Addressing range</p> <p>0: 32-bit address memory pointers</p> <p>1: 64-bit address memory pointers</p> |

63.2.1.6 Host Controller Capability Register (DBOFF)

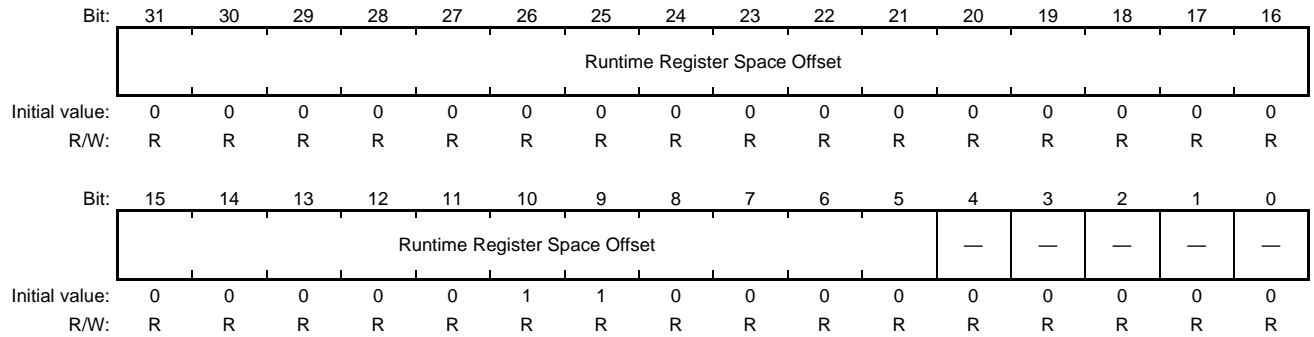
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------------|---------------|-----|---|
| 31 to 2 | Doorbell Array Offset | H'0000_0200 | R | Dword Offset from Base of Doorbell Array base address. Doorbell array's offset is H'800 |
| 1, 0 | — | All 0 | R | Reserved |

63.2.1.7 Host Controller Capability Register (RSTOFF)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------------------------|---------------|-----|---|
| 31 to 5 | Runtime Register Space Offset | H'000_0030 | R | 64-byte offset from Base, for xHCI Runtime Register. Address = Base + runtime register set offset runtime register's offset is H'600 |
| 4 to 0 | — | All 0 | R | Reserved |

63.2.1.8 Host Controller Operational Register (USBCMD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|------|-----|-----|-----|--------|----|----|----|------|------|-------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | EU3S | EWE | CRS | CSS | LHCRST | — | — | — | HSEE | INTE | HCRST | R/S |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 12 | — | All 0 | R | Reserved |
| 11 | EU3S | B'0 | R/W | Enable U3 MFINDEX stop 0: If all root hub ports are in the disconnected, disabled, or powered-off state, xHC can stop count of MFINDEX. 1: If all root hub ports are in the U3, disconnected, disabled, or powered-off state, xHC can stop count of MFINDEX. |
| 10 | EWE | B'0 | R/W | Enable Wrap Event 0: Do not generate MFINDEX wrap event 1: Every time MFINDEX register changes from H'03FFF to H'0, xHC generates MFINDEX wrap event. |
| 9 | CRS | B'0 | R/W | Controller Restore State <ul style="list-style-type: none"> CRS bit = 1, Run/Stop bit = 0 and Save State bit = 1: xHC executes restore state operation to restore the internal state. CRS bit = 0: Restore state operation not executed CRS bit = 1, Run/Stop bit = 1 or Save State bit = 0: Restore state operation not executed read value from AXI is 0. If this bit is set to 1 by AXI, H/W sets USBSTS bit 9 to 1, bit 10 to 0 clear. |
| 8 | CSS | B'0 | R/W | Controller Save State <ul style="list-style-type: none"> CSS bit = 1, Run/Stop bit = 0: xHC saves internal state that is restored at next restore state operation CSS bit = 1, Run/Stop bit = 1: Save state operation is not executed. read value from AXI is 0. If this bit is set to 1 by AXI, H/W sets USBSTS bit 8 to 1, bit 10 to 0 clear. |
| 7 | LHCRST | B'0 | R | Light Host Controller Reset If Light HC Reset Capability bit of HCCPARAMS register is 1, this bit is enabled. Read 0: Reset completed Read 1: Reset not completed |
| 6 to 4 | — | All 0 | R | Reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 3 | HSEE | B'0 | R/W | Host System Error Enable HSEE bit = 1, HSE bit = 1: xHC asserts out-of-band error signal to host. Software clears HSE bit, after which its notice signal is received. |
| 2 | INTE | B'0 | R/W | Interrupt Enable If this bit is 1, host system interrupt is enabled. |
| 1 | HCRST | B'0 | R/W | Host Controller Reset Used to reset host controller. Write 1: Reset Does not influence AXI register. Host controller sets 0, then resetting process completes. Software continues resetting process until this bit changes to 0. Software shall not set 1 when HCHalted bit = 0. |
| 0 | R/S | B'0 | R/W | Run/Stop 0: Stop 1: Run During 1, scheduling is being continued. During 0, completes USB transaction and transits to halted state. xHC shall goes to halted state during the 16 microframes after bit clearing. Software shall not set 1 except when xHC is in halted state. |

63.2.1.9 Host Controller Operational Register (USBSTS)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

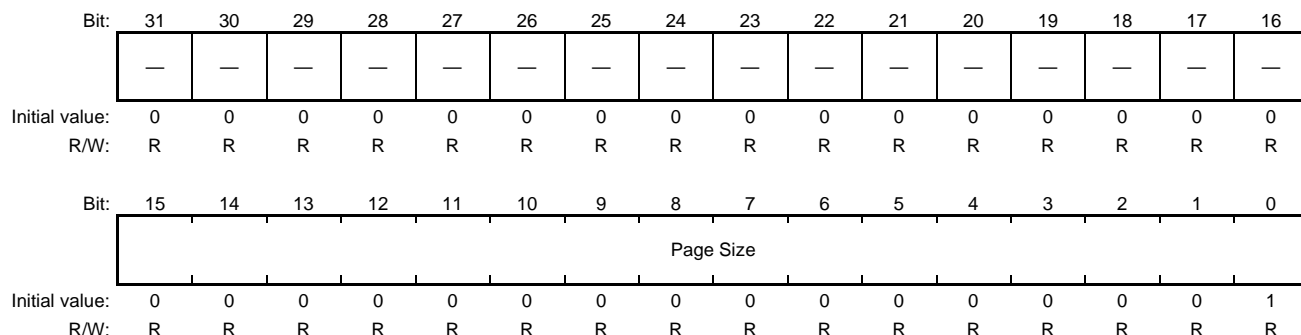
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----|-----|------|-----|-----|----|----|----|------|------|------|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | HCE | CNR | SRE | RSS | SSS | — | — | — | PCD | EINT | HSE | — | HCH |
| Initial value: | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | RW1C | R | R | R | R | R | RW1C | RW1C | RW1C | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|------|--|
| 31 to 13 | — | All 0 | R | Reserved |
| 12 | HCE | B'0 | R | Host Controller Error 0: No internal xHC error condition exists 1: Internal xHC error condition |
| 11 | CNR | B'1 | R | Controller Not Ready 0: Ready 1: Not ready Software shall not set any doorbell and xHC operational registers except USBSTS until CNR = 0. This flag is set by Chip HW reset and cleared when writing is enabled. This flag remains set to 0 until next HW reset. |
| 10 | SRE | B'0 | RW1C | Save/Restore Error When error occurs in save or restore operation, this bit is set to 1. This bit is cleared when save or restore operation starts or when the bit is written with 1. |
| 9 | RSS | B'0 | R | Restore State Status When controller restore state flag in USBCMD is set to 1, this bit is set to 1. During restoring, this bit remains set to 1. When restore state operation finishes, this bit is set to 0. |
| 8 | SSS | B'0 | R | Save State Status This bit is set to 1 when controller save state flag bit in USBCMD is set to 1. Maintains a value of 1 during saving. This bit is set to 0 when save state operation finishes. |
| 7 to 5 | — | All 0 | R | Reserved |
| 4 | PCD | B'0 | RW1C | Port Change Detect This bit is set to 1 when port change bit changes from 0 to 1 and Force Port Resume bit changes from 0 to 1. 1: Detects resuming of suspended port |
| 3 | EINT | B'0 | RW1C | Event interrupt xHC sets 1 when interrupt pending bit is set. 1 write clear |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 2 | HSE | B'0 | RW1C | Host System Error 1: Fatal error occurs when xHC accesses host system. And also DMA transfer error. When these errors occur xHC sets Run/Stop bit to 0 and stops command and event after errors. |
| 1 | — | B'0 | R | Reserved |
| 0 | HCH | B'1 | R | HCHalted If Run/Stop bit is 1, this bit is set to 0. If Run/Stop bit is set to 0 to stop operation, Software or xHC hardware sets this bit to 1. |

63.2.1.10 Host Controller Operational Register (PAGESIZE)

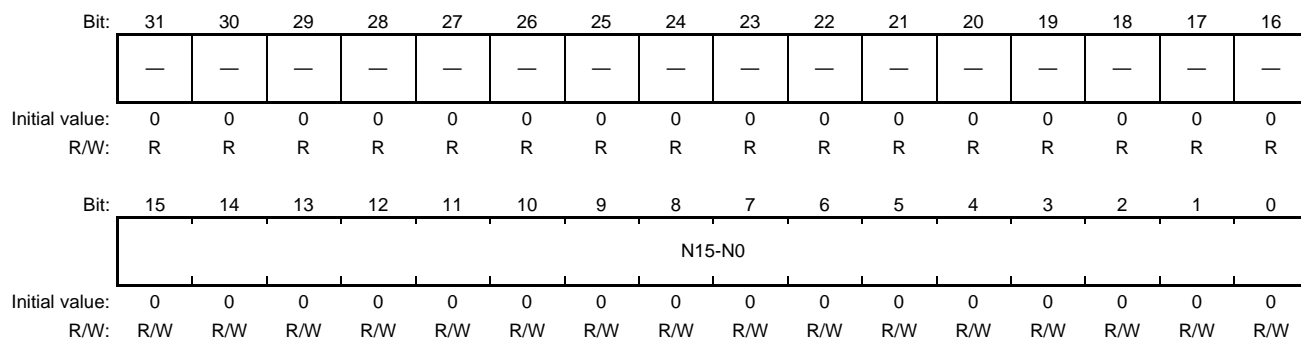
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 to 0 | Page Size | H'0001 | R | xHC's system page size. 8 KB page size |

63.2.1.11 Host Controller Operational Register (DNCTRL)

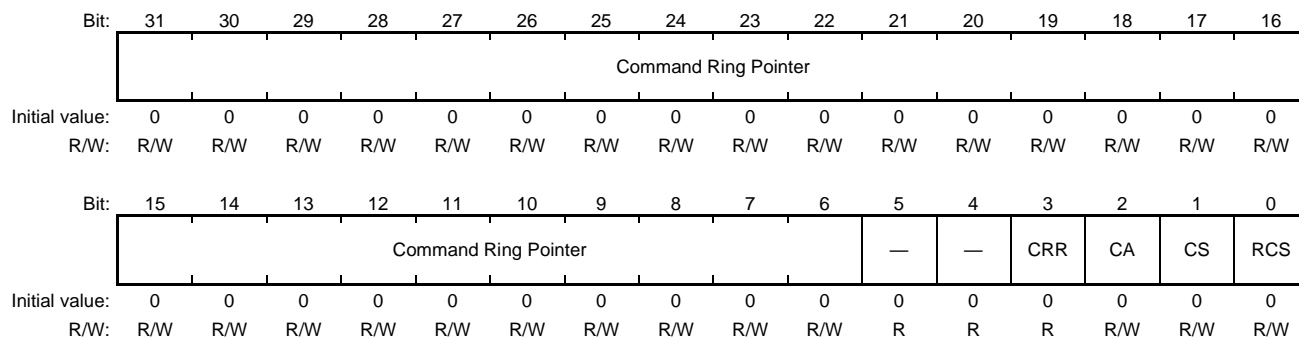
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 to 0 | N15-N0 | H'0000 | R/W | Notification Enable When the Notification Enable bit is set, and the device notification transaction packet has the same value in the notification type field, a device notification event occurs. EX) When N3 is set, and the device notification TP with its notification type field set to 3 (FUNCTION_WAKE) is received, a device notification event occurs. |

63.2.1.12 Host Controller Operational Register (CRCRL)

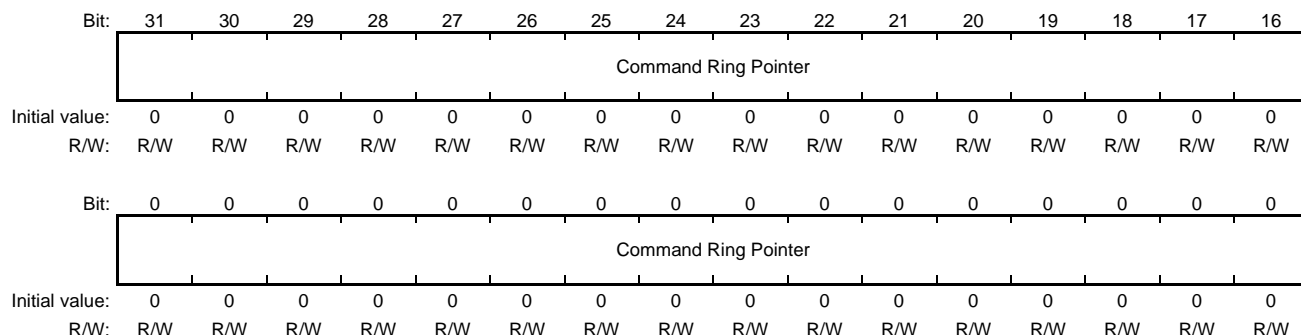
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------------|---------------|-----|---|
| 31 to 6 | Command Ring Pointer | H'000_0000 | R/W | Defines a 64-bit Command Ring Dequeue Pointer default value (lower 32-bit). When Command Ring Running (CRR) is 1, any writing to this bit is ignored. The read value from AXI is 0. |
| 5, 4 | — | All 0 | R | Reserved |
| 3 | CRR | B'0 | R | Command Ring Running This bit is set when the Run/Stop bit is 1 and the DB reason field in the host controller doorbell register is set to 1 by the host controller command. This bit is cleared to 0 when the command ring stops after command stop or command abort is set to 1, or when Run/Stop bit is cleared to 0. |
| 2 | CA | B'0 | R/W | Command Abort When this bit is set to 1, stops all commands and stops command ring, changes completion code to “Command Ring Stop”, and sends command completion with the completion code set to current command ring dequeue pointer value. When Command Ring Running (CRR) is 0, xHC ignores any writing to this field. The read value is 0. |
| 1 | CS | B'0 | R/W | Command Stop Writing 1 causes command ring operation to be stopped after the command’s completion, then the completion code is changed to command ring stop. Command completion with the command TRB pointer set to the current command ring dequeue pointer value is sent. When Command Ring Running (CRR) is 0, xHC ignores any writing to this field. The read value is 0. |
| 0 | RCS | B'0 | R/W | Ring Cycle State Defines the xHC Consumer Cycle State (CCS) referred to by the command ring pointer. When Command Ring Running (CRR) is 1, any writing to this field is ignored. The read value is 0. |

63.2.1.13 Host Controller Operational Register (CRRH)

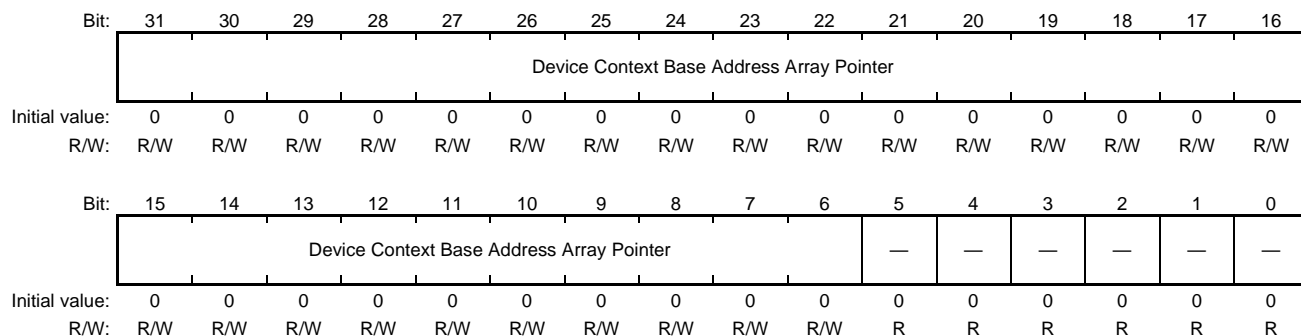
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------------|---------------|-----|---|
| 31 to 0 | Command Ring Pointer | H'0000_0000 | R/W | Defines a 64-bit Command Ring Dequeue Pointer default value (upper 32-bit). When Command Ring Running (CRR) is 1, any writing to this bit is ignored. The read value from AXI is 0. |

63.2.1.14 Host Controller Operational Register (DCBAAPL)

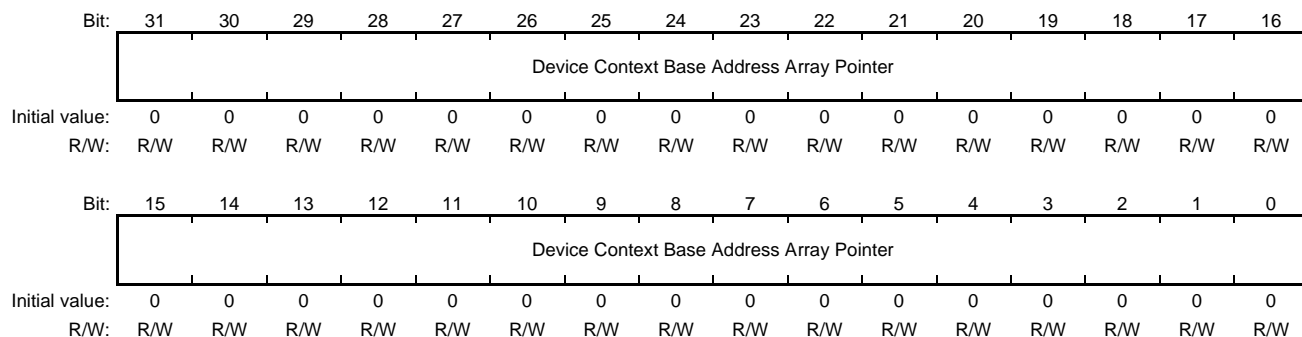
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---|---------------|-----|---|
| 31 to 6 | Device Context Base Address Array Pointer | H'000_0000 | R/W | Defines 64-bit base address of device context pointer array table (lower 32-bit). |
| 5 to 0 | — | All 0 | R | Reserved |

63.2.1.15 Host Controller Operational Register (DCBAAPH)

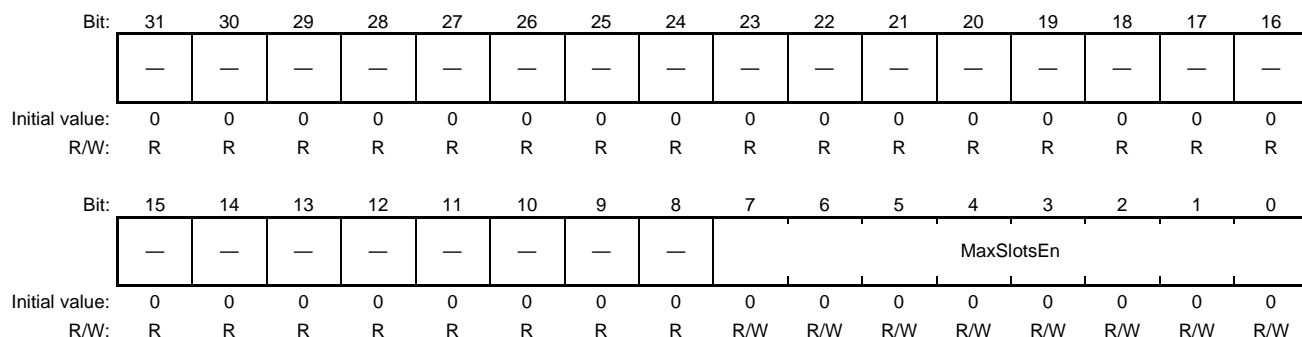
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---|---------------|-----|---|
| 31 to 0 | Device Context Base Address Array Pointer | H'0000_0000 | R/W | Defines 64-bit base address of device context pointer array table (upper 32-bit). |

63.2.1.16 Host Controller Operational Register (CONFIG)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved |
| 7 to 0 | MaxSlotsEn | H'00 | R/W | Max Device Slots Enable Indicates maximum number of enabled device slots. The valid range is 0 to MaxSlots. Enabled device slots are allocated continuously. EX) When 16 is defined, Device Slots 1-16 constitute the active range. 0: Disable Device slots While xHC is running, (Run/Stop bit = 1), it is not possible to write to this field. |

63.2.1.17 AXI Host Control Register (AXH_CON)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-------------|----|----|----|----|----|----|----|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | B3_PHYRST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | SYS_SUSPEND | — | — | — | — | — | — | — | B2_PHYRST |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 17 | — | All 0 | R | Reserved |
| 16 | B3_PHYRST | B'0 | R/W | Bit for reset USB3.0 PHY. 0: Normal condition 1: Reset condition |
| 15 to 9 | — | All 0 | R | Reserved |
| 8 | SYS_SUSPEND | B'0 | R/W | Bit for setting Suspend USB2.0 PHY (PLL stops) by system. 0: Normal condition 1: Suspend condition (USB2PHY PLL stops) |
| 7 to 1 | — | All 0 | R | Reserved |
| 0 | B2_PHYRST | B'0 | R/W | Bit for reset USB2.0 PHY. 0: Normal condition 1: Reset condition |

63.2.1.18 AXI Host Control Status Register (AXH_STA)

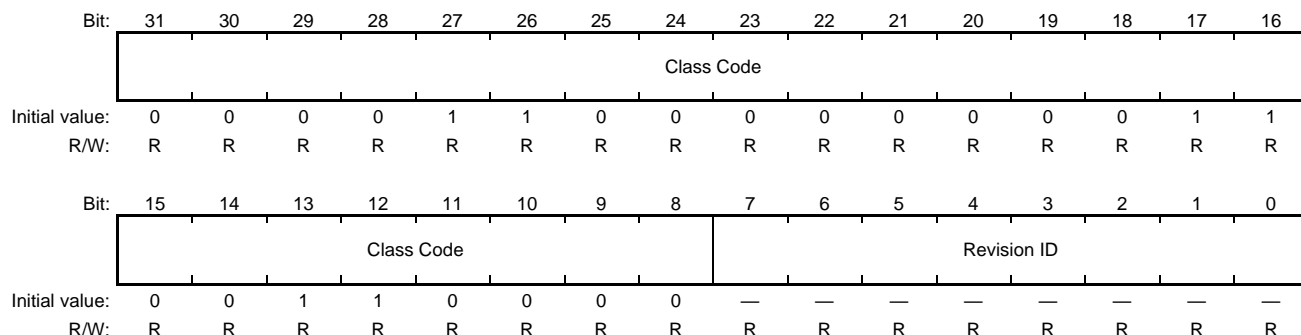
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|------------|----|----|----|----|----|----|----|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | SS_DISABLE | — | — | — | — | — | — | — | B3_PLL_ACTIVE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | B2_PLL_ACTIVE |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 to 25 | — | All 0 | R | Reserved |
| 24 | SS_DISABLE | B'0 | R | Bit for showing SuperSpeed (SS) Port is disabled. The USB3.0 controller core doesn't support SS port when this bit is 1. |
| 23 to 17 | — | All 0 | R | Reserved |
| 16 | B3_PLL_ACTIVE | B'0 | R | Bit for showing that USB3PHY PLL is locked or not. When this bit is 0, it means USB3.0 function is not able to use because valid clock is not output from USB3.0 PHY during USB3PHY PLL is unlocked. 0: PLL Unlock (USB3PHY Clock is invalid) 1: PLL Lock (USB3PHY Clock is valid) |
| 15 to 1 | — | All 0 | R | Reserved |
| 0 | B2_PLL_ACTIVE | B'0 | R | Bit for showing that USB2PHY PLL is locked or not. When this bit is 0, it means USB2.0 function is not able to use because valid clock is not output from USB2.0 PHY during USB2PHY PLL is unlocked. 0: PLL Unlock (USB2PHY Clock is invalid) 1: PLL Lock (USB2PHY Clock is valid) |

63.2.1.19 Pseudo PCI Class Code Register (CLASS_CODE)

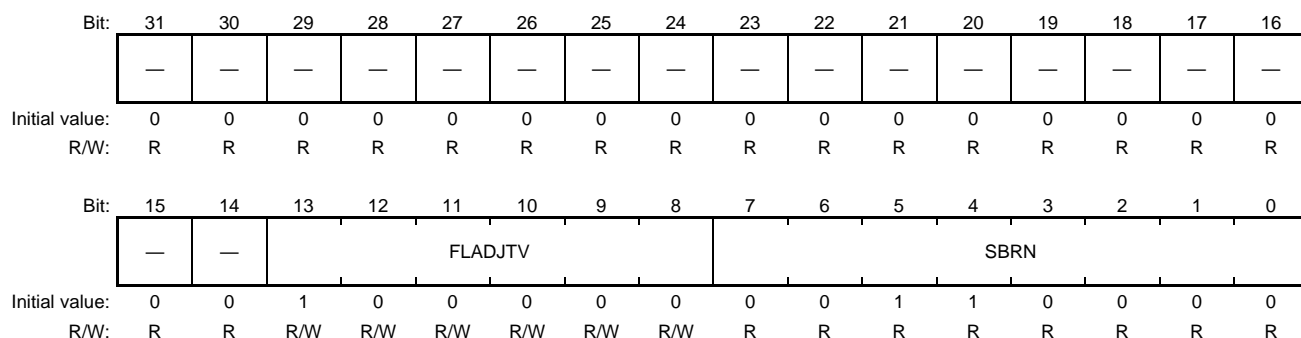
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|---|
| 31 to 8 | Class Code | H'0C_0330 | R | Device Type and Function Base Class: Serial Bus Controller (H'0C) Sub Class: Universal Serial Bus Host Controller (H'03) Program Interface: USB3.0 Controller (H'30) |
| 7 to 0 | Revision ID | — | R | Renesas management number |

63.2.1.20 Frame Length Adjustment Register (FLADJ)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 14 | — | All 0 | R | Reserved |
| 13 to 8 | FLADJTV | H'20 | R/W | Frame Length Adjustment Register (FLADJ) Timing Value Adjust SOF counter Clock. H'20: Frame Length 60,000 SOF cycle time = 59,488 + (Register setting value x16 high-speed bit) |
| 7 to 0 | SBRN | H'30 | R | Serial Bus Release Number Version of universal serial bus specification that is compliant with host controller. H'30: Release 3.0 |

63.2.1.21 Host Interrupt Register (HOST_INT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|---------|---------|---------|---------|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | SMI_STA | LTM_STA | HSE_STA | PME_STA | XHC_STA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | RW1C | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|------|---|
| 31 to 5 | — | All 0 | R | Reserved |
| 4 | SMI_STA | B'0 | R | SMI Interrupt Status bit. When an SMI interrupt occurs, this bit is set to 1. |
| 3 | LTM_STA | B'0 | RW1C | LTM Interrupt status bit. It is asserted when Latency Tolerance Message Device Notification is sent from device and set BELT value at LTR Register. When an LTM interrupt occurs, this bit is set to 1 by FW. AXI sets 1 to clear the bit. |
| 2 | HSE_STA | B'0 | R | HSE Interrupt Status bit. When USBSTS.HSE = 1 & USBCMD.HSEE = 1, this bit is set to 1. AXI sets 1 to clear the bit. |
| 1 | PME_STA | B'0 | R | PME Interrupt Status bit. When PME_STA = 1 & PME_ENA = 1, this bit is set to 1. |
| 0 | XHC_STA | B'0 | R | xHCI Interrupt Status bit. When EventRing is updated and then the INT signal, which indicates asserting of the interrupt in the core, is detected, this bit is set to 1. When the INT signal, which indicates deasserting of the interrupt in the core, is detected, this bit is set to 0 and cleared. |

63.2.1.22 Host Interrupt Enable Register (INT_ENA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|---------|---------|---------|---------|---------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | SMI_ENA | LTM_ENA | HSE_ENA | PME_ENA | XHC_ENA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 5 | — | All 0 | R | Reserved |
| 4 | SMI_ENA | B'0 | R/W | When this bit is set to 1, SMI_STA interrupt is valid. |
| 3 | LTM_ENA | B'0 | R/W | When this bit is set to 1, LTM_STA interrupt is valid. |
| 2 | HSE_ENA | B'0 | R/W | When this bit is set to 1, HSE_STA interrupt is valid. |
| 1 | PME_ENA | B'0 | R/W | When this bit is set to 1, PME_STA interrupt is valid. |
| 0 | XHC_ENA | B'0 | R/W | When this bit is set to 1, XHC_STA interrupt is valid. |

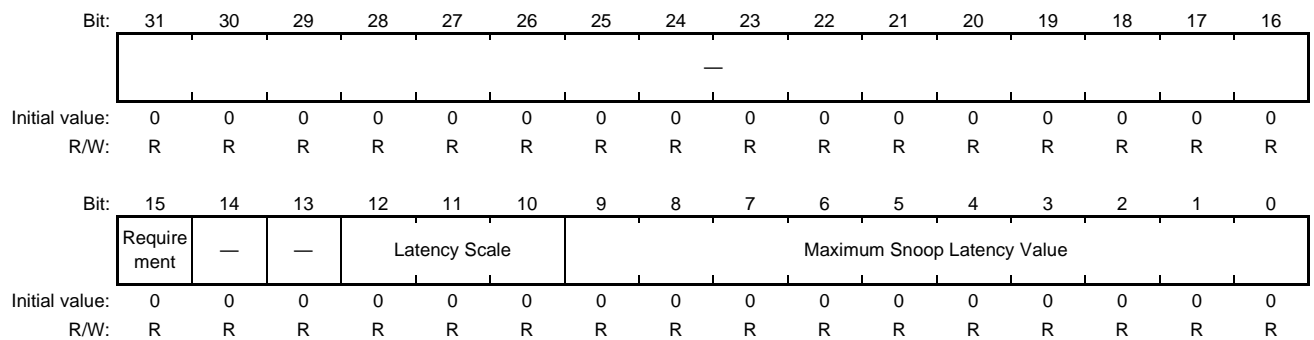
63.2.1.23 Latency Tolerance Reporting Register (LTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Optional normative feature information as USB3.0 Specification.

The register reports the worst value of Best Effort Latency Tolerance (BELT) of Latency Tolerance Message (LTM) Device Notification from device. The “worst” means the maximum BELT value.

Those displayed BELT values are already converted into the format specified by PCI Express Specification.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------------------|---------------|-----|-----------------------|
| 31 to 16 | — | H'0000 | R | Reserved |
| 15 | Requirement | B'0 | R | Requirement (Snoop) |
| 14, 13 | — | All 0 | R | Reserved |
| 12 to 10 | Latency Scale | B'000 | R | Latency Scale (Snoop) |
| 9 to 0 | Maximum Snoop Latency Value | H'000 | R | Latency Value (Snoop) |

63.2.1.24 Battery Charging Register (BTR_CHRG)

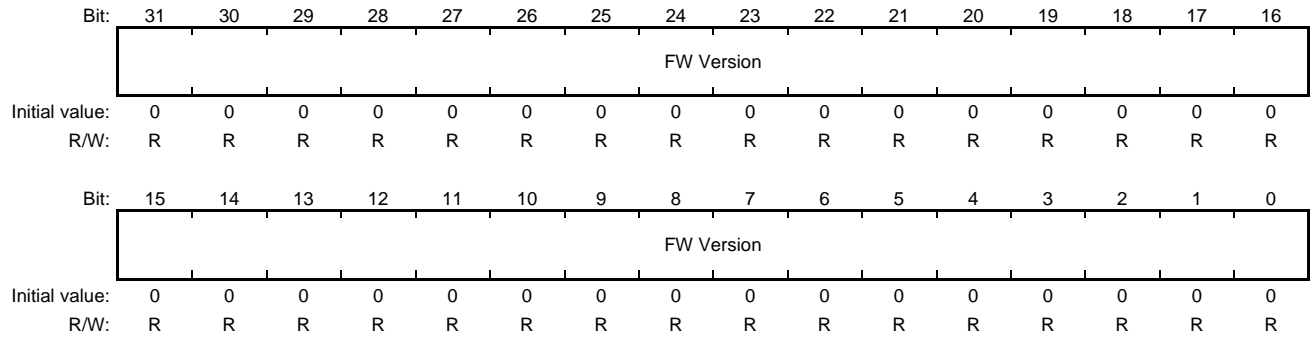
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|-------------------|-----|-----------------|----|----|----|---------|-----|-----|--------------------------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | PTPWE R_CTRL | — | — | — | — | — | — | Renesas Private field | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | LCLK_NONSTOP_FREQ | | — | — | — | — | BC_MODE | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------------------|---------------|-----|---|
| 31 to 25 | — | All 0 | R | Reserved |
| 24 | PTPWER_ CTRL | B'1 | R/W | Set value of PortPowerControl (PPC) at HCCPARAMS in xHCI register. The bit is set to 1 when there is an external Power Switch for control VBUS. |
| 23 to 18 | — | All 0 | R | Reserved |
| 17, 16 | Renesas Private field | B'00 | R/W | Renesas Private bit. This bit can be written only "0". Don't write "1". |
| 15 to 11 | — | All 0 | R | Reserved |
| 10 to 8 | LCLK_NONS TOP_FREQ | B'000 | R/W | Showing a frequency of NONSTOP_CLK entered in USB3.0 controller core. B'000: Default value. Following users shall set this value to 000. <ul style="list-style-type: none"> User who can supply clock (12, 24, 30, 48-MHz) as NONSTOP_CLK to core but don't want to use low-power function. User who can't supply clock (12, 24, 30, 48-MHz) as NONSTOP_CLK to core. B'001: User can supply LCLK_NOSTOP:12 MHz (+/-10%), and uses the function. B'010: User can supply LCLK_NOSTOP:24 MHz (+/-10%), and uses the function. B'011: User can supply LCLK_NOSTOP:30 MHz (+/-10%), and uses the function. B'100: User can supply LCLK_NOSTOP:48 MHz (+/-10%), and uses the function. B'101 - B'111: Reserved |
| 7 to 4 | — | All 0 | R | Reserved |
| 3 to 0 | BC_MODE | B'0000 | R/W | Set mode of Battery Charge function. |

63.2.1.25 FW Version Register (FW_VER)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|--|
| 31 to 0 | FW Version | H'0000_0000 | R | FW Version is showed here. After FW downloading, FW Version can be read. Before FW download: H'0020_0000 After FW download: H'0020_XX00 |

63.2.1.26 FW Download Control & Status Register (DL_CTRL)

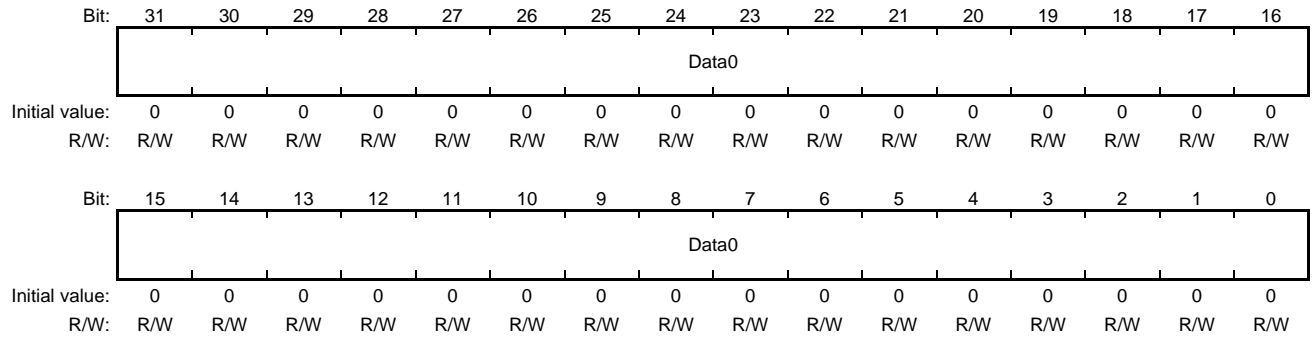
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-----------|-----------|----|-------------|----|----|----|----|--------------------|----------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | Set_Data1 | Set_Data0 | — | Result_Code | | | — | — | BIOS Download Lock | BIOS Download Enable |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | RW1S | RW1S | R | R | R | R | R | R | RW1S | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------|---------------|------|---|
| 31 to 10 | — | All 0 | R | Reserved |
| 9 | Set_Data1 | B'0 | RW1S | When this bit is 1, the value of FW_DATA1 register is set by the system. |
| 8 | Set_Data0 | B'0 | RW1S | When this bit is 1, the value of FW_DATA0 register is set by the system. |
| 7 | — | B'0 | R | Reserved |
| 6 to 4 | Result_Code | B'000 | R | This field presents the results of BIOS Download. B'000: Invalid B'001: Success B'010: Error Others: Setting prohibited |
| 3, 2 | — | All 0 | R | Reserved |
| 1 | BIOS Download Lock | B'0 | RW1S | When this bit is 1, the BIOS download function is locked. |
| 0 | BIOS Download Enable | B'0 | R/W | When this bit is 1, the BIOS download function is enabled. |

63.2.1.27 FW Data0 Register (FW_DATA0)

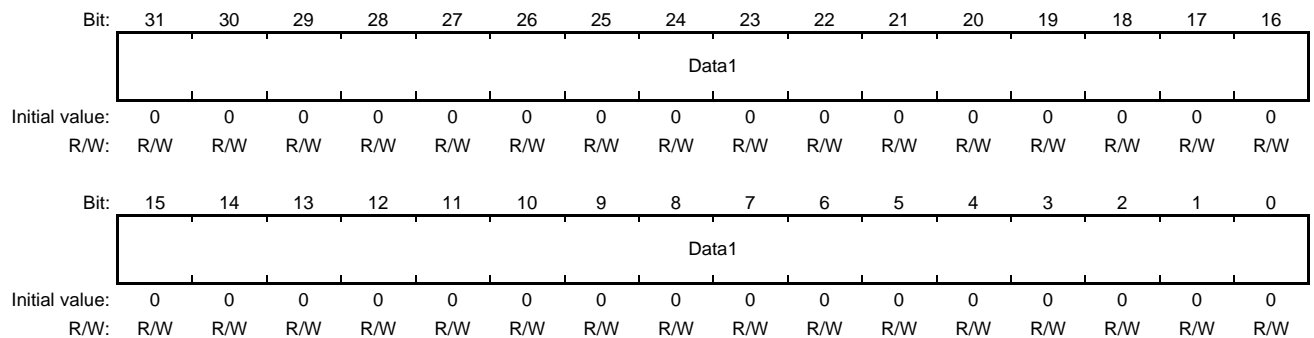
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--------------------------------|
| 31 to 0 | Data0 | H'0000_0000 | R/W | Data register 0 for writing FW |

63.2.1.28 FW Data1 Register (FW_DATA1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--------------------------------|
| 31 to 0 | Data1 | H'0000_0000 | R/W | Data register 1 for writing FW |

63.2.1.29 Core Control 3 Register (CORE_CTRL3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|---------------------|----|----|----|----|----|----|----------|---------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | Renesas Private bit | — | — | — | — | — | — | — | Renesas Private bit |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | PMC_MODE | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|--|
| 31 to 25 | — | All 0 | R | Reserved |
| 24 | Renesas Private bit | B'1 | R/W | Renesas Private bit. Do not write 0 to this bit. |
| 23 to 17 | — | All 0 | R | Reserved |
| 16 | Renesas Private bit | B'1 | R | Renesas Private bit This is write inhibit bit. |
| 15 to 2 | — | All 0 | R | Reserved |
| 1, 0 | PMC_MODE | B'00 | R/W | Sets control of HO_CACTIVE at Low Power Interface. When Bit[0] is 0, it is kept HO_CACTIVE at 1 even if system requests to stop ACLK_H. When Bit[1] is 0, it is kept HO_CACTIVE at 1 and is not to be 0 voluntarily. Bit[0] 1: 0 Control of HO_CACTIVE is valid when HI_CSYSREQ/HI_CSYSACK handshake 0: 0 Control of HO_CACTIVE is invalid when HI_CSYSREQ/HI_CSYSACK handshake Bit[1] 1: 0 Control of HO_CACTIVE is valid when not HI_CSYSREQ/HI_CSYSACK handshake 0: 0 Control of HO_CACTIVE is invalid when not HI_CSYSREQ/HI_CSYSACK handshake |

63.2.1.30 UTMI Plus Vendor Control Interface Set Register (VENDOR_CON_SET)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----------|-----|-----|-----|-----|-----|-----|-------|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | VWDATA | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | VCONTROL | | | | — | — | — | VCWEN | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 24 | — | All 0 | R | Reserved |
| 23 to 16 | VWDATA | H'00 | R/W | Signal for Vendor Write Data. It is write data signal to vendor control register write field. |
| 15 to 8 | — | All 0 | R | Reserved |
| 7 to 4 | VCONTROL | H'0 | R/W | Signal for Vendor Control. It is able to select vendor control register that user wants to use. |
| 3 to 1 | — | All 0 | R | Reserved |
| 0 | VCWEN | B'0 | W | Signal for enabling write VWDATA for Vendor Control Address. If this bit is set to 1, it is enabled to write the value of VWDATA for vendor control address. |

63.2.1.31 UTMI Plus Vendor Control Interface Status Register (VENDOR_CON_STA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------|----|----|----|----|----|----|----|--|--|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | VSTATUS | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved |
| 7 to 0 | VSTATUS | H'00 | R | Signal for Vendor Status. It is read data signal from vendor control register read field. |

63.2.1.32 Port1 Status and Control Register (PORTSC1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------|------------|----|----|-----|-----|-----|-----|------|------|------|------|------|------|------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | WPR | DR | — | — | WOE | WDE | WCE | CAS | CEC | PLC | PRC | OCC | WRC | PEC | CSC | LWS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | RW1S | R | R | R | R/W | R/W | R/W | R | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PIC | Port Speed | | | | PP | PLS | | | | PR | OCA | — | PED | CCS | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | RW1C | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|------|---|
| 31 | WPR | B'0 | RW1S | Warm Port Reset Write 1: Starts warm reset sequence, and sets PR bit to 1. Conformity with the warm reset sequence, is reflected in the PR bit setting of PR, PRC, and WRC. |
| 30 | DR | B'0 | R | Device Removable 0: Device connected to this Port is removable 1: Device connected to this Port is non-removable |
| 29, 28 | — | All 0 | R | Reserved |
| 27 | WOE | B'0 | R/W | Wake on Over-current Enable Write 1: Enables over-current condition as system wake-up event. If Port Power is 0, this bit is set to 0. |
| 26 | WDE | B'0 | R/W | Wake on Disconnect Enable Write 1: Enables device disconnect as system wake-up event. If Port Power is 0, this bit is set to 0. |
| 25 | WCE | B'0 | R/W | Wake on Connect Enable Write 1: Enables device connect as system wake-up event If Port Power is 0, this bit is set to 0. |
| 24 | CAS | B'0 | R | Cold Attach Status 1: Far-end receiver Termination is detected but port status cannot transition to enabled state. If PP is 0 or at USB2.0 Port, this bit is set to 0. |
| 23 | CEC | B'0 | RW1C | Port Config Error Change This bit is set when configuration with link partner fails. |
| 22 | PLC | B'0 | RW1C | Port Link State Change This bit is set when link transitions from U3 to U0 state. 0: No Change 1: Link state Changed Software writes 1 and then the bit is cleared. |
| 21 | PRC | B'0 | RW1C | Port Reset Change This bit is set when port reset process completes. (Port Reset Change from 1 to 0) 0: No change 1: Reset complete Software writes 1 and then the bit is cleared. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|------|---|
| 20 | OCC | B'0 | RW1C | Over-current Change When over-current active changes, this bit is set to 1. Software writes 1 and then the bit is cleared. |
| 19 | WRC | B'0 | RW1C | Warm Port Reset Change This bit is set to 1 when a warm reset is completed. |
| 18 | PEC | B'0 | RW1C | Port Enable/Disable Change 0: No change 1: Port enabled/disabled status has changed This bit is set when USB2 protocol port is disabled due to exiting from appropriate conditions at EO2 point. Software writes 1 and then this bit is cleared. If Port Reset is 0, this bit is set to 0. |
| 17 | CSC | B'0 | RW1C | Connect Status Change 0: No change 1: Change in current connect status xHC sets this bit to 1 at change of port device connect status. Software writes 1 and then this bit is cleared. If Port Reset is 0, this bit is set to 0. |
| 16 | LWS | B'0 | R/W | Port Link State Write Strobe Write 0: Writing to PLS field is ignored. 1: Writing to PLS field is enabled. The read value is 0 |
| 15, 14 | PIC | B'00 | R/W | Port Indicator Control If Port Indicator (PIND) bit in HCCPRAMS register is 0, writing to this field has no effect. If Port Indicator (PIND) bit is 1; B'00: Port indicators are off B'01: Amber B'10: Green B'11: Undefined If Port Reset is 0, this field is set to 0. |
| 13 to 10 | Port Speed | H'0 | R | Speed type of USB device connected to the port. This is enabled only when a device is connected to the port. (Current Connect Status is 1) After a port reset, this field indicates the speed of the device connected to the port. If the reset or speed detection fails, this field indicates Unknown Speed. When the Current Connect Status changes from 0 to 1, this field indicates Undefined Speed. H'0: Undefined Speed (Before Speed detection) H'1: Full-speed device attached H'2: Low-speed device attached H'3: High-speed device attached H'4: SuperSpeed device attached H'5 to H'E: Reserved H'F: Unknown Speed (Error detected during speed detection or Port Reset.) |

| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------|---|---------------|-----|--|------------|---------|------|-------------------------|------|-------------------------|------|-------------------------|------|---|------|-------------------------------|------|-------------------------------|------|-------------------------------|------|------------------------------|------|-------------------------------|------|--------------------------------|------|--------------------------------------|------|-------------------------------|-------------|--------------------|
| 9 | PP | B'1 | R/W | <p>Port Power</p> <p>This field reflects a port's logical power state.</p> <p>0: This port is in the Powered-off state.</p> <p>1: This port is not in the Powered-off state.</p> <p>When the Port Power Control (PPC) bit in the HCSPATAMS1 register is 1, xHC has a port power control switch and this bit shows the current state of the switch. (0: off, 1: on)</p> <p>When the Port Power Control (PPC) bit in the HCSPATAMS1 register is 0, xHC has no port power control switch and each port is hard wired to power and not affected by this bit.</p> <p>If an over-current condition is detected at a powered port, the PP bit of each port is changed from 1 to 0.</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 8 to 5 | PLS | H'5 | R/W | <p>Port Link State</p> <p>This field is used for port power management and shows the Current Link state. When PED is 1, System software sets this bit then sets Link U state.</p> <p>Port Link State is not set to Disabled, RxDetect, or Inactive State.</p> <p>H'0: The link shall transition to the U0 state from any of the U states.</p> <p>H'1: The link should transition to the U1 State.</p> <p>H'2: The link should transition to the U2 State.</p> <p>H'3: The link shall transition to the U3 state from any of the U states.</p> <p>H'4 to H'F: Ignored.</p> <p>To write to this field, the Port Link State Write strobe should be 1. For USB2 Protocol port, writing 2 requests LPM.</p> <p>Software reads this field to determine success/failure of transition to U2 state.</p> <p>Write 0: Deasserts L1 signal</p> <p>Write 1: No influence</p> <table border="1"> <thead> <tr> <th>Read Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>H'0:</td> <td>Link is in the U0 State</td> </tr> <tr> <td>H'1:</td> <td>Link is in the U1 State</td> </tr> <tr> <td>H'2:</td> <td>Link is in the U2 State</td> </tr> <tr> <td>H'3:</td> <td>Link is in the U3 State (Device Suspended)</td> </tr> <tr> <td>H'4:</td> <td>Link is in the Disabled State</td> </tr> <tr> <td>H'5:</td> <td>Link is in the RxDetect State</td> </tr> <tr> <td>H'6:</td> <td>Link is in the Inactive State</td> </tr> <tr> <td>H'7:</td> <td>Link is in the Polling State</td> </tr> <tr> <td>H'8:</td> <td>Link is in the Recovery State</td> </tr> <tr> <td>H'9:</td> <td>Link is in the Hot Reset State</td> </tr> <tr> <td>H'A:</td> <td>Link is in the Compliance Mode State</td> </tr> <tr> <td>H'B:</td> <td>Link is in the Loopback State</td> </tr> <tr> <td>H'C to H'F:</td> <td>Setting prohibited</td> </tr> </tbody> </table> <p>If Port Reset is 0, this field is ambiguous.</p> | Read Value | Meaning | H'0: | Link is in the U0 State | H'1: | Link is in the U1 State | H'2: | Link is in the U2 State | H'3: | Link is in the U3 State (Device Suspended) | H'4: | Link is in the Disabled State | H'5: | Link is in the RxDetect State | H'6: | Link is in the Inactive State | H'7: | Link is in the Polling State | H'8: | Link is in the Recovery State | H'9: | Link is in the Hot Reset State | H'A: | Link is in the Compliance Mode State | H'B: | Link is in the Loopback State | H'C to H'F: | Setting prohibited |
| Read Value | Meaning | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'0: | Link is in the U0 State | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'1: | Link is in the U1 State | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'2: | Link is in the U2 State | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'3: | Link is in the U3 State (Device Suspended) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'4: | Link is in the Disabled State | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'5: | Link is in the RxDetect State | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'6: | Link is in the Inactive State | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'7: | Link is in the Polling State | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'8: | Link is in the Recovery State | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'9: | Link is in the Hot Reset State | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'A: | Link is in the Compliance Mode State | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'B: | Link is in the Loopback State | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H'C to H'F: | Setting prohibited | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 4 | PR | B'0 | R/W | <p>Port Reset</p> <p>0: port is not in reset 1: port reset assert</p> <p>When software sets this bit from 0 to 1, the bus reset sequence is started.</p> <p>Port Enable/Disable bit is 0. Port Enable bit is 0.</p> <p>This bit remains set to 1 until reset by root hub is completed.</p> <p>When error is detected during resetting, Port Speed bits indicate Unknown Speed.</p> <p>If Port Power is 0, this field is set to 0.</p> |
| 3 | OCA | B'0 | R | <p>Over-current Active</p> <p>0: Over-current condition 1: Not over-current condition</p> <p>When an over-current condition is removed, this bit changes from 1 to 0 automatically.</p> |
| 2 | — | B'0 | R | Reserved |
| 1 | PED | B'0 | RW1C | <p>Port Enable/Disable</p> <p>0: Disable 1: Enable</p> <p>After detecting a connection at a port, if SS port initialization or reset by the system software is successful, this bit is automatically enabled by xHC.</p> <p>Port is disabled by fault condition (disconnect event or other fault condition), or USB System Software. Bit status doesn't change until port status changes.</p> <p>If Port Power is 0, this field is set to 0.</p> |
| 0 | CCS | B'0 | R | <p>Current Connect Status</p> <p>0: No device is present 1: Device is present on port</p> <p>Reflects current port status.</p> <p>If Port Power is 0, this field is set to 0.</p> |

63.2.1.33 Port1 Status and Control Register (PORTPMSC1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|-----|-----|-----|-----|-----|-----|-----|------------|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | FLA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | U2 Timeout | | | | | | | | U1 Timeout | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 17 | — | All 0 | R | Reserved |
| 16 | FLA | B'0 | R/W | Force Link PM Accept Write 1: Set Link Function LMP with Force_LinkPM_Accept bit set is generated This bit is set to 0 when PR is 1 or CCS changes from 0 to 1. |
| 15 to 8 | U2 Timeout | H'00 | R/W | U2 inactivity timer Timeout value. If H'FF, port is in the U2 entry disabled state. H'00: Zero (default) H'01: 256 μs H'02: 512 μs ... H'FE: 65.024 ms H'FF: Infinite |
| 7 to 0 | U1 Timeout | H'00 | R/W | U1 inactivity timer Timeout value. If H'FF, port is in the U1 entry disabled state. H'00: Zero (default) H'01: 1 μs H'02: 2 μs ... H'7F: 127 μs H'80 to H'FE: Reserved H'FF: Infinite |

63.2.1.34 Port1 Status and Control Register (PORTPLI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Link Error Count | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|---------------------------------|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 to 0 | Link Error Count | H'0000 | R | Shows Link Error detected Port. |

63.2.1.35 Port2 Status and Control Register (PORTSC2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|------------|----|-----|-----|-----|-----|-----|------|------|------|-----|------|------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | DR | — | — | WOE | WDE | WCE | CAS | — | PLC | PRC | OCC | — | PEC | CSC | LWS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R | R | RW1C | RW1C | RW1C | R | RW1C | RW1C | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PIC | | Port Speed | | | | PP | PLS | | | | PR | OCA | — | PED | CCS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | RW1C | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|------|--|
| 31 | — | B'0 | R | Reserved |
| 30 | DR | B'0 | R | Device Removable 0: Device connected to this Port is removable 1: Device connected to this Port is non-removable |
| 29 to 28 | — | All 0 | R | Reserved |
| 27 | WOE | B'0 | R/W | Wake on Over-current Enable Write 1: Enables over-current condition as System Wake-up Event If Port Power is 0, this bit is set to 0. |
| 26 | WDE | B'0 | R/W | Wake on Disconnect Enable Write 1: Enables device disconnect as System Wake-up Event If Port Power is 0, this bit is set to 0. |
| 25 | WCE | B'0 | R/W | Wake on Connect Enable Write 1: Enables device connect as System Wake-up Event If Port Power is 0, this bit is set to 0. |
| 24 | CAS | B'0 | R | Cold Attach Status 1: Far-end Receiver Termination is detected but Port Status cannot transition to Enabled State. If PP is 0 or at USB2.0 Port, this bit is set to 0. |
| 23 | — | B'0 | R | Reserved |
| 22 | PLC | B'0 | RW1C | Port Link State Change This bit is set when Link transitions from U3 to U0 state. 0: No change 1: Link state changed Software writes 1 and then the bit is cleared. |
| 21 | PRC | B'0 | RW1C | Port Reset Change This bit is set when Port reset process completes. (Port Reset changes from 1 to 0) 0: No change 1: Reset complete Software writes 1 and then the bit is cleared. |

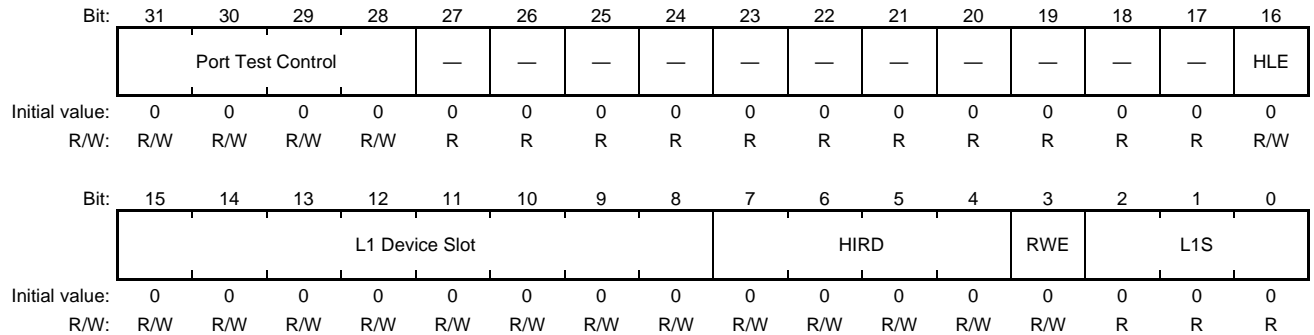
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|------|---|
| 20 | OCC | B'0 | RW1C | Over-current Change When Over-current Active changes, this bit is set to 1. Software writes 1 and then the bit is cleared. |
| 19 | — | B'0 | R | Reserved |
| 18 | PEC | B'0 | RW1C | Port Enable/Disable Change 0: No change 1: Port enabled/disabled status has changed This bit is set when USB2 protocol port is disabled due to exiting from appropriate conditions at EO2 point. Software writes 1 and then this bit is cleared. If Port Power is 0, this bit is set to 0. |
| 17 | CSC | B'0 | RW1C | Connect Status Change 0: No change 1: Change in Current Connect Status xHC sets this bit to 1 at change of port device connect status. Software writes 1 and then this bit is cleared. If Port Power is 0, this bit is set to 0. |
| 16 | LWS | B'0 | R/W | Port Link State Write Strobe Write 0: Writing to PLS field is ignored. 1: Writing to PLS field is enabled. The read value is 0 |
| 15 to 14 | PIC | B'00 | R/W | Port Indicator Control If Port Indicator (PIND) bit in HCCPRAMS Register is 0, writing to this field has no effect. If Port Indicator (PIND) bit is 1; B'00: Port indicators are off B'01: Amber B'10: Green B'11: Undefined If Port Power is 0, this field is set to 0. |
| 13 to 10 | Port Speed | H'0 | R | Speed type of USB device connected to the port. This is enabled only when a device is connected to the port. (Current Connect Status is 1) After a port reset, this field indicates the speed of the device connected to the port. If the reset or speed detection fails, this field indicates Unknown Speed. When the Current Connect Status changes from 0 to 1, this field indicates Undefined Speed. H'0: Undefined Speed (Before Speed detection) H'1: Full-speed device attached H'2: Low-speed device attached H'3: High-speed device attached H'4: SuperSpeed device attached H'5 to H'E: Reserved H'F: Unknown Speed (Error detected during speed detection or Port Reset.) |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 9 | PP | B'1 | R/W | <p>Port Power</p> <p>This field reflects a port's logical power state.</p> <p>0: This port is in the Powered-off state.</p> <p>1: This port is not in the Powered-off state.</p> <p>When the Port Power Control (PPC) bit in the HCSPATAMS1 register is 1, xHC has a port power control switch and this bit shows the current state of the switch. (0: off, 1: on)</p> <p>When the Port Power Control (PPC) bit in the HCSPATAMS1 register is 0, xHC has no port power control switch and each port is hard wired to power and not affected by this bit.</p> <p>If an over-current condition is detected at a powered port, the PP bit of each port is changed from 1 to 0.</p> |
| 8 to 5 | PLS | H'5 | R/W | <p>Port Link State</p> <p>This field is used for port power management and shows the Current Link state.</p> <p>When PED is 1, System software sets this bit then sets Link U state.</p> <p>Port Link State is not set to Disabled, RxDetect, or Inactive State</p> <p>Write Value: Description</p> <p>H'0: The link shall transition to the U0 state from any of the U states.</p> <p>H'1: The link should transition to the U1 State.</p> <p>H'2: The link should transition to the U2 State.</p> <p>H'3: The link shall transition to the U3 state from any of the U states.</p> <p>H'4 to H'F: Ignored.</p> <p>To write to this field, the Port Link State Write strobe should be 1. For USB2 Protocol port, writing 2 requests LPM.</p> <p>Software reads this field to determine success/failure of transition to U2 state.</p> <p>Write 0: Deasserts L1 signal</p> <p>Write 1: No influence</p> <p>Read Value Meaning</p> <p>H'0: Link is in the U0 State</p> <p>H'1: Link is in the U1 State</p> <p>H'2: Link is in the U2 State</p> <p>H'3: Link is in the U3 State</p> <p>(Device Suspended)</p> <p>H'4: Link is in the Disabled State</p> <p>H'5: Link is in the RxDetect State</p> <p>H'6: Link is in the Inactive State</p> <p>H'7: Link is in the Polling State</p> <p>H'8: Link is in the Recovery State</p> <p>H'9: Link is in the Hot Reset State</p> <p>H'A: Link is in the Compliance Mode State</p> <p>H'B: Link is in the Loopback State</p> <p>H'C to H'F: Setting prohibited</p> <p>If Port Power is 0, this field is ambiguous.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|--|
| 4 | PR | B'0 | R/W | <p>Port Reset</p> <p>0: Port is not in reset 1: Port reset assert</p> <p>When software sets this bit from 0 to 1, the bus reset sequence is started.</p> <p>Port Enable/Disable bit is 0. Port Enable bit is 0.</p> <p>This bit remains set to 1 until reset by root hub is completed.</p> <p>When error is detected during resetting, Port Speed field indicates Unknown Speed.</p> <p>If Port Power is 0, this field is set to 0.</p> |
| 3 | OCA | B'0 | R | <p>Over-current Active</p> <p>0: Over-current condition 1: Not over-current condition</p> <p>When an over-current condition is removed, this bit changes from 1 to 0 automatically.</p> |
| 2 | — | B'0 | R | Reserved |
| 1 | PED | B'0 | RW1C | <p>Port Enable/Disable</p> <p>0: Disable 1: Enable</p> <p>After detecting a connection at a port, if SS port initialization or reset by the system software is successful, this bit is automatically enabled by xHC.</p> <p>Port is disabled by fault condition (disconnect event or other fault condition), or USB System Software. Bit status doesn't change until port status changes.</p> <p>If Port Power is 0, this field is set to 0.</p> |
| 0 | CCS | B'0 | R | <p>Current Connect Status</p> <p>0: No device is present 1: Device is present on port</p> <p>Reflects current port status.</p> <p>If Port Power is 0, this field is set to 0.</p> |

63.2.1.36 Port2 Status and Control Register (PORTPMSC2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------|---------------|-----|--|
| 31 to 28 | Port Test Control | H'0 | R/W | 0: port is not operating in test mode Non-zero: port is operating in test mode A non-zero Port Test Control value is valid when the port is in the Disabled, Disconnected, or Suspend State. <USB2 protocol port> Value Test Mode H'0: Test mode not enabled H'1: Test J_STATE H'2: Test K_STATE H'3: Test SE0_NAK H'4: Test Packet H'5: Test FORCE_ENABLE H'6 to H'E: Setting prohibited H'F: Port Test Control Error |
| 27 to 17 | — | All 0 | R | Reserved |
| 16 | HLE | B'0 | R/W | Hardware LPM Enable Writing 1 enables hardware controlled LPM. If Hardware LMP is not supported, this bit is reserved. |
| 15 to 8 | L1 Device Slot | H'00 | R/W | Device Slot ID of Device directly connected to Root Hub Port. 0: No device |
| 7 to 4 | HIRD | H'0 | R/W | Host Initiated Resume Duration Duration of Resume that xHC drives to start resume from L1 state. |
| 3 | RWE | B'0 | R/W | Remote Wake Enable This bit is set to enable or disable device for Remote Wakeup from L1. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 2 to 0 | L1S | B'000 | R | <p>L1 Status</p> <p>Software uses this field to determine whether L1-based suspend request is successful.</p> <p>Value Meaning</p> <p>B'000: Invalid</p> <p>B'001: Success</p> <p> - Port succeed in transition to L1 during resume request (ACK)</p> <p>B'010: Not Yet</p> <p> - Device fails to transition to L1 (NYET)</p> <p>B'011: Not Supported</p> <p> - Device doesn't support L1 transitions (STALL)</p> <p>B'100: Timeout/Error</p> <p> - Device fails LPM Transaction or error occurs</p> <p>B'101 to B'111: Reserved</p> <p>This field is valid when port is in either L0 or L1 state (PLS = 0 or 2).</p> |

63.2.1.37 xHCI Extended Capabilities Register (USBLEGSUP)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------------------|----|----|----|----|----|----|-----------------------------|---------------|----|----|----|----|----|----|-------------------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | HC OS Owned Semaphore | — | — | — | — | — | — | — | HC BIOS Owned Semaphore |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Next Capability Pointer | | | | | | | | Capability ID | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------------|---------------|-----|--|
| 31 to 25 | — | All 0 | R | Reserved |
| 24 | HC OS Owned Semaphore | B'0 | R/W | System Software sets this bit to request xHC ownership. When this bit is 1 and the HC BIOS Owned Semaphore bit is 0, then the OS gets ownership. |
| 23 to 17 | — | All 0 | R | Reserved |
| 16 | HC BIOS Owned Semaphore | B'0 | R/W | BIOS sets this bit to establish xHC ownership. If there is an ownership request from the System Software, the BIOS sets this bit to 0. |
| 15 to 8 | Next Capability Pointer | H'04 | R | Location of next Capability |
| 7 to 0 | Capability ID | H'01 | R | Extended Capability This field is fixed to H'01 due to the Legacy Support Capability. |

63.2.1.38 xHCI Extended Capabilities Register (USBLEGCTLSTS)

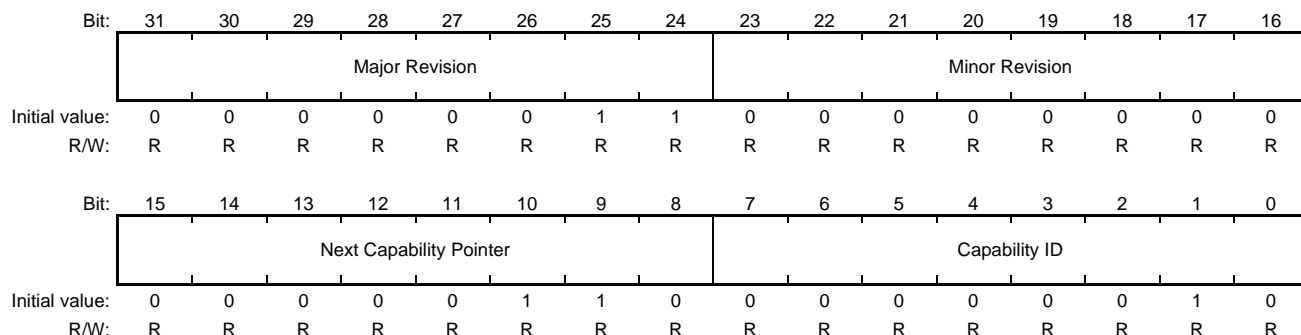
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------------|---------------------------|-----------------------------------|----|----|----|----|----|----|----|----|---------------------------------|----|----|----|------------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | SMI on BAR | SMI on PCI Command | SMI on OS Ownership Change | — | — | — | — | — | — | — | — | SMI on Host System Error | — | — | — | SMI on Event Interrupt |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | RW1C | RW1C | RW1C | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SMI on BAR Enable | SMI on PCI Command Enable | SMI on OS Ownership Change Enable | — | — | — | — | — | — | — | — | SMI on Host System Error Enable | — | — | — | USB SMI Enable |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R | R | R | R | R | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------------------------|---------------|------|---|
| 31 | SMI on BAR | B'0 | RW1C | This bit is set to 1 when the Bass Address Register is written. |
| 30 | SMI on PCI Command | B'0 | RW1C | This bit is set to 1 when the PCI Command Register is written. |
| 29 | SMI on OS Ownership Change | B'0 | RW1C | This Bit is set to 1 when the HC OS Owned Semaphore bit in the USBLEGSUP register changes from 1 to 0, or from 0 to 1. |
| 28 to 21 | — | All 0 | R | Reserved |
| 20 | SMI on Host System Error | B'0 | R | Shadow bit of the Host System Error Interrupt bit in the USBSTS register. If this bit is set to 0, the System Software shall set the Host System Error Interrupt bit in the USBSTS register to 1. |
| 19 to 17 | — | All 0 | R | Reserved |
| 16 | SMI on Event Interrupt | B'0 | R | Shadow bit of the Event Interrupt (EINT) bit in USBSTS register. If EINT is set, this bit is also set. If EINT is cleared, this bit is also cleared. |
| 15 | SMI on BAR Enable | B'0 | R/W | If this bit is 1 and SMI on the BAR bit is 1, the Host controller issues SMI. |
| 14 | SMI on PCI Command Enable | B'0 | R/W | If this bit is 1 and SMI of the PCI Command bit is 1, the Host controller issues SMI. |
| 13 | SMI on OS Ownership Change Enable | B'0 | R/W | If this bit is 1 and SMI of the OS Ownership Change bit is 1, the Host controller issues SMI. |
| 12 to 5 | — | All 0 | R | Reserved |
| 4 | SMI on Host System Error Enable | B'0 | R/W | If this bit is 1 and SMI of the Host System Error bit is 1, the Host controller issues SMI immediately. |
| 3 to 1 | — | All 0 | R | Reserved |
| 0 | USB SMI Enable | B'0 | R/W | If this bit is 1 and SMI of the SMI Event Interrupt bit is 1, the Host controller issues SMI immediately. |

63.2.1.39 xHCI Extended Capabilities Register (HCEXTDCAP1)

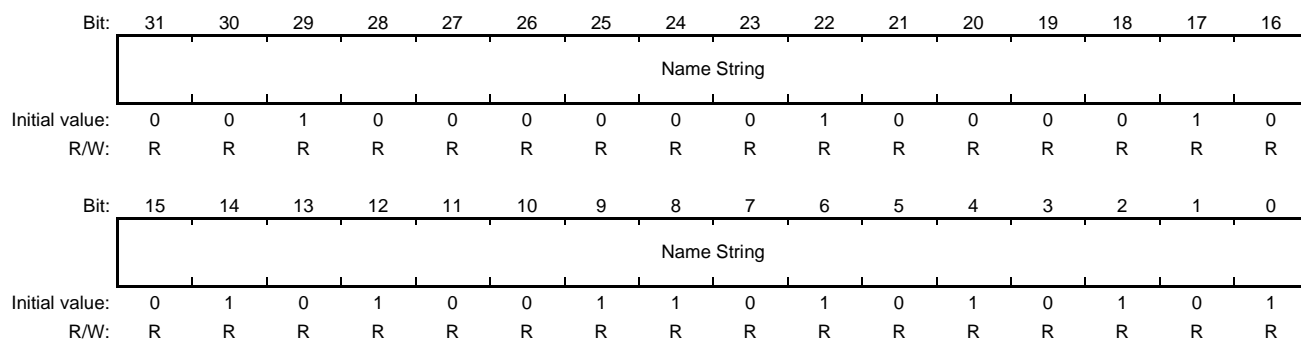
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------------|---------------|-----|---|
| 31 to 24 | Major Revision | H'03 | R | Major release number compliant with xHC specification. |
| 23 to 16 | Minor Revision | H'00 | R | Minor release number compliant with xHC specification. |
| 15 to 8 | Next Capability Pointer | H'06 | R | Next Capability Location |
| 7 to 0 | Capability ID | H'02 | R | Extended Capability Fixed to H'02 due to xHCI Supported Protocol Capability |

63.2.1.40 xHCI Extended Capabilities Register (HCEXTDCAP2)

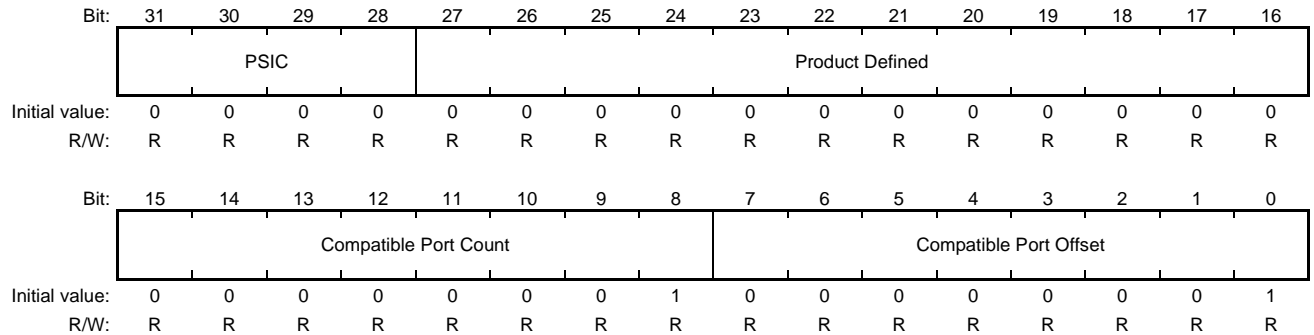
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|---|
| 31 to 0 | Name String | H'2042_5355 | R | Mnemonic name string compliant with xHC specification |

63.2.1.41 xHCI Extended Capabilities Register (HCEXTCAP3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------------|---------------|-----|--|
| 31 to 28 | PSIC | H'0 | R | Product Speed ID Count Number of the Product Speed IDs that this protocol supports. |
| 27 to 16 | Product Defined | H'000 | R | Reserved |
| 15 to 8 | Compatible Port Count | H'01 | R | Number of the Root Hub Ports that this protocol supports. |
| 7 to 0 | Compatible Port Offset | H'01 | R | Starting Port Number of the Root Hubs that this protocol supports. |

Compatible Port Offset is the starting port number of the ports that this protocol supports.

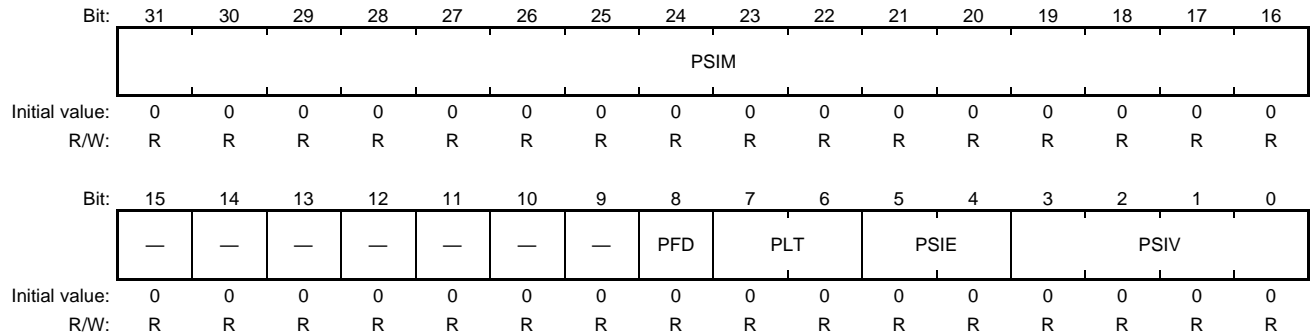
Compatible Port Count is the number of ports that this protocol supports.

In the case of the Host Core, the number of the USB3.0 (Super Speed) Port starts from 1, and the port number is indicated as 1.

PSIC (31:28) is the PSI number that the protocol supports.

63.2.1.42 xHCI Extended Capabilities Register (HCEXTDCAP4)

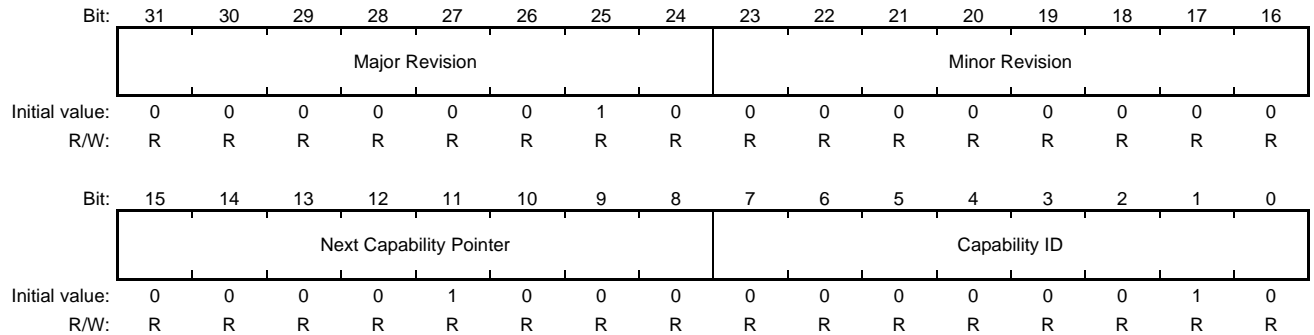
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | PSIM | H'0000 | R | Product Speed ID Mantissa Maximum bit rate defined in this PSI register. Used with the PSIE field. |
| 15 to 9 | — | All 0 | R | Reserved |
| 8 | PFD | B'0 | R | PSI Full-duplex 0: Link is half-duplex. 1: Link is full-duplex. |
| 7, 6 | PLT | B'00 | R | PSI Type Shows whether Symmetric or Asymmetric bit rate is defined for this PSI Register. B'00: Symmetric B'01: Reserved B'10: Asymmetric Rx (Used with Tx PSI Dword) B'11: Asymmetric Tx |
| 5, 4 | PSIE | B'00 | R | Protocol Speed ID Exponent Unit indicated in PSIM . Value Bit Rate B'00: Bits per second B'01: Kb/s B'10: Mb/s B'11: Gb/s |
| 3 to 0 | PSIV | B'0000 | R | Protocol Speed ID Value Used for reporting the Bit rate defined in the PSI register of the Port Speed bits in PORTSC. |

63.2.1.43 xHCI Extended Capabilities Register (HCEXTDCAP5)

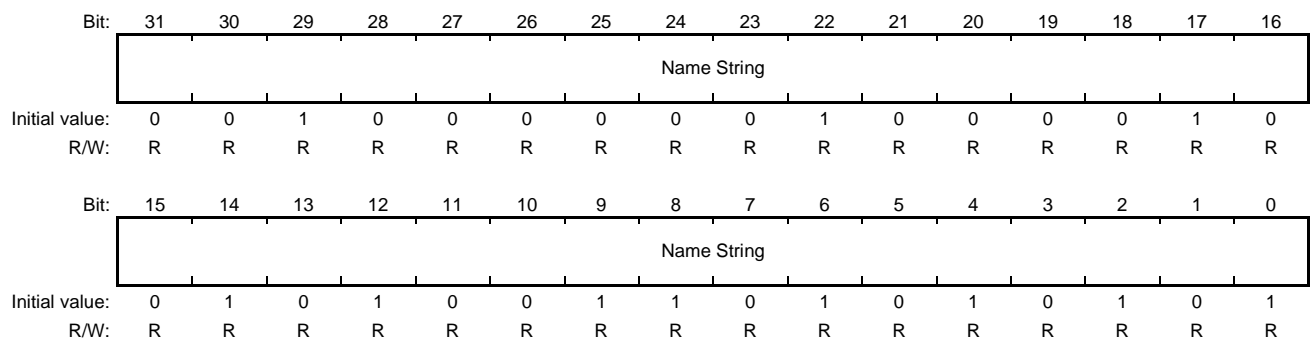
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------------|---------------|-----|--|
| 31 to 24 | Major Revision | H'02 | R | Major release number compliant with xHC specification. |
| 23 to 16 | Minor Revision | H'00 | R | Minor release number compliant with xHC specification. |
| 15 to 8 | Next Capability Pointer | H'08 | R | Next Capability Location |
| 7 to 0 | Capability ID | H'02 | R | Extended Capability Fixed to H'02 due to xHCI Supported Protocol Capability |

63.2.1.44 xHCI Extended Capabilities Register (HCEXTDCAP6)

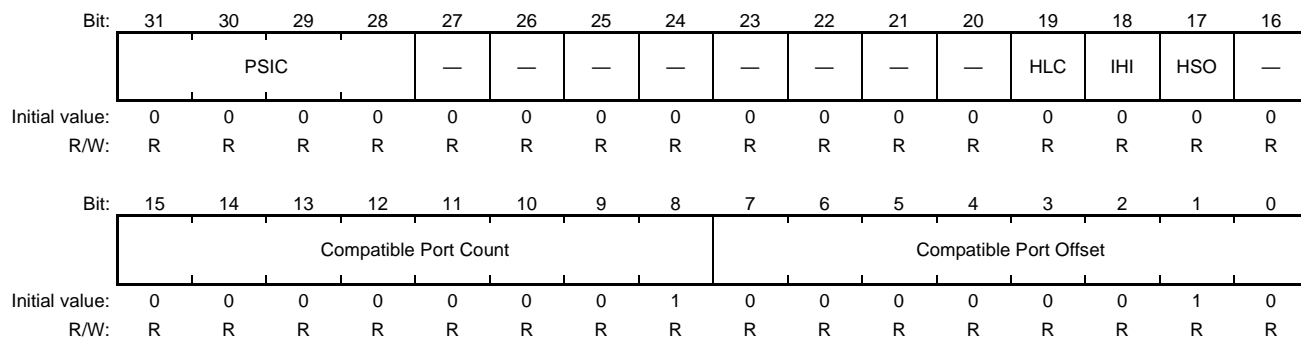
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------|---------------|-----|---|
| 31 to 0 | Name String | H'2042_5355 | R | Mnemonic name string compliant with xHC specification |

63.2.1.45 xHCI Extended Capabilities Register (HCEXTDCAP7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------------|---------------|-----|--|
| 31 to 28 | PSIC | H'0 | R | Product Speed ID Count Number of Product Speed IDs that this protocol supports. |
| 27 to 20 | — | All 0 | R | Reserved |
| 19 | HLC | B'0 | R | Hardware LPM Capability 1: Indicates whether hardware controlled LPM is supported |
| 18 | IHI | B'0 | R | Integrated Hub Implemented 1: Root Hub and Port mapping of External xHC port are not compliant with default mapping specified in 4.24.2.1 in xHCI Spec. |
| 17 | HSO | B'0 | R | High-speed Only 1: USB2 Port indicated with this Capability is HS Only |
| 16 | — | B'0 | R | Reserved |
| 15 to 8 | Compatible Port Count | H'01 | R | Number of Root Hub Ports that this protocol supports. |
| 7 to 0 | Compatible Port Offset | H'02 | R | Starting Port Number of Root Hubs that this protocol supports. |

Compatible Port Offset is the starting port number of the ports that this protocol supports.

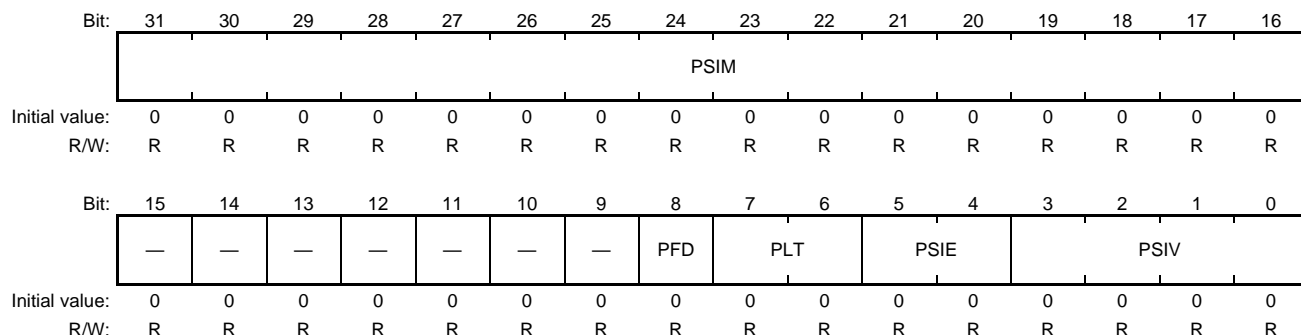
Compatible Port Count is the number of ports that this protocol supports.

In the case of the Host Core, the number of the USB2.0 Port starts from 2, and the port number is indicated as 1.

PSIC (31:28) is the PSI number that the protocol supports.

63.2.1.46 xHCI Extended Capabilities Register (HCEXTCAP8)

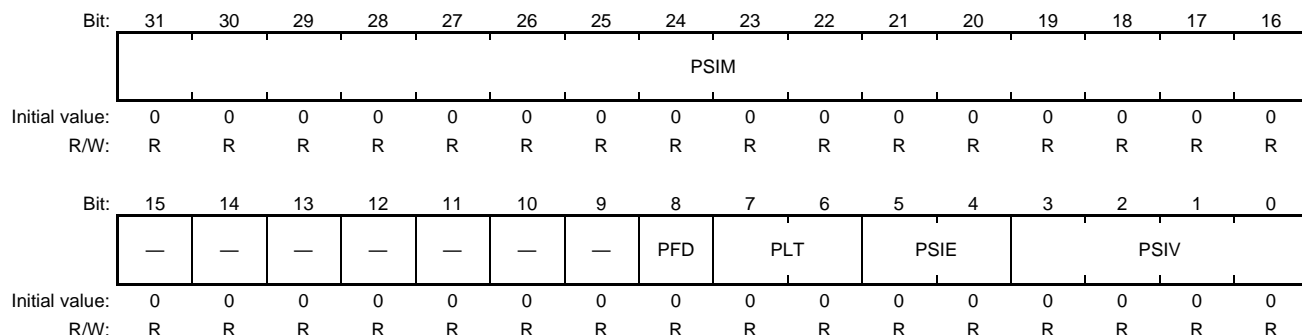
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | PSIM | H'0000 | R | Product Speed ID Mantissa Maximum bit rate defined in this PSI register. Used with the PSIE field. |
| 15 to 9 | — | All 0 | R | Reserved |
| 8 | PFD | B'0 | R | PSI Full-duplex 0: Link is half-duplex. 1: Link is full-duplex. |
| 7, 6 | PLT | B'00 | R | PSI Type Shows whether Symmetric or Asymmetric bit rate is defined for this PSI Register. B'00: Symmetric B'01: Reserved B'10: Asymmetric Rx (Used with Tx PSI Dword) B'11: Asymmetric Tx |
| 5, 4 | PSIE | B'00 | R | Protocol Speed ID Exponent Unit indicated in P . Value Bit Rate B'00: Bits per second B'01: Kb/s B'10: Mb/s B'11: Gb/s |
| 3 to 0 | PSIV | B'0000 | R | Protocol Speed ID Value Used for reporting the Bit rate defined in the PSI Register of the Port Speed field in PORTSC. |

63.2.1.47 xHCI Extended Capabilities Register (HCEXTDCAP9)

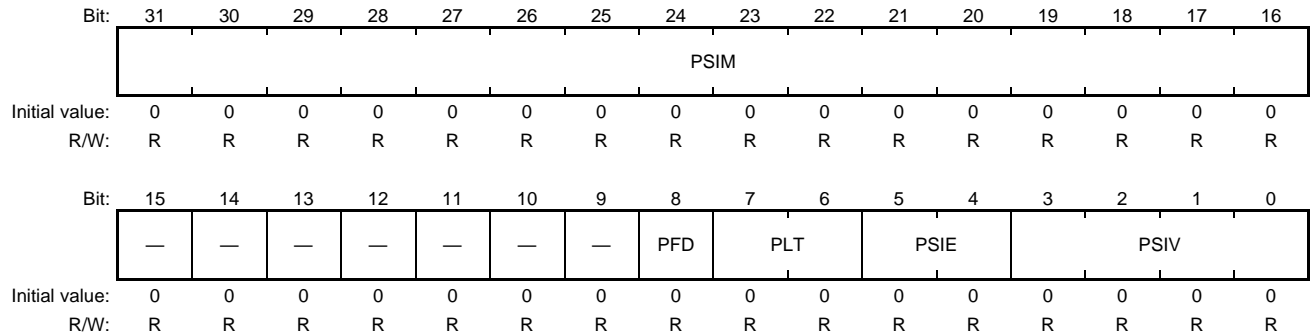
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | PSIM | H'0000 | R | Product Speed ID Mantissa Maximum bit rate defined in this PSI register. Used with the PSIE field. |
| 15 to 9 | — | All 0 | R | Reserved |
| 8 | PFD | B'0 | R | PSI Full-duplex 0: Link is half-duplex. 1: Link is full-duplex. |
| 7, 6 | PLT | B'00 | R | PSI Type Shows whether Symmetric or Asymmetric bit rate is defined for this PSI Registers. B'00: Symmetric B'01: Reserved B'10: Asymmetric Rx (Used with Tx PSI Dword) B'11: Asymmetric Tx |
| 5, 4 | PSIE | B'00 | R | Protocol Speed ID Exponent Unit indicated in PSIM. Value Bit Rate B'00: Bits per second B'01: Kb/s B'10: Mb/s B'11: Gb/s |
| 3 to 0 | PSIV | B'0000 | R | Protocol Speed ID Value Used for reporting the Bit rate defined in the PSI Register of the Port Speed field in PORTSC. |

63.2.1.48 xHCI Extended Capabilities Register (HCEXTCAP10)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | PSIM | H'0000 | R | Product Speed ID Mantissa Maximum bit rate defined in this PSI register. It is used with PSIE field. |
| 15 to 9 | — | All 0 | R | Reserved |
| 8 | PFD | B'0 | R | PSI Full-duplex 0: Link is half-duplex. 1: Link is full-duplex. |
| 7, 6 | PLT | B'00 | R | PSI Type Shows Symmetric or Asymmetric bit rate is defined for this PSI Register. B'00: Symmetric B'01: Reserved B'10: Asymmetric Rx (Used with Tx PSI Dword) B'11: Asymmetric Tx |
| 5, 4 | PSIE | B'00 | R | Protocol Speed ID Exponent Unit indicated in P . Value Bit Rate B'00: Bits per second B'01: Kb/s B'10: Mb/s B'11: Gb/s |
| 3 to 0 | PSIV | B'0000 | R | Protocol Speed ID Value Used for reporting the Bit rate defined in the PSI Register of the Port Speed field in PORTSC. |

63.2.1.49 xHCI Extended Power Management Capability Register (HCEXTPWMNGCAP1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------------------|---------------|------------|------------|------------|---------------|---------------|---------------|----|----|-----|----|--------------|---------|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | PMES D3cold | PMES D3hot | PMES D2 | PMES D1 | PMES D0 | D2 Support | D1 Support | AUX Current | | | DSI | — | PME Clock | Version | | |
| Initial value: | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Next Capability Pointer | | | | | | | Capability ID | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------------|---------------|-----|--|
| 31 | PMESD3cold | B'0 | R | PME Support D3cold Window for accessing contents of AXI Register Power Management Capability Register |
| 30 | PMESD3hot | B'1 | R | PME Support D3hot Window for accessing contents of AXI Register Power Management Capability Register |
| 29 | PMESD2 | B'0 | R | PME Support D2 Window for accessing contents of AXI Register Power Management Capability Register |
| 28 | PMESD1 | B'0 | R | PME Support D1 Window for accessing contents of AXI Register Power Management Capability Register |
| 27 | PMESD0 | B'1 | R | PME Support D0 Window for accessing contents of AXI Register Power Management Capability Register |
| 26 | D2 Support | B'0 | R | Window for accessing contents of AXI Register Power Management Capability Register |
| 25 | D1 Support | B'0 | R | Window for accessing contents of AXI Register Power Management Capability Register |
| 24 to 22 | AUX Current | B'000 | R | Window for accessing contents of AXI Register Power Management Capability Register |
| 21 | DSI | B'0 | R | Window for accessing contents of AXI Register Power Management Capability Register |
| 20 | — | B'0 | R | Reserved |
| 19 | PME Clock | B'0 | R | Window for accessing contents of AXI Register Power Management Capability Register |
| 18 to 16 | Version | B'011 | R | Window for accessing contents of AXI Register Power Management Capability Register |
| 15 to 8 | Next Capability Pointer | H'02 | R | Relative Pointer to next Capability Fixed to H'02 |
| 7 to 0 | Capability ID | H'03 | R | Capability ID H'03: Extended Power Management Capability Fixed to H'03 |

63.2.1.50 xHCI Extended Power Management Capability Register (HCEXTDPWMNGCAP2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|-------------|----|----|----|------------|-----|----|----|----|---------------|----|-------------|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PME Status | Data Scale | Data Select | | | | PME Enable | — | — | — | — | No Soft Reset | — | Power State | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| R/W: | RW1CS | R | R | R | R | R | R | RWS | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-------|--|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 | PME Status | B'0 | RW1CS | Window for accessing contents of AXI Register Power Management Status/Control Register |
| 14, 13 | Data Scale | B'00 | R | Window for accessing contents of AXI Register Power Management Status/Control Register |
| 12 to 9 | Data Select | B'0000 | R | Window for accessing contents of AXI Register Power Management Status/Control Register |
| 8 | PME Enable | B'0 | RWS | Window for accessing contents of AXI Register Power Management Status/Control Register |
| 7 to 4 | — | All 0 | R | Reserved |
| 3 | No Soft Reset | B'1 | R | Window for accessing contents of AXI Register Power Management Status/Control Register |
| 2 | — | B'0 | R | Reserved |
| 1, 0 | Power State | B'00 | R/W | Window for accessing contents of AXI Register Power Management Status/Control Register |

63.2.1.51 xHCI Extended Power Management Capability Register (HCEXTDPMWNGCAP3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------------------|----|----|----|----|----|----|----|---------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Next Capability Pointer | | | | | | | | Capability ID | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 to 8 | Next Capability Pointer | H'04 | R | Relative Pointer to next Capability Fixed to H'04 |
| 7 to 0 | Capability ID | H'C0 | R | Capability ID H'C0: Extended Power Management Capability Fixed to H'C0 |

63.2.1.52 xHCI Extended Power Management Capability Register (HCEXTDPMWNGCAP4)

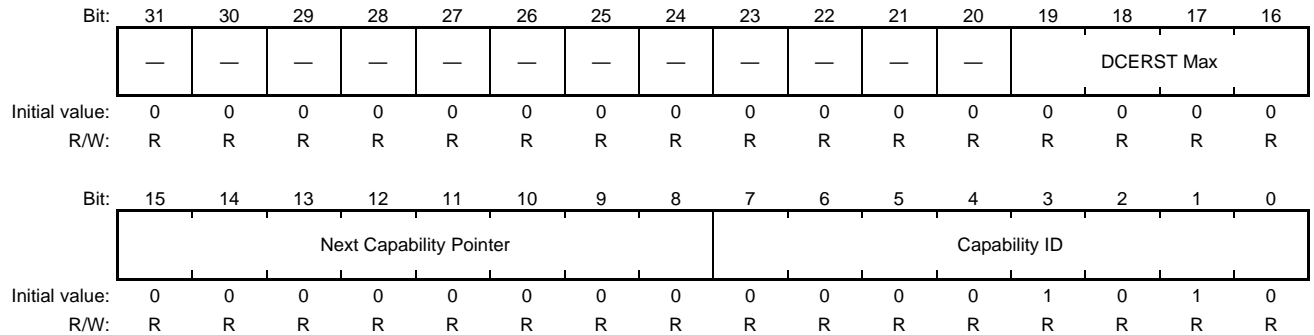
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | Command Enable |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved |
| 0 | Command Enable | B'0 | R/W | 0: Disable Vender Defined Command 1: Enable Vender Defined Command |

63.2.1.53 xHCI Extended Power Management Capability Register (DCID)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------------|---------------|-----|---|
| 31 to 20 | — | All 0 | R | Reserved |
| 19 to 16 | DCERST Max | B'0000 | R | Maximum number of the debug capability event ring segment table base size register that this debug capability has. Maximum number of Event Ring Segment Table entries = $2^{DCERST Max}$. |
| 15 to 8 | Next Capability Pointer | H'00 | R | Relative pointer to next capability. Fixed to H'00 due to there being no next capability. |
| 7 to 0 | Capability ID | H'0A | R | Capability ID H'0A: Debug capability Fixed to H'0A |

63.2.1.54 xHCI Extended Power Management Capability Register (DCDB)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------------|-----|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Doorbell Target | | | | | | | | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 to 8 | Doorbell Target | H'00 | R/W | Doorbell Target Value Definition H'00: Data EP 1 OUT Enqueue Pointer Update H'01: Data EP 1 IN Enqueue Pointer Update H'02 to H'FF: Reserved |
| 7 to 0 | — | All 0 | R | Reserved |

63.2.1.55 xHCI Extended Power Management Capability Register (DCERSTSZ)

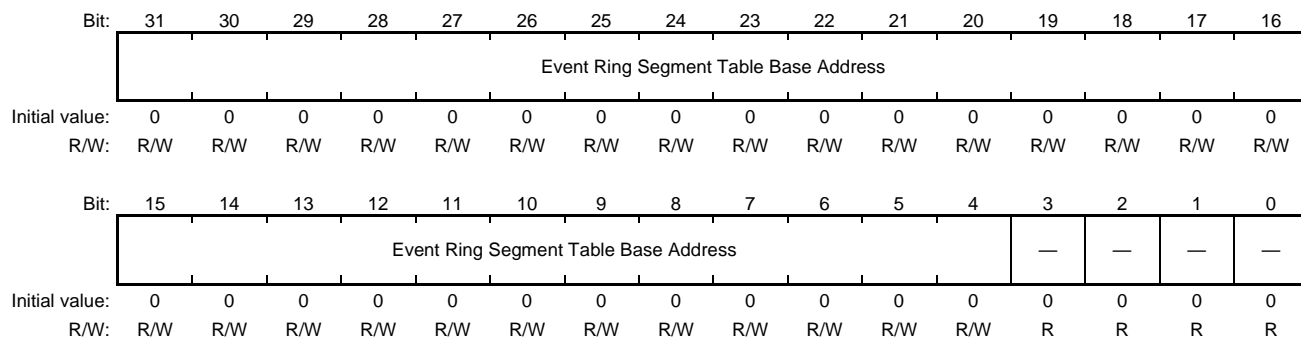
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | Event Ring Segment Table Size | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 to 0 | Event Ring Segment Table Size | H'0000 | R/W | Indicates the number of valid Debug Capability Event Ring Segment Table Entries. The maximum number is defined by the DCERST Max field. |

63.2.1.56 xHCI Extended Power Management Capability Register (DCERSTBA0)

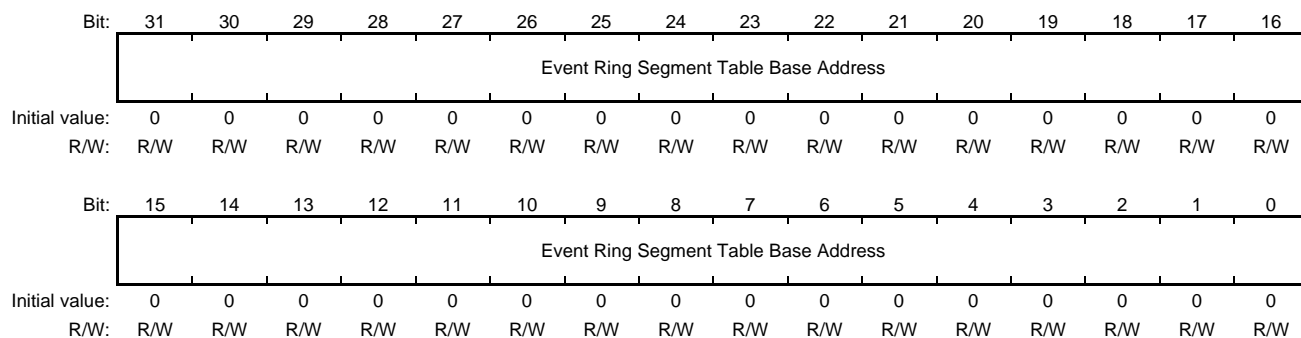
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------------------------|---------------|-----|---|
| 31 to 4 | Event Ring Segment Table Base Address | H'000_0000 | R/W | Start Address of Debug Capability Event Ring Segment Table (lower 32-bit). When define AXI_MASTER_ADDRESS_WIDTH_64 is not set, address Hi is fixed to 0. |
| 3 to 0 | — | All 0 | R | Reserved |

63.2.1.57 xHCI Extended Power Management Capability Register (DCERSTBA1)

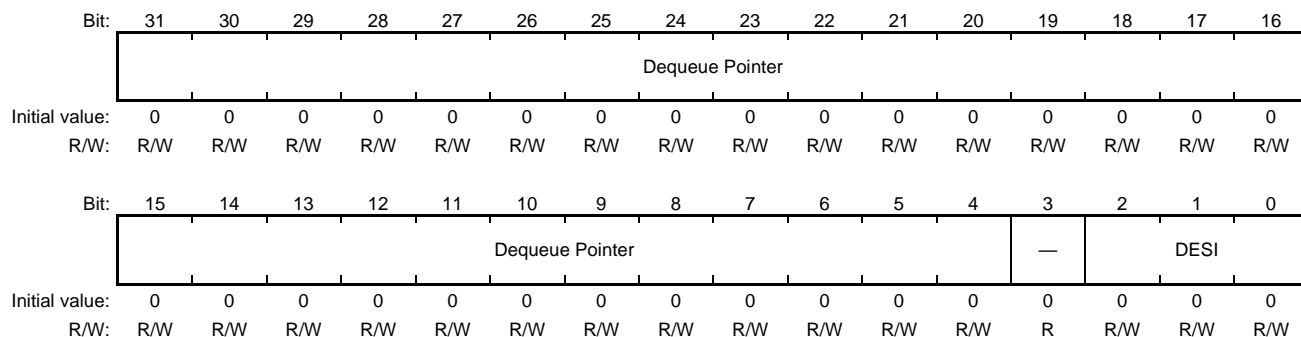
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------------------------|---------------|-----|---|
| 31 to 0 | Event Ring Segment Table Base Address | H'0000_0000 | R/W | Start Address of Debug Capability Event Ring Segment Table (upper 32-bit). When define AXI_MASTER_ADDRESS_WIDTH_64 is not set, address Hi is fixed to 0. |

63.2.1.58 xHCI Extended Power Management Capability Register (DCERDP0)

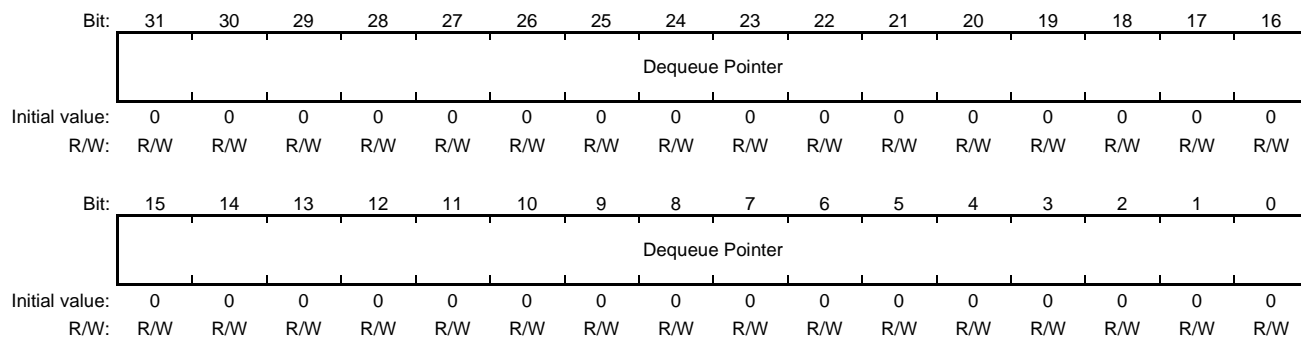
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 31 to 4 | Dequeue Pointer | H'000_0000 | R/W | Address of Debug Capability Event Ring Dequeue Pointer (lower 32-bit). When define AXI_MASTER_ADDRESS_WIDTH_64 is not set, address Hi is fixed to 0. |
| 3 | — | All 0 | R | Reserved |
| 2 to 0 | DESI | B'000 | R/W | Dequeue ERST Segment Index The field is used for the confirmation of the Event Ring full condition by xHC. |

63.2.1.59 xHCI Extended Power Management Capability Register (DCERDP1)

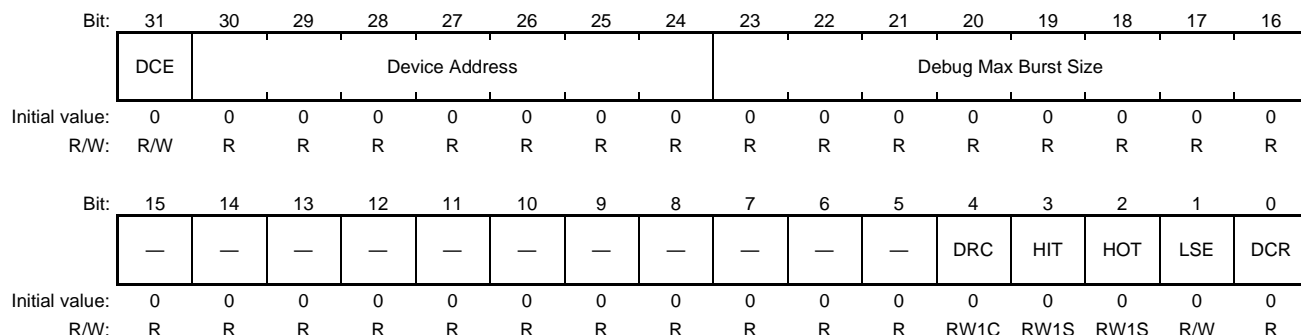
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 31 to 0 | Dequeue Pointer | H'0000_0000 | R/W | Address of Debug Capability Event Ring Dequeue Pointer (upper 32-bit). When define AXI_MASTER_ADDRESS_WIDTH_64 is not set, address Hi is fixed to 0. |

63.2.1.60 xHCI Extended Power Management Capability Register (DCCTRL)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------------|---------------|------|--|
| 31 | DCE | 0 | R/W | Debug Capability Enable If this bit is set to 1, then the xHCI USB Debug Capability is enabled. |
| 30 to 24 | Device Address | H'00 | R | USB device address when debug device is enumerated |
| 23 to 16 | Debug Max Burst Size | H'00 | R | Max Burst Size of Bulk Endpoint in DBC |
| 15 to 5 | — | All 0 | R | Reserved |
| 4 | DRC | B'0 | RW1C | DbC Run Change This bit is set to 1 when DCR is set to 0. Cannot be accessed while this bit is 1. |
| 3 | HIT | B'0 | RW1S | Halt IN TR When this bit changes from 0 to 1, the HIT_SET bit in the DTU.STS2 register is set to 1. This bit status doesn't influence EP1 OUT Endpoint. This bit is fixed to 0 while DCR is 0. |
| 2 | HOT | B'0 | RW1S | Halt OUT TR When this bit changes from 0 to 1, the HOT_SET bit in the DTU.STS2 register is set to 1. This bit status doesn't influence EP1 OUT Endpoint. This bit is fixed to 0 while DCR is 0. |
| 1 | LSE | B'0 | R/W | Link Status Event Enable This bit is set to 1 when the Port Link Status Change bit changes from 0 to 1, after which a Port Status Change Event is generated. |
| 0 | DCR | B'0 | R | DbC Run 0: Debug Device is not in Configured State 1: Debug Device is in Configured State |

63.2.1.61 xHCI Extended Power Management Capability Register (DCST)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Debug Port Number | | | | | | | | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | ER |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------|---------------|-----|--|
| 31 to 24 | Debug Port Number | H'00 | R | Root Hub Port Number when Debug Capability is connected. |
| 23 to 1 | — | All 0 | R | Reserved |
| 0 | ER | B'0 | R | Event Ring Not Empty 0: Debug Capability Event Ring is empty 1: There is a Transfer Event in the Debug Capability Event Ring |

63.2.1.62 xHCI Extended Power Management Capability Register (DCPORTSC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

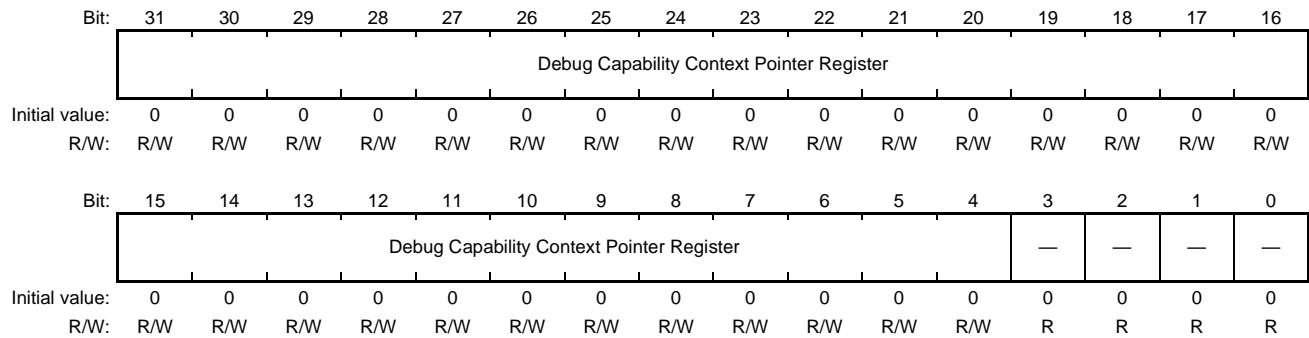
| | | | | | | | | | | | | | | | | |
|----------------|----|----|------------|----|----|----|-----|----|------|------|------|----|----|-----|------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | CEC | PLC | PRC | — | — | — | CSC | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | RW1C | RW1C | RW1C | R | R | R | RW1C | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | Port Speed | | | — | PLS | | | | PR | — | — | PED | CC | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|------|--|
| 31 to 24 | — | All 0 | R | Reserved |
| 23 | CEC | B'0 | RW1C | Port Config Error Change This bit is set to 1 when configuration with Link Partner fails. |
| 22 | PLC | B'0 | RW1C | Port Link Status Change This bit is set to 1 when PLC performs the following transition; <Transition Condition> U0 → U3 Suspend signaling detected from Debug Host U3 → U0 Resume complete Polling → Disabled Training Error Ux or Recovery → Inactive Error When DCE is 0, this bit is set to 0. |
| 21 | PRC | B'0 | RW1C | Port Reset Change After Port Reset status finishes, this bit is set to 1 (PR 1 → 0) When DCE is 0, this bit is set to 0. |
| 20 to 18 | — | All 0 | R | Reserved |
| 17 | CSC | B'0 | RW1C | Connect Status Change This bit is set to 1 when the CCS bit is changed. When DCE is 0, this bit is set to 0. |
| 16 to 14 | — | All 0 | R | Reserved |
| 13 to 10 | Port Speed | B'0000 | R | CCS = 1: 4 (SuperSpeed) CCS = 0: fixed to 0 |
| 9 | — | B'0 | R | Reserved |
| 8 to 5 | PLS | H'4 | R | Port Link Status DCE = 0: fixed to H'4 (Disabled) DCE = 1 and Debug Port Number = 0: fixed to H'5 (RxDetect) |
| 4 | PR | B'0 | R | Port Reset When this bit is 1, Port is in Reset. |
| 3, 2 | — | All 0 | R | Reserved |
| 1 | PED | B'0 | R/W | Port Enable/Disable 0: Disabled 1: Enabled When DCE is 0 or CCS is 0, this bit is set to 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | CC | B'0 | R | Current Connect Status If Debug Port Number is other than 0, this bit is 1. |

63.2.1.63 xHCI Extended Power Management Capability Register (DCCP0)

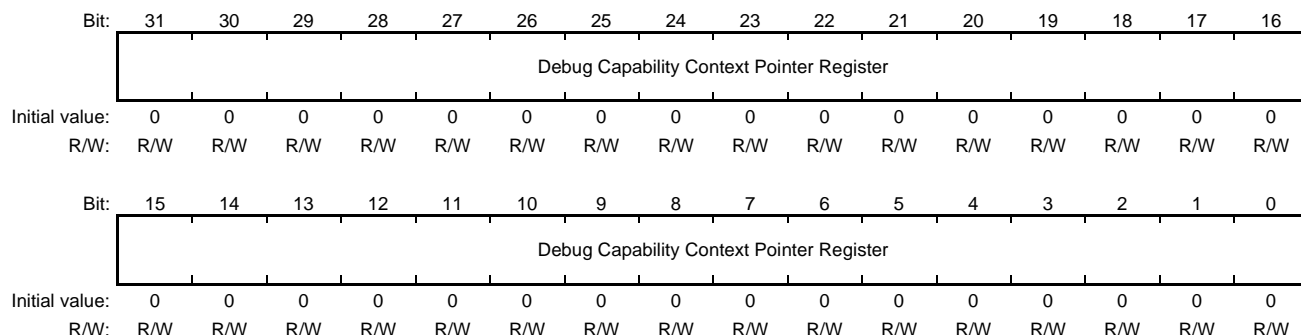
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---|---------------|-----|---|
| 31 to 4 | Debug Capability Context Pointer Register | H'000_0000 | R/W | Beginning address of Debug Capability Context Pointer (lower 32-bit) When define AXI_MASTER_ADDRESS_WIDTH_64 is not set, address Hi is fixed to 0. |
| 3 to 0 | — | All 0 | R | Reserved |

63.2.1.64 xHCI Extended Power Management Capability Register (DCCP1)

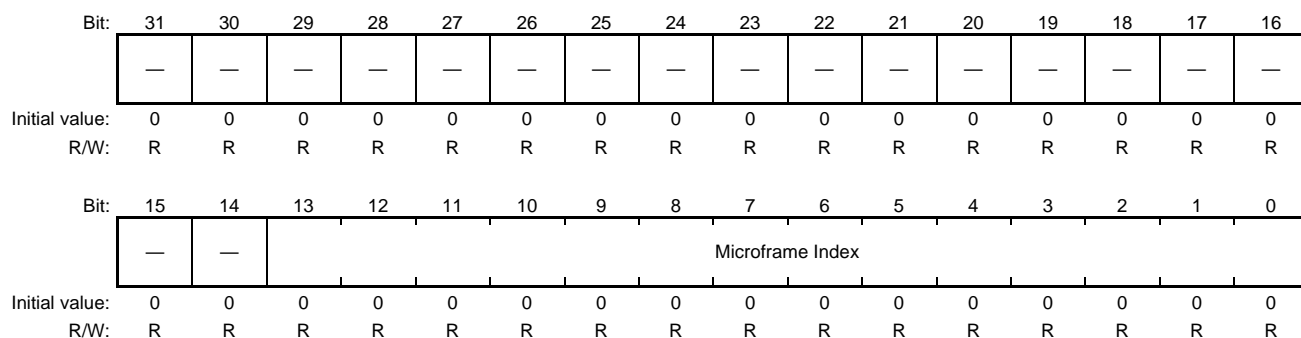
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---|---------------|-----|---|
| 31 to 0 | Debug Capability Context Pointer Register | H'0000_0000 | R/W | Beginning address of Debug Capability Context Pointer (upper 32-bit) When define AXI_MASTER_ADDRESS_WIDTH_64 is not set, address Hi is fixed to 0. |

63.2.1.65 Host Controller Runtime Register (MFINDEX)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|--|
| 31 to 14 | — | All 0 | R | Reserved |
| 13 to 0 | Microframe Index | H'0000 | R | Incremented at the end of every microframe (125 μs) Bits [13:3] are used to define the current 1 ms. Frame Index |

63.2.1.66 Host Controller Runtime Register (IMAN)

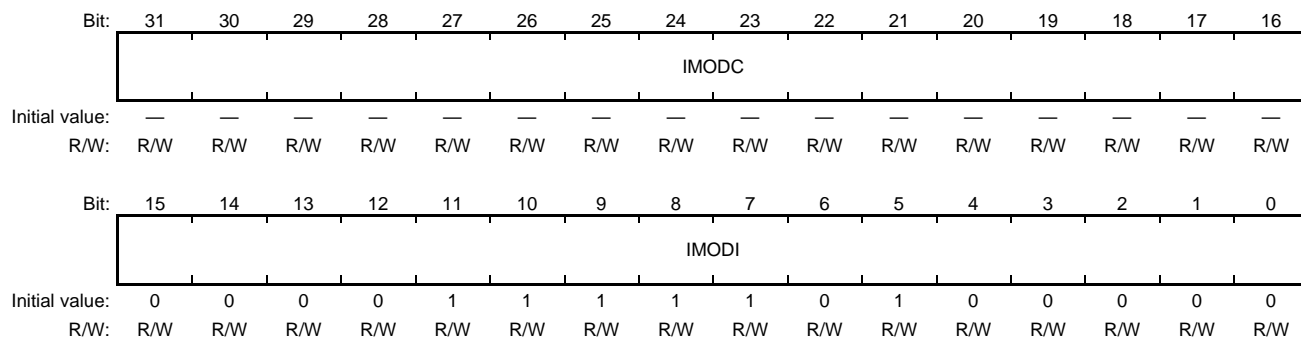
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | IE | IP | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | RW1C |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|----------|---|
| 31 to 2 | — | All 0 | R | Reserved |
| 1 | IE | B'0 | R/W | Interrupt Enable Indicates whether Interrupter can generate an interrupt or not. 0: Does not generate an Interrupt. 1: IP bit is 1, Interrupter Moderation Counter changes to 0 then Interrupter generates an interrupt. |
| 0 | IP | B'0 | RW 1C | Interrupt Pending Current status of Interrupter. 0: Event Ring for Interrupter is empty. There is no pending interrupt. 1: Event Ring for Interrupter is not empty. For interrupter 0, interrupt is pending. |

63.2.1.67 Host Controller Runtime Register (IMOD)

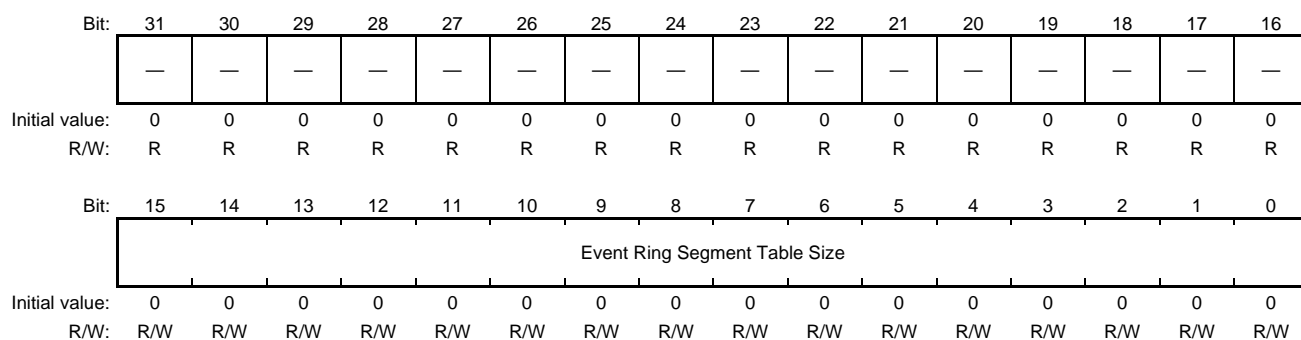
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | IMODC | undefined | R/W | Down Counter: If IP is cleared to 0, the interval value is loaded and then counted down until 0, at which point the 0 counter will stop. An interrupt is generated when the counter is 0 but Event Ring is not empty and IM and IP bits are 1. Anytime Software can write this counter to change the interrupt rate. |
| 15 to 0 | IMODI | H'0FA0 | R/W | Minimum value for the inter-interrupt interval. This interval can be set in increments of 250 ns. 0 is not permitted. |

63.2.1.68 Host Controller Runtime Register (ERSTSZ)

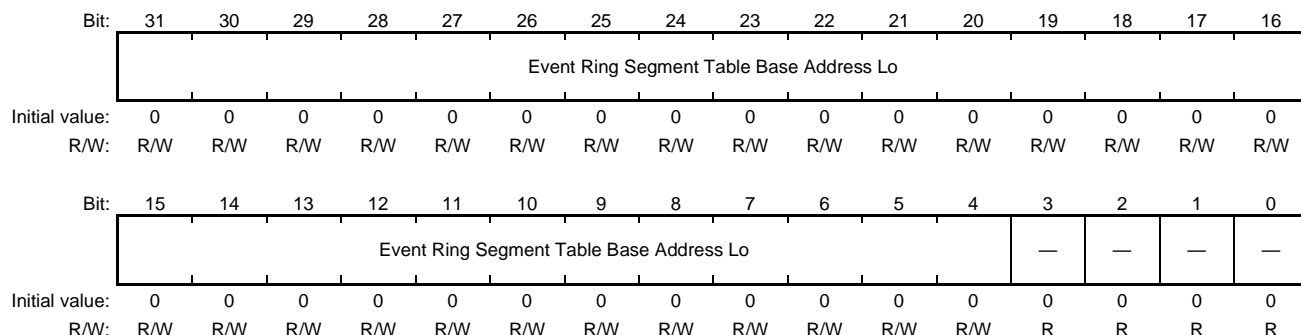
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------------------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 to 0 | Event Ring Segment Table Size | H'0000 | R/W | Defines the number of valid Event Ring Segment Table entries in the Event Ring Segment Table indicated by Event Ring Segment Table Base Address register. The maximum value supported by xHC is defined by ERST Max field in HSCPARAMS2 Register. xHC ignores bit writing when the Run/Stop bit is set to 1. |

63.2.1.69 Host Controller Runtime Register (ERSTBA_L)

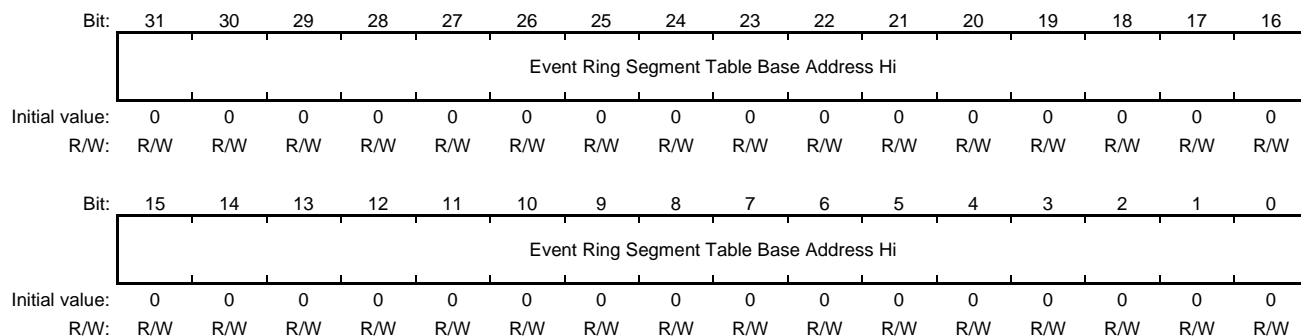
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--|---------------|-----|---|
| 31 to 4 | Event Ring Segment Table Base Address Lo | H'000_0000 | R/W | Defines the Start Address (Lo) of the Event Ring Segment Table. Upon a write to this Register, the Event Ring State Machine transitions to the Start State. |
| 3 to 0 | — | All 0 | R | Reserved |

63.2.1.70 Host Controller Runtime Register (ERSTBA_H)

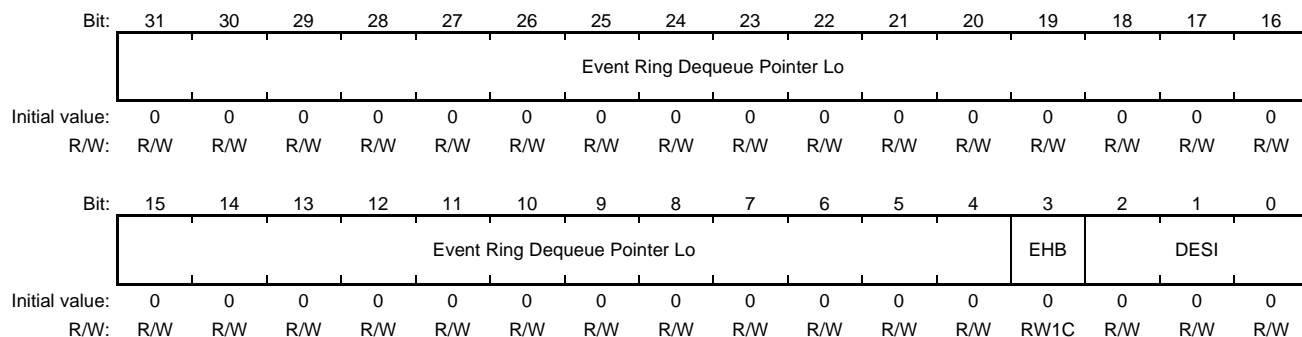
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--|---------------|-----|---|
| 31 to 0 | Event Ring Segment Table Base Address Hi | H'0000_0000 | R/W | Defines the Start Address (Hi) of the Event Ring Segment Table. Upon a write to this Register, the Event Ring State Machine transitions to the Start State. When define AXI_MASTER_ADDRESS_WIDTH_64 is not set, address Hi is fixed to 0. |

63.2.1.71 Host Controller Runtime Register (ERDP_L)

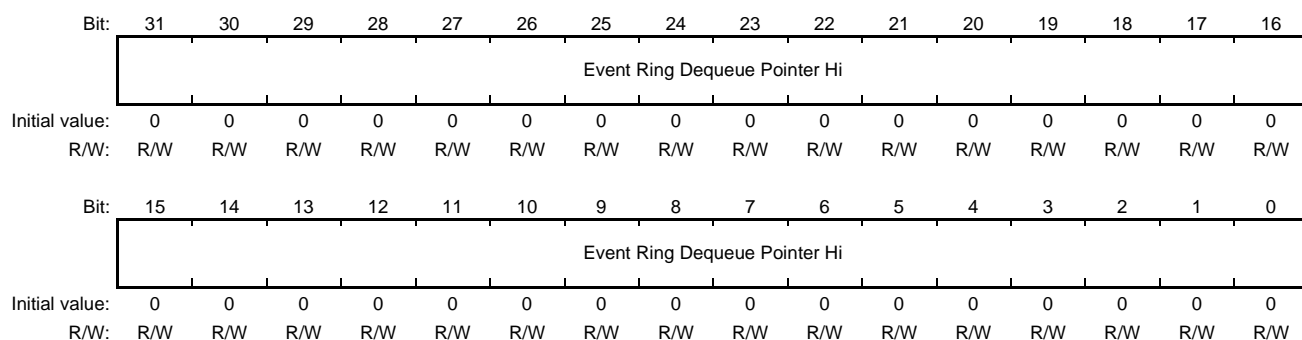
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------------------------|---------------|------|--|
| 31 to 4 | Event Ring Dequeue Pointer Lo | H'000_0000 | R/W | Defines the Address (Lo) of the Current Dequeue Pointer. Software initializes the Register before the Run/Stop bit is set to 1. |
| 3 | EHB | B'0 | RW1C | Event Handler Busy This flag shall be set to '1' when the IP bit is set to '1' and cleared to '0' by software when the Dequeue Pointer register is written. |
| 2 to 0 | DESI | B'000 | R/W | Dequeue ERST Segment Index This field is used for confirmation of the Event Ring full condition by xHC. |

63.2.1.72 Host Controller Runtime Register (ERDP_H)

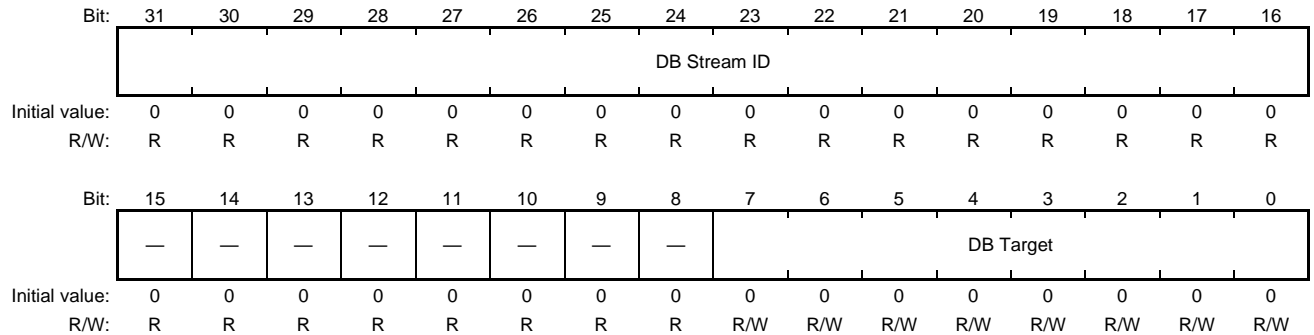
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------------------------|---------------|-----|---|
| 31 to 0 | Event Ring Dequeue Pointer Hi | H'0000_0000 | R/W | Defines the Address (Hi) of Current Dequeue Pointer. Software initializes this Register before the Run/Stop bit is set to 1. When define AXI_MASTER_ADDRESS_WIDTH_64 is not set, address Hi is fixed to 0. |

63.2.1.73 Host Controller Doorbell Register (DOORBELL0)

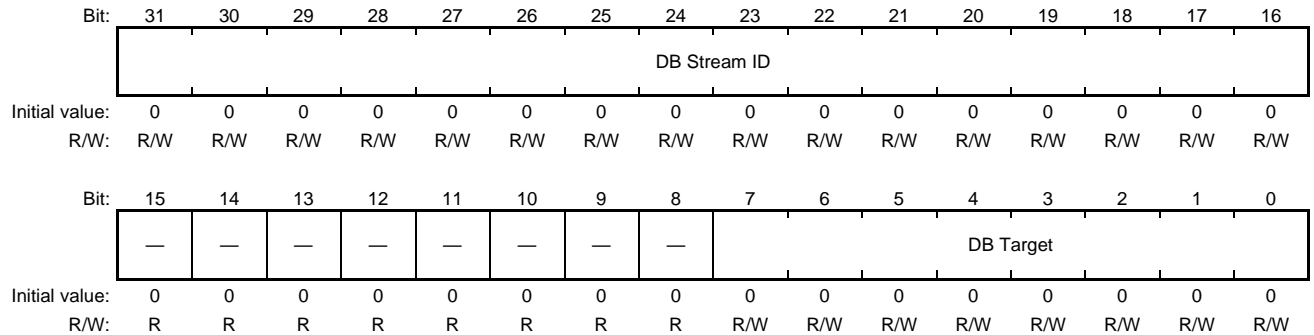
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 16 | DB Stream ID | H'0000 | R | Fixed to H'0000 |
| 15 to 8 | — | All 0 | R | Reserved |
| 7 to 0 | DB Target | H'00 | R/W | Doorbell Target. For Device Context. Command Ring Host Controller Doorbell (0) Value Definition H'00: Host Controller Command H'01 to H'F7: Setting prohibited H'F8 to H'FF: Vendor Defined If the set value is other than 0, it is ignored. The read Value is 0. When the Host Controller Doorbell (0) is set, DB Stream ID field is set to 0. |

63.2.1.74 Host Controller Doorbell Register (DOORBELL1-32)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|--|
| 31 to 16 | DB Stream ID | H'0000 | R/W | Indicates targeted Endpoint's Stream when Endpoint defines Stream. If Endpoint doesn't define Stream (Max PStream = 0), but this bit is not set to 0, then this setting is ignored. The read Value is 0. |
| 15 to 8 | — | All 0 | R | Reserved |
| 7 to 0 | DB Target | H'00 | R/W | Doorbell Target. For Device Context Device Context Doorbells (1 to 255) Value Definition H'00: Setting prohibited H'01: Control EP 0 Enqueue Pointer Update H'02: EP 1 OUT Enqueue Pointer Update H'03: EP 1 IN Enqueue Pointer Update H'04: EP 2 OUT Enqueue Pointer Update H'05: EP 2 IN Enqueue Pointer Update ... H'1E: EP 15 OUT Enqueue Pointer Update H'1F: EP 15 IN Enqueue Pointer Update H'20 to H'F7: Setting prohibited H'F8 to H'FF: Vendor Defined The read Value is 0. |

63.2.2 USB3.0 Peripheral Register

63.2.2.1 AXI Control Register (AXI_CON)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------------|---------------------|-----|----|----|---------------------|----|----|------------|-----|-----|-----|----|-----------|-----|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | AXI_RE SET | — | — | — | — | — | — | — | MST_PCACHE | | | — | — | MST_BSIZE | | |
| Initial value: | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| R/W: | W | R/W | R/W | W | R | R | R | W | R/W | R/W | R/W | R/W | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | Renesas Private bit | | | — | Renesas Private bit | | — | — | — | — | — | — | — | — | MST_WAIT0 |
| Initial value: | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 | AXI_RESET | B'0 | W | Force reset for AXI-IF. Force reset to AXI Master Block when 1 is written in this bit. Read value of this bit is always 0. Reset is done to area except Slave Control Circuit of AXI Bridge and Registers. PRD_EN* shall be "0" in all register field before setting this bit when reset is pursued forcibly |
| 30 | — | B'0 | R/W | The initial value of this bit is 0. This bit can be read and written but it is not allowed to change the value of this bit. Only 0 is allowed as the value to be written. |
| 29 | — | B'1 | R/W | The initial value of this bit is 1. This bit can be read and written but it is not allowed to change the value of this bit. Only 1 is allowed as the value to be written. |
| 28 | — | B'0 | W | This bit is writable and write-only bit but is not allowed to write 1 to this bit. Only 0 is allowed as the value to be written. |
| 27 to 25 | — | All 0 | R | Reserved |
| 24 | — | B'0 | W | This bit is writable and write-only bit but is not allowed to write 1 to this bit. Only 0 is allowed as the value to be written. |
| 23 to 20 | MST_PCACHE | H'0 | R/W | Specifies CACHE signal (MARCACHE/MAWCACHE) for access to Read and Write Back PRD Table in AXI Master IF. This is not applied to CACHE signal for DATA transfer with DMA. Setting of Descriptor in PRD Table is applied during DATA Transfer. This setting should not be changed during DMA Transfer. |
| 19 to 18 | — | All 0 | R | Reserved |
| 17 to 16 | MST_BSIZE | B'11 | R/W | Specifies burst size (AxSIZE) in AXI Master IF. Specifies burst size of transfer that is initiated by Master IF. This setting should not be changed during DMA Transfer. Check transfer is stopped with DMA_TRANS0 after setting MST_WAIT0 in order to change the setting during transfer. B'00: 1 Byte (AxSIZE = B'000) B'01: 2 Byte (AxSIZE = B'001) B'10: 4 Byte (AxSIZE = B'010) B'11: 8 Byte (AxSIZE = B'011) |
| 15 | — | B'0 | R | Reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|---|
| 14 to 12 | Renesas Private bit | B'010 | R | Renesas Private field. This is write inhibit field. |
| 11 | — | B'0 | R | Reserved |
| 10 to 8 | Renesas Private bit | B'010 | R | Renesas Private field. This is write inhibit field. |
| 7 to 1 | — | All 0 | R | Reserved |
| 0 | MST_WAIT0 | B'0 | R/W | Request DMA Transfer of all PRD table in AXI Master IF to be suspended. In case suspension of transfer is requested with this bit during processing transferring, it indicates 1 (during DMA transfer) by AXI_STA/DMA_TRANS bit. |

63.2.2.2 AXI Status Register (AXI_STA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------------|----|----|----|----|----|-----------|------------|--------------|--------------|--------------|--------------|-----------|-----------|-----------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | AXI_RESET_STA | — | — | — | — | — | — | — | EXTPRD_EN4_R | EXTPRD_EN3_R | EXTPRD_EN2_R | EXTPRD_EN1_R | PRD_EN4_R | PRD_EN3_R | PRD_EN2_R | PRD_EN1_R |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PRD_DIR0 | — | — | — | — | — | DMA_PRDNO | PRD_WRITE0 | PRD_READ0 | DMA_PIPE0 | | | | | | DMA_TRANS0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 | AXI_RESET_STA | B'0 | R | Indicate Reset Status of AXI-IF. 1: Reset State 0: Reset is cancelled |
| 30 to 24 | — | All 0 | R | Reserved |
| 23 | EXTPRD_EN4_R | B'0 | R | Indicates status of EXTPRD_EN4. This bit is dedicated for Read of bit 1 (EXTPRD_EN4) of DMA_Ch0_CON4 Register |
| 22 | EXTPRD_EN3_R | B'0 | R | Indicates status of EXTPRD_EN3. This bit is dedicated for Read of bit 1 (EXTPRD_EN3) of DMA_Ch0_CON3 Register |
| 21 | EXTPRD_EN2_R | B'0 | R | Indicates status of EXTPRD_EN2. This bit is dedicated for Read of bit 1 (EXTPRD_EN2) of DMA_Ch0_CON2 Register |
| 20 | EXTPRD_EN1_R | B'0 | R | Indicates status of EXTPRD_EN1. This bit is dedicated for Read of bit 1 (EXTPRD_EN1) of DMA_Ch0_CON1 Register |
| 19 | PRD_EN4_R | B'0 | R | Indicate status of PRD_EN4. This bit is dedicated for Read of bit 0 (PRD_EN4) of DMA_Ch0_CON4 Register. |
| 18 | PRD_EN3_R | B'0 | R | Indicate status of PRD_EN3. This bit is dedicated for Read of bit 0 (PRD_EN3) of DMA_Ch0_CON3 Register. |
| 17 | PRD_EN2_R | B'0 | R | Indicate status of PRD_EN2. This bit is dedicated for Read of bit 0 (PRD_EN2) of DMA_Ch0_CON2 Register. |
| 16 | PRD_EN1_R | B'0 | R | Indicate status of PRD_EN1. This bit is dedicated for Read of bit 0 (PRD_EN1) of DMA_Ch0_CON1 Register. |
| 15 | PRD_DIR0 | B'0 | R | Indicate the direction of transfer for the PIPE where is current processing at AXI Master I/F. This bit indicates the direction of transfer for the PIPE where is currently processing when the transfer is processing (DMA_TRANS = 1). This bit indicates the direction of transfer for the PIPE where was transferred at the last. |
| 14 to 10 | — | All 0 | R | Reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|------------|---------------|-----|---|
| 9 to 8 | DMA_PRDN0 | B'00 | R | Indicate PRD table number that is in process in AXI Master IF. This bit indicates PRD Table number during DMA transfer that is in progress (DMA_TRANS = 1) and PRD Table number that was transferred last time when in stand-by state (DMA_TRANS = 0). |
| 7 | PRD_WRITE0 | B'0 | R | Indicate that PRD Table is Write in AXI Master IF. This bit indicates that Write of PRD Table of PIPE number indicated by DMA_PIPE is in progress. 1: Read of PRD Table of PIPE number indicated by DMA_PIPE is progress. 0: Others |
| 6 | PRD_READ0 | B'0 | R | Indicate that PRD Table is Read in AXI Master IF. This bit indicates that Read of PRD Table of PIPE number indicated by DMA_PIPE is in progress. 1: Read of PRD Table of PIPE number indicated by DMA_PIPE is progress. 0: Others |
| 5 to 1 | DMA_PIPE0 | All 0 | R | Indicate PIPE number that is currently being processed in AXI Master IF. This bit indicates PIPE number that is currently being transferred in case DMA Transfer in progress (DMA_TRANS = 1) or PIPE number that was transferred at the end in case it is in Stand-by state (DMA_TRANS = 0). |
| 0 | DMA_TRANS0 | B'0 | R | Indicate DMA Transfer is in progress in AXI Master IF. DMS transfer control is performed by packet unit in USB. This bit indicates EPC buffer and data transfer is in progress with packet unit. 1: DMA Transfer in Progress 0: Stand-by |

63.2.2.3 AXI Interrupt Status Register (AXI_INT_STA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|----|----|----|----|----|----|----|----|----|----|----------------|----------------|----------------|----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | DMAINT_STA | EPCINT_STA | — | — | — | — | — | — | — | — | — | — | PRDEN4_CLR_STA | PRDEN3_CLR_STA | PRDEN2_CLR_STA | PRDEN1_CLR_STA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | RW1C | RW1C | RW1C | RW1C |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | PRDER0-4_STA | PRDER0-3_STA | PRDER0-2_STA | PRDER0-1_STA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | RW1C | RW1C | RW1C | RW1C |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|------|--|
| 31 | DMAINT_STA | B'0 | R | Indicate an Interrupt occurred due to status change during DMA Transfer. Interrupt with AXI_INT_DMA occurs when this bit is set. Refer to DMA_INT_STA Register to check at which PIPE Transfer was completed in case this bit is set. This bit is cleared when all factors of DMA_INT_STA are cleared. 1: DMA Transfer Completion Interrupt occurred 0: Interrupt has not occurred |
| 30 | EPCINT_STA | B'0 | R | Indicates an Interrupt occurred due to EPC_INTR_USB from EPC. Need to check That factor for Interrupt due to EPC register and to clear the status of Interrupt when this bit is set. This bit is cleared when all status of EPC is cleared. 1: DMA Transfer Completion Interrupt occurred 0: Interrupt has not occurred |
| 29 to 20 | — | All 0 | R | Reserved |
| 19 | PRDEN4_CLR_STA | B'0 | RW1C | Indicate that PRD_EN that is set in CH0-4 is cleared by HW. Interrupt with AXI_INT_SYS occurs when this bit is set. This bit may be set for the interruption when PRD_EN is ready for clearing by HW due to the completion of targeted transfer process even though PRD_EN is cleared by FW after the transfer process is started. |
| 18 | PRDEN3_CLR_STA | B'0 | RW1C | Indicate that PRD_EN that is set in CH0-3 is cleared by HW. Interrupt with AXI_INT_SYS occurs when this bit is set. This bit may be set for the interruption when PRD_EN is ready for clearing by HW due to the completion of targeted transfer process even though PRD_EN is cleared by FW after the transfer process is started. |
| 17 | PRDEN2_CLR_STA | B'0 | RW1C | Indicate that PRD_EN that is set in CH0-2 is cleared by HW. Interrupt with AXI_INT_SYS occurs when this bit is set. This bit may be set for the interruption when PRD_EN is ready for clearing by HW due to the completion of targeted transfer process even though PRD_EN is cleared by FW after the transfer process is started. |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------|---------------|----------|--|
| 16 | PRDEN1_ CLR_STA | B'0 | RW 1C | Indicate that PRD_EN that is set in CH0-1 is cleared by HW. Interrupt with AXI_INT_SYS occurs when this bit is set. This bit may be set for the interruption when PRD_EN is ready for clearing by HW due to the completion of targeted transfer process even though PRD_EN is cleared by FW after the transfer process is started. |
| 15 to 4 | — | All 0 | R | Reserved |
| 3 | PRDERR0-4_ STA | B'0 | RW 1C | Indicate that information of PRD Table set in CH0-4 is invalid. This bit is set to 1 when it is determined that PRD Table set in DMA_Ch0_CON4/DMA_Ch0_PRD_ADR4 Register is not valid and applicable PRD Table is Disabled. This is cleared to 0 by writing 1 from FW. This bit is set in any of the case provided below. Refer to "Procedure of Error Occur" in [63.7.5 AXI DMA Transfer Control]. - Target PIPE is duplicated with other setting. - Falsely specified by PIPE_DIR/PIPE_NO. - Format of PRD Table is invalid. - Transfer has occurred. Interrupt with AXI_INT_ERR occurs when this bit is set. |
| 2 | PRDERR0-3_ STA | B'0 | RW 1C | Indicates information of PRD Table set in CH0-3 is invalid. This bit is set to 1 when it is determined that PRD Table set in DMA_Ch0_CON3/DMA_Ch0_PRD_ADR3 Register is not valid and applicable PRD Table is Disabled. This is cleared to 0 by writing 1 from FW. Interrupt with AXI_INT_ERR occurs when this bit is set. |
| 1 | PRDERR0-2_ STA | B'0 | RW 1C | Indicates information of PRD Table set in CH0-2 is invalid. This bit is set to 1 when it is determined that PRD Table set in DMA_Ch0_CON2/DMA_Ch0_PRD_ADR2 Register is not valid and applicable PRD Table is Disabled. This is cleared to 0 by writing 1 from FW. Interrupt with AXI_INT_ERR occurs when this bit is set. |
| 0 | PRDERR0-1_ STA | B'0 | RW 1C | Indicates information of PRD Table set in CH0-1 is invalid. This bit is set to 1 when it is determined that PRD Table set in DMA_Ch0_CON1/DMA_Ch0_PRD_ADR1 Register is not valid and applicable PRD Table is Disabled. This is cleared to 0 by writing 1 from FW. Interrupt with AXI_INT_ERR occurs when this bit is set. |

63.2.2.4 AXI Interrupt Enable Register (AXI_INT_ENA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------|------------|----|----|----|----|----|----|----|----|----|----|----------------|----------------|----------------|----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | DMAINT_ENA | EPCINT_ENA | — | — | — | — | — | — | — | — | — | — | PRDEN4_CLR_ENA | PRDEN3_CLR_ENA | PRDEN2_CLR_ENA | PRDEN1_CLR_ENA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | PRDERR0-4_ENA | PRDERR0-3_ENA | PRDERR0-2_ENA | PRDERR0-1_ENA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|--|
| 31 | DMAINT_ENA | B'0 | R/W | Enable Interrupt of status change during DMA Transfer Completion. In case this bit is set to 0, Output signal AXI_INT_DMA will not be asserted even when an interrupt factor occurred in AXI_INT_STA.DMAINT_STA. 1: Enable an Interrupt 0: Disable an Interrupt |
| 30 | EPCINT_ENA | B'0 | R/W | Enable Interrupt by signal from EPC. In case this bit is set to 0, Output signal AXI_INT_ALL will not be asserted even when EPC_INTR_USB signal is changed. 1: Enable an Interrupt 0: Disable an Interrupt |
| 29 to 20 | — | All 0 | R | Reserved |
| 19 | PRDEN4_CLR_ENA | B'0 | R/W | Enable an Interrupt of PRDEN4_CLR_STA. 1: Enable an Interrupt 0: Disable an Interrupt |
| 18 | PRDEN3_CLR_ENA | B'0 | R/W | Enable an Interrupt of PRDEN3_CLR_STA. 1: Enable an Interrupt 0: Disable an Interrupt |
| 17 | PRDEN2_CLR_ENA | B'0 | R/W | Enable an Interrupt of PRDEN2_CLR_STA. 1: Enable an Interrupt 0: Disable an Interrupt |
| 16 | PRDEN1_CLR_ENA | B'0 | R/W | Enable an Interrupt of PRDEN1_CLR_STA. 1: Enable an Interrupt 0: Disable an Interrupt |
| 15 to 4 | — | All 0 | R | Reserved |
| 3 | PRDERR0-4_ENA | B'0 | R/W | Enable an Interrupt of PRDERR0-4_STA. 1: Enable an Interrupt 0: Disable an Interrupt |
| 2 | PRDERR0-3_ENA | B'0 | R/W | Enable an Interrupt of PRDERR0-3_STA. 1: Enable an Interrupt 0: Disable an Interrupt |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------------|---------------|-----|--|
| 1 | PRDERR0-2_ ENA | B'0 | R/W | Enable an Interrupt of PRDERR0-2_STA. 1: Enable an Interrupt 0: Disable an Interrupt |
| 0 | PRDERR0-1_ ENA | B'0 | R/W | Enable an Interrupt of PRDERR0-1_STA. 1: Enable an Interrupt 0: Disable an Interrupt |

63.2.2.5 DMA Interrupt Status Register (DMA_INT_STA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|------|------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | P5_PR DEND_ STA | P4_PR DEND_ STA | P3_PR DEND_ STA | P2_PR DEND_ STA | P1_PR DEND_ STA | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------------|---------------|----------|---|
| 31 | — | B'0 | R | Reserved |
| 30 to 6 | — | All 0 | RW 1C | This bit can be written only "0". DO NOT write "1". |
| 5 | P5_PRDEND _STA | B'0 | RW 1C | Indicates the change of the status due to DMA transfer for PIPE5. This bit is set to 1 in any of the cases provided below. Make sure to clear it to 0 b writing 1 from FW. - When transfer processes of PRD Tables that are set is completed and PRD_EN is cleared. (Including the case the error is occurred at data transfer when Short Packet or Transfer is detected at PIPE for OUT transfer. This bit is not set when the transfer error is occurred during Read/Write for PRD_Table) - Transfer is ready at PIPE with disabled PRD Table. - Transfer of descriptor with Int bit set is completed. |
| 4 | P4_PRDEND _STA | B'0 | RW 1C | Indicates the change of the status due to DMA transfer for PIPE4. |
| 3 | P3_PRDEND _STA | B'0 | RW 1C | Indicates the change of the status due to DMA transfer for PIPE3. |
| 2 | P2_PRDEND _STA | B'0 | RW 1C | Indicates the change of the status due to DMA transfer for PIPE2. |
| 1 | P1_PRDEND _STA | B'0 | RW 1C | Indicates the change of the status due to DMA transfer for PIPE1. |
| 0 | — | B'0 | R | Reserved |

63.2.2.6 DMA Interrupt Enable Register (DMA_INT_ENA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | P5_PR DEND_ ENA | P4_PR DEND_ ENA | P3_PR DEND_ ENA | P2_PR DEND_ ENA | P1_PR DEND_ ENA | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 31 | — | B'0 | R | Reserved |
| 30 to 6 | — | All 1 | R/W | This bit can be written only "1". DO NOT write "0". |
| 5 | P5_PRDEND_ENA | B'1 | R/W | Enable an Interrupt by the change of the status due to DMA transfer for PIPE5. In case this bit is set to 0, AXI_INT_STA.DMAINT_STA will not be set even when factor to Interrupt occurs 1: Enable an Interrupt 0: Disable an Interrupt |
| 4 | P4_PRDEND_ENA | B'1 | R/W | Enable an Interrupt by the change of the status due to DMA transfer for PIPE4. |
| 3 | P3_PRDEND_ENA | B'1 | R/W | Enable an Interrupt by the change of the status due to DMA transfer for PIPE3. |
| 2 | P2_PRDEND_ENA | B'1 | R/W | Enable an Interrupt by the change of the status due to DMA transfer for PIPE2. |
| 1 | P1_PRDEND_ENA | B'1 | R/W | Enable an Interrupt by the change of the status due to DMA transfer for PIPE1. |
| 0 | — | B'0 | R | Reserved |

63.2.2.7 Data Enable Status Register (DATAEN_STA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-----------|-----------|-----------|-----------|-----------|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | P5_DATAEN | P4_DATAEN | P3_DATAEN | P2_DATAEN | P1_DATAEN | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 | — | B'0 | R | Reserved |
| 30 to 6 | — | All 0 | R | Reserved |
| 5 | P5_DATAEN | B'0 | R | Indicate the status of DATAEN signal for PIPE5. 1: Ready for DATA Transfer 0: Not ready for DATA Transfer |
| 4 | P4_DATAEN | B'0 | R | Indicate the status of DATAEN signal for PIPE4. |
| 3 | P3_DATAEN | B'0 | R | Indicate the status of DATAEN signal for PIPE3. |
| 2 | P2_DATAEN | B'0 | R | Indicate the status of DATAEN signal for PIPE2. |
| 1 | P1_DATAEN | B'0 | R | Indicate the status of DATAEN signal for PIPE1. |
| 0 | — | B'0 | R | Reserved |

63.2.2.8 AXI Common Input Register (AXI_COM_IN)

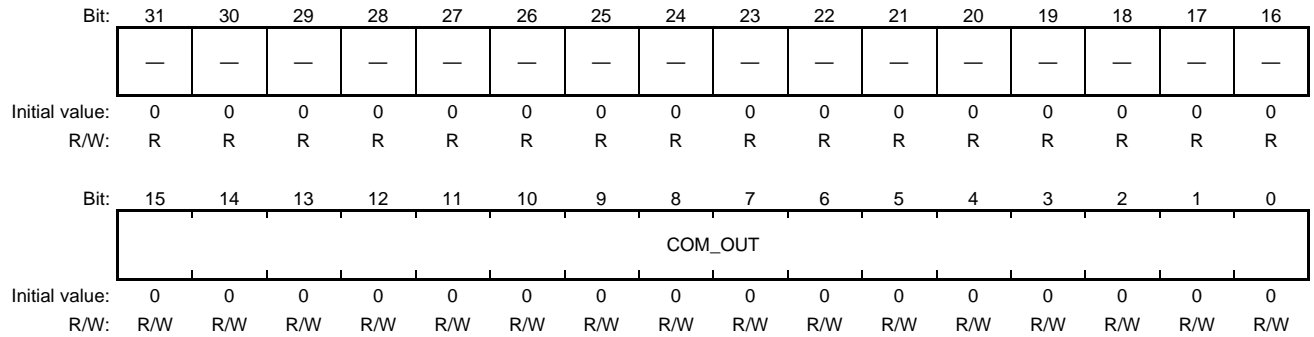
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | COM_IN | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 to 0 | COM_IN | All 0 | R | Indicates the value that was input in Input Pin, COM_IN. |

63.2.2.9 AXI Common Output Register (AXI_COM_OUT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 to 0 | COM_OUT | All 0 | R/W | Output Register value to Output Pin, COM_OUT. |

63.2.2.10 DMA Ch0 Control Register1 (DMA_Ch0_CON1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----------|-----|-----|-----|-----|----|------------|-----|-----|----|--------------|---------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PIPE_DIR1 | — | — | PIPE_NO1 | | | | — | — | EXTPRD_NO1 | | — | — | EXTPRD_D_EN1 | PRD_EN1 | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 | PIPE_DIR1 | B'0 | R/W | Specify Transfer Direction in AXI Master IF. 1: In Transfer (Memory → EPC) 0: Out Transfer (EPC → Memory) |
| 14 to 13 | — | All 0 | R | Reserved |
| 12 to 8 | PIPE_NO1 | All 0 | R/W | Specify PIPE number that this PRD Table is applied. Setting value between 1 and 5 is valid. (Note) |
| 7 to 6 | — | All 0 | R | Reserved |
| 5 to 4 | EXTPRD_NO1 | B'00 | R/W | Specify PRD Table number that is linked to this PRD Table. This is valid only when PRD_EN1 is set to 1. B'00: Reserved B'01: PRD_EN2 B'10: PRD_EN3 B'11: PRD_EN4 |
| 3 to 2 | — | All 0 | R | Reserved |
| 1 | EXTPRD_EN1 | B'0 | R/W | Enable Link of PRD Table by EXTPRD_NO1. In case both PRD_EN1 and this bit are set to 1, setting of this PRD Table is enabled upon completion of transfer of PRD Table that was specified by PRD_NO1. This bit is cleared automatically when PRD Table is enabled. However, this is not cleared when transfer of linked PRD Table has been completed. Confirm that PRD_EN that was linked is still 1 after setting this bit. 1: PRD Table is linked 0: PRD Table is not linked |

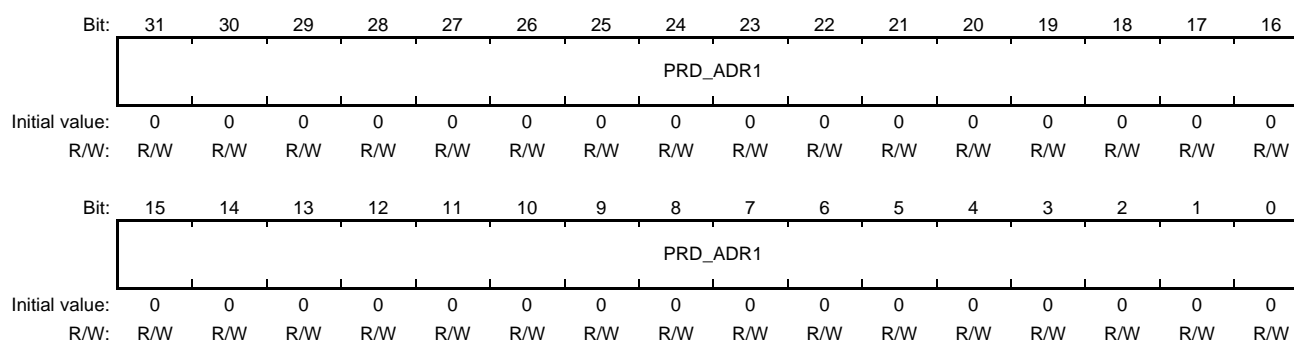
| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | PRD_EN1 | B'0 | R/W | <p>Allow DMA Transfer of PIPEn in AXI Master.</p> <p>Make sure to set this bit after preparing PRD Table and setting address where PRD Table for applicable PIPE is stored from DMA_Ch0_PRD_ADR1 Register.</p> <p>DMA Transfer is enabled as PRD Table in the address set in DMA_Ch0_PRD_ADR1 is valid when PRD_EN1 is 1 and EXTPRD_EN1 is 0. (Note)</p> <p>This is cleared to 0 automatically in case the process specified by PRD Table is completed or the process is aborted due to an Error.</p> <p>1: Allow DMA Transfer 0: Stop DMA Transfer</p> |

Note:

- Those registers shall not be set in each PRD_Table to be valid multiple PRD Tables for same PIPE number.
- PRD_EN is not set to 1 when PRDERR status is set due to recognizing a mistake of setting for PIPE number during requesting for writing of PRD_EN = 1

63.2.2.11 DMA Ch0 PRD Address Register1 (DMA_Ch0_PRD_ADR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | PRD_ADR1 | H'0000_0000 | R/W | <p>Specify address of Physical Region Descriptor (PRD) Table that is located in memory space.</p> <p>Indicates PRD Table Address where is using in case this Register is read during transfer at the applicable PRD Table. Indicates the PRD Table Address where is used when next transfer is started in case reading this bit with stopping the transfer</p> <p>Note that lower 3 bits are all fixed to "0" (Reserved).</p> <p>It is unknown value when PRD Table is completed.</p> |

63.2.2.12 DMA Ch0 Control Register2 (DMA_Ch0_CON2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----------|-----|-----|-----|-----|----|------------|-----|-----|----|--------------|---------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PIPE_DIR2 | — | — | PIPE_NO2 | | | | — | — | EXTPRD_NO2 | | — | — | EXTPRD_D_EN2 | PRD_EN2 | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 | PIPE_DIR2 | B'0 | R/W | Specify Transfer Direction in AXI Master IF. 1: In Transfer (Memory → EPC) 0: Out Transfer (EPC → Memory) |
| 14 to 13 | — | All 0 | R | Reserved |
| 12 to 8 | PIPE_NO2 | All 0 | R/W | Specify PIPE number that this PRD Table is applied. Setting value between 1 and 5 is valid. (Note) |
| 7 to 6 | — | All 0 | R | Reserved |
| 5 to 4 | EXTPRD_NO2 | B'00 | R/W | Specify PRD Table number that is linked to this PRD Table. This is valid only when PRD_EN2 is set to 1. B'00: PRD_EN1 B'01: Setting prohibited B'10: PRD_EN3 B'11: PRD_EN4 |
| 3 to 2 | — | All 0 | R | Reserved |
| 1 | EXTPRD_EN2 | B'0 | R/W | Enable Link of PRD Table by EXTPRD_NO2. In case both PRD_EN2 and this bit are set to 1, setting of this PRD Table is enabled upon completion of transfer of PRD Table that was specified by PRD_NO2. This bit is cleared automatically when PRD Table is enabled. However, this is not cleared when transfer of linked PRD Table has been completed. Confirm that PRD_EN that was linked is still 1 after setting this bit. 1: PRD Table is linked 0: PRD Table is not linked |

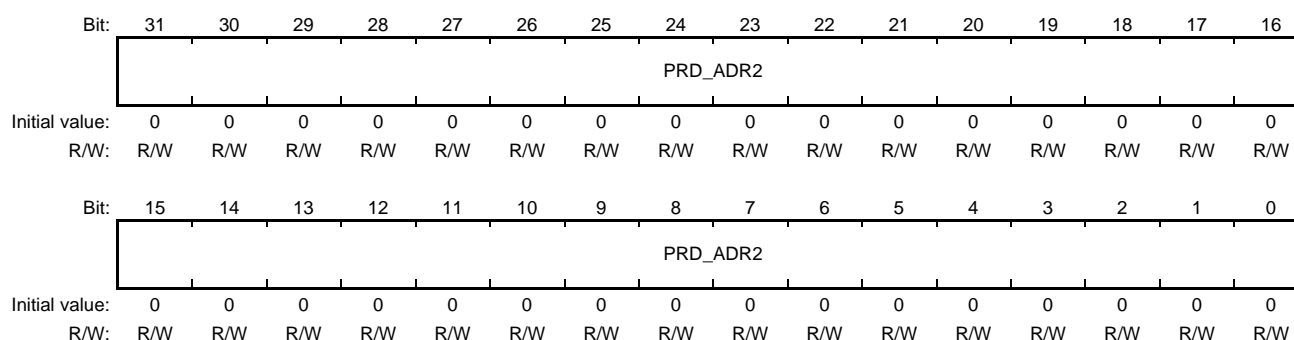
| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | PRD_EN2 | B'0 | R/W | <p>Allow DMA Transfer of PIPEn in AXI Master.</p> <p>Make sure to set this bit after preparing PRD Table and setting address where PRD Table for applicable PIPE is stored from DMA_Ch0_PRD_ADR2 Register.</p> <p>DMA Transfer is enabled as PRD Table in the address set in DMA_Ch0_PRD_ADR2 is valid when PRD_EN2 is 1 and EXTPRD_EN2 is 0. (Note)</p> <p>This is cleared to 0 automatically in case the process specified by PRD Table is completed or the process is aborted due to an Error.</p> <p>1: Allow DMA Transfer 0: Stop DMA Transfer</p> |

Note:

- Those registers shall not be set in each PRD_Table to be valid multiple PRD Tables for same PIPE number.
- PRD_EN is not set to 1 when PRDERR status is set due to recognizing a mistake of setting for PIPE number during requesting for writing of PRD_EN = 1

63.2.2.13 DMA Ch0 PRD Address Register2 (DMA_Ch0_PRD_ADR2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | PRD_ADR2 | H'0000_0000 | R/W | <p>Specify address of Physical Region Descriptor (PRD) Table that is located in memory space.</p> <p>Indicates PRD Table Address where is using in case this Register is read during transfer at the applicable PRD Table. Indicates the PRD Table Address where is used when next transfer is started in case reading this bit with stopping the transfer</p> <p>Note that lower 3 bits are all fixed to "0" (Reserved).</p> <p>It is unknown value when PRD Table is completed.</p> |

63.2.2.14 DMA Ch0 Control Register3 (DMA_Ch0_CON3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----------|-----|-----|-----|-----|----|------------|-----|-----|----|--------------|---------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PIPE_DIR3 | — | — | PIPE_NO3 | | | | — | — | EXTPRD_NO3 | | — | — | EXTPRD_D_EN3 | PRD_EN3 | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 | PIPE_DIR3 | B'0 | R/W | Specify Transfer Direction in AXI Master IF. 1: In Transfer (Memory → EPC) 0: Out Transfer (EPC → Memory) |
| 14 to 13 | — | All 0 | R | Reserved |
| 12 to 8 | PIPE_NO3 | All 0 | R/W | Specify PIPE number that this PRD Table is applied. Setting value between 1 and 5 is valid. (Note) |
| 7 to 6 | — | All 0 | R | Reserved |
| 5 to 4 | EXTPRD_NO3 | B'00 | R/W | Specify PRD Table number that is linked to this PRD Table. This is valid only when PRD_EN3 is set to 1. B'00: PRD_EN1 B'01: PRD_EN2 B'10: Setting prohibited B'11: PRD_EN4 |
| 3 to 2 | — | All 0 | R | Reserved |
| 1 | EXTPRD_EN3 | B'0 | R/W | Enable Link of PRD Table by EXTPRD_NO3. In case both PRD_EN3 and this bit are set to 1, setting of this PRD Table is enabled upon completion of transfer of PRD Table that was specified by PRD_NO3. This bit is cleared automatically when PRD Table is enabled. However, this is not cleared when transfer of linked PRD Table has been completed. Confirm that PRD_EN that was linked is still 1 after setting this bit. 1: PRD Table is linked 0: PRD Table is not linked |

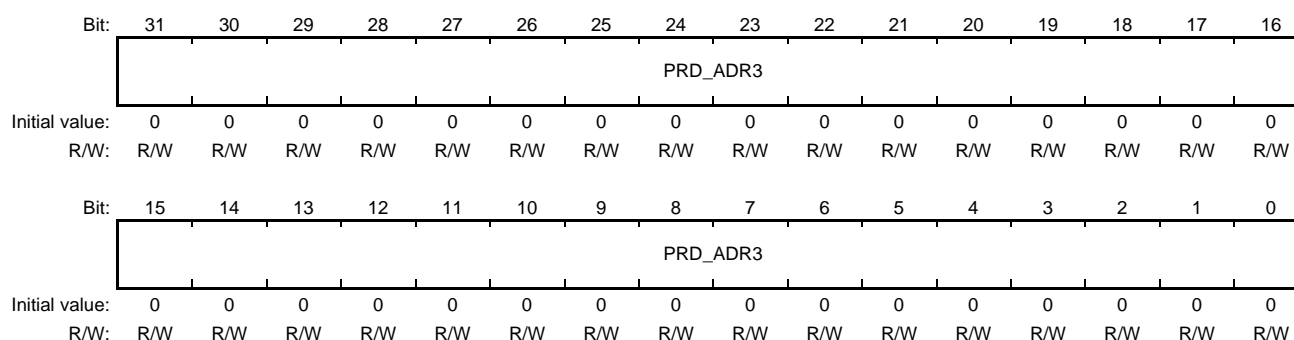
| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | PRD_EN3 | B'0 | R/W | <p>Allow DMA Transfer of PIPEn in AXI Master.</p> <p>Make sure to set this bit after preparing PRD Table and setting address where PRD Table for applicable PIPE is stored from DMA_Ch0_PRD_ADR3 Register.</p> <p>DMA Transfer is enabled as PRD Table in the address set in DMA_Ch0_PRD_ADR3 is valid when PRD_EN3 is 1 and EXTPRD_EN3 is 0. (Note)</p> <p>This is cleared to 0 automatically in case the process specified by PRD Table is completed or the process is aborted due to an Error.</p> <p>1: Allow DMA Transfer 0: Stop DMA Transfer</p> |

Note:

- Those registers shall not be set in each PRD_Table to be valid multiple PRD Tables for same PIPE number.
- PRD_EN is not set to 1 when PRDERR status is set due to recognizing a mistake of setting for PIPE number during requesting for writing of PRD_EN = 1

63.2.2.15 DMA Ch0 PRD Address Register3 (DMA_Ch0_PRD_ADR3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | PRD_ADR3 | H'0000_0000 | R/W | <p>Specify address of Physical Region Descriptor (PRD) Table that is located in memory space.</p> <p>Indicates PRD Table Address where is using in case this Register is read during transfer at the applicable PRD Table. Indicates the PRD Table Address where is used when next transfer is started in case reading this bit with stopping the transfer</p> <p>Note that lower 3 bits are all fixed to "0" (Reserved).</p> <p>It is unknown value when PRD Table is completed.</p> |

63.2.2.16 DMA Ch0 Control Register4 (DMA_Ch0_CON4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|----|----------|-----|-----|-----|-----|----|------------|-----|-----|----|--------------|---------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PIPE_DIR4 | — | — | PIPE_NO4 | | | | — | — | EXTPRD_NO4 | | — | — | EXTPRD_D_EN4 | PRD_EN4 | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R | R | R/W | R/W | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 | PIPE_DIR4 | B'0 | R/W | Specify Transfer Direction in AXI Master IF. 1: In Transfer (Memory → EPC) 0: Out Transfer (EPC → Memory) |
| 14 to 13 | — | All 0 | R | Reserved |
| 12 to 8 | PIPE_NO4 | All 0 | R/W | Specify PIPE number that this PRD Table is applied. Setting value between 1 and 5 is valid. (Note) |
| 7 to 6 | — | All 0 | R | Reserved |
| 5 to 4 | EXTPRD_NO4 | B'00 | R/W | Specify PRD Table number that is linked to this PRD Table. This is valid only when PRD_EN4 is set to 1. B'00: PRD_EN1 B'01: PRD_EN2 B'10: PRD_EN3 B'11: Setting prohibited |
| 3 to 2 | — | All 0 | R | Reserved |
| 1 | EXTPRD_EN4 | B'0 | R/W | Enable Link of PRD Table by EXTPRD_NO4. In case both PRD_EN4 and this bit are set to 1, setting of this PRD Table is enabled upon completion of transfer of PRD Table that was specified by PRD_NO4. This bit is cleared automatically when PRD Table is enabled. However, this is not cleared when transfer of linked PRD Table has been completed. Confirm that PRD_EN that was linked is still 1 after setting this bit. 1: PRD Table is linked 0: PRD Table is not linked |

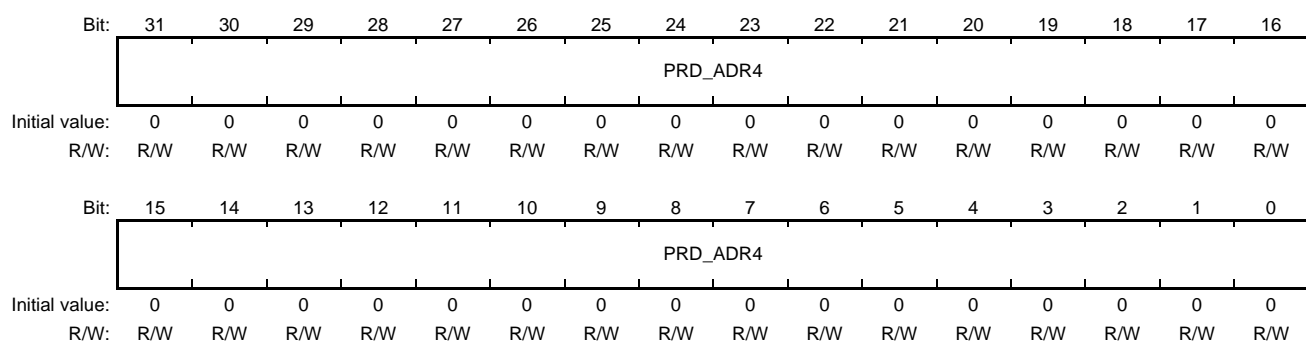
| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | PRD_EN4 | B'0 | R/W | <p>Allow DMA Transfer of PIPEn in AXI Master.</p> <p>Make sure to set this bit after preparing PRD Table and setting address where PRD Table for applicable PIPE is stored from DMA_Ch0_PRD_ADR4 Register.</p> <p>DMA Transfer is enabled as PRD Table in the address set in DMA_Ch0_PRD_ADR4 is valid when PRD_EN4 is 1 and EXTPRD_EN4 is 0. (Note)</p> <p>This is cleared to 0 automatically in case the process specified by PRD Table is completed or the process is aborted due to an Error.</p> <p>1: Allow DMA Transfer 0: Stop DMA Transfer</p> |

Note:

- Those registers shall not be set in each PRD_Table to be valid multiple PRD Tables for same PIPE number.
- PRD_EN is not set to 1 when PRDERR status is set due to recognizing a mistake of setting for PIPE number during requesting for writing of PRD_EN = 1

63.2.2.17 DMA Ch0 PRD Address Register4 (DMA_Ch0_PRD_ADR4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | PRD_ADR4 | H'0000_0000 | R/W | <p>Specify address of Physical Region Descriptor (PRD) Table that is located in memory space.</p> <p>Indicates PRD Table Address where is using in case this Register is read during transfer at the applicable PRD Table. Indicates the PRD Table Address where is used when next transfer is started in case reading this bit with stopping the transfer</p> <p>Note that lower 3 bits are all fixed to "0" (Reserved).</p> <p>It is unknown value when PRD Table is completed.</p> |

63.2.2.18 USB Common Control Register (USB_COM_CON)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----------------|----|----|----|----|----|------|-----------------------|-----------------------|-----------------|----|----|----|--------------|------------|----------------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | CONF | Pn_WD ATAIF_ NL | Pn_RD ATAIF_ NL | Pn_LST TR_PP | — | — | — | SPD_ MODE | EP0_ EN | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R/W | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | Device Address | | | | | | | | | | — | — | — | — | — | RX_DE TECTIO N | PIPE_ CLR |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | W | W | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------------|---------------|-----|---|
| 31 to 25 | — | All 0 | R | Reserved |
| 24 | CONF | B'0 | R/W | Bit to select whether Endpoints other than EP0 shall be enabled. 1: Enable Endpoints other than EP0 0: Disable Endpoints other than EP0 |
| 23 | Pn_ WDATAIF_ NL | B'0 | R/W | This bit is used to enable the write of ZLP to be transmitted in IN transfer through Data interface or on AXI DMA transfer. 1: The write of ZLP to be transmitted through Data Interface or on AXI DMA transfer is enabled 0: The write of ZLP to be transmitted through Data Interface or on AXI DMA transfer is disabled When ZLP is transferred through Data Interface or on AXI DMA transfer, set this bit to 1. Otherwise, set this bit to 0. |
| 22 | Pn_ RDATAIF_ NL | B'0 | R/W | This bit is used to enable the read of received ZLP in OUT transfer through Data Interface or on AXI DMA transfer. 1: The read of received ZLP through Data Interface or on AXI DMA transfer is enabled 0: The read of received ZLP through Data Interface or on AXI DMA transfer is disabled. When ZLP is transferred through Data Interface or on AXI DMA transfer, set this bit to 1. Otherwise, set this bit to 0. |
| 21 | Pn_LSTTR_P P | B'0 | R/W | This bit is used to disable the function to interpret the received packet with PP=0 as the last one of the transfer. 1: The received packet with PP=0 is not interpreted as the last one. 0: The received packet with PP=0 is interpreted as the last one. When ZLP is transferred through Data Interface or on AXI DMA transfer, set this bit to 1. Otherwise, set this bit to 0. |
| 20 to 18 | — | All 0 | R | Reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------------|---------------|-----|---|
| 17 | SPD_MODE | B'0 | R/W | This is a register to set operational mode of EPC. Output clock to SRAM is switched by setting in this register. Setting values are output to Pin, spd_mode as is. 1: USB2.0 (HS/FS) 0: USB3.0 (SS) |
| 16 | EP0_EN | B'0 | R | This is a bit to select whether PIPE0 (EP0) shall be enabled. 1: PIPE0 is enabled 0: PIPE0 is disabled |
| 15 | — | B'0 | R | Reserved |
| 14 to 8 | Device Address | All 0 | R | Value written in this bit is reflected as Device Address upon completion of STATUS Stage of SetAddress Request. Device Address value specified by the HOST shall be set before completion of STATUS Stage of Set Address Request. This is cleared automatically by WarmReset/HotReset/BusReset detection or PIPE_CLR or P0_CLR. |
| 7 to 2 | — | All 0 | R | Reserved |
| 1 | RX_DETECTION | B'0 | W | Write this to 1 before starting RX Detection of USB3.0 from the state USB3.0 is stopped and USB2.0 is in use. This bit is valid at only USB30_CON.B3_CONNECT=1 This is valid when PORTSC.PLS is Disable state |
| 0 | PIPE_CLR | B'0 | W | Write 1 in this bit when initialize (clear Sequence Number, DataPID and Buffer for transmit and receive) all PIPE (PIPE0 and PIPEn) all at once. It will set them in the same condition as all of P0_CLR and Pn_CLR provided for each PIPE are set. Initializing operation of PIPE varies depending on number of PIPE to be installed. This bit keeps indicating 1 until unitization is completed. Do not access other registers until this bit is cleared to 0 after 1 is written in this bit. |

63.2.2.19 USB20 Control Register (USB20_CON)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------|----|----|----|----|-----------|----------------|----------------|---------------------|---------------------|----------------------|-----------------|------------|----------------|---------------|----------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | B2_PUE | — | — | — | — | — | B2_RS UM_IN | B2_SU SPEND | B2_FO RCE_ HS | B2_FO RCE_ FS | B2_SU S_CLK MD | LPM_DI SABLE | LPM_ENABLE | B2_CO NNECT | B2_PH YRST | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | B2_TSTMOD | | — | — | — | — | — | — | — | — | B2_TST MOD_ EN |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R | R | R | R | W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 | B2_PUE | B'0 | R/W | This bit sets whether D+ Signal shall be pulled-up. The value written in this bit is output to B2_PUE Pin without any change. Note that B2_CONNECT needs to be set to 1 as well as this bit in order to connect. 1: Pull-up D+ Signal 0: Not to pull-up D+ Signal |
| 30 to 26 | — | All 0 | R | Reserved |
| 25 | B2_RSUM_IN | B'0 | R/W | This is a bit to set whether USB20 resume signal shall be transmitted or not. The value written in this bit is output to B2_RSUM_IN Pin and cleared automatically when Resume is started internally. This bit is used when Remote Wakeup function is supported. 1: resume signal is transmitted. 0: resume signal is not transmitted. |
| 24 | B2_SUSPEND | B'0 | R/W | This is a bit to set whether PLL of USB20 PHY shall be stopped. By setting this bit to 1 when B2_SPND_OUT that indicates USB20 PHY is Suspended is set to 1, PLL of USB20PH is stopped and power consumption is reduced. PLL can be resumed by setting this to 0 when PLL is stopped. This is cleared to 0 in case B2_RSUM_OUT or B2_PLL_WAKEIN is asserted. Make sure to activate PLL before operating a register that is related to USB2.0 operation other than this bit. 1: Stop PLL of PHY 0: Not to stop PLL of PHY. |
| 23 | B2_FORCE_HS | B'0 | R/W | Force HS mode operation when this bit is set. This allows the simulation time to reduce by removing Bus Reset). This bit should be set to 0 in normal operation 1: Force HS mode operation 0: Normal operation This bit is applicable for only simulation. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|---|
| 22 | B2_FORCE_FS | B'0 | R/W | Force FS mode operation when this bit is set. This allows the simulation time to reduce by removing Device Chirp). This bit should be set to 0 in normal operation 1: Force FS mode operation 0: Normal operation This bit is applicable for only simulation. |
| 21 | B2_SUS_CLKMD | B'0 | R/W | Control gating clock operation of CLK60_G/PG during Sleep/Suspend state in USB2.0. The setting value is present on the signal of B2_SUS_CLKMD. 1: No clock gating (CLK60_G/PG) 0: Clock gating (CLK60_G/PG) during Sleep/Suspend state |
| 20 | LPM_DISABLE | B'0 | R/W | Select Enabling/Disabling LPM 1: Disable 0: Enable No response is made to receiving Extended Token (LPM) from Host when this bit is set to 1 |
| 19 to 18 | LPM_ENABLE | B'00 | R/W | Selects response to LPM reception. B'00: Accepted Automatically transmit ACK/NYET According to status of Buffer of EPC. * NYET Transmission in case any of The conditions provided below is met. There is a Packet Data that is being written in transmission PIPE While CTRL transfer is processing. B'01: Force Accepted Perform ACK Transmission unconditionally Regardless of EPC Buffer status. B'10: Force Rejected Perform NYET Transmission unconditionally Regardless of EPC Buffer status. B'11: Force Unsupported Perform STALL Transmission unconditionally Regardless of EPC Buffer status. * Make sure to use this with Default (=B'00) setting in normal use. |
| 17 | B2_CONNECT | B'0 | R/W | Set this to 1 in case of USB2.0 connection. The value written in this bit is reflected to Output Pin B2_CONNECT without any changes. Set this bit to 1 after approx. 100 ms after detection of VBus in order to avoid chattering of VBus (it is same when it is cleared as 0). Note that bit31 B2_PUE must be set to 1 at the same time as this bit is set to 1 to connect. |
| 16 | B2_PHYRST | B'0 | R/W | This is a bit to reset USB20 PHY. The value of this bit is output to B2_PHY_RESET Pin. When use this bit, connect the pin so that USB20 PHY is reset with this bit set to 1. 1: Reset State 0: Normal state |
| 15 to 11 | — | All 0 | R | Reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|---|
| 10 to 8 | B2_TSTMOD | All 0 | R/W | This bit specifies Test type for USB2 Compliance TEST. Set value in this bit after Status Stage for SET_FEATURE TEST_MODE Request is successfully completed. The Core goes into a test mode when bit0: B2_TSTMOD_EN is set after (or at the same time as) setting this bit. B'000: Normal / B'001: Test_J / B'010: Test_K B'011: Test_SE0_NAK / B'100: Test_Packet B'101 - B'111: Reserved |
| 7 to 1 | — | All 0 | R | Reserved |
| 0 | B2_TSTMOD_EN | B'0 | W | When this bit is set to 1, the values set in bit10...8: B2_TSTMOD is validated at the point. |

63.2.2.20 USB30 Control Register (USB30_CON)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|---------------------|----|----|------------|---------|-----|-----|-------------|-----|----|----|------------|-------------|--------------|----------------|-------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Renesas Private bit | — | — | U3_POW_SEL | POW_SEL | | | B3_PLL WAKE | — | — | — | LPS_ENABLE | B3_CO NNECT | B3_PH YRST | | |
| Initial value: | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | POW_S EL_WEN | B3_HO TRST_CMP | B3_TP_ SEND |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | W | W | W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|--|
| 31 | Renesas Private bit | B'0 | R/W | Renesas Private bit This bit can be written only "0". DO NOT write "1". |
| 30 to 29 | — | All 0 | R | Reserved |
| 28 to 27 | U3_POW_SEL | B'10 | R/W | PHY POWERDOWN mode immediately after transition to U3 State by reception of LGO_U3. Select (PHY_POWERDOWN [1:0]). B'00: P0 B'01: Setting prohibited B'10: P2 B'11: Setting prohibited Use POW_SEL [2:0] in case changing POWERDOWN mode after transition to U3 State. |
| 26 to 24 | POW_SEL | B'000 | R/W | This is set to request change PHY POWERDOWN mode (PHY_POWERDOWN [1:0]). B'000: Transit from P0 to P2 in disconnected state (SS.Disabled) B'001: Transit from P0 to P3 in disconnected state (SS.Disabled) B'010: Transit from P2 to P0 in disconnected state (SS.Disabled) B'011: Setting prohibited B'100: Transit from P0 to P2 in U3 State B'101: Transit from P0 to P3 in U3 State B'110: Transit from P2 to P0 in U3 State B'111: Setting prohibited Transition of POWERDOWN mode other than the above is automatically done by H/W. The value in this bit is transmitted as a request when this bit is set bit2: POW_SEL_WEN. Success/Fail of the requested transition is informed by PS_SUCS/PS_FAIL interrupt. Make sure PLL is operating properly to operate with it when registers regarding to USB3.0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 23 | B3_ PLLWAKE | B'0 | R/W | <p>Transition from P3 to P0 will Wakeup PLL in case PHY POWERDOWN mode is in P3 (PLL Stop). Clear this bit from FW after PLL Wakeup is confirmed.</p> <p>1: PLL Wakeup is requested 0: Normal state</p> <p>Make sure PLL is operating properly to operate with it when registers regarding to USB3.0.</p> |
| 22 to 20 | — | All 0 | R | Reserved |
| 19 to 18 | LPS_ ENABLE | B'00 | R/W | <p>Selects allow/deny transition to U1/U2 State.</p> <p>B'00: Accepted</p> <p>Automatically transmit LGO/LAU/LXU depending on status of EPC Buffer.</p> <p>* Transition is denied in case any of the following conditions is met.</p> <ul style="list-style-type: none"> - ACK_TP is not received for DP transmitted from Transmission PIPE. - There is a Packet Data being written in Transmission PIPE. - PP of DP that was received at the end for Reception PIPE is 1. - While CTRL transfer is processing <p>B'01: Force Accepted</p> <p>Transmit LGO_U1/LGO_U2 unconditionally when U1/U2 Timeout that was set in PORTPMSC Register is reached.</p> <p>Reply LAU for reception of LGO forcibly except in case of LXU response in LINK layer.</p> <p>B'10: Force Rejected</p> <p>LGO_U1/LGO_U2 is not transmitted even when U1/U2 Timeout that was set in PORTPMSC Register is reached.</p> <p>Forcibly respond with LXU for LGO reception.</p> <p>B'11: Setting prohibited</p> <p>* Use this with Default (=B'00) setting.</p> |
| 17 | B3_ CONNECT | B'0 | R/W | <p>Set this to 1 in case of USB3.0 connection.</p> <p>The value written in this bit is reflected to Output Pin, B3_CONNECT without any changes.</p> <p>Set this bit to 1 after approx. 100 ms after detection of VBus in order to avoid chattering of VBus.</p> <p>Set this bit to 1. (it is same when it is cleared as 0)</p> |
| 16 | B3_PHYRST | B'0 | R/W | <p>This is a bit to reset logic area of USB30 PHY.</p> <p>The value of this bit is output to Pin, B3_PHY_RESET. When this bit is used, make sure to connect pin so that USB30 PHY is reset with this bit is set to 1.</p> <p>1: Reset State 0: Normal state</p> |
| 15 to 3 | — | All 0 | R | Reserved |
| 2 | POW_SEL_ WEN | B'0 | W | When this bit is set to 1, it is requested to request any transfer by value of POW_SEL [2:0]. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-------------------|---------------|-----|--|
| 1 | B3_HOTRST _CMP | B'0 | W | <p>This is a bit to inform completion of Hot Reset process to Link Controller.</p> <p>Make sure to write 1 to this bit after necessary process of reset is completed according to the process of HotReset where is described in 7.3.3 Hot Reset Process. This bit shall be set within 12ms from the start of HotReset by B3_HOTRST_STA. The completion of HotReset is told by TS1/TS2 exchanges in Link Controller because of set of this bit.</p> |
| 0 | B3_TP_ SEND | B'0 | W | <p>Write 1 in this bit to allow transmission to TP Data that was set in USBB3_TPDAT_0/1/2 Reg.</p> <p>It is not sent if WarmReset or HotReset is detected before sending TP_Data</p> |

63.2.2.21 USB Status Register (USB_STA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------|----|-----------|---------------|---------------|-------------|-------------|----|----|-----------------------|-----|-----|-------|-----|----------|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | Device Address Status | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | B2_LSTATE | — | B2_USBRST | B2_L1SPND_OUT | B2_L1RSUM_OUT | B2_SPND_OUT | B2_RSUM_OUT | — | — | — | — | — | SPEED | | VBUS_STA | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------------|---------------|-----|--|
| 31 to 23 | — | All 0 | R | Reserved |
| 22 to 16 | Device Address Status | H'00 | R/W | Indicates Device Address. Value in USB_COM_CON Reg. DEV_ADDR [6: 0] is reflected to this bit upon completion of Status Stage for Set Address Request. Value in this bit can be rewritten directly for Debug. This is cleared to 0 automatically in case of Warm/Hot Reset (USB30) or occurrence of Bus Reset (USB20). |
| 15 to 14 | B2_LSTATE | B'00 | R | Indicates status of Input SignalB2_LINESTATE (LineState Signal from UTMI-PHY). B'00: SE0 B'01: J-State B'10: K-State B'11: SE1 |
| 13 | — | B'0 | R | Reserved |
| 12 | B2_USBRST | B'0 | R | Indicates level of Input SignalB2_USB_RST. |
| 11 | B2_L1SPND_OUT | B'0 | R | Indicates level of Input SignalB2_L1_SPND_OUT. |
| 10 | B2_L1RSUM_OUT | B'0 | R | Indicates level of Input SignalB2_L1_RSUM_OUT. |
| 9 | B2_SPND_OUT | B'0 | R | Indicates level of Input SignalB2_SPND_OUT. |
| 8 | B2_RSUM_OUT | B'0 | R | Indicates level of Input SignalB2_RSUM_OUT. |
| 7 to 3 | — | All 0 | R | Reserved |
| 2 to 1 | SPEED | B'00 | R | Indicates Transfer Speed of USB. B'00: SS B'01: FS B'10: HS Make sure this after interruption of SPEED_STA |
| 0 | VBUS_STA | B'0 | R | Indicates level of Input Signal VBUS. |

63.2.2.22 USB20 Frame No. Register (USB20_FRAME)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----------|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | B2_FRAME | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 11 | — | All 0 | R | Reserved |
| 10 to 0 | B2_FRAME | All 0 | R | Indicates frame number received in (u)SOF during USB2.0 operation. |

63.2.2.23 DRD Common Control Register (DRD_CON)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|-------|---------------------|------------|------------|----------|----------|-----------|---------------------|-----------|----------|---------------------|----------------|---------------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | Renesas Private bit | — | — | — | PERI_CON | — | — | — | TISRDPRE | Renesas Private bit | B3_FORCE_INACT | Renesas Private bit | B3_RSP_WPR |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | DCDEN | IDPSRCEN | IDMSI_NKEN | IDPSI_NKEN | VDMSRCEN | VDPSRCEN | SHDN_MODE | Renesas Private bit | OTG_RPUEN | RPDEN | DP_RPU | DM_RPD | DP_RPD | VBOU |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | — |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|--|
| 31 to 29 | — | All 0 | R | Reserved |
| 28 | Renesas Private bit | B'0 | R | Renesas Private bit This bit can be written only "0". DO NOT write "1". |
| 27 to 25 | — | All 0 | R | Reserved |
| 24 | PERI_CON | B'0 | R/W | Peripheral/Host Switching bit Switching Core's role. 1: Select Peripheral function 0: Select Host function |
| 23 to 21 | — | All 0 | R | Reserved |
| 20 | TISRDPRE | B'0 | R/W | [Host Mode] Not used [Peripheral Mode] Control signal of the executing Receiver Detection in USB30PHY. |
| 19 | Renesas Private bit | B'0 | R/W | Renesas Private bit This bit can be written only "0". DO NOT write "1". |
| 18 | B3_FORCE_INACT | B'0 | R/W | Signal for enforced transition of SS.Inactive state. 1: Enable 0: Disable |
| 17 | Renesas Private bit | B'0 | R/W | Renesas Private bit This bit can be written only "0". DO NOT write "1". |
| 16 | B3_RSP_WPR | B'0 | R/W | When this bit is set to 1, a Warm Reset is issued for RSP. The bit is set to 1 and thus determined to be for Role Swap, after which Upstream/Downstream switching starts. HW clears this bit to 0 upon the detection of a Warm Reset. 1: Start Role Swap after Warm Reset |
| 15 to 14 | — | All 0 | R | Reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|------------------------|---------------|-----|--|
| 13 | DCDEN | B'0 | R/W | DCD Enable bit. [Host Mode] Not used [Peripheral Mode] Select bit whether use DCD or not 1: Operate DCD 0: NOT operate DCD |
| 12 | IDSRCEN | B'0 | R/W | [Host Mode] Not used [Peripheral Mode] Signal of D+ Source Voltage Enable. Controlled by SW. |
| 11 | IDMSINKEN | B'0 | R/W | [Host Mode] Not used [Peripheral Mode] Signal of D- Sink Current Enable. Controlled by SW. |
| 10 | IDPSINKEN | B'0 | R/W | [Host Mode] Not used [Peripheral Mode] Signal of D+ Sink Current Enable. Controlled by SW. |
| 9 | VDMSRCEN | B'0 | R/W | [Host Mode] Not used [Peripheral Mode] Signal of D- Source Voltage Enable. Controlled by SW. |
| 8 | VDPSRCEN | B'0 | R/W | [Host Mode] Not used [Peripheral Mode] Signal of D+ Source Voltage Enable. Controlled by SW. |
| 7 | SHDN_ MODE | B'1 | R/W | Shutdown mode bit. Software controls SHDN bit for Power IC such as Power IC type-2). 1: Normal Mode 0: Shutdown Mode |
| 6 | Renesas Private bit | B'0 | R/W | Renesas Private bit This bit can be written only "0". DO NOT write "1". |
| 5 | OTG_RPUEN | B'0 | R/W | OTG DP_RPU Enable bit. Controlled by Software. Set 1, DP_RPU bit is enabled 1: DP_RPU Enable Mode 0: Normal Mode |
| 4 | RPDEN | B'0 | R/W | DP/DM RPD Enable bit. Set 1, bit1 : DP_RPD and bit2 : DM_RPD are enabled. 1: DP/DMRPD Enable Mode 0: Normal Mode |
| 3 | DP_RPU | B'0 | R/W | DP Pull-up with OTG 1: Pull up enable (TermSelect = 1, XcrSelect = 1, OpMode = 0) 0: Pull up disable (TermSelect = 0, XcrSelect = 1, OpMode = 1) |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | DM_RPD | B'0 | R/W | <p>[Host Mode]</p> <p>Not used</p> <p>[Peripheral Mode]</p> <p>UTMI+ Dm Pull down Enable bit. Controlled by SW.</p> <p>1: Pull down enable</p> <p>0: Pull down disable</p> <p>SW controls this bit only when RPDEN = 1</p> |
| 1 | DP_RPD | B'0 | R/W | <p>[Host Mode]</p> <p>Not used</p> <p>[Peripheral Mode]</p> <p>UTMI+ Dp Pull down Enable bit. Controlled by SW.</p> <p>1: Pull down enable</p> <p>0: Pull down disable</p> <p>SW controls this bit only when RPDEN = 1</p> |
| 0 | VBOUT | Note | R/W | <p>VBUS Output Enable bit.</p> <p>[A-device Mode]</p> <p>When This bit is 1, DO_VBOUT is set to 1.</p> <p>[B-device Mode]</p> <p>Not used</p> <p>Note: Default value depends on signal "DI_PERICON_INIT".</p> <p>If DI_PERICON_INIT = 1,</p> <p>- VBOUT = 0</p> <p>If DI_PERICON_INIT = 0,</p> <p>- VBOUT = 1</p> |

63.2.2.24 DRD Common Control 2 Register (DRD_CON2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|---------------------|-----|----|----|----|----|----|----|---------------------|--------------|------------------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | VB_SET_LV | — | — | — | — | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | Renesas Private bit | — | — | — | — | — | — | — | Renesas Private bit | CONNEC T_SEL | SRP_DRP_SET_TIME | | | |
| Initial value: | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R/W | R/W | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|---|
| 31 to 22 | — | All 0 | R | Reserved |
| 21 to 20 | VB_SET_LV | B'10 | R/W | VBUS Set Level bit [Host Mode] Not used [Peripheral Mode] Sets Vbus detection Level. Controlled by Software. B'00: VBUS detection circuit is disabled. B'01: Detect Vbus when Vbus Level reaches VB_LV1 B'10: Detect Vbus when Vbus Level reaches VB_LV2 B'11: Detect Vbus when Vbus Level reaches VB_LV3 |
| 19 to 14 | — | All 0 | R | Reserved |
| 13 to 12 | Renesas Private bit | B'10 | R/W | Renesas Private bit This bit can be written only "B'10". DO NOT write "other value". |
| 11 to 6 | — | All 0 | R | Reserved |
| 5 | Renesas Private bit | B'0 | R/W | User shall not write 0 in this bit. |
| 4 | CONNECT_SEL | B'0 | R/W | [Host Mode] Not used [Peripheral Mode] Selector for USB20_COM.B2_CONNECT output signal to PHY. 1: 1clamp 0: Connect a signal which is connecting USB20_CON.B2_CONNECT |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------------|---------------|-----|--|
| 3 to 0 | SRP_DRP_ SET_TIME | B'0001 | R/W | SRP Data Line Pulsing Set Time bit. [A-Device Mode] Not used [B-Device Mode] SW sets duration of Pull Up of Data Line Pulsing. Default is 6ms. B'0000: 5ms B'0001: 6ms B'0010: 7ms : B'1101: 19ms B'1111: 20ms |

63.2.2.25 USB Interrupt Status 1 Register (USB_INT_STA_1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-----------------|-----------------|----------------|---------------|----------------|----------------|----------------|---------------|----|---------------|----------------|---------------|---------------|--------------|----------------|----------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | B3_PLL_WKUP_STA | B3_LU_PSUCS_STA | B3_POLLING_STA | B3_INACTV_STA | B3_DISABLE_STA | B3_VNDTST_STA | B3_U2INACT_STA | B3_SETLNK_STA | — | B3_LNKCNG_STA | B3_WRMRS_T_STA | B3_HOTRST_STA | B3_PSFAIL_STA | B3_PSUCS_STA | B3_U12_REQ_STA | B3_TP_SUCS_STA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | R | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | B2_LP_MRCV_STA | B2_USBRST_STA | B2_L1S_PND_STA | B2_L1R_SUM_STA | B2_SPND_STA | B2_RSUM_STA | — | — | — | — | — | — | SPEED_STA | VBUS_CNG_STA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | RW1C | RW1C | RW1C | RW1C | RW1C | RW1C | R | R | R | R | R | R | RW1C | RW1C |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-----------------|---------------|------|---|
| 31 | B3_PLLWKU_P_STA | B'0 | RW1C | Set 1 in this bit when clk125 is activated due to any one of PLL Wakeup (assertion of register set of B3_PLL_WAKEUP or assertion of B3_PLL_WAKEIN) or U3 Wakeup LFPS reception. |
| 30 | B3_LUPSUCS_STA | B'0 | RW1C | Set 1 in this bit when Link Up of USB3.0 is successfully completed. (Transition of LTSSM from Polling to U0). |
| 29 | B3_POLLING_STA | B'0 | RW1C | Set 1 in this bit when LTSSM transits to Polling State. |
| 28 | B3_INACTV_STA | B'0 | RW1C | Set this bit to 1 when LTSSM transits to Inactive State. |
| 27 | B3_DISABLE_STA | B'0 | RW1C | Set this bit to 1 when LTSSM transits to Disabled State. (only the case of transit when LTSSM transit to Disable State after setting B3_CONNECT = 1) |
| 26 | B3_VNDTST_STA | B'0 | RW1C | Set this bit to 1 when Vendor Device Test LMP is received. |
| 25 | B3_U2INACT_STA | B'0 | RW1C | Set this bit to 1 when U2 Inactivity Timeout LMP is received. |
| 24 | B3_SETLNK_STA | B'0 | RW1C | Set this bit to 1 when Set Link Function LMP is received. |
| 23 | — | B'0 | R | Reserved |
| 22 | B3_LNKCNG_STA | B'0 | RW1C | Set this bit to 1 when LTSSM transits as below. U0 -> U1 U0 -> U2 U0 -> U3 U3 -> Recovery -> U0 U2 -> Recovery -> U0 U1 -> Recovery -> U0 Refer to PLS[3:0] in PORTSC Register to check current state. |
| 21 | B3_WRMRS_T_STA | B'0 | RW1C | Set this bit to 1 when Warm Reset is received. |
| 20 | B3_HOTRST_STA | B'0 | RW1C | Set this bit to 1 when Hot Reset is received. |
| 19 | B3_PSFAIL_STA | B'0 | RW1C | Set this bit to 1 when Request to transit POWERDOWN mode by POW_SEL[2:0] in USB3.0_PWR_MNGCON Register is failed. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|----------|---|
| 18 | B3_PSSUCS_STA | B'0 | RW1 C | Set this bit to 1 when Request to transit POWERDOWN mode by POW_SEL[2:0] in USB3.0_PWR_MNGCON Register is succeeded. |
| 17 | B3_U12REQ_STA | B'0 | RW1 C | Set this bit to 1 when U1/U2 TIMER is expired in SSIF in case a request to transit to U1/U2 state is accepted. According to the value in LPS_ENABLE[1:0] in USB3.0_PWR_MNGCON Register, either reply with LAU/LXU when the request of U1/U2 transition is received or send LGO_U1/LGO_U2 when U1/U2_TIMER is expired (No FW treatment is needed when the transition request is received) |
| 16 | B3_TP_SUCS_STA | B'0 | RW1 C | This bit is set to 1 in case transmission of TP (Device Notification) is successfully completed by USB30_CON Register B3_TP_SEND. It is not set when the transmit is not occurred due to WarmReset or HotReset during requesting the transmit by B3_TP_SEND |
| 15 to 14 | — | All 0 | R | Reserved |
| 13 | B2_LPMRCV_STA | B'0 | RW1 C | This bit is set to 1 when ACK is replied after receiving normal LPM from USB20 Host. * At the point this status occurs, transition to L1 state is not completed. * Make sure to check for occurrence of B2_L1SPND_STA to process after transition to L1. Reply it automatically according to LPM_ENABLE[1:0] (No FW treatment is needed when the transition request is received). |
| 12 | B2_USBRST_STA | B'0 | RW1 C | This bit is set to 1 when Bus Reset is received from USB20 Host. This bit is set when a change of Input Pin B2_USB_RST from Low to High is detected. |
| 11 | B2_L1SPND_STA | B'0 | RW1 C | This bit is set to 1 when L1 Suspend Request is received from USB20 Host (when B2_L1_SPND_OUT is changed from 0 to 1). |
| 10 | B2_L1RSUM_STA | B'0 | RW1 C | This bit is set to 1 when clk60 is activated at the same time when requesting PLL Wakeup (either B2_SUSPEND = 0 or B2_PLL_WAKEIN assert) or L1 Resume Request is received from USB20 Host (when B2_L1_RSUM_OUT is changed from 0 to 1) during L1 (B2_L1SPND_OUT = 1). |
| 9 | B2_SPND_STA | B'0 | RW1 C | This bit is set to 1 when Suspend Request is received from USB20 Host (when B2_SPND_OUT is changed from 0 to 1). |
| 8 | B2_RSUM_STA | B'0 | RW1 C | This bit is set to 1 when clk60 is activated at the same time when PLL Wakeup is requested (B2_SUSPEND = 0 or B2_PLL_WAKEIN is asserted), Resume Request is received from USB20 Host (when B2_RSUM_OUT is changed from 0 to 1) during L2 (B2_SPND_OUT = 1) or L3 (B2_CONNECT = 0). |
| 7 to 2 | — | All 0 | R | Reserved |
| 1 | SPEED_STA | B'0 | RW1 C | This bit is set to 1 when there is any change in Speed Mode of USB. Refer to USB_STA Register bit2. 1: SPEED to check which speed mode it is running currently. SPEED bit may be taken same value between before and after the speed mode is changed. |
| 0 | VBUS_CNG_STA | B'0 | RW1 C | This bit is set to 1 when Level of Input Signal VBUS is changed. Read Level of VBUS from USB_STA Register. |

63.2.2.26 USB Interrupt Status 2 Register (USB_INT_STA_2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | PIPE5_IN T_STA | PIPE4_IN T_STA | PIPE3_IN T_STA | PIPE2_IN T_STA | PIPE1_IN T_STA | PIPE0_IN T_STA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------------|---------------|-----|---|
| 31 to 6 | — | All 0 | R | Reserved |
| 5 | PIPE5_INT_S TA | B'0 | R | Indicates an Interrupt related to PIPE occurred. Read PIPE5_INT_STA Reg. to check for factor. This bit is automatically cleared when all factors of Interrupts in PIPE5_INT_STA Reg. are cleared. |
| 4 | PIPE4_INT_S TA | B'0 | R | Indicates an Interrupt related to PIPE occurred. Read PIPE4_INT_STA Reg. to check for factor. This bit is automatically cleared when all factors of Interrupts in PIPE4_INT_STA Reg. are cleared. |
| 3 | PIPE3_INT_S TA | B'0 | R | Indicates an Interrupt related to PIPE occurred. Read PIPE3_INT_STA Reg. to check for factor. This bit is automatically cleared when all factors of Interrupts in PIPE3_INT_STA Reg. are cleared. |
| 2 | PIPE2_INT_S TA | B'0 | R | Indicates an Interrupt related to PIPE occurred. Read PIPE2_INT_STA Reg. to check for factor. This bit is automatically cleared when all factors of Interrupts in PIPE2_INT_STA Reg. are cleared. |
| 1 | PIPE1_INT_S TA | B'0 | R | Indicates an Interrupt related to PIPE occurred. Read PIPE1_INT_STA Reg. to check for factor. This bit is automatically cleared when all factors of Interrupts in PIPE1_INT_STA Reg. are cleared. |
| 0 | PIPE0_INT_S TA | B'0 | R | Indicates an Interrupt related to PIPE occurred. Read PIPE0_INT_STA Reg. to check for factor. This bit is automatically cleared when all factors of Interrupts in PIPE0_INT_STA Reg. are cleared. |

63.2.2.27 USB Interrupt Enable 1 Register (USB_INT_ENA_1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|------------------------|------------------------|------------------------|-----------------------|------------------------|-----------------------|------------------------|-----------------------|----|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | B3_PLL WKUP_ ENA | B3_LU PSUCS_ ENA | B3_PO LLING_ ENA | B3_INA CTV_ ENA | B3_DIS ABLE_ ENA | B3_VN DTST_ ENA | B3_U2I NACT_ ENA | B3_SE TLNK_ ENA | — | B3_LN KCNG_ ENA | B3_WR MRST_ ENA | B3_HO TRST_ ENA | B3_PS FAIL_ ENA | B3_PS SUCS_ ENA | B3_U12 REQ_ ENA | B3_TP SUCS_ ENA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | B2_LP MRCV_ ENA | B2_US BRST_ ENA | B2_L1S PND_EN | B2_L1R SUM_EN | B2_SP ND_EN | B2_RS UM_EN | — | — | — | — | — | — | SPEED _ENA | VBUS_C NG_ENA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|---|
| 31 | B3_PLLWKUP_P_ENA | B'0 | R/W | 1: B3_PLLWKUP_STA Interrupt is enabled. |
| 30 | B3_LUPSUCS_ENA | B'0 | R/W | 1: B3_LUPSUCS_STA Interrupt is enabled. |
| 29 | B3_POLLING_ENA | B'0 | R/W | 1: B3_POLLING_STA Interrupt is enabled. |
| 28 | B3_INACTV_ENA | B'0 | R/W | 1: B3_INACTV_STA Interrupt is enabled. |
| 27 | B3_DISABLE_ENA | B'0 | R/W | 1: B3_DISABLE_STA Interrupt is enabled. |
| 26 | B3_VNDTST_ENA | B'0 | R/W | 1: B3_VNDTST_STA Interrupt is enabled. |
| 25 | B3_U2INACT_ENA | B'0 | R/W | 1: B3_U2INACT_STA Interrupt is enabled. |
| 24 | B3_SETLNK_ENA | B'0 | R/W | 1: B3_SETLNK_STA Interrupt is enabled. |
| 23 | — | B'0 | R | Reserved |
| 22 | B3_LNKCNG_ENA | B'0 | R/W | 1: B3_LNKCNG_STA Interrupt is enabled. |
| 21 | B3_WRMRS_T_ENA | B'0 | R/W | 1: B3_WRMRS_STA Interrupt is enabled. |
| 20 | B3_HOTRST_ENA | B'0 | R/W | 1: B3_HOTRST_STA Interrupt is enabled. |
| 19 | B3_PSFAIL_ENA | B'0 | R/W | 1: B3_PSFAIL_STA Interrupt is enabled. |
| 18 | B3_PSSUCS_ENA | B'0 | R/W | 1: B3_PSSUCS_STA Interrupt is enabled. |
| 17 | B3_U12REQ_ENA | B'0 | R/W | 1: B3_U12REQ_STA Interrupt is enabled. |
| 16 | B3_TP_SUCS_ENA | B'0 | R/W | 1: B3_TP_SUCS_STA Interrupt is enabled. |
| 15 to 14 | — | All 0 | R | Reserved |
| 13 | B2_LPMRCV_ENA | B'0 | R/W | 1: B2_LPMRCV_STA Interrupt is enabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|---------------|---------------|-----|--|
| 12 | B2_USBRST_ENA | B'0 | R/W | 1: B2_USBRST_STA Interrupt is enabled. |
| 11 | B2_L1SPND_ENA | B'0 | R/W | 1: B2_L1SPND_STA Interrupt is enabled. |
| 10 | B2_L1RSUM_ENA | B'0 | R/W | 1: B2_L1RSUM_STA Interrupt is enabled. |
| 9 | B2_SPND_ENA | B'0 | R/W | 1: B2_SPND_STA Interrupt is enabled. |
| 8 | B2_RSUM_ENA | B'0 | R/W | 1: B2_RSUM_STA Interrupt is enabled. |
| 7 to 2 | — | All 0 | R | Reserved |
| 1 | SPEED_ENA | B'0 | R/W | 1: SPEED_STA Interrupt is enabled. |
| 0 | VBUS_CNG_ENA | B'0 | R/W | 1: VBUS_CNG_STA Interrupt is enabled. |

63.2.2.28 USB Interrupt Enable 2 Register (USB_INT_ENA_2)

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |

| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------------|----|----|----|----|----|----|---|---|---|---|----------------|----------------|----------------|----------------|----------------|----------------|
| | — | — | — | — | — | — | — | — | — | — | PIPE5_IN_T_ENA | PIPE4_IN_T_ENA | PIPE3_IN_T_ENA | PIPE2_IN_T_ENA | PIPE1_IN_T_ENA | PIPE0_IN_T_ENA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 31 to 6 | — | All 0 | R | Reserved |
| 5 | PIPE5_INT_ENA | B'0 | R | 1: PIPE5_INT_STA Interrupt is enabled. |
| 4 | PIPE4_INT_ENA | B'0 | R | 1: PIPE4_INT_STA Interrupt is enabled. |
| 3 | PIPE3_INT_ENA | B'0 | R | 1: PIPE3_INT_STA Interrupt is enabled. |
| 2 | PIPE2_INT_ENA | B'0 | R | 1: PIPE2_INT_STA Interrupt is enabled. |
| 1 | PIPE1_INT_ENA | B'0 | R | 1: PIPE1_INT_STA Interrupt is enabled. |
| 0 | PIPE0_INT_ENA | B'0 | R | 1: PIPE0_INT_STA Interrupt is enabled. |

63.2.2.29 Setup Data 0 Register (STUP_DAT_0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------|----|----|----|----|----|----|----|--------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | STUP_3 | | | | | | | | STUP_2 | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | STUP_1 | | | | | | | | STUP_0 | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--------------|
| 31 to 24 | STUP_3 | H'00 | R | SETUP Data 3 |
| 23 to 16 | STUP_2 | H'00 | R | SETUP Data 2 |
| 15 to 8 | STUP_1 | H'00 | R | SETUP Data 1 |
| 7 to 0 | STUP_0 | H'00 | R | SETUP Data 0 |

63.2.2.30 Setup Data 1 Register (STUP_DAT_1)

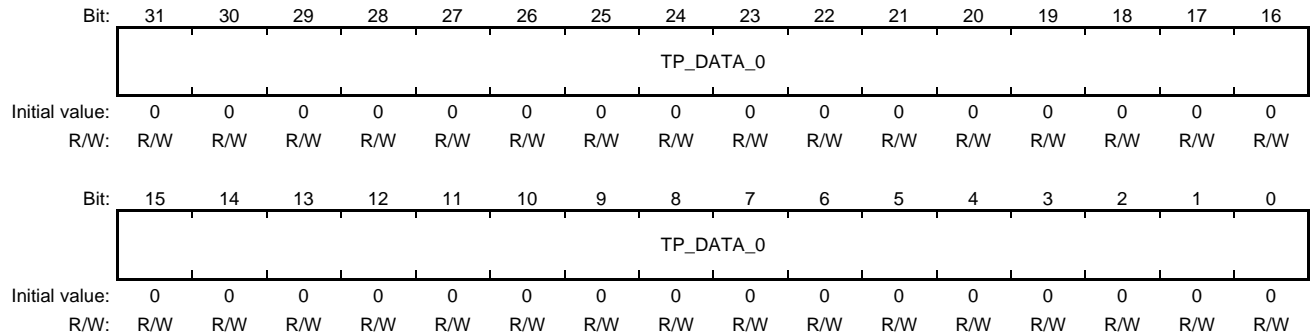
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|--------|----|----|----|----|----|----|----|--------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | STUP_7 | | | | | | | | STUP_6 | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | STUP_5 | | | | | | | | STUP_4 | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--------------|
| 31 to 24 | STUP_7 | H'00 | R | SETUP Data 7 |
| 23 to 16 | STUP_6 | H'00 | R | SETUP Data 6 |
| 15 to 8 | STUP_5 | H'00 | R | SETUP Data 5 |
| 7 to 0 | STUP_4 | H'00 | R | SETUP Data 4 |

63.2.2.31 USB30 TP_Send Data 0 Register (USB3_TPDAT_0)

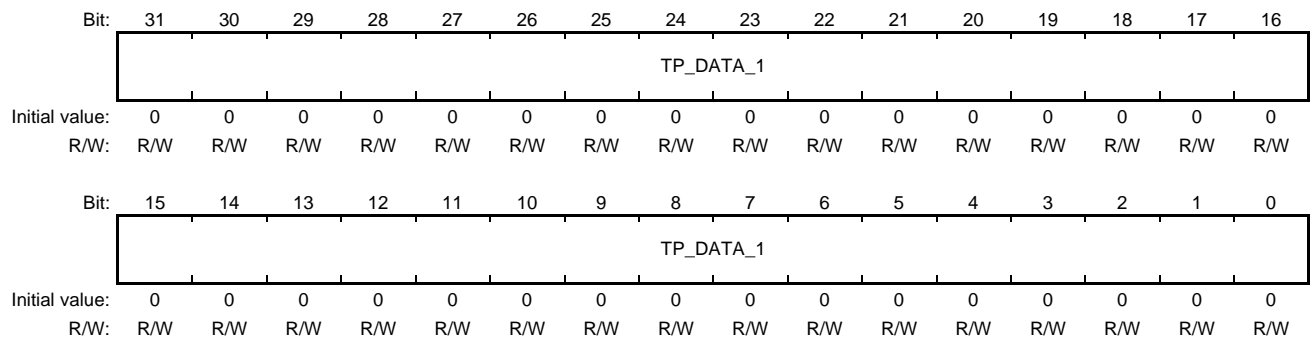
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|----------------|
| 31 to 0 | TP_DATA_0 | H'0000_0000 | R/W | TP Data DWORD0 |

63.2.2.32 USB30 TP_Send Data 1 Register (USB3_TPDAT_1)

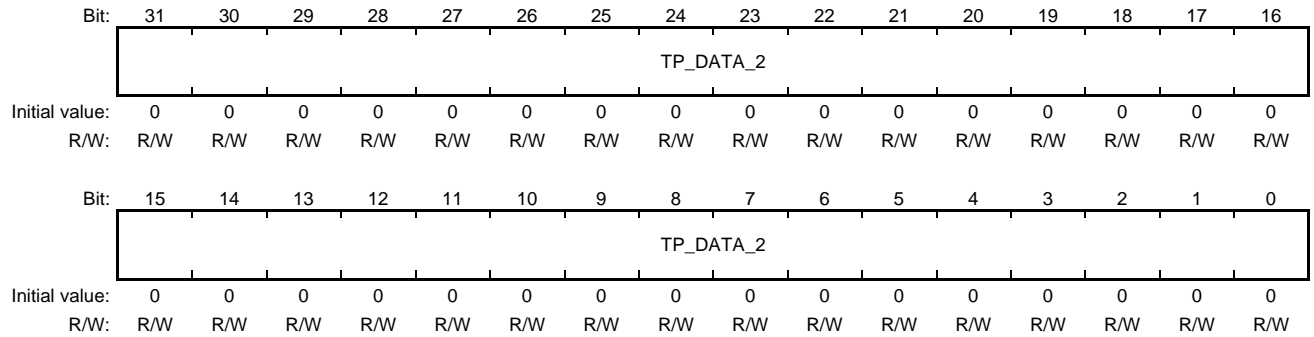
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|----------------|
| 31 to 0 | TP_DATA_1 | H'0000_0000 | R/W | TP Data DWORD1 |

63.2.2.33 USB30 TP_Send Data 2 Register (USB3_TPDAT_2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|----------------|
| 31 to 0 | TP_DATA_2 | H'0000_0000 | R/W | TP Data DWORD2 |

63.2.2.34 USB20 LPM Status Register (USB20_LPM_STA)

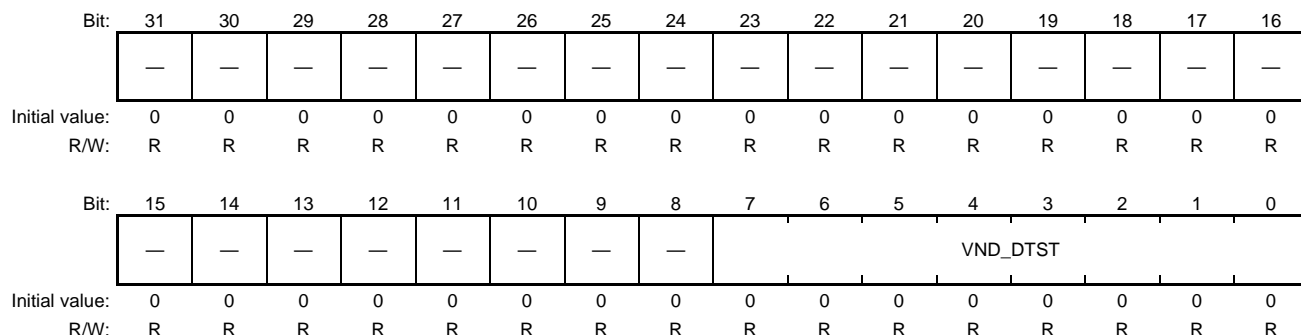
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----|------|----|----|--------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | BRW | HIRD | | | LINKST | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 9 | — | All 0 | R | Reserved |
| 8 | BRW | B'0 | R | Indicates value of bRemoteWake field in received LPM. 1: Remote Wakeup enable 0: Remote Wakeup disable Check the value when B2_L1SPND_STA Interrupt occurs. Register value is updated every time LPM is successfully received. |
| 7 to 4 | HIRD | B'0000 | R | Indicates value of HIRD field in received LPM. Indicates Host Initiated resume signal Drive Period from L1 State. (75us step) B'0000: 50us B'0001: 125us B'0010: 200us ... B'1111: 1200us Check the value when B2_L1SPND_STA Interrupt occurs. Register value is updated every time LPM is successfully received. |
| 3 to 0 | LINKST | B'0000 | R | Indicates value of bLinkState field in received LPM. Indicates Link State (Lx) for LPM Token shall transit to after ACK reply. B'0001: L1 (Sleep) other: Reserved Check the value when B2_L1SPND_STA Interrupt occurs. Register value is updated every time LPM is successfully received. |

63.2.2.35 USB30 Vendor Device Test Register (USB30_VND_DEV)

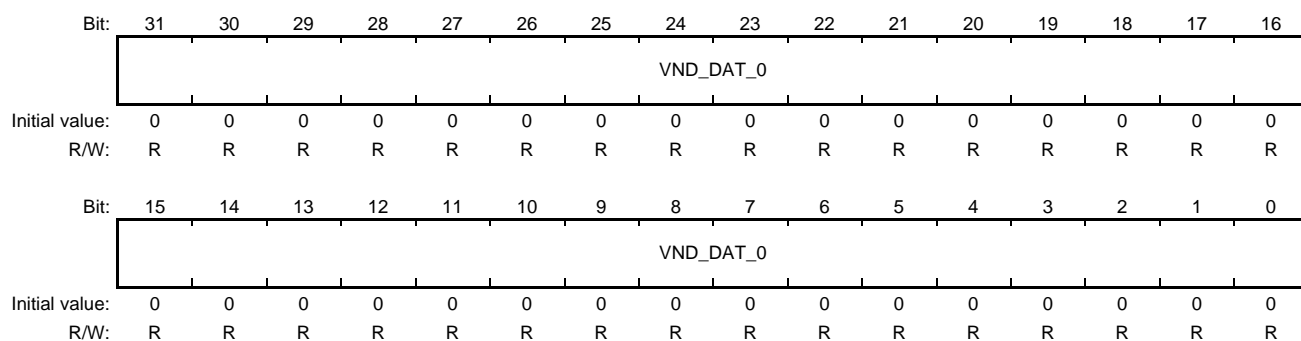
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved |
| 7 to 0 | VND_DTST | H'00 | R | Indicates value of Vendor Device Test of the received Vendor Device Test LMP. This is updated every time Vendor Device Test LMP is received. Make sure to perform required processes according to the value of this bit when VND_DTST_STA Interrupt occurs. |

63.2.2.36 USB30 Vendor Defined Data 0 Register (USB30_VND_DAT_0)

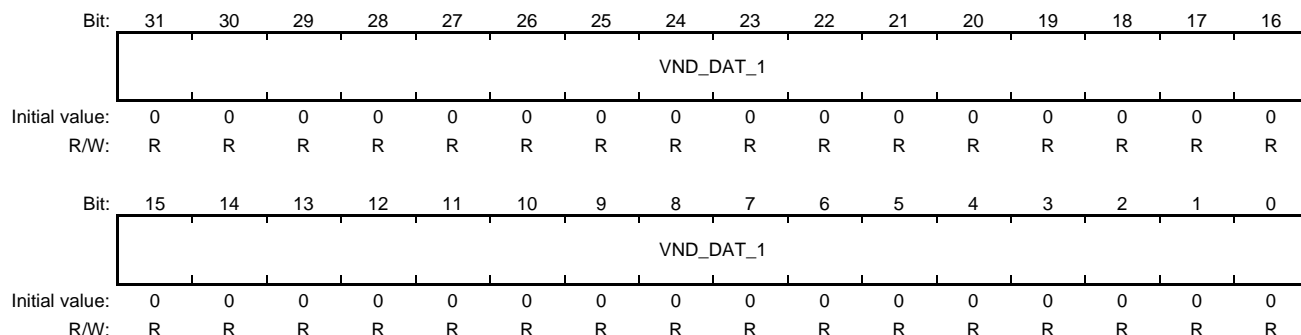
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 0 | VND_DAT_0 | All 0 | R | Indicates value of Vendor Defined Data (04H-07H) in the received Vendor Device Test LMP. This is updated every time Vendor Device Test LMP is received. Make sure to perform required processes according to the value of this bit when VND_DTST_STA Interrupt occurs. |

63.2.2.37 USB30 Vendor Defined Data 1 Register (USB30_VND_DAT_1)

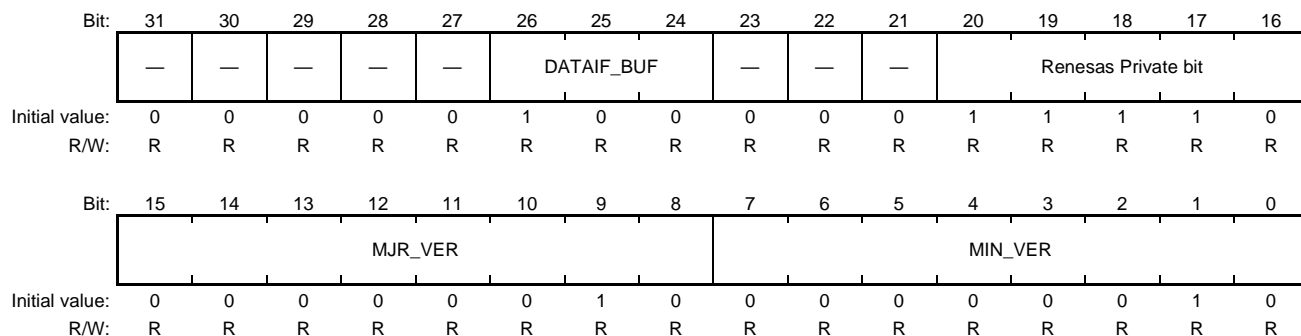
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 0 | VND_DAT_1 | H'0000_0000 | R | Indicates value of Vendor Defined Data (08H-0BH) in the received Vendor Device Test LMP. This is updated everytime Vendor Device Test LMP is received. Make sure to perform required processes according to the value of this bit when VND_DTST_STA Interrupt occurs. |

63.2.2.38 EPC Version Register (USB_VER)

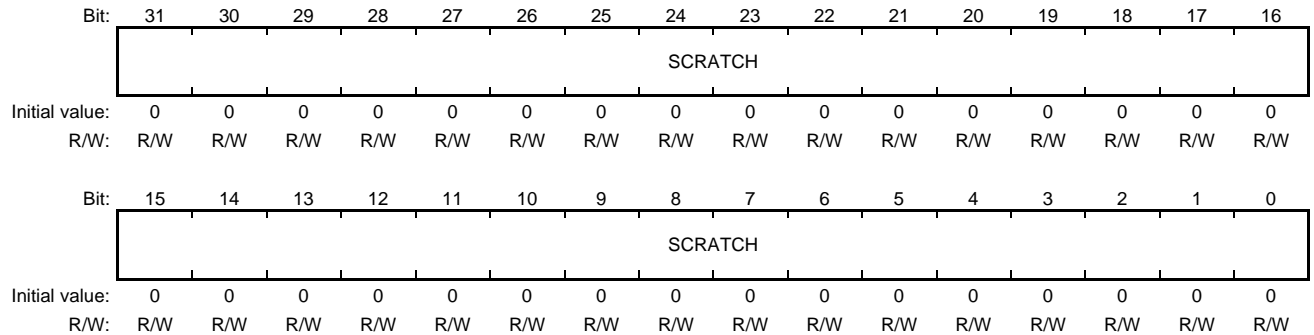
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|--|
| 31 to 27 | — | All 0 | R | Reserved |
| 26 to 24 | DATAIF_BUF | H'4 | R | This field shows the number of asynchronous buffers at Data Interface. |
| 23 to 21 | — | All 0 | R | Reserved |
| 20 to 16 | Renesas Private bit | H'1E | R | Renesas Private bit |
| 15 to 8 | MJR_VER | H'02 | R | Indicates Major Version of EPC. |
| 7 to 0 | MIN_VER | H'02 | R | Indicates Minor Version of EPC. |

63.2.2.39 Scratch Pad Register (USB_SCRATCH)

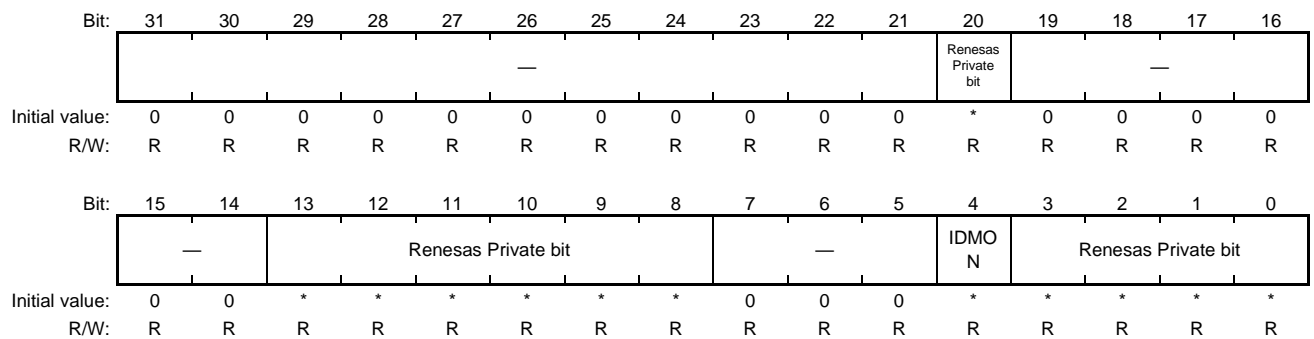
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | SCRATCH | H'0000_0000 | R/W | This is a General Purpose Register. The value of this register is stored into Register-RAM |

63.2.2.40 USB OTG Status Register (USB_OTG_STA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

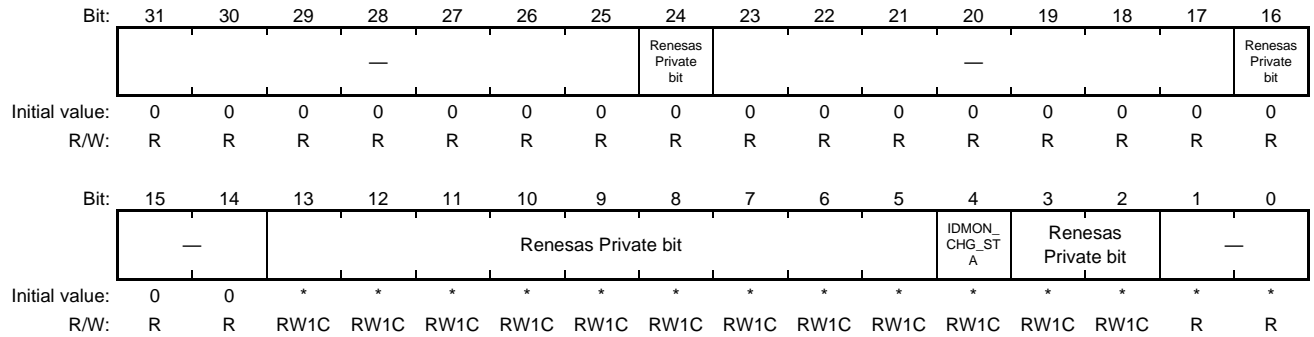


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|--------------------------|
| 31 to 21 | — | All 0 | R | Reserved |
| 20 | Renesas Private bit | * | R | Renesas Private bit |
| 19 to 14 | — | All 0 | R | Reserved |
| 13 to 8 | Renesas Private bit | * | R | Renesas Private bit |
| 7 to 5 | — | All 0 | R | Reserved |
| 4 | IDMON | * | R | Status bit of ID signal. |
| 3 to 0 | Renesas Private bit | * | R | Renesas Private bit |

Note: * Values for these bits are undefined.

63.2.2.41 USB OTG Interrupt Status Register (USB_OTG_INT_STA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

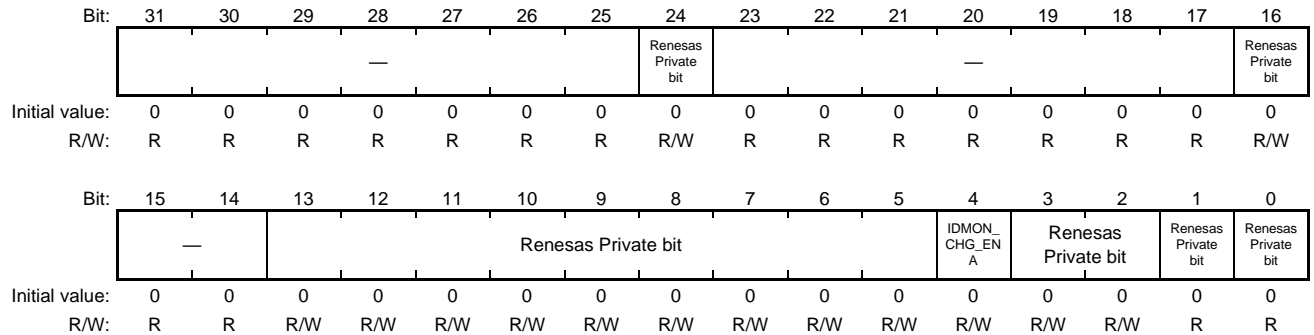


| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|------|--|
| 31 to 25 | — | All 0 | R | Reserved |
| 24 | Renesas Private bit | B'0 | R | Renesas Private bit |
| 23 to 17 | — | All 0 | R | Reserved |
| 16 | Renesas Private bit | B'0 | R | Renesas Private bit |
| 15 to 14 | — | All 0 | R | Reserved |
| 13 to 5 | Renesas Private bit | * | RW1C | Renesas Private bit This bit can be written only "0". DO NOT write "1". |
| 4 | IDMON_CHG_STA | * | RW1C | ID pin Change Status bit. If bit is set 1, ID pin has changed. |
| 3 to 2 | Renesas Private bit | * | RW1C | Renesas Private bit This bit can be written only "0". DO NOT write "1". |
| 1 to 0 | — | * | R | Reserved |

Note: * Values for these bits are undefined.

63.2.2.42 USB OTG Interrupt Enable Register (USB_OTG_INT_ENA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------------|---------------|-----|--|
| 31 to 25 | — | All 0 | R | Reserved |
| 24 | Renesas Private bit | B'0 | R/W | Renesas Private bit This bit can be written only "0". DO NOT write "1". |
| 23 to 17 | — | All 0 | R | Reserved |
| 16 | Renesas Private bit | B'0 | R/W | Renesas Private bit This bit can be written only "0". DO NOT write "1". |
| 15 to 14 | — | All 0 | R | Reserved |
| 13 to 5 | Renesas Private bit | B'0_0000_0000 | R/W | Renesas Private bit These bits can be written only "0". DO NOT write "1". |
| 4 | IDMON_CHG_ENA | B'0 | R/W | IDMON_CHG_STA Interrupt is valid. |
| 3 to 2 | Renesas Private bit | All 0 | R/W | Renesas Private bit These bits can be written only "0". DO NOT write "1". |
| 1 | Renesas Private bit | B'0 | R | Renesas Private bit |
| 0 | Renesas Private bit | B'0 | R | Renesas Private bit This bit can be written only "0". DO NOT write "1". |

63.2.2.43 PIPE0 Mode Setting Register (P0_MOD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|--------|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | P0_DIR | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 7 | — | All 0 | R | Reserved |
| 6 | P0_DIR | B'0 | R | This bit sets transfer direction of DATA Stage. Set this bit according to transfer direction after decoding Request Code of the received SETUP Data. 1: IN (Control Read) 0: OUT (Control Write) Response at Data Stage is set by P0_IN_RES/P0_OT_RES. |
| 5 to 0 | — | All 0 | R | Reserved |

63.2.2.44 PIPE0 Control Register (P0_CON)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|-----------|------------|-------------|----------------|----|----|----|----|----|----|-------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | P0_ST_RES | P0_OT_RES | — | — | — | — | — | — | — | — | P0_IN_RES | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | P0_BYTE_EN | P0_SE ND | P0_RE S_WEN | — | — | — | — | — | — | P0_BC LR | P0_CL R |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | W | W | W | W | R | R | R | R | R | W | W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 31 to 28 | — | All 0 | R | Reserved |
| 27 to 26 | P0_ST_RES | B'00 | R/W | This bit controls response for PIPE0 STATUS reception. B'00: Force NRDY/NAK Response B'01: Normal Response [In case of SS] ACK Response for STATUS_TP reception. [In case of HS/FS] When Control Write/NoData, Transmit Null Data for IN Token of Status Stage. When Control Read Respond ACK for OUT Token of Status Stage. B'10: Force STALL Response B'11: Setting prohibited This bit is updated to B'00 automatically upon reception of SETUP. |
| 25 to 24 | P0_OT_RES | B'00 | R/W | This bit controls response for PIPE0 OUT Data reception. B'00: Force NRDY/NAK Response B'01: Normal Response (Data can be accepted depending on Buffer status) B'10: Force STALL Response B'11: Setting prohibited This bit is updated to B'00 automatically when SETUP Data reception. In case buffer status is "acceptable" in SS, ERDY is transmitted when this bit is changed from B'00 to B'01. Set this bit to B'10 during Data Stage of Control Read Request just in case Transfer Request of different direction generated from the Host. |
| 23 to 18 | — | All 0 | R | Reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 17 to 16 | P0_IN_RES | B'00 | R/W | <p>This bit controls response for PIPE0 IN Data Transmission Request.</p> <p>B'00: Force NRDY/NAK Response</p> <p>B'01: Normal Response (Data can be transmitted depending on Buffer status)</p> <p>B'10: Force STALL Response</p> <p>B'11: Setting prohibited</p> <p>This bit is updated to B'00 automatically when SETUP Data reception.</p> <p>In case buffer status is "transmittable" in SS, ERDY is transmitted when this bit is changed from B'00 to B'01.</p> <p>Set this bit to B'10 during Data Stage of Control Write Request just in case Transfer Request of different direction generated from the Host.</p> |
| 15 to 11 | — | All 0 | R | Reserved |
| 10 to 9 | P0_BYTE_EN | B'00 | W | <p>Specifies number of valid byte of the data written in P0_WRITE Reg.</p> <p>This bit is set at the same time as writing to P0_SEND.</p> <p>B'00: 4 Bytes is valid.</p> <p>B'01: Only 1 Byte is valid. (Only DATA[7:0] is valid.)</p> <p>B'10: Only 2 Byte is valid. (Only DATA[15:0] is valid.)</p> <p>B'11: Only 3 Byte is valid. (Only DATA[23:0] is valid.)</p> |
| 8 | P0_SEND | B'0 | W | Write 1 in this bit to allow to transmit the data written in P0_WRITE Reg. |
| 7 | P0_RES_WEN | B'0 | W | Set this bit to 1 when writing to change the value in P0_OT_RES/P0_IN_RES/P0_ST_RES. |
| 6 to 2 | — | All 0 | R | Reserved |
| 1 | P0_BCLR | B'0 | W | <p>Write 1 in this bit to clear Buffer Data of PIPE0.</p> <p>Set P0_IN_RES/P0_OT_RES to B'00 (Force NRDY/NAK Reply Mode) prior to clear the Buffer and confirm that P0_ACKSTS = 0 (no data is begin transferred) and stopping write operation to data buffer and then, set this bit.</p> <p>The interrupt status and operation mode are not cleared by setting this bit.</p> <p>This bit keeps indicating 1 until initialization is completed. Make sure not to access other registers until this bit is cleared to 0 after writing 1.</p> |
| 0 | P0_CLR | B'0 | W | <p>Write 1 in this bit to initialize PIPE0 (clear Sequence Number/DataPID and transmit/receive Buffer).</p> <p>P0_IN_RES/P0_OT_RES is set to B'00 (forced NRDY/NAK response mode) and confirm that P0_ACKSTS = 0 (there is no data during transfer) and stopping writing to the DATA buffer, and then this bit is set before the initialization of PIPE0 is processing.</p> <p>Interrupt Status and Operational mode setting are not cleared by setting a value in this bit.</p> <p>This bit keeps indicating 1 until initialization is completed. Make sure not to access other registers until this bit is cleared to 0 after writing 1.</p> |

63.2.2.45 PIPE0 Status Register (P0_STA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|------------|-----------|-----------|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | P0_FLOWSTS | P0_ACKSTS | P0_SNDSTS | P0_BUFSTS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 to 4 | — | All 0 | R | Reserved |
| 3 | P0_FLOWSTS | B'0 | R | Indicates FLOW CONTROL State in PIPE0. 1: FLOW CONTROL State 0: Normal state |
| 2 | P0_ACKSTS | B'0 | R | Indicates existence of Data that is currently being transferred. -In case transfer direction is OUT (P0_DIR = 0): 1 is displayed if there is any reception data that has not received ACK for. -In case transfer direction is IN (P0_DIR = 1): 1 is displayed if there is any transmission data that has not received ACK for. |
| 1 | P0_SNDSTS | B'0 | R | 1 is displayed if there is any Data that has not transmitted in Transmission Buffer (FIFO). This bit is cleared when written data is transmitted and ACK is received for the Data. This is invalid in case transfer direction is OUT (P0_DIR = 0). |
| 0 | P0_BUFSTS | B'0 | R | Indicates status of P0_READ/P0_WRITE Reg. 1: Read/Write of Data is enabled. 0: Read/Write of Data is disabled. -In case transfer direction is OUT (P0_DIR = 0): 1 is displayed when there is reception Data that can be read from P0_READ Reg. -In case transfer direction is IN (P0_DIR = 1): 1 is displayed when there is transmission Data that can be written to P0_READ Reg. |

Note: Shall not access to those bits during Read/Write process until completion of transferring one packet size through Pn_READ/Pn_WRITE. Contact to Renesas for more detail.

63.2.2.46 PIPE0 Interrupt Status Register (P0_INT_STA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|------------------|-----------------|-----------------|----|----|----|------------------|------------------|------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | P0_STSE D_STA | P0_STSS T_STA | P0_SET UP_STA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | RW1C | RW1C | RW1C |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | P0_RCV NL_STA | P0_ERD Y_STA | P0_FLO W_STA | — | — | — | P0_STAL L_STA | P0_NRD Y_STA | P0_BFR DY_STA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | RW1C | RW1C | RW1C | R | R | R | RW1C | RW1C | RW1C |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|----------|--|
| 31 to 19 | — | All 0 | R | Reserved |
| 18 | P0_STSED_ STA | B'0 | RW1 C | This bit is set to 1 when STATUS Stage is completed successfully. |
| 17 | P0_STSST_ STA | B'0 | RW1 C | This bit is set to 1 upon start of STATUS Stage. |
| 16 | P0_SETUP_ STA | B'0 | RW1 C | This bit is set to 1 when valid SETUP Data is received. Make sure to start processing a request after clearing this bit. |
| 15 to 9 | — | All 0 | R | Reserved |
| 8 | P0_RCVNL_ STA | B'0 | RW1 C | This bit is set to 1 when Null Data that was received in PIPE0 is readable. Null Data is automatically discarded when P0 Length Register is read. |
| 7 | P0_ERDY_ STA | B'0 | RW1 C | This bit is set to 1 when transmission condition of ERDY for PIPE0 is met. This is invalid in case of HS/FS. In case P0_IN_RES/P0_OT_RES/P0_ST_RES is set to any except Forced NRDY/NAK Response (B'00), ERDY is transmitted to Host. |
| 6 | P0_FLOW_ STA | B'0 | RW1 C | This bit is set to 1 when PIPE0 is transitioned to Flow Control State. This is invalid in case of HS/FS. -In case transfer direction is IN (P0_DIR = 1): This bit is set to 1 upon transmission of NRDY_TP. -In case transfer direction is OUT (P0_DIR = 0): This bit is set to 1 upon transmission of NRDY_TP or transmission of ACK_TP with NumP = 0. |
| 5 to 3 | — | All 0 | R | Reserved |
| 2 | P0_STALL_ STA | B'0 | RW1 C | This bit is set to 1 when STALL is transmitted in response to requests of IN/OUT/STATUS to PIPE0. |
| 1 | P0_NRDY_ STA | B'0 | RW1 C | This bit is set to 1 when NRDY/NAK is transmitted in response to requests of IN/OUT/STATUS to PIPE0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|------------------|---------------|----------|--|
| 0 | P0_BFRDY_ STA | B'0 | RW1 C | <p>In case transfer direction is IN (P0_DIR = 1):</p> <p>This bit is set to 1 when the next data can be written in P0_WRITE Reg.</p> <p>This bit is not set for interruption before writing at the first packet when the buffer is empty, where is like Reset.</p> <p>-In case transfer direction is OUT (P0_DIR = 0):</p> <p>This bit is set to 1 when the next data can be read from 0_READ Reg.</p> |

63.2.2.47 PIPE0 Interrupt Enable Register (P0_INT_ENA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|------------------|-----------------|-----------------|----|----|----|------------------|------------------|------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | P0_STSE D_ENA | P0_STSS T_ENA | P0_SET UP_ENA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | P0_RCV NL_ENA | P0_ERD Y_ENA | P0_FLO W_ENA | — | — | — | P0_STAL L_ENA | P0_NRD Y_ENA | P0_BFR DY_ENA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|-----|--------------------------------|
| 31 to 19 | — | All 0 | R | Reserved |
| 18 | P0_STSED_ ENA | B'0 | R/W | 1: P0_STSED_STAInterrupt Valid |
| 17 | P0_STSST_ ENA | B'0 | R/W | 1: P0_STSST_STAInterrupt Valid |
| 16 | P0_SETUP_ ENA | B'0 | R/W | 1: P0_SETUP_STAInterrupt Valid |
| 15 to 9 | — | All 0 | R | Reserved |
| 8 | P0_RCVNL_ ENA | B'0 | R/W | 1: P0_RCVNL_STAInterrupt Valid |
| 7 | P0_ERDY_ ENA | B'0 | R/W | 1: P0_ERDY_STAInterrupt Valid |
| 6 | P0_FLOW_ ENA | B'0 | R/W | 1: P0_FLOW_STAInterrupt Valid |
| 5 to 3 | — | All 0 | R | Reserved |
| 2 | P0_STALL_ ENA | B'0 | R/W | 1: P0_STALL_STAInterrupt Valid |
| 1 | P0_NRDY_ ENA | B'0 | R/W | 1: P0_NRDY_STAInterrupt Valid |
| 0 | P0_BFRDY_ ENA | B'0 | R/W | 1: P0_BFRDY_STAInterrupt Valid |

63.2.2.48 UTMI Plus Vendor Control Interface Set Register (VENDOR_CON_SET)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----------|-----|-----|-----|-----|-----|-----|-------|-----|---|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | | | |
| | — | — | — | — | — | — | — | — | VWDATA | | | | | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | |
| | — | — | — | — | — | — | — | — | VCONTROL | | | | — | — | — | VCWEN | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | W | | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 24 | — | All 0 | R | Reserved |
| 23 to 16 | VWDATA | H'00 | R/W | Signal for Vendor Write Data. It is write data signal to Vendor Control Register Write field. |
| 15 to 8 | — | All 0 | R | Reserved |
| 7 to 4 | VCONTROL | H'0 | R/W | Signal for Vendor Control. It is able to select Vendor Control Register that user wants to use. |
| 3 to 1 | — | All 0 | R | Reserved |
| 0 | VCWEN | B'0 | W | Signal for enabling write VWDATA for Vendor Control Address. If this bit is set to 1, it is enabled to write the value of VWDATA for Vendor Control Address. |

63.2.2.49 UTMI Plus Vendor Control Interface Status Register (VENDOR_CON_STA)

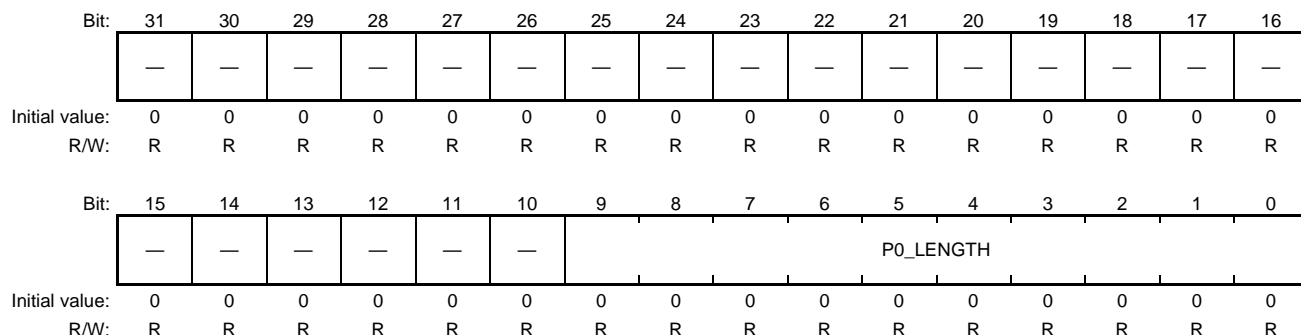
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----------|----|----|----|----|----|----|----|---|---|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | |
| | — | — | — | — | — | — | — | — | VCONTROL | | | | | | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 8 | — | All 0 | R | Reserved |
| 7 to 0 | VSTATUS | H'00 | R | Signal for Vendor Status. It is read data signal from Vendor Control Register Read field. |

63.2.2.50 PIPE0 Length Register (P0_LNG)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



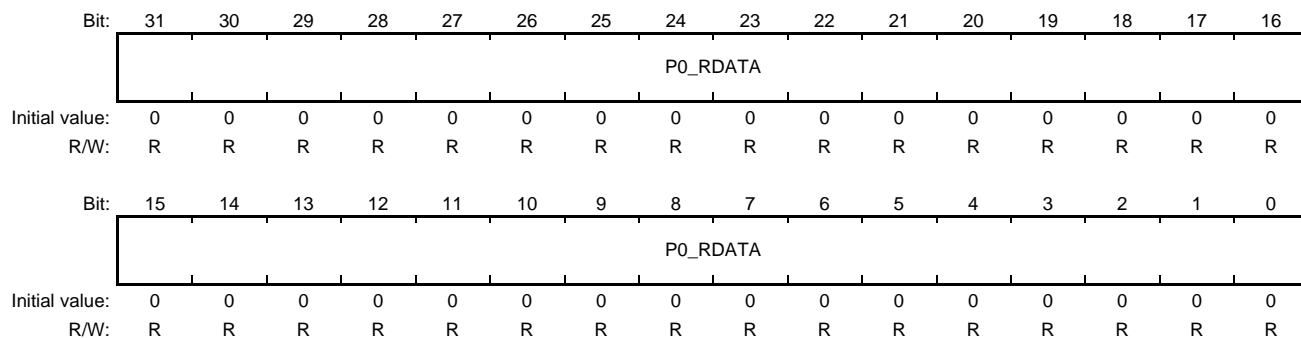
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 10 | — | All 0 | R | Reserved |
| 9 to 0 | P0_LENGTH | B'00_0000_000 | R | <p>This field shows the number of bytes in received packet which is currently read from P0_READ register.</p> <p>The value is decremented every time P0_READ register is read. Since P0_READ register allows only 32-bit access, the value is decremented by 4.</p> <p>Since P0_READ register is controlled in a unit of a packet, the length of received packet is shown in this register when the first 4 bytes of it are shown in P0_READ register and have not been read yet.</p> <p>This register is valid only for OUT direction.</p> |

Note: Once the access to one of P0_READ, Pn_READ, P0_WRITE and Pn_WRITE registers is started in order to read or write a packet, it is prohibited to access to other Px_yyy register before whole of a packet has been read or written with the register which is currently used. The access to these registers is controlled in the unit of packet, so the targeted register can be changed at the point that the read or write of the current packet has been completed and the access for the next packet is not started yet. See section 63.9.2 for details.

63.2.2.51 PIPE0 Read Register (P0_READ)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

If there is no data in buffer, H'0000_0000 is returned. Only 32-bit access is allowed to this register.

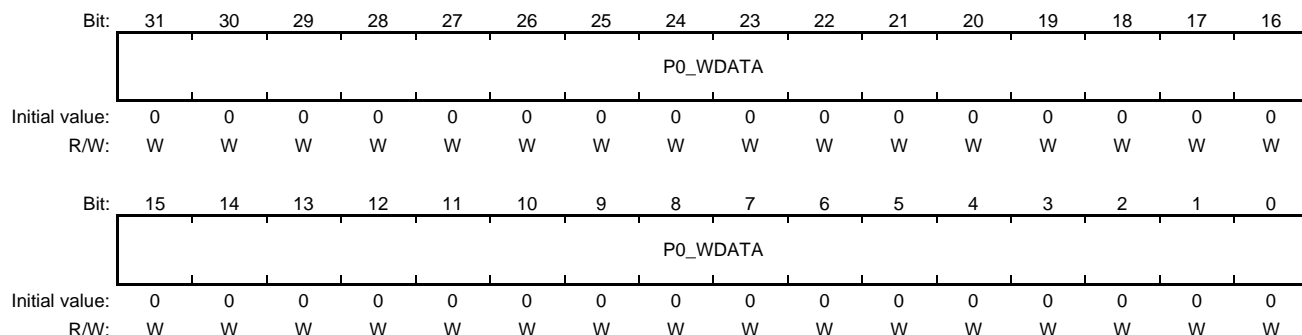


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | P0_RDATA | H'0000_0000 | R | <p>Read data from PIPE0.</p> <p>This register is controlled in a unit of a packet and can't be read across the packet.</p> <p>That is, this register is updated in a unit of a packet. The current packet is never concatenated with the next packet even when the length of the current packet is not multiples of 4 and the last word of it is less than 4 bytes. In that case, the first 4 bytes of the next packet are shown in this register after the last word of the previous packet has been read.</p> <p>If the length of the current packet is not multiples of 4 and the last word of it is less than 4 bytes, the last word has one or more invalid bytes. In order to know how many valid bytes the last word has, see P0_LENGTH[7:0] before the read of the packet and calculate from it. The valid bytes are aligned from P0_RDATA[7:0].</p> <p>This register is valid only for OUT direction.</p> |

Note: Once the access to one of P0_READ, Pn_READ, P0_WRITE and Pn_WRITE registers is started in order to read or write a packet, it is prohibited to access to other Px_yyy register before whole of a packet has been read or written with the register which is currently used. The access to these registers is controlled in the unit of packet, so the targeted register can be changed at the point that the read or write of the current packet has been completed and the access for the next packet is not started yet. See section 63.9.2 for details.

63.2.2.52 PIPE0 Write Register (P0_WRITE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

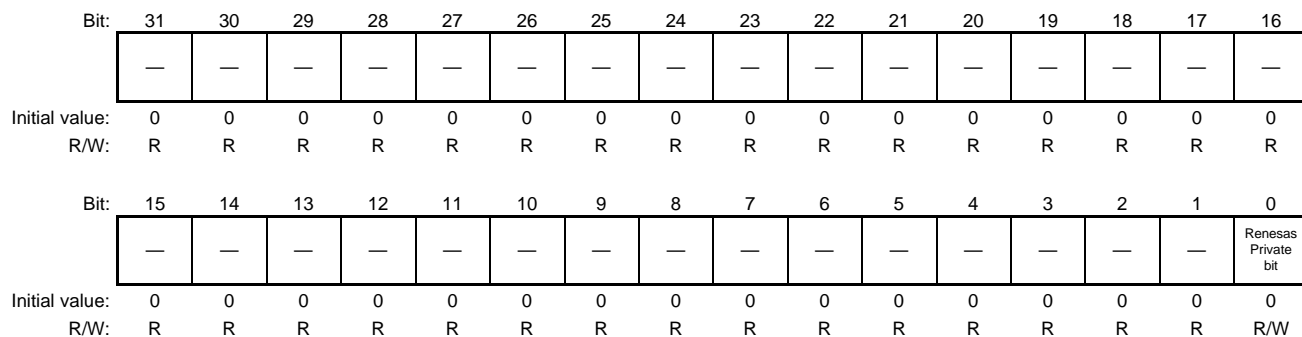


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | P0_WDATA | H'0000_0000 | W | Write data to PIPE0. This register is controlled in a unit of a packet and can't be written across the packet. That is, it is not allowed to concatenate the current packet with the next packet to write them to this register even when the length of the current packet is not multiples of 4 and the last word of it is less than 4 bytes. In that case, the next packet can be written to this register after the write of the previous packet has been completed and it has been transmitted. This register is valid only for IN direction. |

Note: Once the access to one of P0_READ, Pn_READ, P0_WRITE and Pn_WRITE registers is started in order to read or write a packet, it is prohibited to access to other Px_yyy register before whole of a packet has been read or written with the register which is currently used. The access to these registers is controlled in the unit of packet, so the targeted register can be changed at the point that the read or write of the current packet has been completed and the access for the next packet is not started yet. See section 63.9.2 for details.

63.2.2.53 DRD Power Control Register (DRD_PW_CON)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------|---------------|-----|--|
| 31 to 1 | — | All 0 | R | Reserved |
| 0 | Renesas Private bit | B'0 | R/W | Renesas Private bit This bit can be written only "0". DO NOT write "1". |

63.2.2.54 PIPE Common Setting Register (PIPE_COM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | PIPE_NUM | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 17 | — | All 0 | R | Reserved |
| 16 | — | B'0 | R/W | The initial value of this bit is 0. This bit can be read and written but it is not allowed to change the value of this bit. Only 0 is allowed as the value to be written. |
| 15 to 5 | — | All 0 | R | Reserved |
| 4 to 0 | PIPE_NUM | B'0_0000 | R/W | Set the index of the target PIPE 5'd0: Invalid 5'd1-5'd5: Index of PIPE |

Note: Once the access to one of P0_READ, Pn_READ, P0_WRITE and Pn_WRITE registers is started in order to read or write a packet, it is prohibited to access to other Px_yyy register before whole of a packet has been read or written with the register which is currently used. The access to these registers is controlled in the unit of packet, so the targeted register can be changed at the point that the read or write of the current packet has been completed and the access for the next packet is not started yet. See section 63.9.2 for details.

63.2.2.55 PIPEn Mode Setting Register (Pn_MOD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

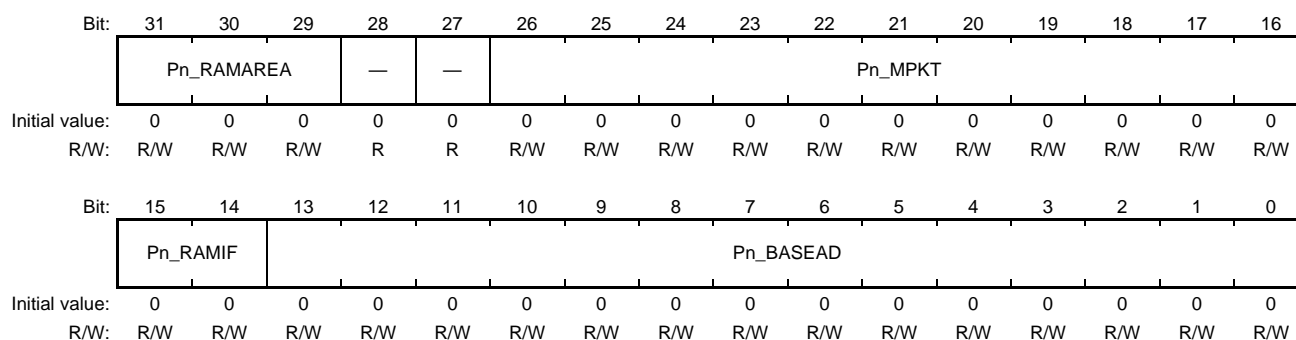
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-------------|-----|----|--------|---------|----------|-----|-----|--------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | Pn_ERDY_CON | — | — | — | — | — | — | — | Pn_BOT | Pn_ST REAM |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | Pn_DIR | Pn_TYPE | Pn_EPNUM | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 26 | — | All 0 | R | Reserved |
| 25 to 24 | Pn_ERDY_CON | B'00 | R/W | <p>This field specifies the condition to resume the transfer for PIPE.</p> <p>In case the device works in SS, the controller requests the exit from U1 or U2 if it is in the low power state or transmits ERDY if it is in the flow-control state, when the condition below is satisfied.</p> <p>In case the device works in HS or FS and the direction of transfer is IN (Pn_MOD.Pn_DIR==1), the controller requests the exit from L1 if it is in the low power state when the condition below is satisfied.</p> <p>The condition is as follows.</p> <p>In case the direction of transfer is OUT (Pn_MOD.Pn_DIR==0): the device is in the flow-control state and the buffer of PIPEn has available space equivalent to the value in this field or more.</p> <p>In case the direction of transfer is IN (Pn_MOD.Pn_DIR==1): the device is in the flow-control state or L1 and the buffer of PIPEn has packets equivalent to the value in this field or more.</p> <p>the device is in the flow-control state or L1 and the buffer of PIPEn has the last packet of the transfer.</p> <p>Note that the controller requires clock from USB3.0-PHY when it requests the exit from U1 or U2 and clock from USB2.0-PHY when it requests the exit from L1. If not, the controller doesn't request the exit from the states.</p> <p>[Setting Value]</p> <p>B'00: 1 Block (Packet) B'01: 2 Blocks (Packets) B'10: 4 Blocks (Packets) B'11: 8 Blocks (Packets)</p> |
| 23 to 18 | — | All 0 | R | Reserved |
| 17 | Pn_BOT | B'0 | R/W | <p>This bit specifies whether Bulk Only Transfer (BOT) mode is used or not.</p> <p>1: BOT mode 0: Normal mode</p> <p>In case this bit is set to 1, Pn_INT_STA.Pn_CBW_STA is enabled.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 16 | Pn_STREAM | B'0 | R/W | This bit is specified whether Stream bulk protocol is used or not. 1: Stream bulk mode 0: Normal mode This bit can be used only for bulk PIPE. (Pn_TYPE[1:0]== B'10) |
| 15 to 7 | — | All 0 | R | Reserved |
| 6 | Pn_DIR | B'0 | R/W | This bit specifies the direction of transfer. 0: OUT 1: IN |
| 5 to 4 | Pn_TYPE | B'00 | R/W | This field specifies the type of transfer of PIPEn. B'00: Reserved (This setting is prohibited) B'01: Isochronous (This setting is prohibited) B'10: Bulk B'11: Interrupt |
| 3 to 0 | Pn_EPNUM | H'0 | R/W | This field specifies the index of endpoint of PIPEn. |

63.2.2.56 PIPEn RAM Mapping Register (Pn_RAMMAP)

| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
|--------|--------------------------|--------|--------|
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|--|
| 31 to 29 | Pn_RAMAREA A | B'000 | R/W | This field specifies the area in SRAM for PIPEn. The area of SRAM attached to the interface specified in Pn_RAMIF[1:0] is given to PIPEn. The area is calculated as 1 Kbytes × 2 ^(Pn_RAMAREA[2:0]) . 1 Block requires 1 Kbytes regardless of the max packet size in Pn_MPKT[10:0]. Pn_RAMAREA[2:0] the area given to PIPEn B'000 1 Kbytes (1 Block) B'001 2 Kbytes (2 Blocks) B'010 4 Kbytes (4 Blocks) B'011 8 Kbytes (8 Blocks) B'100 16 Kbytes (16 Blocks) B'101- B'111 Setting prohibited |
| 28 to 27 | — | All 0 | R | Reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------------|-----|--|
| 26 to 16 | Pn_MPKT | B'000_0000_0000 | R/W | <p>This field specifies the max packet size of PIPEn. Select the value set to this field from among 8, 16, 32, 64, 512 and 1024.</p> <p>PIPE0 or bulk PIPE are defined to have the max packet size from among them in all speed mode.</p> <p>But interrupt PIPE may have the max packet size which doesn't equal to any of them. In that case, select the value which is more than the max packet size and proximate one, and set it to this field. The controller controls buffer in the unit of the values above.</p> |
| 15 to 14 | Pn_RAMIF | B'00 | R/W | <p>This field specifies the index of SRAM interface connected to SRAM allocated to PIPEn.</p> |
| 13 to 0 | Pn_BASEAD | B'00_0000_0000_0000 | R/W | <p>This field specifies the base address in SRAM allocated to PIPEn. Sets Base Address of 64 bit buffer RAM for EPC to be allocated to PIPEn.</p> <p>The address for 64 bit word is used.</p> <p>(ex: If 4K bytes from the start of RAM is assigned to PIPE1 and the area from the boundary is assigned to PIPE2, set H'200 to this field for PIPE2.)</p> <p>This field can be modified when Pn_CON.Pn_EN==0.</p> <p>Since the area is assigned in the unit of 1K bytes, the lower 7 bits of this field is read only and fixed to 0.</p> |

63.2.2.57 PIPEn Control Register (Pn_CON)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------|------------------|----|----|-------------|------------|-------------|----------------|----|----|----|----|----|----|-------------|------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | Pn_EN | Pn_DAT AIF_EN | — | — | — | — | — | Pn_SE ND | — | — | — | — | — | — | Pn_RES | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R | R | R | R | R | W | R | R | R | R | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | Pn_LA ST | Pn_BYTE_EN | Pn_SE ND | Pn_RE S_WEN | — | — | — | — | — | — | Pn_BC LR | Pn_CL R |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | W | W | W | W | W | R | R | R | R | R | W | W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|---|
| 31 | Pn_EN | B'0 | R/W | <p>This bit enables or disables PIPEn.</p> <p>The setting in this bit is valid when USB_COM_CON.CONF==1.</p> <p>In case this bit is disabled, make sure that no data transfer is executed to PIPEn.</p> <p>In case this bit is enabled, make sure that the initialization of PIPEn has been done.</p> <p>1: PIPEn Enabled 0: PIPEn disabled (PIPEn gives no response for any access.)</p> |
| 30 | Pn_DATAIF_EN | B'0 | R/W | <p>This bit enables or disables Data Interface of the controller for PIPEn.</p> <p>As the controller also uses Data Interface of the controller, Data Interface should be enabled in the configuration with AXI-IF.</p> <p>1: Data Interface is enabled 0: Data Interface is disabled</p> <p>Note that this bit is reset to 0 every time CBW is received with Bulk Only Transfer (BOT) mode. (Pn_MOD.Pn_BOT==1).</p> <p>If data transfer is requested in CBW, it is required to set this bit to 1 again before the data transfer.</p> |
| 29 to 25 | — | All 0 | R | Reserved |
| 24 | Pn_SEND | B'0 | W | <p>When 1 is written to this bit, the controller requests the exit from U1, U2 or L1 if it is in the states, or it transmits ERDY with no condition if it is not in the states.</p> <p>Since the controller automatically sends ERDY when it wants to resume the transfer, it is not required to use this bit basically. If there is any case ERDY is required to be sent intentionally, this bit is available for that case.</p> <p>This bit is write only and the value read from this bit is always 0.</p> |
| 23 to 18 | — | All 0 | R | Reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 17 to 16 | Pn_RES | B'00 | R/W | <p>In case of the direction of transfer is OUT (Pn_MOD.Pn_DIR==0), this field specifies the response of PIPE_n for the reception of OUT data.</p> <p>B'00: Forced NRDY/NAK Response B'01: Normal Response (Data can be accepted depending on the buffer status) B'10: Forced STALL Response B'11: Setting prohibited</p> <p>ERDY is transmitted when the value of this bit is changed from B'00 to B'01 if the device works in SS and the buffer has available space for PIPE_n.</p> <p>In case of Stream bulk OUT (Pn_MOD.Pn_STREAM==1), this bit is automatically changed to B'00 (Forced NRDY/NAK response) when DP(PRIME) or DP(NoStream) is received.</p> <p>In case of the direction of transfer is IN (Pn_MOD.Pn_DIR==1), this field specifies the response of PIPE_n for the transmission of IN data.</p> <p>B'00: Forced NRDY/NAK Response B'01: Normal Response (Data can be transmitted depending on the buffer status) B'10: Forced STALL Response B'11: Setting prohibited</p> <p>ERDY is transmitted when the value of this bit is changed from B'00 to B'01 if the device works in SS and the buffer has data to be transmitted from PIPE_n.</p> <p>In case of Stream bulk OUT (Pn_MOD.Pn_STREAM==1), this bit is automatically changed to B'00 (Forced NRDY/NAK response) when ACK (PRIME) is received.</p> <p>Note that Pn_RES_WEN is set to 1 at the same timing this field is modified.</p> |
| 15 to 12 | — | All 0 | R | Reserved |
| 11 | Pn_LAST | B'0 | W | <p>This bit indicates whether the packet written through Pn_WRITE register is the last one of the transfer (of the Stream ID).</p> <p>This bit is set at the same timing Pn_SEND is set in order to transmit the packet written through Pn_WRITE register. In case that the packet is the last one, set this bit and Pn_SEND in one word access.</p> <p>1: The last packet 0: Not the last packet</p> <p>Pn_INT_STA.Pn_LSTTR_STA is asserted when the packet with this bit set has been sent and has been acknowledged by host.</p> <p>This bit is write only and the value read from this bit is always 0.</p> |
| 10 to 9 | Pn_BYTE_EN | B'00 | W | <p>This bit specifies the number of valid bit in the last word that was written through Pn_WRITE register.</p> <p>This bit is set the same time as writing a value in Pn_SEND.</p> <p>00: 4 bytes are valid. 01: Only 1 byte is valid. (Only DATA[7:0] is valid.) 10: Only 2 bytes are valid. (Only DATA[15:0] is valid.) 11: Only 3 bytes are valid. (Only DATA[23:0] is valid.)</p> <p>This field is write only and the value read from this field is always 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------|---------------|-----|---|
| 8 | Pn_SEND | B'0 | W | Write 1 to this bit in order to transmit the packet written to Pn_WRITE register. This bit is write only and the value read from this bit is always 0. |
| 7 | Pn_RES_WE N | B'0 | W | Write 1 to this bit when changing Pn_RES[1:0]. This bit is write only and the value read from this bit is always 0. |
| 6 to 2 | — | All 0 | R | Reserved |
| 1 | Pn_BCLR | B'0 | W | Write 1 to this bit in order to clear buffer of PIPEn. Set Pn_RES[1:0] to B'00 (forced NRDY/NAK response mode) and make sure Pn_ACKSTS==0 (no data is being transferred) and stop the access to buffer before initializing the buffer with this bit. In case of Stream bulk protocol (Pn_MOD.Pn_STREAM==1), Pn STREAM register is also initialized when this bit is used. This bit shows 1 until the initialization is completed. It is not allowed to access other registers until this bit is cleared to 0 after writing 1 to this bit. |
| 0 | Pn_CLR | B'0 | W | Write 1 to this bit in order to initialize PIPEn (clearing sequence number, data PID and buffer). Set Pn_RES[1:0] to B'00 (forced NRDY/NAK response mode) and make sure Pn_ACKSTS==0 (no data is being transferred) and stop the access to the buffer of PIPEn before PIPEn is initialized. This bit shows 1 until the initialization is completed. It is not allowed access other registers until this bit is cleared to 0 after writing 1 to this bit. |

63.2.2.58 PIPEn Status Register (Pn_STA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----------------|---------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | Pn_FL OWSTS | Pn_AC KSTS | Pn_SN DSTS | Pn_BU FSTS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------|---------------|-----|---|
| 31 to 4 | — | All 0 | R | Reserved |
| 3 | Pn_FLOWSTS | B'0 | R | This bit shows the flow control state of PIPEn. 1: flow control state 0: normal state |
| 2 | Pn_ACKSTS | B'0 | R | This bit shows whether PIPEn has data that is currently being transferred or not. In case that the direction of transfer is OUT (Pn_MOD.Pn_DIR = 0), this bit shows 1 if PIPEn has data which has been received but ACK for it has not been transmitted yet. In case that the direction of transfer is IN (Pn_MOD.Pn_DIR==1), this bit shows 1 if PIPEn has data which has already been transmitted but has not received ACK for it yet. |
| 1 | Pn_SNDSTS | B'0 | R | This bit shows 1 if PIPEn has data which has not been transmitted yet. This bit is reset to 0 when all of written data have been transmitted and ACKs for them have been received. This bit is invalid for the case that the direction of transfer is OUT (Pn_MOD.Pn_DIR==0). |
| 0 | Pn_BUFSTS | B'0 | R | This bit shows the buffer status of PIPEn. 0: The buffer can't accept data for write, or it has no packet for read. 1: the buffer can accept data for write, or it has one or more packets for read. In case that the direction of transfer is OUT (Pn_MOD.Pn_DIR==0), this bit shows 1 if it has one or more received packets in the buffer. In case that the direction of transfer is IN (Pn_MOD.Pn_DIR==1), this bit shows 1 if data to be transmitted can be written to the buffer. |

Note: Once the access to one of P0_READ, Pn_READ, P0_WRITE and Pn_WRITE registers is started in order to read or write a packet, it is prohibited to access to other Px_yyy register before whole of a packet has been read or written with the register which is currently used. The access to these registers is controlled in the unit of packet, so the targeted register can be changed at the point that the read or write of the current packet has been completed and the access for the next packet is not started yet. See section 63.9.2 for details.

63.2.2.59 PIPEn Interrupt Status Register (Pn_INT_STA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|------------------|-----------------|-----------------|----|------------------|------------------|------------------|------------------|------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | Pn_CB W_STA | — | — | — | Pn_STE RR_STA | Pn_STR MX_STA | Pn_CST RM_STA | Pn_NST RM_STA | Pn_PRIM E_STA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | RW1C | R | R | R | RW1C | RW1C | RW1C | RW1C | RW1C |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | Pn_RCV NL_STA | Pn_ERD Y_STA | Pn_FLO W_STA | — | Pn_LSTT R_STA | — | Pn_STAL L_STA | Pn_NRD Y_STA | Pn_BFR DY_STA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | RW1C | RW1C | RW1C | R | RW1C | R | RW1C | RW1C | RW1C |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|----------|---|
| 31 to 25 | — | All 0 | R | Reserved |
| 24 | Pn_CBW_ STA | B'0 | RW1 C | <p>This bit is valid only when Bulk Only Transfer is supported (Pn_MOD.Pn_BOT = 1) and the direction of transfer is OUT (Pn_MOD.Pn_DIR = 0).</p> <p>This bit is set to 1 when CBW is received to PIPEn and the data can be read from Pn_READ register.</p> <p>CBW is defined by following conditions.</p> <ul style="list-style-type: none"> — The length of the packet is 31 bytes. — The first 4 bytes of the packet (dCBWSignature) equals to 43425355h. |
| 23 to 21 | — | All 0 | R | Reserved |
| 20 | Pn_STERR_ STA | B'0 | RW1 C | <p>This bit is set to 1 when Stream bulk protocol error occurs in PIPEn.</p> <p>See USB3.0 specification for detail of error in Stream bulk protocol.</p> <p>This bit is invalid in case of HS and FS.</p> |
| 19 | Pn_STRMX_ STA | B'0 | RW1 C | <p>This bit is set to 1 when Stream ID of the packet received at PIPEn from host doesn't equal to the value in Pn_STREAM.Pn_STREAM_C (except for PRIME and NoStream).</p> <p>See Pn_NRDYSTRM register to know the Stream ID of received packet which has been rejected.</p> <p>This bit is invalid in case of HS and FS.</p> |
| 18 | Pn_CSTRM_ STA | B'0 | RW1 C | <p>This bit might be changed and an interrupt due to it might be generated if not masked. Make sure not to set bit 18 in Pn_INT_ENA register in order to disable the interrupt due to this bit.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------|---------------|----------|--|
| 17 | Pn_NSTRM_ STA | B'0 | RW1 C | <p>This bit indicates that NoStream is designated as Stream ID for PIPEn and is valid only in case Stream bulk protocol is used. (Pn_MOD.Pn_STREAM = 1).</p> <p>In case the direction of transfer is IN (Pn_MOD.Pn_DIR = 1), this bit is set when ACK(NoStream) is received. Pn_CON.Pn_RES[1:0] is automatically set to NRDY response (=B'00) in that case.</p> <p>In case of the direction of transfer is OUT (Pn_MOD.Pn_DIR = 0), this bit is set when DP(NoStream) is received. Pn_CON.Pn_ERS[1:0] is automatically set to NRDY response (=B'00) in that case.</p> <p>This bit is invalid in case of HS/FS.</p> |
| 16 | Pn_PRIME_ STA | B'0 | RW1 C | <p>This bit indicates that PRIME is designated as Stream ID for PIPEn and is valid only in case Stream bulk protocol is used. (Pn_MOD.Pn_STREAM = 1).</p> <p>In case the direction of transfer is IN (Pn_MOD.Pn_DIR = 1), this bit is set when ACK(PRIME) is received. Pn_CON.Pn_RES[1:0] is automatically set to NRDY response (=B'00) in that case.</p> <p>In case of the direction of transfer is OUT (Pn_MOD.Pn_DIR = 0), this bit is set when DP(PRIME) is received. Pn_CON.Pn_ERS[1:0] is automatically set to NRDY response (=B'00) in that case.</p> <p>This bit is invalid in case of HS/FS.</p> |
| 15 to 9 | — | All 0 | R | Reserved |
| 8 | Pn_RCVNL_ STA | B'0 | RW1 C | <p>This bit indicates that zero length packet has been received at PIPEn and becomes ready to be read.</p> <p>The zero length packet received is automatically discarded when Pn_LNG register is read for it. Therefore, this bit is asserted when Pn_LNG register shows 0 for the zero length packet received.</p> <p>When this bit is asserted, it is required to read Pn_LNG register as the read action for the zero length packet. If Pn_LNG register shows 0, it means the zero length packet has been read.</p> |
| 7 | Pn_ERDY_ STA | B'0 | RW1 C | <p>This bit indicates that the condition to transmit ERDY from PIPEn is satisfied.</p> <p>If Pn_CON.Pn_RES[1:0] has the value other than NRDY response and this bit is asserted, ERDY is transmitted to host.</p> <p>For Stream bulk protocol, Pn_CON.Pn_RES[1:0] should be set to normal response (=B'01) so that ERDY is transmitted to specify Stream ID.</p> <p>This bit is invalid in case of HS and FS.</p> |
| 6 | Pn_FLOW_ STA | B'0 | RW1 C | <p>This bit indicates that PIPEn enters the flow control state.</p> <p>In case the direction of transfer is IN (Pn_MOD.Pn_DIR = 1), this bit is set to 1 when PIPEn sends NRDY or DP with EOB = 1.</p> <p>In case the direction of transfer is OUT (Pn_MOD.Pn_DIR = 0), this bit is set to 1 when PIPEn sends NRDY or ACK with NumP = 0.</p> <p>This bit is also set to 1 when PIPEn receives the deferred packet.</p> <p>This bit is invalid in case HS and FS.</p> |
| 5 | — | B'0 | R | Reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|--------------|---------------|----------|--|
| 4 | Pn_LSTTR_STA | B'0 | RW1 C | <p>This bit indicates that PIPE_n has transmitted or received the last packet of the transfer.</p> <p>In case the direction of transfer is IN (Pn_MOD.Pn_DIR = 1), this bit is set to 1 when PIPE_n has transmitted the packet for which Pn_CON.Pn_LAST is set.</p> <p>In case the direction of transfer is OUT (Pn_MOD.Pn_DIR = 0), this bit is set to 1 when PIPE_n has received the short packet or the packet with PP = 0.</p> |
| 3 | — | B'0 | R | Reserved |
| 2 | Pn_STALL_STA | B'0 | RW1 C | This bit is set to 1 when STALL has been transmitted from PIPE _n . |
| 1 | Pn_NRDY_STA | B'0 | RW1 C | This bit is set to 1 when NRDY or NAK has been transmitted from PIPE _n . |
| 0 | Pn_BFRDY_STA | B'0 | RW1 C | <p>This bit indicates the buffer status of PIPE_n.</p> <p>In case the direction of transfer is IN (Pn_MOD.Pn_DIR = 1), this bit is set to 1 when the buffer of PIPE_n is ready to accept the next IN data. This bit is not asserted for the write of first packet after the controller is initialized by reset (hardware reset, USB_COM_CON.PIPE_CLR, Pn_CON.Pn_BCLR and Pn_CON.Pn_CLR). After reset, the first packet can be written immediately although this bit is not asserted.</p> <p>In case the direction of transfer is OUT (Pn_MOD.Pn_DIR = 0), this bit is set to 1 when the next packet in the buffer of PIPE_n is ready to be read.</p> |

63.2.2.60 PIPEn Interrupt Enable Register (Pn_INT_ENA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|--------------|-------------|-------------|----|--------------|--------------|--------------|--------------|--------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | Pn_CBW_ENA | — | — | — | Pn_STERR_ENA | Pn_STRMX_ENA | Pn_CSTRM_ENA | Pn_NSTRM_ENA | Pn_PRIME_ENA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | Pn_RCVNL_ENA | Pn_ERDY_ENA | Pn_FLOW_ENA | — | Pn_LSTTR_ENA | — | Pn_STALL_ENA | Pn_NRDY_ENA | Pn_BFRDY_ENA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|--------------|---------------|-----|---|
| 31 to 25 | — | All 0 | R | Reserved |
| 24 | Pn_CBW_ENA | B'0 | R/W | This bit is used to enable the interrupt due to Pn_INT_STA.Pn_CBW_STA. 1: an interrupt due to Pn_INT_STA.Pn_CBW_STA is enabled. 0: an interrupt due to Pn_INT_STA.Pn_CBW_STA is disabled. |
| 23 to 21 | — | All 0 | R | Reserved |
| 20 | Pn_STERR_ENA | B'0 | R/W | This bit is used to enable the interrupt due to Pn_INT_STA.Pn_STERR_STA. 1: an interrupt due to Pn_INT_STA.Pn_STERR_STA is enabled. 0: an interrupt due to Pn_INT_STA.Pn_STERR_STA is disabled. |
| 19 | Pn_STRMX_ENA | B'0 | R/W | This bit is used to enable the interrupt due to Pn_INT_STA.Pn_STRMX_STA. 1: an interrupt due to Pn_INT_STA.Pn_STRMX_STA is enabled. 0: an interrupt due to Pn_INT_STA.Pn_STRMX_STA is disabled. |
| 18 | Pn_CSTRM_ENA | B'0 | R/W | The initial value of this bit is 0. This bit can be read and written but it is not allowed to change the value of this bit. Only 0 is allowed as the value to be written. |
| 17 | Pn_NSTRM_ENA | B'0 | R/W | This bit is used to enable the interrupt due to Pn_INT_STA.Pn_NSTRM_STA. 1: an interrupt due to Pn_INT_STA.Pn_NSTRM_STA is enabled. 0: an interrupt due to Pn_INT_STA.Pn_NSTRM_STA is disabled. |
| 16 | Pn_PRIME_ENA | B'0 | R/W | This bit is used to enable the interrupt due to Pn_INT_STA.Pn_PRIME_STA. 1: an interrupt due to Pn_INT_STA.Pn_PRIME_STA is enabled. 0: an interrupt due to Pn_INT_STA.Pn_PRIME_STA is disabled. |
| 15 to 9 | — | All 0 | R | Reserved |
| 8 | Pn_RCVNL_ENA | B'0 | R/W | This bit is used to enable the interrupt due to Pn_INT_STA.Pn_RCVNL_STA. 1: an interrupt due to Pn_INT_STA.Pn_RCVNL_STA is enabled. 0: an interrupt due to Pn_INT_STA.Pn_RCVNL_STA is disabled. |
| 7 | Pn_ERDY_ENA | B'0 | R/W | This bit is used to enable the interrupt due to Pn_INT_STA.Pn_ERDY_STA. 1: an interrupt due to Pn_INT_STA.Pn_ERDY_STA is enabled. 0: an interrupt due to Pn_INT_STA.Pn_ERDY_STA is disabled. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|--------------|---------------|-----|---|
| 6 | Pn_FLOW_ENA | B'0 | R/W | This bit is used to enable the interrupt due to Pn_INT_STA.Pn_FLOW_STA. 1: an interrupt due to Pn_INT_STA.Pn_FLOW_STA is enabled. 0: an interrupt due to Pn_INT_STA.Pn_FLOW_STA is disabled. |
| 5 | — | B'0 | R | Reserved |
| 4 | Pn_LSTTR_ENA | B'0 | R/W | This bit is used to enable the interrupt due to Pn_INT_STA.Pn_LSTTR_STA. 1: an interrupt due to Pn_INT_STA.Pn_LSTTR_STA is enabled. 0: an interrupt due to Pn_INT_STA.Pn_LSTTR_STA is disabled. |
| 3 | — | B'0 | R | Reserved |
| 2 | Pn_STALL_ENA | B'0 | R/W | This bit is used to enable the interrupt due to Pn_INT_STA.Pn_STALL_STA. 1: an interrupt due to Pn_INT_STA.Pn_STALL_STA is enabled. 0: an interrupt due to Pn_INT_STA.Pn_STALL_STA is disabled. |
| 1 | Pn_NRDY_ENA | B'0 | R/W | This bit is used to enable the interrupt due to Pn_INT_STA.Pn_NRDY_STA. 1: an interrupt due to Pn_INT_STA.Pn_NRDY_STA is enabled. 0: an interrupt due to Pn_INT_STA.Pn_NRDY_STA is disabled. |
| 0 | Pn_BFRDY_ENA | B'0 | R/W | This bit is used to enable the interrupt due to Pn_INT_STA.Pn_BFRDY_STA. 1: an interrupt due to Pn_INT_STA.Pn_BFRDY_STA is enabled. 0: an interrupt due to Pn_INT_STA.Pn_BFRDY_STA is disabled. |

63.2.2.61 PIPEn Reserved Packet Register (Pn_RSVPKT)

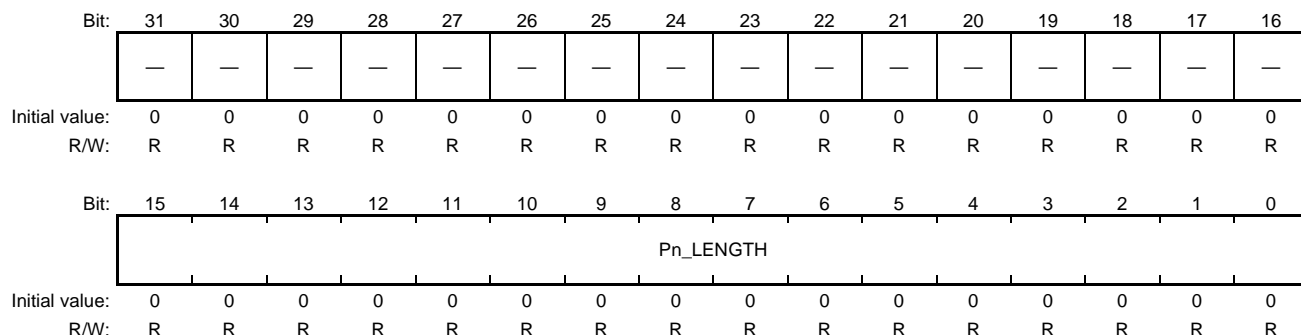
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|-----------|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | Pn_RSVPKT | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 31 to 5 | — | All 0 | R | Reserved |
| 4 to 0 | Pn_RSVPKT | B'00_0000 | R | This field shows the number of packets those are not transmitted yet in the buffer of PIPEn. (n is 0-16) This field is valid only for IN PIPE. |

63.2.2.62 PIPEn Length Register (Pn_LNG)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



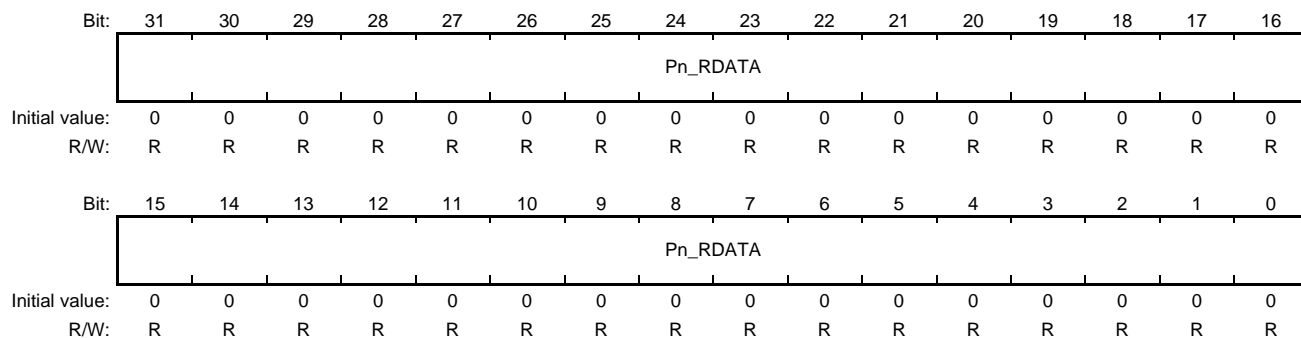
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 to 0 | Pn_LENGTH | H'0000 | R | <p>This field shows the number of bytes in received packet which is currently read from Pn_READ register.</p> <p>The value is decremented every time Pn_READ register is read. Since Pn_READ register allows only 32-bit access, the value is decremented by 4.</p> <p>Since Pn_READ register is controlled in a unit of a packet, the length of received packet is shown in this register when the first 4 bytes of it are shown in Pn_READ register and have not been read yet.</p> <p>This field is valid only for OUT PIPE.</p> |

Note: Once the access to one of P0_READ, Pn_READ, P0_WRITE and Pn_WRITE registers is started in order to read or write a packet, it is prohibited to access to other Px_yyy register before whole of a packet has been read or written with the register which is currently used. The access to these registers is controlled in the unit of packet, so the targeted register can be changed at the point that the read or write of the current packet has been completed and the access for the next packet is not started yet. See section 63.9.2 for details.

63.2.2.63 PIPEn Read Register (Pn_READ)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

If there is no data in buffer, H'0000_0000 is returned. Only 32-bit access is allowed to this register.

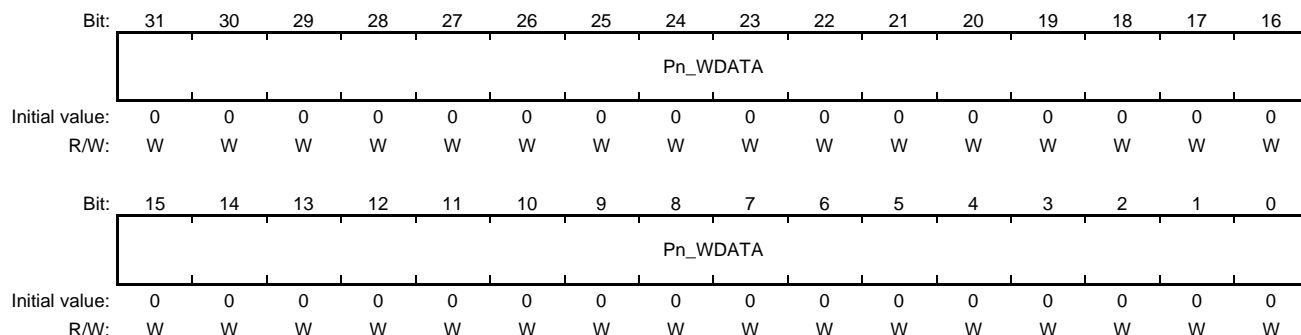


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 0 | Pn_RDATA | H'0000_0000 | R | <p>Read data from PIPEn.</p> <p>This register is controlled in a unit of a packet and can't be read across the packet.</p> <p>That is, this register is updated in a unit of a packet. The current packet is never concatenated with the next packet even when the length of the current packet is not multiples of 4 and the last word of it is less than 4 bytes. In that case, the first 4 bytes of the next packet are shown in this register after the last word of the previous packet has been read.</p> <p>If the length of the current packet is not multiples of 4 and the last word of it is less than 4 bytes, the last word has one or more invalid bytes. In order to know how many valid bytes the last word has, see Pn_LENGTH[7:0] before the read of the packet and calculate from it. The valid bytes are aligned from Pn_RDATA[7:0].</p> <p>This register is valid only for OUT PIPE.</p> |

Note: Once the access to one of P0_READ, Pn_READ, P0_WRITE and Pn_WRITE registers is started in order to read or write a packet, it is prohibited to access to other Px_yyy register before whole of a packet has been read or written with the register which is currently used. The access to these registers is controlled in the unit of packet, so the targeted register can be changed at the point that the read or write of the current packet has been completed and the access for the next packet is not started yet. See section 63.9.2 for details.

63.2.2.64 PIPEn Write Register (Pn_WRITE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

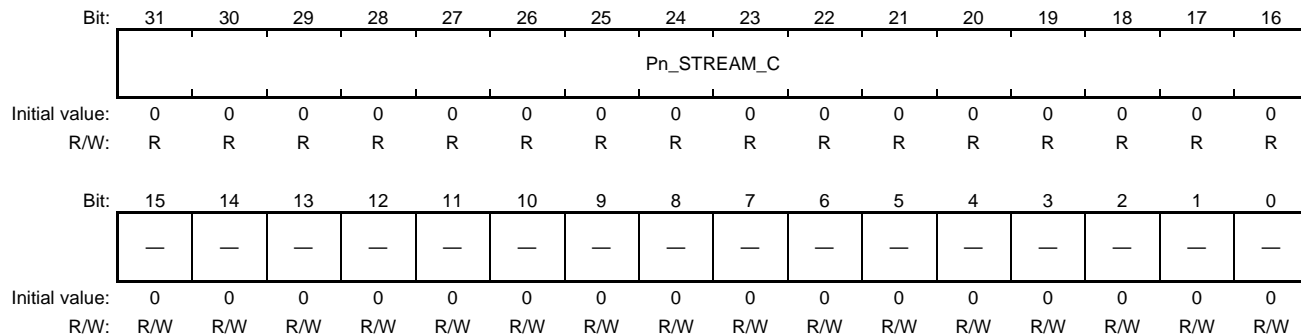


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 0 | Pn_WDATA | H'0000_0000 | W | <p>Write data to PIPEn.</p> <p>This register is controlled in a unit of a packet and can't be written across the packet.</p> <p>That is, it is not allowed to concatenate the current packet with the next packet to write them to this register even when the length of the current packet is not multiples of 4 and the last word of it is less than 4 bytes. In that case, the next packet can be written to this register after the write of the previous packet has been completed and it has been transmitted.</p> <p>This register is valid only for IN PIPE.</p> |

Note: Once the access to one of P0_READ, Pn_READ, P0_WRITE and Pn_WRITE registers is started in order to read or write a packet, it is prohibited to access to other Px_yyy register before whole of a packet has been read or written with the register which is currently used. The access to these registers is controlled in the unit of packet, so the targeted register can be changed at the point that the read or write of the current packet has been completed and the access for the next packet is not started yet. See section 63.9.2 for details.

63.2.2.65 PIPEn Stream ID Register (Pn_STREAM)

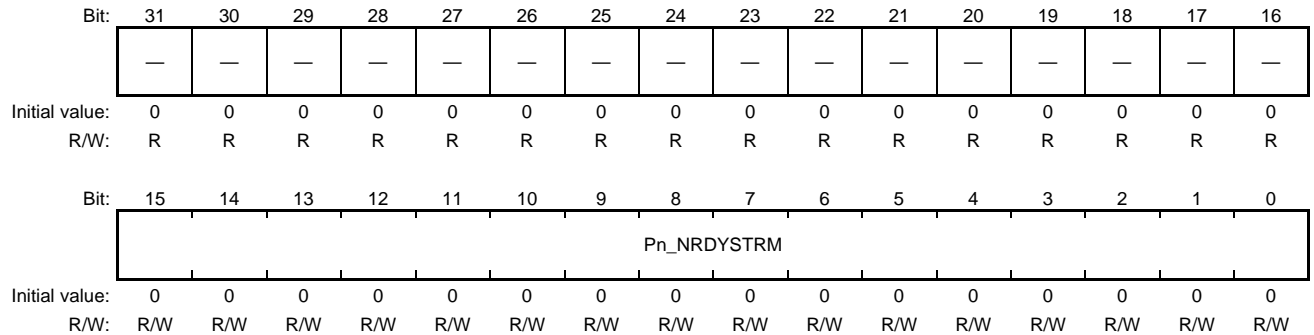
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|--|
| 31 to 16 | Pn_STREAM_C | H'0000 | R | This field indicates the Stream ID that is currently valid. This field is invalid in case of HS/FS. This field is valid only in case that the type of transfer is bulk (Pn_TYPE = B'10) and Stream ID is supported (Pn_MOD.Pn_STREAM = 1) for PIPEn. When this field shows H'0000, it means that Stream ID is not set yet. This field is reset when Pn_CON.Pn_BCLR is set. |
| 15 to 0 | — | All 0 | R/W | The initial value of this field is 0. This field can be read and written but it is not allowed to change the value of this field. Only 0 is allowed as the value to be written. |

63.2.2.66 PIPEn NRDY Stream ID Register (Pn_NRDYSTRM)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 to 0 | Pn_NRDYST RM | H'0000 | R/W | <p>This field indicates Stream ID that was requested from host but responded with NRDY at PIPEn.</p> <p>Peripheral portion responds with NRDY for the Stream ID which doesn't equal to the current Stream ID in Pn_STREAM register.</p> <p>This register is used to know what Stream ID is requested by host and rejected.</p> <p>This field is not updated in case that NRDY is returned due to NoStream or PRIME.</p> <p>This field is invalid in case of HS or FS.</p> |

63.2.2.67 Port Status and Control Register (PORTSC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

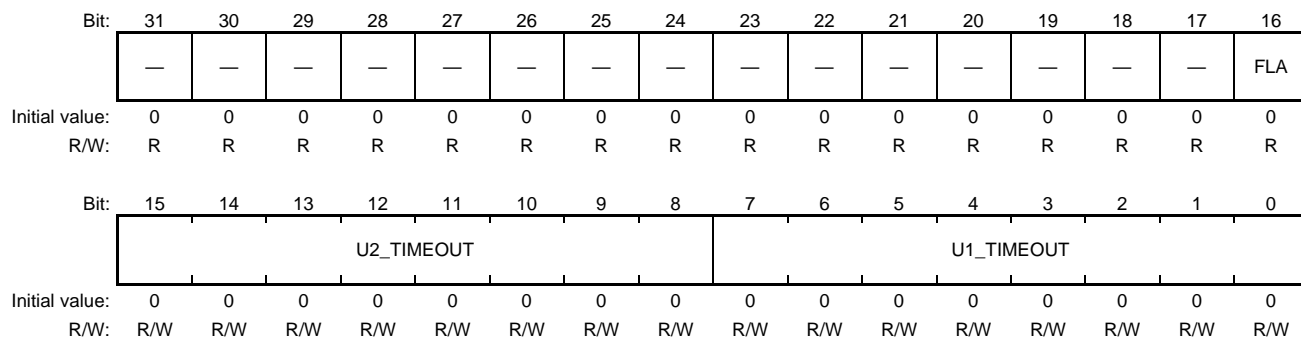
| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|-----|-----|-----|-----|----|----|----|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | LWS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | PLS | | | | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 17 | — | All 0 | R | Reserved |
| 16 | LWS | B'0 | R | Write 1 to this bit in order to enable the write access to PLS[3:0]. When this bit is 0, write to PLS[3:0] is ignored. This bit is write only and the value read from this bit is always 0. |
| 15 to 9 | — | All 0 | R | Reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 8 to 5 | PLS | H'4 | R/W | <p>[for write access]</p> <p>This field is used to request the direct transition of LTSSM state. For the timeout control of the transition to U1 or U2, see section 63.2.2.70.</p> <p>Set LWS bit to 1 when this field is written. As only 32-bit access is allowed to this register, make sure to set LWS to 1 in one word write for this register in order to change PLS[3:0].</p> <p>When the same state as the current state is written to this field, the write access is ignored.</p> <p>If the transition using this function has not been completed but the next write access to this field comes, the next write data is also ignored.</p> <p>If the request of transition to U2 state is written during the transition to U1 state is being executed due to U1 timeout, the request is ignored.</p> <p>If the request of transition to U0 state is written when LTSSM state is in U1, U2 or U3 state, the exit operation or the resume operation is started.</p> <p>If the remote wakeup is requested when LTSSM state is U1 or U2 state, hardware executes the resume operation to U0 state automatically and PLS[3:0] shows U0 after the operation has been completed.</p> <p>H'0: Transition request to U0 state H'1: Transition request to U1 state H'2: Transition request to U2 state H'3-H'F: Setting prohibited</p> <p>It is prohibited to request the transition to U1 or U2 state during control transfer.</p> <p>[for read access]</p> <p>This field shows the current state of LTSSM.</p> <p>H'0: U0 state H'1: U1 state H'2: U2 state H'3: U3 state H'4: Disabled state H'5: RxDetect state H'6: Inactivity state H'7: Polling state H'8: Recovery state H'9: Hot Reset state H'A: Compliance state H'B: Loopback state H'C- H'F: Setting prohibited</p> |
| 4 to 0 | — | All 0 | R | Reserved |

63.2.2.68 Port PM Status and Control Register (PORTPMSC)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



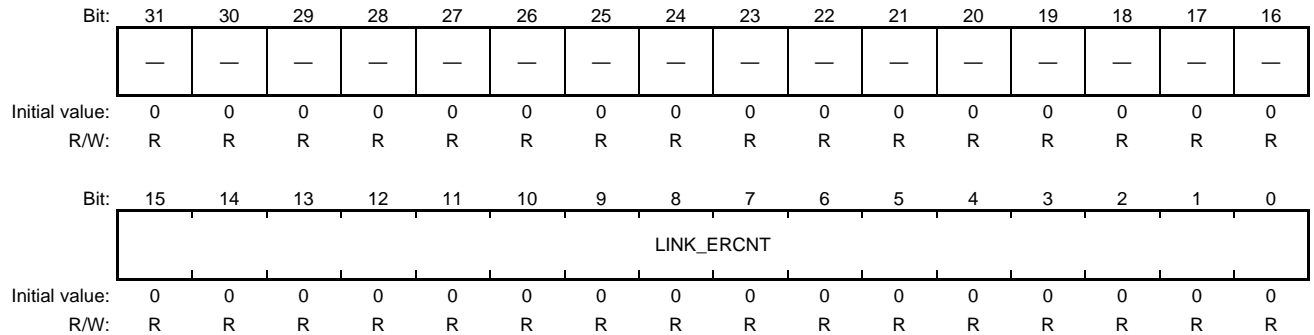
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 17 | — | All 0 | R | Reserved |
| 16 | FLA | B'0 | R | <p>This bit is set to 1 when Force LinkPM Accept bit in the received Set Link Function LMP equals to 1.</p> <p>In case this bit shows 1, the port accepts LGO_U1 and LGO_U2 with no condition.</p> <p>This bit is cleared to 0 when Set Link Function LMP with Force LinkPM Accept bit = 0 is received.</p> <p>Software can write 0 to clear this bit, but basically this bit is set or reset on the reception of Set Link Function LMP.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|--|
| 15 to 8 | U2_TIMEOUT T | H'00 | R/W | <p>This field indicates the timeout value in U2 inactivity timer.</p> <p>H'00: No timeout but LGO_Ux is always denied with LXU: default H'01: 256 us H'02: 512 us H'FE: 65.204 ms H'FF: No timeout and LGO_Ux is basically accepted. (But it might be rejected due to other reasons.)</p> <p>This field is automatically updated when U2 Inactivity LMP is received.</p> <p>When this field has the value from H'01 to H'FE, it means the idle timeout value for LTSSM to request the transition from U0 state to U1state as well as the idle timeout value for to make the transition from U1 state to U2 state.</p> <p>If U1_TIMEOUT[7:0] has H'00 or H'FF but U2_TIMEOUT[7:0] has a value from H'01 to H'FE, U2 timeout occurs although U1 timeout never. In that case Peripheral portion never sends LGO_U1 but sends LGO_U2 to request the transition from U0 state to U2 state. If downstream port accepts the request, the downstream port and Peripheral portion enter U2 state.</p> <p>Otherwise, U1 timeout occurs first and Peripheral portion sends LGO_U1 to request the transition from U0 state to U1 state. If downstream port accepts the request, the downstream port and Peripheral portion enter U1 state. If they stay in U1 state and U2 timeout occurs, they make the transition from U1 state to U2 state silently, without going back to U0 state.</p> <p>The transition is executed automatically by hardware when the timeout occurs.</p> <p>Note: This field is initialized to H'00 during reset and just after reset, but changed to H'FF when Peripheral portion detects disconnect as USB3.0 device. If reset (in this case, it will be power on reset) is asserted but the device is in the disconnected state, H'FF might be loaded to this field.</p> <p>Since this field is changed to H'FF when disconnect is detected, set the adequate value to this field after the connection if required.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------------|---------------|-----|---|
| 7 to 0 | U1_ TIMEOUT | H'00 | R/W | <p>This field indicates the timeout value in U1 inactivity timer.</p> <p>H'00: No timeout but LGO_Ux is always denied with LXU: default H'01: 1 us H'02: 2 us H'7F: 127 us H'80- H'FE: Setting prohibited H'FF: No timeout and LGO_Ux is basically accepted. (But it might be rejected due to other reasons.)</p> <p>This field is not automatically updated unlike U2_TIMEOUT[7:0]. Software is required to set this field.</p> <p>When this field has the value from H'01 to H'7F, it means the idle timeout value for LTSSM to request the transition from U0 state to U1 state. The hardware automatically sends LGO_U1 when the timeout occurs. If it is accepted, the transition is also executed by hardware automatically.</p> <p>Note: This field is initialized to H'00 during reset and just after reset, but changed to H'FF when Peripheral portion detects disconnect as USB3.0 device. If reset (in this case, it will be power on reset) is asserted but the device is in the disconnected state, H'FF might be loaded to this field.</p> <p>Since this field is changed to H'FF when disconnect is detected, set the adequate value to this field after the connection if required.</p> |

63.2.2.69 Port Link Information Register (PORTLI)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved |
| 15 to 0 | LINK_ERCNT | H'0000 | R | This field shows the number of errors detected. This field is reset when LTSSM enters U0 state in the first link-up after warm reset is received or when LTSSM comes back to U0 state after hot reset is received. |

63.2.2.70 SSIF Command Register (SSIFCMD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-----------|-----------|-----------|-----|-----|-----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | SSIF_URES | SSIF_UDIR | SSIF_UREQ | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|---|
| 31 to 10 | — | All 0 | R | Reserved |
| 9 to 8 | SSIF_URES | B'00 | R/W | <p>This field specifies the response of EPC when it receives LGO_U1 or LGO_U2.</p> <p>[bit1] this bit specifies the response when LGO_U2 is received.</p> <p>0: Normal response. (default)</p> <p>Peripheral portion basically accepts LGO_U2. (But it might be rejected due to other reasons.)</p> <p>1: Forced LXU response.</p> <p>Peripheral portion never accepts LGO_U2 and always returns LXU to it.</p> <p>[bit0] this bit specifies the response when LGO_U1 is received.</p> <p>0: Normal response. (default)</p> <p>Peripheral portion basically accepts LGO_U1. (But it might be rejected due to other reasons.)</p> <p>1: Forced LXU response.</p> <p>Peripheral portion never accepts LGO_U1 and always returns LXU to it</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|--|
| 7 to 6 | SSIF_UDIR | B'11 | R/W | <p>This field enables to transmit LGO_U1 or LGO_U2 when the transition to U1 or U2 state is requested directly through PORTSC.PLS[3:0].</p> <p>The transition to U1 or U2 state can be requested by writing the state to PORTSC.PLS[3:0]. If SSIF_UDIR[1:0] enables to send LGO_U1 or LGO_U2, Peripheral portion sends LGO_U1 or LGO_U2. If it is accepted, LTSSM transits to U1 or U2 state. If SSIF_UDIR[1:0] disables, EPC doesn't send LGO_U1 and LGO_U2 and writing to PORTSC_PLS[3:0] has no meaning.</p> <p>[bit1] this bit specifies the behavior when the transition to U2 state is requested. 0: LGO_U2 is transmitted. 1: LGO_U2 is not transmitted. (default)</p> <p>[bit0] this bit specifies the behavior when the transition to U1 state is requested. 0: LGO_U1 is transmitted. 1: LGO_U1 is not transmitted. (default)</p> |
| 5 to 4 | SSIF_UREQ | B'00 | R/W | <p>This field enables the transmission of LGO_U1 or LGO_U2 when the timeout on U1/U2 inactivity timer occurs.</p> <p>[bit1] this bit specifies the behavior when the timeout on U2 inactivity timer occurs. 0: LGO_U2 is transmitted. (default) 1: LGO_U2 is not transmitted.</p> <p>[bit0] this bit specifies the behavior when the timeout on U1 inactivity timer occurs. 0: LGO_U1 is transmitted. (default) 1: LGO_U1 is not transmitted.</p> <p>Note: This field is initialized to B'00 during reset and just after reset, but changed to B'11 when Peripheral portion detects disconnect as USB3.0 device. If reset (in this case, it will be power on reset) is asserted but the device is in the disconnected state, B'11 might be loaded to this field.</p> <p>Since this field is changed to B'11 when disconnect is detected, set this field to enable the transmission of LGO_Ux after the connection if required.</p> |
| 3 to 0 | — | All 0 | R | Reserved |

63.2.2.71 SSIF Link Setting 2 Register (SSIFLINKSET2)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----------------|------------------------|-----|-----|-----|-----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | PLL_WA KEUP | Rx_LFPS _TRG_ P0 | — | — | — | — | — | — | — | — |
| Initial value: | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | SKP_OS_Timer | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 26 | — | H'34 | R | Reserved |
| 25 | PLL_WAKEUP | B'1 | R/W | Select enable/disable of PLL initiation by assertion of E3_PLL_WAKEUP. 0: Disabled / 1: Enabled * Use this with Default setting (= 1). |
| 24 | Rx_LFPS_TRG_P0 | B'1 | R/W | Select enable/disable of transition to P0 by reception of U3 Wakeup LFPS in P3 state. 0: Disabled / 1: Enabled * Use this with Default setting (= 1). |
| 23 to 10 | — | H'0DC0 | R | Reserved |
| 9 to 0 | SKP_OS_Timer | H'0B0 | R/W | Sets interval of SKP OS transmission. * Use this with Default setting (= H'0B0). |

63.2.2.72 SSIF Link Setting 3 Register (SSIFLINKSET3)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|-----|-----|-----|-----|-----|-----|-----|--------------|-----|-----|---------------|-----|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | INTV_BW_RDET | | | SSIF_NUM_RDET | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| R/W: | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 to 15 | — | H'00EC0 | R | Reserved |
| 14 to 8 | — | H'0A | R/W | Reserved The initial value of this field is H'0A. This field can be read and written but it is not allowed to change the value of this bit. Only H'0A is allowed as the value to be written. |
| 7 to 4 | INTV_BW_RDET | B'1100 | R/W | Sets Timeout time in Rx. Detect State. (ms) * Use this with Default setting (=B'1100). <Reference> In USB 3.0 spec 7.5.3.7.1 Rx.Detect.Quiet Requirements, it is mentioned as below. The far-end receiver termination detection shall be disabled. A 12-ms timer shall be started upon entry to the substate. 12ms timer mentioned in the last condition can be adjusted with this field. |
| 3 to 0 | SSIF_NUM_RDET | B'1000 | R/W | This field indicates the number of trials to detect the far-end receiver termination in RxDetect.Active state. |

63.2.2.73 SSIF Link Setting 4 Register (SSIFLINKSET4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|-------------|-----|----|------------|-----|------------|----|----|----|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | SSIF_TXMGN | | | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | SSIF_DEEMPH | | — | — | — | SSIF_SWING | — | — | — | SSIF_SCRD |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R | R | R | R/W | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-------------|---------------|-----|---|
| 31 to 23 | — | All 0 | R | Reserved |
| 22 to 20 | SSIF_TXMGN | B'000 | R/W | This field indicates the level of transmitter voltage of USB3.0-PHY. B'000: Normal operation range B'001- B'111: Setting prohibited * Use this field in default setting (=B'000). |
| 19 to 10 | — | H'03F | R | Reserved |
| 9 to 8 | SSIF_DEEMPH | B'11 | R/W | This field indicates the level of transmitter de-emphasis of USB3.0-PHY. B'00: -6.0db B'01: -3.5db B'10: 0.0db B'11: default setting The compliance test of CP6/CP8 uses 0dB setting. For others, de-emphasis shall be -3.5dB. |
| 7 to 5 | — | All 0 | R | Reserved |
| 4 | SSIF_SWING | B'0 | R/W | This field indicates the level of transmitter swing voltage of USB3.0-PHY. 0: full swing (default) 1: half swing |
| 3 to 1 | — | All 0 | R | Reserved |
| 0 | SSIF_SCRD | B'0 | R/W | This bit is used to set the Disable Scrambling bit in TS1/TS2 Order Set. Write 1 to this bit when the Disable Scrambling bit is required to be set. See 6.4.1.1 of USB3.0 specification for details. |

63.2.3 PHY Register

63.2.3.1 USB30 Clock Setting Register (USB30_CLKSET0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|-----------------|-----|-----------------|-----|-----|-----|-----|-----|------------|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | Renesas Private | | Renesas Private | | | | | | USB30_FSEL | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|---|
| 15 to 11 | — | All 0 | R | Reserved |
| 10 to 9 | Renesas Private | B'10 | R/W | Write the read value back to these bits. |
| 8 to 6 | Renesas Private | B'111 | R/W | Write B'101 to these bits. |
| 5 to 0 | USB30_FSEL | B'10_0111 | R/W | Frequency Select Selects the reference clock frequency used for both SS and HS operations. USB_XTAL/USB_EXTAL used (50 MHz): B'00_0010 USB3S0_CLK_P/USB3S0_CLK_M used (100 MHz): B'10_0111 |

63.2.3.2 USB30 Clock Setting Register (USB30_CLKSET1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|-----------------|-----|-----|-----|-----|-----|-----|-----------------|-----------|-------------|-----------------|-------------|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | USB30_PLL_MULTI | | | | | | | Renesas Private | PHYR ESET | REF_C LKDIV | Renesas Private | REF_USB_SEL | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|---|
| 15 to 13 | — | All 0 | R | Reserved |
| 12 to 6 | USB30_PLL_MULTI | All 0 | R/W | Selects reference clock source. USB_XTAL/USB_EXTAL used (50 MHz): B'110_0100 USB3S0_CLK_P/USB3S0_CLK_M used (100 MHz): B'000_0000 |
| 5 | Renesas Private | B'0 | R/W | Do not write 1 to these bits. |
| 4 | PHYRESET | B'0 | R/W | 0: PHY reset is canceled. 1: PHY reset is issued. |
| 3 | REF_CLKDIV | B'0 | R/W | Selects reference clock source. USB_XTAL/USB_EXTAL used (50 MHz): B'1 USB3S0_CLK_P/USB3S0_CLK_M used (100 MHz): B'0 |
| 2, 1 | Renesas Private | B'01 | R/W | Write B'01 to these bits. |
| 0 | REF_USB_SEL | B'1 | R/W | Selects reference clock source. USB_XTAL/USB_EXTAL used (50 MHz): B'0 USB3S0_CLK_P/USB3S0_CLK_M used (100 MHz): B'1 |

63.2.3.3 USB30 SSC Setting Register (USB30_SSC_SET)

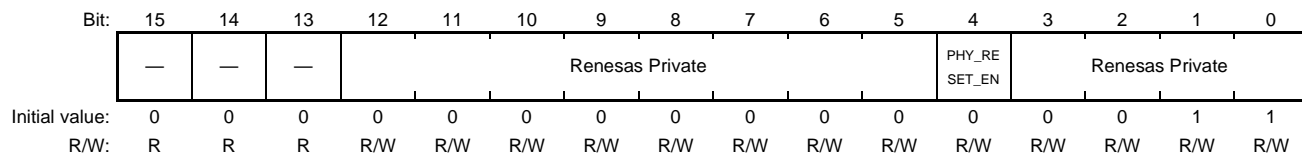
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|--------|-----------|-----|-----|---|---|---|---|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | SSC_EN | SSC_RANGE | | | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------|---------------|-----|--|
| 15 to 13 | — | All 0 | R | Reserved |
| 12 | SSC_EN | B'0 | R/W | Spread Spectrum Enable Enables spread spectrum clock production (0.5% down-spread at ~ 31.5 KHz) in the USB 3.0 PHY. 0: Spread Spectrum Disable 1: Spread Spectrum Enable |
| 11 to 9 | SSC_RANGE | B'000 | R/W | Spectrum Clock Range Selects the range of spread spectrum modulation when SSC_EN is asserted and the PHY is spreading the high-speed transmit clocks. Applies a fixed offset to the phase accumulator. Down Spread of Clock (ppm) B'000: -4,980 B'001: -4,492 B'010: -4,003 |
| 8 to 0 | — | All 0 | R | Reserved |

63.2.3.4 USB30 PHY Enable Register (USB30_PHY_ENABLE)

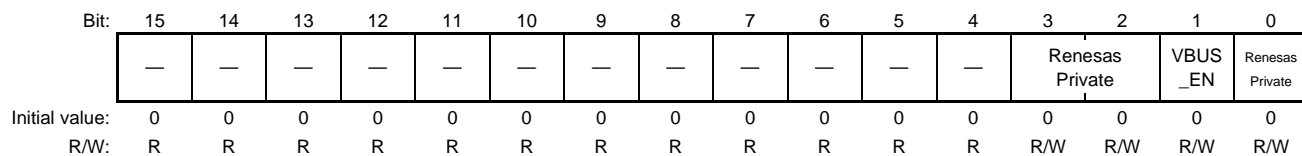
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|-----------------|---------------|-----|---|
| 15 to 13 | — | All 0 | R | Reserved |
| 12 to 5 | Renesas Private | All 0 | R/W | Do not write 1 to these bits. |
| 4 | PHY_RESET_EN | B'0 | R/W | PHY reset enable, when peripheral function 0: USB30_CLKSET1.PHYRESET bit is Disable 1: USB30_CLKSET1.PHYRESET bit is Enable |
| 3 to 0 | Renesas Private | B'0011 | R/W | Prohibit write to these bits. |

63.2.3.5 USB30 VBUS Enable Register (USB30_VBUS_EN)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|---|
| 15 to 4 | — | All 0 | R | Reserved |
| 3 to 2 | Renesas Private | B'00 | R/W | Do not write 1 to this bit. |
| 1 | VBUS_EN | B'0 | R/W | VBUS detect enable, when peripheral function 0: Disable 1: Enable |
| 0 | Renesas Private | B'0 | R/W | Do not write 1 to this bit. |

63.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

63.3.1 Reset

The USB3.0 controller reset diagram is provided below.

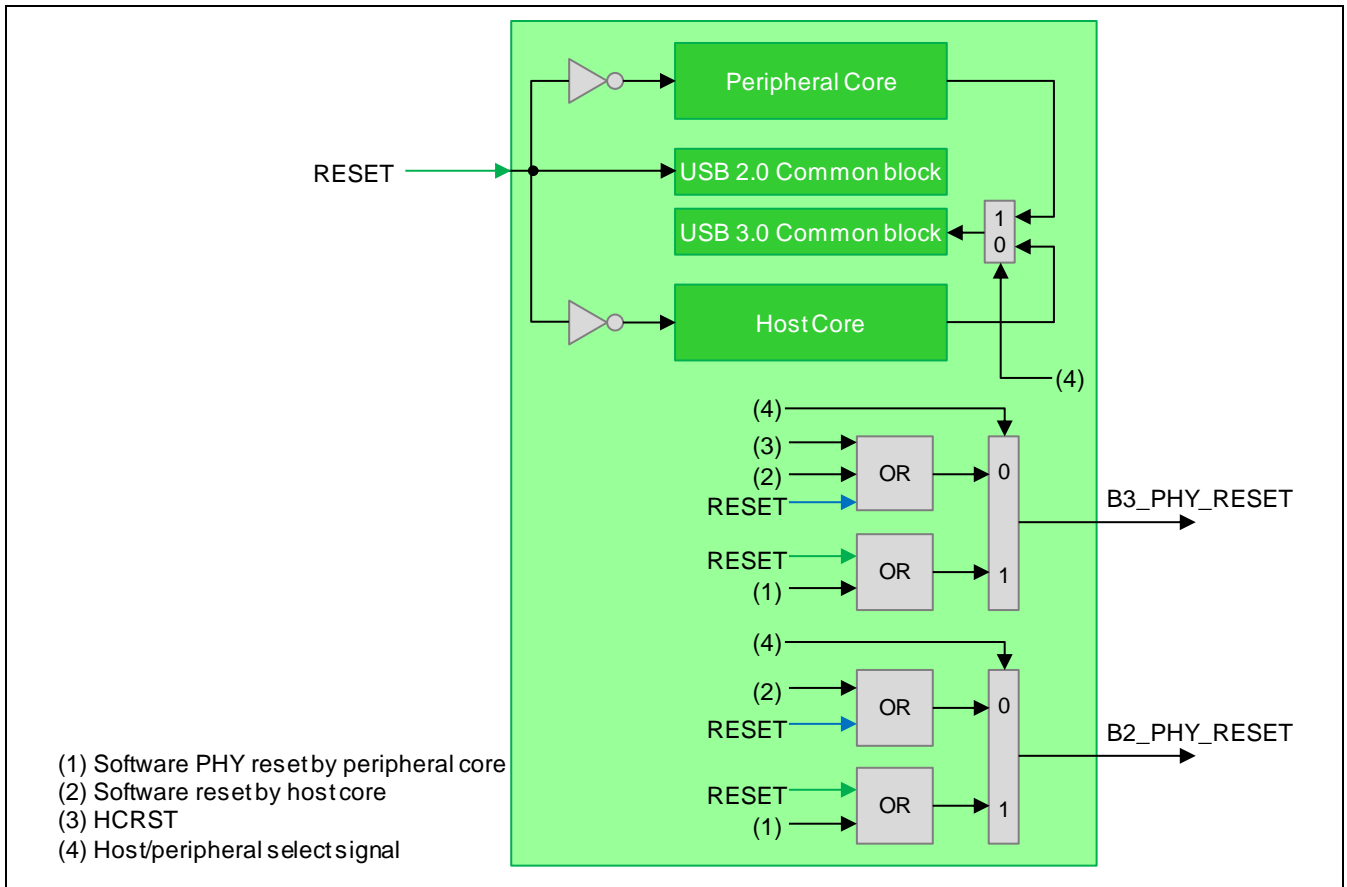


Figure 63.2 Reset Diagram

Note: (2) Software Reset by Host core is output by setting AXH_CON.B3_PHYRST bit.

(3) HCRST, Refer to [HCRST].

RESET in Figure 63.2 is the reset signal from CPG.

63.3.1.1 Host Controller core

(1) Reset Operation

- When the power is ON or USB3.0 module is reset, RESET is asserted.
- The reset is deasserted when USB2.0PLL becomes stable because reset synchronization is cleared.

(2) Reset in Cold Boot

- During cold boot and when the power is ON, RESET is asserted.
- The reset is deasserted when USB2.0 PLL becomes stable because reset synchronization is cleared.

(3) HCRST

- The host controller reset (HCRST) is asserted when the USBCMD.HCRST bit is set to 1 by software, after which the USBSTS.HCH bit (HCHalted bit) will be 0.
- When HCRST is set, Host core is reset by FW, and initialized.

63.3.1.2 Peripheral Controller core

- The Peripheral controller's main Reset is generated by a module reset.
- The Reset is deasserted when AXI clock becomes stable because Reset synchronization is cleared.

63.3.1.3 PHY Reset

B3_PHY_RESET and B2_PHY_RESET are used for a reset of USB3.0 PHY and USB2.0 PHY respectively. Each PHY reset signal is generated by reset signals from the host core. (Refer to Table 63.5)

Table 63.5 B3_PHY_RESET and B2_PHY_RESET

| USB3.0 Controller | Register Setting | Reset Target PHY |
|-------------------|-------------------------|-------------------|
| Host core | AXH_CON.B3_PHYRST = 1 | USB3 PHY is reset |
| | AXH_CON.B2_PHYRST = 1 | USB3 PHY is reset |
| Peripheral core | USB30_CON.B3_PHYRST = 1 | USB3 PHY is reset |
| | USB20_CON.B2_PHYRST = 1 | USB3 PHY is reset |

Table 63.6 Resources of B3_PHY_RESET and B2_PHY_RESET

| | DRD_CON | | RESET (Software Reset) |
|--------------------------|----------------|-------|---------------------------|
| | .PERI_CON | HCRST | |
| B3_PHY_RE SET | 0 (Host) | √ | √ |
| | 1 (Peripheral) | / | √ |
| B2_PHY_RE SET | 0 (Host) | / | √ |
| | 1 (Peripheral) | / | √ |

Note: RESET (Software Reset) shows “RESET” described in Figure 63.2.

The reset signal is selected at the selector by the “Host/Peripheral Select Signal”.

It is controlled by DRD_CON.PERI_CON.

- DRD_CON.PERI_CON = 0, the reset signal from the Host Core is selected.
- DRD_CON.PERI_CON = 1, the reset signal from the Peripheral Core is selected.

63.4 Description of Host Functions

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

63.4.1 FW Download

The USB3.0 controller requires to download FW via AXI to operate as a host.

FW Download is realized by writing FW image data to a specific area in the AXI Register space. Upon the completion of FW Download, there is no need to write or reload FW.

63.4.1.1 Download Data Format

The following format is used for FW Download.

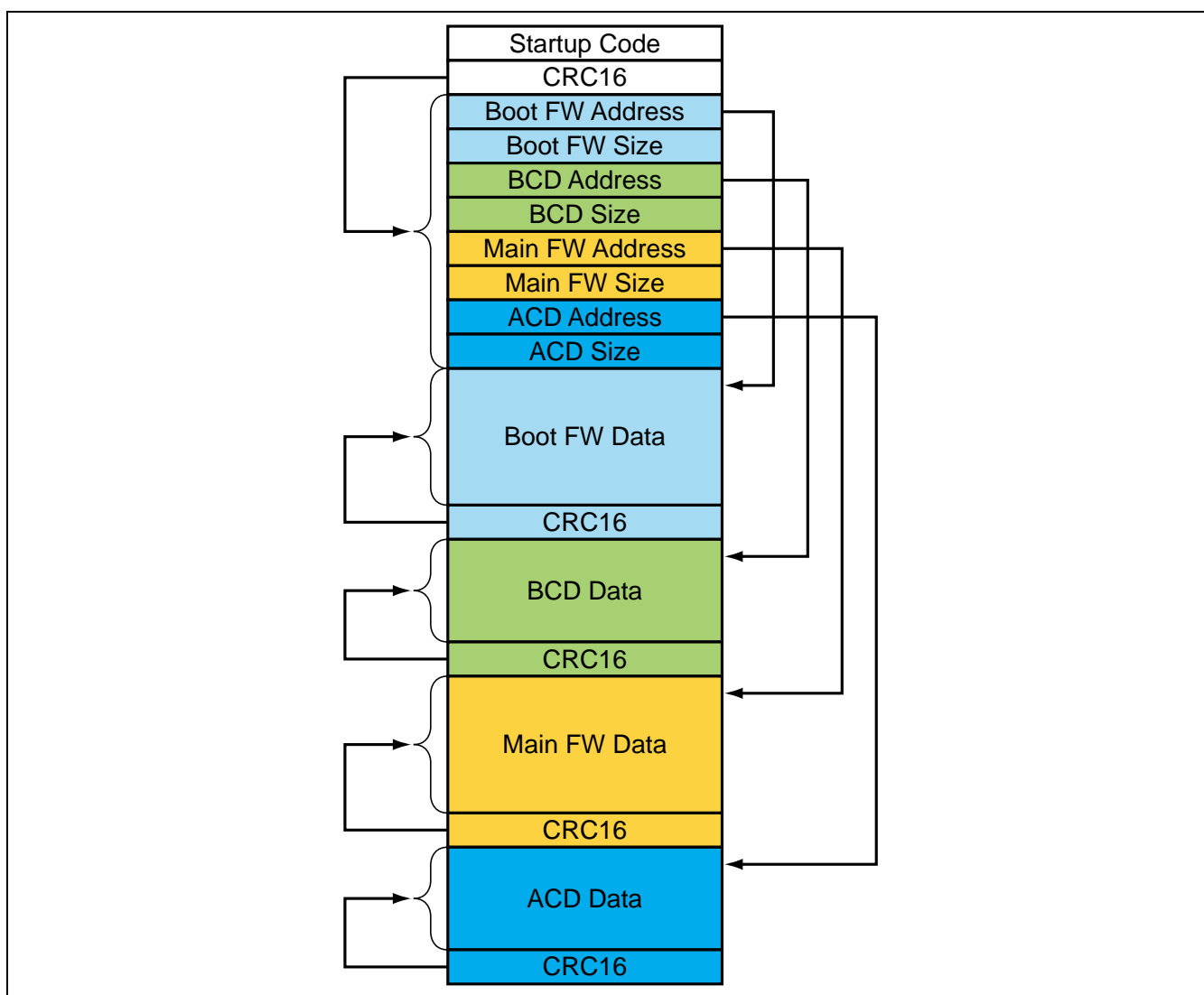


Figure 63.3 The Format of FW Download

| Offset (Byte) | Size (Byte) | Field | Field Explanation |
|---------------------------------|-------------|-----------------|---|
| H'0000 | 2 | Startup Code | Flag bit to show the storing of ROM Data. H'55AA is set. |
| H'0002 | 2 | Header CRC16 | CRC16 value from BCD Address to ACD size. |
| H'0004 | 2 | Boot FW Address | Offset from the beginning of ROM Data in Boot FW Data field. |
| H'0006 | 2 | Boot FW Size | Size of Boot FW field. (L) |
| H'0008 | 2 | BCD Address | Offset from the beginning of Start Code in BCD Data field. |
| H'000A | 2 | BCD Size | Size of BCD Data field. (M) |
| H'000C | 2 | Main FW Address | Offset from the beginning of Startup Code in Main FW Data field. (N) |
| H'000E | 2 | Main FW Size | Size of Main FW field. |
| H'0010 | 2 | ACD Address | Offset from the beginning of Startup Code in ACD Data field. (P) |
| H'0012 | 2 | ACD Size | Size of ACD Data field. |
| H'0014 | L | Boot FW Data | Bootstrap Code |
| (H'0014+L) | l | Boot FW CRC16 | CRC16 of Boot FW Data field. If Boot FW Size field value is 0, it doesn't exist. |
| (H'0014+L+l) | pl | Padding | Padding to assign the beginning of BCD Data to 2-byte boundary. |
| (H'0014+L+l+pl) | M | BCD Data | BCD Data (Boot conf) |
| (H'0014+L+M+l+pl) | m | BCD CRC16 | CRC16 of BCD Data field. If BCD Size field value is 0, it doesn't exist. |
| (H'0014+L+M+l+m+pl) | pm | Padding | Padding to assign the beginning of Main FW Data to 2-byte boundary. |
| (H'0014+L+M+l+m+pl+pm) | N | Main FW Data | Main FW |
| (H'0014+L+M+N+l+m+pl+pm) | n | Main FW CRC16 | CRC16 of Main FW Data field. If Main FW Size field value is 0, it doesn't exist. |
| (H'0014+L+M+N+l+m+n+pl+pm) | pn | Padding | Padding to assign the beginning of ACD to 2-byte boundary. |
| (H'0014+L+M+N+l+m+n+pl+pm+pn) | P | ACD Data | ACD Data (Application conf) |
| (H'0014+L+M+N+P+l+m+n+pl+pm+pn) | p | ACD CRC16 | CRC16 of ACD Data field. If ACD Size field value is 0, it doesn't exist. |

Note: If "l", "m", "n", and "p" exist in each CRC16 field then the value is 2.
If they do not exist, then the value is 0.

63.4.1.2 System Interface

The USB3.0 controller delivers data through the AXI Register.

A register for delivering FW data has no specified data length, so data is always delivered at a 4-byte rate.

If delivered data cannot be allocated to a 4-byte boundary, Host calculates a length of necessary data and discards extra data.

When starting FW Download, the System first sets DL_CTRL.FW Download Enable to 1, and then starts delivering data to the Host.

After writing a value to FW_DATA0 or FW_DATA1, and setting value 1 to the corresponding bit of DL_CTRL.SetData0/1, the System shall put the data delivery on hold until the bit changes to 0.

FW Data0/1 can be used in either of the following ways.

- Use FW Data0 and FW Data1 alternately.
- Use only FW Data0

After writing the last data to FW Data, the System sets DL_CTRL.FW Download Enable to 0 and notifies the Host that Data transfer completes.

After the System makes a notification and confirms that DL_CTRL.Result Code is 1 (Success), the flow completes.

Then if Result Code is 2 (Error), it indicates that Host cannot be used due to some error in FW Download.

(Almost caution is illegally data load.)

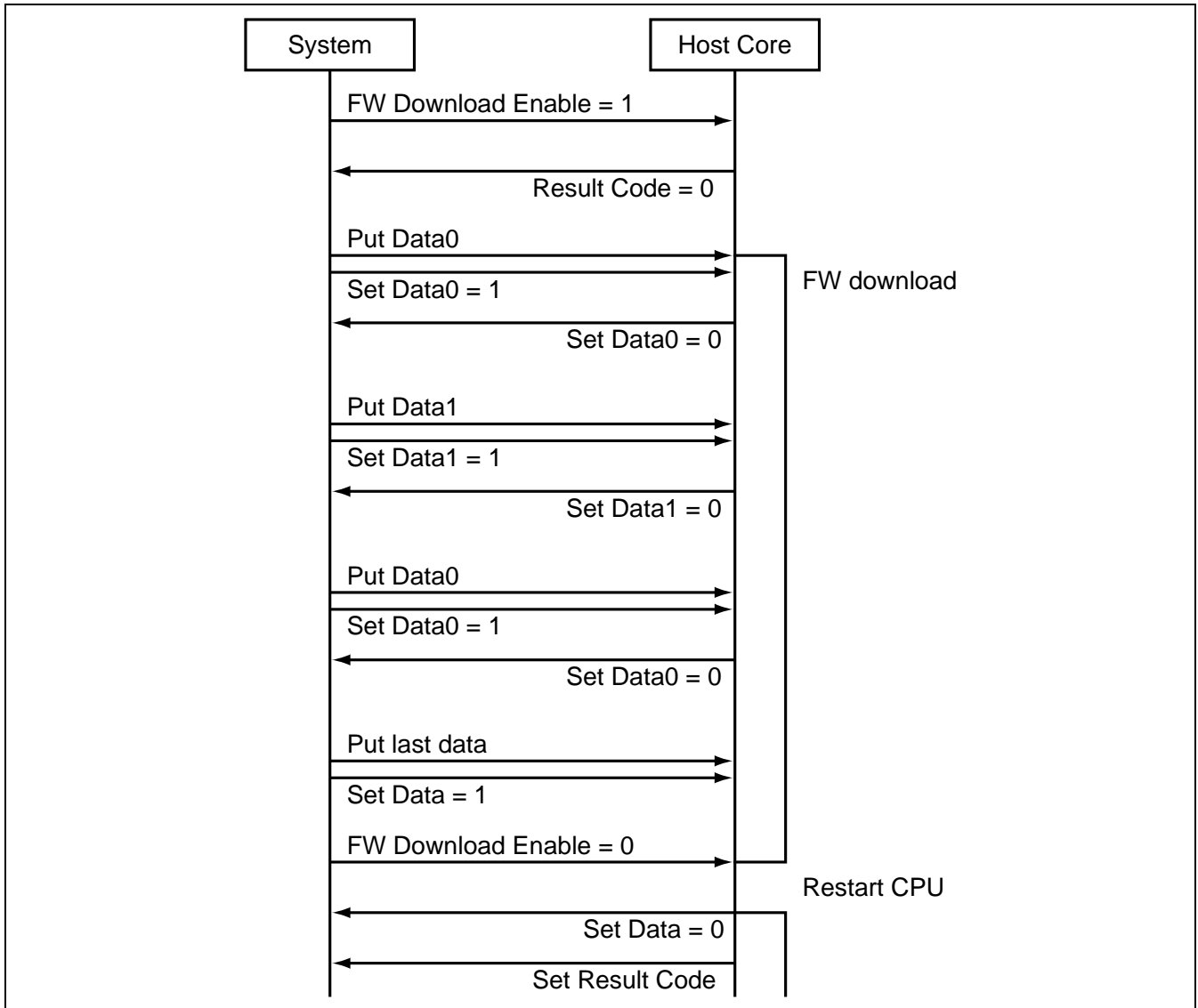


Figure 63.4 The Flow of FW Download

63.4.2 Battery Charging

The USB3.0 controller supports the Battery Charging function of A-Device, which is on the supply side.

In the A-Device, the function is controlled by HW/FW of Host Core.

And it is required that CLK60 (from USB PHY) is working before starting Battery Charging flow.

Note: For details, refer to Battery Charging Specification Revision1.2.

This function is supported in Host.

63.4.2.1 A-Device side (Charging Port Device)

The Battery Charging function is used to pull more current from VBUS than specified in USB specification.

Following Port Modes are supported in A-Device;

- Standard Downstream Port (SDP)
- Charging Downstream Port (CDP)
- Dedicated Charging Port (DCP)

(1) Setting BC_MODE

At first for HW preparing, it shall be set Battery Charging Mode at BC_MODE[3:0] at Battery Charging Register in AXI Register.

(Refer to section 63.2.1.24)

Battery Charging

| | | | | | | | | |
|-------|---|---|---|---|---|---|---|-----------------------|
| H'233 | — | — | — | — | — | — | — | PTPWR_CTRL |
| H'232 | — | — | — | — | — | — | — | Renesas Private field |
| H'231 | — | — | — | — | — | — | — | LCLK_NOSTOP_FREQ |
| H'230 | — | — | — | — | — | — | — | BC_MODE |

Table 63.7 Setting Value for Battery Charging

| Setting Value | BC_MODE | Description |
|---------------|-----------------------------|---|
| 0 | Always SDP Mode | Default value after resetting HW. A-Device operates in the “SDP” mode both normal function state and D3 state. |
| 1 | Always CDP Mode | A-Device operates in the “CDP” mode both normal function state and D3 state. |
| 2 | SDP-DCP Auto-change Mode | A-Device operates in the “SDP” mode in a normal function state and operates in the “DCP” mode in the D3 state. |
| 3 | CDP-DCP Auto-change Mode | A-Device operates in the “CDP” mode in a normal function state and operates in the “DCP” mode in the D3 state. |

Table 63.8 BC Mode and Port Type in D0 and D3 State

| BC_MODE | Normal Operation (D0) Port Type | Suspended (D3) Port Type |
|-----------------------------|------------------------------------|-----------------------------|
| Always SDP Mode | SDP | SDP |
| Always CDP Mode | CDP | CDP |
| SDP-DCP Auto-change Mode | SDP | DCP*1 |
| CDP-DCP Auto-change Mode | CDP | DCP*1 |

Notes: 1. If WakeOnConnect/WakeOnDisconnect is valid, it shall be kept Port type in Normal Operation.

(2) Each case of changing BC_MODE

Case-1) After Host HW Reset, BC_MODE is set “Always SDP Mode”.

SDP is a default Mode.

Red-colored flow in Figure 63.5 is for default setting.

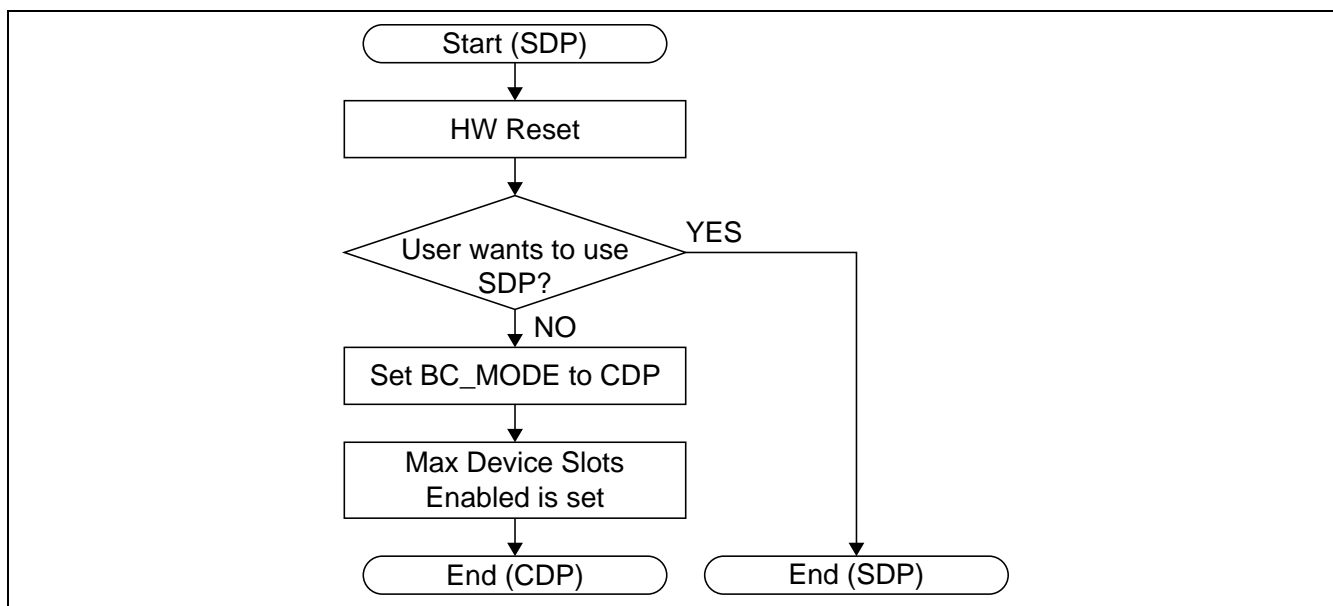


Figure 63.5 Default Setting

Case-2) If user wants to change Mode from SDP/CDP to DCP Mode during A-device working, BC_MODE shall be changed when transition from D0 to D3.

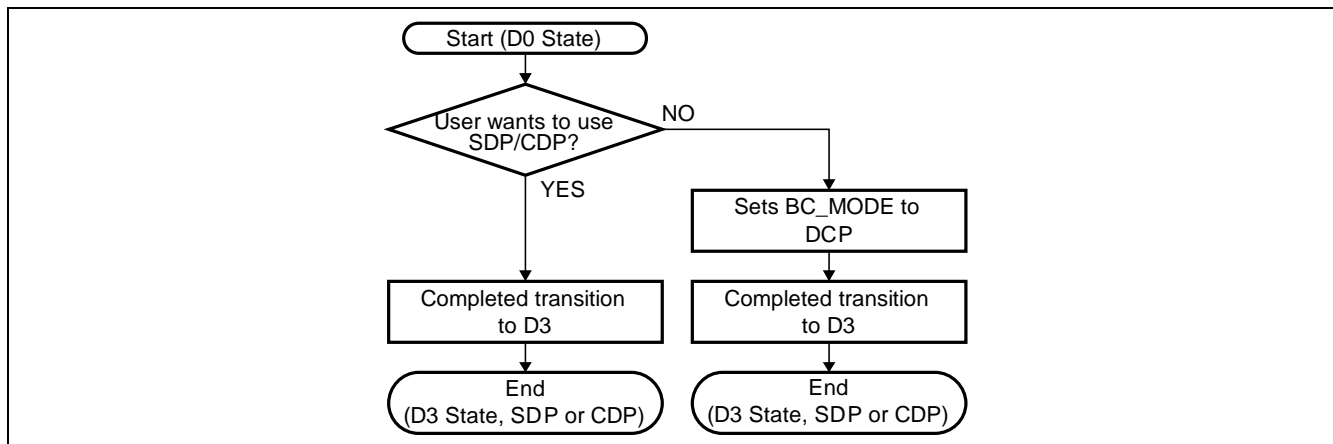


Figure 63.6 Changing BC_MODE during Transition between D0 and D3 States

Case-3) When user wants to use A-Device with except default (SDP) Mode after Host HW Reset, BC_MODE shall be changed before setting value of “Max Device Slots Enabled” in xHCI CONFIG register [H'058].

For the selectable modes, refer to Table 63.8.

Red-colored flow in Figure 63.7 is for BC_MODE setting.

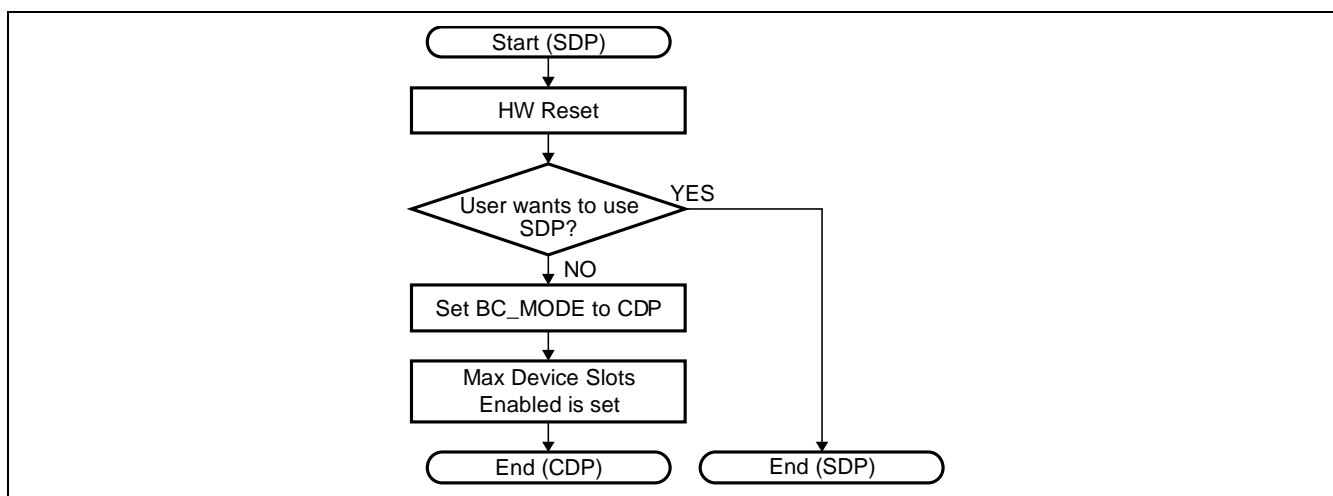


Figure 63.7 Changing BC_MODE

[Caution]

- On the port in which WakeOnConnect/WakeOnDisconnect is enabled, a transition to DCP is not allowed because the port shall detect connect/disconnect of Device.
- When going D3 state with connecting USB device, current Mode shall be kept up setting because that any USB communication doesn't continue with setting DCP.

When changing Battery Charging type, it is required by Battery Charging Specification that VBUS shall be off more than 100-ms and prompting reconnection of device. Host FW operates this process.

63.4.3 Power Management

63.4.3.1 Power Management Capability

USB3.0 controller Host Core sets pseudo D3 state.

It is controlled by xHCI Extended Power Management Capability Register (HCEXTDPMNGCAP2).

- xHCI Extended Power Management Capability (HCEXTDPMNGCAP2).
This register controls pseudo PCIe Power State.
Bit 15: PME Status: Showing which there was a Wakeup event during pseudo D3 state
Bit 8: PME Enable: It asserts an interrupt from pseudo D3 state at Wakeup.
When this bit is set to 1, if PME Status is changed 1 then the interrupt occurs.
Bit [1:0]: Power State: Setting pseudo Power state.
The valid value is as follows.
0: D0 (Do not save power)
3: D3 (Do power save)

Note: In the after section, D3 means pseudo D3 state which is described here.

63.4.3.2 Low Power Management of Receiver Detection

It is intended to saving power of Receiver Detection circuit at PHY Hard Macro (PHYHM).

There are two ways, setting initial value in define file and changing value at field of Register.

For changing value, the field shall be set before starting FW download when occurs after reset.

(1) D3

USB3.0 Common Block operates periodically Receiver Detection with LCLK during RxDetect state of LTSSM and P3 of PHY state.

However if a system is in Suspend and WakeOnConnect is disabled, there is no need to do Receiver Detection.

To reduce power consumption of PHY by above discarded Receiver Detection, FW can halt the process.

(2) D0

Receiver Detection at opened Port in D0 shall be operated.

HW operates reducing power at interval of Receiver Detection.

TimeMax is that decides operational interval of Detection in P3.

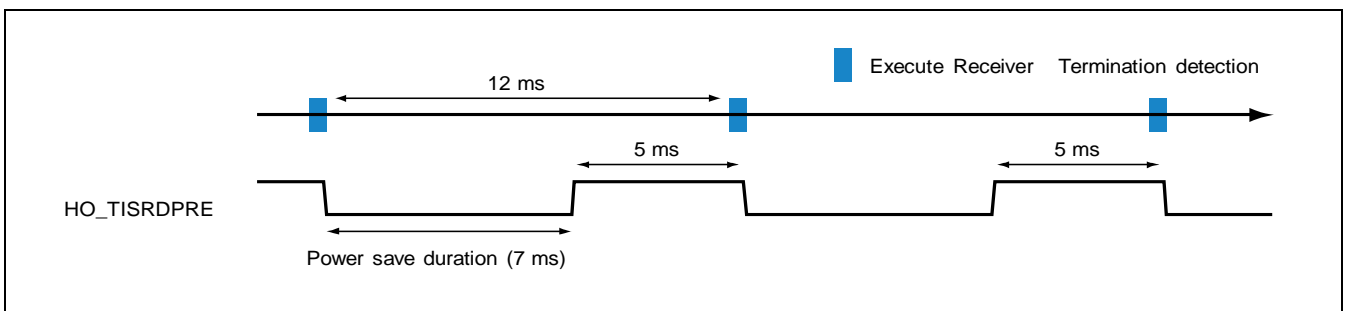
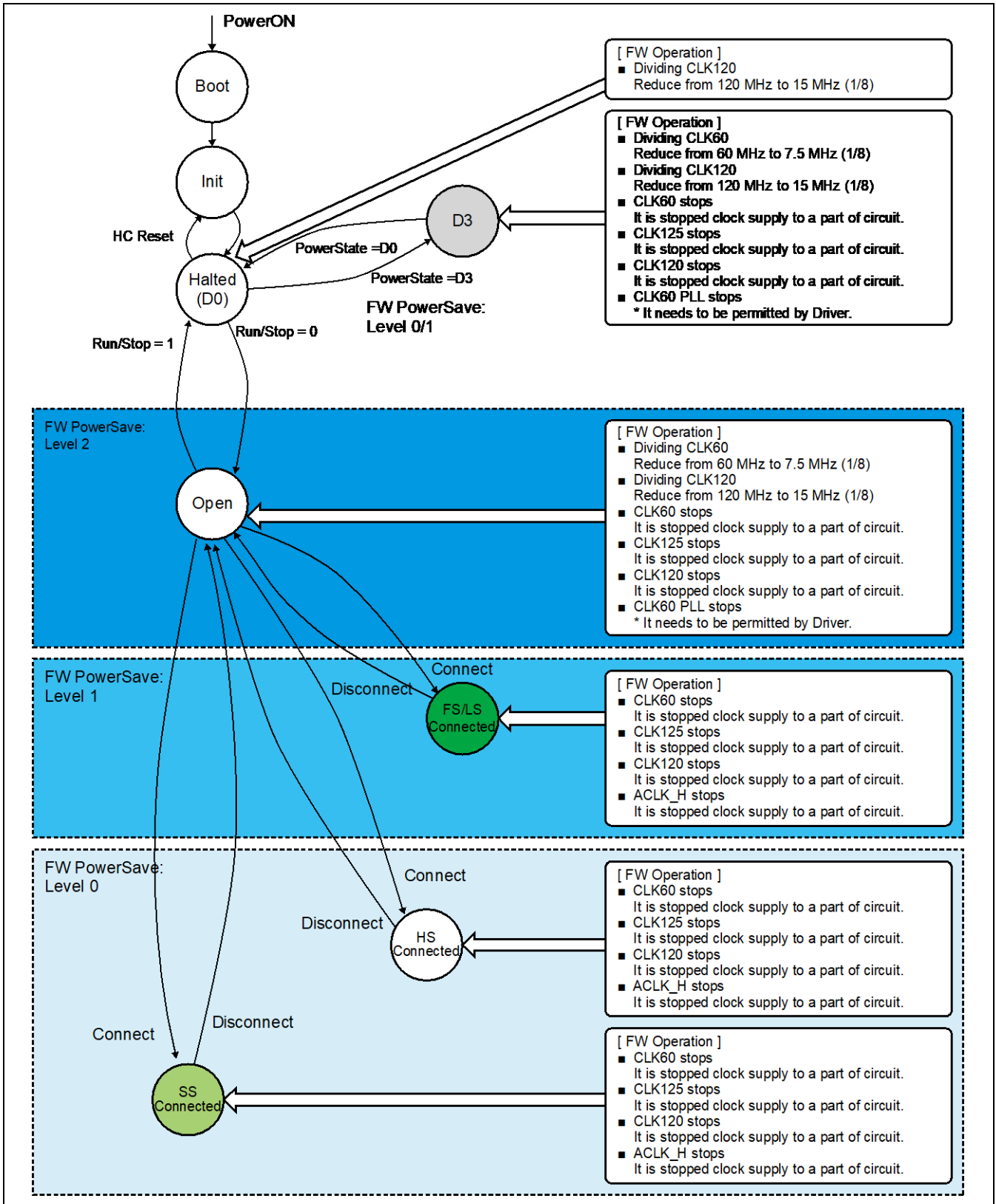


Figure 63.8 Low-Power of Receiver Detection

63.4.3.3 Power Control for Each State

The overview of power control state flow is shown as below.



Note: CLK60/CLK120/CLK125 (from USB PHY)

Figure 63.9 Overview of Power Control State Flow

63.4.3.4 Clock Termination by USB3PHY and LTSSM State

USB3PHY may stop PLL which generates CLK125 by USB3PHY's power state.

Table 63.9 LTSSM State and USB3PHY Condition

| LTSSM State | USB3PHY | | Supplement |
|-----------------------|-----------|---------------|--|
| | POWERDOWN | PLL Operation | |
| SS.Disabled | P2 | Run | |
| Rx.Detect | P2 | Run | Default |
| | P0 | Run | |
| | P3 | Run | |
| U0 | P0 | Run | |
| U1 | P1 | Run | |
| U2 | P2 | Run | |
| U3 | P0 | Run | |
| | P3 | Run | |
| U3 → U3_Wakeup, Reset | P0 | Run | Transition by receiving LFPS |
| U3 → RemoteWakeup | P0 | Run | Transition is required by PLS (Need Register operation) |
| Other | P0 | Run | |

63.4.3.5 Flow of System Suspend and Resume

When following all permission for suspend are met, system goes to suspend.

- Host system sets AXH_CON.SYS_SUSPEND to 1 and completes host core's preparations for suspend.

When host core resumes from System Suspend, user shall confirm the bit of B2_PLL_ACTIVE at AXH_STA register [H'104 to H'107] in AXI register is set to 1 before accessing register.

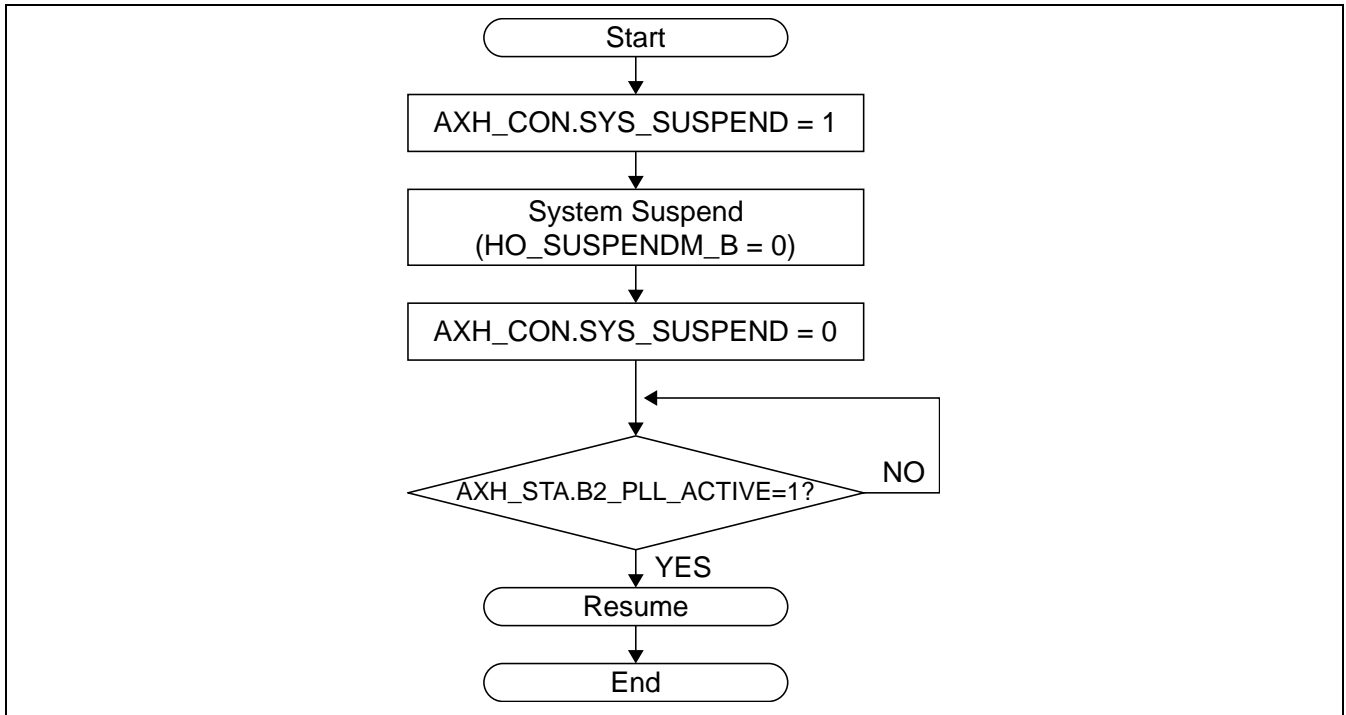


Figure 63.10 System Suspend and Resume Flow

63.4.4 Power Control

The process of VBUS ON/OFF and Over Current detection are described below.

63.4.4.1 VBUS On

When VBUS will be on, at first host driver shall execute processes as below.

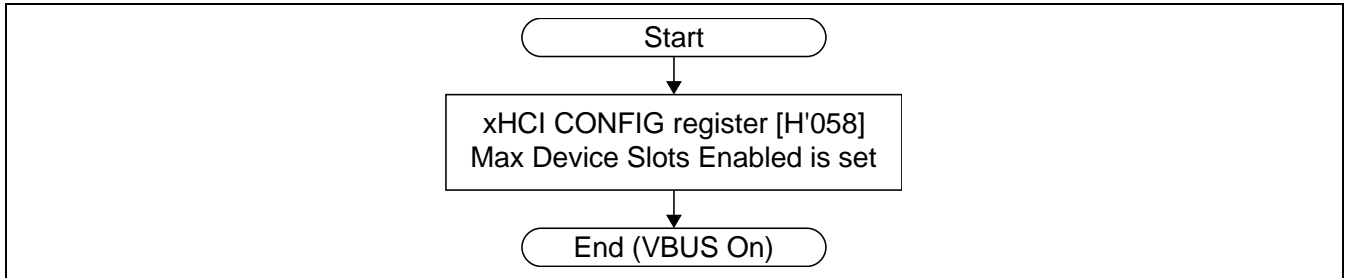


Figure 63.11 Flow of VBUS On

63.4.4.2 VBUS Off

When port power will be off, at first host driver shall execute processes as below.

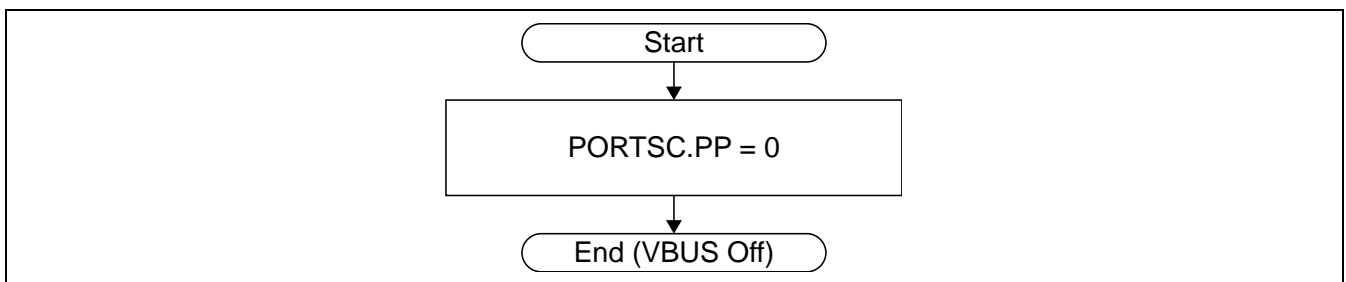


Figure 63.12 Flow of VBUS Off

63.4.4.3 Over Current Detection

When port will detect over current during the core working, following flow is executed on host.

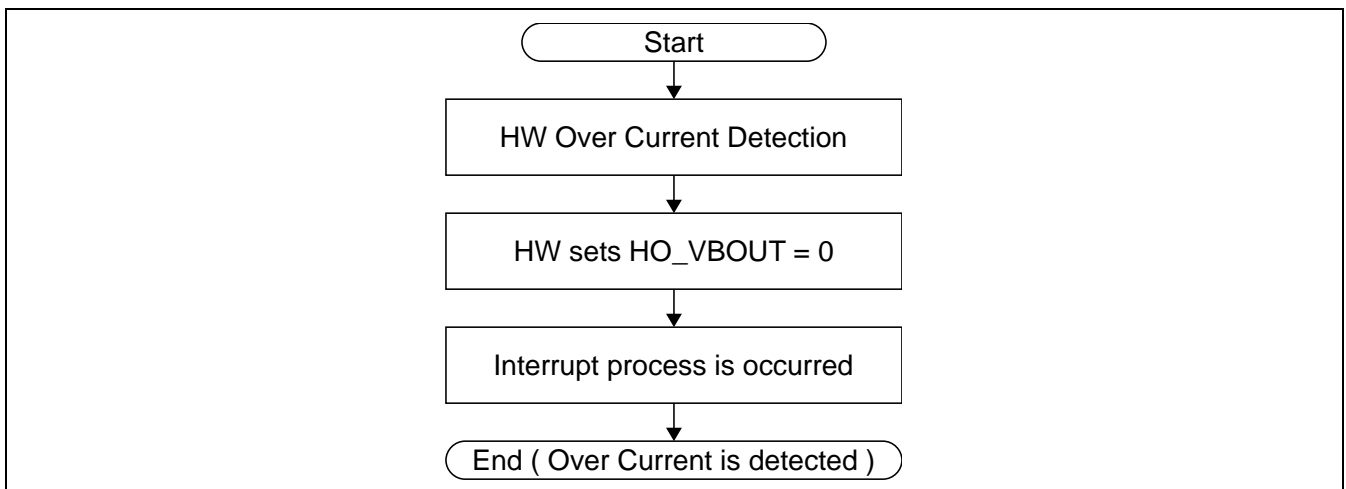


Figure 63.13 Flow of Over Current Detection

63.4.5 Structure of Notification for Elements of Interrupt

In USB3.0 controller, the elements of interrupt are noticed by following layered structure.

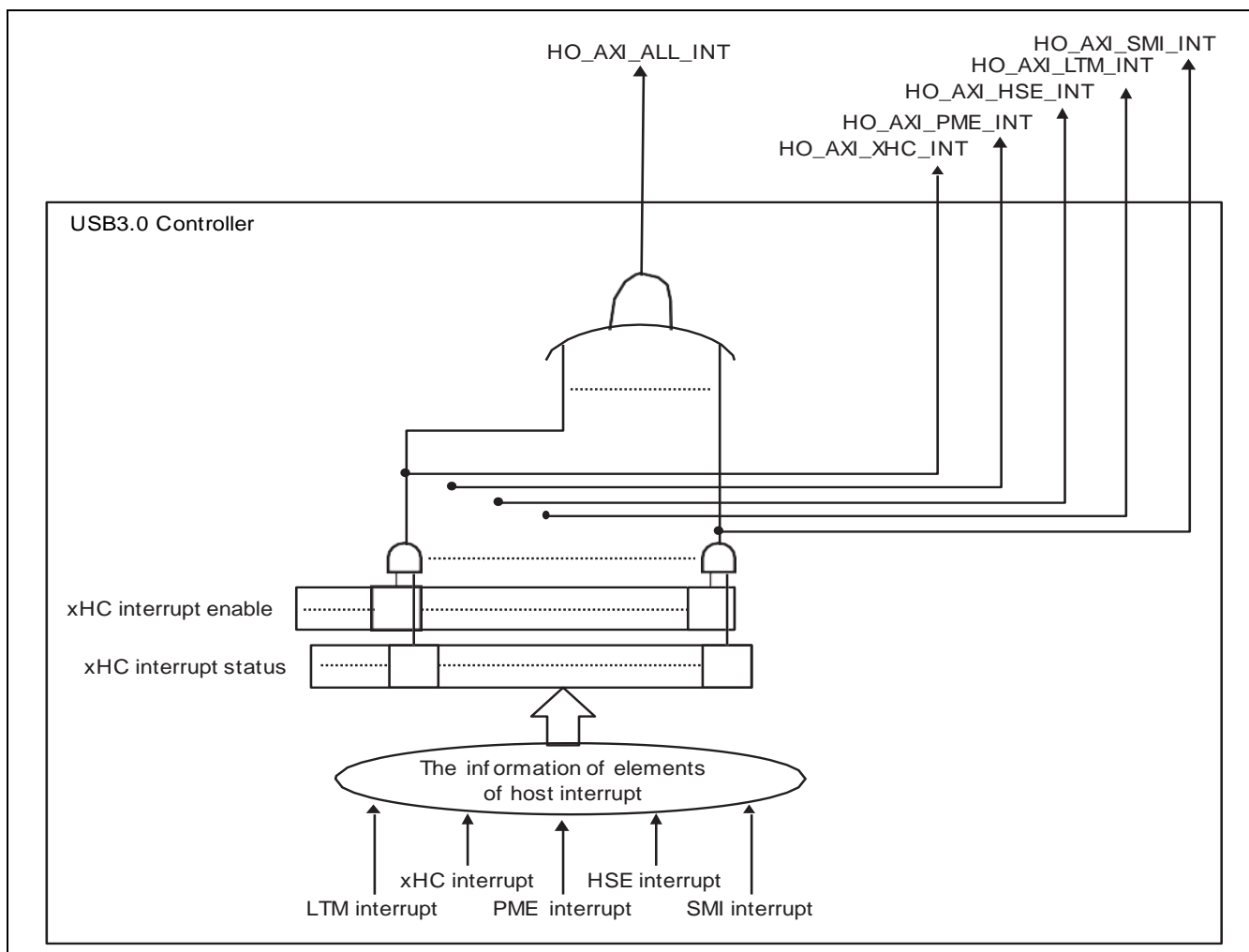


Figure 63.14 Interrupt Layered Structure

- This Host core doesn't use WAKEB but PME. Interrupt pin by PME is HO_AXI_INT_PME.

Note: About occurrence factor of PME interrupt, refer to xHCI Specification.

- Interrupt occurrence factor of xHCI shall be switched its meaning as PCIINT.
- The abstract of Interrupt signals from Figure 63.14 are below.

Table 63.10 Function of Interrupt Signals

| Signal Name | Function |
|----------------|---------------------------------------|
| HO_AXI_INT_XHC | xHCI Interrupt |
| HO_AXI_INT_PME | PME Interrupt |
| HO_AXI_INT_HSE | HSE Interrupt |
| HO_AXI_INT_LTM | LTM Interrupt |
| HO_AXI_INT_SMI | SMI Interrupt |
| HO_AXI_INT_ALL | All Interrupt Signals from Host Core. |

63.4.6 Implementation Constraint of Host Core

The USB3.0 controller core has implementation constraint.

- 1 Packet shall be composed of 8 TRB or less.

63.5 Example of Transaction

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

This section describes transaction when using Host core. As examples, Bulk OUT and Bulk IN transactions are described. The USB3.0 controller core complies with xHCI Specification.

Refer to xHCI Specification for details.

63.5.1 Bulk OUT Transaction

Bulk OUT transaction flow is shown as below.

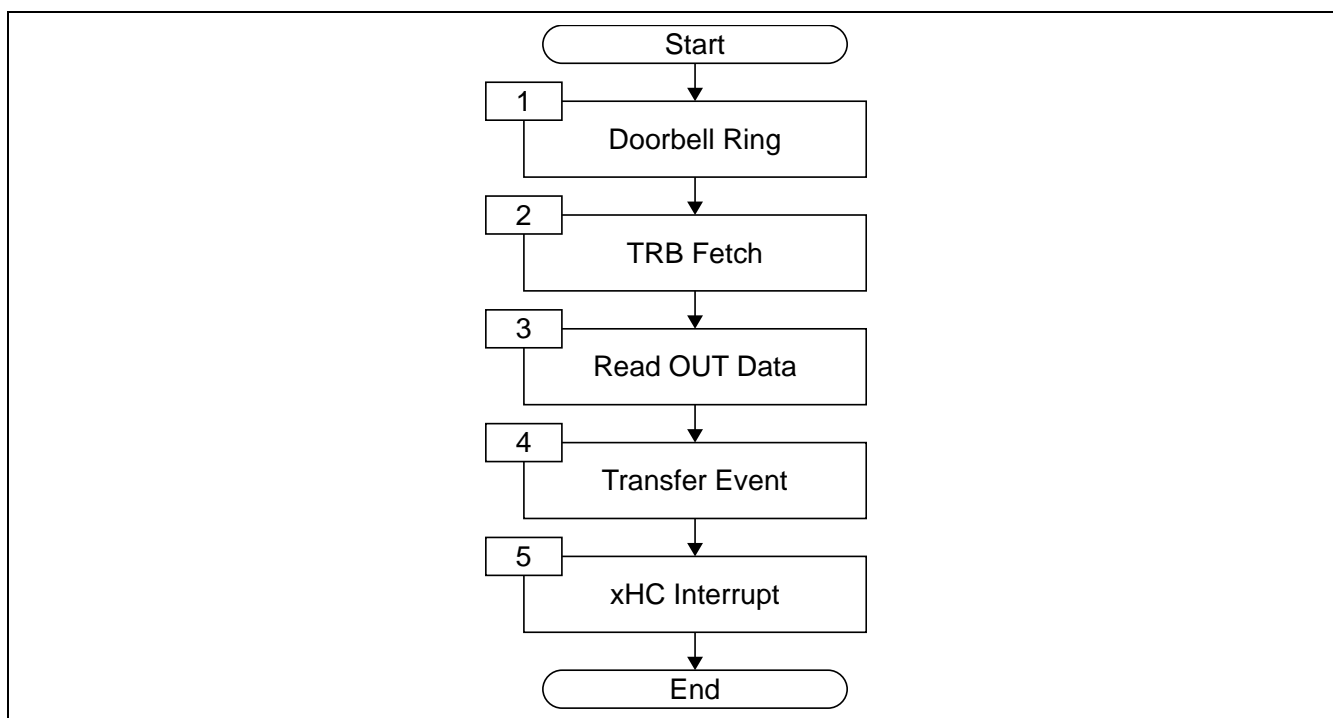


Figure 63.15 Example of Bulk OUT Transaction Flow

- 1) Doorbell Ring
System rings Doorbell to prepare Transfer Data.
- 2) TRB Fetch
Host core fetches TRB from OUT Transfer Ring,
- 3) Read OUT Data
Host reads OUT Data from Data Buffer Pointer.
- 4) Transfer Event
Host writes Transfer Event TRB to Event Ring.
- 5) xHC Interrupt
Host sends Interrupt to System.

63.5.2 Bulk IN Transaction

Bulk IN transaction flow is shown as below.

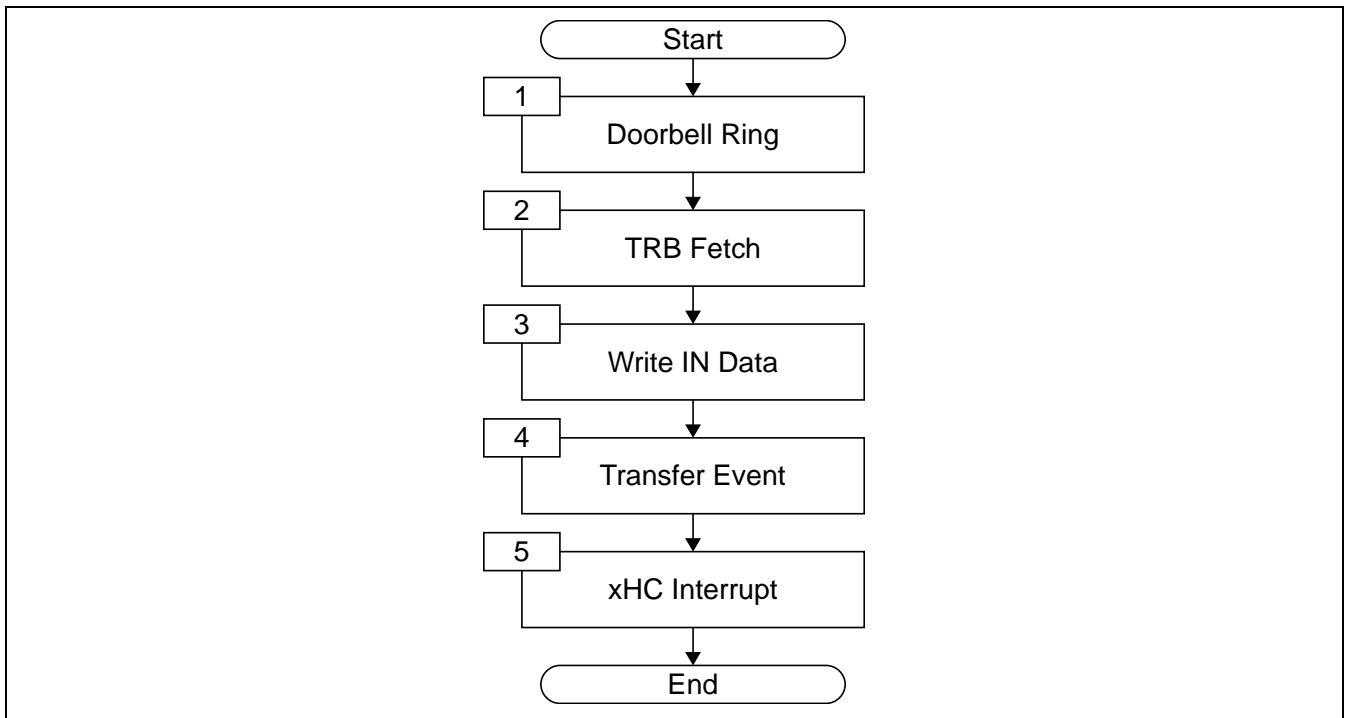


Figure 63.16 Example of Bulk IN Transaction Flow

- 1) Doorbell Ring
System rings Doorbell to prepare Transfer Data.
- 2) TRB Fetch
Host core fetches TRB from OUT Transfer Ring,
- 3) Write IN Data
Host writes IN Data to Data Buffer Pointer.
- 4) Transfer Event
Host writes Transfer Event TRB to Event Ring.
- 5) xHC Interrupt
Host sends Interrupt to System.

63.6 Power IC Interface Connection

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The USB3.0 controller core has power signal detection.

- Power IC has Outputting Over Current Flag signal and the USB3.0 controller can detect the signal.
- Battery Charging function is able to use at the USB3.0 controller.

63.6.1 Usable Power IC type

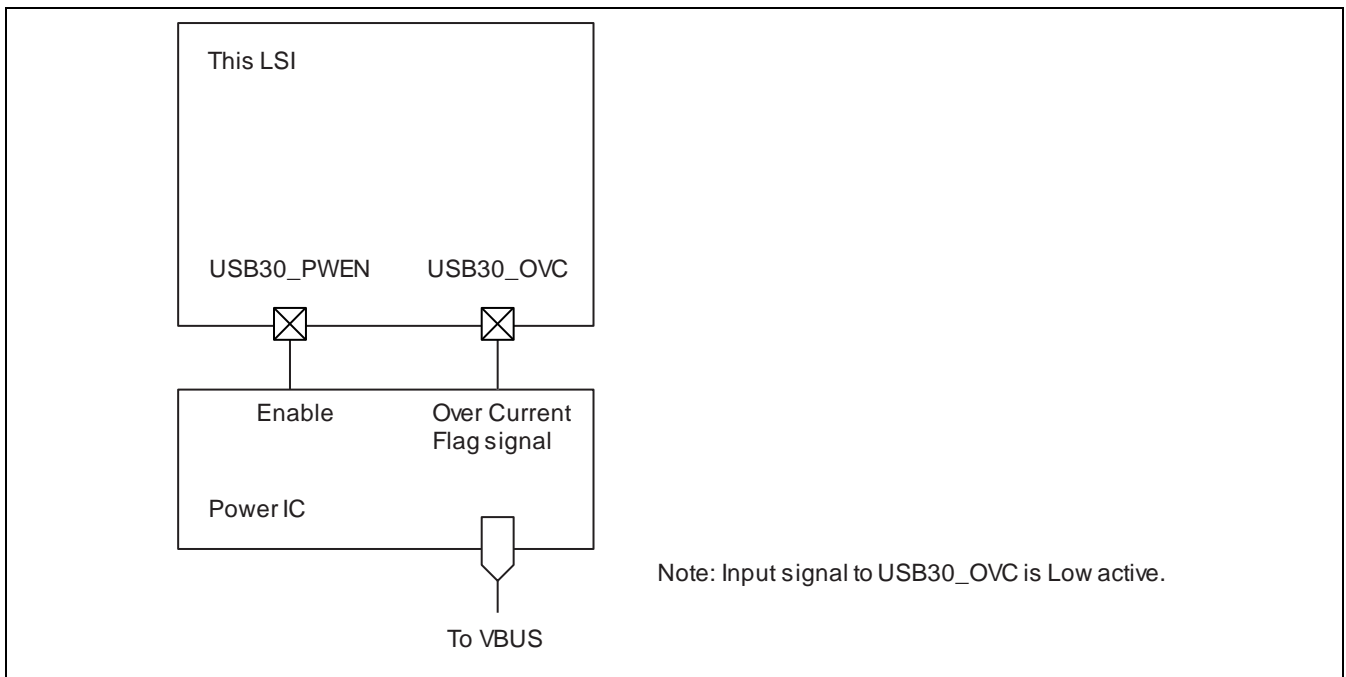


Figure 63.17 Usable Power IC Type

Table 63.11 Setting Registers and Signals for Power IC

| Signal | Description |
|------------|--|
| USB30_OVC | Connecting to Over Current Flag signal |
| USB30_PWEN | Connecting to Enable |

- **Restriction**
Do not use another type of Power IC.

63.7 Descriptions of functions

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

63.7.1 VBUS Connection

The overview of connection of VBUS signal is shown in Figure 63.18.

VBUS is the power supply signal from USB host or hub, device can know that it is connected to host or hub when VBUS is asserted. But since VBUS input might be unstable just after device is connected, device should wait for a certain time till it becomes stable especially if it is bus-powered device. In order to control the interval from the assertion of VBUS to the time VBUS becomes stable, VBUS signal is not directly connected to link layer but USB30_CON.B3_CONNECT is connected to it instead. The value of the bit is output to link layer which sees the value to know whether it is being connected or not. This mechanism enables that firmware inserts the wait time between when it detects the assertion of VBUS by USB_INT_STA_1.VBUS_CNG_STA and when it sets USB30_CON.B3_CONNECT.

In the same way, VBUS signal is not directly connected to UTMI2SIE and USB2.0-PHY but USB20_CON.B2_CONNECT is used to notify them that device is connected.

Note: VBUS or USB30_CON.B3_CONNECT is not provided directly to USB3.0-PHY. USB3.0-PHY provided from Renesas is designed not to see the state of VBUS but to follow the instruction of power down from upper layer on POWERDOWN[1:0] for its power management. But some USB2.0-PHYs provided from Renesas are designed to see the state of VBUS and require that VBUS or similar signal is connected. See the specification of USB-PHY for details.

When device is in suspended state or unconnected, PLLs in USB3.0-PHY and USB2.0-PHY can be stopped in order to reduce their power consumptions. But if the device is disconnected when it is in suspended state, it means that the request to wake up from host never comes any more. In such case, it might be required to resume PLLs if they have been stopped so that device initializes by itself and prepares for the next connection. On the other side, device might be required to resume PLLs if they have been stopped in unconnected state but device detects the connection. The state of connection or disconnection can be known on VBUS state. The change of VBUS state is notified on USB_INT_STA_1.VBUS_CHG_STA. As shown in the cases above, the software is required to control PLLs in USB3.0-PHY and USB2.0-PHY depending on VBUS state.

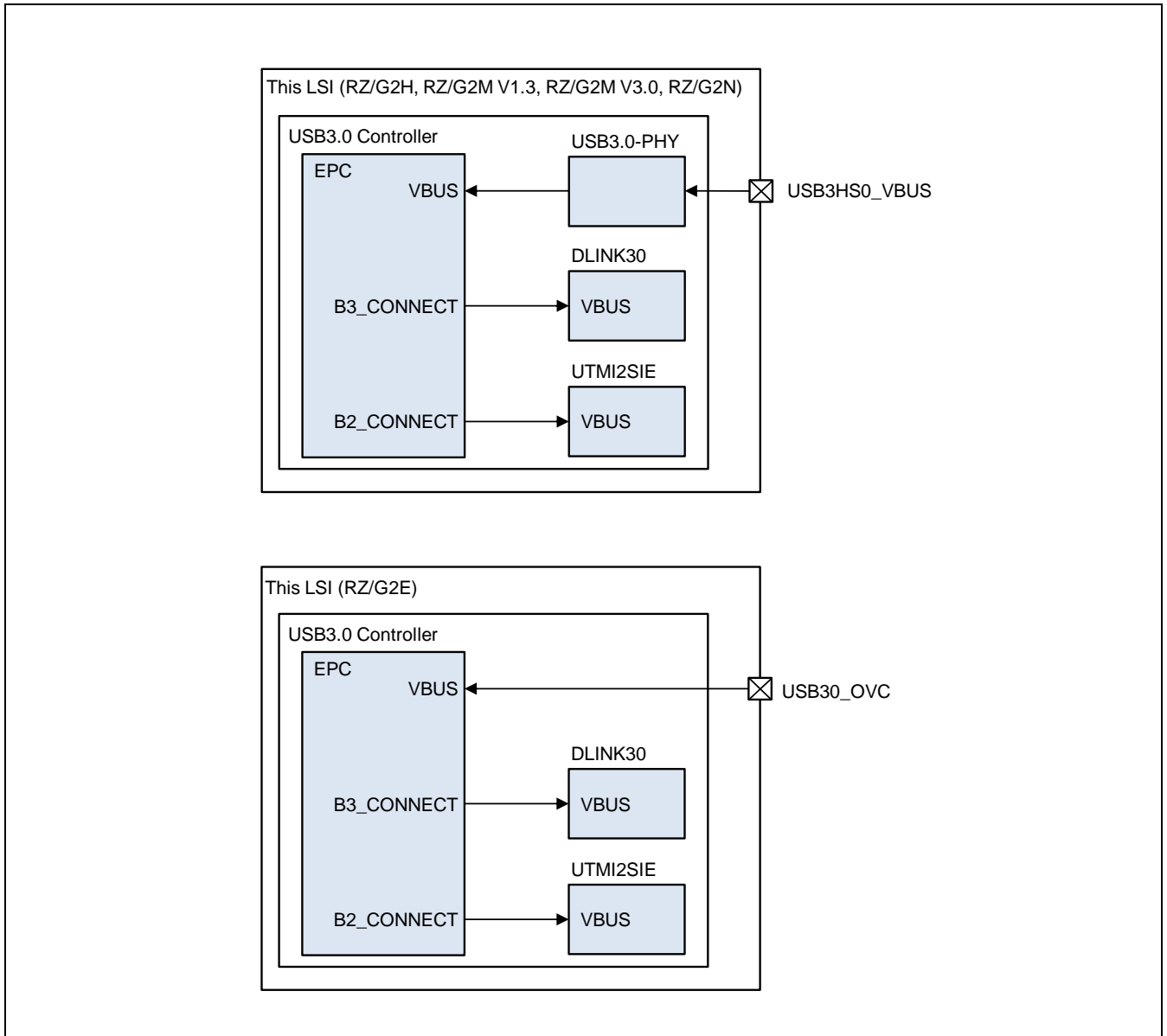


Figure 63.18 VBUS connection

63.7.2 Functions of EPC

63.7.2.1 Buffering for transmission and reception on USB

- Buffer for the received data at OUT endpoint
 - (1) PERIPHERAL PORTION receives OUT packet from host.
 - (2) PERIPHERAL PORTION stores the received packet in RAM if it has an available space.
 - (3) PERIPHERAL PORTION checks whether the received packet has an error (CRC error, PID error, sequence number error etc.) or not.
 If PERIPHERAL PORTION completes the reception of OUT packets and finds no error in them, PERIPHERAL PORTION informs Interrupt Controller with an interrupt signal.
 PERIPHERAL PORTION sends the received packets to UDL side if required.
 - (4) AXI-bus reads the received packets

- Buffer for data to be transmitted at IN endpoint
 - (1) PERIPHERAL PORTION receives IN token (USB2.0) or ACK TP (USB3.0) from host.
 - (2) PERIPHERAL PORTION sends data packet if it has data in RAM.
 - (3) PERIPHERAL PORTION checks whether it receives handshake (ACK, for example) for transmitted packet or not.
 If PERIPHERAL PORTION completes the transmission of IN packets and receives ACKs from host for them, PERIPHERAL PORTION informs Interrupt Controller with an interrupt signal.
 If PERIPHERAL PORTION can't send IN packet correctly or can't receive ACK from host, PERIPHERAL PORTION transmits the same data again when next IN token (USB2.0) or ACK TP (USB3.0) comes.
 - (4) AXI-bus can write data for the next transmission if it wants.

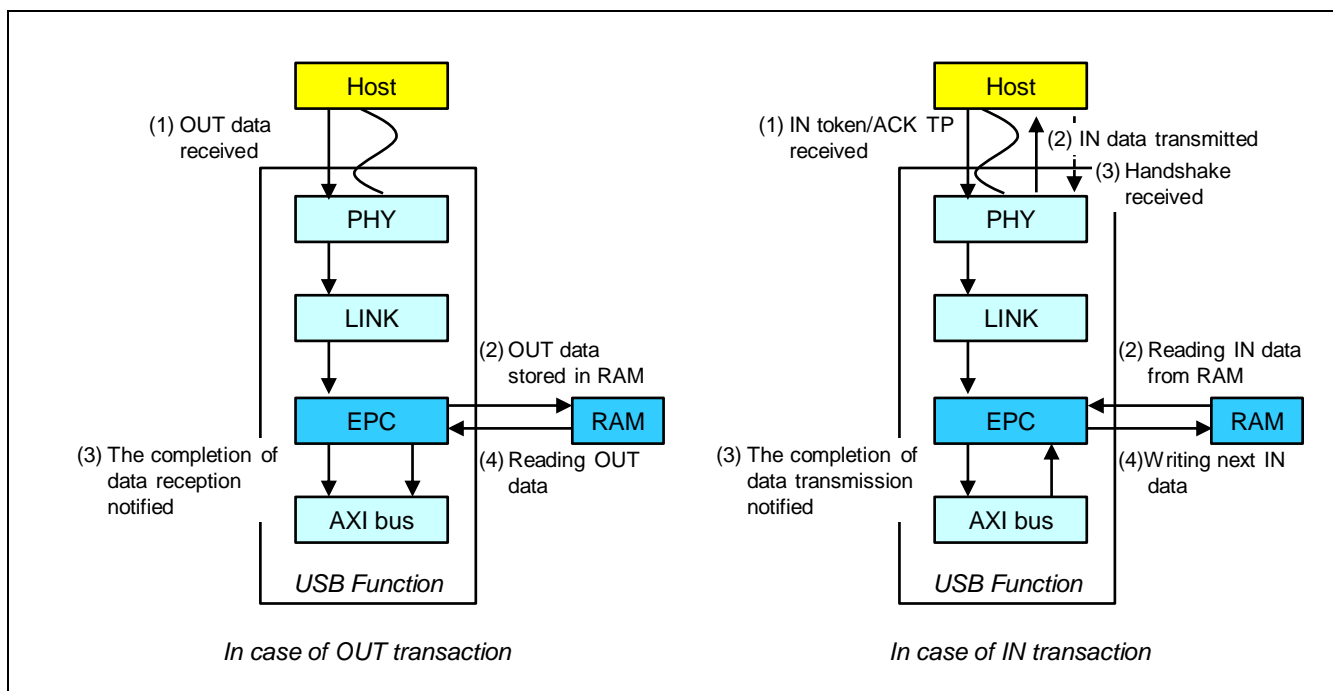


Figure 63.19 buffer for transmission and reception on USB

63.7.2.2 Handshake

Peripheral portion automatically selects the response depending on the setting, its internal status and the receiving status of the packet.

(1) Handshake response for OUT transaction

Peripheral portion sends the handshake response for OUT transaction as shown in Table 63.12 and Table 63.13.

Table 63.12 Handshake for OUT transactions (HS/FS)

| Reception Data Corrupted (Error) | Endpoint Halt | Data Toggle Check | Data Reception Allowed/Not Allowed | Handshake |
|----------------------------------|---------------|-------------------|------------------------------------|-------------|
| Yes | N/A | N/A | N/A | No Response |
| No | Set | N/A | N/A | STALL |
| No | Not Set | NG | N/A | ACK |
| No | Not Set | OK | Allowed | ACK |
| No | Not Set | OK | Not Allowed | NAK |

Table 63.13 Handshake for OUT transaction (SS)

| Invalid DPH Reception | Deferred Reception | Endpoint Halt | Reception Data Corrupted (Error) | Data Reception Allowed or Not allowed | Handshake |
|-----------------------|--------------------|---------------|----------------------------------|---------------------------------------|----------------------------------|
| Yes | N/A | N/A | N/A | N/A | No Response |
| No | Yes | Yes | N/A | N/A | ERDY |
| No | Yes | No | N/A | Not allowed | ERDY when PIPE becomes available |
| No | Yes | No | N/A | Allowed | ERDY |
| No | No | Yes | N/A | N/A | STALL |
| No | No | No | N/A | Not allowed | NRDY |
| No | No | No | Yes | Allowed | ACK (Retry required) |
| No | No | No | No | Allowed | ACK |

(2) Handshake response for IN transaction

Peripheral portion sends the handshake response for IN transaction as shown in Table 63.14 and Table 63.15.

Table 63.14 Handshake for IN transaction (HS/FS)

| Token Corrupted (Error) | Endpoint Halt | Transmission Data is in buffer or not | Handshake |
|--------------------------------|----------------------|--|-------------------|
| Yes | N/A | N/A | No Response |
| No | Set | N/A | STALL |
| No | Not Set | No | NAK |
| No | Not Set | Yes | Data Transmission |

Table 63.15 Handshake for IN transaction (SS)

| Invalid TP Reception | Deferred Reception | Endpoint Halt | Transmission Data is in buffer or not | Handshake |
|-----------------------------|---------------------------|----------------------|--|----------------------------------|
| Yes | N/A | N/A | N/A | No Response |
| No | Yes | Yes | N/A | ERDY |
| No | Yes | No | No | ERDY when PIPE becomes available |
| No | Yes | No | Yes | ERDY |
| No | No | Yes | N/A | STALL |
| No | No | No | No | NRDY |
| No | No | No | Yes | Data Transmission |

63.7.3 USB3.0/USB2.0 connection

In this section the outline of USB3.0 and USB2.0 connection in USB3.0 specification is explained.

See section 63.8.3.6 for details of the software flow.

The behavior of peripheral device upstream port state machine is described in section 10.16.2 of USB3.0 specification Revision 1.0 June 6, 2011. This behavior was proposed as ECN#011 at first.

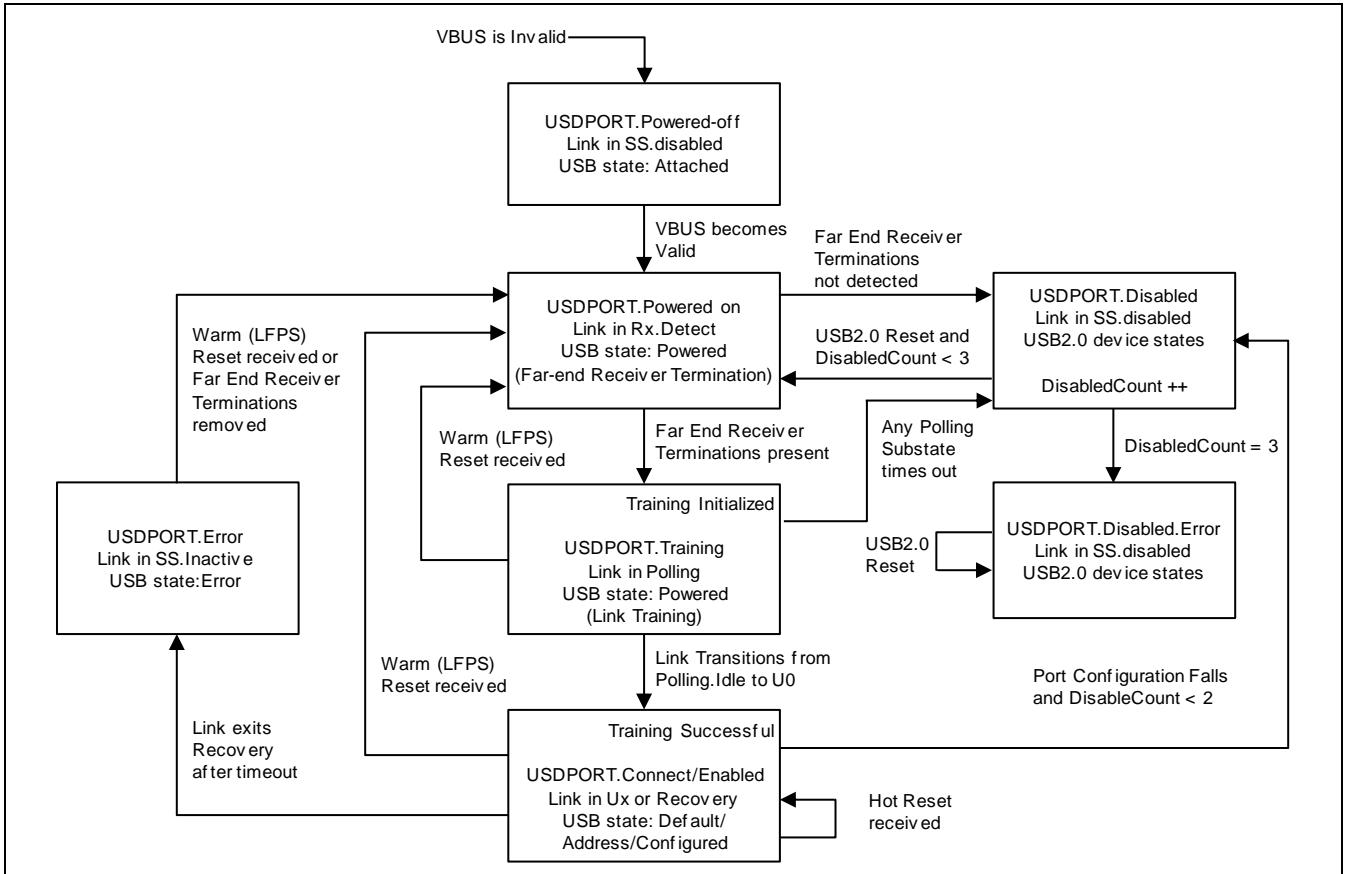


Figure 63.20 Peripheral Upstream Device Port State Machine (from Figure10-25 in USB3.0 specification)

As initial state, in which VBUS is off, the upstream port of the device is in USDPORT.Powered-off. The port should have the variable “DisableCount” to control the transition to or from USDPORT.Disabled. The DisabledCount should be initialized to 0 when power is off or when the port’s link completes the port configuration.

When VBUS becomes valid (on), the port transits to USDPORT.Powered on. In this state, the port’s link is in Rx.Detect. On this first entry to USDPORT.Powered on, the port’s link makes the trial of Rx detection up to 8. If it detects Far End Receiver Termination of the link partner, the upstream port enters USDPORT.Training, and after the training has successfully completed it enters USDPORT.Connected/Enabled. But if the port fails in the port configuration, it enters USDPORT.Disabled.

But if the port’s link can’t detect Far End Receiver Termination even when it tries 8 times, the port enters USDPORT.Disabled. With this transition DiableCount is incremented if it is less than 3. In this state, the function of

USB3.0 is disabled once and the function of USB2.0 starts working.

If downstream port detects D+ line is pulled up, it recognizes the device wants to be connected as USB2.0 device. The downstream port asserts USB2.0 bus reset to start the process of USB2.0 connection.

Receiving USB2.0 bus reset, the upstream port returns to USDPORT.Powered on. On this entry to USDPORT.Powered on, the function of USB3.0 is enabled again and the port's link makes the trial of Rx detection only one time. Note that the pull-up of D+ line is still enabled and the process of USB2.0 connection is still valid.

If the port's link detects Far End Receiver Termination of the link partner, the upstream port enters USDPORT.Training and the connection as USB2.0 device is disabled within tUSB2SwitchDisconnect (=1ms). If the port succeeds in the training, it enters USDPORT.Connected/Enabled.

But if the port's link can't detect Far End Receiver Termination, the function of USB3.0 is disabled again. When the port succeeds in USB2.0 connection, the device works in HS or FS of USB2.0. With this failure of Rx detection, the port enters USDPORT.Disabled as USB3.0 port and DisabledCount is incremented if it is less than 3.

If DisabledCount equals to 3 when the port enters USDPORT.Disabled, it makes the transition to USDPORT.Disabled_Error.

Whenever the port is in USDPORT.Disabled and USB2.0 bus reset comes, the port transits to USDPORT.Powered on and Rx detection for USB3.0 connection is started one time. Pull up of D+ can be seen from downstream port and USB2.0 connection is still active.

But if the port is in USDPORT.Disabled_Error when USB2.0 bus reset comes, it tries Rx detection no more and can only executes USB2.0 connection. Since DisabledCount is initialized to 0 when power is off or when the port configuration is completed, the port can exit from the state only when the power is removed.

63.7.4 Power Management Control

63.7.4.1 USB3.0 Power Control

Table 63.16 USB3.0 power management

| LTSSM State | USB3.0 PHY POWERDOWN | PLL Operation | Note |
|------------------------------|-------------------------|------------------|---|
| SS.Disabled | P2 | Running | Default |
| | P0/P2 | Running | The setting on USB30_CON.POW_SEL[2:0] is required. |
| | P3 | Running | The setting on USB30_CON.POW_SEL[2:0] is required. |
| Rx.Detect | P2 | Running | |
| U0 | P0 | Running | |
| U1 | P1 | Running | |
| U2 | P2 | Running | |
| U3 | P0/P2 | Running | Default Select P0 or P2 as the initial state on USB30_CON.U3_POW_SEL[1:0]. |
| | P0/P2 | Running | The setting on USB30_CON.POW_SEL[2:0] is required. |
| | P3 | Running | The setting on USB30_CON.POW_SEL[2:0] is required. |
| U3 -> U3_Wakeup,Reset | P0 | Running | Transition due to the reception of LFPS from host |
| U3 (P0/2) -> RemoteWakeup | P0 | Running | The setting on PORTSC.PLS[3:0] is required after waking up PLL in USB3.0-PHY. |
| U3(P3) -> RemoteWakeup | P0 | Running | The setting on USB30_CON.POW_SEL[2:0] is required. |
| Others | P0 | Running | |

Table 63.17 USB3.0 power management and configured state

| Configured/Unconfigured (CONF) | USB3.0 PHY POWERDOWN | CLK Gating | | | Note |
|-----------------------------------|-------------------------|------------|---------|---------|----------------------------------|
| | | EPC | DLINK30 | | |
| | | CLK125_P | CLK125 | CLKLFPS | |
| Unconfigured | P0/P1/P2 | Stopped | Running | Running | Only PIPE0 is accessible in EPC. |
| (CONF = 0) | P3 | Stopped | Running | Running | PLL Running |
| Configured | P0/P1/P2 | Running | Running | Running | All PIPEs are accessible in EPC |
| (CONF = 1) | P3 | Running | Running | Running | PLL Running |

[The behavior of PHY_POWERDOWN[1:0] in Disconnected state (SS.Disabled)]

- P2 is designated on PHY_POWERDOWN[1:0] with default setting.
- P0 or P3 is selectable as the request on PHY_POWERDOWN[1:0] with the setting of USB30_CON.POW_SEL[2:0].

[The behavior of PHY_POWERDOWN[1:0] in U3 state]

- P0 or P2 is designated on PHY_POWERDOWN[1:0] depending on the setting of USB30_CON.POW_SEL[2:0].
- In order to place PHY in U3 state, the additional operation is required. See 63.8.8.1 for details.

[In case of recovery from P3]

- Transition from P3 to P0 can be requested on USB30_CON.B3_PLLWAKE or B3_PLL_WAKEIN signal
See 63.8.8.6 for details.

63.7.4.2 USB3.0 Power State Control

(1) In case downstream port requests the transition from U0 to U1

When LGO_U1 as the request of transition to U1 state from the downstream port comes, Peripheral portion decides whether it accepts or not depending on the setting in registers and the internal status.

There are two fields in registers related to accepting LGO_U1, SSIFCMD.SSIF_URES[0] and USB30_CON.LPS_ENABLE[1:0].

SSIFCMD.SSIF_URES[0] is referred first in the decision whether LGO_U1 is accepted or not. If SSIFCMD.SSIF_URES[0]== 1 (forced LXU response), LGO_U1 is rejected and Peripheral portion returns LXU. If SSIFCMD.SSIF_URES[0]== 0 (normal response), USB30_CON.LPS_ENABLE[1:0] is referred next. If USB30_CON.LPS_ENABLE[1:0]==B'10 (forced rejected), LGO_U1 is rejected and Peripheral portion returns LXU. If USB30_CON.LPS_ENABLE[1:0]==B'01 (forced accepted), LGO_U1 is accepted and Peripheral portion returns LAU. If USB30_CON.LPS_ENABLE[1:0]==B'00 (basically accepted), it depends on the internal status of Peripheral portion whether it accepts LGO_U1 or not.

With USB30_CON.LPS_ENABLE[1:0]==B'00, LGO_U1 is rejected and Peripheral portion returns LXU if one of the conditions below is satisfied. Otherwise, it is accepted and Peripheral portion returns LAU.

- Peripheral portion is still in U0 state
- 500ms has not passed yet or the response from host has not come from the time Peripheral portion sends ERDY.
- ACK has not been received for the transmitted DP.
- IN PIPE has one or more packets to be transmitted in buffer.
- PP is 1 in the last packet received at OUT PIPE the control transfer is in progress
- USB_COM_CON.CONF== 0

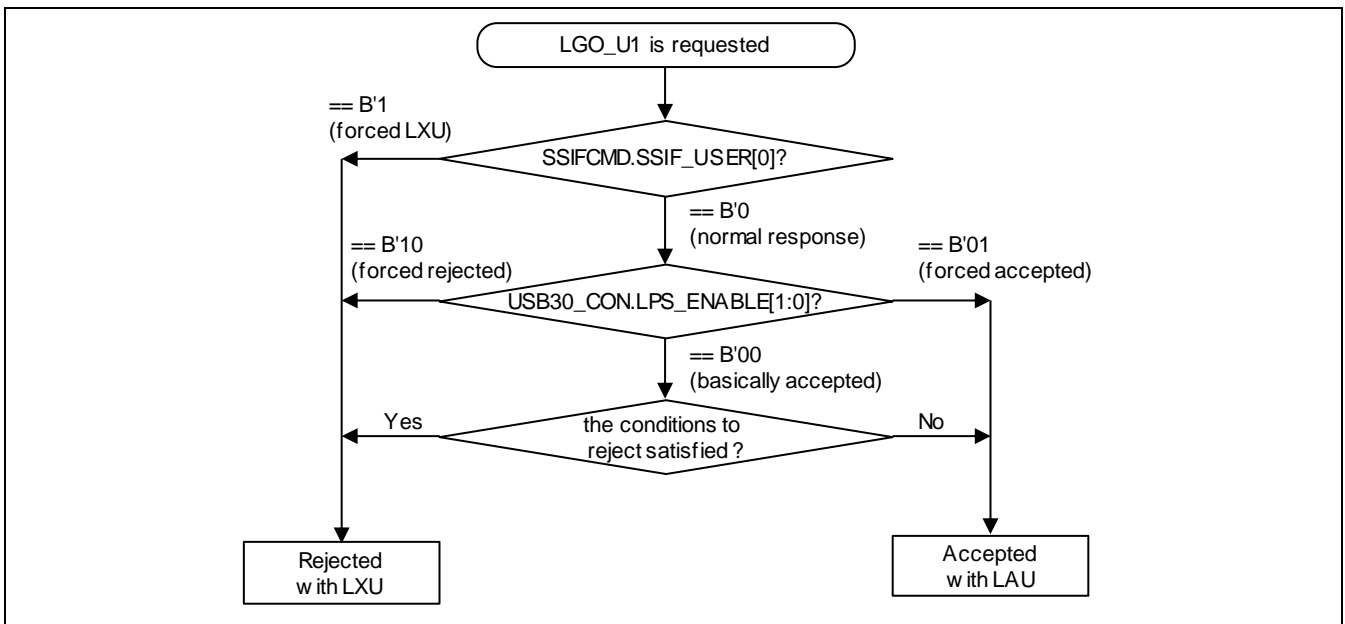


Figure 63.21 the decision flow whether the transition request to U1 is accepted or not

(2) In case device requests the transition from U0 to U1

On the timeout of PORTPMSC.U1_TIMEOUT[15:0], PERIPHERAL PORTION requests the transition from U0 state to U1 state. But it depends on the setting in registers and the internal status of PERIPHERAL PORTION whether it really requests the transition from U0 state to U1 state.

There are two fields in registers related to sending LGO_U1, SSIFCMD.SSIF_UREQ[0] and USB30_CON.LPS_ENABLE[1:0].

SSIFCMD.SSIF.UREQ[0] is referred first in the decision whether LGO_U1 is sent or not. If SSIFCMD.SSIF.UREQ[0]== 1, LGO_U1 isn't sent and PERIPHERAL PORTION never requests the transition from U0 state to U1 state. If SSIFCMD.SSIF.UREQ[1]== 0, USB30_CON.LPS_ENABLE[1:0] is referred next. If USB30_CON.LPS_ENABLE[1:0]==B'10 (forced rejected), LGO_U1 isn't sent and PERIPHERAL PORTION never requests the transition from U0 state to U1 state. If USB30_CON.LPS_ENABLE[1:0]==B'01 (forced accepted), LGO_U1 is sent and PERIPHERAL PORTION requests the transition from U0 state to U1 state on the timeout of PORTPMSC.U1_TIMEOUT[15:0]. If USB30_CON.LPS_ENABLE[1:0]==B'00 (basically accepted), it depends on the internal status of PERIPHERAL PORTION whether it sends LGO_U1 or not.

With USB30_CON.LPS_ENABLE[1:0]==B'00, LGO_U1 isn't sent even when the timeout of PORTPMSC.U1_TIMEOUT[15:0] occurs If one of the conditions below is satisfied. Otherwise, LGO_U1 is sent and PERIPHERAL PORTION requests the transition from U0 to U1 on the timeout of PORTPMSC.U1_TIMEOUT[15:0].

- PERIPHERAL PORTION is still in U0 state
- 500ms has not passed yet or the response from host has not come from the time PERIPHERAL PORTION sends ERDY.
- ACK has not been received for the transmitted DP.
- IN PIPE has one or more packets to be transmitted in buffer.
- PP is 1 in the last packet received at OUT PIPE

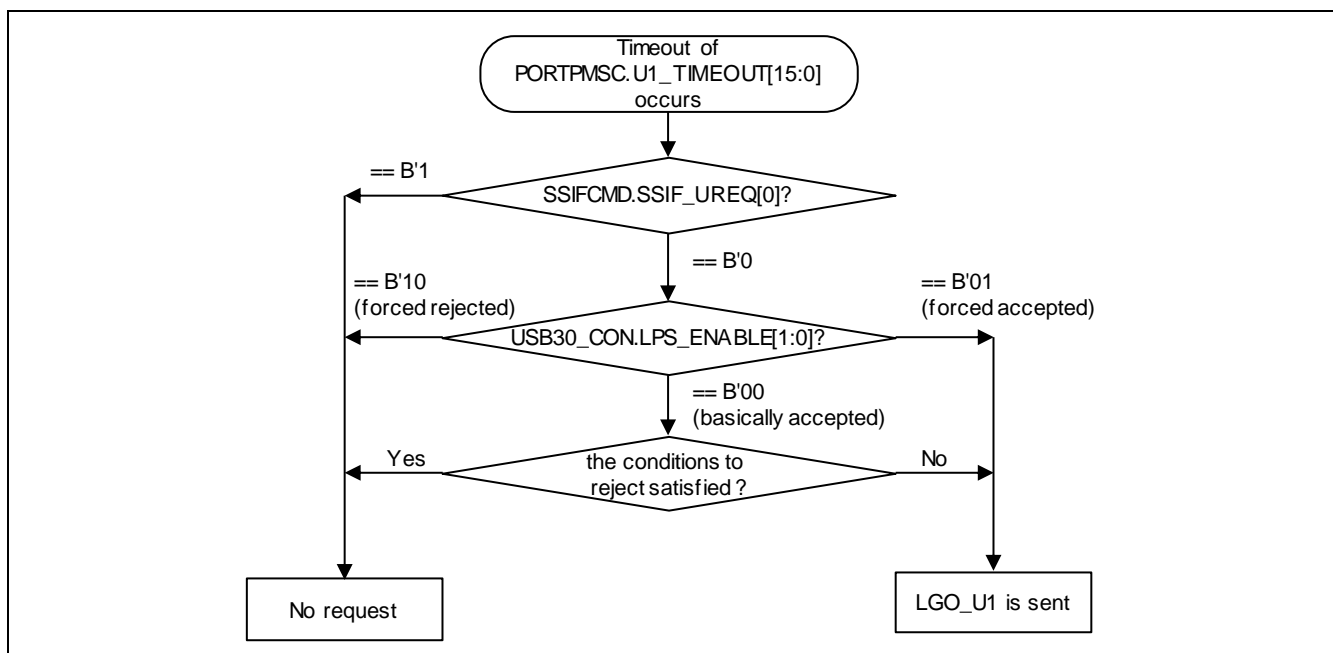


Figure 63.22 the decision flow whether the request of the transition from U0 to U1 is sent or not

It is also available to use PORTPSC.PLS[3:0] in order to send LGO_U1 at any point the software wants. See section 63.8.8.11 for details.

When the downstream port wants to exit from U1 state and U1 Exit LFPS comes, PERIPHERAL PORTION automatically responds to it and returns to U0 state.

At that time, USB_INT_STA_1.B3_LNKCNG_STA is asserted and an interrupt due to it is asserted if enabled. PORTPSC.PLS[3:0] shows U0 state (=H'0) after USB_INT_STA_1.B3_LNKCNG_STA is asserted.

(3) In case device requests the transition from U1 to U0

PERIPHERAL PORTION automatically requests the exit from U1 state to the downstream port by sending U1 Exit LFPS when one of the conditions below is satisfied.

- when PERIPHERAL PORTION wants to send ERDY for the next transfer
- when PERIPHERAL PORTION wants to send the transaction packet written in USB3_TPDAT_x (x=0,1,2) registers

In the first case, PERIPHERAL PORTION sends ERDY when the packet to be transmitted is prepared in buffer for IN PIPE and when the PIPE becomes ready to receive the next packet (Pn_CON.Pn_RES[1:0] is changed from B'00 to B'01, for example) for OUT PIPE.

In the second case, PERIPHERAL PORTION sends the transaction packet in USB3_TPDAT_x (x=0,1,2) registers when USB30_CON.B3_TP_SEND is set to 1.

For both cases, the operations to send packet or start transfer are required, but there is no additional operation to exit from U1 state. Therefore, it is not required for user to mind whether the link is in U1 state or not before sending packet or starting transfer.

(4) In case downstream port requests the transition from U0 to U2.

When LGO_U2 as the request of transition to U2 state from the downstream port comes, PERIPHERAL PORTION decides whether it accepts or not depending on the setting in registers and the internal status.

There are two fields in registers related to accepting LGO_U2, SSIFCMD.SSIF_URES[1] and USB30_CON.LPS_ENABLE[1:0].

SSIFCMD.SSIF.URES[1] is referred first in the decision whether LGO_U2 is accepted or not. If SSIFCMD.SSIF.URES[1]== 1 (forced LXU response), LGO_U2 is rejected and PERIPHERAL PORTION returns LXU. If SSIFCMD.SSIF.URES[1]== 0 (normal response), USB30_CON.LPS_ENABLE[1:0] is referred next. If USB30_CON.LPS_ENABLE[1:0]==B'10 (forced rejected), LGO_U2 is rejected and PERIPHERAL PORTION returns LXU. If USB30_CON.LPS_ENABLE[1:0]==B'01 (forced accepted), LGO_U2 is accepted and PERIPHERAL PORTION returns LAU. If USB30_CON.LPS_ENABLE[1:0]==B'00 (basically accepted), it depends on the internal status of PERIPHERAL PORTION whether it accepts LGO_U2 or not.

With USB30_CON.LPS_ENABLE[1:0]==B'00, LGO_U2 is rejected and PERIPHERAL PORTION returns LXU If one of the conditions below is satisfied. Otherwise, it is accepted and PERIPHERAL PORTION returns LAU.

- PERIPHERAL PORTION is still in U0 state
- 500ms has not passed yet or the response from host has not come from the time PERIPHERAL PORTION sends ERDY.
- ACK has not been received for the transmitted DP.
- IN PIPE has one or more packets to be transmitted in buffer.
- PP is 1 in the last packet received at OUT PIPE

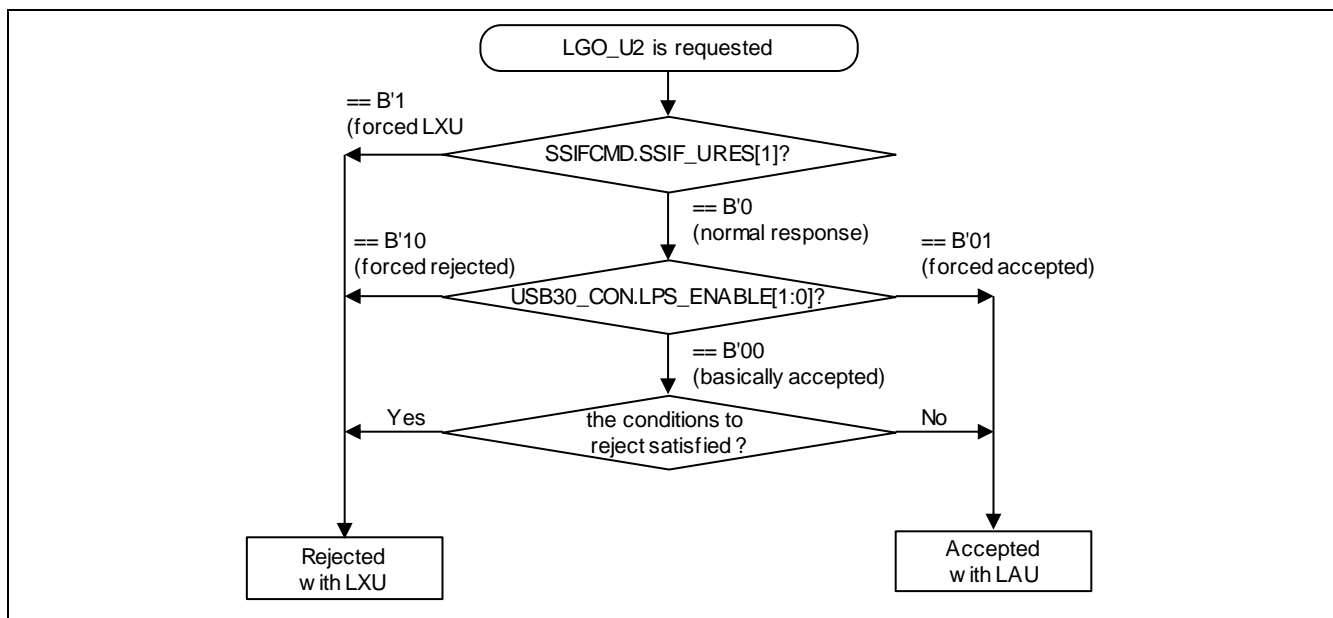


Figure 63.23 the decision flow whether the transition request to U2 is accepted or not

(5) In case device requests the transition from U0 to U2

On the timeout of PORTPMSC.U2_TIMEOUT[15:0], PERIPHERAL PORTION requests the transition from U0 state to U2 state. But it depends on the setting in registers and the internal status of PERIPHERAL PORTION whether it really requests the transition from U0 state to U2 state.

But the timeout value in U1_TIMEOUT[15:0] is smaller than that in U2_TIMEOUT[15:0] in general. In that case, U1 timeout occurs first, then U2 timeout occurs next if the device is still in U1 state. See section 63.7.4.2 (6) for the case.

There are two fields in registers related to sending LGO_U2, SSIFCMD.SSIF_UREQ[1] and USB30_CON.LPS_ENABLE[1:0].

SSIFCMD.SSIF_UREQ[1] is referred first in the decision whether LGO_U2 is sent or not. If SSIFCMD.SSIF_UREQ[1]== 1, LGO_U2 isn't sent and PERIPHERAL PORTION never requests the transition from U0 state to U2 state. If SSIFCMD.SSIF_UREQ[1]== 0, USB30_CON.LPS_ENABLE[1:0] is referred next. If USB30_CON.LPS_ENABLE[1:0]==B'10 (forced rejected), LGO_U2 isn't sent and PERIPHERAL PORTION never requests the transition from U0 state to U2 state. If USB30_CON.LPS_ENABLE[1:0]==B'01 (forced accepted), LGO_U2 is sent and PERIPHERAL PORTION requests the transition from U0 state to U2 state on the timeout of PORTPMSC.U2_TIMEOUT[15:0]. If USB30_CON.LPS_ENABLE[1:0]==B'00 (basically accepted), it depends on the internal status of PERIPHERAL PORTION whether it sends LGO_U2 or not.

With USB30_CON.LPS_ENABLE[1:0]==B'00, LGO_U2 isn't sent even when the timeout of PORTPMSC.U2_TIMEOUT[15:0] occurs If one of the conditions below is satisfied. Otherwise, LGO_U2 is sent and PERIPHERAL PORTION requests the transition from U0 to U2 on the timeout of PORTPMSC.U2_TIMEOUT[15:0].

- PERIPHERAL PORTION is still in U0 state.
- 500ms has not passed yet or the response from host has not come from the time PERIPHERAL PORTION sends ERDY.
- ACK has not been received for the transmitted DP.
- IN PIPE has one or more packets to be transmitted in buffer.
- PP is 1 in the last packet received at OUT PIPE
- the control transfer is in progress
- USB_COM_CON.CONF== 0

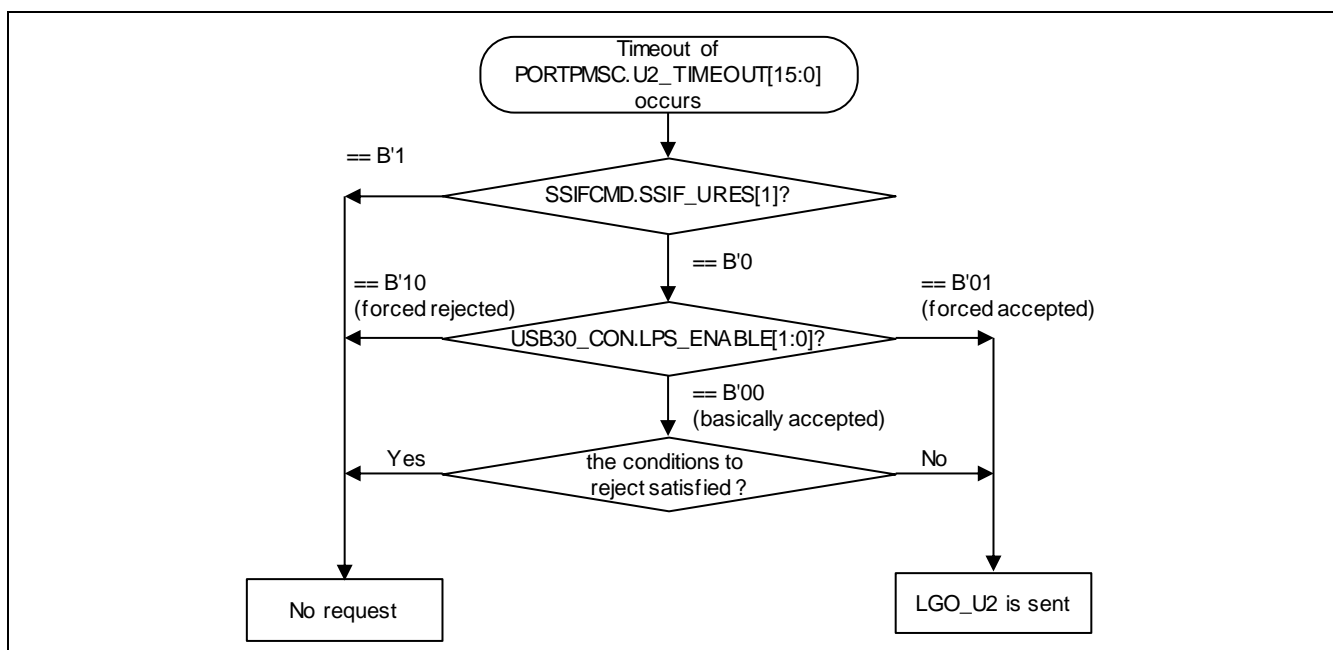


Figure 63.24 the decision flow whether the request of the transition from U0 to U2 is sent or not

(6) In case device requests the transition from U1 to U2

As explained in section 63.7.4.2 (5), When PERIPHERAL PORTION is in U1 state and the timeout of PORTPMSC.U2_TIMEOUT[15:0] occurs, PERIPHERAL PORTION transits from U1 state to U2 state without returning to U0 state. In that case, it is required that SSIFCMD.SSIF_UREQ[1] is set to 0.

At that time, USB_INT_STA_1.B3_LNKCNG_STA is asserted and an interrupt due to it is asserted if enabled. PORTSC.PLS shows U2 state (=H'2) after USB_INT_STA_1.B3_LNKCNG_STA is asserted.

(7) In case downstream port requests the transition from U2 to U0

When the downstream port wants to exit from U2 state and U2 Exit LFPS comes, PERIPHERAL PORTION automatically responds to it and returns to U0 state.

At that time, USB_INT_STA_1.B3_LNKCNG_STA is asserted and an interrupt due to it is asserted if enabled. PORTSC.PLS[3:0] shows U0 state (=H'0) after USB_INT_STA_1.B3_LNKCNG_STA is asserted.

(8) In case device requests the transition from U2 to U0

PERIPHERAL PORTION automatically requests the exit from U2 state to the downstream port by sending U2 Exit LFPS when one of the conditions below is satisfied.

- when PERIPHERAL PORTION wants to send ERDY for the next transfer
- when PERIPHERAL PORTION wants to send the transaction packet written in USB3_TPDAT_x (x=0,1,2) registers

In the first case, PERIPHERAL PORTION sends ERDY when the packet to be transmitted is prepared in buffer for IN PIPE and when the PIPE becomes ready to receive the next packet (Pn_CON.Pn_RES[1:0] is changed from B'00 to B'01, for example) for OUT PIPE.

In the second case, PERIPHERAL PORTION sends the transaction packet in USB3_TPDAT_x (x=0,1,2) registers when USB30_CON.B3_TP_SEND is set to 1.

For both cases, the operation to send packet or start transfer is required, but there is no additional operation to exit from U2 state. Therefore, it is not required for user to mind whether the link is in U1 state or not before sending packet or starting transfer.

(9) In case downstream port requests the transition from U0 to U3

Only downstream port can initiate U3 entry. An upstream port shall not reject U3 entry.

PERIPHERAL PORTION automatically responds to LGO_U3, accepts it and transits to U3 state.

USB_INT_STA_1.B3_LNKCNG_STA is asserted and an interrupt due to it is asserted if enabled. PORTSC.PLS[3:0] shows U3 state (=H'3) after USB_INT_STA_1.B3_LNKCNG_STA is asserted.

But only with the transition to U3 above, USB3.0 PHY is placed in P0 state or P2 state. If it is required to place PHY in P3 state, See the flow shown in section 63.8.8.1.

(10) In case downstream port requests the transition from U3 to U0

When the downstream port wants to exit from U3 state and U3 Wakeup LFPS comes, PERIPHERAL PORTION automatically responds to it and returns to U0 state.

At that time, USB_INT_STA_1.B3_LNKCNG_STA is asserted and an interrupt due to it is asserted if enabled. PORTSC.PLS[3:0] shows U0 (=H'0) after USB_INT_STA_1.B3_LNKCNG_STA is asserted.

(11) In case device requests the transition from U3 to U0

PERIPHERAL PORTION automatically requests the transition from U3 state to U0 by sending U3 Wakeup LFPS when the transition is requested on PORTSC.PLS[3:0]. In U3, PORTSC.PLS[3:0] shows U3 state (=H'3). If PORTSC.PLS[3:0] is changed to U0 state (=H'0), PERIPHERAL PORTION sends U3 Wakeup LFPS to the link partner in order to request the transition from U3 state to U0 state.

When the LFPS handshake is completed and the link goes back to U0 state, USB_INT_STA_1.B3_LNKCNG_STA is asserted and an interrupt due to it is asserted if enabled. PORTSC.PLS[3:0] shows U0 state (=H'0) after USB_INT_STA_1.B3_LNKCNG_STA is asserted.

63.7.4.3 USB2.0 Power State Control

In the specification of USB2.0 link power management, L0 state is defined as “On” state.

L3 state is defined as “Off” state and corresponds to the powered-off, disconnected and disabled state.

L2 state is defined as “Suspend” state and corresponds to so-called “USB2.0 suspend” in the past.

L1 state is defined as “Sleep” state. LPM transaction is defined for the state.

Table 63.18 USB2.0 power management

| State | Configured (CONF) | PLL Operation | CLK Gating | | | | Supplement |
|---------------|-------------------------|---------------|------------|---------|---------|----------|---|
| | | | CLK60 | CLK60_G | CLK60_P | CLK60_PG | |
| L3 | — | Running | Running | Stopped | Stopped | Stopped | (default) |
| | | Stopped | Stopped | Stopped | Stopped | Stopped | PLL stop is requested on USB20_CON.B2_SUSPEND |
| L0 | Unconfigured (CONF = 0) | Running | Running | Running | Stopped | Stopped | Only PIPE0 is running |
| | Configured (CONF = 1) | Running | Running | Running | Running | Running | All PIPEs are running |
| L1/L2 | | Running | Running | Stopped | Running | Stopped | (default) |
| | | Stopped | Stopped | Stopped | Stopped | Stopped | PLL stop is requested on USB20_CON.B2_SUSPEND |
| L1/L2 → L0 | | Running | Running | Running | Running | Running | Wakeup received from Host |
| | | Running | Running | Running | Running | Running | Remote wakeup initiated by device |

[L3 (Off) State]

- Disconnected state for USB2.0
- PLL is running in default setting but it can be stopped on USB20_CON.B2_SUSPEND.
- 60MHz clock other than CLK60 is suspended.

[L0 (On) State]

- Active state in USB2.0 connection
- PLL is running
- In unconfigured state, CLK60_P and CLK60_PG are suspended. It means that clock to PIPEs other than PIPE0 is suspended.
In configured state, all clocks of 60MHz is running.

[L1 (Sleep) State]

- Low power state selected on LPM in USB2.0 connection
- PLL is running.
- CLK60_G and CLK60_PG are suspended.

[L2 (Suspend) State]

- Suspended state in USB2.0 connection
- PLL is running but it can be stopped on USB20_CON.B2_SUSPEND.
PLL is resumed automatically when the wakeup request from downstream port is received.
- CLK60_G and CLK60_PG are suspended.

(1) Transition from L0 to L1

L1 state in USB2.0 is similar to U1 state and U2 state in USB3.0 in the point that the hardware should control the transition to or the exit from the state mainly. But unlike the transition to U1 state or U2 state, the transition to L1 state is initiated only by downstream port.

LPM token is defined as one of extended token packet for the transition. The device returns ACK if it accepts the transition or NYET if it rejects the transition when LPM token comes. It is also allowed to return STALL if it doesn't support LPM function.

PERIPHERAL PORTION can respond with “no response” or return STALL to LPM token with the setting to do so, but it should return only ACK or NYET basically.

When LPM token from downstream port comes, PERIPHERAL PORTION decides whether it accepts or not depending on the setting in the registers and the internal status.

LPM function can be enabled or disabled on USB20_CON.LPM_DISABLE. When USB20_CON.LPM_DISABLE== 1, PERIPHERAL PORTION disables LPM function and return no response to LPM token and take no action for it. But the device should support LPM function, so it should enable LPM function.

USB20_CON.LPM_ENABLE[1:0] also defines the response of PERIPHERAL PORTION to LPM token. If USB20_CON.LPM_ENABLE[1:0]==B'10 (forced rejected), the request is rejected and PERIPHERAL PORTION returns NYET to LPM token. If USB20_CON.LPM_ENABLE[1:0]==B'01 (forced accepted), the request is accepted and PERIPHERAL PORTION returns ACK to LPM token. If USB20_CON.LPM_ENABLE[1:0]==B'00 (basically accepted), it depends on the internal status of PERIPHERAL PORTION whether it accepts the request or not. If USB20_CON.LPM_ENABLE[1:0]==B'11 (unsupported), PERIPHERAL PORTION returns STALL to LPM token. But the device should not return STALL if it supports LPM function.

If one of the conditions below is satisfied, the request is rejected and PERIPHERAL PORTION returns NYET. Otherwise, it is accepted and PERIPHERAL PORTION returns ACK.

- IN PIPE has one or more packets to be transmitted in buffer.

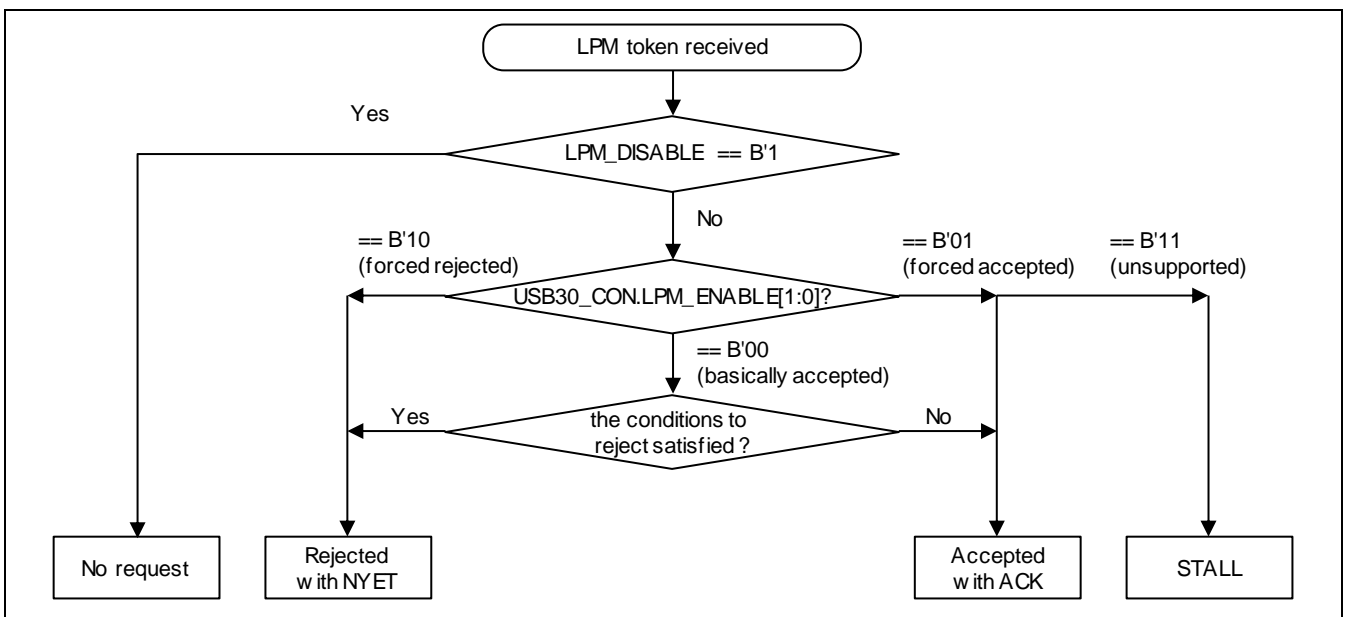


Figure 63.25 the decision flow whether the transition request to L1 is accepted or not

The software can know that the transition to L1 state is requested on `USB_INT_STA_1.B2_L1SPND_STA`. An interrupt due to it is asserted if enabled.

(2) In case downstream port requests the transition from L1 to L0

When the downstream port wants to exit from L1 state and the resume signal comes, PERIPHERAL PORTION automatically responds to it and returns to L0 state.

At that time, `USB_INT_STA_1.B2_L1RSUM_STA` is asserted and an interrupt due to it is asserted if enabled.

(3) In case device requests the transition from L1 to L0

PERIPHERAL PORTION automatically requests the exit from L1 state to the downstream port by sending the resume signal when it has one or more packets to be transmitted in buffer for IN PIPE. In this case, there is no additional operation to exit from L1 state.

The transition from L1 state to L0 state is also required by setting `USB20_CON.B2_RSUM_IN`. When this bit is set, the resume signal to request the exit from L1 state is sent to the downstream port.

(4) Transition from L0 to L2 (Suspend)

L2 state is defined as USB2.0 Suspend in the original USB2.0 specification. The transition to L2 state is requested only from downstream port and is controlled automatically by the hardware of PERIPHERAL PORTION.

The downstream port can request the transition to L2 state by stopping any bus activity and put USB bus in a constant idle state for more than 3.0ms and device can not deny the transition.

When PERIPHERAL PORTION detects that the transition to L2 state is being requested from downstream port (that is, USB bus stays in idle state for more than 3.0ms), it automatically follows it and asserts `USB_INT_STA_1.B2_SPND_STA`.

PERIPHERAL PORTION follows to the request of the transition to L2 state, but PLL in USB2.0-PHY is still running in L2 state. If it is required to stop PLL, set `USB20_CON.B2_SUSPEND`. See section 63.8.8.7 for details.

(5) In case downstream port requests the transition from L2 to L0

When the downstream port wants to exit from L2 state and the resume signal comes, PERIPHERAL PORTION automatically responds to it and returns to L0 state.

At that time, `USB_INT_STA_1.B2_RSUM_STA` is asserted and an interrupt due to it is asserted if enabled.

(6) In case device requests the transition from L2 to L0

The software can request the wakeup of PERIPHERAL PORTION from L2 state. If PLL in USB2.0-PHY is suspended, it should be awoken first. See section 63.8.8.10 for how to awake PLL in USB2.0-PHY.

After PLL is resumed, the software can send the resume signal from PERIPHERAL PORTION by setting `USB20_CON.RSUM_IN`. See section 63.8.8.9 for details

63.7.5 AXI DMA Transfer Control

PERIPHERAL PORTION executes DMA transfer to or from data buffers connected to SRAM interface using the bus master function of Peripheral portion. The DMA master function of Peripheral portion uses Physical Region Descriptor (PRD) Table on system memory, which has source address, destination address, the size of transfer and other informations related to the DMA transfer. It is assumed that the software (driver, for example) prepares PRD table on system memory. PRD tables are prepared separately for each PIPE.

The operation flow of DMA transfer using master function of Peripheral portion is described below.

63.7.5.1 OUT Transfer

- (1) Interrupt is asserted if PRD table is not prepared

Peripheral portion asserts AXI_INT_SYS if a certain PIPE receives OUT packet and the OUT packet becomes ready to be read but PRD table is not prepared for it yet.

If PRD table is already prepared for the PIPE and DMA transfer is enabled, Peripheral portion doesn't assert AXI_INT_SYS here. In that case, (2) and (3) can be skipped.

- (2) Making PRD table

The software makes PRD table according to the defined format.

- (3) Setting for bus master

The software specifies PRD table made at (2) to registers of Peripheral portion, and enables DMA transfer.

- (4) Start of DMA transfer

Peripheral portion reads PRD table when it detects EPC_D_Pn_DATAEN[n] (n is the index of the PIPE for which PRD table is prepared for) is asserted and knows Peripheral portion requests DMA transfer. Then, Peripheral portion writes received data in a unit of packet to the address which is assigned by PRD table.

- (5) Writing back the result of transfer to PRD table

When DMA transfer is completed, Peripheral portion writes back the result of transfer to PRD table.

- (6) Interrupt is asserted to notify completion of DMA transfer

Peripheral portion asserts AXI_INT_DMA when it completes DMA transfer.

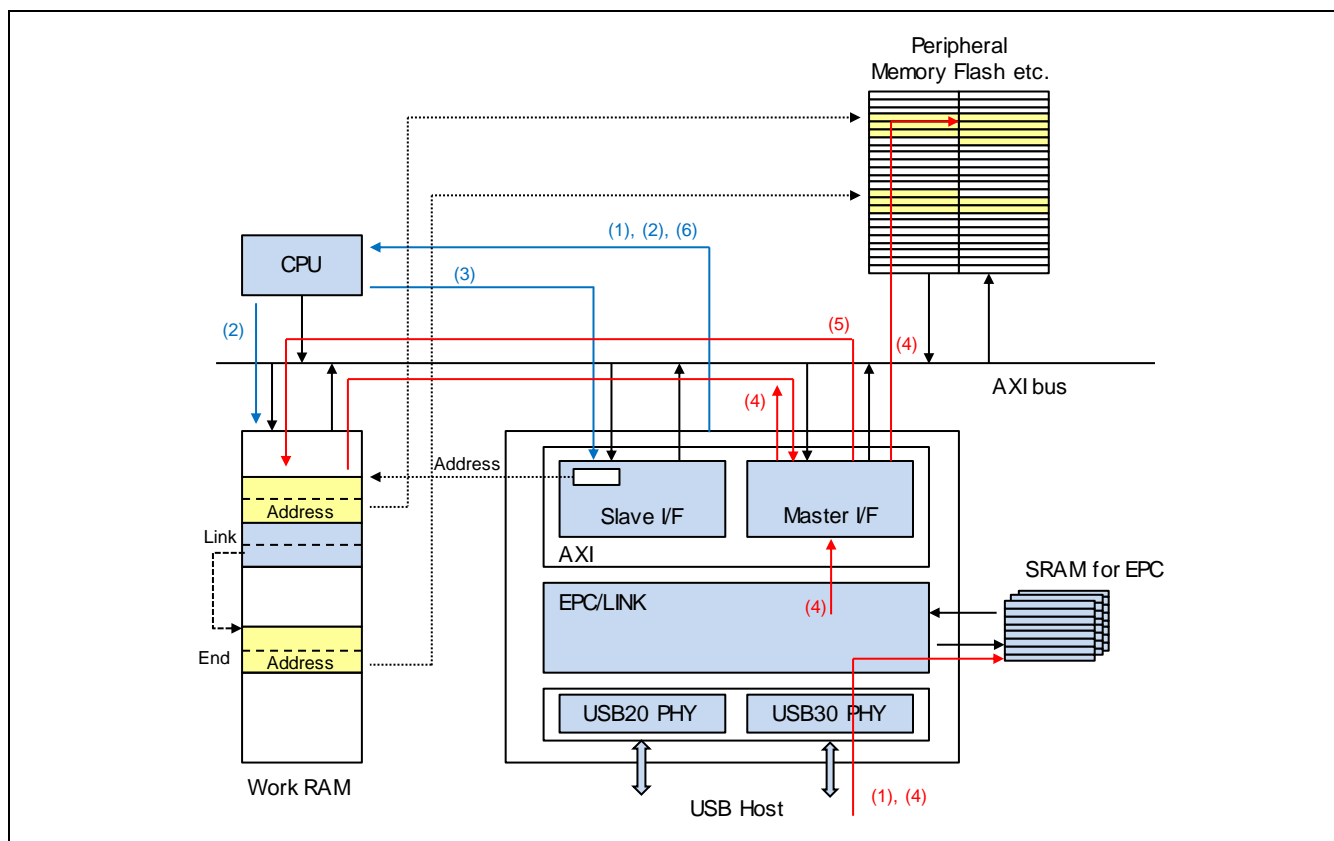


Figure 63.26 example of operation for OUT transfer

63.7.5.2 IN Transfer

(1) Interrupt is asserted if PRD table is not prepared

Peripheral portion asserts AXI_INT_SYS if a certain PIPE becomes ready to be written for IN packet but PRD table is not prepared for it yet.

If PRD table is already prepared for the PIPE and DMA transfer is enabled, Peripheral portion doesn't assert an interrupt. In that case, (2) and (3) can be skipped.

(2) Making PRD table

The software makes PRD table according to the defined format.

(3) Setting for bus master

The software specifies PRD table made at (2) to registers of Peripheral portion, and enables DMA transfer.

(4) Start of DMA transfer

Peripheral portion reads PRD table when it detects EPC_D_Pn_DATAEN[n] (n is the index of the PIPE for which PRD table is prepared for) is asserted and knows Peripheral portion requests DMA transfer. Then, Peripheral portion read data to be transmitted in unit of a packet from the address which is assigned by PRD table.

(5) Write back of result of transfer to PRD table

When DMA transfer is completed, Peripheral portion writes back the result of transfer to PRD table.

(6) Interrupt asserted to notify completion of DMA transfer

Peripheral portion asserts AXI_INT_DMA when it completes DMA transfer.

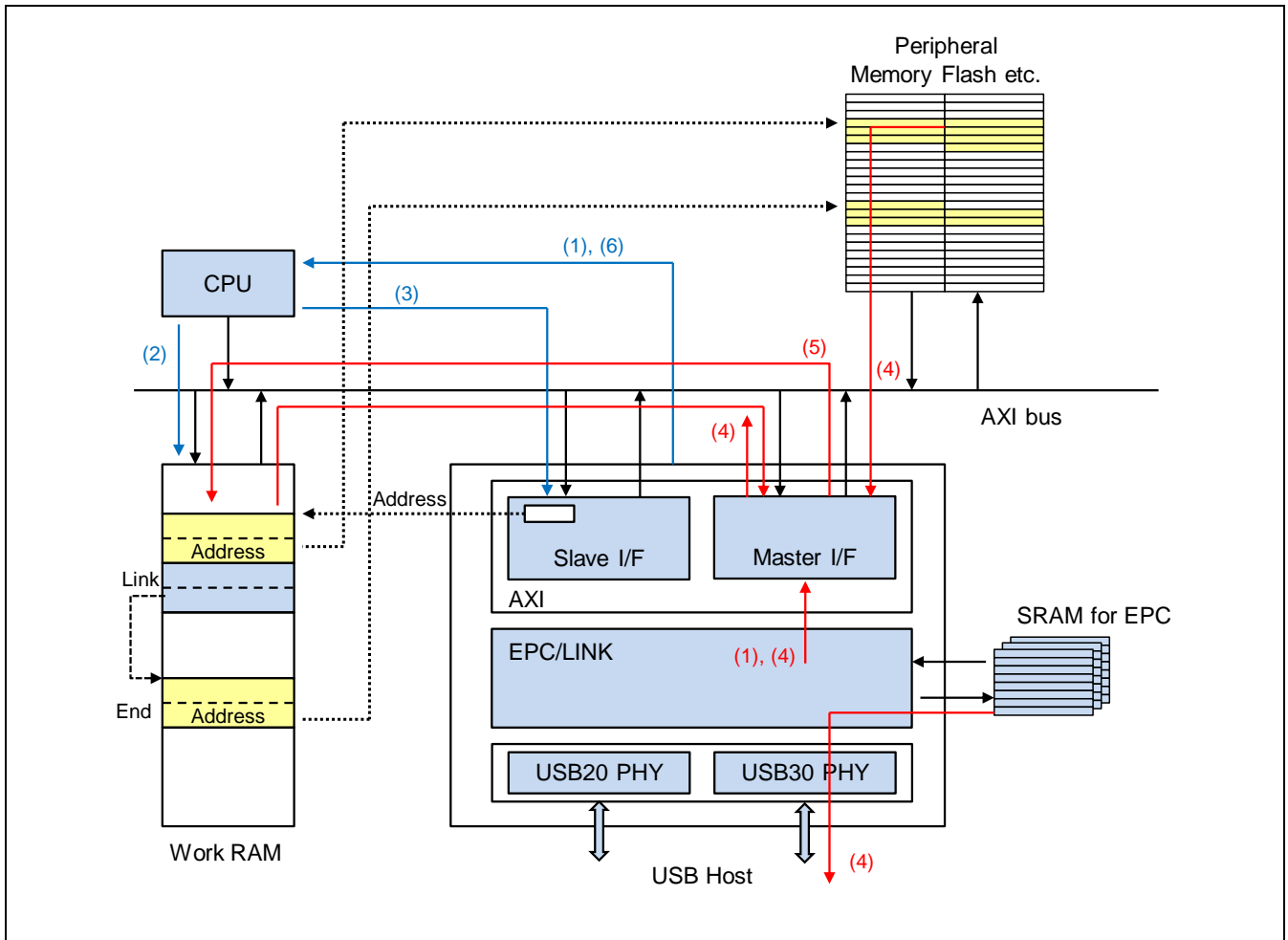


Figure 63.27 example of operation for IN transfer

63.7.5.3 Two PRD tables for one group of transfer

The basic concept of two PRD tables used for one group of transfer is described below. Especially the Stream bulk protocol is assumed here.

For Stream bulk transfer, Stream ID is assigned to PERIPHERAL PORTION and it executes transfer related to the Stream ID. The Stream ID is set before PRD table is allocated to PERIPHERAL PORTION.

From the viewpoint of the performance of system, it is desirable to prepare for the next transfer just before the current OUT transfer is completed. PERIPHERAL PORTION can execute transfer seamlessly in that case.

In order to do so, there is a way to use PRD tables for one group of transfer.

Two PRD tables are prepared so that the group of transfer is divided to them beforehand. One PRD table has the part from the first packet to the (n)-th packet, and the other table has that from the (n+1) th packet to the last packet. The (n) is assumed as the second or the third packet from the last. The Int bits are set for both tables. With this usage, the software can know the current transfer is almost completed by the interrupt of the first PRD table. When the interrupt of the first PRD table comes, software can start the preparation for the next transfer. It can also confirm the completion of the current transfer by the interrupt of the second PRD table.

This method is applicable for non Stream bulk transfer.

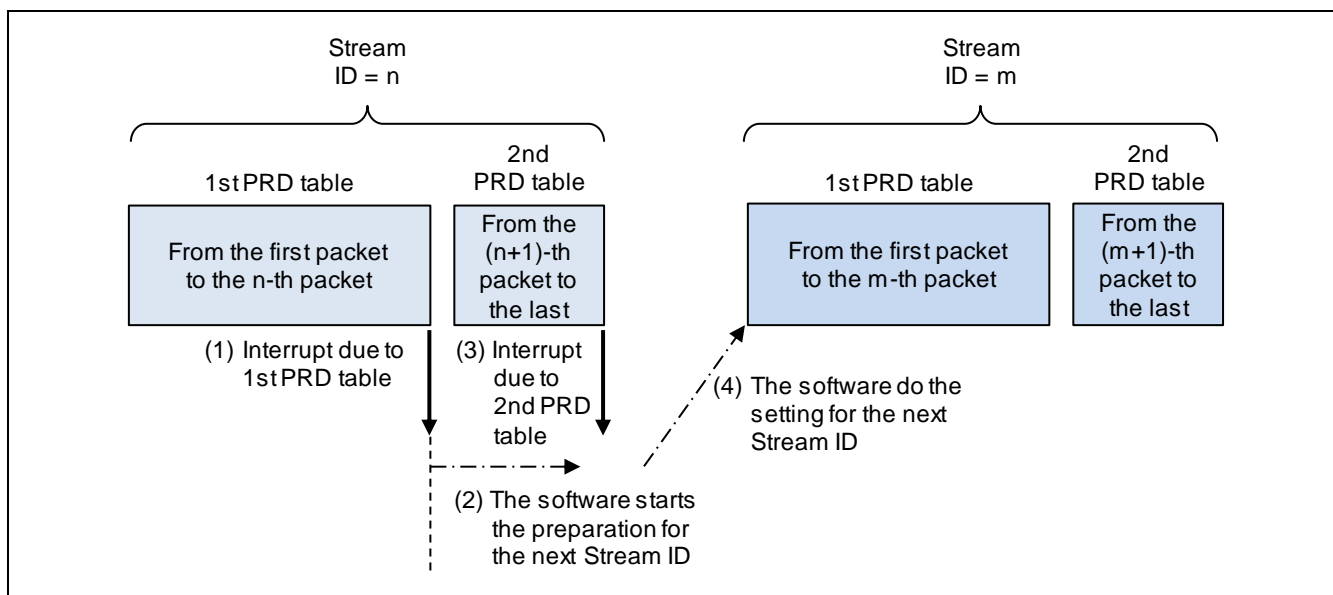


Figure 63.28 Two PRD tables used for one group transfer

63.7.5.4 DMA Transfer using EXTPRD_EN

Peripheral portion has four register groups used for DMA transfers, -- DMA_Ch0_CONx and DMA_Ch0_PRD_ADRx (x = 1 to 4). USB3.0 Controller can coordinate multiple PRD tables and their transfers by making links between those PRD tables on DMA_Ch0_CONx.EXPRD_ENx and DMA_Ch0_CONx.EXTPRD_NOx[1:0].

An example in case that multiple PRD tables and their transfers are coordinated on DMA_Ch0_CONx.EXPRD_ENx is described below. In this example, PRD tables from (1) to (4) are set to registers at first, and then PRD table of (5) is set after DMA transfer of (1) is completed. Since DMA_Ch0_CONx.EXPRD_ENx is set to 1 for (2), (3) and (5), each DMA_Ch0_CONx.PRD_ENx is masked and each DMA transfer is kept waiting until the transfer related on DMA_Ch0_CONx.EXTPRD_NOx[1:0] is completed.

As for (2), DMA_Ch0_CON2.EXTPRD_NO2[1:0]=0 is specified. In this case, the transfer related to (2) is suspended until the transfer at PIPE1 with DMA_Ch0_CON1 is completed. When the transfer at PIPE1 with DMA_CH0_CON1 is completed and DMA_Ch0_CON1.PRD_EN1 is cleared by hardware, DMA_Ch0_CON2.EXTPRD_EN2 is cleared automatically by hardware at the same time. The clearing of DMA_Ch0_CON2.EXTPRD_EN2 means that DMA_Ch0_CON2.PRD_EN2 gets unmasked, and the transfer of (2) is allowed at this point.

[Example of PRD Table Setting]

| No. | | PRD_EN | PIPE_NO | EXTPRD_EN | EXTPRD_NO |
|-----|--------------|--------|---------|--------------|-----------|
| (1) | DMA_Ch0_CON1 | 1 | PIPE1 | 0 (Normal) | — |
| (2) | DMA_Ch0_CON2 | 1 | PIPE1 | 1 (Expanded) | 0 (CON1) |
| (3) | DMA_Ch0_CON3 | 1 | PIPE2 | 1 (Expanded) | 1 (CON2) |
| (4) | DMA_Ch0_CON4 | 1 | PIPE3 | 0 (Normal) | — |
| (5) | DMA_Ch0_CON1 | 1 | PIPE1 | 1 (Expanded) | 2 (CON3) |

[Example of Operation]

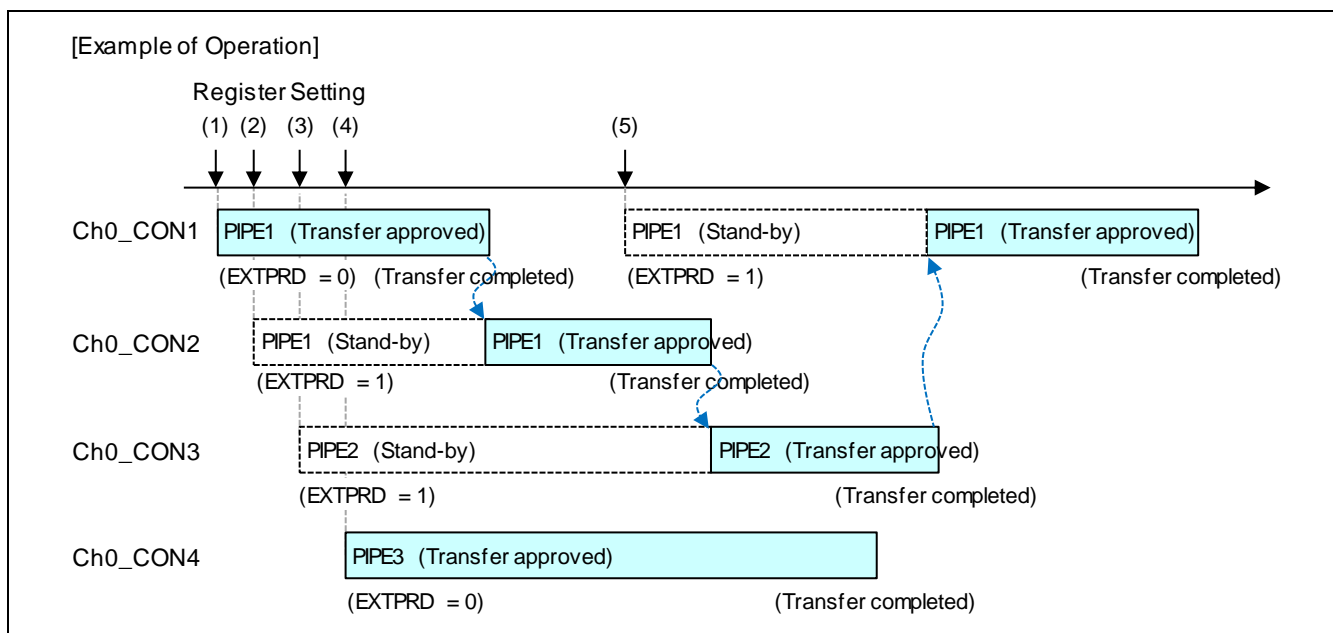


Figure 63.29 Transfer using EXTPRD_EN

This function is also used for the case a certain PIPE is expected to start the transfer just after that at other PIPE is completed.

DMA_Ch0_CONx.EXTPRD_NOx[1:0] and DMA_Ch0_CONx.EXTPRD_ENx should be set before the transfer for PRD table related on DMA_Ch0_CONx.EXTPRD_NOx[1:0] is completed (when it is completed, DMA_Ch0_CONx.PRD_ENx is cleared). Make sure that DMA_Ch0_CONx.PRD_ENx is still 1 just when DMA_Ch0_CONx.EXTPRD_NOx[1:0] and DMA_Ch0_CONx.EXTPRD_ENx are set and related to it. If DMA_Ch0_CONx.PRD_ENx has already been cleared, don't use DMA_Ch0_CONx.EXTPRD_NOx[1:0] and DMA_Ch0_CONx.EXTPRD_ENx for it.

63.7.5.5 Example of transfer using bus master function

An Example in which short packet is received as the last one at OUT (AXI write) PIPE is described below.

Peripheral portion provides 4 register groups to have the information of PRD Tables. The software prepares PRD Table on system memory and requests the transfer by setting the start address of PRD table, the information of PIPE (index of PIPE and direction of transfer) and DMA_Ch0_CONx.PRD_ENx to one of register groups. Especially setting of DMA_Ch0_CONx.PRD_ENx to 1 means the transfer on the PIPE is enabled.

Peripheral portion indicates whether transfer is enabled or not for each PIPE on EPC_D_Pn_DATAEN[n] (n=1-30), which is one of the signals at Data Interface of Peripheral portion.

Peripheral portion is kept waiting until Peripheral portion notifies it that the transfer of PIPE related on the PRD table is enabled. Peripheral portion starts the transfer when it detects that one of transfers related on PRD tables is enabled.

Upon the start of transfer, Peripheral portion reads the first PRD table. When the type of PRD table is a link pointer, Peripheral portion executes writing-back in order to set used bit of the PRD table first and then reads the next PRD table from the address which is held in BAP. When the type of PRD table is a buffer pointer, Peripheral portion starts data transfer according to the PRD table.

Peripheral portion executes data transfer at Data Interface in a unit of a packet of USB (the size of packet is up to the max packet size of USB).

When the transfer of the packet is completed, Peripheral portion updates the PRD table that was read just prior to the transfer and writes back it so that SIZE field and BAP field in the table are updated. If the remain size of the PRD table is not zero and Peripheral portion notifies Peripheral portion that it can still continue transfer of the PIPE when Peripheral portion completes writing-back of PRD table, Peripheral portion uses PRD table currently held and start transfer of the next packet for the PIPE. On the contrary, in case that all transfers specified by the PRD table held currently in Peripheral portion have been done, case that the transfer for the PRD table is terminated by receiving short packet or case that EPC notifies that no transfer on the PIPE is pending, Peripheral portion returns back to the status to wait for transfer.

When the transfer is completed, the size information in SIZE field of table is updated to the value that the size of transferred packet is subtracted from the remaining size before the transfer, then Peripheral portion writes back the PRD table.

If it is required to know the actual size transferred, store the remaining size before the transfer, and subtract the size in the updated PRD table after the transfer from it.

In case that short packet is received in the middle of PRD Table or case that all transfers specified by PRD table have been completed, Peripheral portion completes the process of the current PRD table. At this time Peripheral portion clears DMA_Ch0_CONx.PRD_ENx, sets the status bit in DMA_INT_STA register and asserts AXI_INT_DMA if enabled.

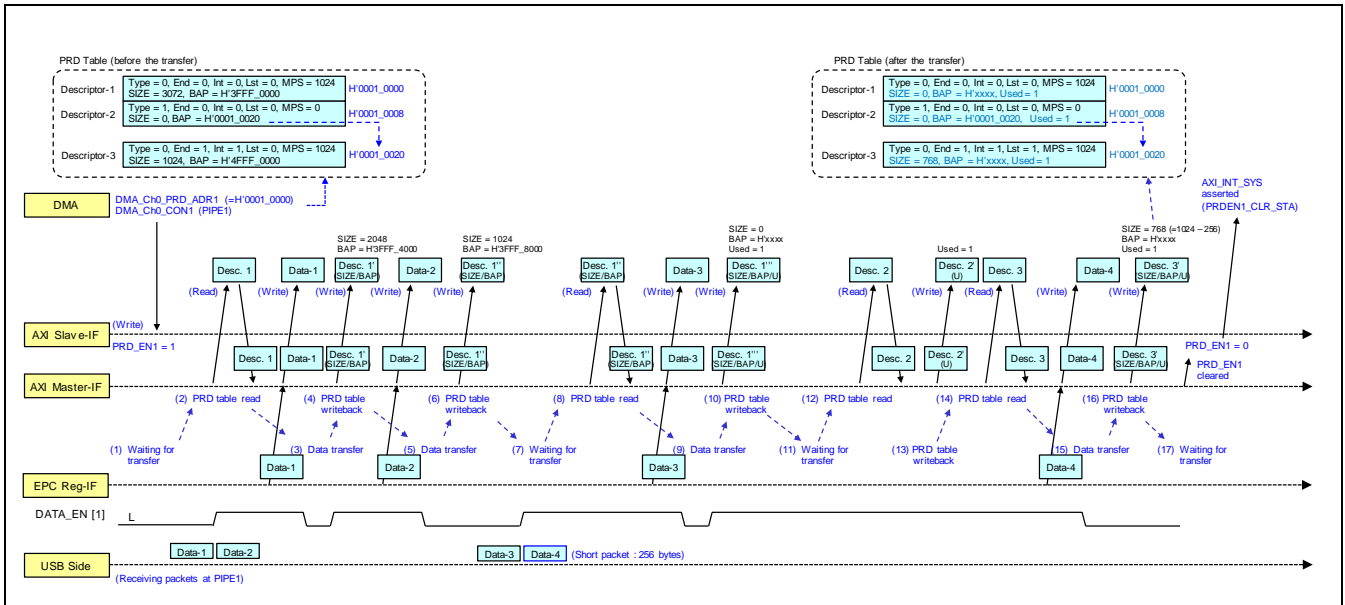


Figure 63.30 example of OUT transfer terminated with short packet

63.7.5.6 Finishing the process for PRD table and its conditions

Peripheral portion has DMA_Ch0_CONx and DMA_Ch0_PRD_ADRx (x = 1 to 4) to have the information of PRD tables, and starts transfer when it detects EPC_D_Pn_DATAEN[n] (n = 1 to 30) at Data Interface is asserted from Peripheral portion. Asserting EPC_D_Pn_DATAEN[n] indicates that transfer on PIPE specified in PRD table is enabled. Transfer is executed in a unit of a packet of USB (that is, the size is up to the max packet size of USB), following the order specified by PRD tables. When one of the conditions to finish the transfer is satisfied, Peripheral portion recognizes that the process of the current PRD table has been completed, clears DMA_Ch0_CONx.PRД_ENx and sets the interrupt status in DMA_INT_STA register. Peripheral portion finishes the transfer of PRD table when one of conditions below is satisfied.

- (1) All transfers related from the first PRD table to the last one (End bit is set) have been completed, or
- (2) Transfer is terminated by receiving short packet in OUT transfer (AXI write transfer), or
- (3) EPC_D_DBRTRANSEND which indicates the end of read transfer is asserted in OUT transfer (AXI Write transfer), or
- (4) PRD_Table is incorrect, or
- (5) Peripheral portion requests ABORT during transfer on Data-IF.

(Peripheral portion never requests ABORT as far as the operation of the software is compliant to the flow in this manual.)

The following cases are the conditions of abnormal termination of transfer due to error. But they are avoidable by the configuration of system and assumed that they never occur in the normal usage. Make sure to configure the system so that those errors never occur.

- (6) An error occurs in read process of PRD table, or
- (7) An error response is received during data transfer on AXI, or
- (8) An error occurs in writing-back of PRD Table.

In cases of (2) and (3) transfer is terminated in the middle of PRD tables. In these cases, used bit in the PRD table of the completed transfer is set. Besides, if the amount of transferred data is less than the size specified in PRD table, the size field of the PRD table is changed to the remaining amount of data after the transfer has been terminated. The actual

amount of transferred data can be calculated by subtracting the remaining amount of data from the size specified in PRD table initially.

The case of (5) occurs when clearing buffers of Peripheral portion is requested on Pn_CON.Pn_BCLR or P0_CON.P0_BCLR during transfer is being executed at Data interface. But since it is not recommended in this manual, the case of (5) is assumed to never occur as long as the software is compliant to the flow. When the case occurs, used bit and Data Error bit in the PRD table for which the transfer is being executed are set to 1 and AXI_INT_STA.PRDERRO_x_STA is asserted.

The cases of (6) and (8) mean that read or write specified by PRD Table could not be executed. In those cases, AXI_INT_STA.PRDERRO_x_STA is asserted.

For the case of (7), as far as (8) doesn't occur concurrently, used bit and Data Error bit in PRD table are set to 1 and AXI_INT_STA.PRDERRO_x_STA is asserted similarly to (5).

But in case (8) occurs at the same time, it means that writing-back of PRD table fails and the result of transfer is not stored in PRD table. Peripheral portion terminates the transfer by asserting EPC_D_DBUABRT at Data Interface with Peripheral portion.

It is assumed in the design of PERIPHERAL PORTION that it would not receive an error response on AXI-bus. Make sure to configure the system not to return an error response on AXI bus to PERIPHERAL PORTION.

63.7.5.7 Case of error

There are two kinds of errors in PERIPHERAL PORTION. One is an error due to the mistake in the setting and the other is the transfer error caused by receiving an error response on AXI bus.

But Peripheral portion was designed on the premise that it would not receive an error response (for both of read and write) on AXI bus when it executes DMA transfer, so PERIPHERAL PORTION only reports that the transfer error occurs for debug use. Make sure to configure system so that it doesn't return an error response on AXI bus to Peripheral portion.

In case an error occurs at Peripheral portion asserts EPC_D_DBTABRT at Data Interface, Peripheral portion automatically clears DMA_Ch0_CONx.PRD_ENx for the current PRD table so that the current transfer is terminated, and asserts AXI_INT_STA.PRDERRO_x_STA at the same time.

In case an error occurs at Peripheral portion, the result of the current transfer is not guaranteed. Make sure not to use transferred data when the error occurs.

63.7.5.8 Ordering of transactions

Peripheral portion operates AXI transactions in order. Hence Peripheral portion always uses the same value "H'0" as MAWID[3:0], MWID[3:0] and MARID[3:0].

63.7.5.9 Write Transfer with all zero strobe

Peripheral portion can execute two outstanding transactions and issues two outstanding addresses on AXI address channel checking the max packet size and the rest of buffer assigned by PRD table.

But, in OUT transfer (which requires the write transfer on AXI bus), the received packet might be the short one and its size might be less than both of the max packet size and the rest of buffer. In that case, Peripheral portion might have no more data to be transferred although it has already issued the transfer address on AXI write address channel. For that case, Peripheral portion executes the write transfer with all zero strobe on MWSTRB[7:0] so that no write occurs at target.

Figure 63.31 shows the case that Peripheral portion receives the short packet and it has data to transferred for ADR1 but doesn't for ADR2. Then, Peripheral portion executes the write transfer with all zero strobe on MWSTRB[7:0] for ADR2.

In Figure 63.31 the write transfer with all zero strobe starts from the head of the transfer, but MWSTRB[7:0] might be set to all zero in the middle of transfer depending on the size of the received packet.

Therefore Peripheral portion requires the target on AXI bus to accept the write transfer with all zero strobe and discard the write data for the transfer.

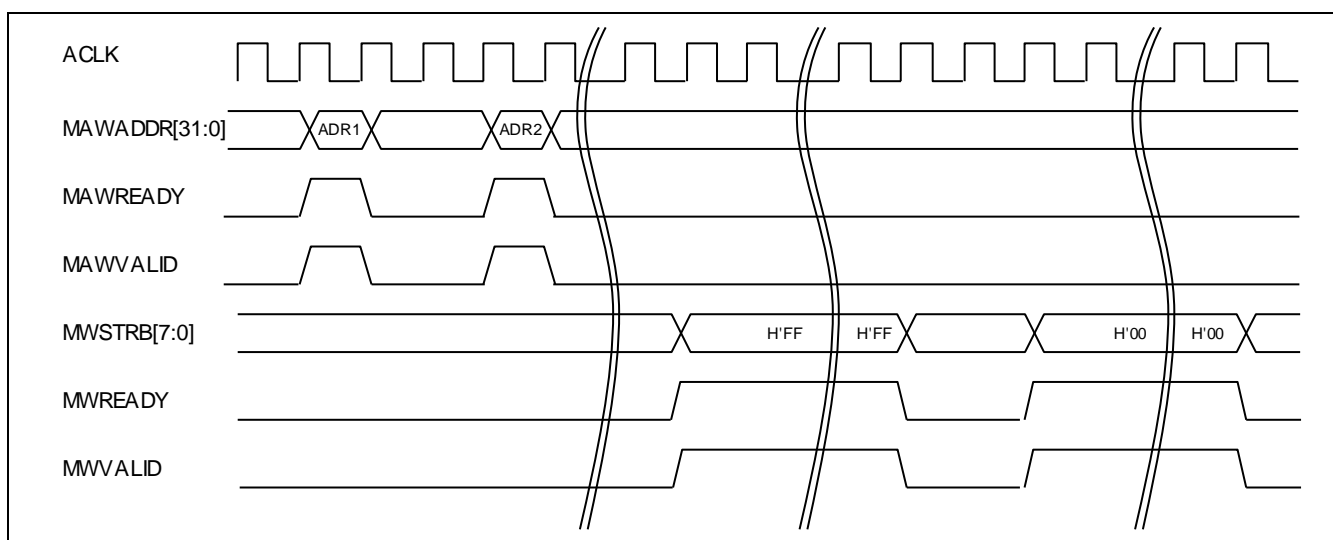


Figure 63.31 write transfer with all zero strobe

63.7.5.10 Interrupts

Pn_PRDEND_STA (n = 1 to 30) in DMA_INT_STA is asserted in the cases below.

1. PRD_ENx in DMA_Ch0_CONx (x = 1 to 4) assigned for PIPE n is cleared due to the completion of transfer. (Completion of transfer means that Lst bit in PRD table is set and data assigned to it has been transferred, the short packet has been received for OUT PIPE, PP bit is cleared in received USB3.0 data packet with non ZLP mode or that an error has occurred in transfer.)
2. PIPE which is not assigned to DMA_Ch0_CONx becomes ready. (It can accept the next packet for IN transfer, or it has new received packet for OUT transfer.)
3. Transfer of descriptor in which Int bit is set is completed.

Especially Pn_PRDEND_STA is asserted in case 2 as follows.

When PIPE n is IN pipe and DMA_Ch0_CON1 (for example) is assigned for the DMA transfer with it, Pn_PRDEND_STA is asserted at the completion of DMA transfer. If multiple pipes are enabled, DMA_Ch0_CON1 is assigned to another PIPE (PIPE m in this example) as soon as DMA transfer with PIPE n is completed. But the transmission of packets has not been completed at PIPE n. Pn_PRDEND_STA is asserted again when no DMA_Ch0_CONx is assigned to PIPE n and it has transmitted the last packet.

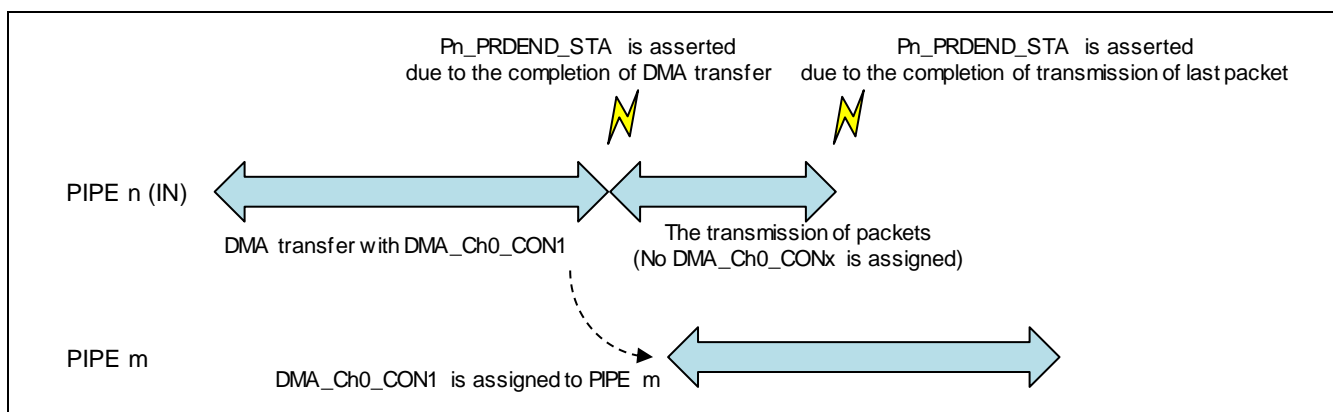


Figure 63.32 the behavior of Pn_PRDEND_STA in IN transfer

When PIPE n is OUT pipe but no DMA_Ch0_CONx is assigned to it, Pn_PRDEND_STA is asserted every time PIPE n receives a packet.

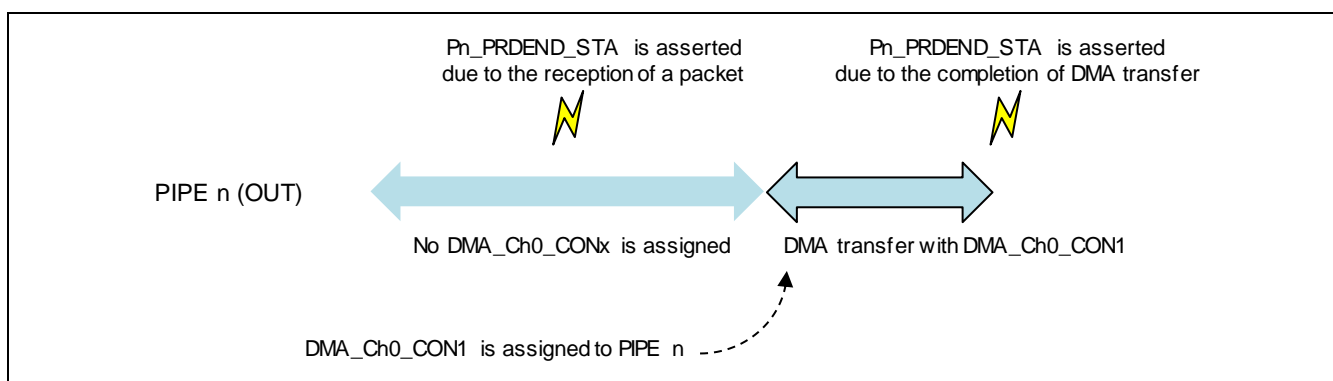


Figure 63.33 the behavior of Pn_PRDEND_STA in OUT transfer

Note that Pn_PRDEND_STA is not asserted if DMA_Ch0_CONx is assigned when the last IN packet has been transmitted or when the OUT packet has been received. As noted above, Pn_PRDEND_STA might be asserted due to the causes except for DMA transfer. When it is required to know only the completion of DMA transfer, see PRD_ENx_CLR_STA in AXI_INT_STA instead.

63.7.6 List of Interrupt factors

The interrupt factors for Peripheral portion are shown below.

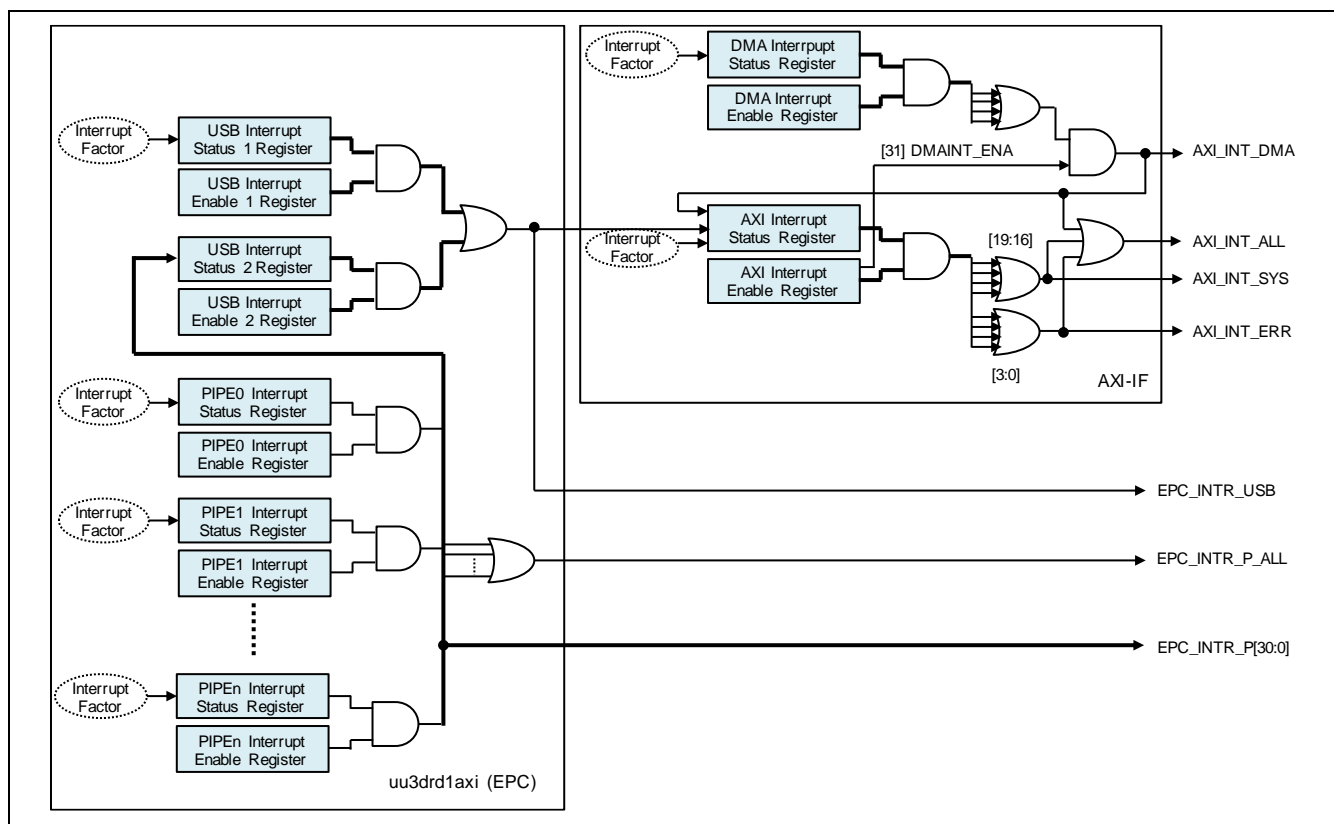


Figure 63.34 The system diagram for the interrupt

Table 63.19 Interrupt factors at USB3.0 Controller Peripheral function: AXI_INT_STA register

| Interrupt factors | Descriptions |
|-------------------|--|
| DMAINT_STA | This factor shows the status of interrupt by DMA transfer |
| EPCINT_STA | This factor shows the status of interrupt of EPC_INTR_USB. |
| PRDEN4_CLR_STA | This factor indicates that DMA_CH0_PRD_CON4.PRD_EN4 has been cleared by hardware. |
| PRDEN3_CLR_STA | This factor indicates that DMA_CH0_PRD_CON3.PRD_EN3 has been cleared by hardware. |
| PRDEN2_CLR_STA | This factor indicates that DMA_CH0_PRD_CON2.PRD_EN2 has been cleared by hardware. |
| PRDEN1_CLR_STA | This factor indicates that DMA_CH0_PRD_CON1.PRD_EN1 has been cleared by hardware. |
| PRDERR0_4_STA | This factor indicates that there was an error in PRD table assigned to DMA_CH0_CON4. |
| PRDERR0_3_STA | This factor indicates that there was an error in PRD table assigned to DMA_CH0_CON3. |
| PRDERR0_2_STA | This factor indicates that there was an error in PRD table assigned to DMA_CH0_CON2. |
| PRDERR0_1_STA | This factor indicates that there was an error in PRD table assigned to DMA_CH0_CON1. |

Table 63.20 Interrupt factors at USB3.0 Controller Peripheral function: DMA_INT_STA register

| Interrupt factors | Descriptions |
|-------------------|--|
| P[5-1]_PRDEND_STA | This factor shows the status of the interrupt by DMA transfer for PIPE[5-1]. |

Table 63.21 Interrupt factors at USB3.0 Controller Peripheral function: USB_INT_STA_1 register

| Interrupt factors | Descriptions |
|-------------------|--|
| B3_PLLWKUP_STA | This factor indicates that PHY30_CLK125_IN was resumed due to PLL wakeup (USB30_CON.B3_PLLWAKE was set or the input signal B3_PLL_WAKEIN is asserted) or U3 Wakeup LFPS from host. |
| B3_LUPSUCS_STA | This factor indicates that the link of USB3.0 has been established (LTSSM enters U0 state from Polling state). |
| B3_POLLING_STA | This factor indicates that LTSSM has made the transition to Polling state. |
| B3_INACTV_STA | This factor indicates that LTSSM has made the transition to Inactive state. |
| B3_DISABLE_STA | This factor indicates that LTSSM has made the transition to Disabled state. |
| B3_VNDTST_STA | This factor indicates that Vendor Device Test LMP has been received. |
| B3_U2INACT_STA | This factor indicates that U2 Inactivity Timeout LMP has been received. |
| B3_SETLNK_STA | This factor indicates that Set Link Function LMP has been received. |
| B3_LNKCNG_STA | This factor indicates that LTSSM has made the transition as listed below. U0 -> U1 U0 -> U2 U0 -> U3 U3 -> Recovery -> U0 U2 -> Recovery -> U0 U1 -> Recovery -> U0 |
| B3_WRM_RST_STA | This factor indicates that warm reset has been received. |
| B3_HOTRST_STA | This factor indicates that hot reset has been received. |
| B3_PSFAIL_STA | This factor indicates that the transition of POWERDOWN mode on USB30_CON.POW_SEL[2:0] failed. |
| B3_PSSUCS_STA | This factor indicates that the transition of POWERDOWN mode on USB30_CON.POW_SEL[2:0] has been completed successfully. |
| B3_U12REQ_STA | This factor indicates that PERIPHERAL PORTION received LGO_U1/U2 or when U1/U2_TIMER in SSIF was expired. |
| B3_TPSUCS_STA | This factor indicates that TP (Device Notification) requested on USB30_CON.B3_TP_SEND has been successfully completed. |
| B2_LPMRCV_STA | This factor indicates that PERIPHERAL PORTION has received LPM request from USB2.0 host and has returned ACK to it. |
| B2_USBRST_STA | This factor indicates that USB2.0 bus reset has been received. |
| B2_L1SPND_STA | This factor indicates that L1 suspend request has been received from USB2.0 host. |
| B2_L1RSUM_STA | This factor indicates that PLL wakeup has been requested (USB20_CON.B2_SUSPEND = 0 or B2_PLL_WAKEIN = 1) or when the resume request from USB2.0 host has been received (USB_STA.B2_L1RSUM_OUT changes from 0 to 1) during L1 state (USB_STA.B2_L1_SPND_OUT = 1). |
| B2_SPND_STA | This factor indicates that the suspend request has been received from USB2.0 host and USB_STA.B2_SPND_OUT changes from 0 to 1. |
| B2_RSUM_STA | This factor indicates that PLL wakeup has been requested (USB20_CON.B2_SUSPEND=0 or B2_PLL_WAKEIN = 1) or when the resume request from USB2.0 host has been received (USB_STA.B2_RSUM_OUT changes from 0 to 1) during L2 state (USB_STA.B2_SPND_OUT = 1) or L3 state (USB30_CON.B3_CONNECT = 0). |

| Interrupt factors | Descriptions |
|-------------------|--|
| SPEED_STA | This factor indicates that there was change in the speed mode of USB. |
| VBUS_CNG_STA | This factor indicates that the level of the input signal "VBUS" changed. |

Table 63.22 Interrupt factors: USB_INT_STA_2 register

| Interrupt factors | Descriptions |
|-------------------|--|
| PIPE[5:0]_INT_STA | This factor indicates that an interrupt related to PIPE[5-0] occurred. |

Table 63.23 Interrupt factors: P0_INT_STA register

| Interrupt factors | Descriptions |
|-------------------|---|
| P0_STSED_STA | This factor indicates that status stage of control transfer has been completed successfully. |
| P0_STSST_STA | This factor indicates that status stage of control transfer has been started at PIPE0. |
| P0_SETUP_STA | This factor indicates that valid setup data has been received at PIPE0. |
| P0_RCVNL_STA | This factor indicates that null data has been received at PIPE0. |
| P0_ERDY_STA | This factor indicates that the condition to send ERDY was satisfied at PIPE0. |
| P0_FLOW_STA | This factor indicates that PIPE0 has entered flow control state. |
| P0_STALL_STA | This factor indicates that STALL has been sent at PIPE0. |
| P0_NRDY_STA | This factor indicates that NRDY or NAK has been sent at PIPE0. |
| P0_BFRDY_STA | In case the direction of transfer is IN (P0_MOD.P0_DIR = 1), the next data can be written to P0_WRITE register. In case the direction of transfer is OUT (P0_MOD.P0_DIR = 0), the next data can be read from P0_READ register. |

Table 63.24 Interrupt factors: Pn_INT_STA register

| Interrupt factors | Descriptions |
|-------------------|--|
| Pn_CBW_STA | This factor indicates that CBW has been received at PIPE _n and is ready to be read from Pn_READ register. |
| Pn_STERR_STA | This factor indicates that Stream bulk protocol error has occurred at PIPE _n . |
| Pn_STRMX_STA | This factor indicates that Stream ID in received packet was different from that in Pn_STREAM.Pn_STREAM_C. |
| Pn_NSTRM_STA | In case the direction of transfer is IN (Pn_MOD.Pn_DIR==1), this factor indicates that ACK_TP(NoStream) has been received at PIPE _n . In case the direction of transfer is OUT (Pn_MOD.Pn_DIR==0), this factor indicates that DP(NoStream) has been received at PIPE _n . |
| Pn_PRIME_STA | In case the direction of transfer is IN (Pn_MOD.Pn_DIR==1), this factor indicates that ACK_TP(PRIME) has been received at PIPE _n . In case the direction of transfer is OUT (Pn_MOD.Pn_DIR==0), this factor indicates that DP(PRIME) has been received at PIPE _n . |
| Pn_RCVNL_STA | This factor indicates that zero length packet has been received at PIPE _n . |
| Pn_ERDY_STA | This factor indicates that the condition to send ERDY has been satisfied at PIPE _n . |
| Pn_FLOW_STA | This factor indicates that PIPE _n has entered flow control state. |
| Pn_LSTTR_STA | In case the direction of transfer is IN (Pn_MOD.Pn_DIR==1), this factor indicates that the last data of the transfer (short packet, packet for which Pn_LAST was set or packet for which EPC_D_DBWTRANSEND was asserted) has been sent at PIPE _n . In case the direction of transfer is OUT (Pn_MOD.Pn_DIR==0), this factor indicates that the last data of the transfer (short packet or packet with PP = 0) has been received at PIPE _n . |
| Pn_STALL_STA | This factor indicates that STALL has been sent at PIPE _n . |

| Interrupt factors | Descriptions |
|--------------------------|---|
| Pn_NRDY_STA | This factor indicates that NRDY or NAK has been sent at PIPEn. |
| Pn_BFRDY_STA | In case the direction of transfer is IN (Pn_MOD.Pn_DIR==1), the next data can be written to Pn_WRITE register. In case the direction of transfer is OUT (Pn_MOD.Pn_DIR==0), the next data can be read from Pn_READ register. |

63.7.7 Zero Length Packet (ZLP)

63.7.7.1 Definition of Zero Length Packet

Zero length packet (ZLP) is the data packet (DP) but it has no data word in its data payload. ZLP is used to show the boundary of the transfer in some USB device classes, in case that the size of the last data packet equals to the max packet size of the PIPE.

Short packet, which length of data is less than the max packet size of the PIPE, is interpreted as the last packet of transfer in general. For example, it is defined that the size of packet except the last one in bulk transfer should be the max packet size of the PIPE. But it differs how to treat the case that the size of the last packet of transfer equals to the max packet size of the PIPE, depending on USB device class.

It is difficult to know whether the received packet which has the length same as the max packet size is the last one or not for the receiver (host or device). In some USB device classes, ZLP is sent next to show that there is the boundary of transfer. But in Bulk Only Transfer (BOT), ZLP is not used because CBW notifies the amount of transfer in advance. The device might transmit data less than the amount shown in CBW and the size of the last data packet might be the max packet size, but CSW is transferred next even in that case and CSW of 13 bytes is the short packet in bulk transfer. Therefore host can know that the packet just before CSW was the last data packet. In Stream bulk transfer, the packet with PP = 0 to show the end of transfer related the current Stream ID.

See the specification of USB device class to know whether ZLP is really required or not.

63.7.7.2 Transfer of ZLP support mode

PERIPHERAL PORTION has the mode to support ZLP transfer. Set USB_COM_CON.Pn_WDATAIF_NL, USB_COM_CON.Pn_RDATAIF_NL and USB_COM_CON.Pn_LSTTR_PP to 1 in order to use the mode.

63.7.7.3 In the configuration with AXI-IF

In the configuration with AXI-IF, Peripheral portion sends ZLP for OUT transfer and receives

ZLP for IN transfer automatically with the settings for them.

In order to transmit ZLP in ZLP transfer, the software sets ZL bit in PRD table where ZLP is requested to be transmitted in advance. When Peripheral portion receives ZLP in OUT transfer, it sets ZL bit in PRD table where ZLP is received.

USB_COM_CON.Pn_WDATAIF_NL, USB_COM_CON.Pn_RDATAIF_NL and USB_COM_CON.Pn_LSTTR_PP should be set to 1 as noted in section 63.7.7.2.

For IN transfer, with PRD table in which Lst = 1 and ZL = 1, Peripheral portion sends ZLP after the last data packet of the PRD table in order to show the boundary of transfer. With PRD table in which Lst = 0 or ZL = 0, Peripheral portion never sends ZLP.

Therefore, set both of Lst bit and ZL bit to 1 in the PRD table which has the last data packet and requires ZLP after it. In other PRD tables, set both of Lst bit and ZL bit to 0.

Note that the meaning of H'0000 in SIZE field of PRD table differs depending on ZL bit and Lst bit. If ZL = 1 and Lst = 1, H'0000 in SIZE field means that 0 byte transfer is requested by the PRD table. Otherwise, H'0000 means that 64 Kbytes transfer is requested.

If ZLP is required after 64 Kbytes data has been transferred, prepare two PRD tables. The first one has 64 Kbytes data (SIZE field is H'0000) with Lst = 0 and ZL = 0, and the second one has 0 bytes data (SIZE field is H'0000) with Lst = 1 and ZL = 1.

For OUT transfer, the software is never required to set Lst bit and ZL bit in PRD table. When Peripheral portion receives ZLP in OUT transfer, it sets both of Lst bit and ZL bit to 1 in the PRD table. If Peripheral portion has already received some packets before ZLP, Peripheral portion transfers them to system memory where the PRD table designates, then it sets Lst bit and ZL bit in the PRD table.

If Peripheral portion doesn't receive ZLP but short packet (which length is not 0), it sets Lst bit to 1 but leave ZL bit as 0.

Note that the meaning of H'0000 in SIZE field of PRD table differs depending on ZL bit and Lst bit. If ZL = 1 and Lst = 1, H'0000 in SIZE field means that 0 byte data is contained in the buffer designated by the PRD table. Otherwise, H'0000 means 64 Kbytes data is contained in the buffer.

If ZLP is received after 64Kbytes data has been received, two PRD tables are used. The first one shows SIZE = H'0000, Lst = 0 and ZL = 0, and it means that 64 Kbyte data has been received but ZLP isn't included. The second one shows SIZE = H'0000, Lst = 1 and ZL = 1, and it means that only ZLP has been received for the PRD table.

63.7.8 Physical Region Descriptor (PRD) Table

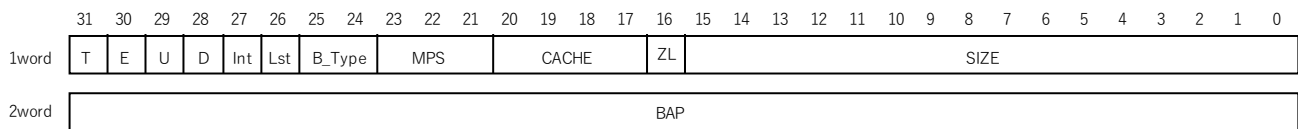
Use Physical Region Descriptor (PRD) Table for source/destination Address and transfer size of DMA Transfer in AXI Master-IF and setting required for applicable DMA Transfer. Creation of PRD Table on memory area and setting address of the created PRD Table shall be done by the Software (FW). And PRD Table shall be created for each PIPE. Make sure to locate PRD Table in the area where the table does not cross over 4KB Address boundary.

Format of PRD Table and the definition of each field are described below.

[PRD Table Format]

Format of PRD Table shall be compliant with ASIC-PF (SS Development Guide) “Descriptor Format A” and it is defined as below. The 3rd word is omitted and the rest of two words will form a descriptor.

PRD Table Format



Details of PRD Table

[1 word]

| Bit | Field Name | Update | Description |
|-----|------------|---------|---|
| 31 | T | FW — | Type bit that indicates type of descriptor. 0: Buffer Pointer 1: Link Pointer |
| 30 | E | FW — | End bit that indicates the End of Descriptor Chain. Specifies 0 in case of a Link Pointer. 0: Other than the above. 1: Indicates the End of Chain. |
| 29 | U | — HW | Used bit that indicates Transfer of the applicable descriptor is completed. Specifies 0 in case of PRD Table generated by FW 0: Indicates incomplete transfer or transfer in progress. 1: Indicates completion of transfer |
| 28 | D | — HW | Data Error bit that indicates occurrence of an Error during Transfer. Specify 0 in case of PRD Table generated by FW. 0: No Error occurred 1: Error occurred |

| Bit | Field Name | Update | Description |
|-------------|------------|---------|---|
| 27 | Int | FW — | <p>Specifies existence of occurrence of Interrupt due to Transfer at applicable descriptor.</p> <p>In case this bit is set to 1, transfer at applicable descriptor is regarded to be completed and DMA_INT_STA.Pn_PRDEND_STA is set to 1.</p> <p>Specify 0 in case of a Link Pointer.</p> <p>0: Interrupt not occurred 1: Interrupt occurred</p> <p>In case Short Packet is received in OUT Transfer, an interrupt is set as transfer of PRD Table is completed regardless of setting of this bit.</p> |
| 26 | Lst | FW — | <p>Specifies the last packet for transfer at applicable descriptor</p> <p>Assert EPC_D_DBWTRANSEND signal when writing the last data in case this bit is set to 1 in WRITE (PIPE for In Transfer) to EPC in order to indicate the last data for Transfer (same Stream ID). Set the interrupt status for EPC, where is Pn_INT_STA.Pn_LSTTR_STA bit, when the last packet including the data asserted EPC_D_DBRTRANSEND is completed.</p> <p>Specify 0 in case of READ (PIPE for OUT Transfer PIPE) to EPC. This bit is set to 1 by HW when the transfer is completed with asserting EPC_D_DBRTRANSEND at DATA-IF. This bit is specified 0 in case of a Link Pointer.</p> <p>0: Other than the above 1: Last Packet</p> |
| 25, 24 | B_Type | FW — | <p>Indicates burst type.</p> <p>Specifies 00 in case of a Link Pointer.</p> <p>B'00: Fixed Burst B'01: Incrementing burst B'10: <i>Reserved</i> B'11: <i>Reserved</i></p> |
| 23 to 21 | MPS | FW — | <p>Indicates Max Packet Size / Unit of transfer in Data-IF.</p> <p>Specify 00 in case of a Link Pointer.</p> <p>B'000: 8Byte / B'001: 16Byte / B'010: 32Byte / B'011: 64Byte / B'100: 512Byte / B'101: 1024Byte / others: <i>Reserved</i></p> |
| 20 to 17 | CACHE | FW — | <p>Specify CACHE signal (MARCACHE/MAWCACHE) during DATA Transfer Access.</p> |
| 16 | ZL | FW HW | <p>Set this bit to 1 for USB device class which uses ZLP.</p> <p>0: ZLP is not transferred through Data Interface or on AXI DMA transfer. 1: ZLP is transferred through Data Interface or on AXI DMA transfer</p> <p>Notes: 1 If T bit shows link pointer, set this bit to 0. 2 If the bits are not set to 1 but ZLP is used, it is treated as PRD table format error and AXI_INT_STA.PRDERRO_x_STA.</p> |

| Bit | Field Name | Update | Description |
|---------|------------|----------|--|
| 15 to 0 | SIZE | FW HW | <p>Indicates number of byte to be transferred.</p> <p>This is updated to “the number of remaining bytes to transfer” by HW when transfer with EPC in a packet is completed during transfer.</p> <p>Set value can be specified to a value between 1 and 64 Kbyte in steps of a byte. Set H'0000 to specify 64 Kbyte. SIZE specified for a descriptor shall be integral multiple of a Packet in USB Transfer. Multiple Descriptors cannot be used to compose 1 USB Packet.</p> <p>Transfer is stopped automatically when short Packet is received during OUT (AXI-Write) transfer. Setting values shall be integral multiple of Max Packet Size in USB packet.</p> <p>The remain is transmitted as Short Packet in USB Transfer when specified SIZE is not integral multiple of Max Packet Size during In (AXI-Read) Transfer.</p> <p>Specify 0 in case of a Link Pointer.</p> |

[2word]

| Bit | Field Name | Update | Description |
|---------|------------|----------|--|
| 31 to 0 | BAP | FW HW | <p>In case Type bit is Buffer Pointer: Indicates Storage Address of transfer Data.</p> <p>In case Type bit is Link Pointer: Indicates address of the next descriptor to be located. Make sure that lower three bits should be "0" at the address of the next descriptor to be located.</p> <p>In case it is a Descriptor Chain Buffer Pointer, address of the next descriptor shall be located at continuous address. In case Type bit is a Buffer Pointer, address of the next descriptor shall be located at continuous address.</p> <p>This is updated by HW when transfer with EPC in a unit of a packet is completed during transfer. The value in this field is invalid if the transfer is stopped due to reception of short packet or occurrence of transfer error.</p> |

Int bit is used to generate a PRDEND Interrupt upon completion of transfer of the set descriptor.

This is used to start creation of PRD Table for the next process prior to the completion of current process of PRD Table in cases such as CPU with slow response speed for an Interrupt.

[Setting and validating PRD Table]

FW generates a PRD Table that specifies destination of data transfer and parameters for transfer for each PIPE on memory. PRD Table is valid and Transfer is enabled when address of the generated PRD table, applicable PIPE number, transfer direction of the PIPE in one of the four PRD Table Setting Area Registers provided in AXI-IF and PRD_EN is 1 and EXTPRD_EN is 0 after that

[Set the following register to set PRD Table]

| Register | Field | Description |
|-----------------|---------|---|
| DMA_Ch0_PRD_ADR | PRD_ADR | Specifies the Initial Address of PRD Table. |

| Register | Field | Description |
|-------------|-----------|--|
| DMA_Ch0_CON | PIPE_DIR | Specifies transfer direction of the PIPE specified in PIPE_NO. |
| | PIPE_NO | Specifies PIPE number. |
| | EXTPRD_NO | This is specified only when setting area to be linked exists. |
| | EXTPRD_EN | This is set to 1 only when setting area to be linked exists. |
| | PRD_EN | This is set to 1 to allow (enable) the Transfer that was specified in PRD Table. |

63.8 Example of Software flow

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

63.8.1 Outline of software flow

The Software for PERIPHERAL PORTION is described in this section. The flow includes initialization, interrupt handling, control transfer process, bulk transfer process and interrupt transfer process.

Note that the flow described here is an example in general usage of PERIPHERAL PORTION. Modification of the flow or additional operation might be required depending on application in which user’s system assumes.

The loop as below is used at some points in the flow and assumes the read polling of register by software. The interrupt notification might be used instead, but note that the setting to enable the interrupt is required in that case.

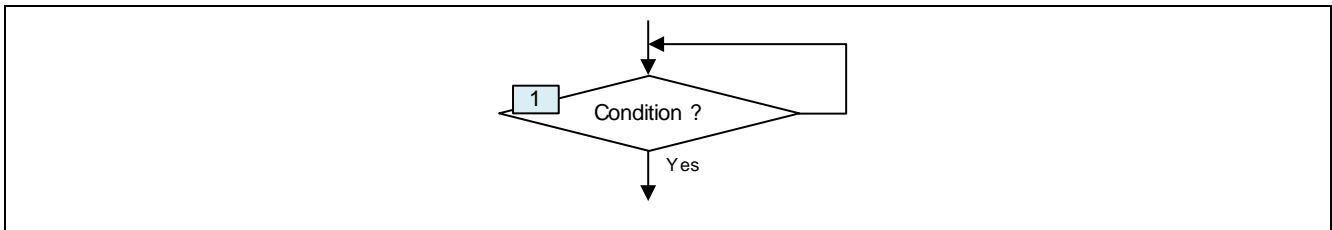


Figure 63.35 The read polling of register by software

In order to access registers related to PIPEn, the index of PIPE should be set in PIPE_COM.PIPE_NUM[4:0] in advance. Note that the setting of the index of PIPE is omitted in each step of this section.

63.8.2 Initialization

The initialization should be completed before the device is used as USB device.

It is recommended that the initialization is done immediately after the power to device is turned on.

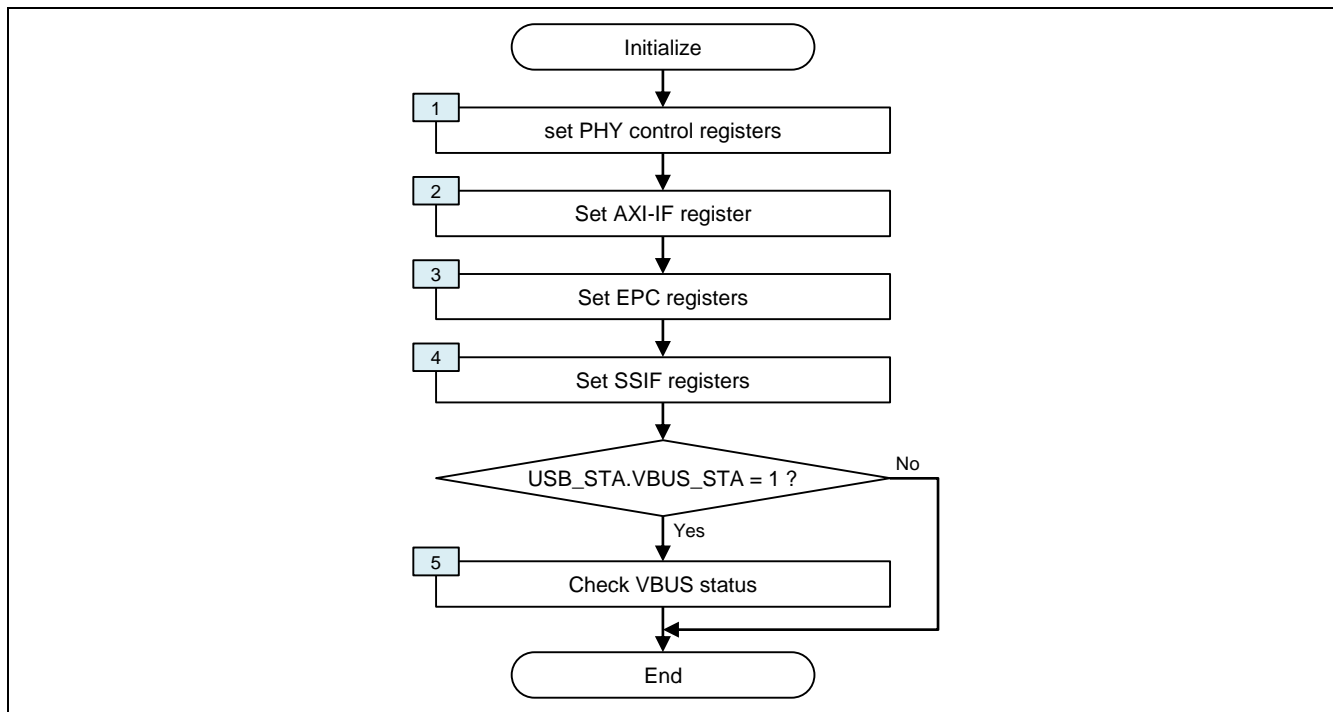


Figure 63.36 Initialization flow

(1) Set PHY register

The setting of Reference clock setting & SSC setting is changed here if default values of the parameters for Reference clock & SSC below are not consistent with the user's system.

- USB3S0_CLK_P/USB3S0_CLK_M used
 - USB30_CLKSET0. USB30_FSEL[5:0]: B'10_0111
 - USB30_CLKSET1. USB30_PLL_MULTI [7:0]: B'000_0000
 - USB30_CLKSET1. USB30_REF_CLKDIV: B'0
 - USB30_CLKSET1. USB30_REF_USB_SEL: B'1
- USB_XTAL/USB_EXTAL used
 - USB30_CLKSET0. USB30_FSEL[5:0]: B'00_0010
 - USB30_CLKSET1. USB30_PLL_MULTI [7:0]: B'110_0100
 - USB30_CLKSET1. USB30_REF_CLKDIV: B'1
 - USB30_CLKSET1. USB30_REF_USB_SEL: B'0
- SSC used
 - USB30_SSC_SET. SSC_RANGE [2:0]: setting appropriate value for user system
 - USB30_SSC_SET. SSC_EN: B'1
- Peripheral function used
 - USB30_VBUS_EN. VBUS_EN: B'1

After the above setting,

- Write 0x0010 to USB30_PHY_ENABLE
- Write 0x191A to USB30_CLKSET1
- Wait 10us
- Write 0x190A to USB30_CLKSET1

(2) Set AXI-IF registers

The setting of AXI-Bridge is changed here if default values of the parameters for AXI-bus below are not consistent with the user's system.

- AXI_CON.MST_BSIZE[1:0] (Burst size setting: default value 8 bytes)

(3) Set EPC registers

Set the operation mode and initialize the interrupt status in EPC as follows.

a) USB30_CON register

If U3_POW_SEL[1:0] is required to be changed from default settings, change them here.

b) USB_INT_STA_1 register

Write 1 to all status bits in USB_INT_STA_1 in order to clear them.

c) USB_INT_ENA_1 register

Set VBUS_CNG_ENA to 1 in order to enable VBUS change interrupt.

It is also recommended to enable B3_HOTRST_ENA. See section 63.8.3.5 for details.

(4) Set SSIF registers

The setting of SSIF is changed here if required. But it is recommended to use them with default settings basically.

(5) Check VBUS status

Check whether the status of USB_STA.VBUS_STA has already changed to 1 or not.

If USB_STA.VBUS_STA==1, proceed to the operation of connection.

63.8.3 Interrupt process

When an interrupt is asserted, see the interrupt status register to know what factor causes the interrupt, and then do the operation required.

Examples of the flow for interrupt are shown in this section.

63.8.3.1 Interrupt handler process

The process in interrupt handler when an interrupt is asserted is described here.

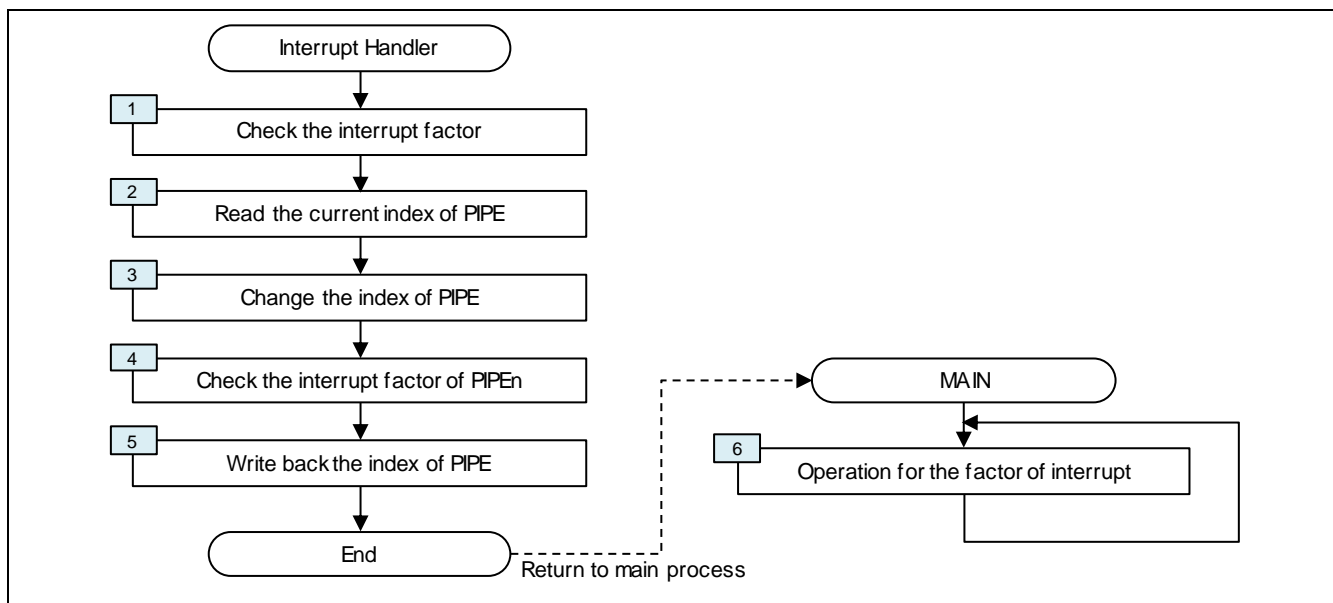


Figure 63.37 Interrupt handler process flow

- (1) Check the interrupt factor
See interrupt status register to know what factor is the cause of the interrupt. Then clear the status bit in interrupt status register.
It is assumed here that the interrupt is asserted due to PIPE_n event, so registers related to PIPE_n are accessed later. If it is required to remember the interrupt factor in main process, prepare the variable to remember it.
- (2) Read the current index of PIPE
The software should read the current index of PIPE and remember it before it changes the index of PIPE in PIPE_COM.PIPE_NUM[4:0] due to the interrupt asserted by PIPE_n event and the succeeding operation for PIPE_n. The saved index of PIPE is written back at (5) if required.
- (3) Change the index of PIPE
Change the index of PIPE in PIPE_COM.PIPE_NUM[4:0] to that of PIPE which generates the interrupt.

- PIPE_COM.PIPE_NUM[4:0] = the index of PIPE
- (4) Check the interrupt factor of PIPE_n
See the interrupt status register for PIPE_n (P_n_INT_STA) to know what factor at PIPE_n is the cause of the interrupt. Note that P_n_INT_STA register shows the interrupt status of PIPE which is designated by the index of PIPE in PIPE_COM.PIPE_NUM[4:0].
Then clear the status bit in interrupt status register.
If it is required to remember the interrupt factor in main process, prepare the variable to remember it.

(5) Write back the index of PIPE

When the software exits the interrupt handler, write back the saved index of PIPE at (2) to PIPE_COM.PIPE_NUM[4:0] if required.

- PIPE_COM.PIPE_NUM[4:0] = the saved index of PIPE at (2)

(6) Operation for the factor of interrupt

The firmware does the operation required for the factor of interrupt after it returns to main process. In order to remember the factor, use the variable as noted at (1) and (4). After completing the operation, initialize the variable.

Especially note that it is prohibited to change the index of PIPE until read of the whole packet from P0_READ/Pn_READ or write of it to P0_WRITE/Pn_WRITE is completed.

If the interrupt due to PIPE_n event is asserted (it can be known by USB_INT_STA_2 register) but read of the whole packet from P0_READ/Pn_READ or write of it to P0_WRITE/Pn_WRITE has been started, wait for the completion of read or write of the packet. Then change PIPE_COM.PIPE_NUM[4:0] to see the factor of interrupt at PIPE_n.

See section 63.9.2 for details.

63.8.3.2 USB Connection

The operation required for USB connection is described here.

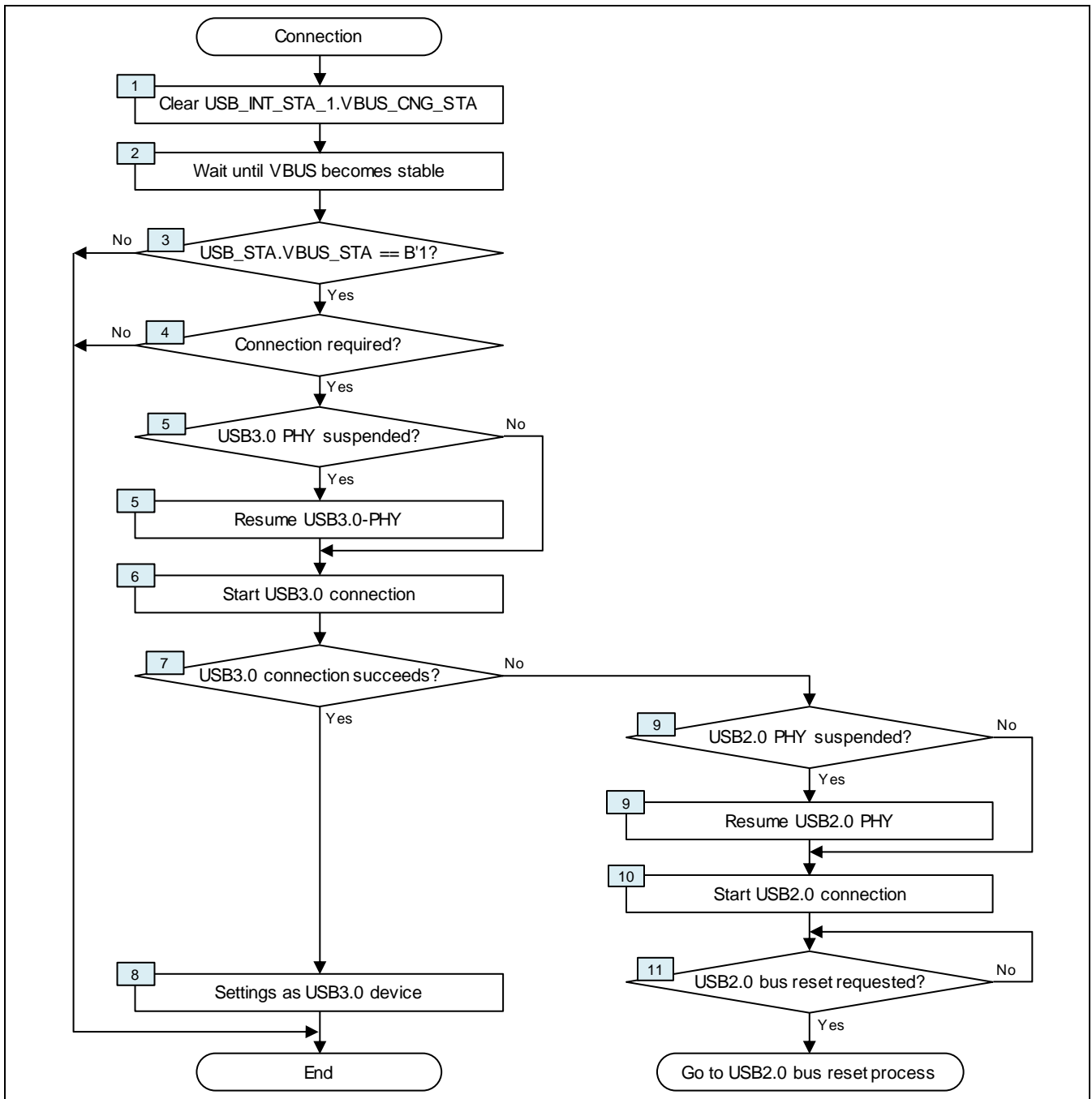


Figure 63.38 USB connection flow

(1) Clear USB_INT_STA_1.VBUS_CNG_STA

When VBUS status changes, USB_INT_STA_1.VBUS_CNG_STA is asserted. It is notified on an interrupt if enabled. If it is asserted, write 1 to USB_INT_STA_1.VBUS_CNG_STA to clear.

- USB_INT_STA_1.VBUS_CNG_STA=1 (Clearing the event that the VBUS status is changed)

(2) Wait until VBUS becomes stable

In this subsection the device is assumed to be in disconnected state first. Therefore, it is required to check VBUS status in order to know if the device is really connected or not, because the change of VBUS status might be caused by noise.

To know VBUS status, USB_STA.VBUS_STA is available. If it shows 1, VBUS voltage level is valid and it means that the device is connected at present.

But VBUS voltage level is supposed to be unstable due to chattering just after the device is connected to host or hub. VBUS level will swing between 1 and 0 (therefore, USB_STA.VBUS_STA will also swing) for a while. During the chattering time, it is difficult to know the status of USB connection. Therefore, it is required to wait for VBUS to become stable. Wait for several dozens of milli-second in order to remove the influence of chattering of VBUS level.

If USB_INT_ENA_1.VBUS_CNG_ENA is enabled during the chattering time, there might be many interrupts caused by the swing of VBUS level. Disable USB_INT_ENA_1.VBUS_CNG_ENA to prevent that the software is annoyed at the swing of VBUS level for the chattering time.

- USB_INT_ENA_1.VBUS_CNG_ENA = 0 (The change of VBUS is masked)

(3) Is USB_STA.VBUS_STA == 1?

Check USB_STA.VBUS_STA to know whether the interrupt is caused by the change of VBUS level or merely noise. If USB_STA.VBUS_STA == 1, it means new USB connection. But if it shows 0, the interrupt detected at (1) would be caused by noise.

[Note]

If you find frequently that your device is unconnected even when you have confirmed USB_STA.VBUS_STA == 1 at this point, increase the wait time at (2). In that case, it might take more time to stabilize VBUS level in your system.

(4) Is connection required?

If device requires to be connected, proceed to (5) to request the connection.

It is possible to insert the grace time here before the connection. The software can use the grace time for the preparation of the connection or other use.

(5) Resume USB3.0-PHY

If USB3.0-PHY is in the power down state and PLL is suspended, resume USB3.0-PHY so that it provides USB3.0 clock.

See section 63.8.8.6 for details.

(6) Start the operation for USB3.0 connection

Start the connecting operation as USB3.0 device when it is confirmed that device is connected to host.

- USB30_CON.B3_CONNECT = 1 (Starting the connection process for SS)

- USB_COM_CON.EP0_EN = 1 (PIPE0 is enabled)

- USB_COM_CON.SPD_MODE = 0 (SS mode is requested)

- SSIFLINKSET3.SSIF_NUM_RDDET[3:0] = H'8 (The number of trial for Rx.Detect = 8)

- USB_COM_CON.RX_DETECTION = 1 (RxDetect is started)

- USB_INT_ENA_1.B3_LUPSUCS_ENA = 1 (Interrupt due to the success of link up is enabled)

- USB_INT_ENA_1.B3_DISABLE_ENA = 1 (Interrupt due to the failure of link up is enabled)

Besides, if USB_INT_ENA_1.SPEED_ENA is enabled, the interrupt that SPEED state of USB is changed is asserted after LTSSM enters U0 as SS device or bus reset and chirp are completed as HS or FS device.

- USB_INT_ENA_1.SPEED_ENA = 1 (Interrupt of SPEED state is enabled)

(7) Does USB3.0 connection succeed?

As the result of the trial for USB3.0 connection, one of USB_INT_STA_1.B3_LUPSUCS_STA or USB_INT_STA_1.B3_DISABLE_STA is asserted. USB_INT_STA_1.B3_LUPSUCS_STA means that the link up process of LTSSM has been successfully done and LTSSM is in U0. USB_INT_STA_1.B3_DISABLE_STA means that the link up process of LTSSM has failed and LTSSM is in SS.Disabled. When they are asserted, an interrupt is generated if enabled.

- Check USB_INT_STA_1.B3_LUPSUCS_STA== 1 or USB_INT_STA_1.B3_DISABLE_STA== 1

After the confirmation above, write 1 to USB_INT_STA_1.B3_LUPSUCS_STA or USB_INT_STA_1.B3_DISABLE_STA to clear them.

- Write 1 to USB_INT_STA_1.B3_LUPSUCS_STA if asserted (Clearing the event that the link up has succeeded)

- Write 1 to USB_INT_STA_1.B3_DISABLE_STA if asserted (Clearing the event that USB3.0 connection has failed)

If USB_INT_STA_1.B3_LUPSUCS_STA is asserted, it means USB3.0 connection is successful, proceed to (8).

If USB_INT_STA_1.B3_DISABLE_STA is asserted, it means USB3.0 connection has failed, proceed to (9).

(8) Transition to default state

After the connecting operation has been successfully completed, enable the detection of control transfer at PIPE0 (endpoint 0). At the same time, the interrupts required for the operation mode should be enabled. An example for those cases are shown below. If there are other interrupts required for the usage of core, enable them as well.

- USB_INT_ENA_2.PIPE0_INT_ENA = 1 (Interrupt at PIPE0 is enabled)

- All bits in P0_INT_ENA register should be enabled (All interrupt factors for PIPE0 are enabled)

If the interrupt due to the change of USB SPEED state is enabled at (6), the interrupt is asserted when USB3.0 or USB2.0 connection is established. See USB_STA.SPEED[1:0] to know which speed the device is acting as. Then set the bits below according to the speed state.

In this state, the device is acting as USB3.0 device.

- USB_INT_ENA_1.B3_HOTRST_ENA = 1 (Interrupt due to hot reset is enabled)

- USB_INT_ENA_1.B3_WRM_RST_ENA = 1 (Interrupt due to warm reset is enabled)

(9) Resume USB2.0-PHY

If USB2.0-PHY is in the power down state and PLL is suspended, resume USB2.0-PHY so that it provides USB2.0 clock.

See section 63.8.8.10 for details.

(10) Start USB2.0 connection

Start the connecting operation as USB2.0 device. Note that USB30_CON.B3_CONNECT is still 1. USB3.0 connection might be tried again after USB2.0 bus reset.

- USB_COM_CON.EP0_EN = 1 (PIPE0 is enabled: this field is already enabled at (6))

- USB_COM_CON.SPD_MODE = 1 (USB2.0 mode is requested)

- USB20_CON.B2_PUE = 1 (Pull up on D+ is enabled)

- USB20_CON.B2_CONNECT = 1 (Starting the connection process for USB3.0)

(11) Is USB2.0 bus reset requested?

When USB2.0 bus reset comes, go to USB2.0 bus reset process. See section 63.8.3.6 for details.

63.8.3.3 USB Disconnection

The operation required for USB disconnection is described here.

(1) Is `USB_STA.VBUS_STA == 0`?

If you find that `USB_STA.VBUS_STA == B'0` (it notifies on `USB_INT_STA_1.VBUS_CNG_STA` for example), the device is disconnected.

(2) Initialize the settings

If the device is disconnected, initialize the settings below.

- `USB20_CON.B2_PUE = 0` (D+ pull up is disabled)
- `USB20_CON.B2_CONNECT = 0` (the request of connection as HS or FS device is disabled)
- `USB30_CON.B3_CONNECT = 0` (the request of connection as SS device is disabled)
- `USB_COM_CON.COF = 0` (the device is in unconfigured state)

PIPEs should be initialized as well.

- `USB_COM_CON.PIPE_CLR = 1` (PIPE is initialized)

* Make sure that `USB_COM_CON.PIPE_CLR` returns to 0 after setting `USB_COM_CON.PIPE_CLR = 1`.

(3) Place PHYs in low power states

It is recommended to place USB3.0-PHY and USB2.0-PHY in low power state and PLLs in them are suspended in order to reduce the power consumption.

If you stop USB3.0-PHY(PLL), refer to section 63.8.8.1 (3), (4), (5). In this case, “U3 state” should be replaced with “disconnected state(SS.Disabled)”. Note that the values of `USB30_CON.POW_SEL[2:0]` are different in U3 and disconnected state. See section 63.2.2.20 for details.

If you stop USB2.0-PHY(PLL), refer to section 63.8.8.7 (2).

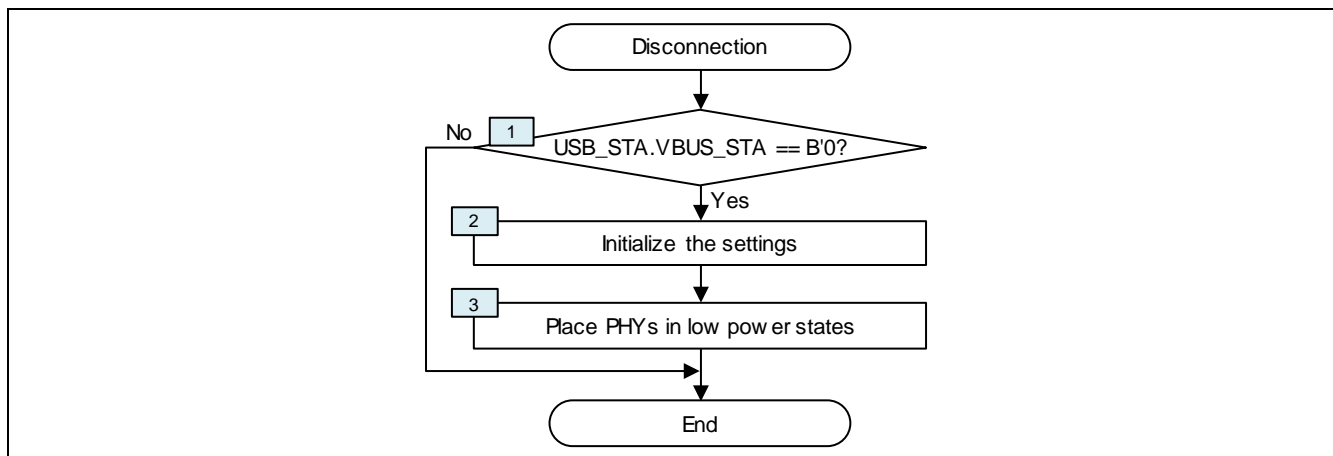


Figure 63.39 USB disconnection flow

63.8.3.4 Warm Reset Process

The operation required when warm reset is detected is described here.

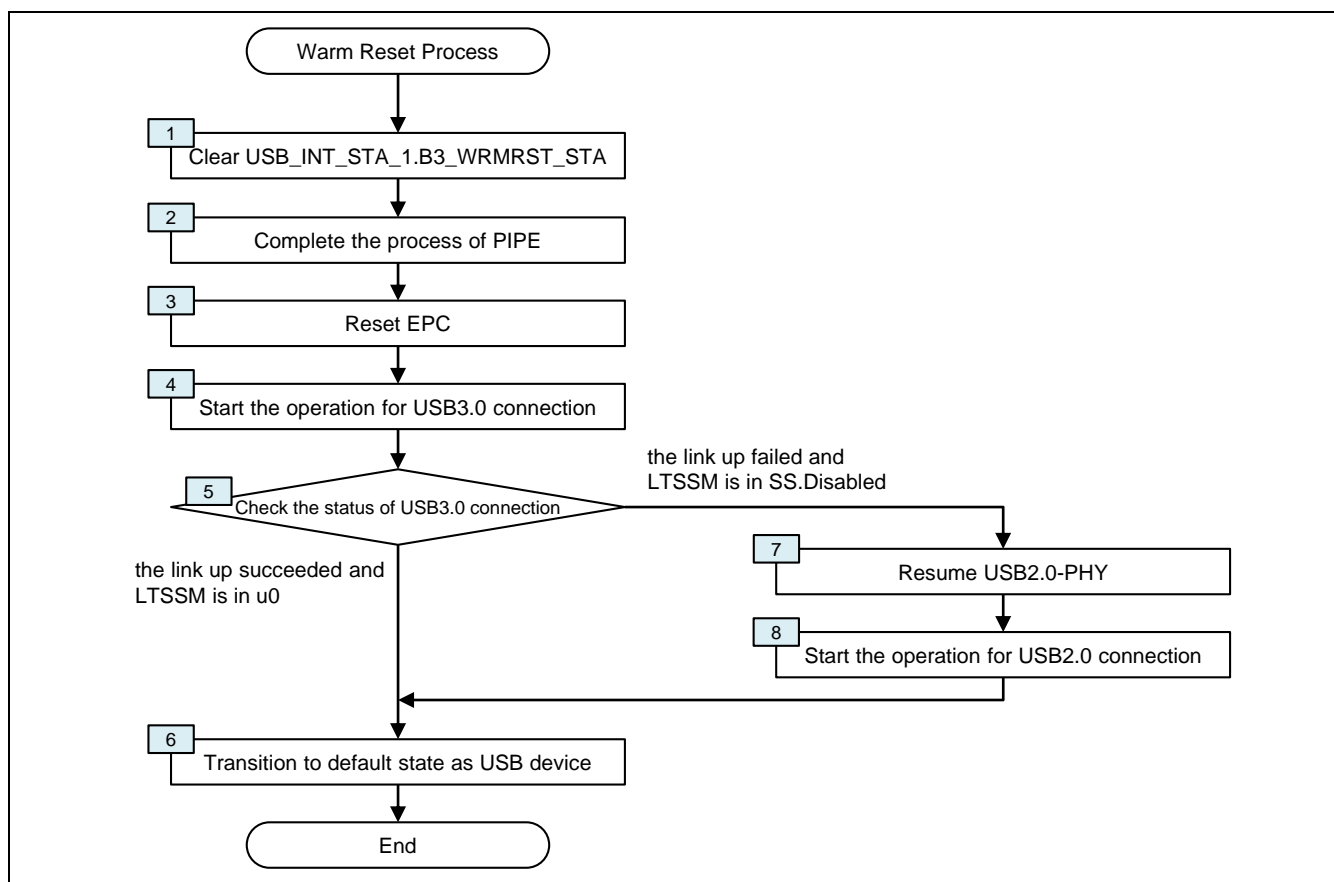


Figure 63.40 warm reset process flow

(1) Clear USB_INT_STA_1.B3_WRM_RST_STA

When an interrupt is asserted and USB_INT_STA_1.B3_WRM_RST_STA is set to 1, it means that PERIPHERAL PORTION receives warm reset. Write 1 to USB_INT_STA_1.B3_WRM_RST_STA in order to clear it.

- Interrupt due to USB_INT_STA_1.B3_WRM_RST_STA is generated
- Check that USB_INT_STA_1.B3_WRM_RST_STA == 1
- USB_INT_STA_1.B3_WRM_RST_STA = 1 (Clearing the event that warm reset is asserted)

(2) Complete the process of PIPE

Basically, as warm reset is requested when the device is just connected or when host and the device can't communicate, it is unlikely that the device is executing data transfer when warm reset comes.

But if it is doing, stop the process of data transfer if it is being executed.

In the configuration with AXI-IF, AXI_COM.MST_WAIT0 can be used to suspend the transfer.

- AXI_CON.MST_WAIT0 = 1 (DMA transfer at USB3.0 Controller is requested to be suspended)

See AXI_STA.DMA_TRANS0 to know whether the transfer is really suspended or not. If the transfer is suspended, then clear DMA_Ch0_CONx.PRD_ENx to 0.

- DMA_Ch0_CONx.PRD_ENx = 0 (DMA transfer in USB3.0 Controller is disabled)

In the configuration without AXI-IF, stop reading from or writing to PERIPHERAL PORTION through Data

Interface. In that case, since the transfer through Data Interface is executed in a unit of a packet, so reading or writing of the current packet should be completed. Then stop reading or writing for the next packet.

(3) Reset EPC portion

Set the registers as follows in order to initialize EPC. In addition to the operations, clear all interrupt statuses.

- USB_COM_CON.CONF = 0
- USB_COM_CON.EP0_EN = 0
- USB_COM_CON.PIPE_CLR = 1 (PIPE is initialized)
- * Make sure that USB_COM_CON.PIPE_CLR returns to 0 after setting USB_COM_CON.PIPE_CLR = 1.
- Write 1 to all interrupt status bits in order to clear them

(4) Start the operation for USB3.0 connection

Start the connecting operation as USB3.0 device.

- SSIFLINKSET3.SSIF_NUM_RDET[3:0] = H'8 (The number of trial of Rx.Detect = 8)
- USB_COM_CON.EP0_EN = 1
- USB_COM_CON.SPD_MODE = 0
- USB30_CON.B3_CONNECT = 1 (Starting the connection process for USB3.0)
- USB_INT_ENA_1.SPEED_ENA = 1 (Interrupt of SPEED state is enabled)

If USB3.0 connection is established (the device works as USB3.0 device) before warm reset comes, the fields already has the values above. But SSIFLINKSET3.SSIF_NUM_RDET[3:0] might be changed. Set the field again to make sure.

(5) Check the status of USB3.0 connection

Check the status of connection process as USB3.0 device on the bits below.

If USB_INT_STA_1.B3_LUPSUCS_STA is asserted, USB3.0 connection is completed successfully. Clear the bit and proceed to (6).

If USB_INT_STA_1.B3_DISABLE_STA is asserted, the link up process as USB3.0 device has failed. Clear the bit and proceed to (7).

- Interrupt due to USB_INT_STA_1.B3_LUPSUCS_STA or USB_INT_STA_1.B3_DISABLE_STA is asserted
- Check whether USB_INT_STA_1.B3_LUPSUCS_STA == 1 or not (The link up succeeded or not)
- Check whether USB_INT_STA_1.B3_DISABLE_STA == 1 or not (The link up failed or not)
- Write 1 to the asserted bit to clear (Clearing the event that the link up succeeded or failed)

(6) Transition to default state as USB device

After the connecting operation has been successfully done, enable the detection of control transfer at PIPE0 (endpoint 0). At the same time, the interrupts required for the operation mode should be enabled. Examples for those cases are shown below. If there are other interrupts required for the usage of core, enable them as well.

- USB_INT_ENA_2.PIPE0_INT_ENA = 1 (Interrupt at PIPE0 is enabled)
- All bits in P0_INT_ENA register should be enabled (All interrupt factors for PIPE0 are enabled)

If the interrupt due to the change of USB SPPED state is enabled at (3), the interrupt is asserted when USB3.0 or USB2.0 connection is established. See USB_STA.SPEED[1:0] to know which speed the device is acting as. Then set the bits according to SPEED state as follows.

[when acting as USB3.0 device]

- USB_INT_ENA_1.B3_HOTRST_ENA = 1 (Interrupt due to hot reset is enabled)
- USB_INT_ENA_1.B3_WRMST_ENA = 1 (Interrupt due to warm reset is enabled)

[when acting as USB2.0 device]

- USB_INT_ENA_1.B2_SPND_ENA = 1 (Interrupt due to USB2.0 suspend is enabled)
- USB_INT_ENA_1.B2_L1SPND_ENA = 1 (Interrupt due to L1 suspend is enabled)

- USB_INT_ENA_1.B2_USBRST_ENA = 1 (Interrupt due to USB2.0 bus reset is enabled)

(7) Resume USB2.0-PHY

If USB20 PHY is in the power down state and PLL is suspended, resume PLL so that it provides USB2.0 clock.

(8) Start the operation for USB2.0 connection

Start the connecting operation as USB2.0 device when USB3.0 connection is failed at (6).

- USB_COM_CON.EP0_EN = 1

- USB_COM_CON.SPD_MODE = 1

- USB20_CON.B2_PUE = B'1 (Pull up on D+ is enabled)

- USB20_CON.B2_CONNECT = 1 (Starting the connection process for USB2.0)

If device is acting as USB2.0 device and USB3.0- PHY is expected to be in low power state, set it in low power state here. But note that USB2.0 bus reset will come when the operation for USB2.0 connection is started and it is required to try USB3.0 connection again collaterally. It means that USB3.0-PHY can't be placed in low power state before USB2.0 connection is established. Make sure that the device is acting as USB2.0 device and the sequence of USB2.0 bus reset has already been done before setting USB3.0-PHY in low power state.

63.8.3.5 Hot Reset Process

The operation required when hot reset is detected is described here.

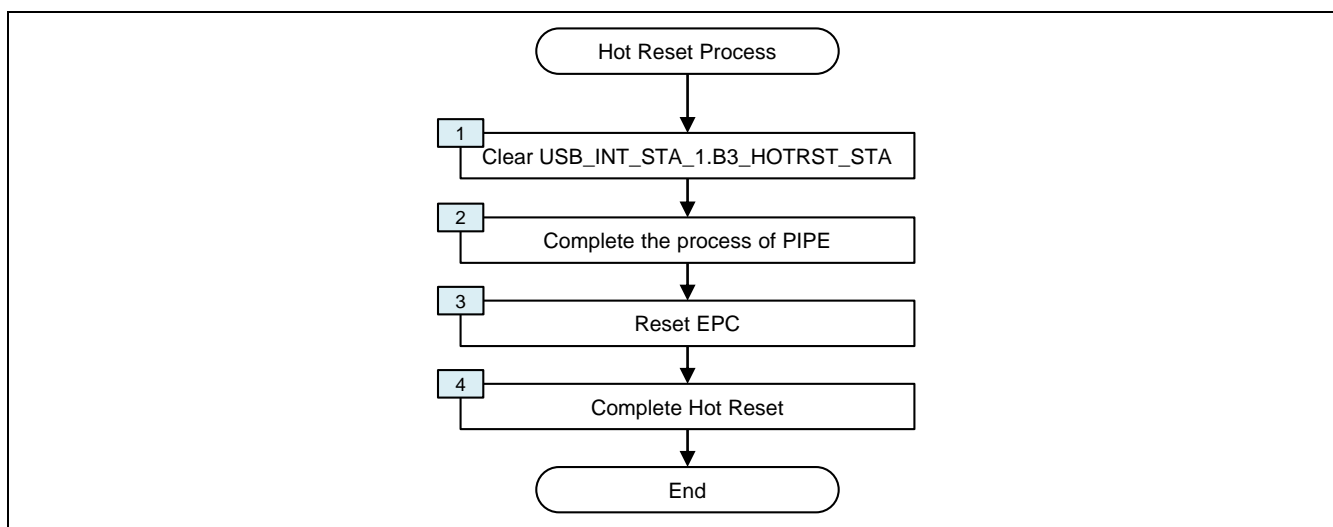


Figure 63.41 hot reset process flow

(1) Clear USB_INT_STA_1.B3_HOTRST_STA

When an interrupt is asserted and USB_INT_STA_1.B3_HOTRST_STA is set to 1, it means that PERIPHERAL PORTION receives hot reset. Write 1 to USB_INT_STA_1.B3_HOTRST_STA in order to clear it.

- Interrupt due to USB_INT_STA_1.B3_HOTRST_STA is generated
- Check that USB_INT_STA_1.B3_HOTRST_STA == 1
- USB_INT_STA_1.B3_HOTRST_STA = 1 (Clearing the event that hot reset is asserted)

(2) Complete the process of PIPE

Since hot reset comes during the training sequence, it is unlikely that the device is executing data transfer when hot reset comes.

But if it is doing, stop the process of data transfer if it is being executed.

In the configuration with AXI-IF, AXI_COM.MST_WAIT0 can be used to suspend the transfer.

- AXI_CON.MST_WAIT0 = 1 (DMA transfer at Peripheral portion is requested to be suspended)

See AXI_STA.DMA_TRANS0 to know whether the transfer is really suspended or not. If the transfer is suspended, then clear DMA_Ch0_CONx.PRD_ENx to 0.

- DMA_Ch0_CONx.PRD_ENx = 0 (DMA transfer in Peripheral portion is disabled)

In the configuration without AXI-IF, stop reading from or writing to PERIPHERAL PORTION through Data Interface. In that case, since the transfer through Data Interface is executed in a unit of a packet, so reading or writing of the current packet should be completed. Then stop reading or writing for the next packet.

(3) Reset EPC portion

Set the registers as follows in order to initialize EPC. In addition to the operations, clear all interrupt statuses.

- USB_COM_CON.CONF = 0
- USB_COM_CON.EP0_EN = 0
- USB_COM_CON.PIPE_CLR = 1 (PIPE is initialized)
- * Make sure that USB_COM_CON.PIPE_CLR returns to 0 after setting USB_COM_CON.PIPE_CLR = 1.
- Write 1 to all interrupt status bits in order to clear them

(4) Complete Hot Reset

Set the bit below to complete hot reset.

- USB_COM_CON.EP0_EN = 1
- USB30_CON.B3_HOTRST_CMP = 1

By setting USB30_CON.B3_HOTRST_CMP= 1, LTSSM state of PERIPHERAL PORTION transits from HotReset.Active to HotReset.Exit.

Setting B3_HOTRST_CMP= 1 allows the device to transit to HotReset.Exit from HotReset.Active. This process should be completed within 12 ms from detecting the start of HotReset at (1)

[Note]

The function of hot reset of device is tested in TD.7.28 of link layer test for compliance. The notations for the test are described here.

The downstream port can initiate a hot reset. When it transmits TS2 ordered sets with Reset bit asserted, the device shall respond by sending TS2 ordered sets with Reset bit asserted. When the device completes the operation for reset, it sends TS2 ordered set with Reset bit deasserted. The downstream port shall respond by sending TS2 ordered sets with Reset bit deasserted. Once both ports receive the TS2 ordered sets with Reset bit deasserted, it means the completion of hot reset and they shall exit from the hot reset state.

Note that PERIPHERAL PORTION automatically starts to transmit TS2 ordered sets with Reset bit asserted when it receives TS2 ordered sets with Reset bit asserted. (that is, hot reset requested from downstream port.)

But it is required to set USB30_CON.B3_HOTRST_CMP in order to return TS2 ordered sets with Reset bit deasserted, if PERIPHERAL PORTION enters the state it returns TS2 ordered set with Reset bit asserted once. If there is any operation required for reset, it should be done before USB30_CON.B3_HOTRST_CMP is set. It is notified on USB_INT_STA_1.B3_HOTRST_STA that PERIPHERAL PORTION receives TS2 ordered sets with Reset bit asserted and starts to respond to them.

If the device doesn't return TS2 ordered sets with Reset bit asserted and hot reset isn't completed with tHotResetActiveTimeout (=12 ms), the device fails in the test of TD.7.28.

If the device including PERIPHERAL PORTION never deasserts Reset bit in TS2 ordered set, the software may not recognize that USB_INT_STA_1.B3_HOTRST_STA is asserted. In the test of TD.7.28, TS2 ordered sets with Reset bit asserted may come immediately after the device is connected to the downstream port. That case doesn't occur in normal operation. If the software masks interrupts except that due to the change of VBUS (USB_INT_STA_1.VBUS_CNG_STA) immediately after the device is connected, the software may not be able to know USB_INT_STA_1.B3_HOTRST_STA asserted. It is recommended to enable interrupts due to USB_INT_STA_1.VBUS_CNG_STA and USB_INT_STA_1.B3_HOTRST_STA just after the device is connected.

63.8.3.6 USB2.0 Bus Reset Process

The operation required when USB2.0 bus reset is detected is described here.

There are two cases assumed as that USB2.0 bus reset comes.

- Device is just connected but fails in USB3.0 connection. USB2.0 connection is in progress.
- Device is working in HS or FS but host requests USB2.0 bus reset for some reason.
This case also means device has failed in USB3.0 connection once.

That is, USB2.0 bus reset doesn't come when device works in SS, but it comes when device works in HS or FS.

USB2.0 bus reset means not only the reset as USB2.0 device but also the retry of USB3.0 connection for USB3.0 device. See section 63.7.3 for the description of USB3.0 specification.

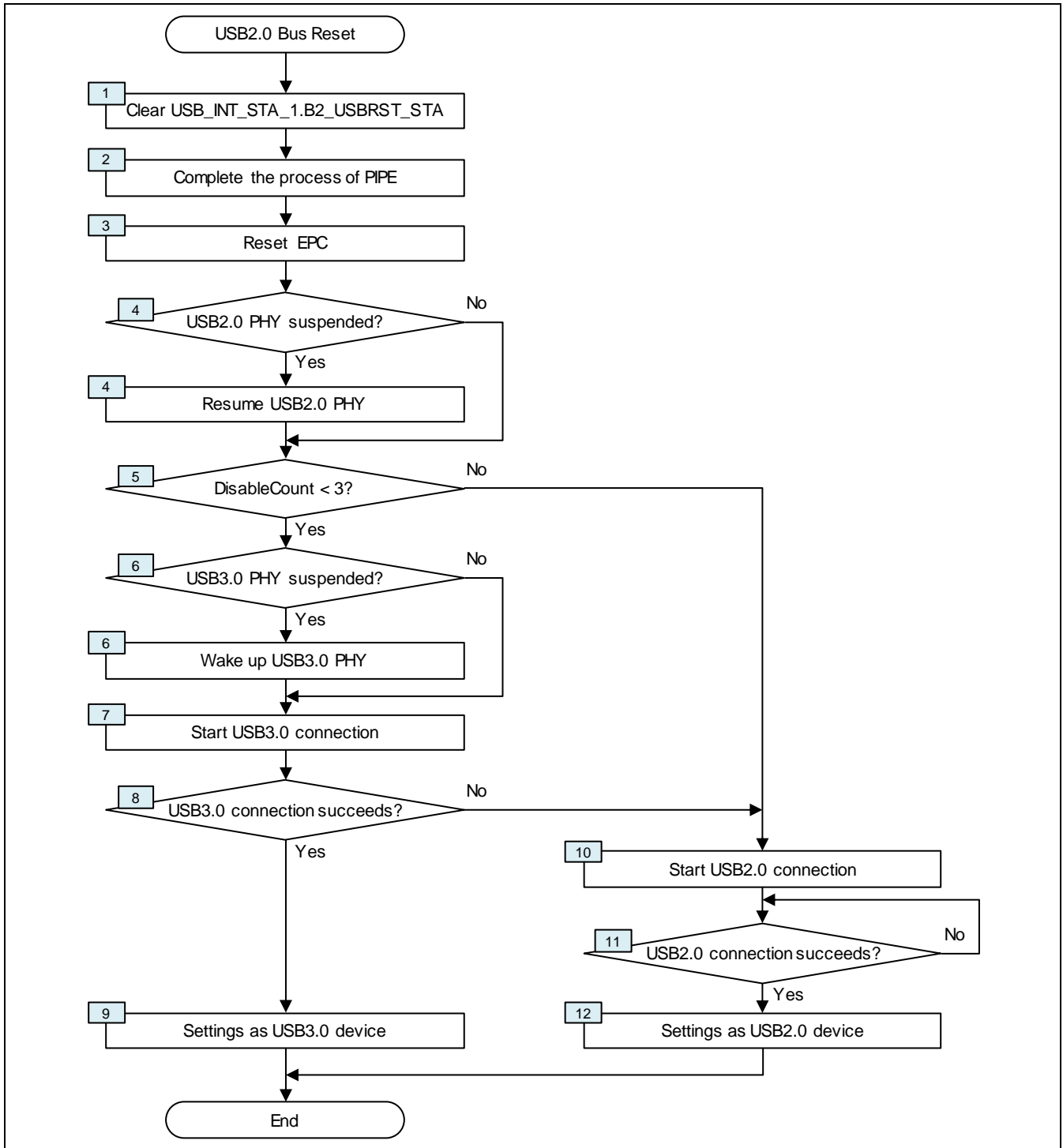


Figure 63.42 USB2.0 Bus Reset (switching from USB2.0 to USB3.0) Flow

The software is required to prepare the variable “DisabledCount” with this flow. The variable means the number of failure in link up. The initial value of DisabledCount is 0 and it is initialized to 0 when one of the following conditions is satisfied.

- VBUS is disabled (No power is provided from VBUS)
- Port configuration exchange completes successfully.
- Power is removed.

Furthermore, Being requested USB2.0 bus reset means that the device has already pulled up D+ (as HS or FS device) and has been detected by host. It is assumed that the software has already set USB20_CON.B2_CONNECT = 1 when USB2.0 bus reset comes.

(1) Clear USB_INT_STA_1.B2_USBRST_STA

When an interrupt is asserted and USB_INT_STA_1.B2_USBRST_STA is set to 1, it means that PERIPHERAL PORTION receives USB2.0 bus reset. Write 1 to USB_INT_STA_1.B2_USBRST_STA in order to clear it.

(2) Complete the process of PIPE

Basically, as USB2.0 bus reset is requested when the device is just connected or when host and the device can't communicate, it is unlikely that the device is executing data transfer when bus reset comes.

But if it is doing, stop the process of data transfer if it is being executed.

In the configuration with AXI-IF, AXI_COM.MST_WAIT0 can be used to suspend the transfer.

- AXI_CON.MST_WAIT0 = 1 (DMA transfer at Peripheral portion is requested to be suspended)

See AXI_STA.DMA_TRANS0 to know whether the transfer is really suspended or not. If the transfer is suspended, then clear DMA_Ch0_CONx.PRD_ENx to 0.

- DMA_Ch0_CONx.PRD_ENx = 0 (DMA transfer in Peripheral portion is disabled)

In the configuration without AXI-IF, stop reading from or writing to PERIPHERAL PORTION through Data Interface. In that case, since the transfer through Data Interface is executed in a unit of a packet, so reading or writing of the current packet should be completed. Then stop reading or writing for the next packet.

(3) Reset EPC portion

Set the registers as follows in order to initialize EPC. In addition to the operations, clear all interrupt statuses.

- USB_COM_CON.CONF = 0
- USB_COM_CON.EP0_EN = 0
- USB_COM_CON.PIPE_CLR = 1 (PIPE is initialized)
- * Make sure that USB_COM_CON.PIPE_CLR returns to 0 after setting USB_COM_CON.PIPE_CLR = 1.
- USB20_CON.B2_TSTMOD[2:0] = H'0 (if test mode is set)
- USB20_CON.B2_TSTMOD_EN = 0 (if test mode is set)
- Write 1 to all interrupt status bits in order to clear them

(4) Resume USB2.0-PHY

It is assumed that the device works in HS or FS, but the device might be in suspended state and PLL in USB2.0-PHY might be stopped when USB2.0 bus reset comes.

In that case resume USB2.0-PHY so that it provides USB2.0 clock. USB3.0 connection might be requested depending on the value of DisabledCount, but the pull up of D+ for USB2.0 connection is still valid and USB2.0 connection is tried immediately when USB3.0 connection fails. That is, USB2.0-PHY clock is required regardless of the value of DisabledCount.

(5) Is DisabledCount<3?

If DisabledCount < 3, it means the device state is in USDPORT.Powered on due to USB2.0 bus reset and the device requests USB3.0 connection. Proceed to (6).

If DisabledCount==3, it means the device state is in USDPORT.Disabled_Error and the device doesn't request USB3.0 connection. Proceed to (10).

(6) Resume USB3.0-PHY

If USB3.0-PHY is in the power down state and PLL is suspended, resume USB3.0-PHY so that it provides USB3.0 clock.

(7) Start USB3.0 connection

As DisabledCount<3, USB3.0 connection is tried again.

If USB2.0 bus reset is the first one just after the device is connected but fails in USB3.0 connection, Most of the fields below has already been set at (6) in section 63.8.3.2. In that case, the settings for USB2.0 connection at (10) in section 63.8.3.2 should be still active as they are because USB2.0 connection might be required when USB3.0 connection fails.

- USB30_CON.B3_CONNECT = 1 (Starting the connection process for USB3.0)
- USB_COM_CON.EP0_EN = 1 (PIPE0 is enabled)
- USB_COM_CON.SPD_MODE = 0 (USB3.0 mode is requested)
- SSIFLINKSET3.SSIF_NUM_RDET[3:0] = H'1 (The number of trial for Rx.Detect = 1)
- USB_COM_CON.RX_DETECTION = 1 (RxDetect is started)
- USB_INT_ENA_1.B3_LUPSUCS_ENA = 1 (Interrupt due to the success of link up is enabled)
- USB_INT_ENA_1.B3_DISABLE_ENA = 1 (Interrupt due to the failure of link up is enabled)
- USB_INT_ENA_1.SPEED_ENA = 1 (Interrupt of SPEED state is enabled)

The difference with (6) in section 63.8.3.2 is that the SSIFLINKSET3.SSIF_NUM_RDET[3:0] = H'1 although it is set to H'8 at (6) in section 63.8.3.2. It means that the device can do only one trial of Rx detection.

(8) Does USB3.0 connection succeed?

As the result of the trial for USB3.0 connection, one of USB_INT_STA_1.B3_LUPSUCS_STA or USB_INT_STA_1.B3_DISABLE_STA is asserted. USB_INT_STA_1.B3_LUPSUCS_STA means that the link up process of LTSSM has been successfully done and LTSSM is in U0. USB_INT_STA_1.B3_DISABLE_STA means that the link up process of LTSSM has failed and LTSSM is in SS.Disabled. When they are asserted, an interrupt is generated if enabled.

- Check USB_INT_STA_1.B3_LUPSUCS_STA== 1 or USB_INT_STA_1.B3_DISABLE_STA== 1

After the confirmation above, write 1 to USB_INT_STA_1.B3_LUPSUCS_STA or USB_INT_STA_1.B3_DISABLE_STA to clear them.

- Write 1 to USB_INT_STA_1.B3_LUPSUCS_STA if asserted (Clearing the event that the link up has succeeded)
- Write 1 to USB_INT_STA_1.B3_DISABLE_STA if asserted (Clearing the event that USB3.0 connection has failed)

If USB_INT_STA_1.B3_LUPSUCS_STA is asserted, it means USB3.0 connection is successful, proceed to (9).

If USB_INT_STA_1.B3_DISABLE_STA is asserted, it means USB3.0 connection has failed, proceed to (10).

(9) Settings as USB3.0 device

After the connecting operation as SS device has been successfully completed, disable USB2.0 connection. It is defined in USB3.0 specification that USB2.0 connection should be disabled within tUSB2SwitchDisconnect (= 1ms) after Far End Receiver Termination is detected and LTSSM enters Polling.

- USB20_CON.B2_CONNECT = 0 (USB2.0 connection is disabled)
- USB20_CON.B2_PUE = 0 (Pull up of D+ is disabled)

Then enable the detection of control transfer at PIPE0 (endpoint 0). At the same time, the interrupts required for the operation mode should be enabled. An example for that cases are shown below. If there is another interrupt required for the usage of core, enable it as well.

- USB_INT_ENA_2.PIPE0_INT_ENA = 1 (Interrupt at PIPE0 is enabled)
- All bits in P0_INT_ENA register should be enabled (All interrupt factors for PIPE0 are enabled)

As the device is working as SS device, interrupts due to hot reset or warm reset shall be enabled.

- USB_INT_ENA_1.B3_HOTRST_ENA = 1 (Interrupt due to hot reset is enabled)
- USB_INT_ENA_1.B3_WRM_RST_ENA = 1 (Interrupt due to warm reset is enabled)

When the port configuration is completed, DisabledCount is reset to 0.

If USB2.0 clock is not needed, USB2.0-PHY can be suspended. See section 63.8.8.7 for details.

(10) Start USB2.0 connection

Since USB3.0 connection has failed, disable USB3.0 connection.

- USB30_CON.B3_CONNECT = 0
- DisabledCount++

Then, start USB2.0 connection.

The most of the fields below has already set, but USB_COM_CON.SPD_MODE is set for USB3.0 connection at (7), so it should be changed for USB2.0 connection.

- USB_COM_CON.EP0_EN = 1 (PIPE0 is enabled: this field is already enabled at (6))
- USB_COM_CON.SPD_MODE = 1 (USB2.0 mode is requested)
- USB20_CON.B2_PUE = 1 (Pull up on D+ is enabled)
- USB20_CON.B2_CONNECT = 1 (Starting the connection process for USB3.0)
- USB_INT_ENA_1.SPEED_ENA = 1 (Interrupt of SPEED state is enabled)

(11) Does USB2.0 connection succeed?

After USB2.0 bus reset and chirp, the device works as HS or FS device. For USB2.0 connection, there is no link training as that in USB3.0 connection, so it is assumed to be always completed successfully.

(12) Setting as USB2.0 device

After the connecting operation as HS or FS has been successfully done, enable the detection of control transfer at PIPE0 (endpoint 0). At the same time, the interrupts required for the operation mode should be enabled. Examples for those cases are shown below. If there are other interrupts required for the usage of core, enable them as well.

- USB_INT_ENA_2.PIPE0_INT_ENA = 1 (Interrupt at PIPE0 is enabled)
- All bits in P0_INT_ENA register should be enabled (All interrupt factors for PIPE0 are enabled)

If the interrupt due to the change of USB SPPED state is enabled at (10), the interrupt is asserted when USB2.0 connection is established. See USB_STA.SPEED[1:0] to know which speed the device is acting as. Then set the bits according to SPEED state as follows.

[when acting as USB2.0 device]

- USB_INT_ENA_1.B2_SPND_ENA = 1 (Interrupt due to USB2.0 suspend is enabled)
- USB_INT_ENA_1.B2_L1SPND_ENA = 1 (Interrupt due to L1 suspend is enabled)
- USB_INT_ENA_1.B2_USBRST_ENA = 1 (Interrupt due to USB2.0 bus reset is enabled)

63.8.4 Control Transfer

The operation for control transfer is described here.

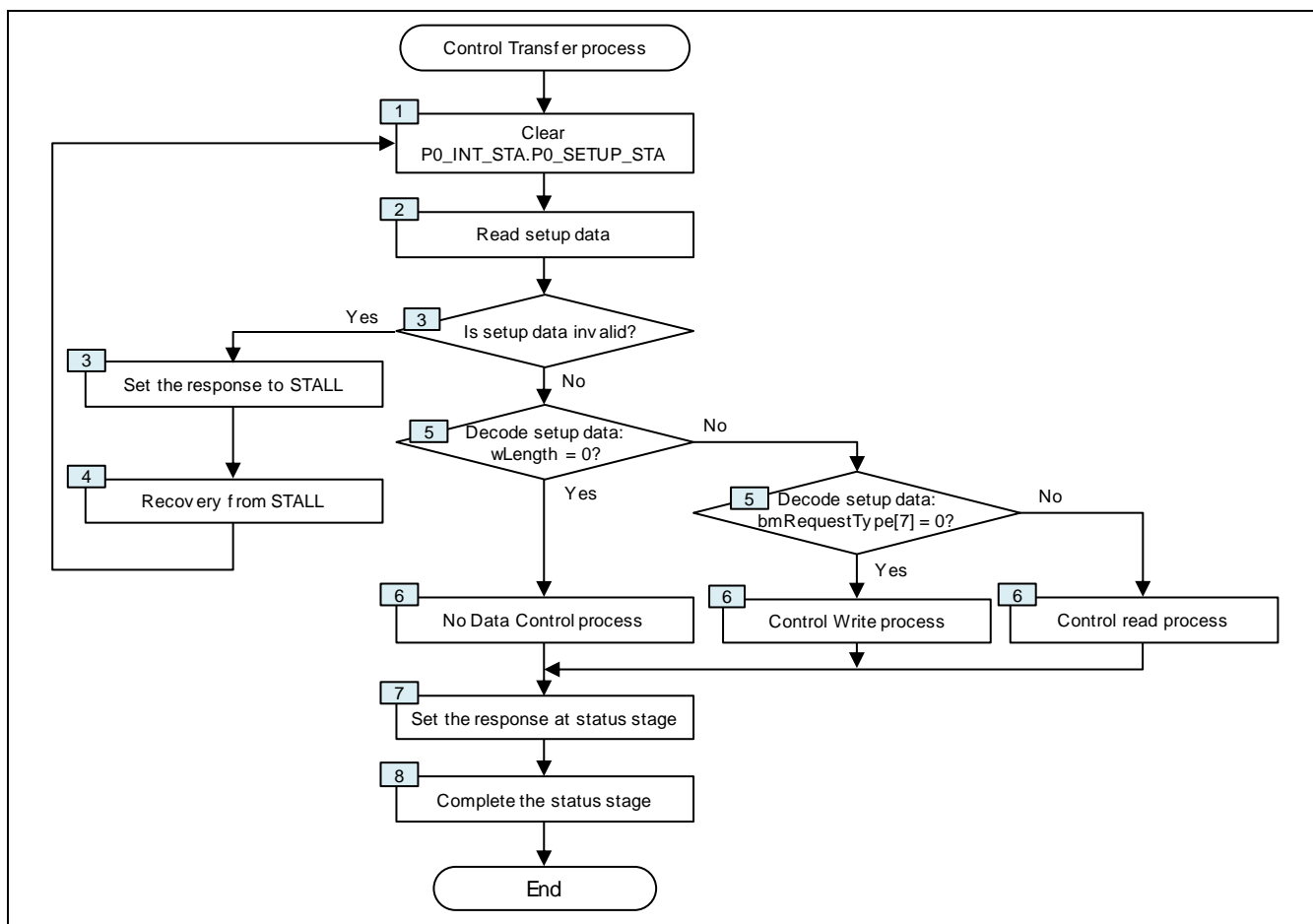


Figure 63.43 control transfer process flow

(1) Clear P0_INT_STA.P0_SETUP_STA
 When an interrupt is asserted and P0_INT_STA.P0_SETUP_STA is set to 1, it means that PERIPHERAL PORTION has received setup data. Write 1 to P0_INT_STA.P0_SETUP_STA in order to clear it.

- Interrupt due to P0_INT_STA.P0_SETUP_STA is generated
- Check that P0_INT_STA.P0_SETUP_STA == 1
- P0_INT_STA.P0_SETUP_STA = 1 (Clearing the event setup data has been received)

In case the status bits related to control transfer are set, they should be cleared as follows.

- P0_INT_STA.P0_STSST_STA = 1
- P0_INT_STA.P0_STSED_STA = 1
- P0_INT_STA.P0_RCVNL_STA = 1
- P0_INT_STA.P0_ERDY_STA = 1
- P0_INT_STA.P0_FLOW_STA = 1
- P0_INT_STA.P0_STALL_STA = 1
- P0_INT_STA.P0_NRDY_STA = 1

The data buffer for PIPE0 should also be cleared just in case data of previous transfer is left in the buffer.

- P0_CON.P0_BCLR = 1 (Clearing data buffer for PIPE0)

(2) Read setup data

Read setup data from STUP_DAT_0 and STUP_DAT_1 registers.

(3) Is setup data invalid?

In case that setup data is invalid, respond with STALL. Set P0_ST_RES[1:0], P0_OT_RES[1:0] and P0_IN_RES[1:0] in P0_CON register to return STALL response in that case.

- P0_CON.P0_RES_WEN = 1
- P0_CON.P0_ST_RES[1:0] = B'10 (forced STALL response)
- P0_CON.P0_OT_RES[1:0] = B'10 (forced STALL response)
- P0_CON.P0_IN_RES[1:0] = B'10 (forced STALL response)

With the settings above, PERIPHERAL PORTION returns STALL for all requests (IN, OUT or STATUS) to PIPE0. Whenever PERIPHERAL PORTION responds with STALL, P0_INT_STA.P0_STALL_STA is asserted. Since USB device is not required to change the response from PIPE0 until new setup data comes to PIPE0 as noted in (5) or any reset comes in that case, an interrupt due to P0_INT_STA.P0_STALL_STA can be ignored (that is, it can be masked).

(4) Recovery from STALL

If host wants to recovery the status, it will send ClearFeature (ENDPOINT_HALT) or other request. The device should initialize the response for it.

When host sends new setup data including it requests ClearFeature (ENDPOINT_HALT), P0_INT_STA.P0_SETUP_STA will be asserted again.

When the assertion of P0_INT_STA.P0_SETUP_STA is detected (it is notified on an interrupt if enabled), PERIPHERAL PORTION automatically sets P0_CON.P0_ST_RES[1:0], P0_CON.P0_OT_RES[1:0] and P0_CON.P0_IN_RES[1:0] to B'00 (NRDY/NAK response). Note that it might take time to decide and prepare a response here. Hence it is assumed that the response is changed to normal one after the decision and the preparation. If it doesn't take time, normal responses can be set here.

- New setup data is received. (It is notified on the assertion of P0_INT_STA.P0_SETUP_STA.)
- (- PERIPHERAL PORTION automatically sets P0_CON.P0_*_RES[1:0] to B'00, but the software can change them to normal response.)

Then, go back to (1).

(5) Decode setup data

Decode the content of setup data which is read at (2).

If wLength = 0, no data control process is being requested.

If wLength is not 0 and bmRequestType[7] is 0, control write data process is being requested. Otherwise control read process is being requested.

(6) No Data Control, Control Write or Control Read process

Refer each subsection as follows.

- a) No Data Control process: Refer to section 63.8.4.1
- b) Control Write process: Refer to section 63.8.4.2
- c) Control Read process: Refer to section 63.8.4.3

(7) Set the response at status stage

From (7) to (8) the settings for status stage are described. For the operations required in each process before status stage, refer section 63.8.9. Basically the process requested by setup data should be finished before the device completes the status stage. The status stage might be started before the completion of the process, but the device can put off the completion of the status stage by returning NRDY or NAK.

Set P0_CON.P0_ST_RES[1:0] as follows.

- P0_CON.P0_RES_WEN = 1

- P0_CON.P0_ST_RES[1:0] = B'01 (Normal response)

For P0_CON.P0_OT_RES[1:0] and P0_CON.P0_IN_RES[1:0], keep the value set at (6) in case of Control Write process or Control Read process. In case of No Data Control process, they should be set to forced STALL response at the same time as the setting of P0_CON.P0_ST_RES[1:0].

- P0_CON.P0_OT_RES[1:0] = B'10 (STALL response)

- P0_CON.P0_IN_RES[1:0] = B'10 (STALL response)

(8) Complete Status Stage

Wait for the interrupt due to P0_INT_STA.P0_STSED_STA, which shows PERIPHERAL PORTION completes the status stage of the control transfer. Confirm P0_INT_STA.P0_STSED_STA is asserted, and then write 1 to it to clear.

- Interrupt due to P0_INT_STA.P0_STSED_STA is generated

- Check that P0_INT_STA.P0_STSED_STA == 1

- P0_INT_STA.P0_STSED_STA = 1 (Clearing the event that the status stage of control transfer has been completed.)

Note: In the specification of USB, it is defined that the sequence of control transfer should be initialized when new setup data is received during control transfer. Device should stop the operation for previous setup data and switch to the operation of new setup data. That case seldom occurs, but the consideration of it is required.

63.8.4.1 No Data Control Process

The operation for no data control transfer is described here.

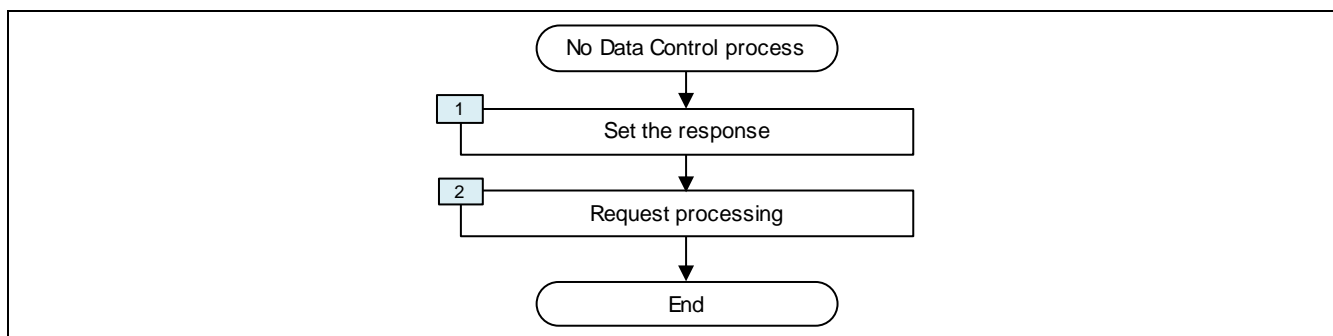


Figure 63.44 no data control transfer process flow

(1) Set the response

After decoding setup data, set the response for the request.

- P0_MOD.P0_DIR = 0 (Direction = OUT)

Select NRDY/NAK response for the status stage until the operation at (2) is completed. If the operation is not needed, PERIPHERAL PORTION can respond with ACK immediately. The final response at the status stage is selected at (7) in the flow of section 63.8.4.

As no data is sent or received in this flow, select STALL response for IN and OUT transfer request. All of the settings below are for P0_CON register and it allows only 32-bit access, so note that they should be done in one word access.

- P0_CON.P0_RES_WEN = 1

- P0_CON.P0_ST_RES[1:0] = B'00 (NRDY/NAK response. ACK response is also selectable.)

- P0_CON.P0_OT_RES[1:0] = B'10 (STALL response)

- P0_CON.P0_IN_RES[1:0] =B'10 (STALL response)

(2) Request processing

If any operation for the request is required, do it here.

See section 63.8.9 for the operations of major device requests.

63.8.4.2 Control Write Process

The operation for control writer transfer is described here.

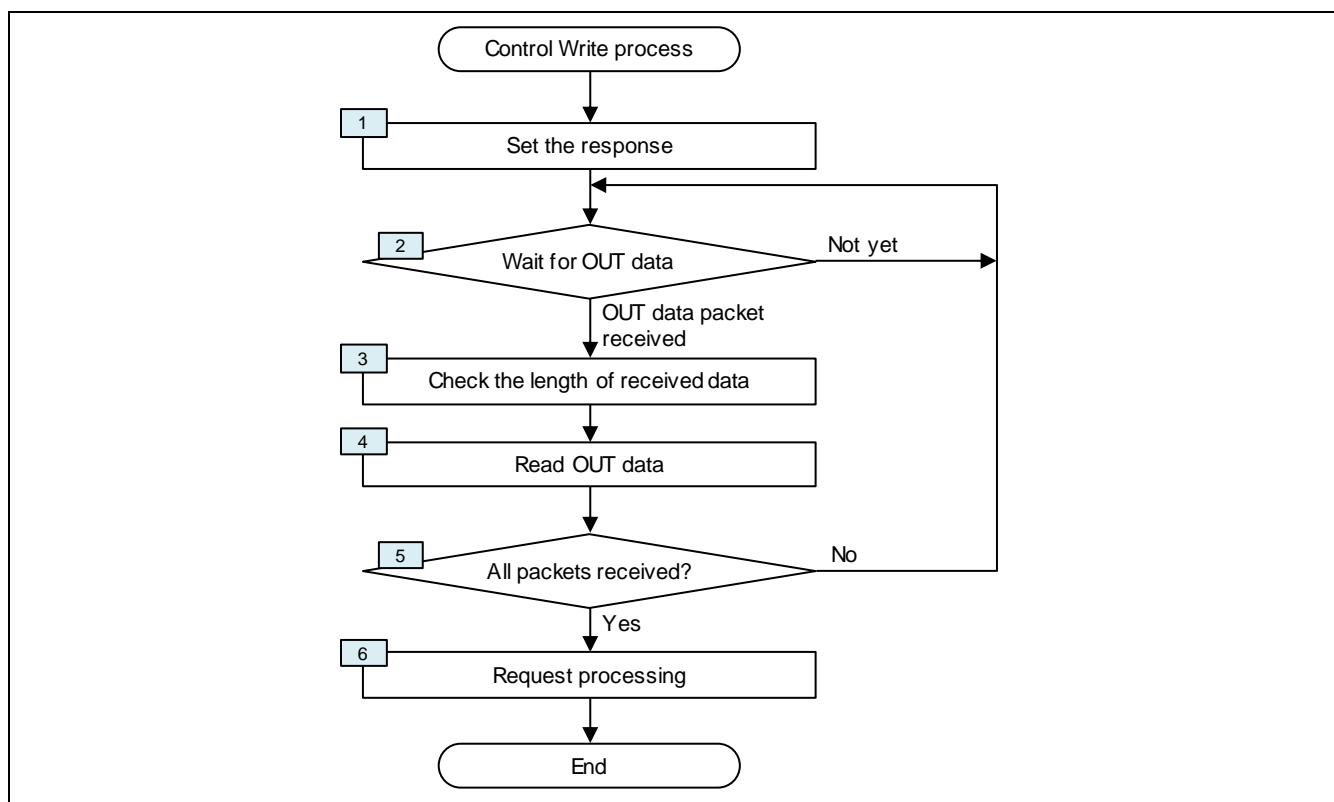


Figure 63.45 control write process flow

(1) Set the response

After decoding setup data, set the response for the request.

As control write transfer is requested here, select OUT as the direction of transfer.

Set NRDY/NAK response for the status stage since the data stage has not been finished yet. Set normal response for OUT and STALL response for IN.

- P0_MOD.P0_DIR = 0 (Direction = OUT)
- P0_CON.P0_RES_WEN = 1
- P0_CON.P0_ST_RES[1:0] = B'00 (NRDY/NAK response)
- P0_CON.P0_OT_RES[1:0] = B'01 (normal response)
- P0_CON.P0_IN_RES[1:0] = B'10 (STALL response)

(2) Wait for OUT data

For control write transfer, the software reads each packet in data stage of the transfer through P0_READ register.

When OUT data packet to PIPE0 is received, an interrupt due to P0_INT_STA.P0_BFRDY_STA is asserted. In that case, write 1 to P0_INT_STA.P0_BFRDY_STA to clear.

- Interrupt due to P0_INT_STA.P0_BFRDY_STA is generated
- Check that P0_INT_STA.P0_BFRDY_STA == 1
- P0_INT_STA.P0_BFRDY_STA = 1 (Clearing the event that OUT data has been received at PIPE0)

When receiving OUT data packet, P0_STA.P0_BUFSTS is also asserted. The meaning of the bit is OUT data packet is ready to be read from P0_READ register. Polling P0_STA.P0_BUFSTS is available to know that OUT data packet has been received for the software as well.

(3) Check the length of received data

Read the following field to know the length of received data.

- Read P0_LNG.P0_LENGTH[9:0] (This field shows the number of bytes of received data)

(4) Read OUT data

Read OUT data through P0_READ register. Note that OUT data in control transfer can't be read from Data Interface of Peripheral portion or can't be transferred by AXI DMA transfer.

P0_READ register should be read in a unit of 32-bit. P0_READ register is updated and the next 32-bit word of received data is shown in P0_READ register every time it is read. The length of received data is known at (3) and the number of read from P0_READ register can be calculated from the length. The number of read = the length divided by 4, round-up is required if there is any residue. If there is residue, it also means the last 32-bit word read has one or more unnecessary bytes. In that case, discard unnecessary bytes.

(5) Are all packets received?

If wLength shown in setup data is more than the max packet size in control transfer, it means two or more OUT data packets would come. In that case, repeat the steps from (2) to (4) for the number of packets.

After all of OUT packets have been received and read, set the registers as follows. Note P0_CON register should be written in a unit of 32-bit and other fields should hold its value. That is, read-modify-write will be required.

- P0_CON.P0_RES_WEN = 1

- P0_CON.P0_OT_RES[1:0] = B'10 (STALL response)

(6) Request processing

If any operation for the request is required, do it here.

See section 63.8.9 for the operations of major device requests.

63.8.4.3 Control Read Process

The operation for control read transfer is described here.

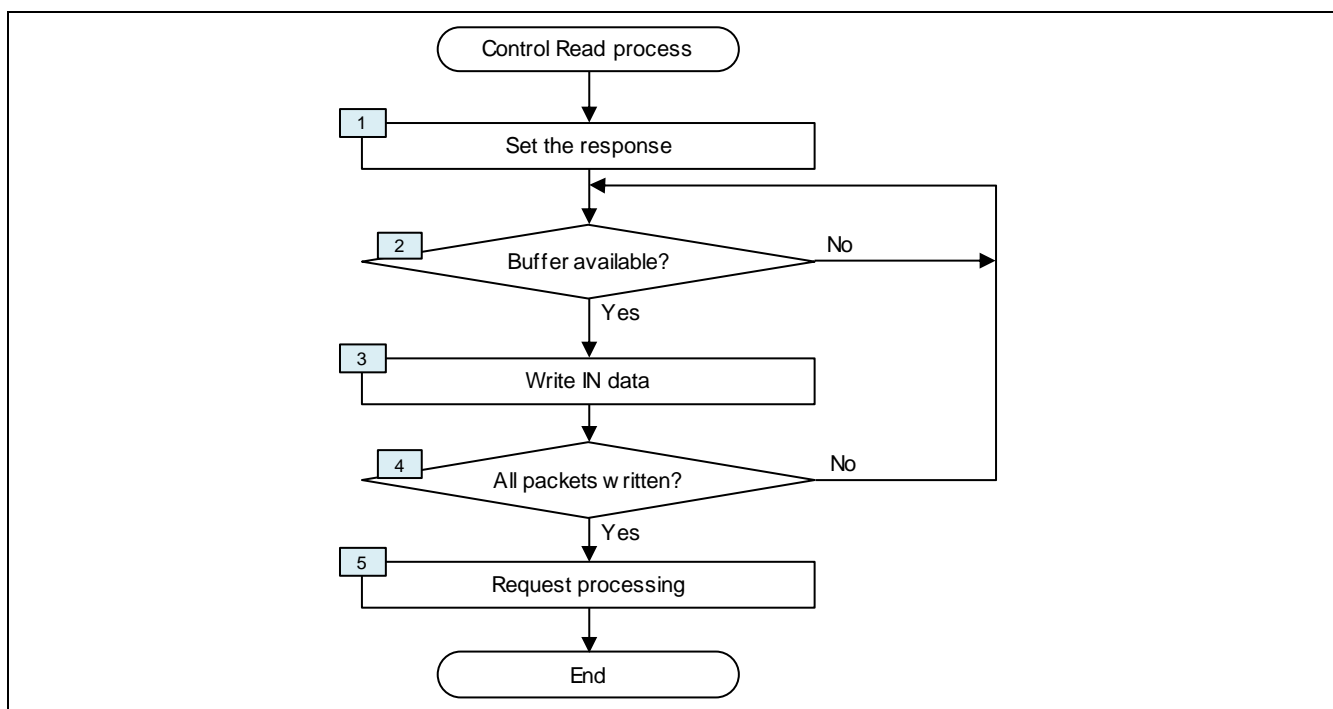


Figure 63.46 control read process flow

(1) Set the response

After decoding setup data, set the response for the request.

As control read transfer is requested here, select IN as the direction of transfer.

Set NRDY/NAK response for the status stage since the data stage has not been finished yet. Set normal response for IN and STALL response for OUT.

- P0_MOD.P0_DIR = 1 (Direction = IN)
- P0_CON.P0_RES_WEN = 1
- P0_CON.P0_ST_RES[1:0] = B'00 (NRDY/NAK response)
- P0_CON.P0_OT_RES[1:0] = B'10 (STALL response)
- P0_CON.P0_IN_RES[1:0] = B'01 (Normal response)

(2) Check if buffer is available

For control read transfer, the software writes each packet in data stage of the transfer through P0_WRITE register.

But if there is no space available in the buffer for PIPE0, it can't write. P0_STA.P0_BUFSTS shows if there is available space in the buffer. See the bit before writing IN data packet to P0_WRITE register.

- Check whether P0_STA.P0_BUFSTS == 1 or not

If P0_STA.P0_BUFSTS is 0 and there is no available space in the buffer currently, Wait until

P0_STA.P0_BUFSTS becomes 1 or an interrupt due to P0_INT_STA.P0_BFRDY_STA is asserted. When IN is selected as the direction of PIPE0 (that is, P0_MOD.P0_DIR=1), P0_INT_STA.P0_BFRDY_STA is asserted when the buffer for PIPE0 becomes available for new write data.

Therefore, the software can know the timing the buffer for PIPE0 prepares new available space by polling P0_STA.P0_BUFSTS or an interrupt due to P0_INT_STA.P0_BFRDY_STA.

But, Note that P0_INT_STA.P0_BFRDY_STA is not asserted just after reset (hardware reset,

USB_COM_CON.PIPE_CLR, P0_CON.P0_BCLR and P0_CON.P0_CLR) and when the software writes the first packet. In that case, since the buffer has available space just after reset, so the software can write the packet without being notified on an interrupt due to P0_INT_STA.P0_BFRDY_STA. The software still can see P0_STA.P0_BUFSTS in that case.

(3) Write IN data

Write IN data through P0_WRITE register. Note that IN data in control transfer can't be written through Data Interface of Peripheral portion or can't be transferred by AXI DMA transfer.

P0_WRITE register should be written in a unit of 32-bit. The number of write can be calculated as the number of write = the length divided by 4, round-up is required if there is any residue. The written words are concatenated as one packet.

When the length of packet is not multiples of 32-bit, the last word has unnecessary bytes. The last written word should be also written as a 32-bit word. The number of valid bytes in the last word is set in P0_CON.P0_BYTE_EN. Unnecessary bytes in the last word are discarded.

To send the packet, set the registers as follows after the last word has been written.

- P0_CON.P0_SEND = 1

- P0_CON.P0_BYTE_EN[1:0] = B'xx (the number of valid bytes in last written word)

(4) Are all packets written?

If wlength shown in setup data is more than the max packet size in control transfer, it means two or more IN data packets are required. In that case, repeat the steps from (2) to (3) for the number of packets.

After all of IN packets have been sent, set the registers as follows. Note P0_CON register should be written in a unit of 32-bit and other fields should hold its value. That is, read-modify-write will be required.

- P0_CON.P0_RES_WEN = 1

- P0_CON.P0_IN_RES[1:0] = B'01 (STALL response)

(5) Request processing

If any operation for the request is required, do it here.

See section 63.8.9 for the operations of major device requests.

63.8.5 Bulk Transfer

The operation for bulk transfer is described here.

PIPEs other than PIPE0 (PIPE0 is used only for control transfer) can be used for bulk transfer. The settings for bulk transfer are shown below.

- PIPE_COM.PIPE_NUM[4:0] = the index of PIPE used for the current bulk transfer
- PIPE_MOD.Pn_EPNUM[3:0] = the endpoint number of the PIPE
- PIPE_MOD.Pn_TYPE[1:0] = B'10 (bulk transfer)
- PIPE_MOD.DIR = the direction of transfer
- PIPE_MOD.Pn_STREAM = whether the PIPE is used for Stream bulk transfer or not
- PIPE_MOD.Pn_BOT = whether the PIPE uses BOT or not
- PIPE_MOD.Pn_ERDY_CON[1:0] = threshold to send ERDY
- PIPE_RAMMAP.Pn_BASEAD[13:0] = the base address of buffer assigned for the PIPE
- PIPE_RAMMAP.Pn_RAMIF[1:0] = the index of RAM interface of the PIPE
- PIPE_RAMMAP.Pn_MPKT[10:0] = 1024bytes for SS, 512 bytes for HS or 64 bytes for FS
- PIPE_RAMMAP.Pn_RAMAREA = the size of RAM area assigned for the PIPE
- Pn_CON.Pn_DATAIF_EN = whether Data Interface of Peripheral portion is used or not.
- Pn_CON.Pn_EN = 1 (the PIPE is enabled)

These settings are done just after Set Configuration is completed and the device enters configured state. But if the device has only single configuration for each PIPE, these settings can be done after the device is turned on.

63.8.5.1 Bulk-IN Process

The normal bulk IN process is described here. See section 63.8.5.3 for Stream bulk process.

In this subsection the process to write bulk IN packet through Pn_WRITE register is described. DMA transfer as the master function of Peripheral portion is available in the configuration with AXI-IF. See section 63.8.7 for the process using DMA transfer.

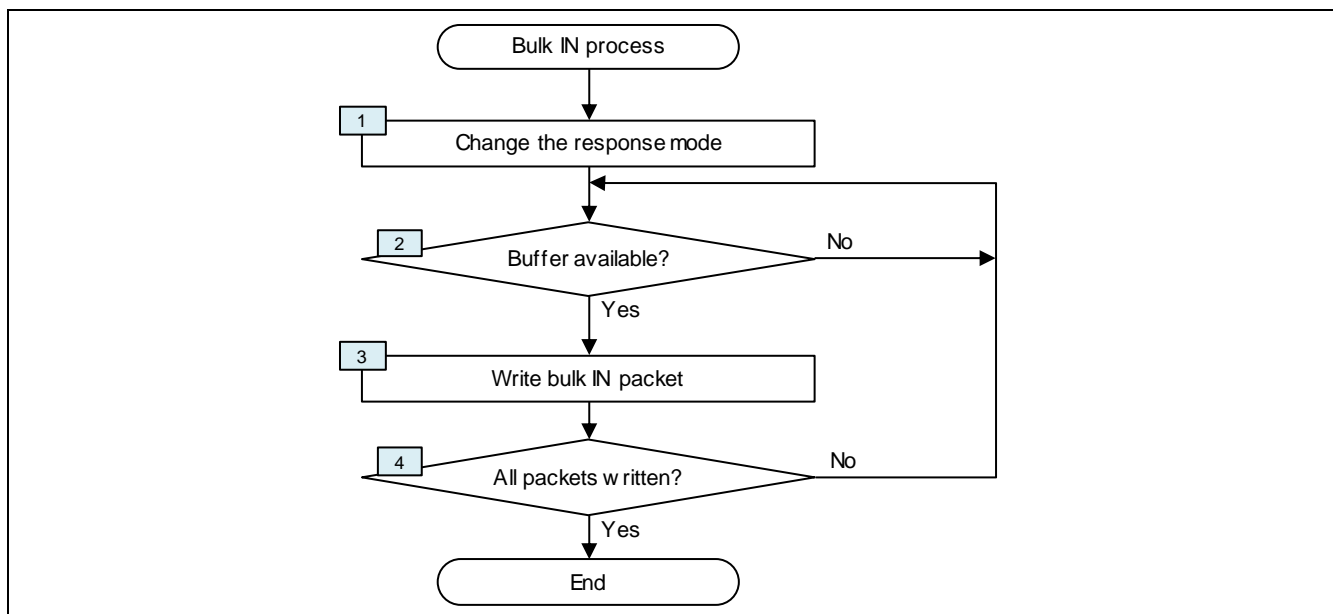


Figure 63.47 bulk IN process flow

(1) Change the response mode of the bulk IN PIPE

Change the response mode of the bulk IN PIPE to normal response so that the PIPE can transmit data.

- Pn_CON.Pn_RES_WEN = 1
- Pn_CON.Pn_RES[1:0] = B'01 (Normal response)

(2) Check if buffer is available

Check if buffer for the bulk IN PIPE is available before writing bulk IN packet. Pn_STA.Pn_BUFSTS shows if the buffer is available or not.

- Check whether Pn_STA.Pn_BUFSTS == 1 or not

If Pn_STA.Pn_BUFSTS is 0 and there is no available space in the buffer currently, Wait until Pn_STA.Pn_BUFSTS becomes 1 or an interrupt due to Pn_INT_STA.Pn_BFRDY_STA is asserted. When IN is selected as the direction of the bulk PIPE (that is, Pn_MOD.Pn_DIR = 1), Pn_INT_STA.Pn_BFRDY_STA is asserted when the buffer of the PIPE becomes available for new write data.

Therefore, the software can know the timing the buffer of the PIPE prepares new available space by polling Pn_STA.Pn_BUFSTS or an interrupt due to Pn_INT_STA.Pn_BFRDY_STA.

But, Note that Pn_INT_STA.Pn_BFRDY_STA is not asserted just after reset (hardware reset, USB_COM_CON.PIPE_CLR, Pn_CON.Pn_BCLR and Pn_CON.Pn_CLR) and when the software writes the first packet. In that case, since the buffer has available space just after reset, so the software can write the packet without being notified on an interrupt due to Pn_INT_STA.Pn_BFRDY_STA. The software still can see Pn_STA.Pn_BUFSTS in that case.

(3) Write bulk IN packet

Write bulk IN packet to be transmitted to the buffer. Bulk IN data is written in a unit of packet and the buffer can't accept data across the max packet size defined in PIPE_RAMMAP.Pn_MPKT[10:0].

[writing through Pn_WRITE register]

a-1) Write data through Pn_WRITE register. Since only 32-bit access is allowed to the register, data should be written in a unit of 32-bit. The number of write can be calculated as the number of write = the length divided by 4, round-up is required if there is any residue. The written words are concatenated as one packet.

When the length of packet is not multiples of 32-bit, the last written word may have unnecessary bytes.

The last word should be also written as a 32-bit word. The number of valid bytes in the last word is set in Pn_CON.Pn_BYTE_EN[1:0]. Unnecessary bytes in the last word are discarded.

a-2) Set fields as follows to send the packet. Since only 32-bit access is allowed to Pn_CON register, all fields below should be set in one word access.

- Pn_CON.Pn_SEND = 1

- Pn_CON.Pn_BYTE_EN[1:0] = B'xx (the number of valid bytes in last written word)

- In case of the last packet of data group, set Pn_CON.Pn_LAST = 1

(4) Are all packets written?

If it is required to send two or more packets, repeat the steps from (2) to (3).

Note: When the information of a certain PIPE is read from or write to PIPEn registers, PIPE_COM.PIPE_NUM[4:0] should have the index of the PIPE.

63.8.5.2 Bulk-OUT Process

The normal Bulk-OUT process is described here. See section 63.8.5.3 for Stream bulk process.

In this subsection the process to read bulk OUT packet through Pn_READ register is described. DMA transfer as the master function of Peripheral portion is available in the configuration with AXI-IF. See section 63.8.7 for the process using DMA transfer.

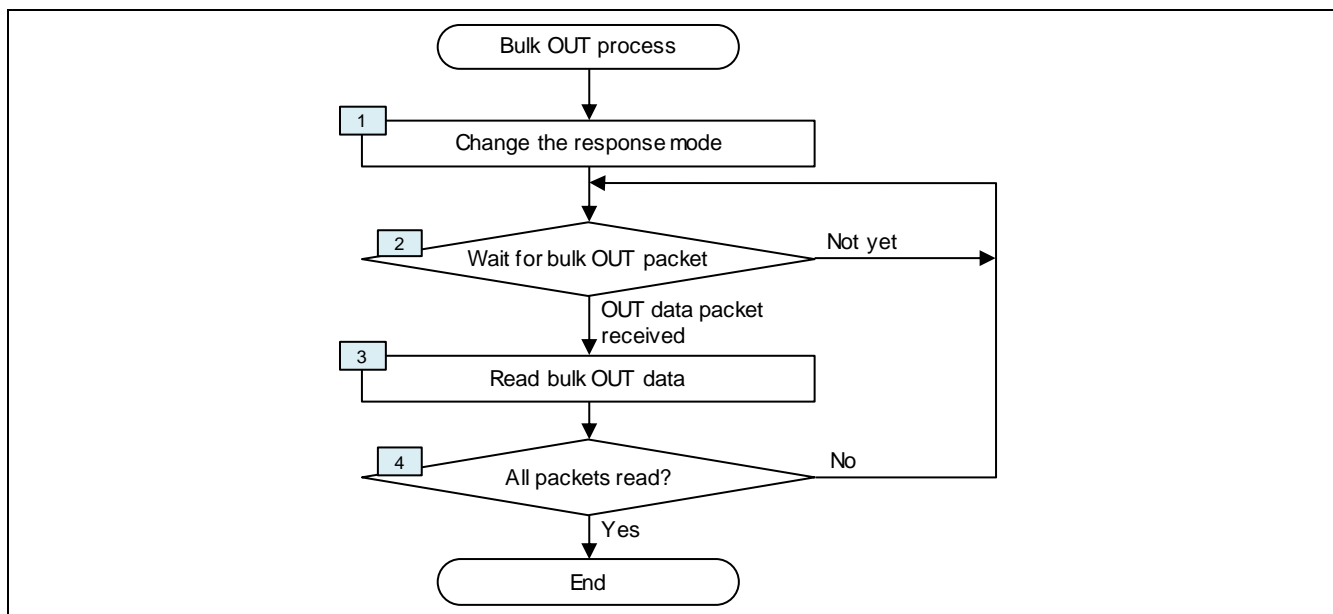


Figure 63.48 bulk OUT process flow

- (1) Change the response mode of the bulk OUT PIPE
Change the response mode of the bulk OUT PIPE to normal response so that the PIPE can receive data.

- Pn_CON.Pn_RES_WEN = 1
- Pn_CON.Pn_RES[1:0] = B'01 (Normal response)

- (2) Wait for bulk OUT data
When bulk OUT data to the PIPE is received, an interrupt due to Pn_INT_STA.Pn_BFRDY_STA is asserted. In that case, write 1 to Pn_INT_STA.Pn_BFRDY_STA to clear.

- Interrupt due to Pn_INT_STA.Pn_BFRDY_STA asserted
- Check whether Pn_INT_STA.Pn_BFRDY_STA == 1 or not.
- Pn_INT_STA.Pn_BFRDY_STA = 1 (clearing the event that OUT data is received at the PIPE)

When receiving OUT data packet, Pn_STA.Pn_BUFSTS is also asserted. The meaning of the bit is bulk OUT data packet is ready to be read from Pn_READ register. Polling Pn_STA.Pn_BUFSTS is available to know that OUT data packet has been received for the software as well.

(3) Read bulk OUT packet

Read bulk OUT packet from the buffer. Bulk OUT data is read in a unit of packet and the buffer can't be read across the max packet size defined in PIPE_RAMMAP.Pn_MPKT[10:0].

[reading through Pn_READ register]

a-1) Read Pn_LNG register in order to know the length of received data

a-2) Read received packet through Pn_READ register.

Pn_READ register should be read in a unit of 32-bit. Pn_READ register is updated and the next 32-bit word of received data is shown in Pn_READ register every time it is read. The length of received data is known at a-1) and the number of read from Pn_READ register can be calculated from the length. The number of read = the length divided by 4, round-up is required if there is any residue. If there is residue, it also means the last 32-bit word read has one or more unnecessary bytes. In that case, discard unnecessary bytes.

(4) Are all packets read?

If the bulk OUT PIPE has more packets to be read, repeat the steps from (2) to (3).

Pn_INT_STA.Pn_LSTTR_STA is asserted when the last packet (short packet or packet with PP = 0) is received.

When a short packet or a packet with PP = 0 (PP is used only in USB3.0) is received, PERIPHERAL PORTION recognizes it is the last packet of the current transfer. It returns ACK with NumP = 0 to host after receiving the packet. ACK with NumP = 0 means that the device can't receive any more packet currently. The device should send ERDY when it wants to resume the transfer. When the last packet (short packet or packet with PP = 0) has been read to AXI-Bus and the buffer of the PIPE has available space, PERIPHERAL PORTION automatically send ERDY to request the resume of transfer.

Note that PERIPHERAL PORTION may return ACK with NumP = 0 even when the received packet is not the last packet if it has no available buffer at that time. But when PERIPHERAL PORTION gets available space in the buffer of the PIPE, it automatically sends ERDY to resume the transfer for the PIPE.

Note: When the information of a certain PIPE is read from or write to PIPEn registers, PIPE_COM.PIPE_NUM[4:0] should have the index of the PIPE.

63.8.5.3 Stream Bulk Process

The process of stream bulk, especially UASP (USB Attached SCSI protocol) application as an example is described here.

(1) PIPE structure of UASP

UASP assumes 4 pipe models, not including control pipe (PIPE0). The pipes are Command (OUT), Status (IN), Data-in (IN) and Data-out (OUT).

UAS driver sends a command through host stack to Command pipe, device receives it and returns the status or the result for the command from Status pipe. Device receives OUT packet at Data-out pipe or returns IN packet from Data-In pipe, if required.

Stream protocol is used for the parallel operation for commands. Device can receive multiple commands to Command pipe at once. It returns statuses from Status pipe or IN packets from Data-in pipe, or receives OUT packets to Data-out pipe, in “out-of-order” way. Each command has a Stream ID as a tag, and Stream protocol is used for the tagging. Note Status pipe, Data-in pipe and Data-out pipe are Stream bulk pipes but Command pipe is not. Stream ID is provided in CIU (Command IU) which is received at Command pipe.

(2) Basic Stream Bulk IN process

The basic process of Stream bulk IN is described here.

Stream bulk IN pipe should have PIPE_MOD.Pn_STREAM = 1 as noted in the first part of section 63.8.5.

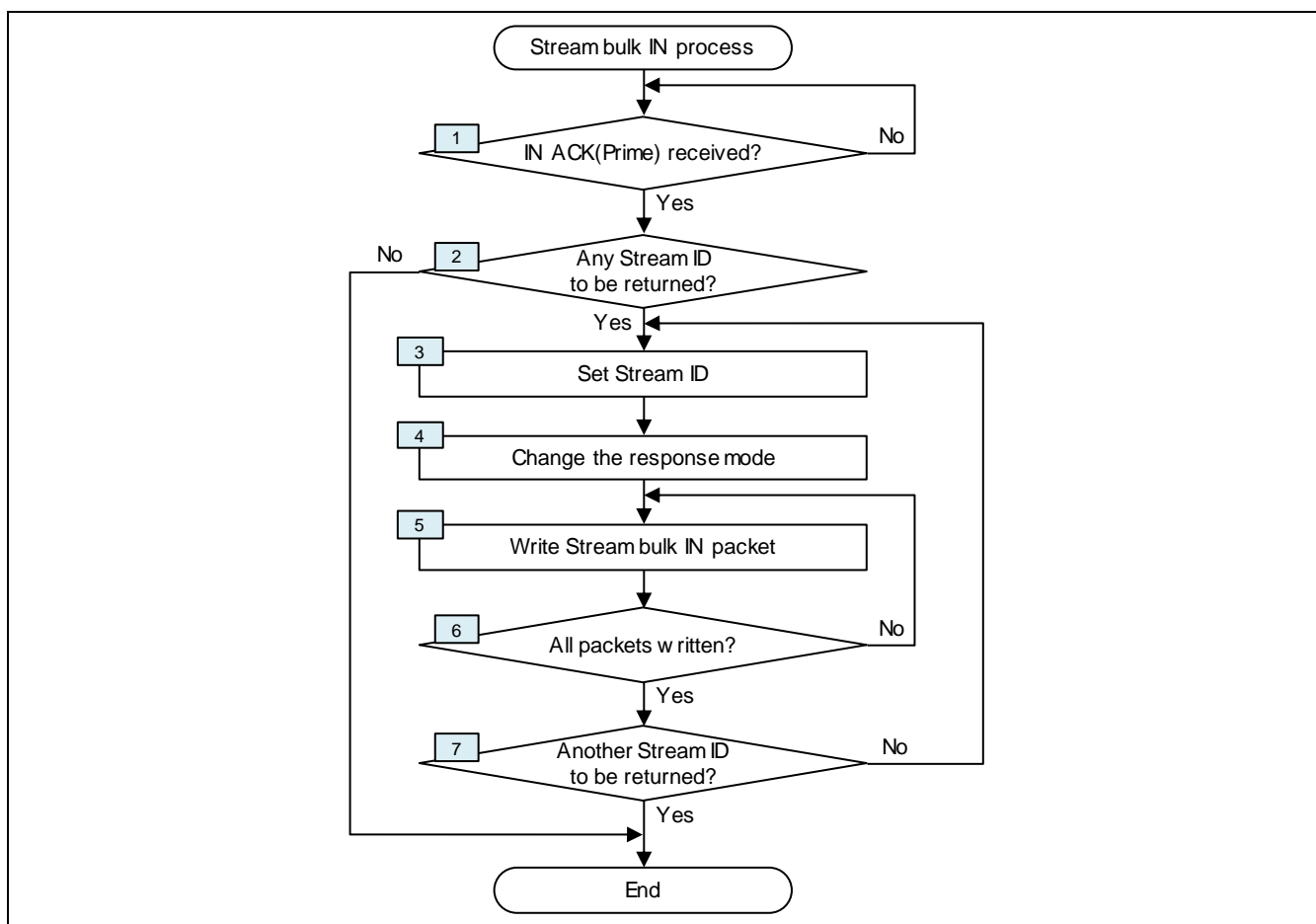


Figure 63.49 Basic Stream bulk IN process flow

(1) Is IN ACK(Prime) received?

IN ACK(Prime) is used to inform device that one or more streams are added to the IN pipe and transfer for it will be started. In case Stream Bulk IN pipe receives IN ACK(Prime), it always responds with NRDY(Prime). Receiving IN ACK(Prime) means the pipe transits to Prime Pipe state of Stream Bulk state machine, and sending NRDY(Prime) means it goes back to Idle state.

PERIPHERAL PORTION automatically changes the setting of Pn_CON.Pn_RES[1:0] of the IN pipe to return NRDY (= B'00) and sends NRDY(Prime) to host.

The reception of IN ACK(Prime) is notified on the assertion of Pn_INT_STA.Pn_PRIME_STA. An interrupt due to it is asserted if enabled.

- Interrupt due to Pn_INT_STA.Pn_PRIME_STA is asserted, if enabled.

- Check that Pn_INT_STA.Pn_PRIME_STA == 1

After the confirmation above, clear Pn_INT_STA.Pn_PRIME_STA.

- Pn_INT_STA.Pn_PRIME_STA = 1 (Clearing the event that IN ACK(Prime) is received)

Then, the IN pipe is required to prepare to transfer Stream Bulk IN data.

(2) Is there any Stream ID to be returned?

The data returned from Stream bulk IN pipe should be tagged by Stream ID. The ID is notified in the command from host in advance. In UASP, CIU (Command IU) is sent to Command pipe and the command ID in CIU is used as Stream ID.

If device has received multiple commands, it should decide which command is processed first. In UASP, the operation for Stream is closely related to NCQ (native command queuing) of storage. Stream bulk IN pipe returns data in the order the storage returns.

If there is any Stream ID to be returned, proceed to (3).

(3) Set Stream ID

Set Stream ID of data which is to be returned next to Pn_STREAM.Pn_STREAM_C[15:0].

- Pn_STREAM.Pn_STREAM_C[15:0] = CStream (The current stream ID)

(4) Change the response mode of Stream bulk IN pipe

As the IN pipe returned NRDY at (1), so it is in the flow control state. The response mode of the IN pipe was changed to "forced NRDY or NAK response" (= B'00).

Change the response mode of the IN pipe to normal response mode.

- Pn_CON.Pn_RES_WEN = 1

- Pn_CON.Pn_RES[1:0] = B'01 (Normal response)

(5) Write Stream bulk IN packet

Write Stream bulk IN packet to be transmitted to the buffer. This step is same as that for normal bulk IN.

Stream bulk IN data is written in a unit of packet and the buffer can't accept data across the max packet size defined in PIPE_RAMMAP.Pn_MPKT[10:0].

It is also required to see whether there is available space in the buffer or not on Pn_INT_STA.Pn_BFRDY_STA or Pn_STA.Pn_BUFSTS. See section 63.8.5.1 for details.

[writing through Pn_WRITE register]

a-1) Write data through Pn_WRITE register. Since only 32-bit access is allowed to the register, data should be written in a unit of 32-bit. The number of write can be calculated as the number of write = the length divided by 4, round-up is required if there is any residue. The written words are concatenated as one packet.

When the length of packet is not multiples of 32-bit, the last written word has unnecessary bytes.

The last word should be also written as a 32-bit word. The number of valid bytes in the last word is set in Pn_CON.Pn_BYTE_EN[1:0]. Unnecessary bytes in the last word are discarded.

a-2) Set fields as follows to send the packet. Since only 32-bit access is allowed to Pn_CON register, all fields below should be set in one word access.

- Pn_CON.Pn_SEND = 1
- Pn_CON.Pn_BYTE_EN[1:0] = B'xx (The number of valid bytes in last written word)
- In case of the last packet of data group, set Pn_CON.Pn_LAST = 1

See section 63.8.7 for details.

When the first packet for the Stream ID (= CStream) is written to buffer, PERIPHERAL PORTION returns ERDY(CStream) to host. This means not only the exit from flow control state but the selection of Stream from device. If host agrees the selection, it sends IN ACK(CStream) to start transfer for the Stream ID.

(6) Are all packets written?

If it is required to send two or more packets, repeat the step of (5).

When the last packet for the Stream ID is written, make sure to indicate that on Pn_CON.Pn_LAST (in case data is written through pn_WRITE register), Lst bit in PRD table (with AXI-IF).

(7) Is there another Stream ID to be returned?

When the last packet written at (6) is sent to host, EOB (end of burst) bit in it is set to 1 so that it shows the packet is the last one in the burst. Host will send ACK(NumP = 0) as the response to the packet with EOB = 1. This means the IN pipe enters flow control state again.

If there is another Stream ID to be returned, repeat the steps from (2) to (6).

Note that PERIPHERAL PORTION might set EOB = 1 in the packet which is not the last one for the Stream ID (= CStream) in case that the preparation of data is delayed and there is no packet in buffer to be sent next to the current packet. In that case, actually EOB is set to 1 in the packet and the IN pipe is considered as being in the flow control state, but no additional operation is required by the software. When the next packet is written in buffer, the IN pipe sends ERDY(CStream) automatically. This means not only the exit from the flow control state but the selection of CStream again. The IN transfer for CStream will be started.

Note: When the information of a certain PIPE is read from or write to PIPEn registers, PIPE_COM.PIPE_NUM[4:0] should have the index of the PIPE.

Note: It is possible for host to send ACK (Stream x) and initiates the selection of the Stream. But, basically, to decrease the size of buffer and increase the performance of device, it is desirable that device decides what Stream is processed next seeing its status and initiates the selection of stream. Beside, device can't send data related to the Stream ID which is not prepared yet. Therefore, PERIPHERAL PORTION responds with NRDY if Stream x doesn't equal to the Stream ID in Pn_STREAM.Pn_STREAM_C[15:0]. PERIPHERAL PORTION asserts Pn_INT_STA.Pn_STRMX_STA in case Stream ID in the received packet is not same as that in Pn_STREAM.Pn_STREAM_C[15:0], Prime nor NoStream. In that case, the Stream ID requested in the received packet is shown in Pn_NRDYSTRM register.

It is possible to give priority to the Stream ID requested from host by checking Pn_NRDYSTRM register, but it is assumed here that device requests another Stream ID which it decided to process.

Especially, READ READY IU and WRITE READY IU are defined to show which stream (command) is processed next from device in UASP. ERDY(Stream) is used in Super Speed Stream Bulk of UASP just as the substitution of them. Therefore the device initiates the selection of stream in this flow.

Note: From the definition in USB3.0 specification, device which selects the Stream by returning ERDY (Stream) may receive ACK (No Stream) from host. ACK (No Stream) means the rejection from host for proposed Stream ID in ERDY. In such case, device should consider the Stream as "Not Ready", and may select another Stream. It depends on the application whether No Stream is really used or not.

(3) Basic Stream Bulk OUT process

The basic process of Stream bulk OUT is described here.

Stream bulk OUT pipe should have PIPE_MOD.Pn_STREAM = 1 as noted in the first part of section 63.8.5.

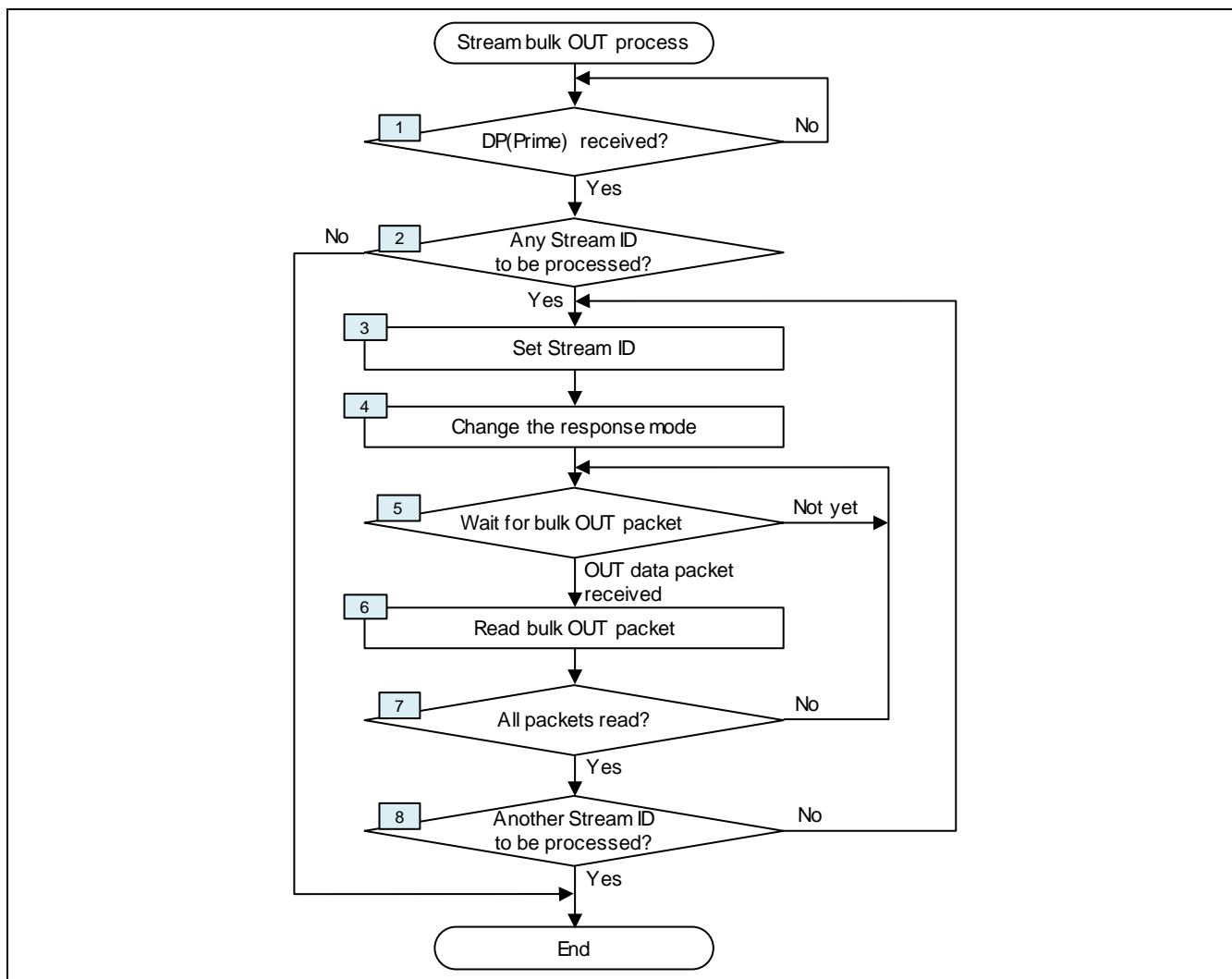


Figure 63.50 Basic Stream bulk OUT process flow

(1) Is DP(Prime) received?

DP(Prime) is used to inform device that one or more streams are added to the OUT pipe and transfer for it will be started. In case Stream Bulk OUT pipe receives DP(Prime), it always responds with NRDY(Prime). Receiving DP(Prime) means the pipe is requested to enter Prime Pipe state of Stream Bulk state machine, and sending NRDY(Prime) means it goes back to Idle state.

PERIPHERAL PORTION automatically changes the setting of Pn_CON.Pn_RES[1:0] of the OUT pipe to return NRDY (= B'00) and sends NRDY(Prime) to host.

The reception of DP(Prime) is notified on the assertion of Pn_INT_STA.Pn_PRIME_STA. An interrupt due to it is asserted if enabled.

- Interrupt due to Pn_INT_STA.Pn_PRIME_STA is asserted, if enabled.
- Check that Pn_INT_STA.Pn_PRIME_STA == 1

After the confirmation above, clear Pn_INT_STA.Pn_PRIME_STA.

- Pn_INT_STA.Pn_PRIME_STA = 1 (Clearing the event that DP (Prime) is received)

Then, the OUT pipe is required to prepare to transfer Stream Bulk OUT data.

(2) Is there any Stream ID to be processed

The data received at Stream bulk OUT pipe should be tagged by Stream ID. The ID is notified in the command from host in advance. In UASP, CIU (Command IU) is sent to Command pipe and the command ID in CIU is used as Stream ID.

If device has received multiple commands, it should decide which command is processed. In UASP, the operation for Stream is closely related to NCQ (native command queuing) of storage. Stream bulk OUT pipe processes the stream in the order the storage specifies

If there is any Stream ID to be processed, proceed to (3).

(3) Set Stream ID

Set Stream ID of the Stream which is processed next to Pn_STREAM.Pn_STREAM_C[15:0].

- Pn_STREAM.Pn_STREAM_C[15:0] = CStream (The current stream ID)

(4) Change the response mode of Stream bulk OUT pipe

As the OUT pipe returned NRDY at (1), so it is in the flow control state. The response mode of the OUT pipe was changed to “forced NRDY or NAK response” (=B'00).

Change the response mode of the OUT pipe to normal response mode.

- Pn_CON.Pn_RES_WEN = 1

- Pn_CON.Pn_RES[1:0] = B'01 (Normal response)

After the setting above, ERDY (CStream) is sent from the OUT pipe to host. This means not only the exit from flow control state but the selection of Stream from device. If host agrees the selection, it starts the transfer for the stream and sends DP (CStream).

(5) Wait for bulk OUT data

When bulk OUT data to the OUT PIPE is received, an interrupt due to Pn_INT_STA.Pn_BFRDY_STA is asserted. In that case, write 1 to Pn_INT_STA.Pn_BFRDY_STA to clear. This step is same as that for normal bulk OUT.

- Interrupt due to Pn_INT_STA.Pn_BFRDY_STA asserted

- Check Pn_INT_STA.Pn_BFRDY_STA == 1 or not.

- Pn_INT_STA.Pn_BFRDY_STA = 1 (Clearing the event that OUT data is received at the PIPE)

When receiving OUT data packet, Pn_STA.Pn_BUFSTS is also asserted. The meaning of the bit is bulk OUT data packet is ready to be read from Pn_READ register. The software can poll Pn_STA.Pn_BUFSTS to know that OUT data packet has been received for the software as well.

(6) Read bulk OUT packet

Read bulk OUT packet from the buffer. Bulk OUT data is read in a unit of packet and the buffer can't be read across the max packet size defined in PIPE_RAMMAP.Pn_MPKT[10:0]. This step is same as that for normal bulk OUT.

a-1) Read Pn_LNG.Pn_LENGTH[15:0] in order to know the length of received data

a-2) Read received packet through Pn_READ register.

Pn_READ register should be read in a unit of 32-bit. Pn_READ register is updated and the next 32-bit word of received data is shown in Pn_READ register every time it is read. The length of received data is known at a-1) and the number of read from Pn_READ register can be calculated from the length. The number of read = the length divided by 4, round-up is required if there is any residue. If there is residue, it also means the last 32-bit word read has one or more unnecessary bytes. In that case, discard unnecessary bytes.

See section 63.8.7 for details.

(7) Are all packets read?

If the bulk OUT PIPE has more packets to be read, repeat the steps from (5) to (6).

Pn_INT_STA.Pn_LSTTR_STA is asserted when the last packet (short packet or packet with PP = 0) is received.

(8) Is there another Stream ID to be processed?

When the last packet is received at (7), PERIPHERAL PORTION returns ACK with NumP = 0. This means the OUT pipe enters flow control state again.

If there is another Stream ID to be returned, repeat the steps from (2) to (7). Note that Pn_CON.Pn_RES[1:0] was changed to B'00 (NAK/NRDY response) when the PIPE received the last packet and returned ACK with NumP = 0. Pn_CON.Pn_RES[1:0] is changed to B'01 (normal response) at (4) again, and the PIPE transmits ERDY to resume the transfer.

It is also noted that PERIPHERAL PORTION might return ACK with NumP = 0 for the packet which is not the last one for the Stream ID (= CStream) in case that there is no available space in the buffer for the PIPE currently. In that case, actually NumP is set to 0 in the ACK packet and the OUT pipe is considered as being in the flow control state, but no additional operation is required for the software. When the received packet is transferred to system memory and PERIPHERAL PORTION gets available space in the buffer, the OUT pipe sends ERDY(CStream) automatically. This means not only the exit from the flow control state but the selection of CStream again. The OUT transfer for CStream will be started.

Note: When the information of a certain PIPE is read from or write to PIPE_n registers, PIPE_COM.PIPE_NUM[4:0] should have the index of the PIPE.

Note: It is possible for host to send ACK (Stream x) and initiates the selection of the Stream. But, basically, to decrease the size of buffer and increase the performance of device, it is desirable that device decides what Stream is processed first seeing its status and initiates the selection of stream. Beside, device can't send for Stream ID which is not available yet. Therefore, PERIPHERAL PORTION responds with NRDY if Stream x doesn't equal to the Stream ID in Pn_STREAM.Pn_STREAM_C[15:0]. PERIPHERAL PORTION asserts Pn_INT_STA.Pn_STRMX_STA in case Stream ID in the received packet doesn't equal to that in Pn_STREAM.Pn_STREAM_C[15:0], Prime nor NoStream. In that case, the Stream ID contained in the received packet is shown in Pn_NRDYSTRM register.

It is possible to give priority to the Stream ID requested from host by checking Pn_NRDYSTRM register, but it is assumed here that device requests another Stream ID it decided to process.

Especially, READ READY IU and WRITE READY IU are defined to select which stream (command) is processed next from device in UASP. ERDY(Stream) used in super speed Stream Bulk of UASP is alternative of them. So the device initiates the selection of stream in UASP.

Note: From the definition in USB3.0 specification, device selects the Stream by returning ERDY (Stream) may receive ACK (No Stream) from host. ACK (No Stream) means the rejection from host for proposed Stream ID in ERDY. In such case, device should consider the Stream as "Not Ready", and may select another Stream. It depends on the application whether No Stream is really used or not.

63.8.5.4 SuperSpeed UASP Non-Data Transfer

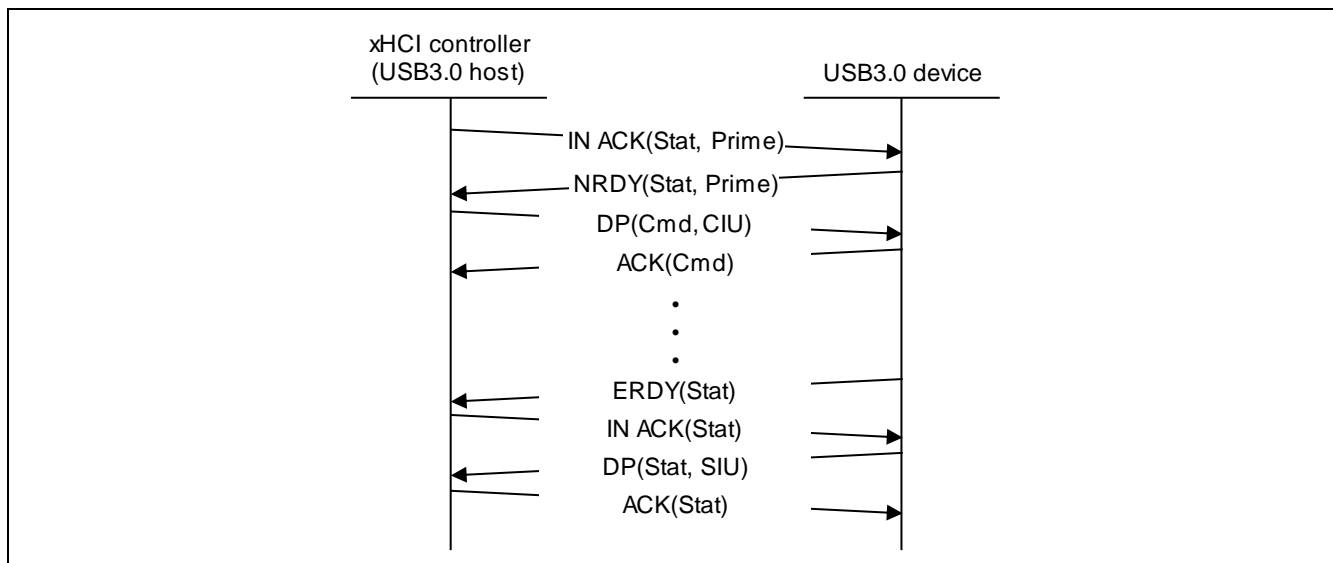


Figure 63.51 Example SuperSpeed UASP Non-Data Transfer

Figure 63.51 shows the part between USB3.0 host and USB3.0 device of Figure 6 in UASP specification Rev.1.0. The example is called as “Non-Data Transfer” but the status PIPE is working as the Stream bulk IN PIPE. In this case, IN ACK(Stat, Prime) comes to the status PIPE first, then the status PIPE returns NRDY(Stat, Prime). The some operation is requested on DP(Cmd, CIU) to the command PIPE. The command ID in CIU is used as Stream ID later. When the operation is completed, SIU as the result of the operation is returned from the status PIPE. Pn_CON.Pn_RES[1:0] is set to B'01 (normal response) at (4) of section 63.8.5.3 (2). When SIU is written to buffer of the status PIPE and it is ready to be transmitted, ERDY(Stat) is sent in order to resume the transfer. IN ACK(Stat) comes to the status PIPE, it sends DP(Stat, SIU) and DP(Stat, SIU) is acknowledged by ACK(Stat).

63.8.5.5 SuperSpeed UASP Data-IN transfer

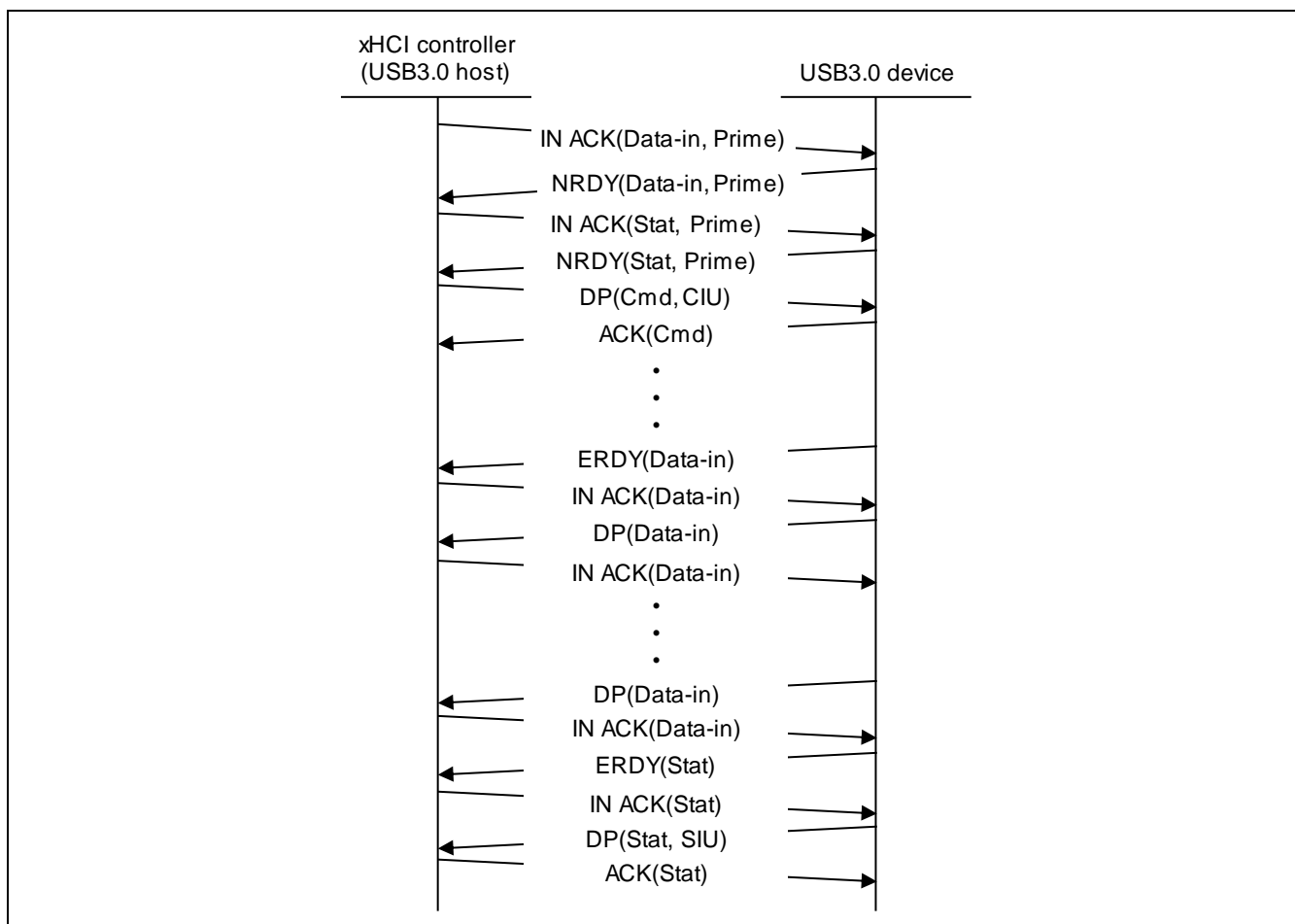


Figure 63.52 Example SuperSpeed UASP Data-IN transfer

Figure 63.52 shows the part between USB3.0 host and USB3.0 device of Figure 7 in UASP specification Rev.1.0.

In this case, the Data-in PIPE and the status PIPE are working as Stream bulk IN PIPE.

To set each PIPE in Prime Pipe state, IN ACK (Data-in, Prime) and IN ACK (Stat, Prime) comes. Both PIPEs return NRDY (Prime) to them and go back to Idle state.

The bulk IN operation is requested on DP (Cmd, CIU) to the command PIPE. The command ID in CIU is used as Stream ID later.

If Pn_CON.Pn_RES[1:0] of the Data-in PIPE is already set to B'01 (normal response), the Data-in PIPE sends ERDY (Data-in) when IN data becomes ready to be transmitted from it. The data transfer from the Data-in PIPE is started by receiving IN ACK (Data-in).

All data transfers from Data-in PIPE are completed, SIU is made as the result of the operation.

If Pn_CON.Pn_RES[1:0] of the status PIPE is already set to B'01 (normal response), the status PIPE sends ERDY (Stat) when SIU is written to buffer and is ready to be transmitted from it. The SIU is sent as DP (Stat, SIU) when IN ACK (Stat) comes.

63.8.5.6 SuperSpeed UASP Data-OUT transfer

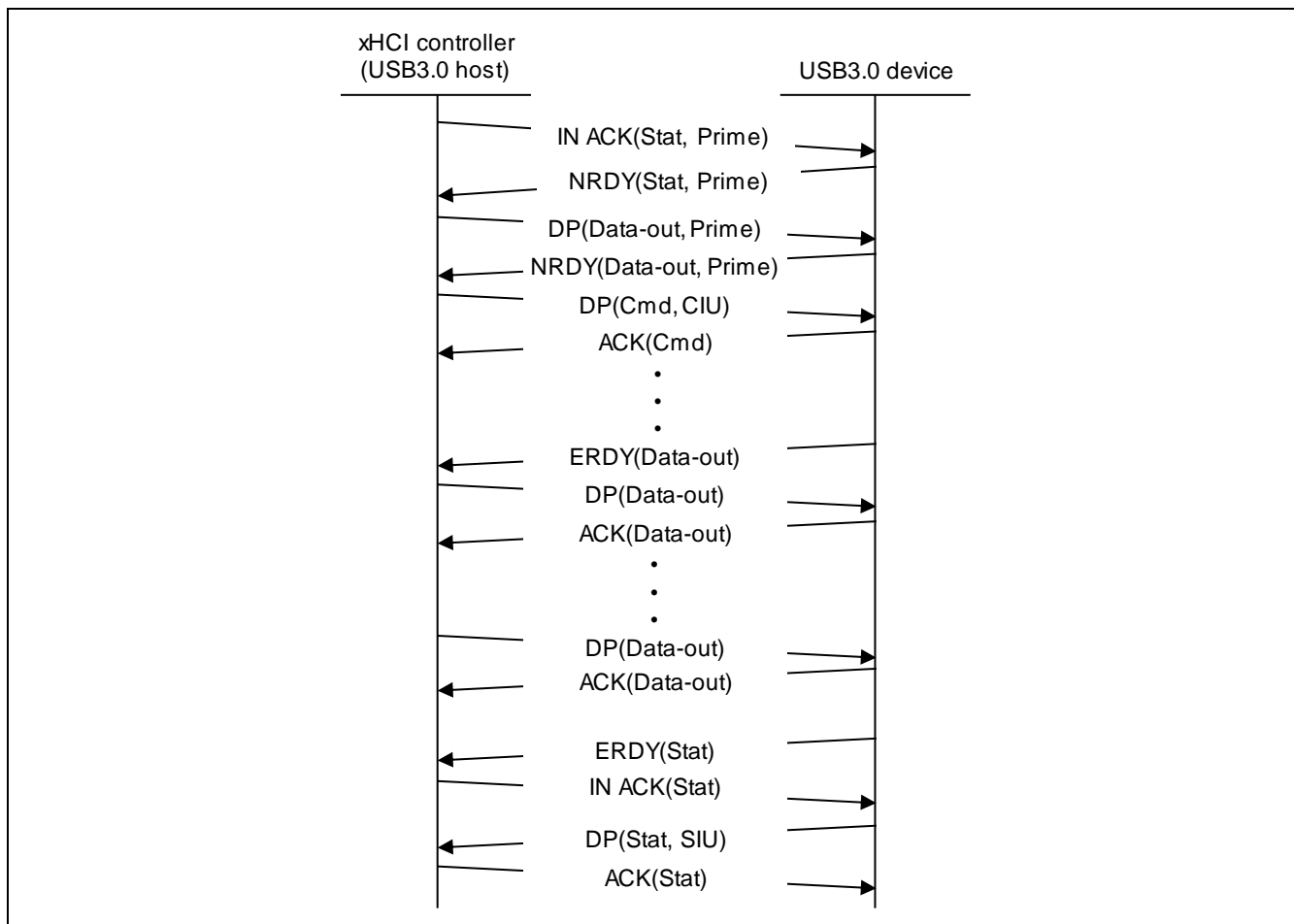


Figure 63.53 Example SuperSpeed UASP Data-OUT transfer

Figure 63.53 shows the part between USB3.0 host and USB3.0 device of Figure 8 in UASP specification Rev.1.0.

In this case, the status PIPE is working as Stream bulk IN PIPE, and the Data-OUT PIPE is working as Stream bulk OUT PIPE.

To set each PIPE in Prime Pipe state, IN ACK (Stat, Prime) and DP (Data-out, Prime) comes. Both PIPEs return NRDY(Prime) to them and go back to Idle state.

The bulk IN operation is requested on DP (Cmd, CIU) to the command PIPE. The command ID in CIU is used as Stream ID later.

When Pn_CON.Pn_RES[1:0] of the Data-out PIPE is changed from B'00 to B'01 (normal response, note that Pn_CON.Pn_RES[1:0] is set to B'00 once for Stream bulk PIPE when it receives DP (Prime) or ACK (Prime)), the Data-out PIPE sends ERDY (Data-out) if it has available space in buffer. The data transfer to the Data-out PIPE is started.

All data transfers to Data-out PIPE are completed, SIU is made as the result of the operation.

If Pn_CON.Pn_RES[1:0] of the status PIPE is already set to B'01 (normal response), the status PIPE sends ERDY (Stat) when SIU is written to buffer and is ready to be transmitted from it. The SIU is sent as DP (Stat, SIU) when IN ACK (Stat) comes.

63.8.6 Interrupt Transfer

Interrupt Transfer process is described here.

PIPEs other than PIPE0 (PIPE0 is used only for control transfer) can be used for interrupt transfer. The settings for interrupt transfer are shown below.

- PIPE_COM.PIPE_NUM[4:0] = the index of PIPE used for the current interrupt transfer
- PIPE_MOD.Pn_EPNUM[3:0] = the endpoint number of the PIPE
- PIPE_MOD.Pn_TYPE[1:0] = B'11 (interrupt transfer)
- PIPE_MOD.DIR = the direction of transfer
- PIPE_RAMMAP.Pn_BASEAD[13:0] = the base address of buffer assigned for the PIPE
- PIPE_RAMMAP.Pn_RAMIF[1:0] = the index of RAM interface of the PIPE
- PIPE_RAMMAP.Pn_MPKT[10:0] = the max packet size of the PIPE
- PIPE_RAMMAP.Pn_RAMAREA = the size of RAM area assigned for the PIPE
- Pn_CON.Pn_DATAIF_EN = whether Data Interface of the controller is used or not.
- Pn_CON.Pn_EN = 1 (the PIPE is enabled)

These settings are done just after Set Configuration is completed and the device enters configured state. But if the device has only single configuration for each PIPE, these settings can be done after the device is turned on.

63.8.6.1 Interrupt IN Transfer

In this subsection it is assumed that interrupt transfer is used to return small data periodically, so the process to write through Pn_WRITE register is described here.

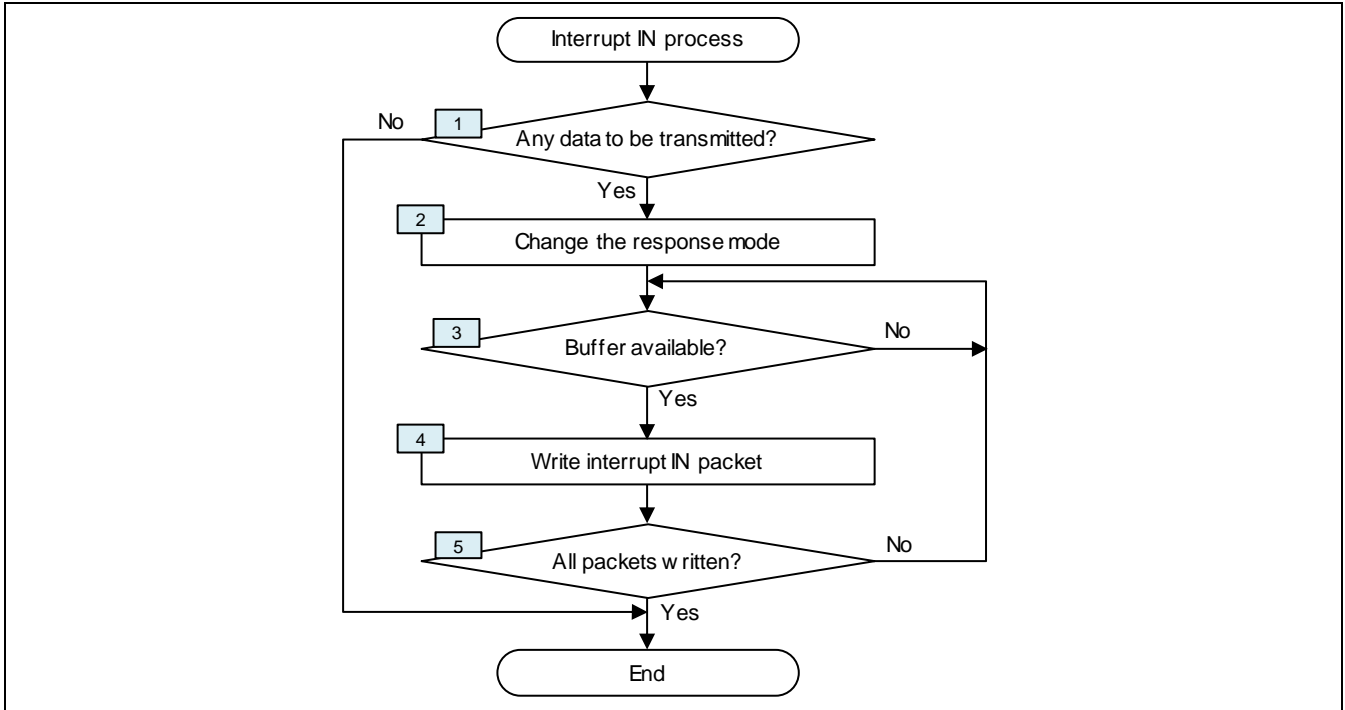


Figure 63.54 Interrupt IN process flow

- (1) Is there any data to be transmitted?
 Check if there is any data to be transmitted from interrupt IN PIPE.
 If not, set Pn_CON.Pn_RES[1:0] = B'00 so that the PIPE returns NRDY or NAK for IN request. When Pn_CON.Pn_RES[1:0] already has the value, leave it as it is.

 - Pn_CON.Pn_RES_WEN = 1
 - Pn_CON.Pn_RES[1:0] = B'00 (NRDY/NAK response)

 If there is, proceed to (2).
- (2) Change the response mode of the interrupt IN PIPE
 As it is confirmed that there is data to be transmitted at (1), change the response mode of the interrupt IN PIPE to enable the transmission of a packet.

 - Pn_CON.Pn_RES_WEN = 1
 - Pn_CON.Pn_RES[1:0] = B'01 (normal response)
- (3) Check if buffer is available
 Check if buffer for the interrupt IN PIPE is available before writing interrupt IN packet. Pn_STA.Pn_BUFSTS shows if the buffer is available or not.

 - Check whether Pn_STA.Pn_BUFSTS == 1 or not

 If Pn_STA.Pn_BUFSTS is 0 and there is no available space in the buffer currently, Wait until Pn_STA.Pn_BUFSTS becomes 1 or an interrupt due to Pn_INT_STA.Pn_BFRDY_STA is asserted. When IN is

selected as the direction of the interrupt PIPE (that is, `Pn_MOD.Pn_DIR==1`), `Pn_INT_STA.Pn_BFRDY_STA` is asserted when the buffer for the PIPE becomes available for new write data.

Therefore, the software can know the timing the buffer of the PIPE prepares new available space by polling `Pn_STA.Pn_BUFSTS` or an interrupt due to `Pn_INT_STA.Pn_BFRDY_STA`.

(4) Write interrupt IN packet

Write interrupt IN packet to be transmitted to the buffer. Interrupt IN data is written in a unit of packet and the buffer can't accept data across the max packet size defined in `PIPE_RAMMAP.Pn_MPKT[10:0]`.

[writing through `Pn_WRITE` register]

a-1) Write data through `Pn_WRITE` register. Since only 32-bit access is allowed to the register, data should be written in a unit of 32-bit. The number of write can be calculated as the number of write = the length divided by 4, round-up is required if there is any residue. The written words are concatenated as one packet.

When the length of packet is not multiples of 32-bit, the last written word has unnecessary bytes.

The last word should be also written as a 32-bit word. The number of valid bytes in the last word is set in `Pn_CON.Pn_BYTE_EN[1:0]`. Unnecessary bytes in the last word are discarded.

a-2) Set fields as follows to send the packet. Since only 32-bit access is allowed to `Pn_CON` register, all fields below should be set in one word access.

- `Pn_CON.Pn_SEND = 1`

- `Pn_CON.Pn_BYTE_EN[1:0] = B'xx` (The number of valid bytes in last written word)

- In case of the last packet of data group, set `Pn_CON.Pn_LAST = 1`

(5) Are all packets written?

If it is required to send two or more packets, repeat the steps from (3) to (4).

Note: When the information of a certain PIPE is read from or write to `PIPEn` registers, `PIPE_COM.PIPE_NUM[4:0]` should have the index of the PIPE.

63.8.7 DMA Transfer

63.8.7.1 Settings for DMA transfer

Data transfer using DMA in Peripheral portion is described here.

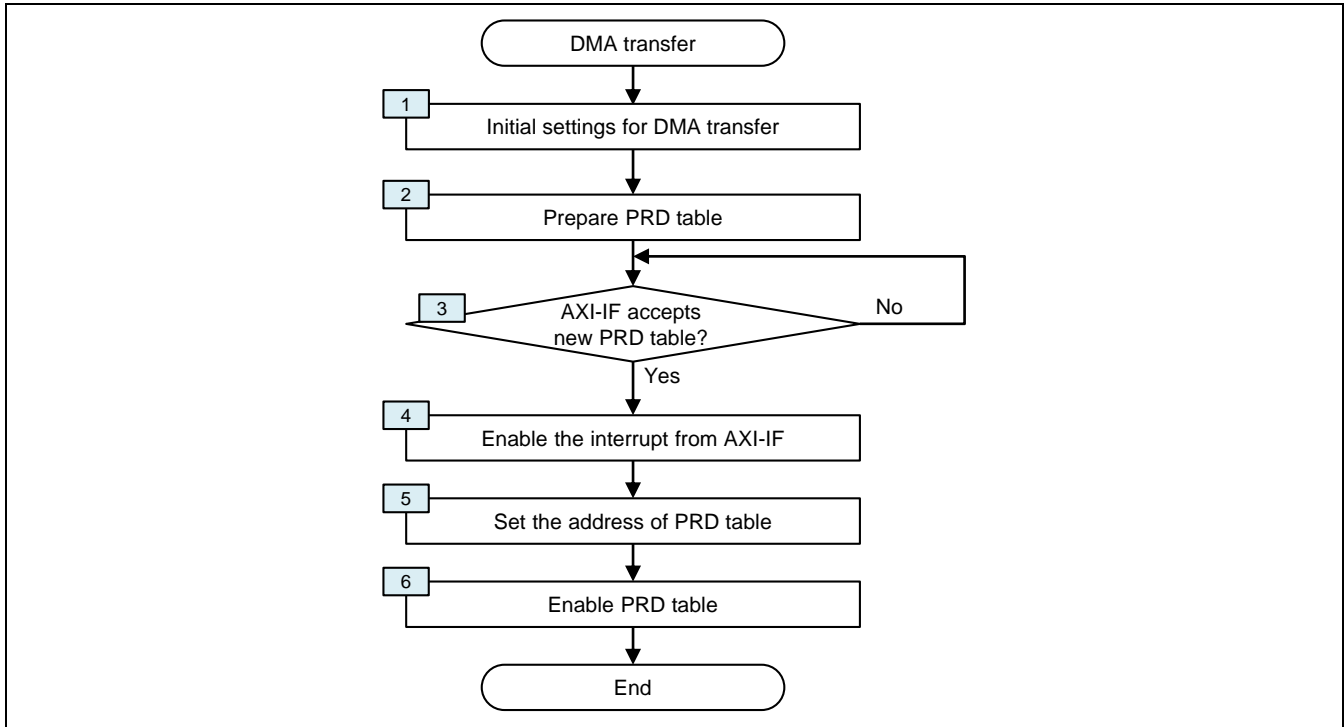


Figure 63.55 Transfer Setting Flow

(1) Initial settings for DMA transfer

Change the response mode of the PIPE to enable data transfer.

Since USB3.0 Controller uses EPC_D_Pn_DATAEN[30:1] to see if the PIPE is ready for the transfer or not, Pn_CON.Pn_DATAIF_EN should be enabled.

- Pn_CON.Pn_RES_WEN = 1
- Pn_CON.Pn_RES[1:0] = B'00 (NRDY/NAK response)
- Pn_CON.Pn_DATAIF_EN = 1

(2) Prepare PRD table

Prepare PRD table in system memory following the format defined in section 63.7.8.

(3) Can Peripheral portion accept new PRD table?

Peripheral portion handles up to 4 PRD tables concurrently. If the number of PRD tables running in Peripheral portion is less than 4, new PRD table can be set for Peripheral portion. The “PRD table running” means here that PRD table is assigned to one of DMA_Ch0_PRD_ADRx register and DMA_Ch0_CONx.PRD_ENx is enabled for it.

If there is any DMA_Ch0_CONx in which PRD_ENx is disabled (= 0), new transfer can be assigned to it.

If there is no DMA_Ch0_CONx with PRD_ENx = 0, wait for the completion of one of PRD tables running.

(4) Enable interrupt from AXI-IF

Enable the assertion of an interrupt from Peripheral portion if it is disabled, and clear the interrupt status registers.

But the status register might have the interrupt status of completed transfer, and it is also possible that the concurrent transfer is finished and set its status in registers just when the software clears them for new transfer.

Make sure not to clear statuses of other PIPE or other DMA_Ch0_CONx when clearing the status of PIPE and

DMA_Ch0_CONx to be used. The bits in status register have “write 1 to clear” function, which means it is not cleared when writing 0. Write 1 just to the bit to be cleared.

- DMA_INT_STA.Pxx_PRDEND_STA = 1 (The status of the PIPE to be used is cleared)
- AXI_INT_STA.PRDENx_CLR_STA = 1 (The status of DMA_Ch0_CONx to be used is cleared)
- AXI_INT_STA.PRDERRO_x_STA = 1 (The error status of DMA_Ch0_CONx to be used is cleared)

After clearing the interrupt status, enable the interrupt for the PIPE and DMA_Ch0_CONx to be used.

- AXI_INT_ENA.DMA_INT_ENA = 1
- AXI_INT_ENA.PRDENx_CLR_ENA = 1
- AXI_INT_ENA.PRDERRO_x_STA = 1

(5) Set the address of PRD table

Set the address of PRD table generated at (2) in DMA_Ch0_PRD_ADRx register. If the transfer consists of some PRD tables, set the address of the first PRD table.

- DMA_Ch0_PRD_ADRx = the address of PRD table

(6) Enable PRD table

Set the information of PRD table to DMA_Ch0_CONx register, and enable it on DMA_Ch0_PRD_CONx.PRD_ENx. The fields below can be set in one word access.

- DMA_Ch0_PRD_CONx.PIPE_DIRx = the direction of the PIPE
- DMA_Ch0_PRD_CONx.PIPE_NOx = the index of the PIPE
- DMA_Ch0_PRD_CONx.PRD_ENx = 1

In case the transfer of new PRD table is expected to wait for the completion of one of the concurrent transfer and be executed after it, use DMA_Ch0_PRD_CONx.EXTPRD_ENx. See section 63.7.5 for details of DMA_Ch0_PRD_CONx.EXTPRD_ENx.

- DMA_Ch0_PRD_CONx.EXTPRD_NOx = the index of DMA_Ch0_PRD_CONy which completion of transfer new PRD table is waiting for.
- DMA_Ch0_PRD_CONx.EXTPRD_ENx = 1 (the use of EXTPRD_EN is enabled)

Note that EXTPRD_ENx is not cleared if PRD_ENy of DMA_Ch0_PRD_CONy which completion is waited for is already 0. That is, EXTPRD_ENx can't be used for DMA_Ch0_PRD_CONy which has already cleared its enable bit (PRD_ENy). Make sure that PRD_ENy is still 1 and the related transfer is still being executed when EXTPRD_EN is associated. When you find PRD_ENy is already cleared to 0, don't use EXTPRD_EN. In that case, there is no need to wait for the completion of DMA_Ch0_PRD_CONy since it has already been finished.

63.8.8 Power Management

63.8.8.1 Suspend in USB3.0 (transition of state from U0 to U3)

The process to transit to U3 state of LTSSM in USB3.0 specification is described here.

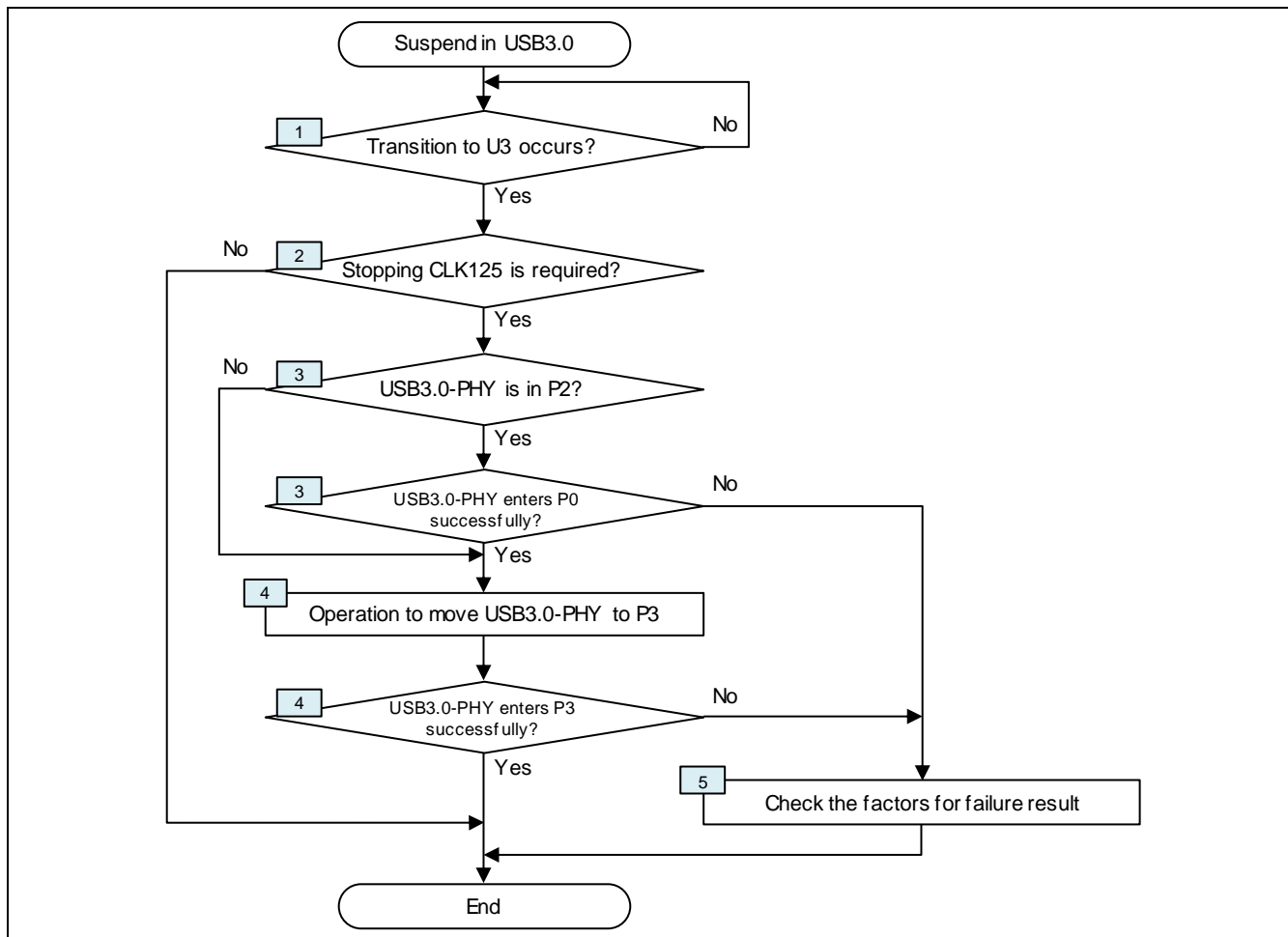


Figure 63.56 Suspend in USB3.0 flow

(1) Transition to U3 is informed

It is defined that only a downstream port (host or hub) can initiate U3 entry and an upstream port can't reject it in USB3.0 specification. The downstream port initiates U3 entry by sending LGO_U3.

When the controller receives LGO_U3, it accepts it and executes the transition to U3 inside. At this time,

USB_INT_STA_1.B3_LNKCNG_STA is asserted since the change of the link status occurs. The current status of LTSSM can be known by seeing PORTSC.PLS[3:0].

Check whether the transition to U3 is requested or not when LTSSM is in U0 state and

USB_INT_STA_1.B3_LNKCNG_STA is asserted. If LTSSM has changed to U3 state, the transition to U3 has been requested from downstream port.

- Interrupt due to USB_INT_STA_1.B3_LNKCNG_STA is generated

- Check whether PORTSC.PLS[3:0] shows U3 or not.

Note that there is another possibility that disconnection or error occurs as the factor of the change of the link status. In such case, follow the flows noted in other sections.

After confirming the factor of the change, clear `USB_INT_STA_1.B3_LNKCNG_STA`. Then, proceed to (2).

- `USB_INT_STA_1.B3_LNKCNG_STA = 1` (Clearing the event that the link has changed)

(2) Is stopping USB3.0-PHY clock required?

LTSSM in Peripheral portion has transited to U3 at (1), but USB3.0-PHY is in P0 or P2 state of USB3.0-PHY state since only P0 or P2 can be selected as the setting of `USB30_CON.U3_POW_SEL[1:0]`. Note the difference between U0/U1/U2/U3 and P0/P1/P2/P3. Ux is the power management states in LTSSM. Px is the power management states for USB3.0-PHY and the upper layer. `USB30_CON.U3_POW_SEL[1:0]` defines the PHY state in which USB3.0-PHY is placed when the transition to U3 is requested in LTSSM. P3 is defined as the state in which the clock is suspended. The reason why P3 is not selected as the setting of `USB30_CON.U3_POW_SEL[1:0]` is that it may cause issues if the clock from USB3.0-PHY is stopped suddenly.

If USB3.0-PHY is not required to be placed in P3 when the upper layer is in U3 state, no additional operation is required here. If it is required, proceed to (3).

(3) Is USB3.0-PHY placed in P2?

As noted above, P0 or P2 can be selected as USB3.0-PHY state on `USB30_CON.U3_POW_SEL[1:0]` when LTSSM of upper layer enters U3. If USB3.0-PHY is in P2 at present and expected to enter P3, it should be move to P0 once before it is operated to enter P3.

- `USB30_CON.POW_SEL_WEN = 1`

- `USB30_CON.POW_SEL[2:0] = B'110` (the transition from P2 to P0 is requested when LTSSM is in U3)

When the transition to P0 requested has been done, an interrupt due to the transition is asserted. Since there is no reason to fail in the transition, so USB3.0-PHY surely enters P0, but confirm the result of the transition.

- Interrupt due to the transition asserted

- Check whether `USB_INT_STA_1.B3_PSSUCS_STA` is asserted or not. If it is asserted, the transition has been done successfully. Clear it after the confirmation.

- If `USB_INT_STA_1.B3_PSSUCS_STA` is not asserted, check whether `USB_INT_STA_1.B3_PSFALL_STA` is asserted or not. If it is asserted, the presumed factors are explained at (5).

If `USB30_CON.U3_POW_SEL[1:0]==B'00` (P0) and USB3.0-PHY is placed in P0 when LTSSM enters U3, this operation is not required.

Hence, USB3.0-PHY is P0 at present.

(4) Operation to move USB3.0-PHY to P3

Set `USB30_CON.POWSEL[2:0]` to place USB3.0-PHY in P3.

- `USB30_CON.POW_SEL_WEN = 1`

- `USB30_CON.POWSEL[2:0] = B'101` (the transition from P0 to P3 is requested when LTSSM is in U3)

When the transition to P3 requested has been done, an interrupt due to the transition is asserted. Since there is no reason to fail in the transition, so USB3.0-PHY surely enters P3, but confirm the result of the transition.

- Interrupt due to the transition asserted

- Check whether `USB_INT_STA_1.B3_PSSUCS_STA` is asserted or not. If it is asserted, the transition has been done successfully. Clear it after the confirmation.

- If `USB_INT_STA_1.B3_PSSUCS_STA` is not asserted, check whether `USB_INT_STA_1.B3_PSFALL_STA` is asserted or not. If it is asserted, the presumed factors are explained at (5).

(5) Check the factors for failure result

If case the transition of power state noted here failed, the presumed factors are as follows.

- (1) the disconnection has occurred and LTSSM transits to SS.Disabled.
- (2) U3Wakeup LFPS from host has come and LTSSM has started the transition to U0.
- (3) the warm reset from host has come

See the statuses below to know whether the factors above occur or not.

- USB_INT_STA_1.B3_DISABLE_STA = 1 (if (1) occurred)
- PORTSC.PLS[3:0] = H'0 (if (2) occurred)
- USB_INT_STA_1.B3_WRM_RST_STA = 1 (if (3) occurred)

If (1) occurs, see section 63.8.3.3 for the disconnection process.

If (2) occurs, see section 63.8.8.2 for the resume process from U3.

If (3) occurs, see section 63.8.3.4 for the warm reset process.

In case the transition of power state failed but the factors above has not caused it, fatal error might occur in the system.

63.8.8.2 Resume in USB3.0 (from U3 to U0: Initiated by host)

The process for the transition to U0 initiated by host in USB3.0 is described here.

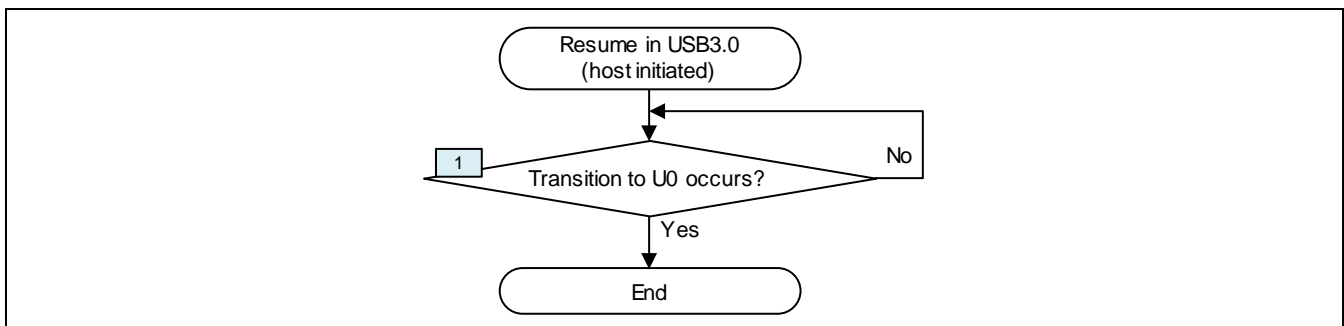


Figure 63.57 Resume in USB3.0 (host initiated) flow

(1) Does the transition to U0 occur?

In case Peripheral portion is in U3 and receives U3Wakeup LFPS from host, it automatically responds to the LFPS and go back to U0. At this time, USB3.0-PHY is moved back to P0 as well.

The event is notified on the assertion of USB_INT_STA_1.B3_LNKCNG_STA. An interrupt due to the bit is asserted if enabled.

- Interrupt due to USB_INT_STA_1.B3_LNKCNG_STA is asserted, if enabled
- Check that USB_INT_STA_1.B3_LNKCNG_STA == 1
- Check that PORTSC.PLS[3:0] shows U0 (= H'0)

After the confirmation above, clear USB_INT_STA_1.B3_LNKCNG_STA.

- USB_INT_STA_1.B3_LNKCNG_STA = 1 (clearing the event of the link change)

If USB3.0-PHY is placed in P3 following the flow in section 63.8.8.1 and the clock from it is stopped in U3, the clock is also resumed as USB3.0-PHY goes back to P0. USB_INT_STA_1.B3_PLLWKUP_STA is asserted in that case. If it is asserted, clear it.

- USB_INT_STA_1.B3_PLLWKUP_STA = 1 (clearing the event of USB3.0 PLL Wakeup)

63.8.8.3 Resume in USB3.0 (from U3 to U0: Initiated by device)

The process for the transition to U0 initiated by device in USB3.0 is described here.

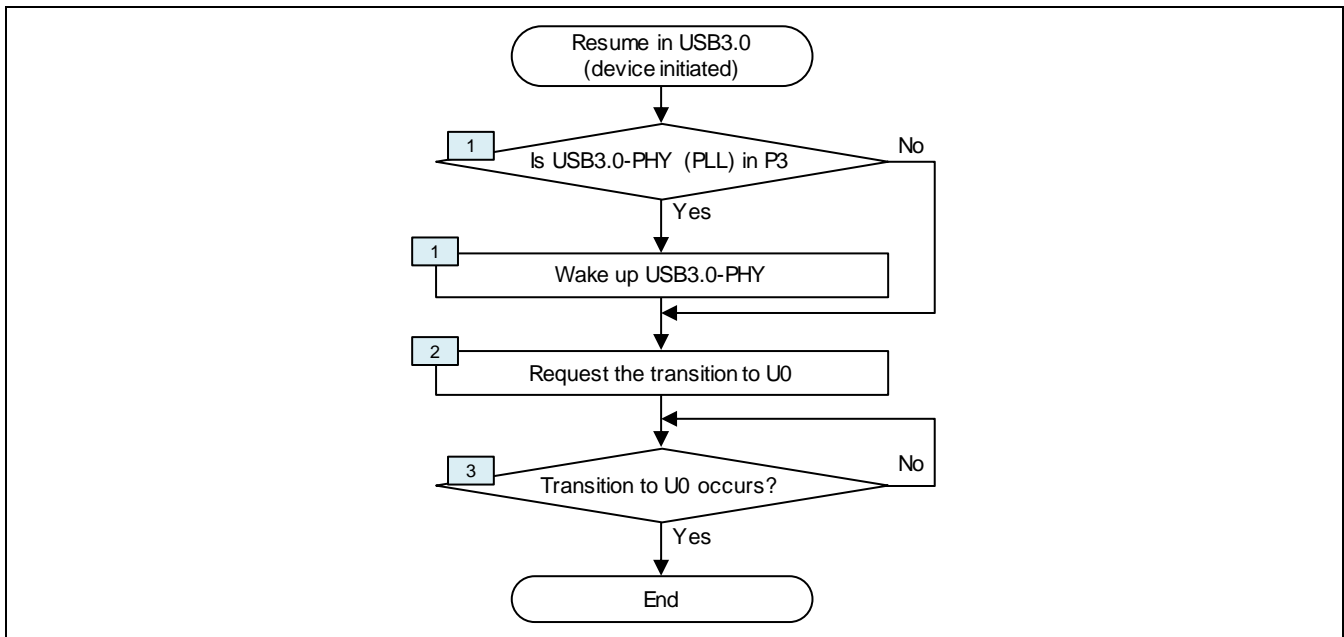


Figure 63.58 Resume in USB3.0 (device initiated) flow

(1) Is USB3.0-PHY in P3?

If USB3.0-PHY is placed in P3 following the flow in section 63.8.8.1 and the clock from it is stopped in U3, it is required to wake up USB3.0-PHY since the clock from USB3.0-PHY should be provided for the operations here. See section 63.8.8.6 for the process of wakeup of USB3.0-PHY.

If USB3.0-PHY is P0 or P2 and the clock from it is being provided, it is not required to wake up USB3.0-PHY.

(2) Request the transition to U0

Request the transition to U0 in LTSSM as follows.

- PORTSC.LWS = 1
- PORTSC.PLS[3:0] = H'0 (U0 state is requested)

(3) Does the transition to U0 occur?

With the request to transit to U0, Peripheral portion asserts U3Wakeup LFPS to host. When host responds to the LFPS and LFPS handshake is established, Peripheral portion has transited to U0.

At that time, USB_INT_STA_1.B3_LNKCNG_STA is asserted and an interrupt due to it is caused if enabled.

- Interrupt due to USB_INT_STA_1.B3_LNKCNG_STA is asserted, if enabled
- Check that USB_INT_STA_1.B3_LNKCNG_STA == 1
- Check that PORTSC.PLS[3:0] shows U0 (= H'0)

After the confirmation above, clear USB_INT_STA_1.B3_LNKCNG_STA.

- USB_INT_STA_1.B3_LNKCNG_STA = 1 (Clearing the event of the link change)

63.8.8.4 Function Suspend

The process for function suspend in USB3.0 is described here.

The function suspend is defined for SuperSpeed device and functions are placed into function suspend on the reception of Set Feature(FUNCTION_SUSPEND). Since function has one or more interfaces, what function is selected to be suspend is recognized by seeing what interface receives the request. If the function has multiple interfaces, the request comes to the first interface of the function.

Note that the specification defines about the function suspend as follows.

- A function may be placed into function suspend independently of other functions within the device.
- A device may be transitioned into device suspend regardless of the function suspend state of any function within the device.
- Function suspend state is retained while in device suspend and throughout the device suspend entry and exit processes.

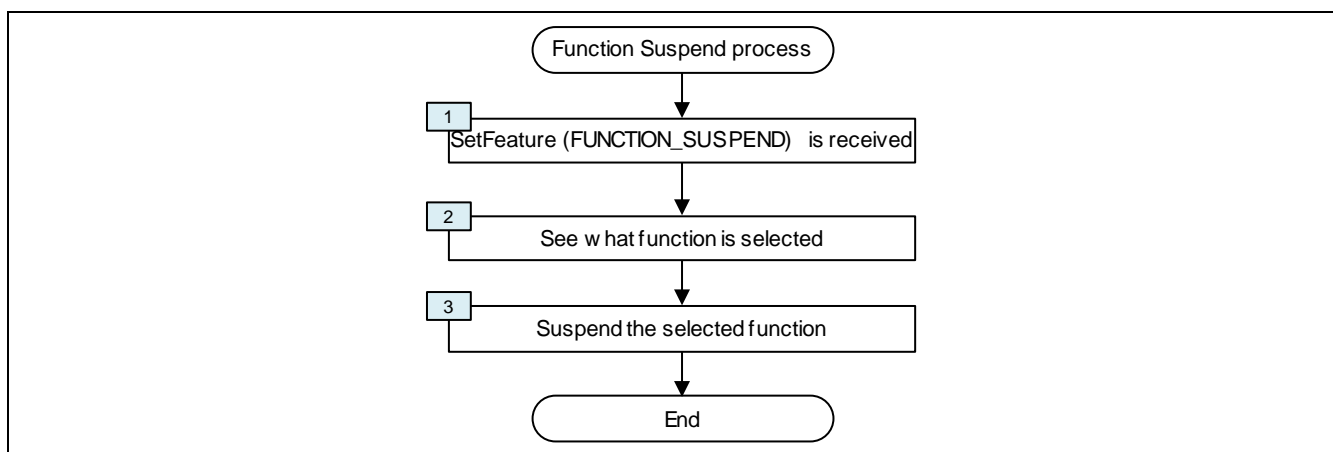


Figure 63.59 Function Suspend process

- (1) Set Feature(FUNCTION_SUSPEND) is received
In case Set Feature(FUNCTION_SUSPEND) comes, the function suspend is requested. The function suspend is requested only via Set Feature(FUNCTION_SUSPEND).
- (2) See what function is selected
See what function within the device is selected. The interface number of the function is specified in the lower byte of wIndex. If the function has multiple interfaces, the lower byte of wIndex shall be the first interface of the function.
Bit 0 in the upper byte of wIndex shows which of normal state (= 0) or suspended state (= 1) is requested. When bit 0 is 1 and the function has been enabled, the selected function is requested to suspend. When bit 0 is 0 and the function has been suspended, the selected function is requested to exit from suspended state.
Bit 1 in the upper byte for wIndex shows whether the function remote wakeup is enabled or not.
- (3) Suspend the selected function
Suspend the function which is selected at (2).
If the function can't be suspended, the device may be allowed to return STALL response to Set Feature (FUNCTION_SUSPEND).

63.8.8.5 Function Remote Wakeup

The process of function remote wakeup is described here.

The function remote wakeup is defined for SS device and DEVICE_REMOTE_WAKEUP is not supported by SuperSpeed device.

As a function may be suspended independently of other functions, the link is still in U0 state even when one or more functions are suspended. In case the suspended function within the device requires to be woken up and the link is U0 state, the device simply sends Function Remote Wakeup Notification. In case the function is requesting the remote wake up but the link is in the low power state (U1, U2 or U3), resume the link to U0 state first and then sends Function Remote Wakeup Notification.

The function shows whether it supports the function remote wakeup or not in the response to Get Status from the interface which belongs to the function.

It is also required that Function Remote Wake is enabled to request the function remote wakeup from the device. Function Remote Wake is disabled (= 0) in the default state and set when Set Feature (FUNCTION_SUSPEND) is received and bit 1 of the upper byte of wIndex in setup data is 1. The current status of Function Remote Wake is shown in the response to GetStatus from the interface which belongs to the function.

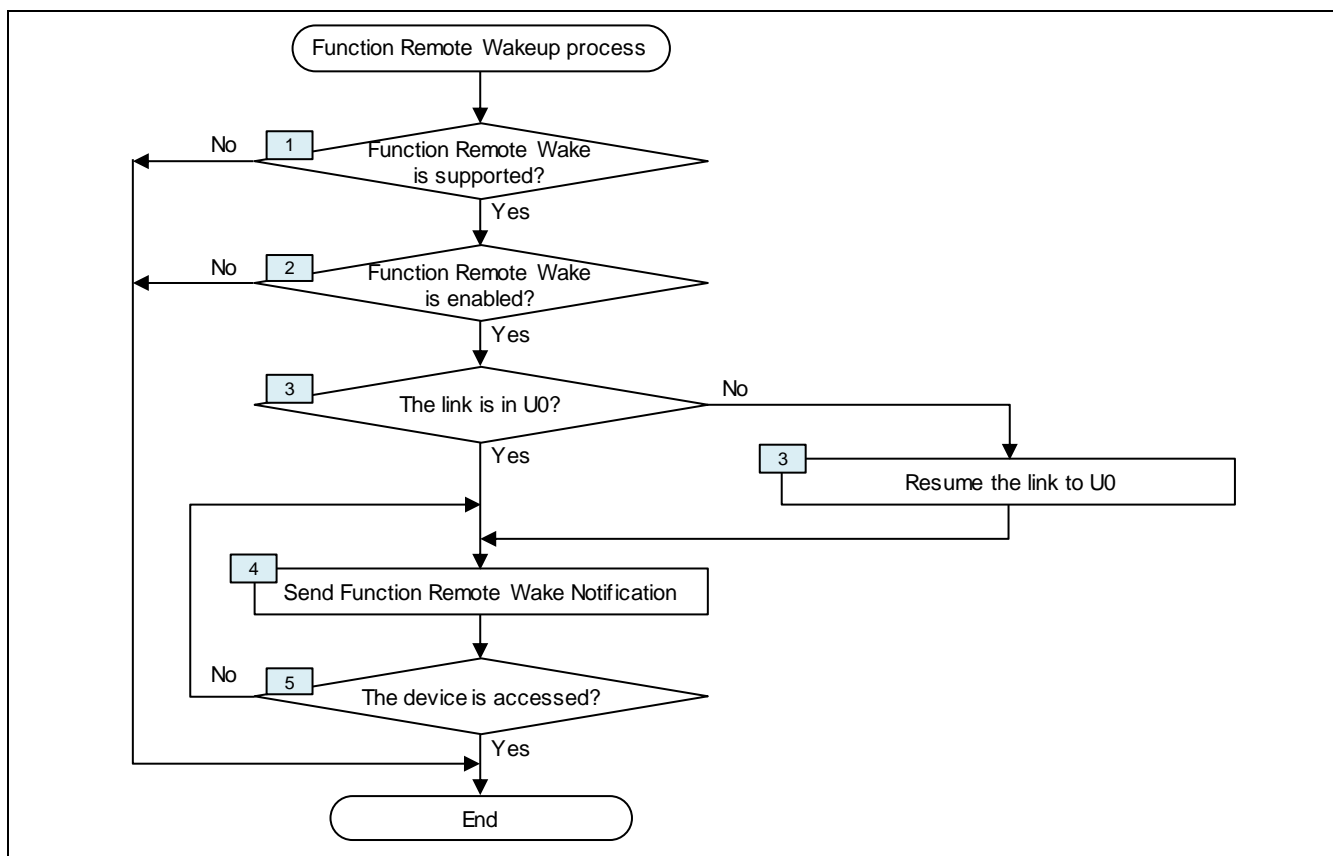


Figure 63.60 Function Remote Wakeup process

- (1) Is Function Remote Wake supported?
If the function within the device doesn't support Function Remote Wake, it can't request the function remote wakeup. The function shows whether it supports Function Remote Wake or not in the response to Get Status from the interface which belongs to the function. See section 63.8.9.5 for details.
- (2) Is Function Remote Wake enabled?
If Function Remote Wake is disabled for the function within the device, it can't request the function remote wakeup. The current status of Function Remote Wake is shown in the response to Get Status from the interface which belongs to the function. See section 63.8.9.5 for details.
- (3) The link is in U0 state?
To send Function Remote Wake Notification, the link is required to be in U0 state.
If PERIPHERAL PORTION is in U1 or U2 state, it automatically goes back to U0 state when Notification written in USB3_TP_DAT_x register is sent. Therefore, the software need not mind the link status except the case the link is U3. Proceed to (4).
If PERIPHERAL PORTION is in U3 state, it is required to exit from the state first. See 63.8.8.6 for the process of resume in USB3.0.
- (4) Send Function Remote Wake Notification
If the link is in U0 state, then send Function Remote Wake Notification.
 - Write Function Remote Wake Notification to USB3_TPDAT_x (x=0,1,2) registers
 - USB30_CON.B3_TP_SEND = 1 (Notification in USB3_TPDAT_x (x=0,1,2) registers is sent)
- (5) Is the device accessed?
If tNotification (= 2500ms) has passed and the device has not been accessed yet from the time Function Remote Wake Notification was sent, it is required to send Function Remote Wake Notification again. The device should send the notification until the function is resumed.
When Set Feature(FUNCTION_SUSPEND) with bit 0 = 1 in the upper byte of wIndex comes, the function is resumed.

63.8.8.6 USB3.0-PHY (PLL) Wakeup Process

The process to wake up USB3.0-PHY (PLL) in case it is suspended is described here.

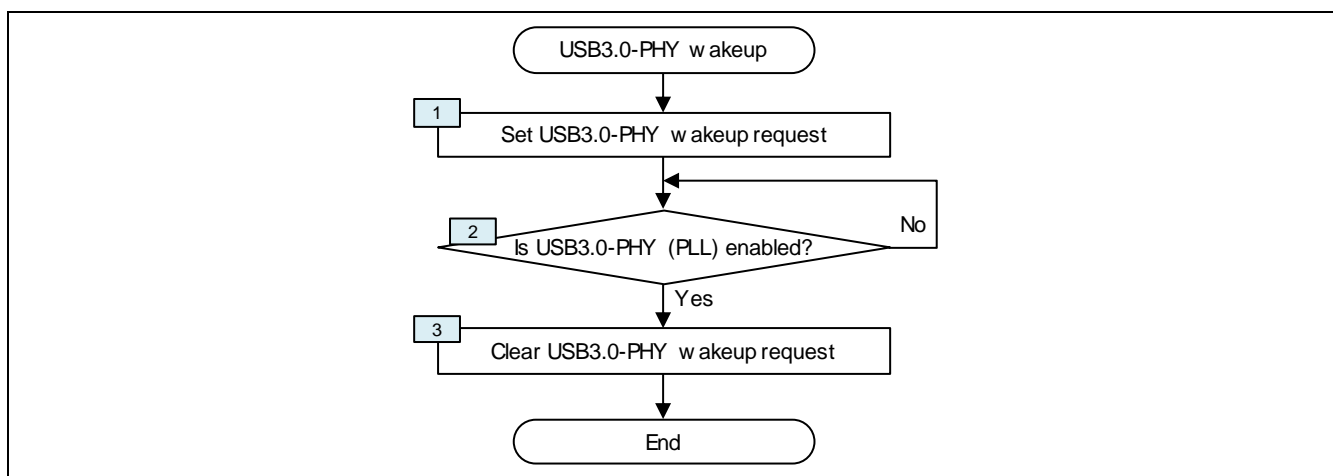


Figure 63.61 USB3.0-PHY (PLL) wakeup process flow

(1) Set USB3.0-PHY wakeup request

There are 2 ways to request USB3.0-PHY to wake up.
The first way is to set USB30_CON.B3_PLLWAKE.

[In case register is used]

- USB30_CON.B3_PLLWAKE = 1

(2) Is USB3.0-PHY (PLL) enabled?

Responding to the wake up request at (1), USB3.0-PHY (PLL) becomes enabled.

At that time, USB_INT_STA_1.B3_PLLWKUP_STA is asserted and an interrupt due to it is generated if enabled.

- Interrupt due to USB_INT_STA_1.B3_PLLWKUP_STA is generated

- Check that USB_INT_STA_1.B3_PLLWKUP_STA==1

After the confirmation above, clear USB_INT_STA_1.B3_PLLWKUP_STA.

- USB_INT_STA_1.B3_PLLWKUP_STA = 1 (Clearing the event of USB3.0 PLL Wakeup)

(3) Clear USB3.0-PHY wakeup request

After the resume of USB3.0-PHY is confirmed at (2), clear USB3.0-PHY wakeup request set in (1).

[In case register is used]

- USB30_CON.B3_PLLWAKE = 0

[In case input signal is used]

- B3_PLL_WAKEIN = 0

63.8.8.7 Suspend in USB2.0

The process when the state transition to suspend in USB2.0 occurs is described here.

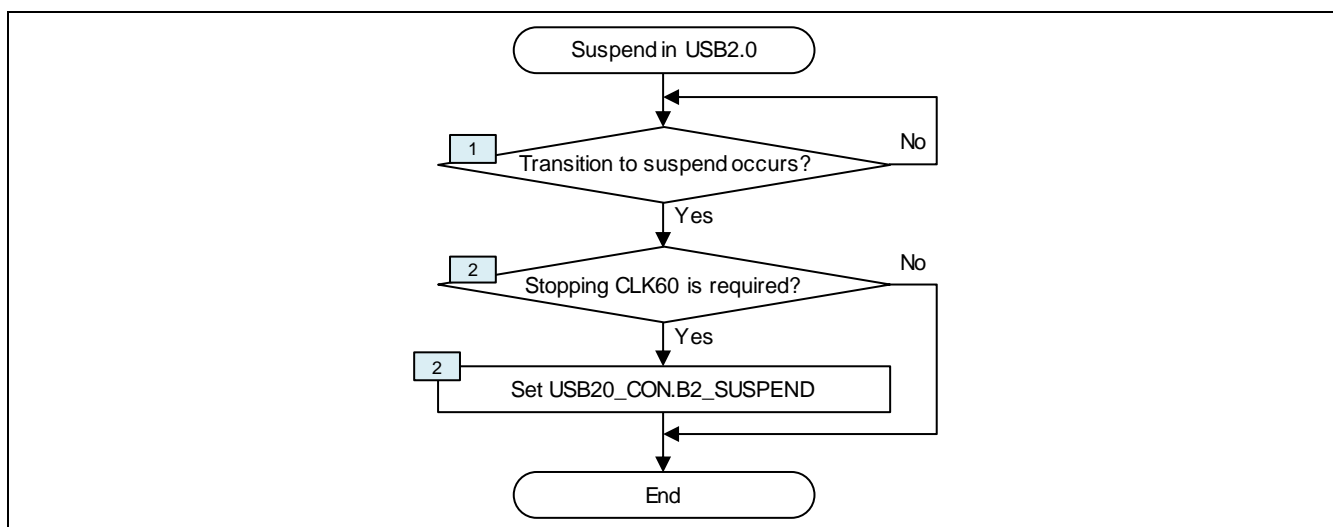


Figure 63.62 Suspend in USB2.0 process flow

(1) Does the transition to suspend occur?

In USB2.0 (HS/FS) host requests device to make a transition to suspended state by sending no signal to it for a certain period. Device transits to suspended state when it detects constant idle state on USB bus for more than 3.0ms.

When the controller works as HS or FS device and finds constant idle state on USB bus and make the transition to suspended state, it asserts `USB_INT_STA_1.B2_SPND_STA`. An interrupt due to it is generated if enabled.

- Interrupt due to `USB_INT_STA_1.B2_SPND_STA` is generated, if enabled
- Check whether `USB_INT_STA_1.B2_SPND_STA==1` or not.

After the confirmation above, clear `USB_INT_STA_1.B2_SPND_STA`.

- `USB_INT_STA_1.B2_SPND_STA = 1` (Clearing the event of entering USB2.0 suspension)

(2) Is stopping USB2.0-PHY clock required?

PLL in USB2.0-PHY is still running even when the controller enters the suspended state of USB2.0. In order to stop PLL in USB2.0-PHY, set `USB20_CON.B2_SUSPEND`

- `USB20_CON.B2_SUSPEND = 1` (Clock from USB2.0-PHY is stopped)

If it is not required to stop the clock from USB2.0-PHY, no operation is needed here.

63.8.8.8 Resume in USB2.0 (Initiated by host)

The process to resume in USB2.0 initiated by host is described here.

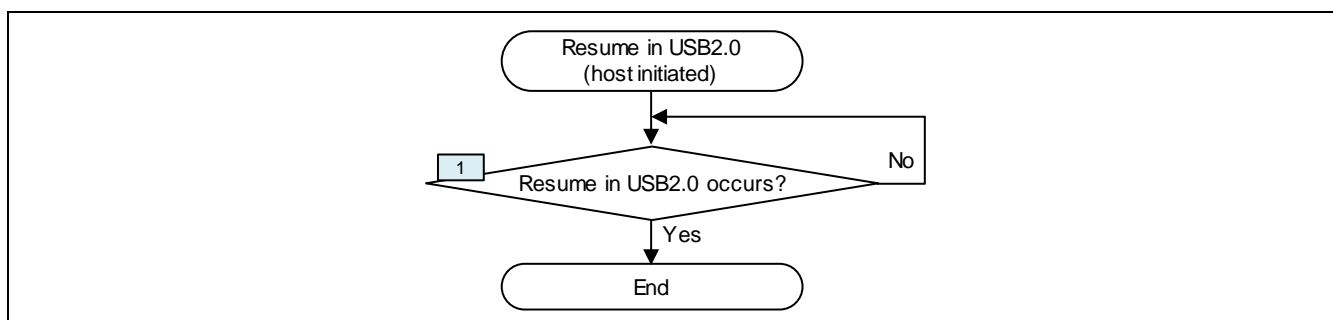


Figure 63.63 Resume in USB2.0 (Initiated by host) process flow

(1) Does the resume in USB2.0 occur?

In case Peripheral portion is in suspended state and receives the resume signal from host, it responds to it and exits from the suspended state automatically.

At that time Peripheral portion asserts `USB_INT_STA_1.B2_RSUM_STA` or `USB_INT_STA_1.B2_L1RSUM_STA` depending on the low power state it stayed in. An interrupt due to them is generated if enabled.

- Interrupt due to `USB_INT_STA_1.B2_RSUM_STA` or `USB_INT_STA_1.B2_L1RSUM_STA` is generated, if enabled.
- Check whether `USB_INT_STA_1.B2_RSUM_STA` or `USB_INT_STA_1.B2_L1RSUM_STA` == 1 or not.

After the confirmation above, clear the status bit asserted.

- `USB_INT_STA_1.B2_RSUM_STA` = 1 (In case of resume from suspend)
- `USB_INT_STA_1.B2_L1RSUM_STA` = 1 (In case of resume from sleep)

If USB2.0 PLL is stopped in the suspend state, it is also resumed when Peripheral portion exits from the suspend state.

63.8.8.9 Resume in USB2.0 (Initiated by device)

The process to resume in USB2.0 initiated by device is described here.

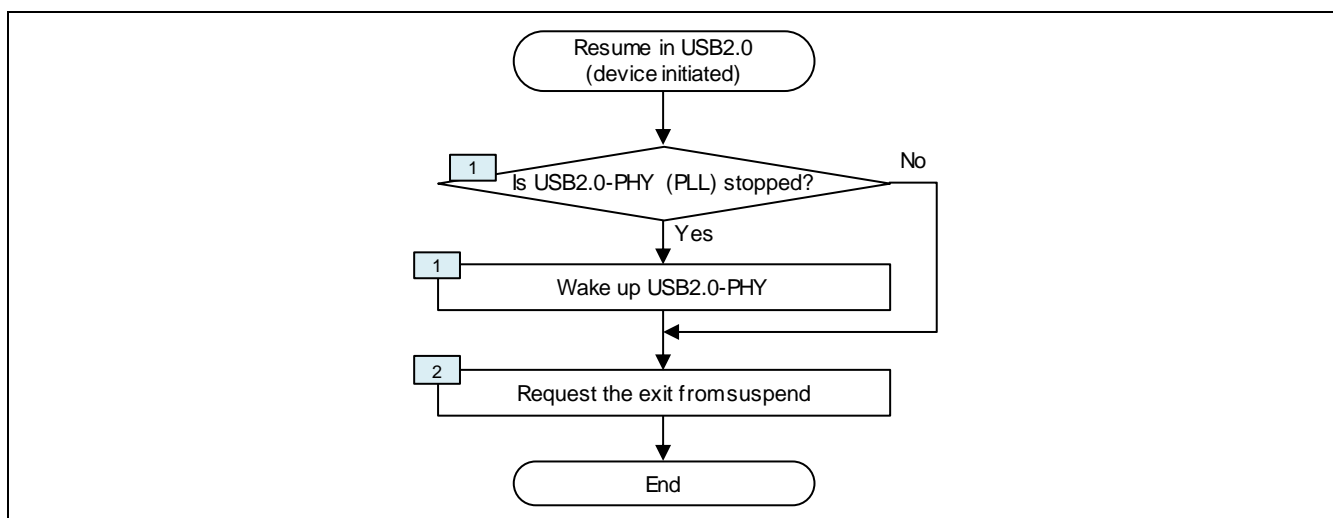


Figure 63.64 Resume in USB2.0 (Initiated by device) process flow

(1) Is the clock from USB2.0-PHY stopped?

If the clock from USB2.0-PHY is stopped following the flow in section 63.8.8.7, it is required to wake up USB2.0-PHY since the clock from USB2.0-PHY should be provided for the operations here. See section 63.8.8.10 for the process to wakeup of USB2.0-PHY.

(2) Request the exit from the suspended state

Request to exit from the suspended state by setting as follows.

- USB20_CON.B2_RSUM_IN = 1 (resume request)

If PLL in USB2.0-PHY is not stopped in suspended state, that is, if USB20_CON.B2_SUSPEND is not set to 1 in suspended state, USB_INT_STA_1.B2_RSUM_STA or USB_INT_STA_1.B2_L1RSUM_STA may be asserted when downstream port responds to the resume request and bus activity comes back.

But, if it is stopped, they are not asserted here because they are asserted when USB20_CON.B2_SUSPEND is cleared so they would have already been asserted at (1).

In either case host should respond to the resume request from the device and bus becomes active again, so the device is required only to wait for the resume of transfer.

63.8.8.10 USB2.0 PHY (PLL) Wakeup Process

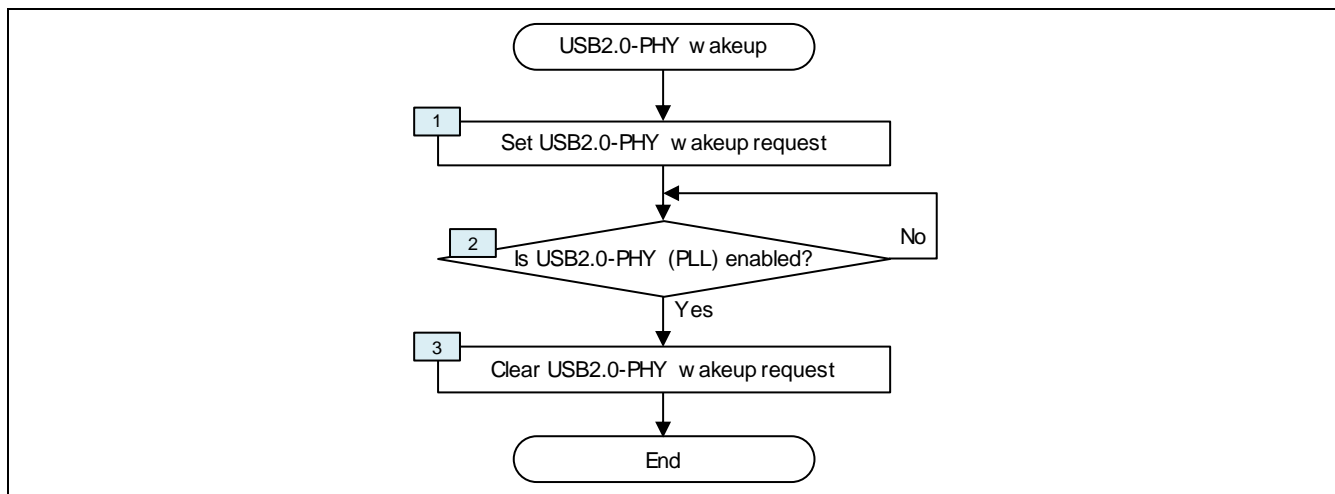


Figure 63.65 USB2.0-PHY (PLL) wakeup process flow

(1) Set USB2.0-PHY wakeup request

There are 2 ways to request USB2.0-PHY to wake up.
The first way is to clear USB20_CON.B2_SUSPEND.

[In case register is used]

- USB20_CON.B2_SUSPEND = 0 (Clearing the value set in section 63.8.8.7)

(2) Is USB2.0-PHY (PLL) enabled?

Responding to the wake up request at (1), USB2.0-PHY (PLL) becomes enabled.

At that time, USB_INT_STA_1.B2_RSUM_STA is generated and an interrupt due to it is caused if enabled.

- Interrupt due to USB_INT_STA_1.B2_RSUM_STA is generated

- Check that USB_INT_STA_1.B2_RSUM_STA==1

After the confirmation above, clear USB_INT_STA_1.B2_RSUM_STA.

- USB_INT_STA_1.B2_RSUM_STA = 1 (Clearing the event of USB2.0 PLL Wakeup)

If PERIPHERAL PORTION was in L1 state, USB_INT_STA_1.B2_L1RSUM_STA is asserted instead.

(3) Clear USB2.0-PHY wakeup request

After the resume of USB2.0-PHY is confirmed at (2), clear B2_PLL_WAKEIN input signal if used.

If USB20_CON.B2_SUSPEND is used to place USB2.0-PHY in suspended state, USB20_CON.B2_SUSPEND is automatically cleared.

[In case input signal is used]

- B2_PLL_WAKEIN = 0

63.8.8.11 Process of transition from U0 to U1/U2 in USB3.0 (Initiated by device)

With the settings as follows, PERIPHERAL PORTION automatically controls the transition from U0 to U1/U2 by sending LGO_U1/LGO_U2 request or by responding to LGO_U1/LGO_U2 request from host.

- PORTPMSC.U1_TIMEOUT[15:0] = U1 timeout value defined by user
- PORTPMSC.U2_TIMEOUT[15:0] = U2 timeout value in received U2 Inactivity Timeout LMP
- SSIFCMD.SSIF_URES[1:0] = B'00 (LAU responses to LGO_U1/LGO_U2 are allowed)
- SSIFCMD.SSIF_UREQ[1:0] = B'00 (sending LGO_U1/LGO_U2 by timeout is allowed)

PERIPHERAL PORTION controls the exit from U1/U2 state when it has any available data to be transmitted in buffer or when host requests wakeup as well. No additional operation by the software is required for the cases.

But there might be a case the software requests the transition to U1/U2 state at any point it wants. The process for such case is described here. But note that U1 timeout value can be defined by user, so consider it first to change the value of PORTPMSC.U1_TIMEOUT[15:0] in case LGO_U1 is required to be sent.

The process of Sleep (U1/U2) transition in USB3.0 when the device initiates at any timing is explained. This process flow is for usage when the device initiates LGO_U1/LGO_U2 at any time, and the timer which determine the duration for no data on the bus may be set in PORTPMSC.U1_TIMEOUT. The device can issue LGO_U1 after the timeout of timer.

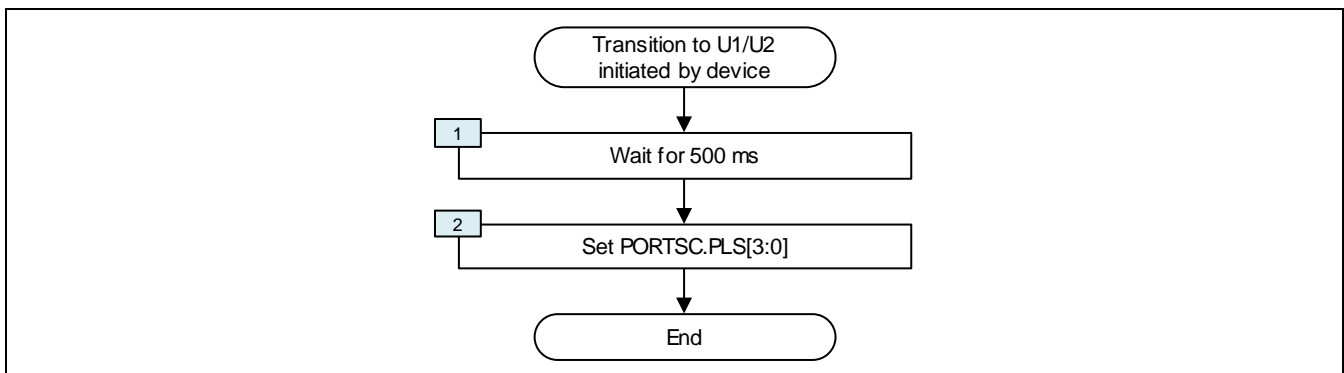


Figure 63.66 Transition to U1/U2 (Initiated by device) process flow

(1) Wait for 500ms

In C.3.2.2 and C3.2.1 of USB3.0 specification, it is defined that the link should be kept in U0 after device has sent ERDY until host sends a request in response to the ERDY or until tERDYtimeout (= 500ms) occurs.

Since PERIPHERAL PORTION sends ERDY automatically depending on the status of buffer, so it might be difficult to know when it has sent ERDY exactly.

If the software doesn't assure when the last ERDY has been sent, wait for 500ms from the point. If no response from host comes, it can request the transition to U1/U2 state. Note that PERIPHERAL PORTION might send ERDY again depending on the process of transfer if the response from host comes before the timeout.

If the software assures when the last ERDY has been sent and 500ms has passed, it can request the transition to U1/U2 immediately. The triggers to send ERDY for PERIPHERAL PORTION are :

- 1) IN packet is written to buffer
- 2) Pn_CON.Pn_RES[1:0] is changed to normal response (= B'01)
- 3) buffer for OUT endpoint was full once but becomes available
- 4) buffer for OUT endpoint had the last packet of transfer (short packet or packet with PP = 0) but it has been transferred to system memory.

If these triggers has not occurred for 500ms, it can be said that 500ms has passed at least from the last ERDY the device has sent.

(2) Set PORTSC.PLS[3:0]

The software can requests the transition to U1 or U2 by setting PORTSC.PLS[3:0]. It is also required to set SSIFCMD.SSIF_UDIR[1:0] to enable the function.

- SSIFCMD.SSIF_UDIR[1:0] = B'00 (If only one of LGO_U1 or LGO_U2 is requested, unused bit doesn't require to be enabled.)
- PORTSC.PLS[3:0] = H'1 (If the transition to U1 is requested)
H'2 (If the transition to U2 is requested)

With the operations above, PERIPHERAL PORTION sends LGO_U1 or LGO_U2 to request the transition to U1 or U2. If host accepts it, the link make a transition to the low power state.

When LTSSM in PERIPHERAL PORTION enters U1 or U2, USB_INT_STA_1.B3_LNKCNG_STA is asserted. An interrupt due to it is generated if enabled. PORTSC.PLS[3:0] shows the current state of LTSSM.

- Interrupt due to USB_INT_STA_1.B3_LNKCNG_STA is asserted
- Check that USB_INT_STA_1.B3_LNKCNG==1
- Check that PORTSC.PLS[3:0] shows the requested state (U1 or U2).

Note that PORTSC.PLS[3:0] might show another state if the link transits to U1/U2 but host requests the exit form the state soon.

If host doesn't accept the request, USB_INT_STA_1.B3_LNKCNG_STA is not asserted. In case the software can't detect the assertion of the bit even when it waits for a certain time, it should consider the possibility that host rejected the request. The software can send the same request again, but host might intend new transfer, so the software had better the state of PERIPHERAL PORTION in U0 for the moment.

After the confirmation above, write 1 to USB_INT_STA_1.B3_LNKCNG_STA to clear.

- USB_INT_STA_1.B3_LNKCNG_STA = 1 (Clearing the event that the link status is changed)

63.8.9 Device Requests

This section describes the operations required for USB device requests.

63.8.9.1 Clear Feature

This request is used to clear or disable a specific feature.

In case the recipient is device and the device works in SS, U1_ENABLE, U2_ENABLE and LTM_ENABLE are defined as feature selectors.

- (a) When Clear Feature(U1_ENABLE) comes, disable the initiation of U1 entry of PERIPHERAL PORTION as follows. Note that U1_ENABLE enables or disables only the initiation of U1 entry but the device is still allowed to accept LGO_U1 from downstream port and enters U1 state regardless of the feature.
 - SSIFCMD.SSIF_UDIR[0] = 1 (The initiation of LGO_U1 due to direct request is disabled.)
 - SSIFCMD.SSIF_UREQ[0] = 1 (The initiation of LGO_U1 due to timeout is disabled.)
- (b) When Clear Feature(U2_ENABLE) comes, disable the initiation of U2 entry of PERIPHERAL PORTION as follows. Note that U2_ENABLE enables or disables only the initiation of U2 entry but the device is still allowed to accept LGO_U2 from downstream port and enters U2 state regardless of the feature.
 - SSIFCMD.SSIF_UDIR[1] = 1 (The initiation of LGO_U2 due to direct request is disabled.)
 - SSIFCMD.SSIF_UREQ[1] = 1 (The initiation of LGO_U2 due to timeout is disabled.)
- (c) When Clear Feature(LTM_ENABLE) comes, disable the transmission of LTM (Latency Tolerance Message). Note that SS device must support U1 and U2 features, but LTM feature is optional. SS device shows whether it supports LTM feature or not on LTM capable bit in SuperSpeed USB Device Capabilities Descriptor.

In case the recipient is device and the device works in HS or FS, DEVICE_REMOTE_WAKEUP and TEST_MODE are defined as feature selectors.

- (d) When Clear Feature(DEVICE_REMOTE_WAKEUP) comes, disable the remote wakeup function of the device.
- (e) When TEST_MODE feature is enabled, the device is in test mode. But it is defined that the power cycle is used for the device to exit from test mode and Clear Feature(TEST_MODE) doesn't clear the feature.

For FUNCTION_SUSPEND, Clear Feature is not used to change the feature. Only Set Feature(FUNCTION_SUSPEND) is used to change the feature.

In case the recipient is endpoint, ENDPOINT_HALT is defined as feature selector. All of SS, HS and FS devices support the feature.

- (f) When Clear Feature (ENDPOINT_HALT) comes, it means the reset of Halt feature and the initialization of the endpoint.
 - At first, set Pn_CON.Pn_CLR and wait for Pn_CON.Pn_CLR to return to 0.
 - Pn_CON.Pn_CLR = 1 (Initialization of the endpoint (pipe) is required.)
 - Wait for Pn_CON.Pn_CLR==0
 - After Pn_CON.Pn_CLR==0, set enable bits below again.
 - Pn_CON.Pn_EN = 1 (The endpoint (pipe) is enabled)
 - Pn_CON.Pn_DATAIF_EN = 1 (EPC_D_Pn_DATAEN[30:1] is enabled for the endpoint)
 - Then, set Pn_CON.Pn_RES[1:0] = B'01 so that the endpoint responds in normal state.
 - Pn_CON.Pn_RES[1:0] = B'01 (Normal response)

It is not recommended that the endpoint 0 (PIPE0) supports function STALL for HS and FS device. It is also defined that SS device doesn't support the feature. If the device doesn't support Halt feature for the endpoint 0, it

executes the control transfer of Clear Feature(ENDPOINT_HALT) for the endpoint 0 normally but can ignore the request.

It is noted in section 63.8.9.5 how each feature is controlled as the status of the device.

63.8.9.2 Get Configuration

This request returns the current device configuration value.

In case this request is received, return the current configuration value. It has been selected with Set Configuration. If not, return the initial value (= zero).

63.8.9.3 Get Descriptor

This request returns the specified descriptor if the descriptor exists.

The length of transfer is specified in wLength of setup data. In case the size of the descriptor equals to or is smaller than the value in wLength, transmit all of the descriptor from the beginning to the end. In case the size of the descriptor is larger than the value in wLength, transmit a part of the descriptor from the beginning to the length specified in wLength. Each packet transporting the descriptor should equal to or smaller than the max packet size defined for each speed mode (512byte for SS, 64byte for HS and 8/16/32/64byte for FS). If the size of the last packet equals to the max packet size defined, zero length packet should follow it.

63.8.9.4 Get Interface

This request returns the selected alternate setting of the specified interface.

In case this request is received, return the current alternate setting of the interface. It has been selected with Set Interface. If not, return the initial value (= zero).

63.8.9.5 Get Status

This request returns status for the specified recipient.

In case the recipient is device, the information returned for this request as SuperSpeed device is shown in Figure 63.67.

| | | | | | | | |
|-----------------------------|-----|-----|---------------|-----------|-----------|------------------|------------------|
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| Reserved (Reset to Zero) | | | LTM Enable | U2 Enable | U1 Enable | Remote Wakeup | Self- Powered |
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |
| Reserved (Reset to Zero) | | | | | | | |

Figure 63.67 Information as SS device in case the recipient is device

D0: This bit shows whether the device is self-powered or not.

D1: This bit shows whether the device supports Remote Wakeup or not.

SuperSpeed device is defined to set this bit to 0 and use Function Remote Wakeup instead.

D2: This bit shows whether the device is currently enabled to initiate U1 entry or not. If this bit is 0, the device is disabled to initiate U1 entry. Regardless of whether this bit is set or not, the device can accept the request for U1 entry from downstream port and enter U1 state.

This bit is set to 1 on the reception of Set Feature(U1_ENABLE) and set to 0 on the reception of Clear Feature(U1_ENABLE). This bit is reset to 0 when the device is reset.

D3: This bit shows whether the device is currently enabled to initiate U2 entry or not. If this bit is 0, the device is disabled to initiate U2 entry. Regardless of whether this bit is set or not, the device can accept the request for U2 entry from downstream port and enter U2 state.

This bit is set to 1 on the reception of Set Feature(U2_ENABLE) and set to 0 on the reception of Clear Feature(U2_ENABLE). This bit is reset to 0 when the device is reset.

D4: This bit shows whether the device is currently enabled to send LTM (Latency Tolerance Message) or not. If this bit is 0, the device is disabled to send LTM.

This bit is set to 1 on the reception of Set Feature(LTM_ENABLE) and set to 0 on the reception of Clear Feature(LTM_ENABLE). This bit is reset to 0 when the device is reset.

In case the recipient is device, the information returned for this request as HighSpeed or FullSpeed device is shown in Figure 63.68.

| | | | | | | | |
|--------------------------|-----|-----|-----|-----|-----|---------------|--------------|
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| Reserved (Reset to Zero) | | | | | | Remote Wakeup | Self-Powered |
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |
| Reserved (Reset to Zero) | | | | | | | |

Figure 63.68 Information as HS/FS device in case the recipient is device

D0: This bit shows whether the device is self-powered or not.

D1: This bit shows whether the device supports Remote Wakeup or not.

The default value is 0. The value is modified on the reception of Set Feature(DEVICE_REMOTE_WAKEUP) or Clear Feature(DEVICE_REMOT_WAKEUP). When Set Feature(DEVICE_REMOTE_WAKEUP) is received, the status of this bit is changed to 1. When Clear Feature(DEVIE_REMOT_WAKEUP) is received, the status of this bit is changed to 0.

When this feature is enabled, device can request the remote wakeup in suspended state.

This bit is reset to 0 when the device is reset.

In case the recipient is interface, the information returned for this request as SuperSpeed device is shown in Figure 63.69.

| | | | | | | | |
|--------------------------|-----|-----|-----|-----|-----|------------------------|--------------------------------|
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| Reserved (Reset to Zero) | | | | | | Function Remote Wakeup | Function Remote Wakeup Capable |
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |
| Reserved (Reset to Zero) | | | | | | | |

Figure 63.69 Information as SS device in case the recipient is interface

D0: This bit shows whether the interface supports Function Remote Wakeup Capable or not.

Note that Function Remote Wakeup function is used instead of Remote Wakeup function in SuperSpeed device. If the device doesn't support Function Remote Wakeup, it can't request the remote wakeup in suspended state (U3).

D1: This bit shows whether the function is currently enabled to request the remote wakeup or not.

The definition of function depends on the specification of device.

The default value is 0. The value is modified on the reception of Set Feature(FUNCTION_SUSPEND). When Set Feature(FUNCTION_SUSPEND) is received and bit 1 in the upper byte of wIndex is 1, Function Remote Wake is enabled and the status of this bit is changed to 1. When Set Feature(FUNCTION_SUSPEND) is received and bit1 in the upper byte of wIndex is 0, Function Remote Wake is disabled and the status of this bit is changed to 0.

When this feature is enabled, device can request the remote wakeup for the suspended function.

This bit is reset to 0 when the device is reset.

In case the recipient is interface, the information returned for this request as HighSpeed or FullSpeed device is shown in Figure 63.70. All fields are reserved.

| | | | | | | | |
|--------------------------|-----|-----|-----|-----|-----|----|----|
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| Reserved (Reset to Zero) | | | | | | | |
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |
| Reserved (Reset to Zero) | | | | | | | |

Figure 63.70 Information as HS/FS device in case the recipient is interface

In case the recipient is endpoint, the information returned for this request as SuperSpeed, HighSpeed or FullSpeed device is shown in Figure 63.71.

| | | | | | | | |
|--------------------------|-----|-----|-----|-----|-----|----|------|
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| Reserved (Reset to Zero) | | | | | | | Halt |
| D15 | D14 | D13 | D12 | D11 | D10 | D9 | D8 |
| Reserved (Reset to Zero) | | | | | | | |

Figure 63.71 Information as SS/HS/FS device in case the recipient is endpoint

D0: This bit shows the current status of Halt feature of the endpoint. If the endpoint is currently halted, the Halt feature is set to 1.

The Halt feature is required to be implemented for all interrupt and bulk endpoints.

The Halt feature might be set due to the internal status of device (when unsupported request comes, for example), or it might be set by Set Feature(ENDPOINT_HALT).

During the feature is set to 1, the endpoint should return STALL response to any request. To return STALL response, set Pn_CON.Pn_RES[1:0] = B'10.

- Pn_CON.Pn_RES[1:0] = B'10 (STALL response)

The Halt feature is cleared on the reception of ClearFeature(ENDPOINT_HALT), SetConfiguration and SetInterface. Note that the sequence number in SS and data toggle in HS/FS is initialized on those requests. The feature might be cleared due to the internal status of device (when the cause of error is removed, for example).

When the Halt feature is reset, the endpoint changed its status not to return STALL response. Set Pn_CON.Pn_RES[1:0] = B'01 to request the endpoint to respond in the normal mode.

- Pn_CON.Pn_RES[1:0] = B'01 (Normal response)

SS device doesn't support function STALL on control endpoint.

63.8.9.6 Set Address

This request sets the device address for all future device accesses. The address is shown in wValue field of setup data. In case this request is received, read the address in wValue field of setup data. But the device can't update its address to the value given on Set Address until it completes the control transfer of Set Address, since it is defined in USB specification that stages after the initial setup packet assume the same device address as the setup packet. Write the address given on Set Address to the field below before completing the status stage of the control transfer of Set Address. (That is, write the address to the field below, then P0_CON.P0_ST_RES[1:0] to normal response.) The hardware of PERIPHERAL PORTION updates the field to new address after it completes the status stage of the control transfer of Set Address.

— USB_COM_CON.DEV_ADDR[6:0] = the value in wValue field of setup packet.

Set Address 0 has the special meaning to initialize the settings in the device. See Table 9-9 in USB3.0 specification.

63.8.9.7 Set Configuration

This request sets the device configuration. The configuration value selected is shown in lower byte of wValue field of setup data.

The configuration value shall be zero or the value shown in the configuration descriptor.

- (1) In case the configuration value matches the value shown in the configuration descriptor, do the operations as follows before the status stage of the control transfer is completed.
 - Configure PERIPHERAL PORTION to follow the configuration selected.
 - USB_COM_CON.CONF = 1 (PERIPHERAL PORTION is in configured state)

Remember the configuration value since Get Configuration might be requested later.

Then, complete the status stage of the control transfer by setting P0_CON.P0_ST_RES[1:0] = B'01 (normal response).

- (2) In case the configuration value is zero, it means that the device is requested to go back to address state. Do the operation as follows before the status stage of the control transfer is completed.
 - USB_COM_CON.CONF = 0 (PERIPHERAL PORTION is not in configured state)

The buffer of each PIPE should be initialized as well. Do the operation noted in (f) of section 63.8.9.1 for PIPES those have been used in configured state.

Note that the clocks to Data-IN RAMs or Data-OUT RAM is suspended with the setting above. (They are suspended in unconfigured state.) In that case, the clocks to Register RAM and EP0 RAM are still running. Remember the configuration value since Get Configuration might be requested later.

Then, complete the status stage of the control transfer by setting P0_CON.P0_ST_RES[1:0] = B'01 (normal response).

63.8.9.8 Set Descriptor

This request is optional and may be used to update existing descriptors or new descriptors may be added. If this request is not defined in the specification of device or the class specification which the device supports, this request is not required to be implemented. When SetDescriptor comes, update the specified descriptor.

63.8.9.9 Set Feature

This request is used to set or enable a specific feature.

In case the recipient is device and the device works in SS, U1_ENABLE, U2_ENABLE and LTM_ENABLE are defined as feature selectors.

- (a) When Set Feature(U1_ENABLE) comes, enable the initiation of U1 entry of PERIPHERAL PORTION as follows. Note that U1_ENABLE enables or disables only the initiation of U1 entry but the device is still allowed to accept LGO_U1 from downstream port and enters U1 state regardless of the feature.
 - SSIFCMD.SSIF_UDIR[0] = 0 (The initiation of LGO_U1 due to direct request is enabled.)
 - SSIFCMD.SSIF_UREQ[0] = 0 (The initiation of LGO_U1 due to timeout is enabled.)
 - PORTPMSC.U1_TIMEOUT = U1 timeout value defined by user
- (b) When Set Feature(U2_ENABLE) comes, enable the initiation of U2 entry of PERIPHERAL PORTION as follows. Note that U2_ENABLE enables or disables only the initiation of U2 entry but the device is still allowed to accept LGO_U2 from downstream port and enters U2 state regardless of the feature.
 - SSIFCMD.SSIF_UDIR[1] = 0 (The initiation of LGO_U2 due to direct request is enabled.)
 - SSIFCMD.SSIF_UREQ[1] = 0 (The initiation of LGO_U2 due to timeout is enabled.)
 - PORTPMSC.U2_TIMEOUT = U2 timeout value in received U2 Inactivity Timeout LMP
- (c) When Set Feature(LTM_ENABLE) comes, enable the transmission of LTM (Latency Tolerance Message). Note that SS device must support U1 and U2 features, but LTM feature is optional. SS device shows whether it supports LTM feature or not on LTM capable bit in SuperSpeed USB Device Capabilities Descriptor.

In case the recipient is device and the device works in HS or FS, DEVICE_REMOTE_WAKEUP and TEST_MODE are defined as feature selectors.

- (d) When Clear Feature(DEVICE_REMOTE_WAKEUP) comes, enable the remote wakeup function of the device. When requesting the remote wakeup, see the flow in section 63.8.8.9.
- (e) When TEST_MODE feature is enabled, the device is in test mode. The upper byte of wIndex of setup data specifies test mode selector and it is defined in Table 9-7 of USB2.0 specification. PERIPHERAL PORTION has the function for TEST_MODE with the setting of USB20_CON.B2_TSTMOD [2:0] and USB20_CON.B2_TSTMOD_EN. When Set Feature (TEST_MODE) comes, set USB20_CON.B2_TSTMOD[2:0] and USB20_CON.B2_TSTMOD_EN according to the test mode selector specified.

In case the recipient is interface and the device works in SS, FUNCTION_SUSPEND is defined as feature selector.

- (f) The function within the device shows whether it supports Function Remote Wakeup function or not in Function Remote Wakeup Capable bit in Figure 63.69. When the function supports Function Remote Wake and Set Feature(FUNCTION_SUSPEND) is received and the bit 1 in the upper byte of wIndex is 1, enable Function Remote Wake. When the function supports Function Remote Wake and Set Feature(FUNCTION_SUSPEND) is received and the bit 1 in the upper byte of wIndex is 0, disable Function Remote Wake. To request the function remote wakeup, See section 63.8.8.5.

In case the recipient is endpoint, ENDPOINT_HALT is defined as feature selector. All of SS, HS and FS devices support the feature.

- (g) When Set Feature(ENDPOINT_HALT) comes, set Pn_CON.Pn_RES[1:0] = B'10 to return STALL response from the endpoint.
 - Pn_CON.Pn_RES[1:0] = B'10 (STALL response)

Note that the state machine of Stream bulk transits to Disabled state by receiving Set Feature(ENDPOINT_HALT).

If the endpoint works as the Stream bulk one, disable all transfers on the reception of Set Feature(ENDPOINT_HALT).

It is not recommended that the endpoint 0 supports function STALL for HS and FS device. It is also defined that SS device doesn't support the feature. If the device doesn't support Halt feature for the endpoint 0, it executes the control transfer of Set Feature(ENDPOINT_HALT) for the endpoint 0 normally but can ignore the request.

It is noted in section 63.8.9.5 how each feature is controlled as the status of the device.

63.8.9.10 Set Interface

This request allows host to select an alternate setting of the specified interface.

In case this request is received, configure PERIPHERAL PORTION to follow the configuration of interface selected.

- Configure PERIPHERAL PORTION to follow the configuration of interface selected

As Set Interface means the initialization of selected interface as well, it is recommended to initialize the PIPEs related to the interface using Pn_CON.Pn_CLR. The operations as follows are required for all PIPEs related to the interface.

- Pn_CON.Pn_CLR = 1 (Initialization of PIPE)
- Wait for Pn_CON.Pn_CLR returns to 0
- Pn_CON.Pn_EN = 1 (Enabling the PIPE again)
- Pn_CON.Pn_DATAIF_EN = 1 (Enabling data interface for the PIPE again.)

Remember the alternate setting of the interface since Get Interface might be requested later.

63.8.9.11 Set Isochronous Delay

This request informs the device of the delay from the time host transmits a packet to the time it is received by the device.

As PERIPHERAL PORTION doesn't support Isochronous transfer, so there is no function to use the value received on this request.

From the viewpoint of implementation the device can return STALL response in order to show it doesn't support this request, or can ignore the request although it responds correctly to the control transfer of this request.

63.8.9.12 Set SEL

This request is valid when the device works in SS and sets both the U1 and U2 System Exit Latency and the U1 or U2 exit latency for all the links between a device and a root port on the host.

When Set SEL comes, update BELT (Best Effort Latency Tolerance) value referring the information provided on this request, if required.

63.8.9.13 Synch Frame

This request is used to set and then report an endpoint's synchronization frame.

As PERIPHERAL PORTION doesn't support Isochronous transfer, so there is no function to use the value received on this request.

From the viewpoint of implementation the device can return STALL response in order to show it doesn't support this request, or can ignore the request although it responds correctly to the control transfer of this request.

63.8.10 BOT Control

Bulk Only Transport (BOT) is defined as one of transfer methods in mass storage class. An example of BOT operation is described here.

63.8.10.1 Overall

BOT consists of two bulk endpoints as its name suggests.

- Bulk OUT PIPE (for the reception of CBW or OUT data)
- Bulk IN PIPE (for the transmission of CSW or IN data)

In BOT protocol, Command Block Wrapper (CBW) is sent to request an operation in advance of IN or OUT data transfers, and Command Status Wrapper (CSW) is sent to notify the result of the operation after the IN or OUT data transfers. There are some commands which don't require the data transfers. For such command, only CBW and CSW are exchanged.

63.8.10.2 BOT assistance mode

PERIPHERAL PORTION has BOT assistance mode. The mode is available when Pn_MOD.Pn_BOT is set to 1 for OUT PIPE in BOT.

In BOT assistance mode, PERIPHERAL PORTION notifies the reception of CBW by asserting Pn_INT_STA.Pn_CBW_STA. If an interrupt due to the bit is enabled, the interrupt is generated.

PERIPHERAL PORTION interprets the received packet as CBW when both of two conditions below are satisfied for the packet.

- The length of the packet is 31bytes.
- The first 4bytes of the packet (dCBWSignature) equals to 43425355h.

In this mode, Pn_CON.Pn_DATAIF_EN of the OUT PIPE that receives CBW is disabled (reset to 0) every time CBW (the packet which satisfies two conditions above) is received. As the result, CBW is not transferred through Data Interface or on AXI DMA transfer. Pn_CON.Pn_DATAIF_EN of IN PIPE is not changed.

CBW can be transferred through Data Interface or on AXI DMA transfer if Pn_CON.Pn_DATAIF_EN is set to 1 again, but it is recommended to read CBW through Pn_READ register because the software should read and decode it for BOT operation.

As CBW is a short packet in bulk transfer, the OUT PIPE in BOT enters flow control after the reception of CBW if it works as USB3.0 device. When CBW has been read through Pn_READ register (or when it has been transferred through Data Interface or on AXI DMA transfer), PERIPHERAL PORTION returns ERDY from the OUT PIPE automatically. But if OUT data transfer is requested, make sure to set Pn_CON.Pn_DATAIF_EN to 1 after CBW has been read or transferred.

If BOT assistance mode is not used (Pn_MOD.Pn_BOT = 0) for BOT transfer, the OUT PIPE in BOT doesn't distinguish CBW and CBW is transferred as one of short packets on AXI DMA transfer. In that case Pn_CON.Pn_DATAIF_EN is not disabled and Pn_INT_STA.Pn_CBW_STA is not asserted on the reception of CBW. (PERIPHERAL PORTION doesn't know that the packet is CBW.)

The software is required to find CBW in system memory (where the received packet is transferred to) when a short packet is received.

63.8.10.3 BOT Process

It is assumed that BOT assistance mode in section 63.8.10.2 and the flow of BOT process with the mode is described here.

Make sure to set Pn_MOD.Pn_BOT = 1.

As noted above, one bulk OUT PIPE and one bulk IN PIPE are required for BOT. For the settings of bulk OUT PIPE, see section 63.8.5.2. For the settings of bulk IN PIPE, see section 63.8.5.1.

As the preparation of BOT, enable an interrupt due to Pn_INT_STA.Pn_CBW_STA of the OUT PIPE. That is, set Pn_INT_ENA.Pn_CBW_ENA of bulk OUT PIPE to 1.

- Pn_INT_ENA.Pn_CBW_ENA= 1 (Interrupt due to Pn_INT_STA.Pn_CBW_STA is enabled)

(1) Is CBW received?

When CBW is received, Pn_INT_STA.Pn_CBW_STA is asserted and an interrupt due to it is generated.

- Interrupt due to Pn_INT_STA.Pn_CBW_STA is asserted

- Check that Pn_INT_STA.Pn_CBW_STA == 1

After the confirmation above, clear Pn_INT_STA.Pn_CBW_STA.

- Pn_INT_STA.Pn_CBW_STA = 1 (Clearing the event of the reception of CBW)

(2) Read CBW

Read CBW through Pn_READ register.

(3) Decode CBW

Decode the content of CBW which is read at (2).

If any operation other than data transfer is requested, do it here.

(4) Which direction of transfer is requested?

If OUT transfer is requested, proceed to (5).

If IN transfer is requested, proceed to (6).

If No data transfer is requested, proceed to (7).

(5) Read OUT data

It is assumed here that OUT data received is transferred through Data Interface or on AXI DMA transfer. Set Pn_CON.Pn_DATAIF_EN = 1 since it is reset to 0 at the reception of CBW.

For bulk OUT transfer, see section 63.8.5.2. If two or more packets are received, repeat the flow shown there. Proceed to (7).

(6) Write IN data

It is assumed here that IN data to be transmitted is transferred through Data Interface or on AXI DMA transfer.

Pn_CON.Pn_DATAIF_EN of IN PIPE is not changed on the reception of CBW.

For bulk IN transfer, see section 63.8.5.1. If two or more packets are to be transmitted, repeat the flow shown there. Proceed to (7).

(7) Send CSW

It is assumed here that the software prepares CSW and writes it through Pn_WRITE register.

Before writing CSW to the buffer of the IN PIPE, confirm that the buffer has available space.

- Check that Pn_STA.Pn_BUFSTS== 1 for bulk IN PIPE in order to confirm that the buffer for the PIPE is available.

If the buffer has available space for CSW, write CSW through Pn_WRITE register. Since the size of CSW is 13 bytes, the access to Pn_WRITE register is required 4 times. The last write word has only 1 valid byte.

- Write CSW to Pn_WRITE register. 4 write accesses are required for 13 bytes of CSW.

After CSW has been written to Pn_WRITE register, send CSW as follows. As only 32-bit access is allowed to the register, both fields should be set in one word access.

-Pn_CON.Pn_SEND = 1

-Pn_CON.Pn_BYTE_EN[1:0] = B'01 (The last word has only 1 valid byte.)

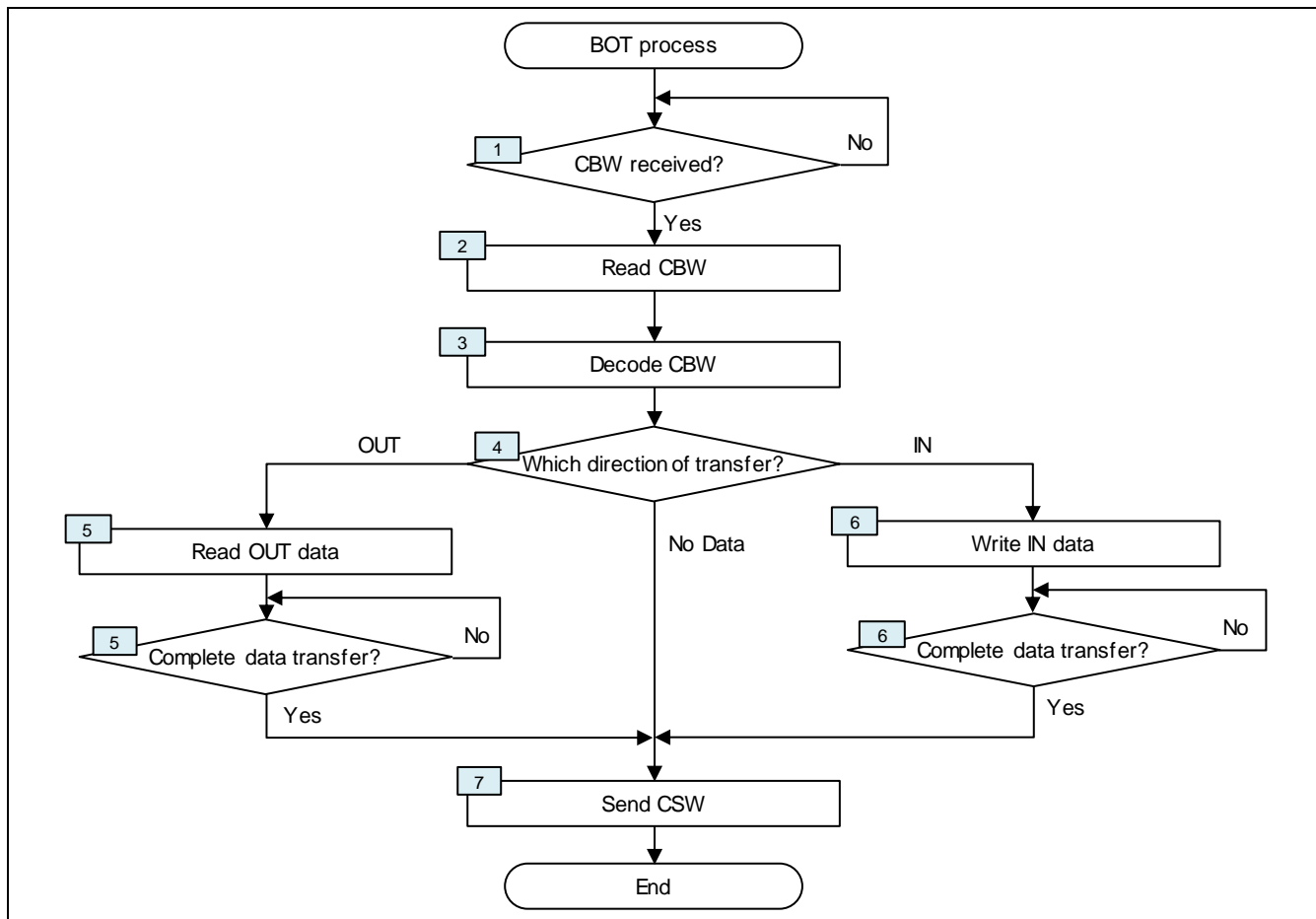


Figure 63.72 BOT process flow

63.9 Instructions and Directions for Use

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Instructions and directions for using Peripheral Core is described in this section.

63.9.1 Method of Specifying RAM Mapping

Peripheral core has configurable EPC data buffer RAM to be allocated for each PIPE.

Table 63.25 Number of SRAMs

| | Usage | Bit Width | Number of words | Quantity | Total size per channel |
|----|-----------------------------|-----------|-----------------|----------|------------------------|
| 1. | For data buffer EP0 (PIPE0) | 64 bit | 64 | 1 | 512byte |
| 2. | For data buffer IN PIPE | 64 bit | 2048 | 4 | 64kbyte |
| 3. | For data buffer OUT PIPE | 64 bit | 2048 | 4 | 64kbyte |

Setting Example)

Number of PIPEs: 2 PIPEs for each of IN/OUT. (Total: 4 PIPEs)

Pn_RAMAREA: (PIPE1/OUT) 4k Byte (4 Block)

(PIPE2/IN) 4k Byte (4 Block)

(PIPE3/OUT) 2k Byte (2 Block)

(PIPE4/IN) 2k Byte (2 Block)

Quantity of SRAM to be connected: For IN PIPE = 1, For OUT PIPE = 1

⇒ (4k + 2k) for each of IN/OUT / 8 Byte = 768 word

63.9.2 Example of Setting

The buffer space for each PIPE in EPC is set by 63.2.2.55 PIPE_n Mode Setting Register (Pn_MOD). The buffer size for each PIPE is specified in Pn_RAMAREA[2:0], and it is specified 1Block (1KByte) size per one packet in USB transaction regardless Max_Packet_Size at Pn_MPKT. Thus regarding to Pn_BASEAD[13:0] which specify Base_Address, the lower 7 bits need to be set to all 0 and specified its size as 1KB unit.

Following notice is the key to use RAM field efficiently.

- Set the PIPE which Max_Packet_Size is 1Kbyte at the top of each RAM (IN/OUT separately)
- Set the PIPE which Max_Packet_Size is less than 1Kbyte at the end of RAM as much as possible
- Note that the space other than 1Block occupied 1 packet size is required 1Kbyte for each, when the PIPE which Max_Packet_Size is less than 1Kbyte and Pn_RAMAREA is set to the value other than B'00 (1Block).

Example)

Table 63.26 Specifications of PIPEs

| | Endpoint Index | Direction of Transfer | Endpoint Type | Max Burst Size | Max Packet Size |
|-------|----------------|-----------------------|---------------|----------------|-----------------|
| PIPE1 | Endpoint 1 | IN | BULK | 4 | 1024 Byte |
| PIPE2 | Endpoint 2 | OUT | BULK | 4 | 1024 Byte |
| PIPE3 | Endpoint 3 | IN | BULK | 2 | 1024 Byte |
| PIPE4 | Endpoint 4 | OUT | BULK | 2 | 1024 Byte |
| PIPE5 | Endpoint 5 | IN | INTERRUPT | 1 | 512 Byte |

Table 63.27 SRAM interfaces and sizes of SRAMs

| RAM-IF Index/ IN or OUT | RAM Size | |
|----------------------------|-----------------|--------------------|
| 0/IN | 64bit × 576word | (4KByte + 512Byte) |
| 1/IN | 64bit × 264word | (2KByte) |
| 2/IN | None | — |
| 3/IN | None | — |
| 0/OUT | 64bit × 512word | (4KByte) |
| 1/OUT | 64bit × 256word | (2KByte) |
| 2/OUT | None | — |
| 3/OUT | None | — |

Table 63.28 Settings for SRAMs

| | Pn_EPNUM | Pn_DIR | Pn_TYPE | Pn_MPKT | Pn_RAMAREA | Pn_RAMIF | Pn_BASEAD |
|--------------|----------|--------|---------|---------|------------|----------|-----------|
| PIPE1 | B'001 | B'1 | B'10 | H'400 | B'010 | B'00 | H'0000 |
| PIPE2 | B'010 | B'0 | B'10 | H'400 | B'010 | B'00 | H'0000 |
| PIPE3 | B'011 | B'1 | B'10 | H'400 | B'001 | B'01 | H'0000 |
| PIPE4 | B'100 | B'0 | B'10 | H'400 | B'001 | B'01 | H'0000 |
| PIPE5 | B'101 | B'1 | B'11 | H'200 | B'000 | B'00 | H'0200 |

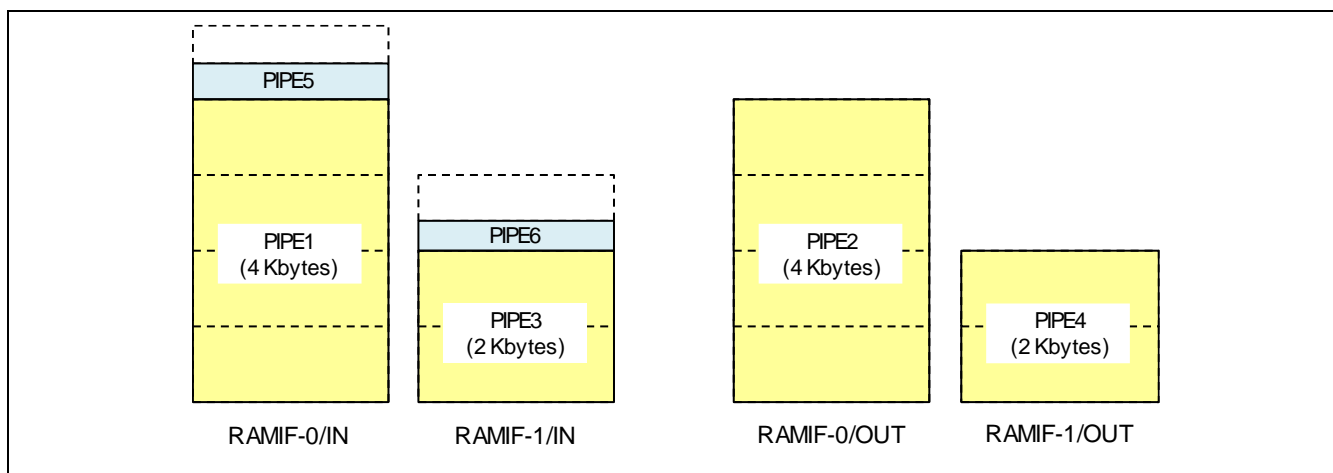


Figure 63.73 RAM Mapping Diagram for Example

63.9.3 Register Access

P0_READ, P0_WRITE, Pn_READ and Pn_WRITE registers are used to read or write a packet to each PIPE through these registers.

But as noted in the subsection of each register, it is prohibited to access to other Px_yyy (x = 0 to 15, yyy = register name) registers before whole of a packet has been read or written through the register which is currently used. Px_yyy registers. The packet might not be read or written correctly otherwise.

It is also prohibited to change the targeted PIPE of Pn_yyy registers on PIPE_COM.PIPE_NUM[4:0] before read or write of a packet has been completed to the PIPE currently selected.

When P0_WRITE or Pn_WRITE register is used, it is required to send the packet which has been written by setting P0_CON.P0_SEND or Pn_CON.Pn_SEND before the access to other Px_yyy registers.

Make sure to complete read or write of a packet before the access to other Px_yyy registers, once it is started.

If Px_yyy register is read or written in an interrupt routine, it might violate the notation without awareness. To avoid the case, mask the interrupt related to the routine during read or write of a packet, or never access to Px_yyy registers in an interrupt service.

The lists of registers prohibited to access during P0_READ or Pn_READ register is used are as follows.

| | |
|--|--|
| Prohibited to access during P0_READ register is used | P0_WRITE, Pn_READ, Pn_WRITE, Pn_STA, Pn_LNG, Pn_RSVPKT |
| Prohibited to access during Pn_READ register is used | P0_READ, P0_WRITE, P0_STA, P0_LNG, Pn_WRITE |

The lists of registers prohibited to access during P0_WRITE or Pn_WRITE register is used are as follows.

| | |
|--|---|
| Prohibited Registers for accessing in case of P0_WRITE | P0_READ, Pn_READ, Pn_WRITE, Pn_STA, Pn_LNG, Pn_RSVPKT |
| Prohibited Registers for accessing in case of Pn_WRITE | P0_READ, P0_WRITE, P0_STA, P0_LNG, Pn_READ |

64. RCLK Watchdog Timer (RWDT)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

64.1 Overview

This LSI includes the RCLK watchdog timer (RWDT).

This LSI can be reset by the overflow of the counter when the value of the counter has not been updated because of a system runaway.

The RWDT is a single-channel timer that uses the RCLK as an input and can be used as a watchdog timer. RCLK is a clock, which is generated by the clock pulse generator (CPG).

Frequency of RCLK:

- RZ/G2H: Select from 32.89kHz and 32.55kHz
- RZ/G2M V1.3: Select from 32.89kHz and 32.55kHz
- RZ/G2M V3.0: Select from 32.89kHz and 32.55kHz
- RZ/G2N: Select from 32.89kHz and 32.55kHz .
- RZ/G2E: Select from 31.3 kHz and 32.8 kHz.

64.1.1 Features

- One channel is provided.
- Can be used as a watchdog timer. Reset is generated when the counter overflows.
- A counter input clock can be chosen from:
Clocks (RCLK/1 to RCLK/ (128 × 60 × 30)) that are obtained by dividing the RCLK.

64.1.2 Block Diagram

Figure 64.1 shows block diagrams of the RWDT.

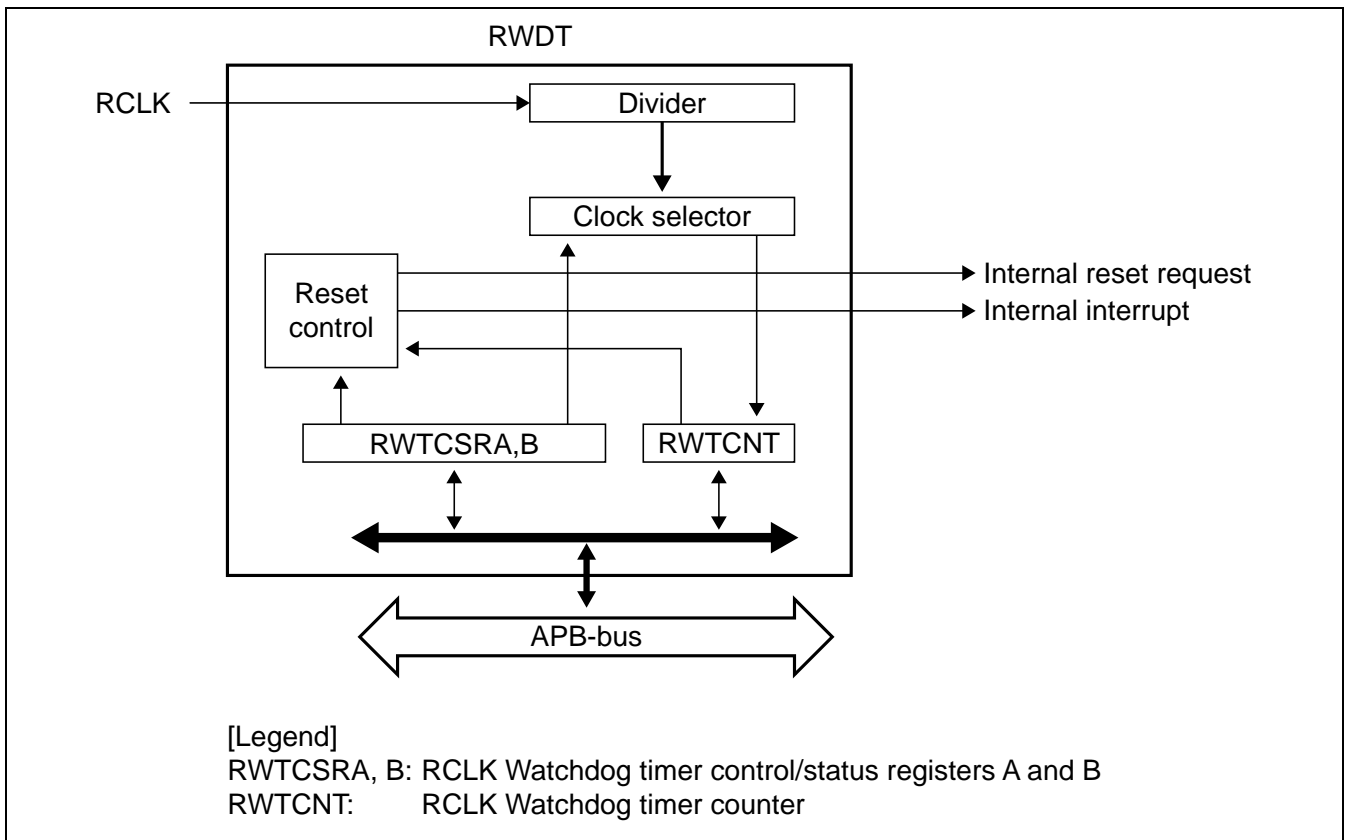


Figure 64.1 Block Diagram of RWDT

64.1.3 Register Configuration

Table 64.1 shows the RWDT register configuration. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined. Table 64.2 shows the register state in each processing mode.

Table 64.1 Register Configuration of RWDT

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|--------------|-------------------|-------------|---------------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| RCLK watchdog timer counter | RWTCNT | R/W | H'E602_0000 | 16/32* ¹ | √ | √ | √ | √ |
| RCLK watchdog timer control/status register A | RWTCSRA | R/W* ³ | H'E602_0004 | 8/32* ² | √ | √ | √ | √ |
| RCLK watchdog timer control/status register B | RWTCSR B | R/W | H'E602_0008 | 8/32* ² | √ | √ | √ | √ |

- Notes:
1. Write is performed in 32 bits and read in 16 bits. Operation cannot be guaranteed if the register is not written as a longword (32bits) and not read as a word (16bits).
 2. Write is performed in 32 bits and read in 8 bits. Operation cannot be guaranteed if the register is not written as a longword (32bits) and not read as a byte (8bits).
 3. Except for bit [4] that is R/WC0 register.

Table 64.2 Register State of RWDT in Each Processing Mode

| Register Abbreviation | | Reset Caused by Other than Overflow | Reset Caused by Overflow* |
|-----------------------|-----------------|-------------------------------------|---------------------------|
| Register | Bit | | |
| RWTCNT | All | Initialized | Initialized |
| RWTCSRA | WOVF | Initialized | Retained |
| | Other than WOVF | Initialized | Initialized |
| RWTCSR B | All | Initialized | Initialized |

Note: * Section 65. System Watchdog Timer (SWDT) and RWDT overflow reset.

64.2 Register Description

[Legend for Register Description]

Initial value: Register value after a reset.

—: Undefined value

R/W: Bit or field is readable and writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R: Read-only. The write value should always be initial value.

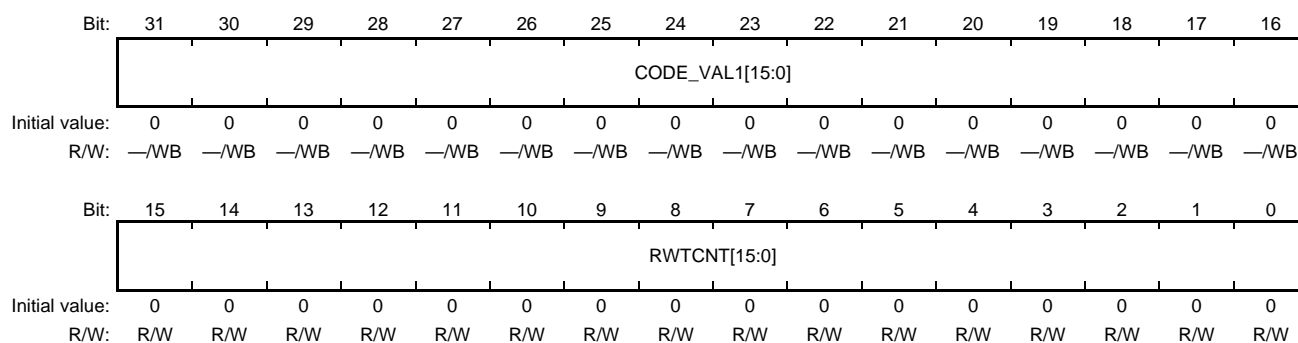
—/WB: Write-only. The read value is initial value.

64.2.1 RCLK Watchdog Timer Counter (RWCNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

RWCNT is a 16-bit readable/writable register that increments on the selected clock. When an overflow occurs, it generates a Reset. The RWCNT counter is initialized to H'0000 by a power-on reset and RWDT software reset.

Use a long word access to write to the RWCNT counter, with H'5A5A in the upper byte. Use a word access to read RWCNT.



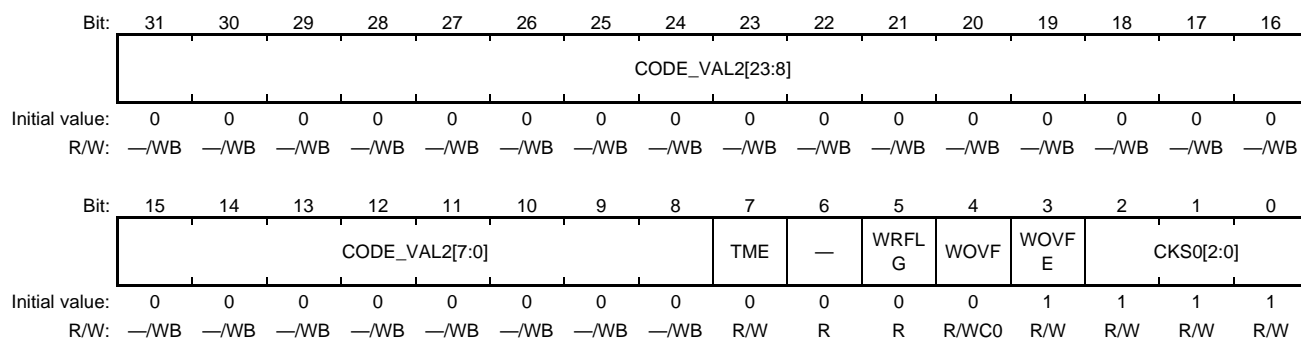
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|------|---|
| 31 to 16 | CODE_VAL1 [15:0] | H'0000 | —/WB | Code value 1 When writing this register, write H'5A5A to this bit field. When reading, these bits are always read as 0. |
| 15 to 0 | RWCNT[15:0] | H'0000 | R/W | Timer Counter Bits |

64.2.2 RCLK Watchdog Timer Control/Status Register A (RWTCSPA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

RWTCSPA is an 8-bit readable/writable register composed of bits to select the clock used for the count, overflow flag, and enable bit.

Use a long word access to write to RWTCSPA, with H'A5_A5A5 in the upper byte. Use a byte access to read RWTCSPA.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------|---------------|-------|--|
| 31 to 8 | CODE_VAL2 [23:0] | H'00_0000 | —/WB | Code value 2 When writing this register, write H'A5_A5A5 to this bit field. When reading, these bits are always read as 0. |
| 7 | TME | B'0 | R/W | Starts and stops timer operation. 0: Timer disabled: Count-up stops and RWTCNT value is retained 1: Timer enabled |
| 6 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 5 | WRFLG | B'0 | R | Write Status Flag When this bit is 1, write access to RWTCNT is prohibited. If it isn't maintained, operation isn't secured. This bit indicates the period that the writing to RWTCNT is masked for synchronization after the writing to RWTCNT Confirm that this flag is 0 when RWTCNT is written to continuously. |
| 4 | WOVF | B'0 | R/WC0 | Indicates that the RWTCNT has overflowed. This bit isn't initialized by a generated reset by an overflow of RWDT and section 65. System Watchdog Timer (SWDT). Write 0 to this bit before using the RWDT. 0: No overflow 1: RWTCNT has overflowed Note: Only 0 can be written to clear the flag. |
| 3 | WOVFE | B'1 | R/W | Overflow Interrupt Disable/Enable 0: Disables interrupts due to overflow 1: Enables interrupts due to overflow |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|--|
| 2 to 0 | CKS0[2:0] | B'111 | R/W | <p>RTC Clock Select</p> <p>These bits select the clock to be used for the RWTCNT count from the eight types obtainable by dividing the RCLK. The overflow period that is shown inside the parenthesis in the table is the value when the RCLK is 32.768 kHz.</p> <p>B'000: RCLK (RWTCNT: H'0000 = 2.0 s (RWTCNT: H'FF00 = 7.8 ms))</p> <p>B'001: RCLK/4 (RWTCNT: H'0000 = 8.0 s (RWTCNT: H'FF00 = 31.3 ms))</p> <p>B'010: RCLK/16 (RWTCNT: H'0000 = 32.0 s (RWTCNT: H'FF00 = 125.0 ms))</p> <p>B'011: RCLK/32 (RWTCNT: H'0000 = 64.0 s (RWTCNT: H'FF00 = 250.0 ms))</p> <p>B'100: RCLK/64 (RWTCNT: H'0000 = 128.0 s (RWTCNT: H'FF00 = 500.0 ms))</p> <p>B'101: RCLK/128 (RWTCNT: H'0000 = 256.0 s (RWTCNT: H'FF00 = 1.0 s))</p> <p>B'110: RCLK/1024 (RWTCNT: H'0000 = 2048.0 s (RWTCNT: H'FF00 = 8.0 s))</p> <p>B'111: RCLK select expanded mode</p> <p>The clock cycle varies depending on the CKS1 bit in RWTCSRB. If RWTCSRB.CKS1[3:0] is H'0, division ratio is RCLK/4096 and values of overflow time is RWTCNT: H'0000 = 8192.0 s (RWTCNT: H'FF00 = 32.0 s). If RWTCSRB.CKS1[3:0] is other than H'0, values of overflow time is calculated with formula in chapter 64.3.4.</p> <p>Note. These overflow periods are calculated from the rounded value of RCLK = 32.768 kHz as example.</p> |

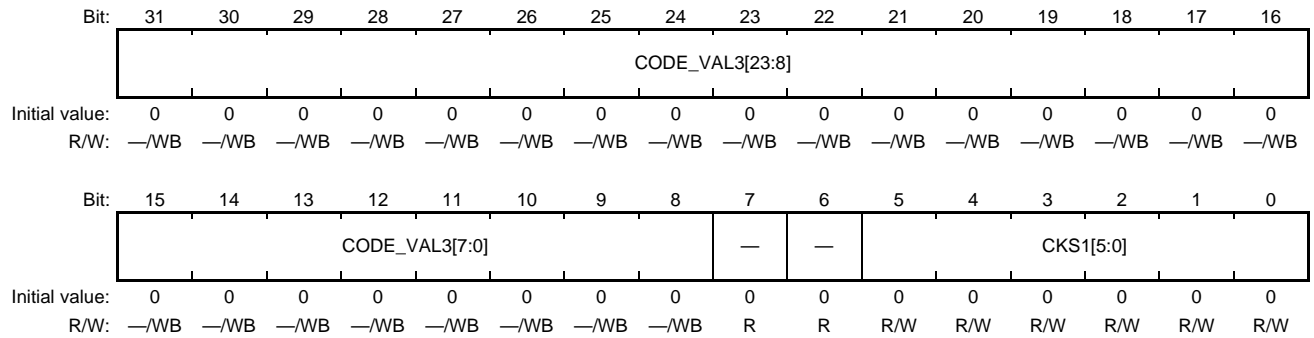
Note: If bits CKS0 [2:0] are modified when the RWDT is operating, the up-count may not be performed correctly. Ensure that the bits CKS0 [2:0] are modified only when the RWDT is not operating.

64.2.3 RCLK Watchdog Timer Control/Status Register B (RWTC SRB)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

RWTC SRB is an 8-bit readable/writable register composed of bits to select the clock used for the count.

Use a word access to write to RWTC SRB, with H'A5_A5A5 in the upper byte. Use a byte access to read RWTC SRB.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|------|--|
| 31 to 8 | CODE_VAL3[23:0] | H'00_0000 | -/WB | Code value 3 When writing this register, write H'A5_A5A5 to this bit field. When reading, these bits are always read as 0. |
| 7, 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|---|
| 5 to 0 | CKS1[5:0] | B'00_0000 | R/W | RCLK Select for RCLK Select Expanded Mode |

Selects the clock used for the RWTCNT count when the CKS0 bit in RWTCSCRA is B'111.

The following table shows the time values used to cause an overflow from the counter H'0000(Overflow time from counter H'FF00). Assuming the overflow cycle in the table is calculated for the case when the RCLK is 32.768 kHz. For more values of overflow time, refer chapter 64.3.4.

| CKS1[3:0] | CKS1[5:4] | | | |
|-----------|----------------------------|----------------------------|----------------------------|----------------------------|
| | B'00 | B'01 | B'10 | B'11 |
| H'0 | 8192s (32 s) | | | |
| H'1 | 256 s (1 s) | 1280 s (5 s) | 256 min (1 min) | 1280 min (5 min) |
| H'2 | 512 s (2 s) | 2560 s (10 s) | 512 min (2 min) | 2560 min (10 min) |
| H'3 | 768 s (3 s) | 3840 s (15 s) | 768 min (3 min) | 3840 min (15 min) |
| H'4 | 1024 s (4 s) | 5120 s (20 s) | 1024 min (4 min) | 5120 min (20 min) |
| H'5 | 1280 s (5 s) | 6400 s (25 s) | 1280 min (5 min) | 6400 min (25 min) |
| H'6 | 1536 s (6 s) | 7680 s (30 s) | 1536 min (6 min) | 7680 min (30 min) |
| H'7 | 1792 s (7 s) | 8960 s (35 s) | 1792 min (7 min) | Prohibited (Prohibited) |
| H'8 | 2048 s (8 s) | 10240 s (40 s) | 2048 min (8 min) | Prohibited (Prohibited) |
| H'9 | 2304 s (9 s) | 11520 s (45 s) | 2304 min (9 min) | Prohibited (Prohibited) |
| H'A | 2560 s (10 s) | 12800 s (50 s) | 2560 min (10 min) | Prohibited (Prohibited) |
| H'B | Prohibited (Prohibited) | 14080 s (55 s) | Prohibited (Prohibited) | Prohibited (Prohibited) |
| H'C | Prohibited (Prohibited) | 15360 s (60 s) | Prohibited (Prohibited) | Prohibited (Prohibited) |
| from H'D | Prohibited (Prohibited) | Prohibited (Prohibited) | Prohibited (Prohibited) | Prohibited (Prohibited) |

Note. These overflow periods are calculated from the rounded value of RCLK = 32.768 kHz as example.

Note: If the CKS1 [5] to CKS1 [0] bits are modified when the RWDT is running, the up-count may not be performed correctly. Ensure that these bits are modified only when the RWDT is not running.

64.2.4 Notes on Register Access

RWTCSRA: Bits TME, CKS0

RWTCSRB: Bits CKS1

After writing a value in this register, it is possible to read that value 2 cycles in counter input clock (RCLK) after writing has finished. And 2 cycles of RCLK need to be inputted until the set-up value is reflected.

RWTCNT: Bits 15 to 0

After writing a value in this register, it is possible to read that value 2 cycles in counter input clock (RCLK) after writing has finished. And 2 cycles of RCLK need to be inputted until the set-up value is reflected. Writing to this register, once started, is protected from further write operations until the writing operation has been completed. Read RWTCSRA. If WRFLG is 0, the next value can be written in RWTCNT.

The writing procedure to RWTCNT, RWTCSRA, and RWTCSRB differs from that of other registers with the purpose of preventing an unintended write. The procedure for writing to these registers is given below.

Writing to RWTCNT, RWTCSRA, and RWTCSRB:

- These registers must be written by a long word transfer instruction. They cannot be written by a byte or word transfer instruction.
- When writing to RWTCNT, set the upper byte to H'5A5A and transfer the lower byte as the write data. When writing to RWTCSRA and RWTCSRB, set the upper byte to H'A5_A5A5 and transfer the lower byte as the write data.

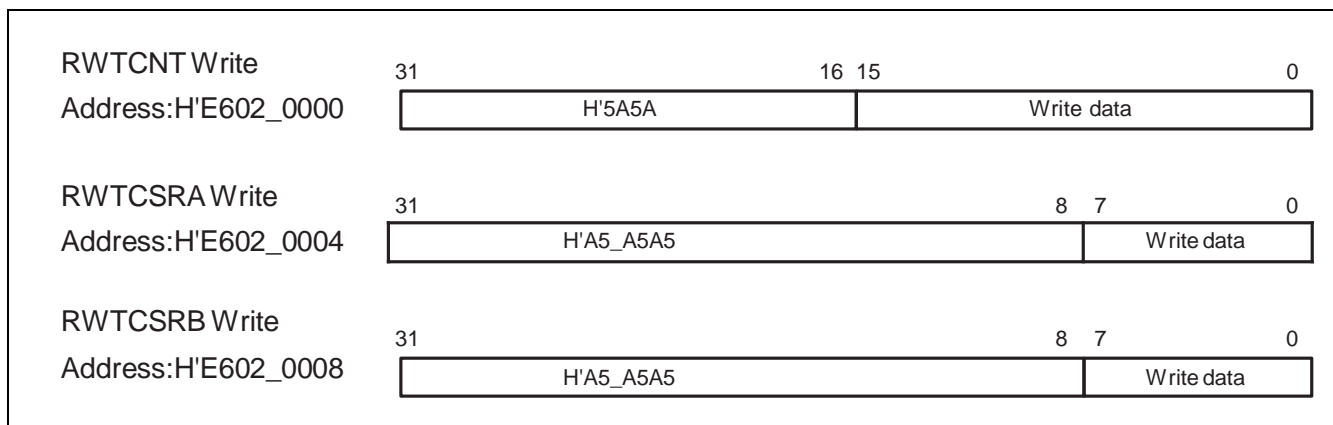


Figure 64.2 Writing to RWTCNT, RWTCSRA, and RWTCSRB

64.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

64.3.1 Control of System Runaway

After a reset, RWDT is disabled. When the counter overflow occurs after the counter starts, an internal reset is again generated. By this function, a reset can be automatically generated even when this LSI has caused a system runaway. While this LSI is operating correctly, write H'0000 (or the set value) to RWTCNT periodically so that RWTCNT does not overflow.

Take the following steps to use the RWDT.

1. Clear the TME bit in RWTCSSRA to 0 to temporarily stop counting.
(Confirm that RCLK was input more than 2 cycles in case RWTCNT is not initialized)
2. Write H'0000 or the set value in RWTCNT
3. Clear the WOVF bit in RWTCSSRA to 0.
4. Set the kind of count clock to the CKS0 [2:0] bits in RWTCSSRA, and CKS1 [5:0] bits in RWTCSSRB.
5. Confirm that RWTCSSRA.WRFLG becomes 0.
If WRFLG is 1, wait until it'll be 0.
6. Start the counting by setting the TME bit in RWTCSSRA to 1.
(TME and CKS0 can't be set at the same time. When setting that at the same time, behavior isn't guaranteed.)
7. While this LSI is operating correctly, write H'0000 (or the set value) to RWTCNT periodically so that RWTCNT does not overflow.
8. When RWTCNT overflows, Reset is generated because the RWDT sets the WOVF flag in RWTCSSRA to 1. At this time, RWTCNT, RWTCSSRA (excluding the WOVF bit), and RWTCSSRB are initialized.

Execute a power-on reset or a software reset at that time and initialize RWDT.

When not initializing and using RWDT, behavior isn't guaranteed.

64.3.2 Module Stop

A setup of a stop of the Clock by a Module Stop signal is possible when RWDT is an idle state. The Idle state of RWDT is when the condition of following all was satisfied.

- When RWTCSSRA.TME bit is 0
- When RWTCSSRA.WRFLG is 0
- When 3 or more cycles have passed in RCLK since the last Register access

64.3.3 Debugging Operation

When debugging the CPU core, stop counting up RWTCNT and retain the value of RWTCNT.

64.3.4 Formula to calculate overflow time

This is the formula to calculate the overflow time for RWDT when choosing Expanded Mode.

$$\text{Overflow time} = 128 \times T_{\text{RCLK}} \times \text{opset_val} \times (65536 - \text{RWTCNT})$$

T_{RCLK} : is the period of RCLK clock.

RWTCNT: is value of RWTCNT in decade.

opset_val: refer table below.

| CKS1[5:0] | opset_val | CKS1[5:0] | opset_val | CKS1[5:0] | opset_val | CKS1[5:0] | opset_val |
|------------|-----------|------------|-----------|------------|-----------|------------|-----------|
| H'00 | 32 | H'10 | 32 | H'20 | 32 | H'30 | 32 |
| H'01 | 1 | H'11 | 5 | H'21 | 60 | H'31 | 300 |
| H'02 | 2 | H'12 | 10 | H'22 | 120 | H'32 | 600 |
| H'03 | 3 | H'13 | 15 | H'23 | 180 | H'33 | 900 |
| H'04 | 4 | H'14 | 20 | H'24 | 240 | H'34 | 1200 |
| H'05 | 5 | H'15 | 25 | H'25 | 300 | H'35 | 1500 |
| H'06 | 6 | H'16 | 30 | H'26 | 360 | H'36 | 1800 |
| H'07 | 7 | H'17 | 35 | H'27 | 420 | Other case | 1 |
| H'08 | 8 | H'18 | 40 | H'28 | 480 | | |
| H'09 | 9 | H'19 | 45 | H'29 | 540 | | |
| H'0A | 10 | H'1A | 50 | H'2A | 600 | | |
| Other case | 1 | H'1B | 55 | Other case | 1 | | |
| | | H'1C | 60 | | | | |
| | | Other case | 1 | | | | |

65. System Watchdog Timer (SWDT)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

65.1 Overview

This LSI includes the System Watchdog Timer (SWDT).

This LSI can be reset by the overflow of the counter when the value of the counter has not been updated because of a system runaway.

The System Watchdog timer is a single-channel timer that uses the OSCCLK as an input clock and can be used as a watchdog timer. OSCCLK is a clock which is generated by CPG.

65.1.1 Features

- One channel is provided.
- Can be used as a watchdog timer. A reset is generated when the counter overflows.

65.1.2 Block Diagram

Figure 65.1 shows a block diagram of the System Watchdog Timer.

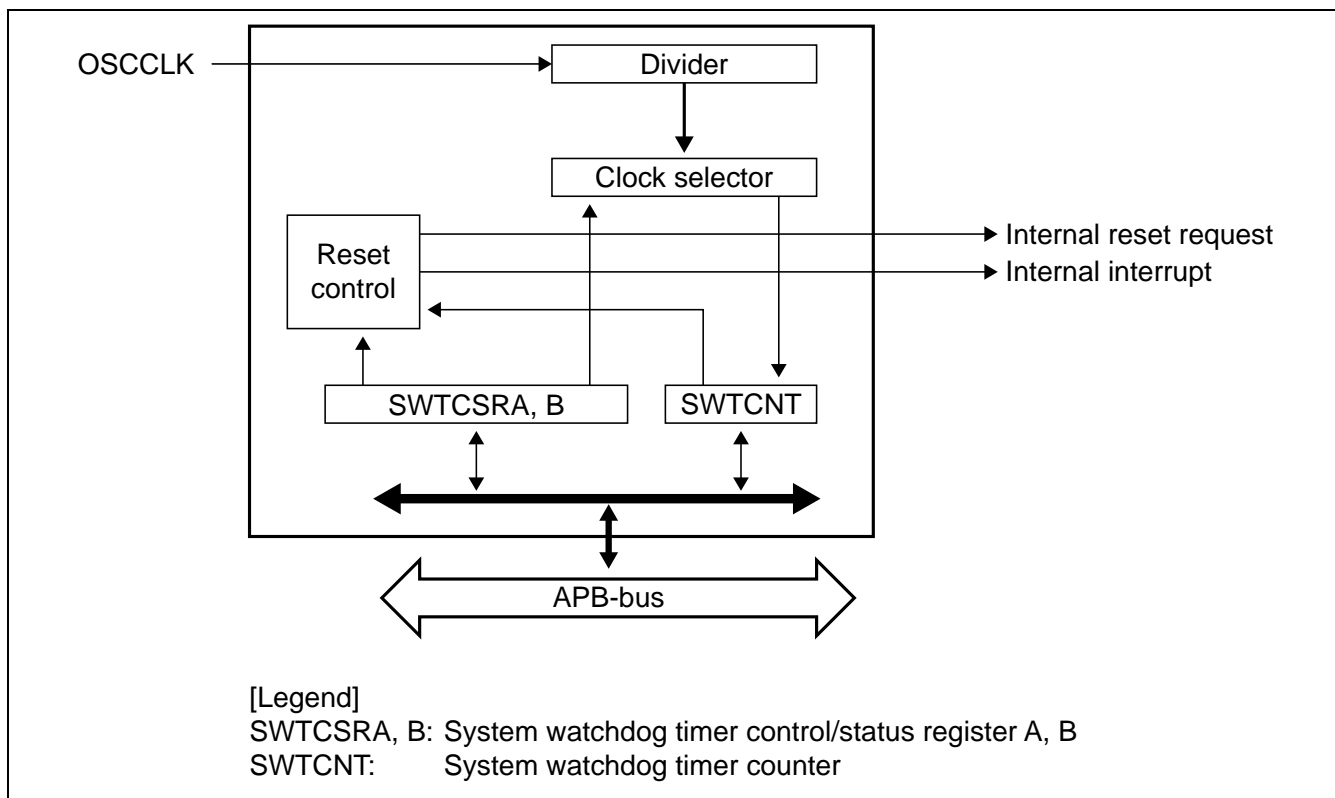


Figure 65.1 Block Diagram of System Watchdog Timer

65.1.3 Register Configuration

Table 65.1 shows the System Watchdog Timer register configuration. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined.

Table 65.2 shows the register state in each processing mode.

Table 65.1 Register Configuration of System Watchdog Timer

| Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|--------------|-------|-------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| System watchdog timer counter | SWTCNT | R/W | H'E603_0000 | 16/32*1 | √ | √ | √ | √ |
| System watchdog timer control/status register A | SWTCSRA | R/W*3 | H'E603_0004 | 8/32*2 | √ | √ | √ | √ |
| System watchdog timer control/status register B | SWTCSR B | R/W | H'E603_0008 | 8/32*2 | √ | √ | √ | √ |

- Notes: 1. Write is performed in 32 bits and read in 16 bits. Operation cannot be guaranteed if the register is not written as a longword (32bits) and not read as a word (16bits).
 2. Write is performed in 32 bits and read in 8 bits. Operation cannot be guaranteed if the register is not written as a longword (32bits) and not read as a byte (8bits).
 3. Except for bit [4] that is R/WC0 register.

Table 65.2 Register State of System Watchdog Timer in Each Processing Mode

| Register Abbreviation | | Reset Caused by Other than Overflow | Reset Caused by Overflow* |
|-----------------------|-----------------|-------------------------------------|---------------------------|
| Register | Bit | | |
| SWTCNT | All | Initialized | Initialized |
| SWTCSRA | WOVF | Initialized | Retained |
| | Other than WOVF | Initialized | Initialized |
| SWTCSR B | All | Initialized | Initialized |

Note: * System Watchdog Timer (SWDT) and section 64. RCLK Watchdog Timer (RWDT) overflow reset.

65.2 Register Description

[Legend for Register Description]

Initial value: Register value after a reset.

—: Undefined value

R/W: Bit or field is readable and writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R: Read-only. The write value should always be initial value.

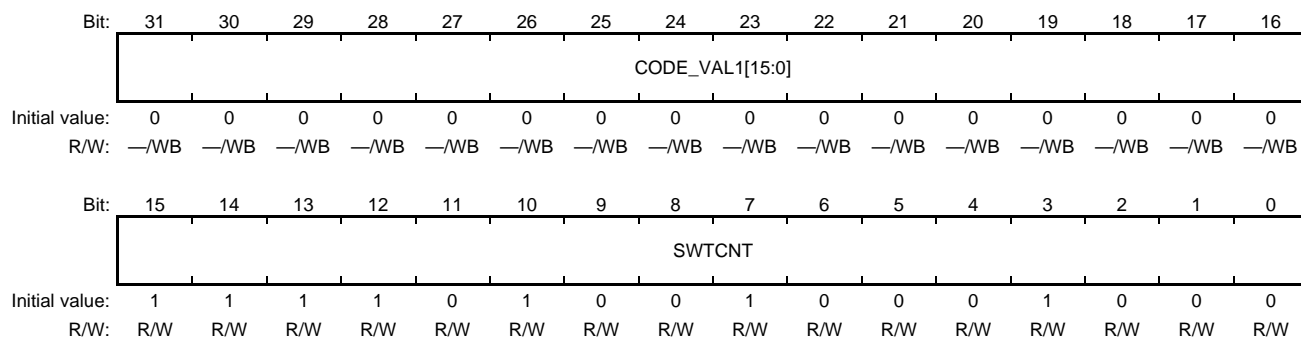
—/WB: Write-only. The read value is initial value.

65.2.1 System Watchdog Timer Counter (SWTCNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SWTCNT is a 16-bit readable/writable register that increments on the selected clock. When an overflow occurs, it generates a reset. The SWTCNT counter is initialized to H'F488 by a power-on reset.

Use a long word access to write to the SWTCNT counter, with H'5A5A in the upper byte. Use a word access to read SWTCNT.



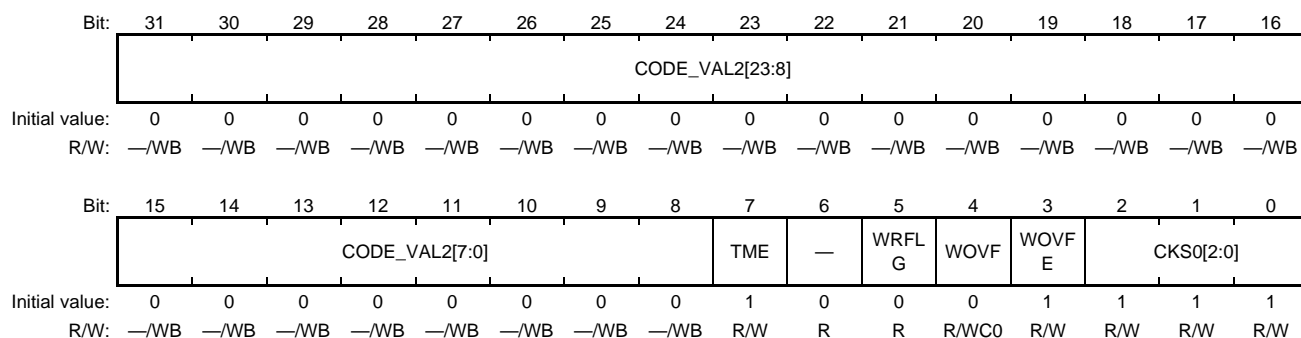
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------------|---------------|------|---|
| 31 to 16 | CODE_VAL1 [15:0] | H'0000 | —/WB | Code value 1 When writing this register, write H'5A5A to this bit field. When reading, these bits are always read as 0. |
| 15 to 0 | SWTCNT | H'F488 | R/W | Timer Counter Bits |

65.2.2 System Watchdog Timer Control/Status Register A (SWTCSRA)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SWTCSRA is an 8-bit readable/writable register composed of bits to select the clock used for the count, overflow flag, and enable bit.

Use a long word access to write to SWTCSRA, with H'A5_A5A5 in the upper byte. Use a byte access to read SWTCSRA.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------|---------------|-------|---|
| 31 to 8 | CODE_VAL2 [23:0] | H'00_0000 | —/WB | Code value 2 When writing this register, write H'A5_A5A5 to this bit field. When reading, these bits are always read as 0. |
| 7 | TME | B'1 | R/W | Starts and stops timer operation. 0: Timer disabled: Count-up stops and SWTCNT value is retained 1: Timer enabled |
| 6 | — | B'0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 5 | WRFLG | B'0 | R | Write Status Flag When this bit is 1, write access to SWTCNT is prohibited. If it isn't maintained, operation isn't secured. This bit indicates the period that the writing to SWTCNT is masked for synchronization after the writing to SWTCNT. Confirm that this flag is 0 when SWTCNT is written to continuously. |
| 4 | WOVF | B'0 | R/WC0 | Indicates that the SWTCNT has overflowed. This bit isn't initialized by a generated reset by an overflow of section 64. RCLK Watchdog Timer (RWDT) and System Watchdog Timer. Write 0 to this bit before using the System Watchdog Timer. 0: No overflow 1: SWTCNT has overflowed Note: Only 0 can be written to clear the flag. |
| 3 | WOVFE | B'1 | R/W | Overflow Interrupt Disable/Enable 0: Disables interrupts due to overflow 1: Enables interrupts due to overflow |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|---|
| 2 to 0 | CKS0[2:0] | B'111 | R/W | <p>RTC Clock Select</p> <p>These bits select the clock to be used for the SWTCNT count from the eight types obtainable by dividing the OSCCLK. The overflow period that is shown inside the parenthesis in the table is the value when the OSCCLK is 131.578 kHz.</p> <p>B'000: OSCCLK (SWTCNT: H'0000 = 498 ms (SWTCNT: H'FBFC = 7.8 ms))</p> <p>B'001: OSCCLK /4 (SWTCNT: H'0000 = 2.0 s (SWTCNT: H'FBFC = 31.3 ms))</p> <p>B'010: OSCCLK /16 (SWTCNT: H'0000 = 8.0 s (SWTCNT: H'FBFC = 125.0 ms))</p> <p>B'011: OSCCLK /32 (SWTCNT: H'0000 = 16 s (SWTCNT: H'FBFC = 250.0 ms))</p> <p>B'100: OSCCLK /64 (SWTCNT: H'0000 = 32 s (SWTCNT: H'FBFC = 500.0 ms))</p> <p>B'101: OSCCLK /128 (SWTCNT: H'0000 = 64 s (SWTCNT: H'FBFC = 1.0 s))</p> <p>B'110: OSCCLK /1024 (SWTCNT: H'0000 = 510.0 s (SWTCNT: H'FBFC = 8.0 s))</p> <p>B'111: OSCCLK select expanded mode</p> <p>The clock cycle varies depending on the CKS1 bit in SWTCSR.B. If SWTCSR.B.CKS1[3:0] is H'0, division ratio is OSCCLK/4096 and values of overflow time is SWTCNT: H'0000 = 2040.0 s (SWTCNT: H'FBFC = 32.0 s). If SWTCSR.B.CKS1[3:0] is other than H'0, values of overflow time is calculated with formula in chapter 65.3.3.</p> <p>Note: These overflow periods are calculated from the rounded value of OSCCLK = 131.578 kHz.</p> |

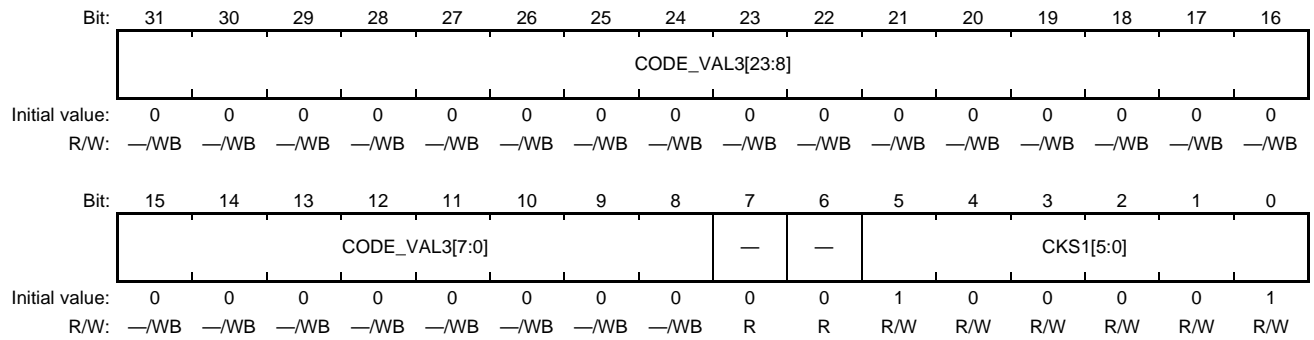
Note: If bits CKS0 [2:0] are modified when the System Watchdog Timer is operating, the up-count may not be performed correctly. Ensure that the bits CKS0 [2:0] are modified only when the System Watchdog Timer is not operating.

65.2.3 System Watchdog Timer Control/Status Register B (SWTCSR_B)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

SWTCSR_B is an 8-bit readable/writable register composed of bits to select the clock used for the count.

Use a word access to write to SWTCSR_B, with H'A5_A5A5 in the upper byte. Use a byte access to read SWTCSR_B.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|------|--|
| 31 to 8 | CODE_VAL3[23:0] | H'00_0000 | —/WB | Code value 3 When writing this register, write H'A5_A5A5 to this bit field. When reading, these bits are always read as 0. |
| 7, 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|---|
| 5 to 0 | CKS1[5:0] | B'10_0001 | R/W | OSCCLK Select for OSCCLK Select Expanded Mode |

Selects the clock used for the SWTCNT count when the CKS0 bit in SWTCSRA is B'111.

The following table shows the time values used to cause an overflow from the counter H'0F13 (Overflow from counter H'FBFC). Assuming the overflow cycle in the table is calculated for the case when the OSCCLK is 131.578 kHz. For more values of overflow time, refer to chapter 65.3.3

| CKS1[3:0] | CKS1[5:4] | | | |
|-----------|----------------------------|----------------------------|----------------------------|----------------------------|
| | B'00 | B'01 | B'10 | B'11 |
| H'0 | 1920 s (32 s) | | | |
| H'1 | 60 s (1 s) | 5 min (5 s) | 60 min (1 min) | 300 min (5 min) |
| H'2 | 120 s (2 s) | 10 min (10 s) | 120 min (2 min) | 600 min (10 min) |
| H'3 | 180 s (3 s) | 15 min (15 s) | 180 min (3 min) | 900 min (15 min) |
| H'4 | 240 s (4 s) | 20 min (20 s) | 240 min (4 min) | 1200 min (20 min) |
| H'5 | 5 min (5 s) | 25 min (25 s) | 300 min (5 min) | 1500 min (25 min) |
| H'6 | 6 min (6 s) | 30 min (30 s) | 360 min (6 min) | 1800 min (30 min) |
| H'7 | 7 min (7 s) | 35 min (35 s) | 420 min (7 min) | Prohibited (Prohibited) |
| H'8 | 8 min (8 s) | 40 min (40 s) | 480 min (8 min) | Prohibited (Prohibited) |
| H'9 | 9 min (9 s) | 45 min (45 s) | 540 min (9 min) | Prohibited (Prohibited) |
| H'A | 10 min (10 s) | 50 min (50 s) | 600 min (10 min) | Prohibited (Prohibited) |
| H'B | Prohibited (Prohibited) | 55 min (55 s) | Prohibited (Prohibited) | Prohibited (Prohibited) |
| H'C | Prohibited (Prohibited) | 60 min (60 s) | Prohibited (Prohibited) | Prohibited (Prohibited) |
| from H'D | Prohibited (Prohibited) | Prohibited (Prohibited) | Prohibited (Prohibited) | Prohibited (Prohibited) |

Note: If the CKS1 [5] to CKS1 [0] bits are modified when the System Watchdog Timer is running, the up-count may not be performed correctly. Ensure that these bits are modified only when the System Watchdog Timer is not running.

65.2.4 Notes on Register Access

SWTCSRA: Bits TME, CKS0

SWTCSRБ: Bits CKS1

After writing a value in this register, it is possible to read that value 2 cycles in counter input clock (OSCCLK) after writing has finished. And 2 cycles of OSCCLK need to be inputted until the set-up value is reflected.

SWTCNT: Bits 15 to 0

After writing a value in this register, it is possible to read that value 2 cycles in counter input clock (OSCCLK) after writing has finished. And 2 cycles of OSCCLK need to be inputted until the set-up value is reflected. Writing to this register, once started, is protected from further write operations until the writing operation has been completed. Read SWTCSRA. If WRFLG is 0, the next value can be written in SWTCNT.

The writing procedure to SWTCNT, SWTCSRA, and SWTCSRБ differs from that of other registers with the purpose of preventing an unintended write. The procedure for writing to these registers is given below.

Writing to SWTCNT, SWTCSRA, and SWTCSRБ

- These registers must be written by a long word transfer instruction. They cannot be written by a byte or word transfer instruction.
- When writing to SWTCNT, set the upper byte to H'5A5A and transfer the lower byte as the write data. When writing to SWTCSRA and SWTCSRБ, set the upper byte to H'A5_A5A5 and transfer the lower byte as the write data.

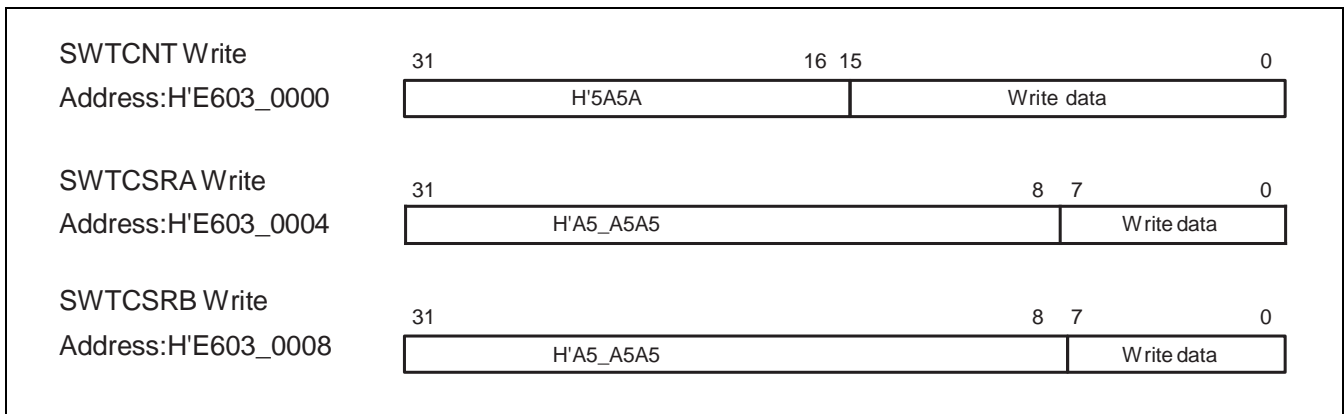


Figure 65.2 Writing to SWTCNT, SWTCSRA, and SWTCSRБ

65.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

65.3.1 Control of System Runaway

After a reset, System Watchdog Timer is enabled. When the counter overflow occurs after the counter starts, an internal reset is again generated. By this function, a reset can be automatically generated even when this LSI has caused a system runaway. While this LSI is operating correctly, write H'0000 (or the set value) to SWTCNT periodically so that SWTCNT does not overflow.

Take the following steps to use the System Watchdog Timer.

1. Clear the TME bit in SWTCSRA to 0 to temporarily stop counting.
(Confirm that OSCCLK was input more than 2 cycles in case SWTCNT is not initialized)
2. Write H'0000 or the set value in SWTCNT
3. Clear the WOVF bit in SWTCSRA to 0.
4. Set the kind of count clock to the CKS0 [2:0] bits in SWTCSRA, and CKS1 [5:0] bits in SWTCSRB.
5. Confirm that SWTCSRA.WRFLG becomes 0.
If WRFLG is 1, wait until it'll be 0.
6. Start the counting by setting the TME bit in SWTCSRA to 1.
(TME and CKS0 can't be set at the same time. When setting that at the same time, behavior isn't guaranteed.)
7. While this LSI is operating correctly, write H'0000 (or the set value) to SWTCNT periodically so that SWTCNT does not overflow.
8. When SWTCNT overflows, Reset is generated because the System Watchdog Timer sets the WOVF flag in SWTCSRA to 1. At this time, SWTCNT, SWTCSRA (excluding the WOVF bit), and SWTCSRB are initialized.

Execute a power-on reset or a software reset at that time and initialize System Watchdog Timer.

When not initializing and using System Watchdog Timer, behavior isn't guaranteed.

65.3.2 Debugging Operation

When debugging the CPU core, stop counting up SWTCNT and retain the value of SWTCNT.

65.3.3 Formula to calculate overflow time

This is the formula to calculate the overflow time for SWDT when choosing Expanded Mode.

$$\text{Overflow time} = 128 \times T_{\text{OSCCLK}} \times \text{opset_val} \times (65536 - \text{SWTCNT})$$

T_{OSCCLK} : is the period of OSCCLK clock.

SWTCNT: is value of SWTCNT in decade.

opset_val: refer to table below.

| CKS1[5:0] | opset_val | CKS1[5:0] | opset_val | CKS1[5:0] | opset_val | CKS1[5:0] | opset_val |
|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| H'00 | 32 | H'10 | 32 | H'20 | 32 | H'30 | 32 |
| H'01 | 1 | H'11 | 5 | H'21 | 60 | H'31 | 300 |
| H'02 | 2 | H'12 | 10 | H'22 | 120 | H'32 | 600 |
| H'03 | 3 | H'13 | 15 | H'23 | 180 | H'33 | 900 |
| H'04 | 4 | H'14 | 20 | H'24 | 240 | H'34 | 1200 |

| | | | | | | | |
|------------|----|------------|----|------------|-----|------------|------|
| H'05 | 5 | H'15 | 25 | H'25 | 300 | H'35 | 1500 |
| H'06 | 6 | H'16 | 30 | H'26 | 360 | H'36 | 1800 |
| H'07 | 7 | H'17 | 35 | H'27 | 420 | Other case | 1 |
| H'08 | 8 | H'18 | 40 | H'28 | 480 | | |
| H'09 | 9 | H'19 | 45 | H'29 | 540 | | |
| H'0A | 10 | H'1A | 50 | H'2A | 600 | | |
| Other case | 1 | H'1B | 55 | Other case | 1 | | |
| | | H'1C | 60 | | | | |
| | | Other case | 1 | | | | |

66. 16-Bit Timer Pulse Unit (TPU)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

66.1 Overview

This LSI has a 16-bit timer pulse unit (TPU), which consists of four 16-bit timers.

Internal clock:

- RZ/G2H: S3D4 ϕ
- RZ/G2M V1.3: S3D4 ϕ
- RZ/G2M V3.0: S3D4 ϕ
- RZ/G2N: S3D4 ϕ
- RZ/G2E: S3D4C ϕ

66.1.1 Features

The TPU has the following features.

- Four timers are incorporated.
- Four pulse outputs are possible.
- Each timer of the TPU has four general registers (TGRA, TGRB, TGRC, and TGRD) for each timer. TGRA enables an output compare setting. TGRB, TGRC and TGRD in each timer can be used as timer-counter clearing registers. TGRC and TGRD can also be used as buffer registers.
- The following operations can be set for each timer.

Waveform output on compare match: output of a 0 or 1 or toggling of output is selectable.

Counter clearing operation: Counters can be cleared in response to a compare match.

PWM mode allows the output of a PWM waveform with any duty cycle.

The four outputs can be used to produce PWM output in up to four phases.

- A stepping motor control mode is available for timer 0.
- Buffer operation can be set for each timer.
- Automatic rewriting of values in output compare registers is possible.
- One interrupt request from TPU

Enabling or disabling of compare match and counter overflows as interrupt sources can be set independently for each timer.

In stepping motor control mode, enabling or disabling of state transitions, requests for data, and overflows of data as interrupt sources is possible for timer 0.

Table 66.1 shows the TPU functions.

Table 66.1 TPU Functions

| Item | | Timer 0 | Timer 1 | Timer 2 | Timer 3 |
|---|-----------------------------|--|------------------------|------------------------|------------------------|
| Counter clock | | Internal clock/1 | Internal clock/1 | Internal clock/1 | Internal clock/1 |
| | | Internal clock/4 | Internal clock/4 | Internal clock/4 | Internal clock/4 |
| | | Internal clock/16 | Internal clock/16 | Internal clock/16 | Internal clock/16 |
| | | Internal clock/64 | Internal clock/64 | Internal clock/64 | Internal clock/64 |
| General registers | | TGRA0 TGRB0 | TGRA1 TGRB1 | TGRA2 TGRB2 | TGRA3 TGRB3 |
| General and registers/ Buffer register | | TGRC0 TGRD0 | TGRC1 TGRD1 | TGRC2 TGRD2 | TGRC3 TGRD3 |
| Output pins | Normal operation | TPU0TO0 | TPU0TO1 | TPU0TO2 | TPU0TO3 |
| | Stepping motor control mode | TPU0TO0 to TPU0TO3 | None | None | None |
| Counter clearing | | Compare match with TGR | Compare match with TGR | Compare match with TGR | Compare match with TGR |
| Output in response to compare match | 0 | √ | √ | √ | √ |
| | 1 | √ | √ | √ | √ |
| | Toggling | √ | √ | √ | √ |
| PWM mode | | √ | √ | √ | √ |
| Buffer operation | | √ | √ | √ | √ |
| Interrupt | | 5 sources Compare match/ overflow 6 more sources added in stepping motor control mode/ Transition to acceleration state /transition to normal state /transition to deceleration state /transition to stop state /data request /data overflow | | | |

66.1.2 Block Diagram

Figure 66.1 is a block diagram of the TPU.

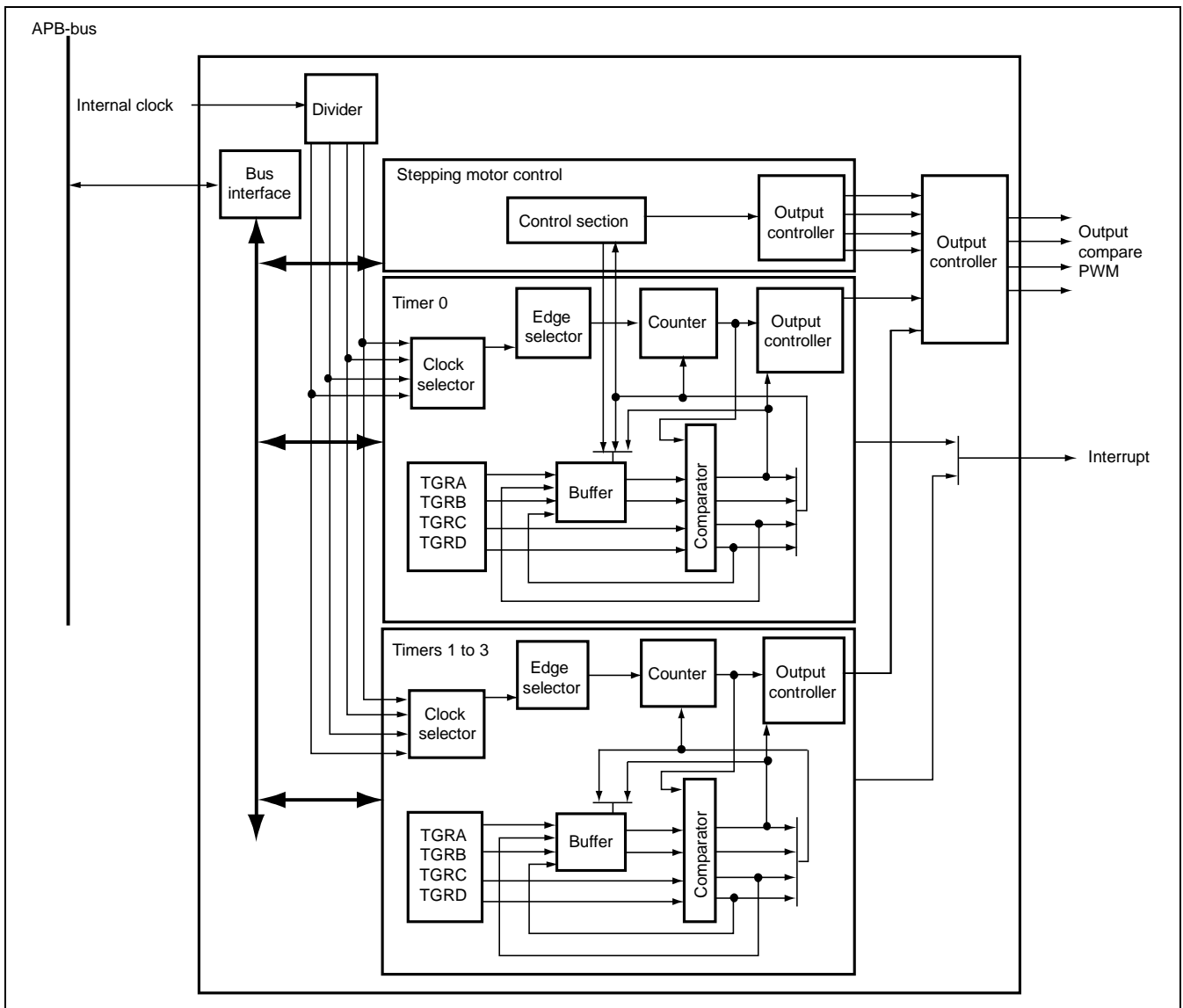


Figure 66.1 Block Diagram of the TPU

66.1.3 External Pins

Table 66.2 shows the pin configuration of the TPU.

Table 66.2 Pin Configuration

| Module | Timer | Pin Name | Function | I/O | Description | Second Generation RZ/G Series Products | | | |
|--------|-------|----------|----------------------------|--------|--|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| TPU | 0 | TPU0TO0 | TPU output compare match 0 | Output | TGRA0 output compare output/ PWM output pin/pattern output pin 0 in stepping motor control mode | √ | √ | √ | √ |
| | 1 | TPU0TO1 | TPU output compare match 1 | Output | TGRA1 output compare output/ PWM output pin/pattern output pin 1 in stepping motor control mode | √ | √ | √ | √ |
| | 2 | TPU0TO2 | TPU output compare match 2 | Output | TGRA2 output compare output/ PWM output pin/pattern output pin 2 in stepping motor control mode | √ | √ | √ | √ |
| | 3 | TPU0TO3 | TPU output compare match 3 | Output | TGRA3 output compare output/ PWM output pin/pattern output pin 3 in stepping motor control mode | √ | √ | √ | √ |

66.1.4 Register Configuration

Table 66.3 shows the TPU register configuration. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined. Access the register as a word (16 bits). Operation cannot be guaranteed if the register is not accessed as a word.

Table 66.4 shows the register states in each operating mode.

Table 66.3 Register Configuration

| | | | | | Second Generation RZ/G Series Products | | | |
|-----------------------------------|--------------|------|-------------|-------------|---|----------------------------|--------|--------|
| Register Name | Abbreviation | R/W | Address | Access Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| | | | | | | | | |
| Timer start register | TSTR | R/W | H'E6E8_0000 | 16 | √ | √ | √ | √ |
| Timer control register 0 | TCR0 | R/W | H'E6E8_0010 | 16 | √ | √ | √ | √ |
| Timer mode register 0 | TMDR0 | R/W | H'E6E8_0014 | 16 | √ | √ | √ | √ |
| Timer I/O control register 0 | TIOR0 | R/W | H'E6E8_0018 | 16 | √ | √ | √ | √ |
| Timer interrupt enable register 0 | TIER0 | R/W | H'E6E8_001C | 16 | √ | √ | √ | √ |
| Timer status register 0 | TSR0 | R/WC | H'E6E8_0020 | 16 | √ | √ | √ | √ |
| Timer counter 0 | TCNT0 | R/W | H'E6E8_0024 | 16 | √ | √ | √ | √ |
| Timer general register A0 | TGRA0 | R/W | H'E6E8_0028 | 16 | √ | √ | √ | √ |
| Timer general register B0 | TGRB0 | R/W | H'E6E8_002C | 16 | √ | √ | √ | √ |
| Timer general register C0 | TGRC0 | R/W | H'E6E8_0030 | 16 | √ | √ | √ | √ |
| Timer general register D0 | TGRD0 | R/W | H'E6E8_0034 | 16 | √ | √ | √ | √ |
| Timer control register 1 | TCR1 | R/W | H'E6E8_0050 | 16 | √ | √ | √ | √ |
| Timer mode register 1 | TMDR1 | R/W | H'E6E8_0054 | 16 | √ | √ | √ | √ |
| Timer I/O control register 1 | TIOR1 | R/W | H'E6E8_0058 | 16 | √ | √ | √ | √ |
| Timer interrupt enable register 1 | TIER1 | R/W | H'E6E8_005C | 16 | √ | √ | √ | √ |
| Timer status register 1 | TSR1 | R/W | H'E6E8_0060 | 16 | √ | √ | √ | √ |
| Timer counter 1 | TCNT1 | R/W | H'E6E8_0064 | 16 | √ | √ | √ | √ |
| Timer general register A1 | TGRA1 | R/W | H'E6E8_0068 | 16 | √ | √ | √ | √ |
| Timer general register B1 | TGRB1 | R/W | H'E6E8_006C | 16 | √ | √ | √ | √ |
| Timer general register C1 | TGRC1 | R/W | H'E6E8_0070 | 16 | √ | √ | √ | √ |
| Timer general register D1 | TGRD1 | R/W | H'E6E8_0074 | 16 | √ | √ | √ | √ |
| Timer control register 2 | TCR2 | R/W | H'E6E8_0090 | 16 | √ | √ | √ | √ |
| Timer mode register 2 | TMDR2 | R/W | H'E6E8_0094 | 16 | √ | √ | √ | √ |
| Timer I/O control register 2 | TIOR2 | R/W | H'E6E8_0098 | 16 | √ | √ | √ | √ |
| Timer interrupt enable register 2 | TIER2 | R/W | H'E6E8_009C | 16 | √ | √ | √ | √ |
| Timer status register 2 | TSR2 | R/W | H'E6E8_00A0 | 16 | √ | √ | √ | √ |
| Timer counter 2 | TCNT2 | R/W | H'E6E8_00A4 | 16 | √ | √ | √ | √ |
| Timer general register A2 | TGRA2 | R/W | H'E6E8_00A8 | 16 | √ | √ | √ | √ |
| Timer general register B2 | TGRB2 | R/W | H'E6E8_00AC | 16 | √ | √ | √ | √ |
| Timer general register C2 | TGRC2 | R/W | H'E6E8_00B0 | 16 | √ | √ | √ | √ |
| Timer general register D2 | TGRD2 | R/W | H'E6E8_00B4 | 16 | √ | √ | √ | √ |
| Timer control register 3 | TCR3 | R/W | H'E6E8_00D0 | 16 | √ | √ | √ | √ |

**Second Generation
RZ/G Series Products**

| Register Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | |
|---|--------------|-----|-------------|-------------|--|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Timer mode register 3 | TMDR3 | R/W | H'E6E8_00D4 | 16 | √ | √ | √ | √ |
| Timer I/O control register 3 | TIOR3 | R/W | H'E6E8_00D8 | 16 | √ | √ | √ | √ |
| Timer interrupt enable register 3 | TIER3 | R/W | H'E6E8_00DC | 16 | √ | √ | √ | √ |
| Timer status register 3 | TSR3 | R/W | H'E6E8_00E0 | 16 | √ | √ | √ | √ |
| Timer counter 3 | TCNT3 | R/W | H'E6E8_00E4 | 16 | √ | √ | √ | √ |
| Timer general register A3 | TGRA3 | R/W | H'E6E8_00E8 | 16 | √ | √ | √ | √ |
| Timer general register B3 | TGRB3 | R/W | H'E6E8_00EC | 16 | √ | √ | √ | √ |
| Timer general register C3 | TGRC3 | R/W | H'E6E8_00F0 | 16 | √ | √ | √ | √ |
| Timer general register D3 | TGRD3 | R/W | H'E6E8_00F4 | 16 | √ | √ | √ | √ |
| Motor control setting register | TMIR | R/W | H'E6E8_0100 | 16 | √ | √ | √ | √ |
| Motor deceleration (stop) transition register | TMRR | R/W | H'E6E8_0104 | 16 | √ | √ | √ | √ |
| Motor control status register | TMSR | R | H'E6E8_0108 | 16 | √ | √ | √ | √ |
| Motor operation pattern storing register 0 | TMMPR0 | R/W | H'E6E8_0110 | 16 | √ | √ | √ | √ |
| Motor operation pattern storing register 1 | TMMPR1 | R/W | H'E6E8_0114 | 16 | √ | √ | √ | √ |
| Motor stop pattern storing register 0 | TMSPR0 | R/W | H'E6E8_0118 | 16 | √ | √ | √ | √ |
| Motor stop pattern storing register 1 | TMSPR1 | R/W | H'E6E8_011C | 16 | √ | √ | √ | √ |
| Motor output pattern storing register | TMOPR | R/W | H'E6E8_0120 | 16 | √ | √ | √ | √ |
| Motor Acceleration Step Count Register | TMASR | R/W | H'E6E8_0130 | 16 | √ | √ | √ | √ |
| Motor normal the number of steps register | TMTSR | R/W | H'E6E8_0134 | 16 | √ | √ | √ | √ |
| Motor deceleration the number of steps register | TMRSR | R/W | H'E6E8_0138 | 16 | √ | √ | √ | √ |
| Motor control sequence counter register | TMSCR | R/W | H'E6E8_0140 | 16 | √ | √ | √ | √ |
| Motor control normal counter register | TMTCR | R | H'E6E8_0144 | 16 | √ | √ | √ | √ |

Table 66.4 Register States in Each Operating Mode

| Register Abbreviation | Power-on Reset/Software Reset * | Module Standby |
|------------------------------|--|-----------------------|
| TSTR | Initialized | Retained |
| TCR0 to TCR3 | Initialized | Retained |
| TMDR0 to TMDR3 | Initialized | Retained |
| TIOR0 to TIOR3 | Initialized | Retained |
| TIER0 to TIER3 | Initialized | Retained |
| TSR0 to TSR3 | Initialized | Retained |
| TCNT0 to TCNT3 | Initialized | Retained |
| TGRA0 to TGRA3 | Initialized | Retained |
| TGRB0 to TGRB3 | Initialized | Retained |
| TGRC0 to TGRC3 | Initialized | Retained |
| TGRD0 to TGRD3 | Initialized | Retained |
| TMIR | Initialized | Retained |
| TMRR | Initialized | Retained |
| TMSR | Initialized | Retained |
| TMMPR0 | Initialized | Retained |
| TMMPR1 | | |
| TMSPR0 | Initialized | Retained |
| TMSPR1 | | |
| TMOPR | Initialized | Retained |
| TMASR | Initialized | Retained |
| TMTSR | Initialized | Retained |
| TMRSR | Initialized | Retained |
| TMSCR | Initialized | Retained |
| TMTCR | Initialized | Retained |

Note: * Refer to 12. Module Standby, Software Reset and 11. Reset (RST).

66.2 Register Description

Registers in the TPU are allocated to and arranged in the address space of the internal bus.

Legend for Register Description

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R: Read-only. When the bit is reserved, the write value should always be 0. Writing 1 to these bits can cause a malfunction of TPU.

—/W: Write-only. The read value is undefined.

66.2.1 Timer Control Register (TCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TCR is provided for each of timers 0 to 3 and is used to control TCNT. TCR is initialized to H'0000 by a reset.

Writing to TCR is only allowed only when TCNT is stopped.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|-----------|-----|-----|-----------|-----|-----------|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | CCLR[2:0] | | | CKEG[1:0] | | TPSC[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 15 to 8 | — | All 0 | R | Reserved These bits are always read as 0 and cannot be modified. |
| 7 to 5 | CCLR[2:0] | B'000 | R/W | Counter Clear Specify the TCNT clearing source. B'000: TCNT clearing disabled B'001: TCNT cleared in response to TGRA compare match B'010: TCNT cleared in response to TGRB compare match B'011: Setting prohibited B'100: TCNT clearing disabled B'101: TCNT cleared in response to TGRC compare match B'110: TCNT cleared in response to TGRD compare match B'111: Setting prohibited |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|-----------|---------------|-----|--|
| 4, 3 | CKEG[1:0] | B'00 | R/W | <p>Clock Edge</p> <p>Select the input clock edge for counting up.</p> <p>Counting of both edges by the internal clock has the same effect as halving the cycle of the input clock (e.g., the rate of counting both edges of Internal clock/4 is same as the rate of counting rising edges of Internal clock/2). When the phase coefficient mode is selected, this setting is ignored.</p> <p>B'00: Count rising edges B'01: Count falling edges* 1X: Count both edges* [Legend] X: Don't care</p> <p>Note: * If the input clock is Internal clock/1, the timer does not operate with this setting.</p> |
| 2 to 0 | TPSC[2:0] | B'000 | R/W | <p>Timer Prescaler</p> <p>Select the clock for counting by TCNT.</p> <p>The clock source can be selected independently for each timer. Table 66.5 shows the clock sources that can be set for each timer. For more information on count clock selection, see Table 66.6.</p> |

Table 66.5 TPU Clock Sources

| Timer | Internal Clock | | | |
|-------|-------------------|-------------------|--------------------|--------------------|
| | Internal Clock /1 | Internal Clock /4 | Internal Clock /16 | Internal Clock /64 |
| 0 | √ | √ | √ | √ |
| 1 | √ | √ | √ | √ |
| 2 | √ | √ | √ | √ |
| 3 | √ | √ | √ | √ |

[Legend]

√: Setting

Table 66.6 Count Clock Selection by the TPSC [2:0] Bits

| Timer | TPSC[2] | TPSC[1] | TPSC[0] | Description | |
|--------|---------|---------|---------|-----------------------------------|--------------------|
| 0 to 3 | B'0 | B'0 | B'0 | Internal clock /1 (initial value) | |
| | | | B'1 | Internal clock /4 | |
| | | | B'1 | B'0 | Internal clock /16 |
| | | | B'1 | B'1 | Internal clock /64 |
| | B'1 | * | * | Setting prohibited | |

[Legend]

*: Don't care.

66.2.2 Timer Mode Register (TMDR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TMDR is provided for each of timers 0 to 3, and is used to specify the operating mode of the corresponding timer. TMDR is initialized to H'0000 by a reset.

Writing to TMDR is only allowed when TCNT is stopped.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|------|-----|-----|---|---------|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | BFWT | BFB | BFA | — | MD[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 15 to 7 | — | All 0 | R | Reserved These bits are always read as 0 and cannot be modified. |
| 6 | BFWT | B'0 | R/W | Buffer Write Timing Specifies when to update TGRA and TGRB when TGRC and TGRD are in use as compare match buffers. When TGRC and TGRD are not in use as compare match buffer registers, this has no effect. 0: TGRA and TGRB are rewritten on a compare match with the counters. 1: TGRA and TGRB are rewritten on clearing of the counter. |
| 5 | BFB | B'0 | R/W | Buffer Operation B Specifies operation mode of TGRB. 0: TGRB is in normal operation. 1: TGRB and TGRD are used for buffered operation. |
| 4 | BFA | B'0 | R/W | Buffer Operation A Specifies operation mode of TGRA. 0: TGRA normal operation. 1: TGRA and TGRC are used for buffered operation. |
| 3 | — | B'0 | R | Reserved This bit is always read as 0 and cannot be modified. |
| 2 to 0 | MD[2:0] | B'000 | R/W | Timer Operating Mode Specifies the timer operating mode. B'000: Normal operation B'001: Setting prohibited B'010: PWM mode B'011 to B'111: Setting prohibited |

66.2.3 Timer I/O Control Register (TIOR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TIOR is provided for each of timers 0 to 3, and is used to control the corresponding TPU0TO0 to TPU0TO3 pins. TIOR is initialized to H'0000 by a reset.

Writing to TIOR is only allowed when TCNT is stopped. Note that the setting of TMDR may affect TIOR.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|---|---|---|----------|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | IOA[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 15 to 3 | — | All 0 | R | Reserved These bits are always read as 0 and cannot be modified. |
| 2 to 0 | IOA[2:0] | B'000 | R/W | I/O Control Bits IOA2 to IOA0 specify the functions of TGRA and the TPU0TO0 to TPU0TO3 pins. For details, see Table 66.7. |

Table 66.7 Settings for Bits IOA [2:0], Initial States of Pins TPU0TO0 to TPU0TO3, and Results of Matching with TGRA

| Timer | IOA[2] | IOA[1] | IOA[0] | Description | |
|--------|--------|--------|--------|--|---------------------------------------|
| 0 to 3 | B'0 | B'0 | B'0 | Always output 0 (initial value) | |
| | | | B'1 | Initial output from the TPU0TO0 to TPU0TO3 pins is 0 | Output 0 on compare match with TGRA.* |
| | | | B'1 | B'0 | Output 1 on compare match with TGRA. |
| | | | B'1 | Toggle output on compare match with TGRA.* | |
| | B'1 | B'0 | B'0 | Always output 1 | |
| | | | B'1 | Initial output from the TPU0TO0 to TPU0TO3 pins is 1 | Output 0 on compare match with TGRA. |
| | | | B'1 | B'0 | Output 1 on compare match with TGRA.* |
| | | | B'1 | Toggle output on compare match with TGRA.* | |

Note: * This setting is prohibited in PWM mode.

66.2.4 Timer Interrupt Enable Register (TIER)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TIER is provided for each of timers 0 to 3 and is used to enable or disable the sources of interrupt requests for the corresponding timer. TIER is initialized to H'0000 by a reset. Bits 8 to 13 are only present in timer 0.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|------------|------------|------------|------------|------------|------------|---|---|---|-------|-------|-------|-------|-------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | TMD OFE | TMD RFE | TMS 1ER | TMS 1ET | TMS 1EA | TMS 1ES | — | — | — | TC1EV | TG1ED | TG1EC | TG1EB | TG1EA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 15, 14 | — | All 0 | R | Reserved These bits are always read as 0 and cannot be modified. |
| 13 | TMDOFE | B'0 | R/W | Motor Control Data Transfer Overflow Detection Enable This bit enables or disables interrupt request in response to the state of the TMDOFS flag in the stepping motor control mode of timer 0. This bit is reserved in timers 1 to 3, is always read as 0, and cannot be modified. 0: Interrupt request in response to the TMDOFS flag disabled 1: Interrupt request in response to the TMDOFS flag enabled |
| 12 | TMDRFE | B'0 | R/W | Motor Control Data Transfer Request Detection Enable This bit enables or disables interrupt request in response to the state of the TMDRFS flag in the stepping motor control mode of timer 0. This bit is reserved in timers 1 to 3, is always read as 0, and cannot be modified. 0: Interrupt request in response to the TMDRFS flag disabled 1: Interrupt request in response to the TMDRFS flag enabled |
| 11 | TMS1ER | B'0 | R/W | Motor Control Deceleration Transition Detection Enable This bit enables or disables interrupt request in response to the state of the TMCFR flag in the stepping motor control mode of timer 0. This bit is reserved in timers 1 to 3, is always read as 0, and cannot be modified. 0: Interrupt request in response to the TMCFR flag disabled 1: Interrupt request in response to the TMCFR flag enabled |
| 10 | TMS1ET | B'0 | R/W | Motor Control Normal Transition Detection Enable This bit enables or disables interrupt request in response to the state of the TMCFT flag in the stepping motor control mode of timer 0. This bit is reserved in timers 1 to 3, is always read as 0, and cannot be modified. 0: Interrupt request in response to the TMCFT flag disabled 1: Interrupt request in response to the TMCFT flag enabled |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 9 | TMS1EA | B'0 | R/W | <p>Motor Control Acceleration Transition Detection Enable</p> <p>This bit enables or disables interrupt request in response to the state of the TMCFA flag in the stepping motor control mode of timer 0.</p> <p>This bit is reserved in timers 1 to 3, is always read as 0, and cannot be modified.</p> <p>0: Interrupt request in response to the TMCFA flag disabled 1: Interrupt request in response to the TMCFA flag enabled</p> |
| 8 | TMS1ES | B'0 | R/W | <p>Motor Control Stop Transition Detection Enable</p> <p>This bit enables or disables interrupt request in response to the state of the TMCFS flag in the stepping motor control mode of timer 0,</p> <p>This bit is reserved in timers 1 to 3, is always read as 0, and cannot be modified.</p> <p>0: Interrupt request in response to the TMCFS flag disabled 1: Interrupt request in response to the TMCFS flag enabled</p> |
| 7 to 5 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0 and cannot be modified.</p> |
| 4 | TC1EV | B'0 | R/W | <p>Overflow Interrupt Enable</p> <p>This bit enables or disables the interrupt request in response to the state of the TCFV flag.</p> <p>0: Interrupt request in response to the TCFV flag disabled 1: Interrupt request in response to the TCFV flag enabled</p> |
| 3 | TG1ED | B'0 | R/W | <p>TGR Interrupt Enable D.</p> <p>This bit enables or disables interrupt request in response to the state of the TGFD bit.</p> <p>0: Interrupt request in response to the TGFD bit disabled 1: Interrupt request in response to the TGFD bit enabled</p> |
| 2 | TG1EC | B'0 | R/W | <p>TGR Interrupt Enable C</p> <p>This bit enables or disables interrupt request in response to the state of the TGFC bit.</p> <p>0: Interrupt request in response to the TGFC bit disabled 1: Interrupt request in response to the TGFC bit enabled</p> |
| 1 | TG1EB | B'0 | R/W | <p>TGR Interrupt Enable B</p> <p>This bit enables or disables interrupt request in response to the state of the TGFB bit.</p> <p>0: Interrupt request in response to the TGFB bit disabled 1: Interrupt request in response to the TGFB bit enabled</p> |
| 0 | TG1EA | B'0 | R/W | <p>TGR Interrupt Enable A</p> <p>This bit enables or disables interrupt request in response to the state of the TGFA bit.</p> <p>0: Interrupt request in response to the TGFA bit disabled 1: Interrupt request in response to the TGFA bit enabled</p> |

66.2.5 Timer Status Registers (TSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TSR is provided for each of timers 0 to 3 and indicates the state of the corresponding timer. TSR is initialized to H'0000 by a reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|------------|------------|-------|-------|-------|-------|---|---|---|-------|-------|-------|-------|-------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | TMD OFS | TMD RFS | TMCFR | TMCFT | TMCFA | TMCFS | — | — | — | TCFV | TGFD | TGFC | TGFB | TGFA |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R/WC0 | R/WC0 | R/WC0 | R/WC0 | R/WC0 | R/WC0 | R | R | R | R/WC0 | R/WC0 | R/WC0 | R/WC0 | R/WC0 |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-------|--|
| 15, 14 | — | All 0 | R | Reserved These bits are always read as 0 and cannot be modified. |
| 13 | TMDOFS | B'0 | R/WC0 | Motor Control Data Transfer Overflow Detection This flag indicates that a data transfer overflow has been detected in the stepping motor control mode of timer 0. This bit is reserved in timers 1 to 3, is always read as 0, and cannot be modified. [Clearing condition] Writing 0 to the TMDOFS bit after reading the TMDOFS bit while TMDOFS is set to 1 [Setting condition] When the DMA transfer overflow occurs |
| 12 | TMDRFS | B'0 | R/WC0 | Motor Control Data Transfer Request Detection This flag indicates that a data transfer request interrupt has been detected in the stepping motor control mode of timer 0. If DMA transfer is not in use, data are written to TGRD0 in response to this interrupt. This bit is reserved in timers 1 to 3, is always read as 0, and cannot be modified. [Clearing condition] Writing 0 to the TMDRFS bit after reading the TMDRFS bit while TMDRFS is set to 1 [Setting condition] When data transfer request occurs |
| 11 | TMCFR | B'0 | R/WC0 | Motor Control Deceleration Transition Detection This flag indicates that the current sequence state of timer 0 in the stepping motor control mode is the deceleration state. This bit is reserved in timers 1 to 3, is always read as 0, and cannot be modified. [Clearing condition] Writing 0 to the TMCFR bit after reading TMCFR bit = 1 [Setting condition] When the sequence state is shifted to the deceleration state |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-------|--|
| 10 | TMCFT | B'0 | R/WC0 | <p>Motor Control Normal Transition Detection</p> <p>This flag indicates that the current sequence state of timer 0 in the stepping motor control mode is the normal state.</p> <p>This bit is reserved in timers 1 to 3, is always read as 0, and cannot be modified.</p> <p>[Clearing condition] Writing 0 to the TMCFT bit after reading TMCFT bit = 1</p> <p>[Setting condition] When the sequence state is shifted to the normal state</p> |
| 9 | TMCFA | B'0 | R/WC0 | <p>Motor Control Acceleration Transition Detection</p> <p>This flag indicates that the current sequence state of timer 0 in the stepping motor control mode is the acceleration state.</p> <p>This bit is reserved in timers 1 to 3, is always read as 0, and cannot be modified.</p> <p>[Clearing condition] Writing 0 to the TMCFA bit after reading TMCFA bit = 1</p> <p>[Setting condition] When the sequence state is shifted to the acceleration state</p> |
| 8 | TMCFS | B'0 | R/WC0 | <p>Motor Control Stop Transition Detection</p> <p>This flag indicates that the current sequence state of timer 0 in the stepping motor control mode is the stop state.</p> <p>This bit is reserved in timers 1 to 3, is always read as 0, and cannot be modified.</p> <p>[Clearing condition] Writing 0 to the TMCFS bit after reading TMCFS bit = 1</p> <p>[Setting condition] When the sequence state is shifted to the stop state</p> |
| 7 to 5 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0 and cannot be modified.</p> |
| 4 | TCFV | B'0 | R/WC0 | <p>Overflow</p> <p>This flag indicates that TCNT has overflowed.</p> <p>[Clearing condition] Writing 0 to the TCFV bit after reading TCFV bit = 1</p> <p>[Setting condition] When the value of TCNT overflows (when its value changes from H'FFFF to H'0000)</p> |
| 3 | TGFD | B'0 | R/WC0 | <p>Compare Flag D</p> <p>This flag indicates that the values of TCNT and TGRD have matched (a compare match).</p> <p>[Clearing condition] Writing 0 to the TGFD bit after reading TGFD bit = 1</p> <p>[Setting condition] When the values of TCNT and TGRD have matched</p> |
| 2 | TGFC | B'0 | R/WC0 | <p>Compare Flag C</p> <p>This flag indicates that the values of TCNT and TGRC have matched (a compare match).</p> <p>[Clearing condition] Writing 0 to the TGFC bit after reading TGFC bit = 1</p> <p>[Setting condition] When the values of TCNT and TGRC have matched</p> |

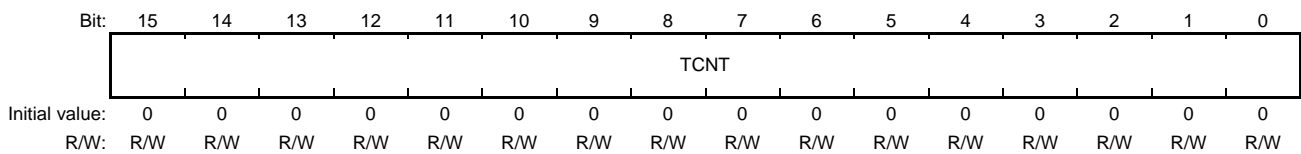
| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 1 | TGFB | B'0 | R/WC0 | <p>Compare Flag B</p> <p>This flag indicates that the values of TCNT and TGRB have matched (a compare match).</p> <p>[Clearing condition]</p> <p>Writing 0 to the TGFB bit after reading TGFB bit = 1</p> <p>[Setting condition]</p> <p>When the values of TCNT and TGRB have matched</p> |
| 0 | TGFA | B'0 | R/WC0 | <p>Output Compare Flag A</p> <p>This flag indicates that the values of TCNT and TGRA have matched (a compare match).</p> <p>[Clearing condition]</p> <p>Writing 0 to the TGFA bit after reading TGFA bit = 1</p> <p>[Setting condition]</p> <p>When the values of TCNT and TGRA are matched</p> |

66.2.6 Timer Counter (TCNT)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TCNT is a 16-bit counter, and is provided for each of timers 0 to 3.

TCNT is initialized to H'0000 by a reset.



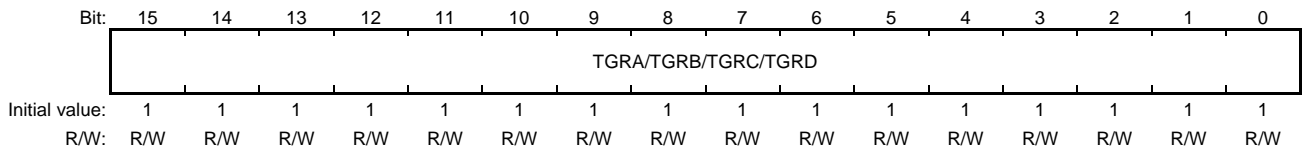
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--------------------|
| 15 to 0 | TCNT | H'0000 | R/W | Timer Counter Bits |

66.2.7 Timer General Register A to D (TGRA/TGRB/TGRC/TGRD)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TGRA, TGRB, TGRC and TGRD are 16 bits general registers that are provided for each of timers 0 to 3. TGRC and TGRD can be designated for operation as buffer registers*. TGR is initialized to H'FFFF by a reset.

Note: * The combinations of TGR registers and buffer registers are TGRA with TGRC and TGRB with TGRD.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-------------------------|---------------|-----|-----------------------------|
| 15 to 0 | TGRA/TGRB/ TGRC/TGRD | H'FFFF | R/W | Timer General Register Bits |

66.2.8 Timer Start Register (TSTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TSTR is used to start or stop the timer counter (TCNT) of timers 0 to 3.

TSTR is initialized to H'0000 by a reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|---|------|------|------|------|------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | TMST | CST3 | CST2 | CST1 | CST0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 15 to 5 | — | All 0 | R | Reserved These bits are always read as 0 and cannot be modified. |
| 4 | TMST | B'0 | R/W | Motor Control Sequence Start Selects whether to start or stop the motor control sequence. This bit is cleared to 0 when the timer has shifted to the stop state after motor control operation was started. 0: Stops motor control sequence 1: Starts motor control sequence |
| 3 | CST3 | B'0 | R/W | Counter Start Selects whether to start or stop the TCNT3 operation. 0: Stops the TCNT3 counting operation 1: Starts the TCNT3 counting operation |
| 2 | CST2 | B'0 | R/W | Counter Start Selects whether to start or stop the TCNT2 operation. 0: Stops the TCNT2 counting operation 1: Starts the TCNT2 counting operation |
| 1 | CST1 | B'0 | R/W | Counter Start Selects whether to start or stop the TCNT1 operation. 0: Stops the TCNT1 counting operation 1: Starts the TCNT1 counting operation |
| 0 | CST0 | B'0 | R/W | Counter Start Selects whether to start or stop the TCNT0 operation. 0: Stops the TCNT0 counting operation 1: Starts the TCNT0 counting operation |

66.2.9 Motor Control Setting Register (TMIR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TMIR is used to configure operation in the motor control mode.

TMIR is initialized to H'0000 by a reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|---|-------|----------------|---------------------|-------|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | TDMAE | MTRPAT DOWN | MTRPATKIND [1:0] | MTRON | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------|---------------|-----|---|
| 15 to 5 | — | All 0 | R | Reserved These bits are always read as 0 and cannot be modified. |
| 4 | TDMAE | B'0 | R/W | Selection of DMA Use Selects whether to use DMA for step interval input (writing to TGRD0) in the motor control mode. 0: DMA is not used. 1: DMA is used. |
| 3 | MTRPAT DOWN | B'0 | R/W | Operation and Stop Patterns Transition Ascending/Descending Select Selects the direction of transitions within operation and stop patterns as ascending or descending in the motor control mode. 0: Ascending 1: Descending |
| 2, 1 | MTRPAT KIND[1:0] | B'00 | R/W | Operation and Stop Pattern Type Select Selects the number of patterns in the cycles of stop and start patterns in the motor control mode. B'00: Four B'01: Eight B'10: Setting prohibited B'11: Setting prohibited |
| 0 | MTRON | B'0 | R/W | TPU Mode/Stepping Motor Control Mode Select 0: TPU mode 1: Stepping motor control mode Note: This bit also controls a selecting signal to switch TPU0TO0 to TPU0TO3 outputs. Set this bit to 1 after making initial settings for the pattern cycle. Do not modify this bit during the counter operation. |

66.2.10 Motor Deceleration (Stop) Transition Register (TMRR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TMRR is used to place timer 0 in to the deceleration state from the normal state, in the motor control mode.

TMRR is initialized to H'0000 by a reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|---|---|---|---|-------------|------------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | REDU ON0 | REDU ON |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 15 to 2 | — | All 0 | R | Reserved These bits are always read as 0 and cannot be modified. |
| 1 | REDUON0 | B'0 | R/W | This bit is used to force a transition of timer 0 to the deceleration (or stop) state from the normal state in motor control mode. Note that the state transition only proceeds when pattern 0 is the currently specified pattern. Setting this bit has no effect if timer 0 is not in the normal state. This bit is automatically cleared to 0 when the state of timer 0 shifts to the stop state. 0: No operation 1: Timer 0 is forcibly placed in the deceleration (or stop) state from the normal state. |
| 0 | REDUON | B'0 | R/W | This bit is used to force a transition of timer 0 to the deceleration (or stop) state from the normal state in motor control mode. Setting this bit has no effect if timer 0 is not in the normal state. This bit is automatically cleared to 0 when the state of timer 0 shifts to the stop state. 0: No operation 1: Timer 0 is forcibly placed in the deceleration (or stop) state from the normal state. |

66.2.11 Motor Control Status Register (TMSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TMSR is a read-only register that indicates the sequence state in motor control mode.

TMSR is initialized to H'0001 by a reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|---|---|------|------|------|------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | SITR | SITT | SITA | SITS |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 15 to 4 | — | All 0 | R | Reserved These bits are always read as 0 and cannot be modified. |
| 3 | SITR | B'0 | R | This bit indicates the sequence state in motor control mode. 1: The sequence state is deceleration. 0: The sequence state is not deceleration. |
| 2 | SITT | B'0 | R | This bit indicates the sequence state in motor control mode. 1: The sequence state is normal. 0: The sequence state is not normal. |
| 1 | SITA | B'0 | R | This bit indicates the sequence state in motor control mode. 1: The sequence state is acceleration. 0: The sequence state is not acceleration. |
| 0 | SITS | B'1 | R | This bit indicates the sequence state in motor control mode. 1: The sequence state is stop. 0: The sequence state is not stop. |

66.2.12 Motor Operation Pattern Storing Register 0 (TMMPR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TMMPR0 is used to set motor operation patterns [3] to [0] in motor control mode.

TMMPR0 is initialized to H'0000 by a reset.

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MP33 | MP32 | MP31 | MP30 | MP23 | MP22 | MP21 | MP20 | MP13 | MP12 | MP11 | MP10 | MP03 | MP02 | MP01 | MP00 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 15 | MP33 | B'0 | R/W | Motor Operation Pattern [3] in Motor Control Mode |
| 14 | MP32 | B'0 | R/W | MP33: TPU0TO3 output value in motor operation pattern [3] |
| 13 | MP31 | B'0 | R/W | MP32: TPU0TO2 output value in motor operation pattern [3] |
| 12 | MP30 | B'0 | R/W | MP31: TPU0TO1 output value in motor operation pattern [3] MP30: TPU0TO0 output value in motor operation pattern [3] |
| 11 | MP23 | B'0 | R/W | Motor Operation Pattern [2] in Motor Control Mode |
| 10 | MP22 | B'0 | R/W | MP23: TPU0TO3 output value in motor operation pattern [2] |
| 9 | MP21 | B'0 | R/W | MP22: TPU0TO2 output value in motor operation pattern [2] |
| 8 | MP20 | B'0 | R/W | MP21: TPU0TO1 output value in motor operation pattern [2] MP20: TPU0TO0 output value in motor operation pattern [2] |
| 7 | MP13 | B'0 | R/W | Motor Operation Pattern [1] in Motor Control Mode |
| 6 | MP12 | B'0 | R/W | MP13: TPU0TO3 output value in motor operation pattern [1] |
| 5 | MP11 | B'0 | R/W | MP12: TPU0TO2 output value in motor operation pattern [1] |
| 4 | MP10 | B'0 | R/W | MP11: TPU0TO1 output value in motor operation pattern [1] MP10: TPU0TO0 output value in motor operation pattern [1] |
| 3 | MP03 | B'0 | R/W | Motor Operation Pattern [0] in Motor Control Mode |
| 2 | MP02 | B'0 | R/W | MP03: TPU0TO3 output value in motor operation pattern [0] |
| 1 | MP01 | B'0 | R/W | MP02: TPU0TO2 output value in motor operation pattern [0] |
| 0 | MP00 | B'0 | R/W | MP01: TPU0TO1 output value in motor operation pattern [0] MP00: TPU0TO0 output value in motor operation pattern [0] |

66.2.13 Motor Operation Pattern Storing Register 1 (TMMPR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TMMPR1 is used to set motor operation pattern [7] to [4] in motor control mode.

TMMPR1 is initialized to H'0000 by a reset.

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | MP73 | MP72 | MP71 | MP70 | MP63 | MP62 | MP61 | MP60 | MP53 | MP52 | MP51 | MP50 | MP43 | MP42 | MP41 | MP40 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 15 | MP73 | B'0 | R/W | Motor Operation Pattern [7] in Motor Control Mode |
| 14 | MP72 | B'0 | R/W | MP73: TPU0TO3 output value in motor operation pattern [7] |
| 13 | MP71 | B'0 | R/W | MP72: TPU0TO2 output value in motor operation pattern [7] |
| 12 | MP70 | B'0 | R/W | MP71: TPU0TO1 output value in motor operation pattern [7] MP70: TPU0TO0 output value in motor operation pattern [7] |
| 11 | MP63 | B'0 | R/W | Motor Operation Pattern [6] in Motor Control Mode |
| 10 | MP62 | B'0 | R/W | MP63: TPU0TO3 output value in motor operation pattern [6] |
| 9 | MP61 | B'0 | R/W | MP62: TPU0TO2 output value in motor operation pattern [6] |
| 8 | MP60 | B'0 | R/W | MP61: TPU0TO1 output value in motor operation pattern [6] MP60: TPU0TO0 output value in motor operation pattern [6] |
| 7 | MP53 | B'0 | R/W | Motor Operation Pattern [5] in Motor Control Mode |
| 6 | MP52 | B'0 | R/W | MP53: TPU0TO3 output value in motor operation pattern [5] |
| 5 | MP51 | B'0 | R/W | MP52: TPU0TO2 output value in motor operation pattern [5] |
| 4 | MP50 | B'0 | R/W | MP51: TPU0TO1 output value in motor operation pattern [5] MP50: TPU0TO0 output value in motor operation pattern [5] |
| 3 | MP43 | B'0 | R/W | Motor Operation Pattern [4] in Motor Control Mode |
| 2 | MP42 | B'0 | R/W | MP43: TPU0TO3 output value in motor operation pattern [4] |
| 1 | MP41 | B'0 | R/W | MP42: TPU0TO2 output value in motor operation pattern [4] |
| 0 | MP40 | B'0 | R/W | MP41: TPU0TO1 output value in motor operation pattern [4] MP40: TPU0TO0 output value in motor operation pattern [4] |

66.2.14 Motor Stop Pattern Storing Register 0 (TMSPR0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TMSPR0 is used to set motor stop pattern (3) to (0) in motor control mode.

TMSPR0 is initialized to H'0000 by a reset.

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SP33 | SP32 | SP31 | SP30 | SP23 | SP22 | SP21 | SP20 | SP13 | SP12 | SP11 | SP10 | SP03 | SP02 | SP01 | SP00 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 15 | SP33 | B'0 | R/W | Motor Stop Pattern (3) in Motor Control Mode |
| 14 | SP32 | B'0 | R/W | SP33: TPU0TO3 output value in motor stop pattern (3) |
| 13 | SP31 | B'0 | R/W | SP32: TPU0TO2 output value in motor stop pattern (3) |
| 12 | SP30 | B'0 | R/W | SP31: TPU0TO1 output value in motor stop pattern (3) SP30: TPU0TO0 output value in motor stop pattern (3) |
| 11 | SP23 | B'0 | R/W | Motor Stop Pattern (2) in Motor Control Mode |
| 10 | SP22 | B'0 | R/W | SP23: TPU0TO3 output value in motor stop pattern (2) |
| 9 | SP21 | B'0 | R/W | SP22: TPU0TO2 output value in motor stop pattern (2) |
| 8 | SP20 | B'0 | R/W | SP21: TPU0TO1 output value in motor stop pattern (2) SP20: TPU0TO0 output value in motor stop pattern (2) |
| 7 | SP13 | B'0 | R/W | Motor Stop Pattern (1) in Motor Control Mode |
| 6 | SP12 | B'0 | R/W | SP13: TPU0TO3 output value in motor stop pattern (1) |
| 5 | SP11 | B'0 | R/W | SP12: TPU0TO2 output value in motor stop pattern (1) |
| 4 | SP10 | B'0 | R/W | SP11: TPU0TO1 output value in motor stop pattern (1) SP10: TPU0TO0 output value in motor stop pattern (1) |
| 3 | SP03 | B'0 | R/W | Motor Stop Pattern (0) in Motor Control Mode |
| 2 | SP02 | B'0 | R/W | SP03: TPU0TO3 output value in motor stop pattern (0) |
| 1 | SP01 | B'0 | R/W | SP02: TPU0TO2 output value in motor stop pattern (0) |
| 0 | SP00 | B'0 | R/W | SP01: TPU0TO1 output value in motor stop pattern (0) SP00: TPU0TO0 output value in motor stop pattern (0) |

66.2.15 Motor Stop Pattern Storing Register 1 (TMSPR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TMSPR1 is used to set motor stop pattern (7) to (4) in motor control mode.

TMSPR1 is initialized to H'0000 by a reset.

| | | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | SP73 | SP72 | SP71 | SP70 | SP63 | SP62 | SP61 | SP60 | SP53 | SP52 | SP51 | SP50 | SP43 | SP42 | SP41 | SP40 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 15 | SP73 | B'0 | R/W | Motor Stop Pattern (7) in Motor Control Mode |
| 14 | SP72 | B'0 | R/W | SP73: TPU0TO3 output value in motor stop pattern (7) |
| 13 | SP71 | B'0 | R/W | SP72: TPU0TO2 output value in motor stop pattern (7) |
| 12 | SP70 | B'0 | R/W | SP71: TPU0TO1 output value in motor stop pattern (7) SP70: TPU0TO0 output value in motor stop pattern (7) |
| 11 | SP63 | B'0 | R/W | Motor Stop Pattern (6) in Motor Control Mode |
| 10 | SP62 | B'0 | R/W | SP63: TPU0TO3 output value in motor stop pattern (6) |
| 9 | SP61 | B'0 | R/W | SP62: TPU0TO2 output value in motor stop pattern (6) |
| 8 | SP60 | B'0 | R/W | SP61: TPU0TO1 output value in motor stop pattern (6) SP60: TPU0TO0 output value in motor stop pattern (6) |
| 7 | SP53 | B'0 | R/W | Motor Stop Pattern (5) in Motor Control Mode |
| 6 | SP52 | B'0 | R/W | SP53: TPU0TO3 output value in motor stop pattern (5) |
| 5 | SP51 | B'0 | R/W | SP52: TPU0TO2 output value in motor stop pattern (5) |
| 4 | SP50 | B'0 | R/W | SP51: TPU0TO1 output value in motor stop pattern (5) SP50: TPU0TO0 output value in motor stop pattern (5) |
| 3 | SP43 | B'0 | R/W | Motor Stop Pattern (4) in Motor Control Mode |
| 2 | SP42 | B'0 | R/W | SP43: TPU0TO3 output value in motor stop pattern (4) |
| 1 | SP41 | B'0 | R/W | SP42: TPU0TO2 output value in motor stop pattern (4) |
| 0 | SP40 | B'0 | R/W | SP41: TPU0TO1 output value in motor stop pattern (4) SP40: TPU0TO0 output value in motor stop pattern (4) |

66.2.16 Motor Output Pattern Storing Register (TMOPR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TMOPR is used to specify the motor output pattern in motor control mode. When read, it indicates the pattern currently being output. Writing to this register while the sequence is in the stop state leads to output of the pattern with the corresponding number. TMOPR is initialized to H'0000 by a reset.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|---|---|---|-------------|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | NOWPAT[2:0] | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|--|
| 15 to 3 | — | All 0 | R | Reserved These bits are always read as 0 and cannot be modified. |
| 2 to 0 | NOWPAT [2:0] | B'000 | R/W | In motor control mode, Reading: The pattern number currently being output is read. Writing: If the sequence is in the stop state, the pattern corresponding to the written pattern number is output. |

66.2.17 Motor Acceleration Step Count Register (TMASR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TMASR is used to set the number of steps in the acceleration state of motor control mode. TMASR is initialized to H'0000 by a reset.

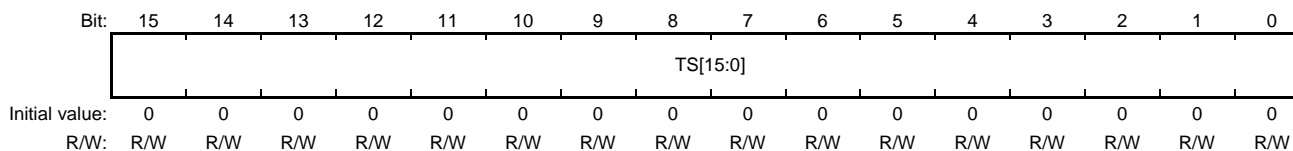
| | | | | | | | | | | | | | | | | |
|----------------|----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | AS[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 15 to 0 | AS[15:0] | H'0000 | R/W | Specify the number of steps in the acceleration state of motor control mode. When the setting is H'0000, the acceleration state is skipped. |

66.2.18 Motor Normal the Number of Steps Register (TMTSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TMTSR is used to set the number of steps in the normal state of motor control mode. TMTSR is initialized to H'0000 by a reset.

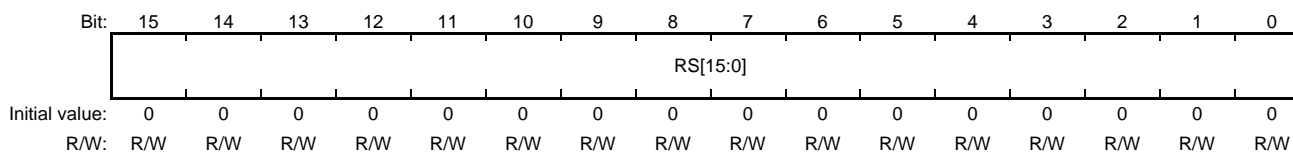


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 15 to 0 | TS[15:0] | H'0000 | R/W | Specify the number of steps in the normal state of motor control mode. When the setting is H'0000, the normal state is maintained until 1 is written to the REDUON0 bit or REDUON bit in TMRR. |

66.2.19 Motor Deceleration the Number of Steps Register (TMRSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TMRSR is used to set the number of steps in the deceleration state of motor control mode. TMRSR is initialized to H'0000 by a reset.



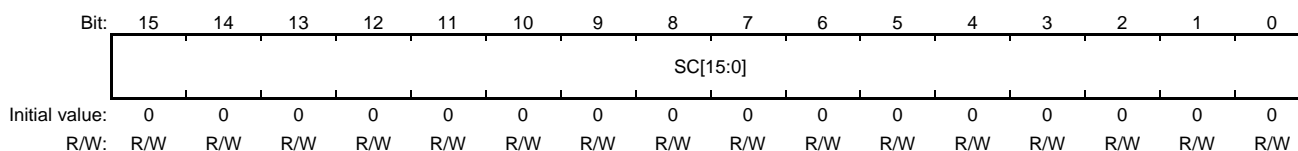
| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 15 to 0 | RS[15:0] | H'0000 | R/W | Specify the number of steps in the deceleration state of motor control mode. When the setting is H'0000, the deceleration state is skipped. |

66.2.20 Motor Control Sequence Counter Register (TMSCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TMSCR indicates or controls the value of the sequence counter in motor control mode. Writing to this register is only allowed when the motor control mode is in use and the current state is the normal state.

TMSCR is initialized to H'0000 by a reset.

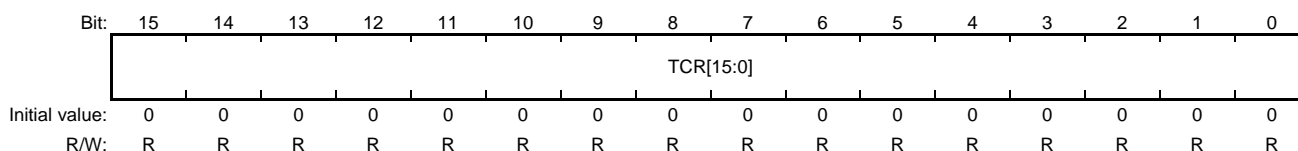


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 15 to 0 | SC[15:0] | H'0000 | R/W | Value of Sequence Counter in Motor Control Mode Reading: The value of the sequence counter is read. Writing: Writing is only allowed in the normal state. The written value is directly stored in the sequence counter. |

66.2.21 Motor Control Normal Counter Register (TMTCR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TMTCR is a read-only register that indicates the number of steps in the normal state of motor control mode. TMTCR is initialized to H'0000 by a reset.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|---|
| 15 to 0 | TCR[15:0] | H'0000 | R | Indicate the number of steps in the normal state of motor control mode. |

66.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

66.3.1 Overview

An overview of operation in the various modes is given below.

(1) Ordinary Operation

Each timer has TCNT and TGR registers. TCNT counts up, and is capable of free-running operation and counting to define periods corresponding to register settings.

(a) Buffered Operation

When a compare match occurs, the buffer register value in the corresponding timer is transferred to TGR. Either a compare match or clearing of the counter can be selected as the trigger that defines the timing of updating from a buffer register.

(b) PWM Mode

In PWM mode, a PWM waveform is output. The duty cycle can be set by TGR. PWM waveforms with duty cycles in the range from 0 to 100% can be output in accord with the settings of TGRA and TGRB.

(c) Stepping Motor Control Mode

Pre-defined patterns can be output from TPU0TO3 to TPU0TO0 by placing timer 0 to stepping motor control mode.

66.3.2 Basic Functions

(1) Counter Operation

When a bit from among CST3 to CST0 in TSTR is set to 1, the TCNT for the corresponding timer starts counting. A TCNT can operate as a free-running counter, periodic counter, and so on.

(a) Example of Procedure for Setting up Counter Operations

Figure 66.2 shows an example of the procedure for setting up counter operation.

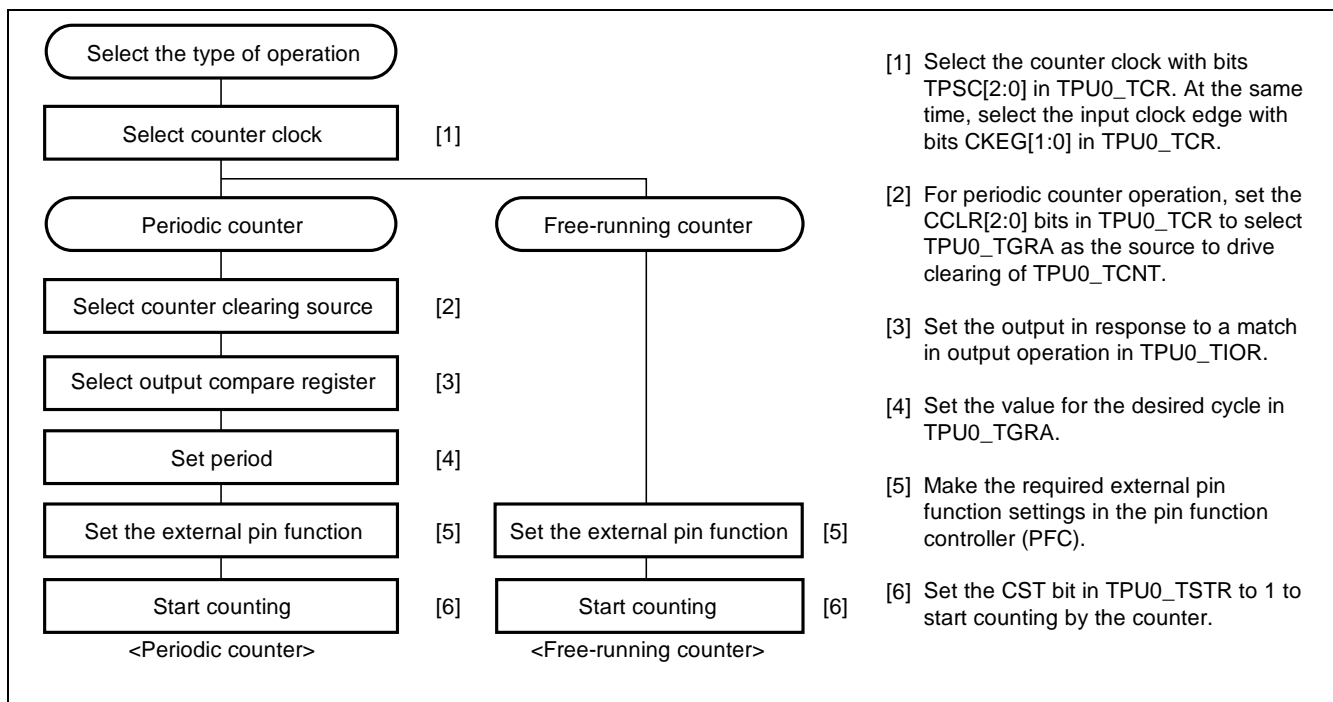


Figure 66.2 Example of Procedure for Setting up Counter Operation

(b) Free-Running Counter Operation and Periodic Counter Operation

Immediately after a reset, the TCNT counters are all designated as free-running counters. When the relevant bit in TSTR is set to 1, the corresponding TCNT starts counting up as a free-running counter. When TCNT has overflowed (when its value changes from H'FFFF to H'0000), the TCFV bit in TSR is set to 1. After it overflows, TCNT starts counting up again from H'0000.

Figure 66.3 illustrates free-running counter operation.

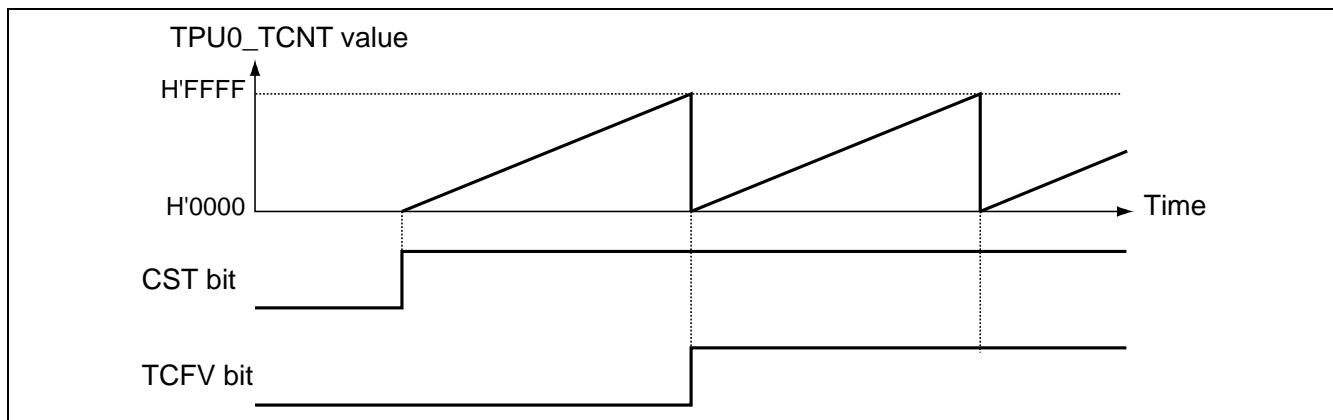


Figure 66.3 Free-Running Counter Operation

When a compare match is selected as the source to drive clearing of TCNT, the TCNT counter for the relevant timer operates as a periodic counter (counting a period defined by a general register). The TGR register for setting the period is designated as the output compare register, and counter clearing in response to a compare match is selected by means of bits CCLR[2:0] in TCR. After the settings have been made, TCNT starts counting as a periodic counter when the corresponding bit in TSTR is set to 1. When the counted value matches the value in the selected TGR, the TGF bit in TSR is set to 1 and TCNT is cleared to H'0000.

After a compare match, TCNT starts counting up again from H'0000.

Figure 66.4 illustrates periodic counter operation.

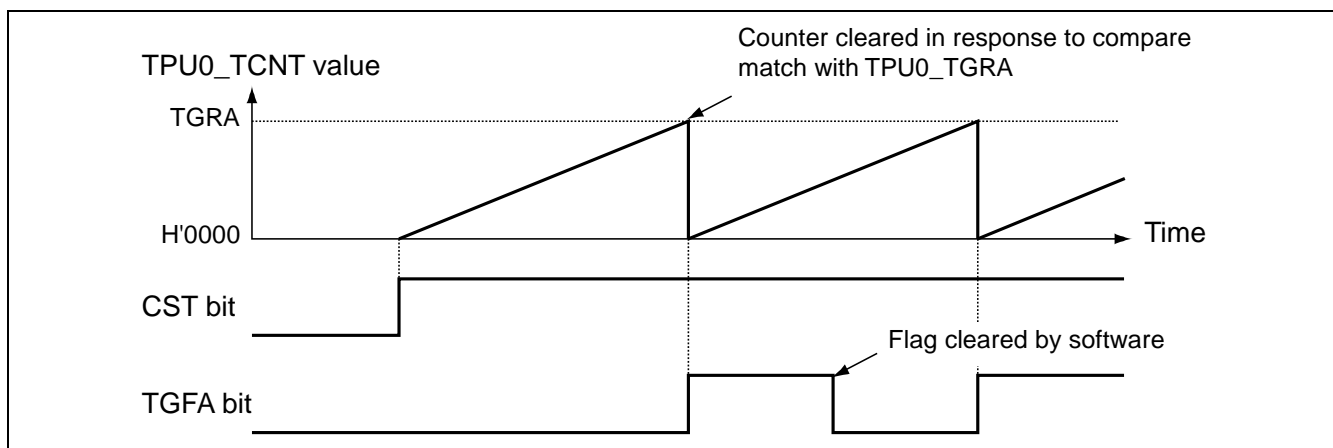


Figure 66.4 Periodic Counter Operation

(2) Waveform Output by Compare Match

The TPU can be set for the output of logical 0 or 1, or toggling of the output on the output pin in response to a compare match with TGRA.

(a) Example of Procedure for Setting up Waveform Output under Compare Match Control

Figure 66.5 shows an example of the procedure for setting up waveform output under compare match control.

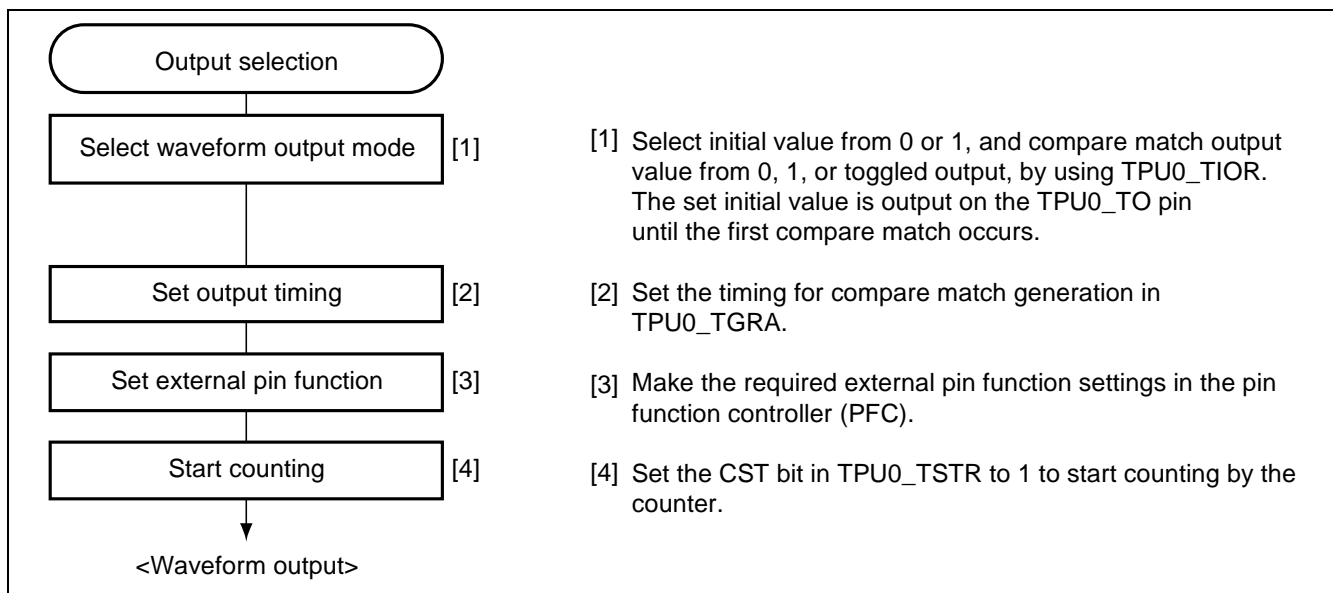


Figure 66.5 Example of Procedure for Setting up Waveform Output under Compare Match Control

(b) Examples of Waveform Output Operation

Figure 66.6 shows an example of the output of 0 or 1.

In this example, TCNT has been designated as a free-running counter, and settings have been made so that 1 or 0 is output in response to compare match A. When the set level is the same as the level on the pin, the level on the pin does not change.

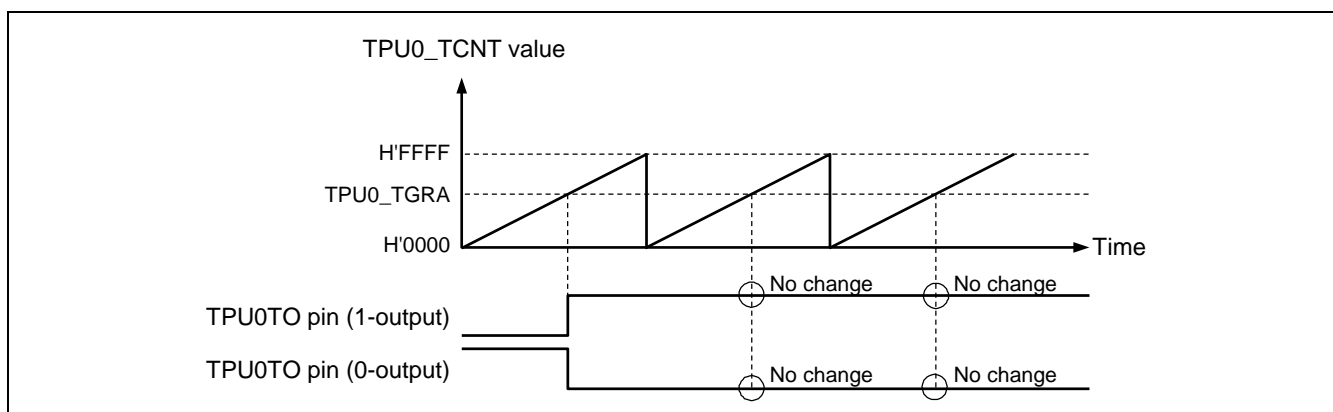


Figure 66.6 Example of Output of 0 or 1

Figure 66.7 shows an example of toggled output.

In this example, TCNT has been designated as a periodic counter (with counter clearing handled by compare match B), and settings have been made so that the output is toggled in response to compare match A.

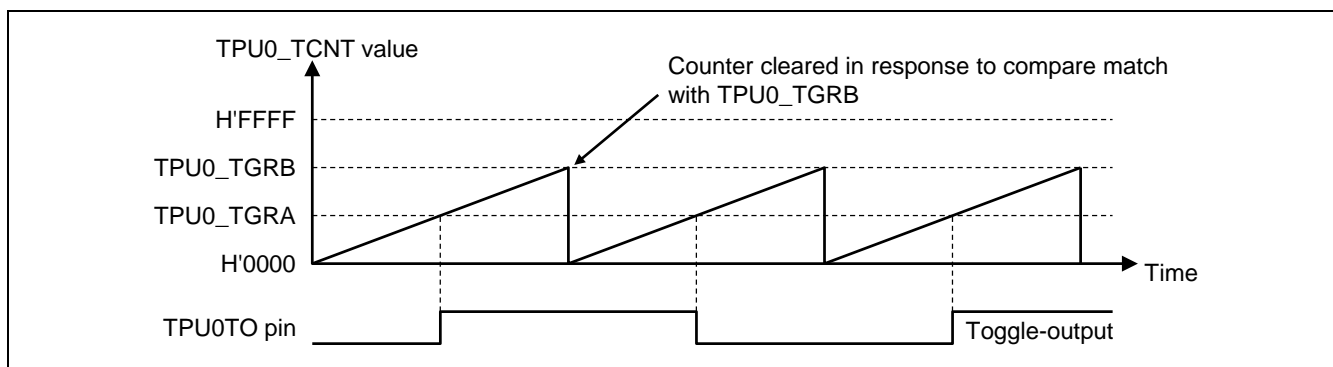


Figure 66.7 Example of Toggled Output Operation

66.3.3 Buffered Operation

TGRC and TGRD can be used as buffer registers.

Table 66.8 shows the register combinations used in buffered operation.

Table 66.8 Register Combinations in Buffered Operation

| Timer General Register | Buffer Register |
|------------------------|-----------------|
| TGRA | TGRC |
| TGRB | TGRD |

When a compare match occurs, the value in the buffer register for the corresponding timer is transferred to the timer general register. Either a compare match or clearing of the counter can be selected as the trigger that defines the timing of updating from a buffer register.

This operation is illustrated in Figure 66.8.

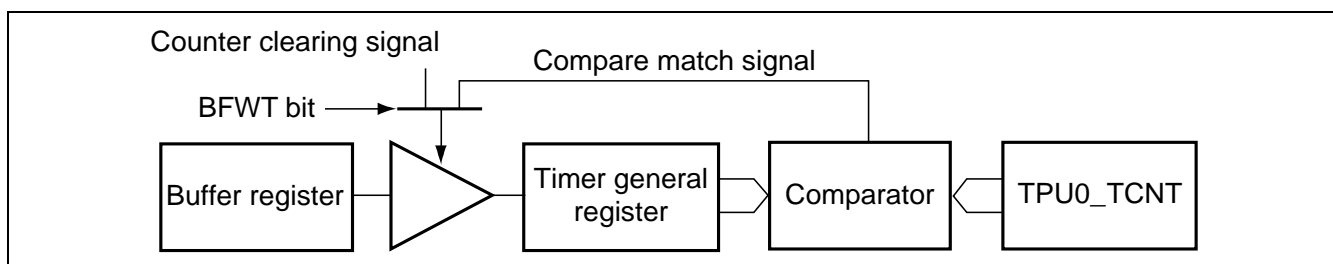


Figure 66.8 Buffered Compare Match Operation

(1) Example of Procedure for Setting up Buffered Operation

Figure 66.9 shows an example of the procedure for setting up buffered operation.

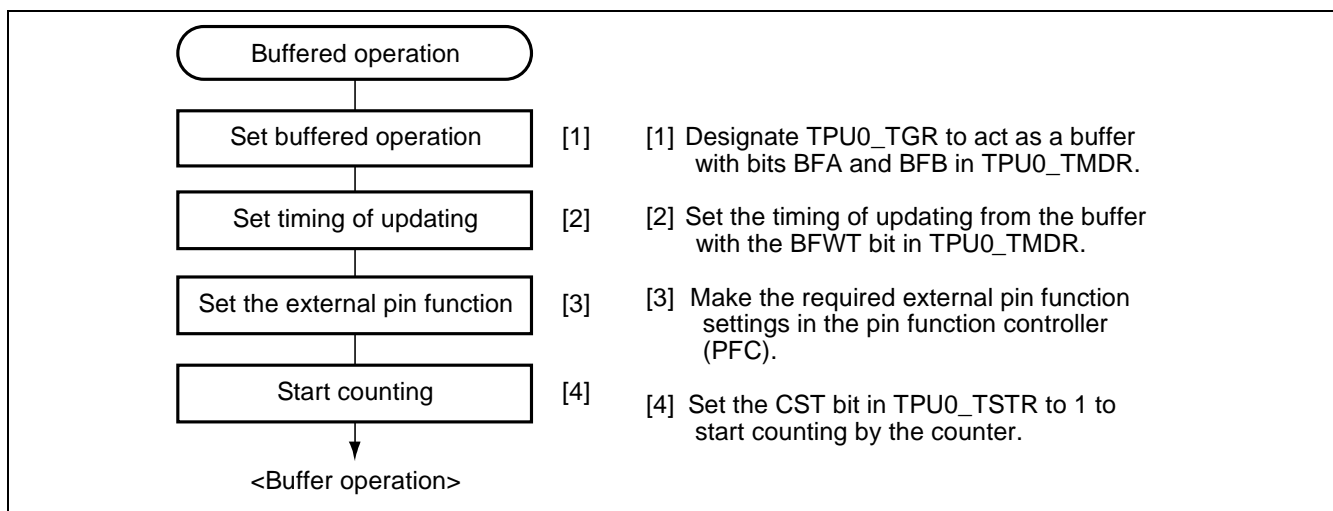


Figure 66.9 Example of Procedure for Setting up Buffered Operation

(2) Examples of Buffered Operation

Figure 66.10 shows an example of operation in which PWM mode has been designated for timer 0, and buffered operation has been designated for TGRA and TGRC. The settings used in this example are for clearing of TCNT in response to compare match B, output of 1 (on the TPU0TO0 to 3 for the selected timer) on compare match A, output of 0 as the initial value by clearing the counter, and the timing for updating from the buffer register as clearing of the counter.

When compare match A occurs, the output changes. When counter clearing is generated by a match with TGRB, the output changes and the value in buffer register TGRC is simultaneously transferred to the timer general register TGRA. This operation is repeated every time compare match A occurs.

For details on the PWM mode, see section 66.3.4, PWM Mode.

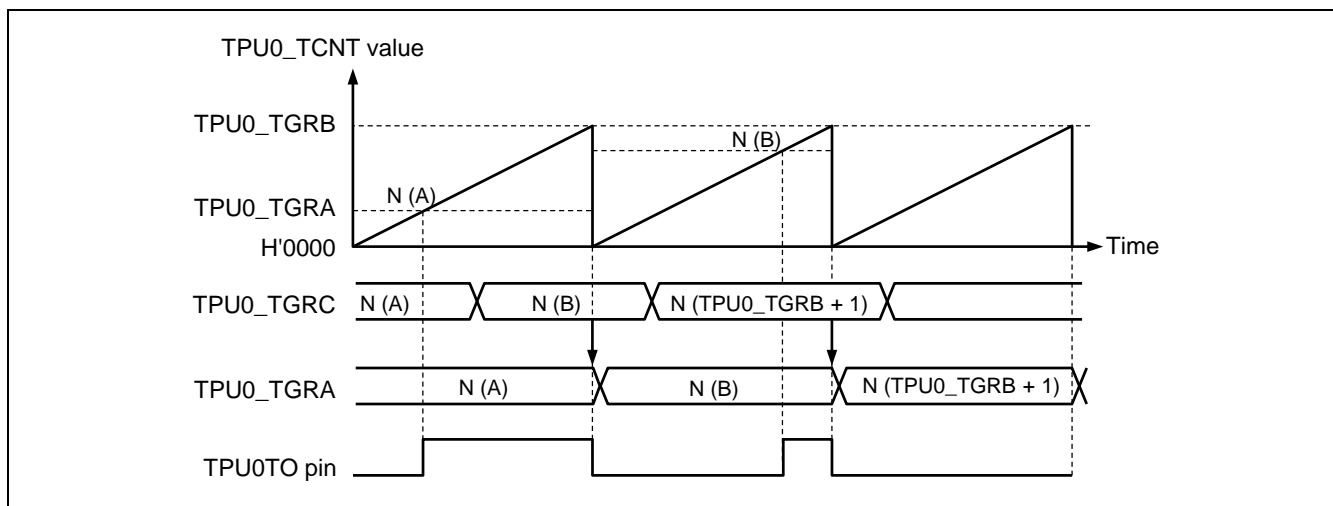


Figure 66.10 Example of Buffer Operation

66.3.4 PWM Mode

In PWM mode, PWM waveforms are output from the output pins. The output of 0 or 1 can be selected as the response to a compare match with the given TGRA.

Designating TGRB compare match as the counter clearing source enables setting of the overall cycle in that register. All timers can be independently placed in PWM mode.

PWM output is generated from the output pins by using TGRA and TGRB to control the duty cycle and period, respectively. The initial output specified by TIOR is output on the output pin by counter clearing in response to a compare match with the period register. Be sure to set TIOR so that the initial output level is different from the compare match output. Selecting the same level or toggled output leads to no operation.

Conditions for 0% and 100% duty cycles are shown below.

- 0% duty cycle: The setting of the duty-cycle register (TGRA) being equal to that of the period register plus one (TGRB + 1)
- 100% duty cycle: The setting of the duty-cycle register (TGRA) being 0

(1) Example of Procedure for Setting up PWM Mode Operation

Figure 66.11 shows an example of the procedure for setting up the PWM mode operation.

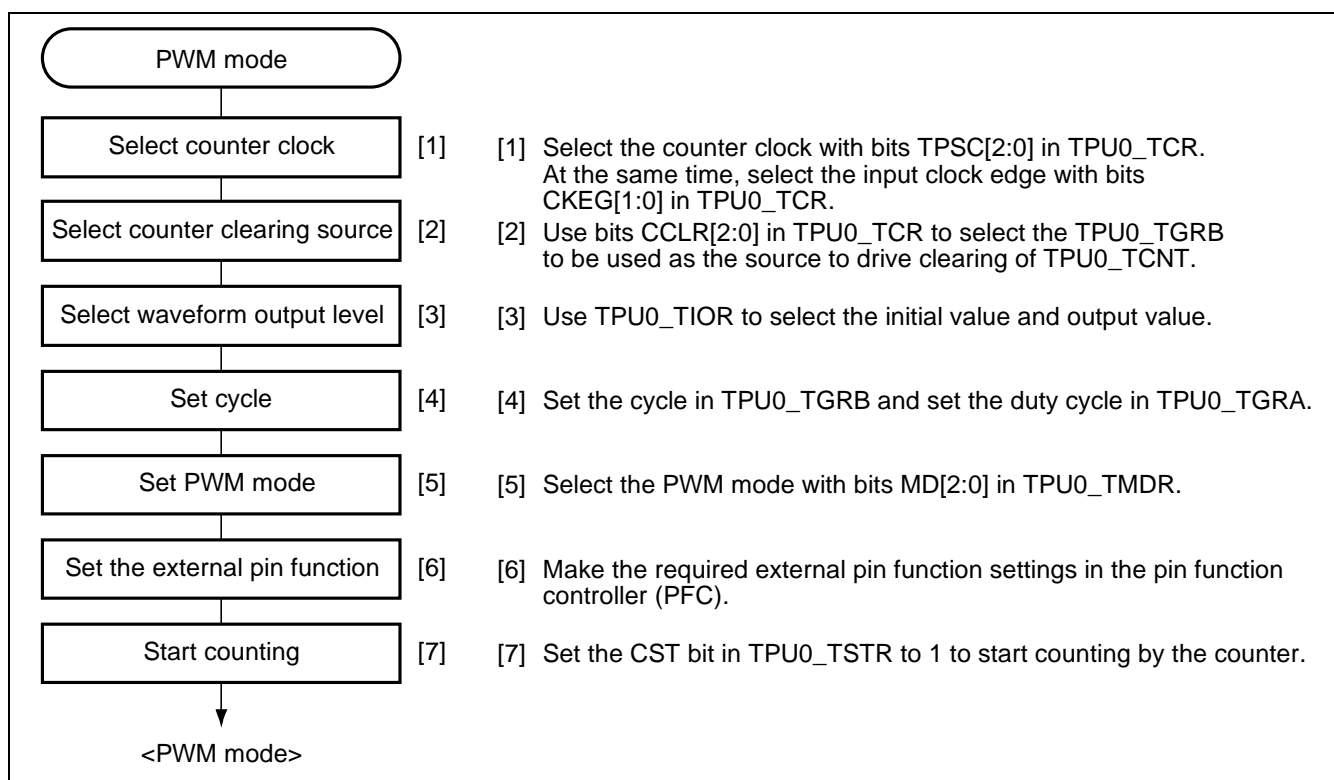


Figure 66.11 Example of Procedure for Setting up PWM Mode Operation

(2) Examples of PWM Mode Operation

Figure 66.12 shows an example of PWM mode operation.

In this example, compare match with TGRB is set as the source to drive clearing of TCNT, 0 is set as the initial output value for TGRA, and 1 is set as the output value on a compare match with TGRA.

In this case, the value set in TGRB defines the cycle time (period), and the value set in TGRA defines the duty cycle.

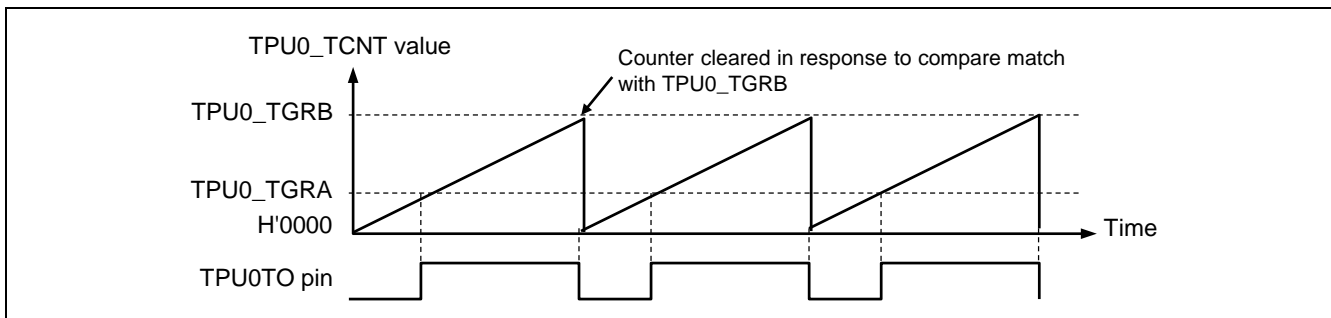


Figure 66.12 Example of PWM Mode Operation (1)

Figure 66.13 shows an example of PWM waveform output with 0% and 100% duty cycles in PWM mode.

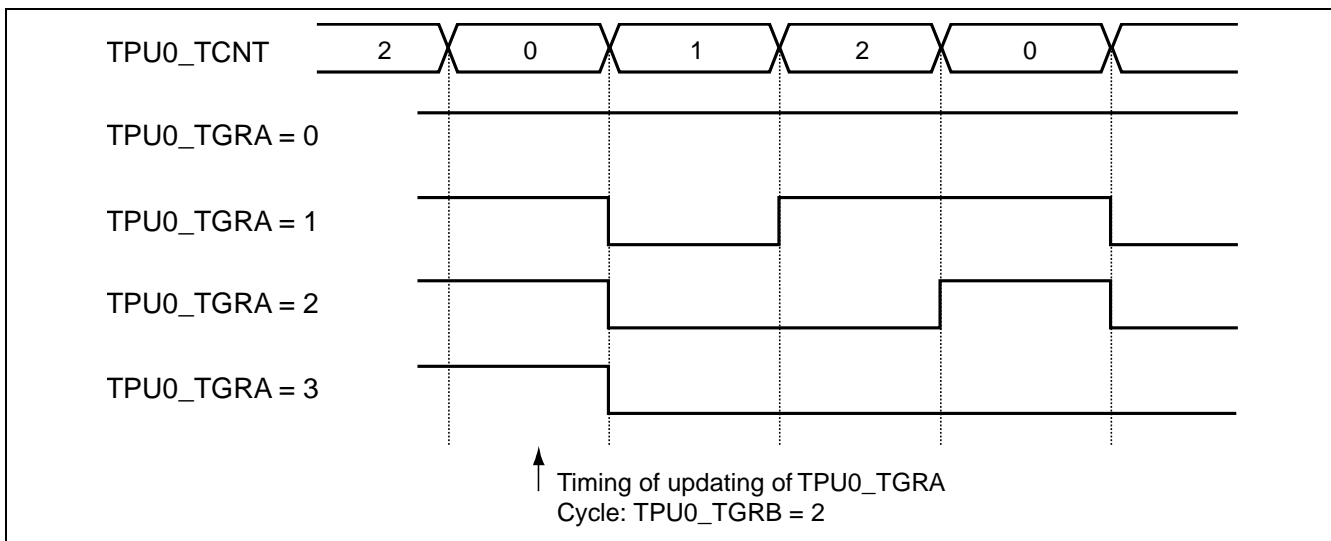


Figure 66.13 Example of PWM Mode Operation (2)

66.3.5 Stepping-Motor Control Mode

(1) Features of Stepping-Motor Control Mode

- This mode is for the output of signals for stepping-motor control on four pins and is only available for timer 0.
- States of the output pins change in response to compare matches with the counter of timer 0.
- By using the DMAC or an interrupt to rewrite the values of the compare-match registers, the intervals between transitions of the states of the output pins can be specified as desired.
- Specified patterns are output on the output pins.
- Four or eight patterns can be selected.
- The pattern-number counter is incremented or decremented by 1 on every compare match with the counter on timer 0. When the pattern-number counter overflows in normal mode, its value returns to 0. When the pattern-number counter underflows, its value becomes 3 (for a four-pattern cycle) or 7 (for an eight-pattern cycle).
- Patterns consist of stop patterns and operating patterns. Operating patterns are output while the counter is running, and stop patterns are output while the counter is stopped.
- While the counter is running, operation is in three states (the acceleration, deceleration, and normal states). Only for acceleration and deceleration, the interval between changes of pattern can be controlled.
- The acceleration or deceleration state can be skipped by writing H'0000 to TMASR and TMRSR.
- TMRR can be used to force a transition from the normal to the deceleration state.

(2) Glossary

| | |
|--------------------|--|
| Sequence State: | The four states in the sequence, i.e. the acceleration, normal, deceleration, and stop states |
| Sequence Counter: | Counter to control transitions to next sequence states The sequence counter is decremented with every compare match with the counter of timer 0 and, when the value reaches 1, the control state will make the transition to the next state in the sequence on a further compare match with the counter of timer 0. |
| Auto Mode: | Mode of operation when a value other than H'0000 has been written to TMTSR The number of pattern transitions in the normal state corresponding to the setting of TMTSR, and this is followed by an automatic transition to the deceleration state (or, if TMRSR = H'0000, a transition to the stop state). |
| Manual Mode: | Mode of operation when H'0000 has been written to TMTSR. In this case, the transition from the normal to the deceleration state (or the stop state if TMRSR = H'0000) is not made unless 1 is written to the REDUON0 and REDUON bits in TMRR. |
| Pulse Mode: | Mode of operation when H'0000 has been written to TMASR and TMRSR. Operation is only in the normal state, so the output pattern is changed with a constant interval. |
| Step: | The pattern is changed with every compare match with the counter of timer 0. Step refers to the interval of time between the change to one pattern and the change to the next. |
| Operation Pattern: | The four-bit patterns for output from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the acceleration, normal, and deceleration states. 8 types of Operation Pattern are set by TMMPR0 and TMMPR1. “[]” in Figure 66.14, 66.16, 66.18, 66.20 to 66.23 and Table 66.12 means 8 types of Operation Pattern set by TMMPR0 and TMMPR1. |
| Stop Pattern: | The four-bit patterns for output from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the stop state. 8 types of Operation Pattern are set by TMSPR0 and TMSPR1. “()” in Figure 66.14, 66.16, 66.18, 66.20 to 66.23 and Table 66.12 means 8 types of Operation Pattern set by TMMPR0 and TMMPR1. |

(3) Setting up Basic Motor Control Operation

An example of operation in stepping motor control is shown in Figure 66.14.

Stepping-motor control mode allows the output of patterns with a variable step interval.

“()” in Figure 66.14 means Stop Pattern outputted from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the stop state set by TMSPR0 and TMSPR1.

“[]” in Figure 66.14 means Operation Pattern outputted from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the acceleration, normal, and deceleration states set by TMMPR0 and TMMPR1.

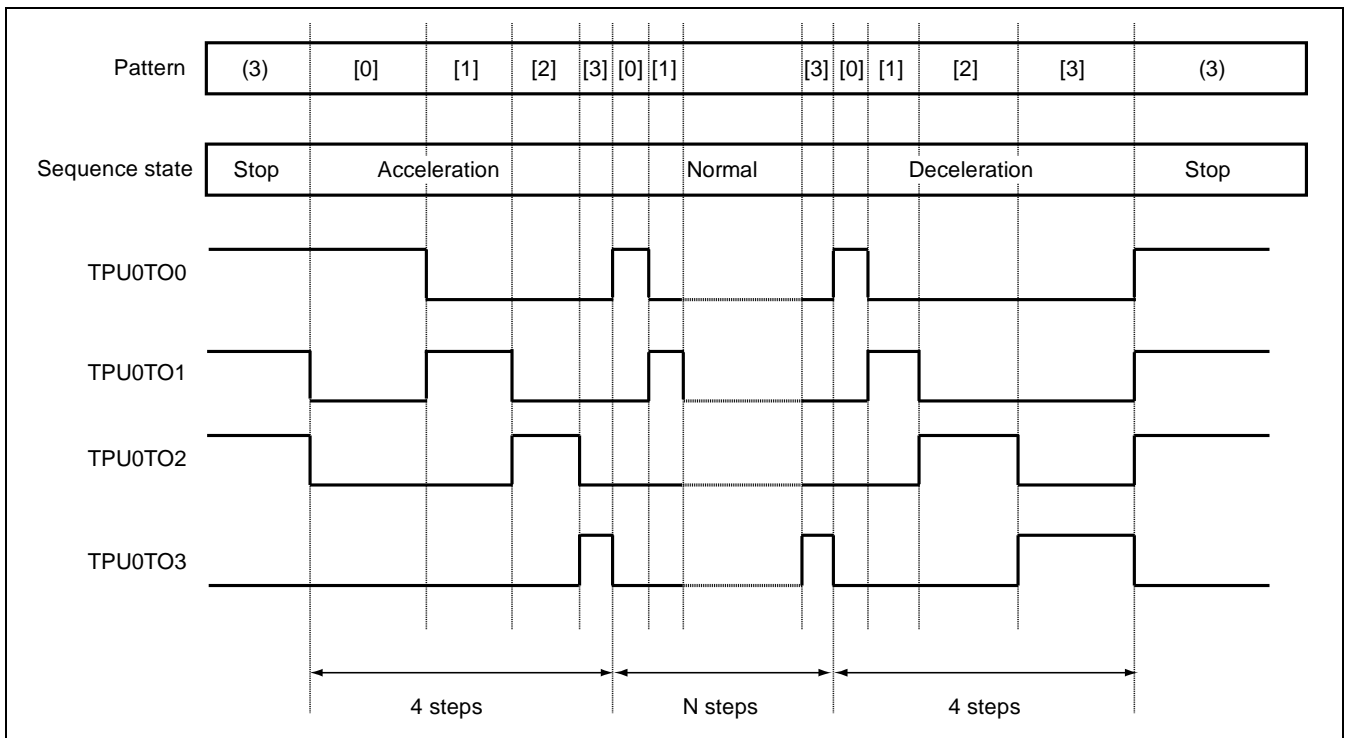


Figure 66.14 Example of Stepping-Motor Control

(a) Compare Match Settings for Timer 0

For stepping motor control, set TCR0 and TMDR0 as follows:

Table 66.9 Compare Match Settings for Timer 0

| Register | Setting | Description |
|----------|-----------------------|--|
| TCR0 | B'0000_0000_0100_0xxx | CCLR[2:0] bits = B'010 CKEG[1:0] bits = B'00 TPSC[2:0] bits = B'xxx (optional) |
| TMDR0 | B'0000_0000_0110_0000 | BFWT bit = 1 BFB bit = 1 BFA bit = 0 MD[2:0] bits = B'000 |

[Legend]

x: Don't care.

With the above settings, TGRD0 is used as a buffer register for TGRB0, and the contents of the general register are rewritten when the timer counter is cleared. The step interval can be changed by using DMA transfer or interrupt control to write a desired value to TGRD0.

The TPSC [2:0] bits in TCR0 are for a timer prescaler setting. The step interval is calculated from the following formula.

[Calculation example for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

With Internal clock at 66.66 MHz, B'011 as the value of TPSC[2:0] in TCR0, H'1425 as the value of TGRD0, the result is $15.02 \text{ ns} \times 64 \times 5157 \text{ (H'1425)} = 4,955,675.7 \text{ ns} \cong 5.0 \text{ ms}$. That is, the pattern will change after approximately 5.0 ms.

[Calculation example for RZ/G2E]

With Internal clock at 62.5 MHz, B'011 as the value of TPSC[2:0] in TCR0, H'1425 as the value of TGRD0, the result is $16.0 \text{ ns} \times 64 \times 5157 \text{ (H'1425)} = 5,280,768 \text{ ns} \cong 5.3 \text{ ms}$. That is, the pattern will change after approximately 5.3 ms.

(b) Sequence State and the Number of Steps

In stepping-motor control mode, operation is in three states (acceleration, normal, and deceleration) while the counter is running. In the acceleration and deceleration states, the step interval can be changed. Accordingly, the value of TGRD0 must be changed the same number of times as the patterns are changed.

In the normal state, the step interval cannot be changed. So, TGRD0 is only set once for a period in the normal state.

The numbers of steps in the acceleration, normal, and deceleration states are set in TMASR, TMTSR, and TMRSR, respectively. If H'0000 is written to TMASR or TMRSR, the corresponding state is skipped. Furthermore, the manual mode can be selected by writing H'0000 to TMTSR. In this mode, the transition from normal mode can only be initiated by changing the value in TMRR.

Table 66.10 gives descriptions of the sequence states and Figure 66.15 shows the transitions between sequence states.

Table 66.10 Description of Sequence States

| Start/Stop | State | Description | |
|------------|--------------|---------------|--|
| | | Step Interval | Definition in TGRD0 |
| Start | Acceleration | Variable | The number of steps specified for this state |
| | Normal | Fixed | Only once |
| | Deceleration | Variable | The number of steps specified for this state |
| Stop | Stop | — | — |

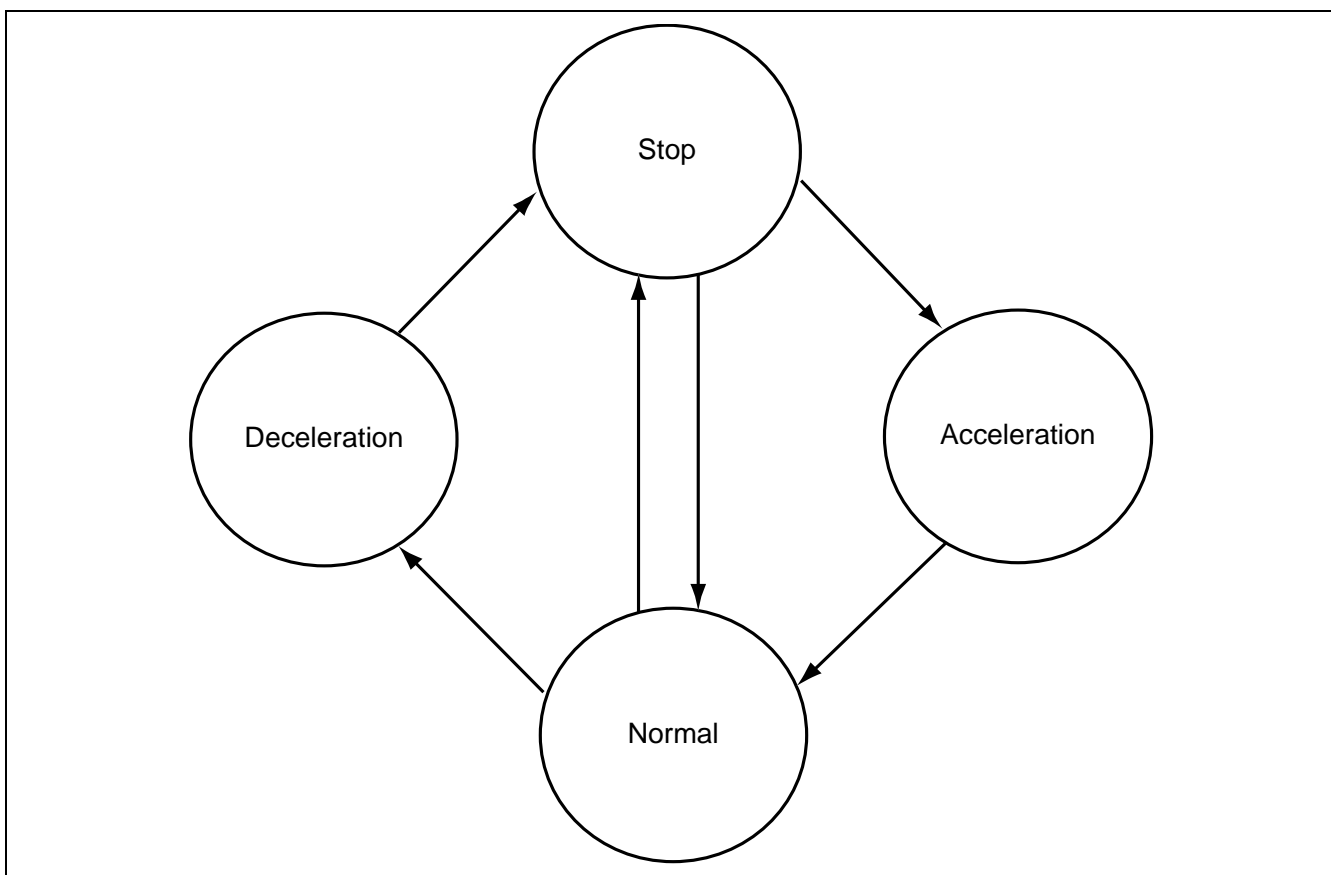


Figure 66.15 Sequence State Transition

(c) Setting of the Step Interval

Direct memory access or interrupt-driven transfer can be used to set the step interval

- DMA transfer

Setting the TDMAE bit in TMIR to 1 selects the use of DMA transfer to modify the step interval.

The number of requests for DMA transfer is given by $TMASR + 1$ (to transfer the step interval for the normal state) + $TMRSR$. The step intervals must be defined in sequence from the address indicated by the DMA source address register. Figure 66.16 shows an example of timing in operation with DMA transfer requests used to set the step intervals. “()” in Figure 66.16 means Stop Pattern outputted from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the stop state set by TMSPR0 and TMSPR1. “[]” in Figure 66.16 means Operation Pattern outputted from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the acceleration, normal, and deceleration states set by TMMPR0 and TMMPR1. Figure 66.17 shows the corresponding definitions of step intervals in the address map.

Table 66.11 lists an example of the DMA settings.

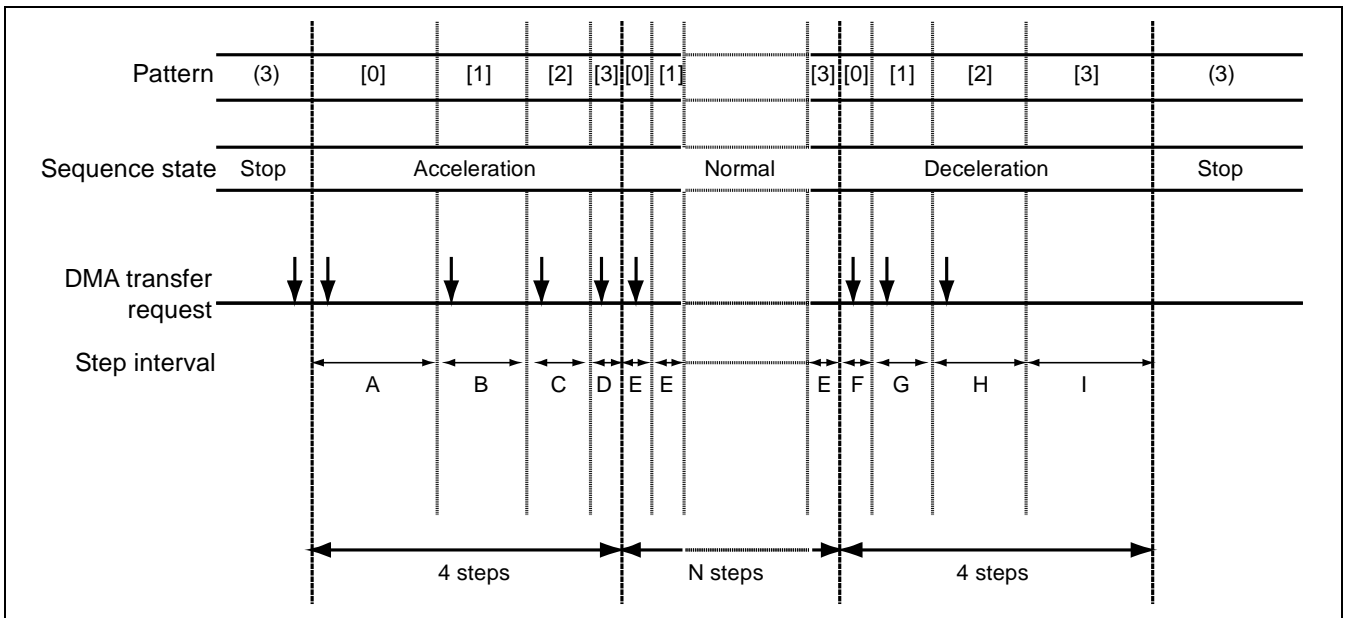


Figure 66.16 Operation with DMA Transfer Requests Used to Adjust the Step Interval

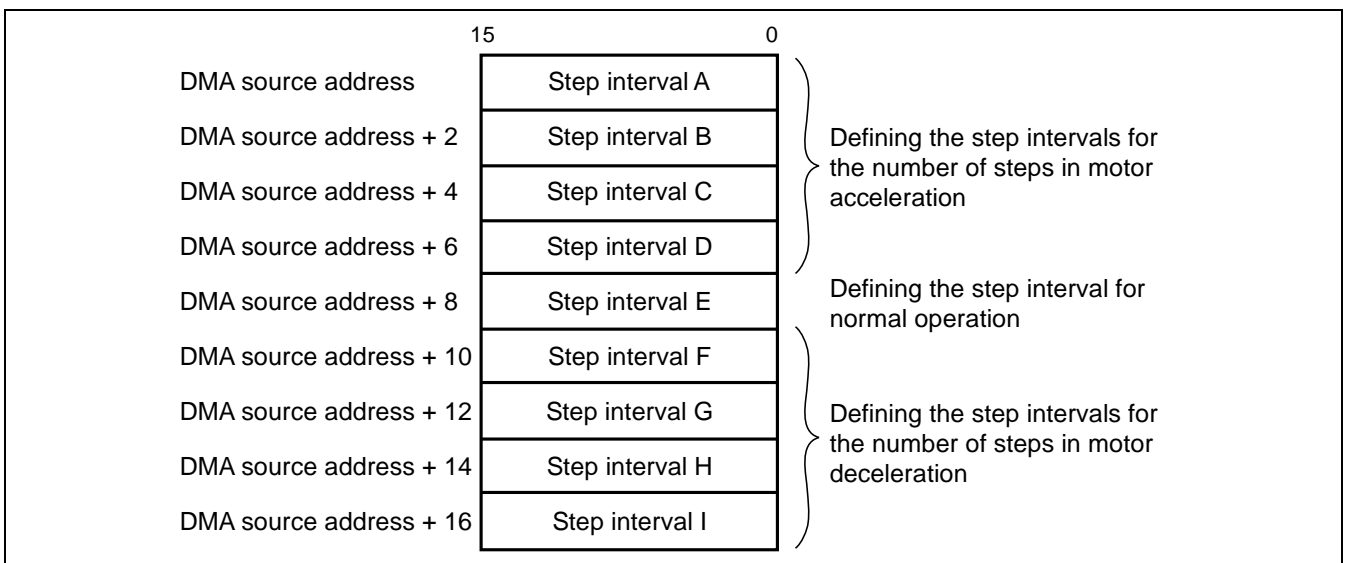


Figure 66.17 Address Map of Step Interval Definitions for the Example of Operation

Table 66.11 DMA Settings

| DMA Register | Setting | Description |
|----------------------------------|--|---|
| DMA source address register | H'XXXX_XXXX | Address where the table of step intervals starts |
| DMA destination address register | H'E6E8_0034 | Address of TGRD0 |
| DMA transfer count register | H'0000_0000 | (Setting for the maximum number of transfer operations) * |
| DMA transfer count register B | H'XXXX_XXXX | The number of DMA transfer requests is set in the higher- and lower-order 16 bits. (for Figure 66.16, H'0009_0009) |
| DMA channel control register | H'E6E8_80*C | Reload mode: Use source address registers (SARs) as the source for reloading. Fixed destination address. Incrementation of source addresses. The DMA extension resource selector is used. DMA transfer is in word (two-byte) units. The other bits are in the default settings. |
| DMA extension resource selector | H'17 in the bit field for the selected DMA channel | Select TPU |
| DMA operation register | H'0001 | Permits DMA transfer on all channels. The other bits are in the default settings |

Note: * Transfer is stopped when the value of the DMA transfer counter reaches 0, so change the value of the DMA transfer counter back to 1 after TPU-related operations are finished.

- Interrupt-driven transfer

Setting the TMDRFE bit in TIER0 to 1 allows the use of interrupt-driven transfer to change the step interval.

$TMASR-1 + 1$ (to transfer the step interval in the normal state) + TMRSR gives the number of times TGRD0 must be written to by interrupt-driven transfer.

This method is unlike DMA transfer in that an interrupt is not generated when the initial transfer is required. The initial value should thus be written to TGRD0 before counter operation starts.

Once interrupt generation starts, the interrupt handler writes the step intervals to TGRD0 and writes 0 to the TMDRFS bit after reading it as 1.

Figure 66.18 shows an example of timing in operation with interrupt-driven transfer used to set the step intervals. “()” in Figure 66.18 means Stop Pattern outputted from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the stop state set by TMSPR0 and TMSPR1. “[]” in Figure 66.18 means Operation Pattern outputted from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the acceleration, normal, and deceleration states set by TMMPR0 and TMMPR1.

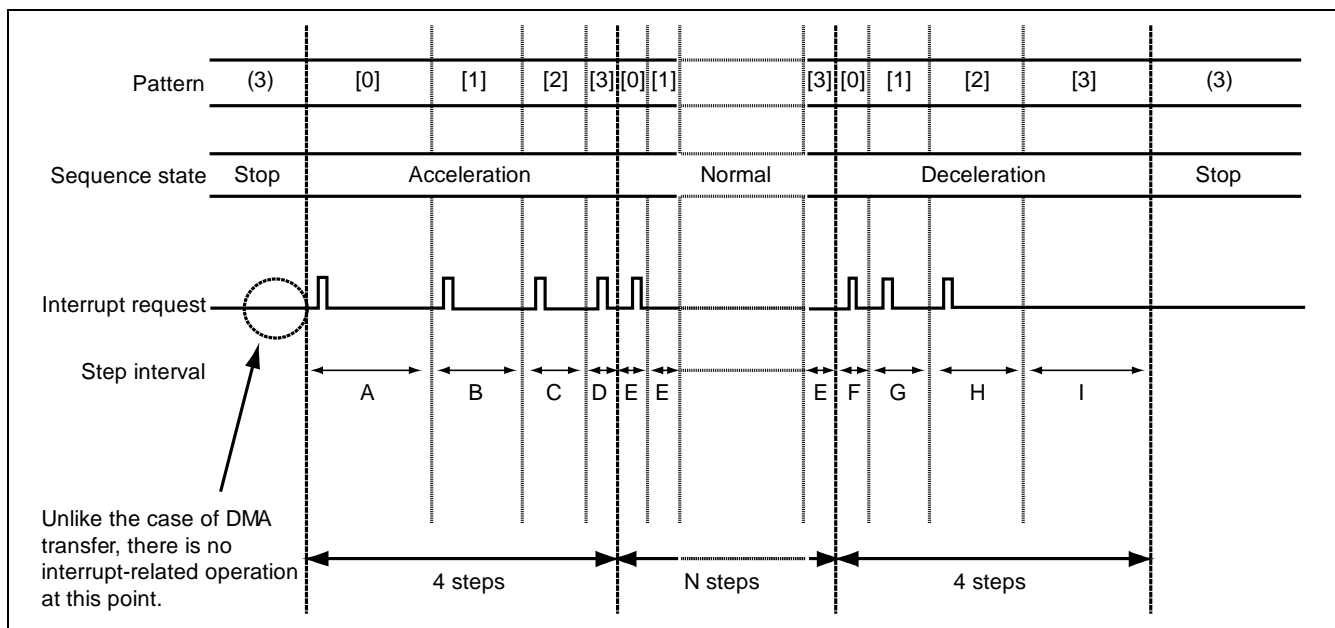


Figure 66.18 Interrupt-Driven Adjustment of the Step Interval

The step-intervals in the example of operation shown in Figure 66.18 are defined in the following stages.

- Write step interval A to TGRD0 before starting the counter.
 - Write step interval B to TGRD0 in response to the first interrupt.
 - Write step interval C to TGRD0 in response to the second interrupt.
 - Write step interval D to TGRD0 in response to the third interrupt.
 - Write step interval E to TGRD0 in response to the fourth interrupt.
 - Write step interval F to TGRD0 in response to the fifth interrupt.
 - Write step interval G to TGRD0 in response to the sixth interrupt.
 - Write step interval H to TGRD0 in response to the seventh interrupt.
 - Write step interval I to TGRD0 in response to the eighth interrupt.
- } Defining the step intervals for the number of steps in motor acceleration
- } Defining the step interval for normal operation
- } Defining the step intervals for the number of steps in motor deceleration

(d) Pattern Setting

In stepping-motor control mode, each four-bit value for output on pins TPU0TO0 to TPU0TO3 is regarded as a single pattern. The patterns are in a sequence, and a compare match with the timer counter triggers each transition to the next pattern.

The following rules apply to pattern transfer.

- Either a four- or an eight-pattern cycle may be used. The number is determined by the MTRPATKIND bit in TMIR0.
- Each pattern is indicated by a pattern number. The current pattern number is either incremented or decremented on each compare match with the timer counter. Incrementation or decrementation of the pattern numbers is determined by the MTRPATDOWN bit in TMIR0.
- The pattern-number counter returns to 0 when it overflows. When it underflows, the pattern number becomes three (for a four-pattern cycle) or seven (for an eight-pattern cycle).
- Operating patterns are defined as the output patterns in the acceleration, normal, and deceleration states, and stop patterns are defined as the output patterns in the stop state. In transitions from an operating pattern to a stop pattern, the pattern number remains the same. In transitions from a stop pattern to an operating pattern, the current pattern number changes to that of the next pattern.

The operating patterns are set in TMMPR0 and TMMPR1, and the stop patterns are set in TMSPR0 and TMSPR1.

Table 66.12 lists the order of patterns for different settings of the MTRPATDOWN and MTRPATKIND bits in TMIR.

Table 66.12 Settings of the MTRPATDOWN and MTRPATKIND Bits in TMIR and Orders of Patterns

| TMIR | | Example of Pattern Order (Operating Pattern [], Stop Pattern ()) |
|----------------|-----------------|---|
| MTRPATDOWN Bit | MTRPATKIND Bits | |
| 0 | B'00 | [0] → [1] → [2] → [3] → [0] → [1] → [2] → [3] → (3) → [0] |
| 1 | | [0] → [3] → [2] → [1] → [0] → [3] → [2] → [1] → (1) → [0] |
| 0 | B'01 | [0] → [1] → [2] → [3] → [4] → [5] → [6] → [7] → (7) → [0] |
| 1 | | [0] → [7] → [6] → [5] → [4] → [3] → [2] → [1] → (1) → [0] |

(e) Specifying the Output Pattern in the Stop State

Specifying a pattern number in the NOWPAT [2:0] bits of TMOPR while output is in the stop state leads to output of the corresponding pattern on pins TPU0TO3 to TPU0TO0. The initial value is 0.

(f) Selecting the Stepping-Motor Control Mode

Stepping-motor control by the TPU circuit is selected by setting the MTRON bit in TMIR to 1. The output on pins TPU0TO3 to TPU0TO0 is switched to the output corresponding to stepping-motor control.

Switching should proceed while the motor is stopped.

The stop pattern specified as pattern 0 in TMOPR is output after switching. Therefore, verify the value in TMOPR or write the desired pattern to TMOPR.

(g) Starting Stepping-Motor Control

Stepping-motor control is started by setting the TMST bit in TSTR to 1. This bit is automatically cleared (to 0) in the stop state after motor-control operations are over.

(h) Example of the Procedure for Setting up Stepping-Motor Control

Figure 66.19 is a flowchart of stepping-motor control.

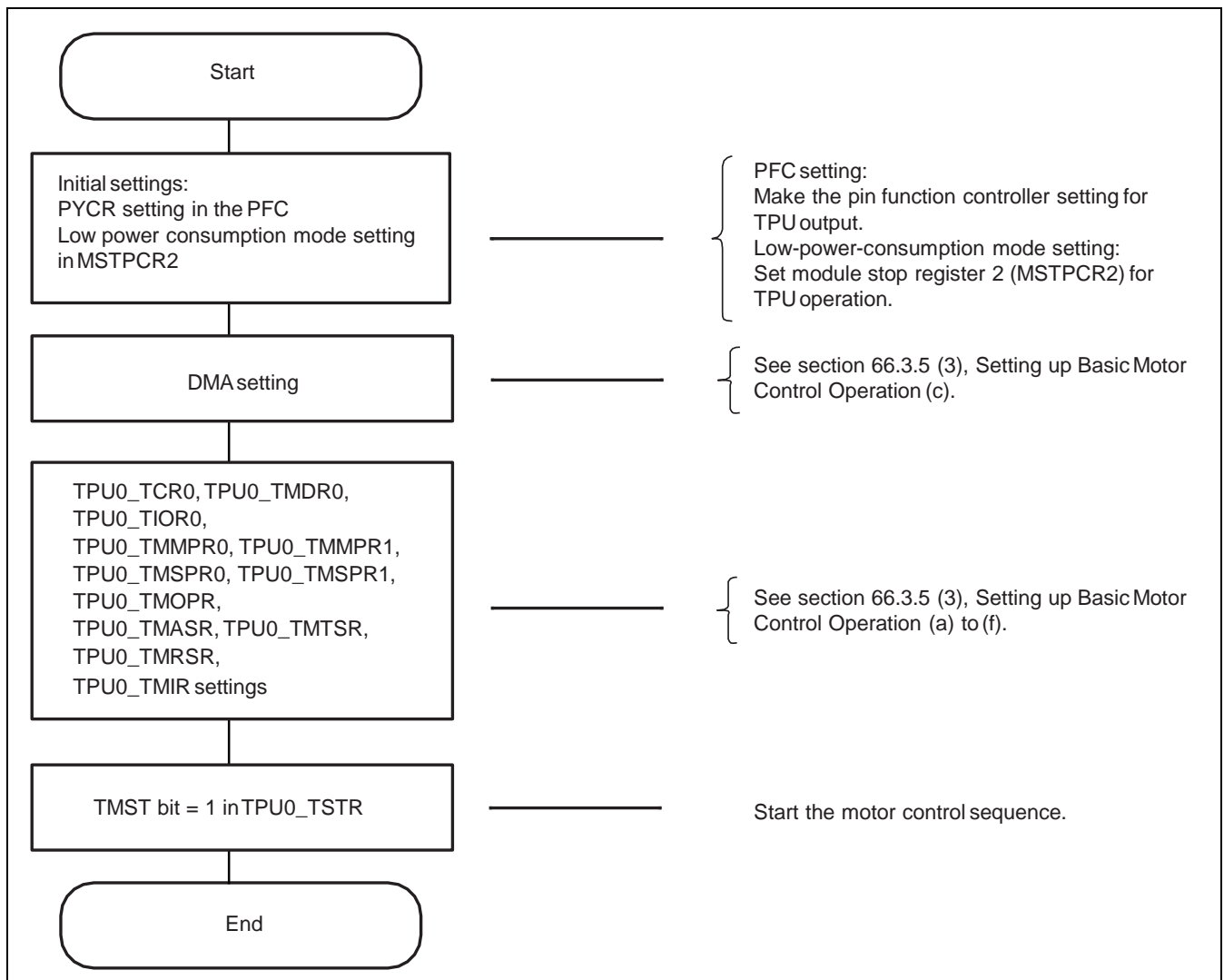


Figure 66.19 Flowchart

(4) Changing the Number of Steps in the Normal State.

The number of steps in the normal state is set in TMTSR. The operations for changing the number of steps are described under (a) and (b) below.

In manual mode (TMTSR = H'0000), the operation under (a) is the only way of switching from the normal state.

(a) Changes by TMRR

Writing 1 to the REDUON0 bit or the REDUON bit in TMRR initiates the transition to the deceleration state (or, if TMRSR = H'0000, to the stop state).

Table 66.13 describes how operation in response to the writing of 1 to the REDUON0 bit varies with the operating state. Table 66.14 describes how operation in response to the writing of 1 to the REDUON bit varies with the operating state.

Table 66.13 Timing of Writing 1 to the REDUON0 Bit and Operation

| Timing of Writing 1 to the REDUON0 Bit | Operation |
|--|---|
| Acceleration | Operation continues normally until the end of acceleration. Control then enters the normal state. |
| Normal | Control remains in the normal state until the pattern number becomes [0]. The transition to the deceleration state (or the stop state if TMRSR is H'0000) follows the end of this step in the normal state. |
| Deceleration | Writing 1 to the bit has no effect on deceleration. |
| Stop | TMRR is cleared to H'0000 after control has made the transition from the normal or deceleration state to the stop state. When 1 is written to the REDUON0 bit after control is in the stop state, the sequence in the next round of motor control continues until the pattern number in the normal state becomes [0]. The transition to the deceleration state (or the stop state if TMRSR is H'0000) is made at the end of this step in the normal state. |

Table 66.14 Timing of Writing 1 to the REDUON Bit and Operation

| Timing to Write 1 in REDUON Bit | Operation |
|---------------------------------|---|
| Acceleration | Operation continues normally until the end of acceleration. |
| Normal | The transition to the deceleration state (or the stop state if TMRSR is H'0000) is made after a single step in the normal state. |
| Deceleration | Writing 1 to the bit has no effect on deceleration. |
| Stop | TMRR is cleared to H'0000 after control has made the transition from the normal or deceleration state to the stop state. When 1 is written to the REDUON0 bit after control is in the stop state, the sequence in the next round of motor control continues until it enters the normal state. The transition to the deceleration state (or the stop state if TMRSR is H'0000) is made at the end of a single step in the normal state. |

(b) Changing the Number of Steps by Changing the Value in TMSCR

In auto mode (i.e. when TMTSR ≠ H'0000), TMSCR indicates the remaining number of steps in the current state.

When the control sequence is in the normal state, the number of steps can be changed by writing the desired number of steps. Writing is ineffective in states other than the normal state.

Do not write to this register when control is in manual mode (TMTSR = H'0000).

(5) Acquiring the Number of Steps in the Normal State

In the stop state, TMTCR can be read to acquire the number of steps in the normal state of the immediately preceding sequence.

(6) Pulse Mode

Writing H'0000 to TMSR and TMRSR selects operation in the normal state alone, with a constant step interval. That is, operation starts with the step interval defined for the normal mode and the interval does not subsequently change. This is referred to as pulse mode operation.

Examples of pulse-mode operation are described below.

(a) Single-Pulse Shifts in Pulse Mode and Auto Mode

Table 66.15 gives an example of register settings for single-pulse steps in auto mode.

Figure 66.20 is a timing chart for single-pulse shifts in auto mode. “()” in Figure 66.20 means Stop Pattern outputted from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the stop state set by TMSPR0 and TMSPR1. “[]” in Figure 66.20 means Operation Pattern outputted from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the normal states set by TMMPR0 and TMMPR1.

Table 66.15 Register Settings to Select One Pulse as the Number of Steps in Auto Mode

| Register Setting Value | Contents |
|------------------------|-----------------------------------|
| TMSR: H'0000 | No acceleration state |
| TMTSR: H'0001 | Only one step in the normal state |
| TMRSR: H'0000 | No deceleration state |

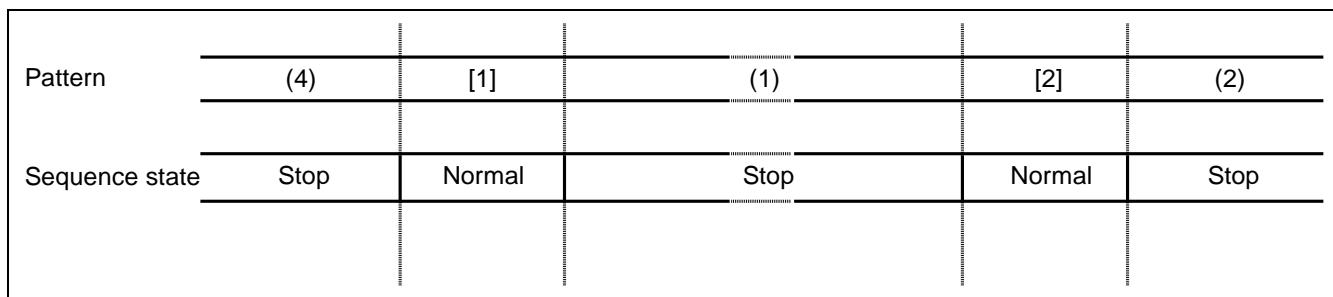


Figure 66.20 Timing of Single-Pulse Shifts in Auto Mode

(b) Single-Pulse Shifts in Pulse Mode and Manual Mode

In manual mode, setting the REDUON0 or REDUON bit in TMRR to 1 while the sequence is in the normal state enables a single pulse shift.

Table 66.16 gives an example of register settings for single-pulse steps in manual mode.

Figure 66.21 is a timing chart for single-pulse shifts in manual mode. “()” in Figure 66.21 means Stop Pattern outputted from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the stop state set by TMSPR0 and TMSPR1. “[]” in Figure 66.21 means Operation Pattern outputted from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the normal states set by TMMPR0 and TMMPR1.

Table 66.16 Register Settings to Select One Pulse as the Number of Steps in Manual Mode

| Register Setting Value | Contents |
|------------------------|-----------------------------|
| TMASR: H'0000 | No acceleration state |
| TMTSR: H'0000 | Looping in the normal state |
| TMRSR: H'0000 | No deceleration state |

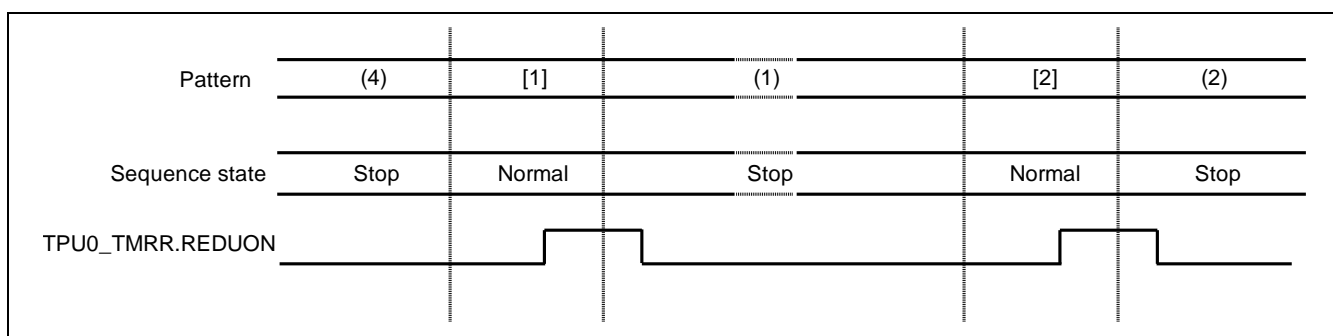


Figure 66.21 Timing of Single-Pulse Shifts in Manual Mode

(7) Interrupts

Timer 0 has six added interrupt conditions for use with stepping-motor control mode.

The interrupt conditions are described below.

(a) Detection of Overflows of Data Transferred for Motor Control

Setting the TMDOFE bit in TIER0 to 1 selects the detection of overflows of data transferred for motor control (motor control data transfer overflows). Here, overflow refers to the condition where buffer-related operation, i.e. the transfer of the data from TGRD0, is delayed; that is, when the operation TGRD0→ TGRB0 is delayed. To cancel this interrupt, write 0 to the TMDOFS bit of TSR0 after having read it as 1.

Figure 66.22 shows an example of DMA transfer where an overflow in data transfer for motor control occurs and detection of overflows has been enabled. “()” in Figure 66.22 means Stop Pattern outputted from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the stop state set by TMSPR0 and TMSPR1. “[]” in Figure 66.22 means Operation Pattern outputted from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the acceleration, normal, and deceleration states set by TMMPR0 and TMMPR1.

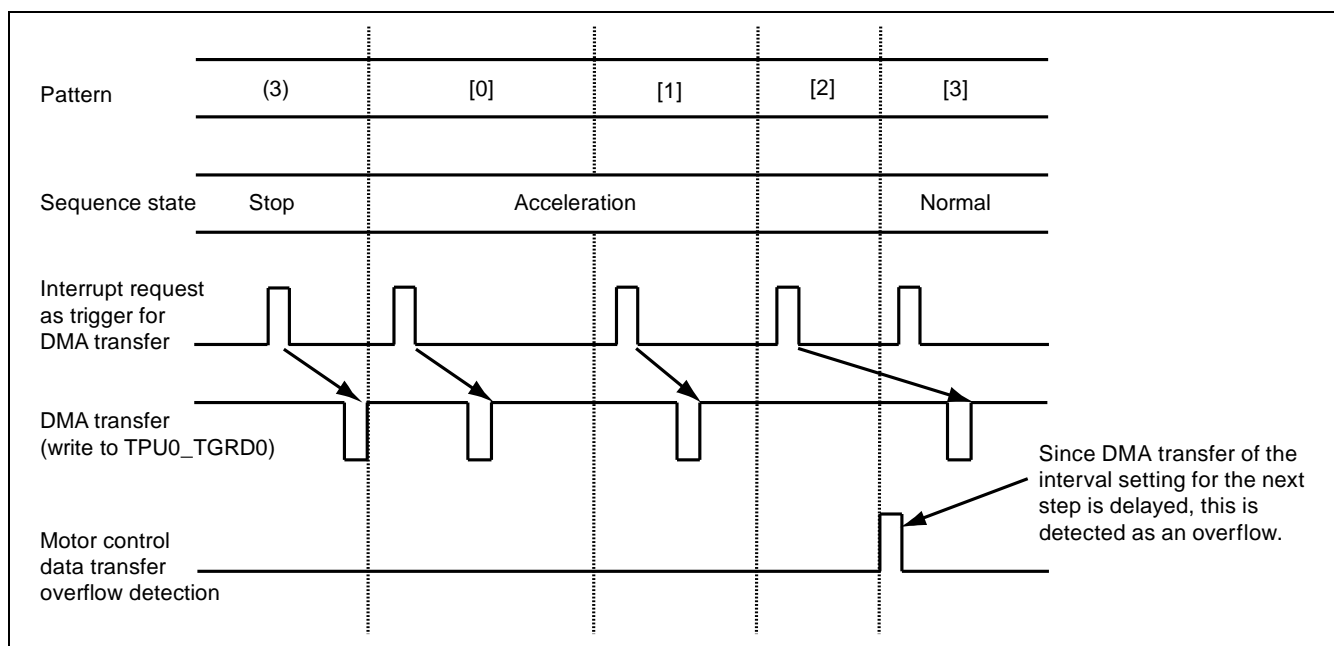


Figure 66.22 DMA Transfer with Detection of Motor Control Data Transfer Overflows Enabled

Since this interrupt is for use in the debugging of defective operation, normal operation will not be restored even if it is detected.

(b) Detection of Requirement for Transfer of Motor control Data

Detection of requirements for the transfer of motor control data (detect motor control data transfer request) is enabled by setting the TMDRFE bit of TIER0. This interrupt is generated when the next pattern interval is to be changed from the current interval. On detection of this interrupt, the handler should write the setting for the next interval to TGRD0 and then write 0 to the TMDRFS bit of TSR0 after having read it as 1.

This interrupt is not available if DMA transfer has been selected (the TDMAE bit in TMIR = 1).

See Figure 66.18 for an example of usage of this interrupt.

(c) Detecting the Transition to Deceleration

Setting the TMS1ER bit in TIER0 to 1 enables detection of the transition to deceleration. The handler should clear the TMCFR flag of TSR0 by writing 0 to the bit after having read it as 1.

(d) Detecting the Transition to the Normal State

Setting the TMS1ET bit in TIER0 to 1 enables detection of the transition to the normal state. The handler should clear the TMCFT flag of TSR0 by writing 0 to the bit after having read it as 1.

(e) Detecting the Transition to Acceleration

Setting the TMS1EA bit in TIER0 to 1 enables detection of the transition to acceleration. The handler should clear the TMCFA flag of TSR0 by writing 0 to the bit after having read it as 1.

(f) Detecting the Transition to the Stop State

Setting the TMS1ES bit in TIER0 to 1 enables detection of the transition to the stop state. The handler should clear the TMCFS flag of TSR0 by writing 0 to the bit after having read it as 1.

(8) Checking the State of Operation

Read TMSR and TMSCR to check the current state of operations. TMSCR indicates the value resulting from decrementation (on every compare match with the timer counter) from the values in the number of steps register (TMASR, TMTSR, or TMRSR) during sequence transitions.

Figure 66.23 shows how the values in TMSR and TMSCR vary through the four states. “()” in Figure 66.23 means Stop Pattern outputted from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the stop state set by TMSPR0 and TMSPR1. “[]” in Figure 66.23 means Operation Pattern outputted from TPU0TO0, TPU0TO1, TPU0TO2, and TPU0TO3 in the acceleration, normal, and deceleration states set by TMMPR0 and TMMPR1. As is shown in the figure, the current step within the overall sequence of state transitions can be estimated by, for example, reading TMSR = H'0008 and TMSCR = H'0003 in the interval indicated in the figure below.

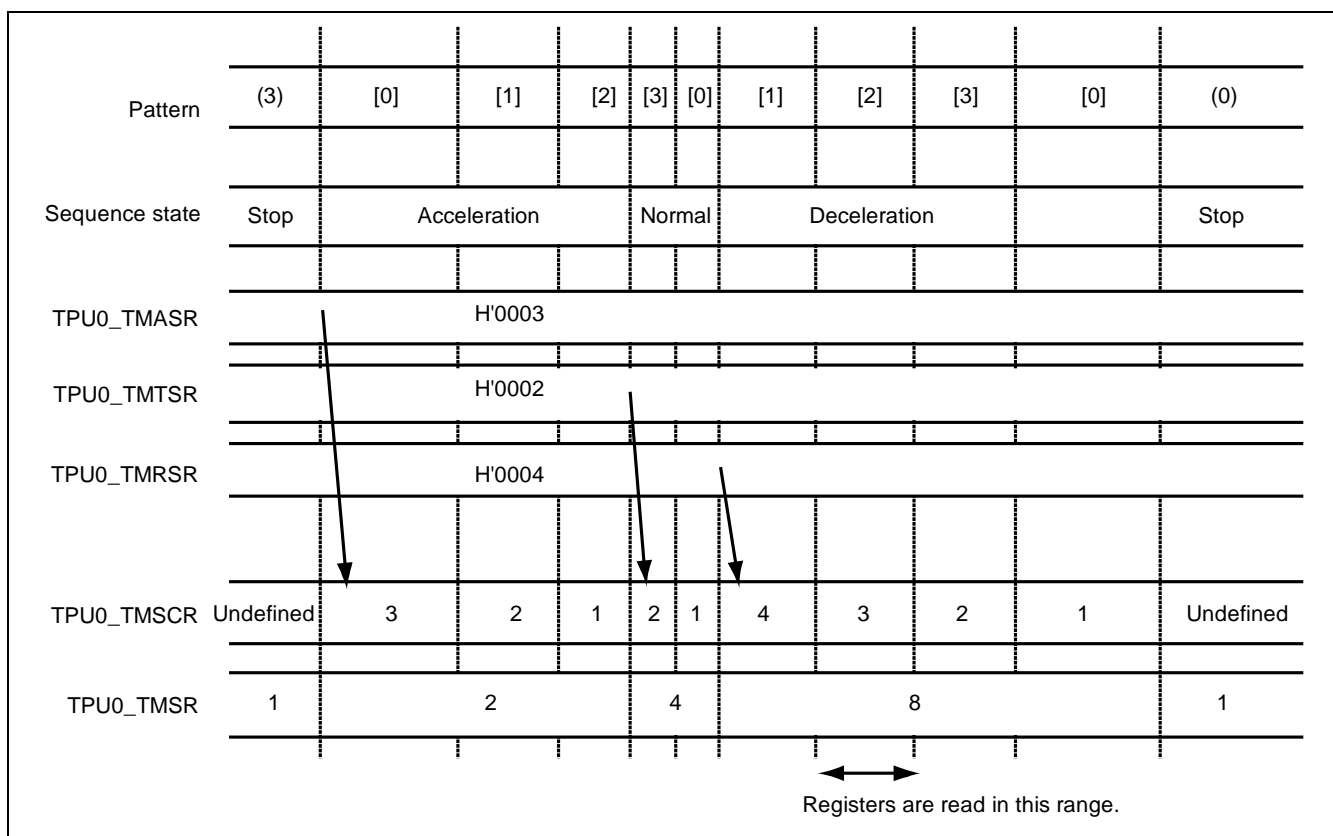


Figure 66.23 Values of TMSR and TMSCR

67. Compare Match Timer Type0 (CMT0)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

67.1 Overview

Compare Match Timer Type0 (CMT0) is a 32-bit compare match timer of two channel.

67.1.1 Features

- Two channels
- 16 bits/32 bits can be selected as counter size (bit-width).
- Provided with 32-bit constant registers and 32-bit up counters that can be written or read at any time.
- Following four clocks can be selected as counter clocks:
 - RCLK: 1/1, 1/8, 1/32, and 1/128
- One-shot operation or free-running operation is selectable.
- Compare match or overflow can be selected as interrupt source.
- Counter operation can be enabled or disabled at the time of debugging of CPU core using the debugging mode operation selector.

67.1.2 Block Diagram

Figure 67.1 shows a block diagram of the CMT0.

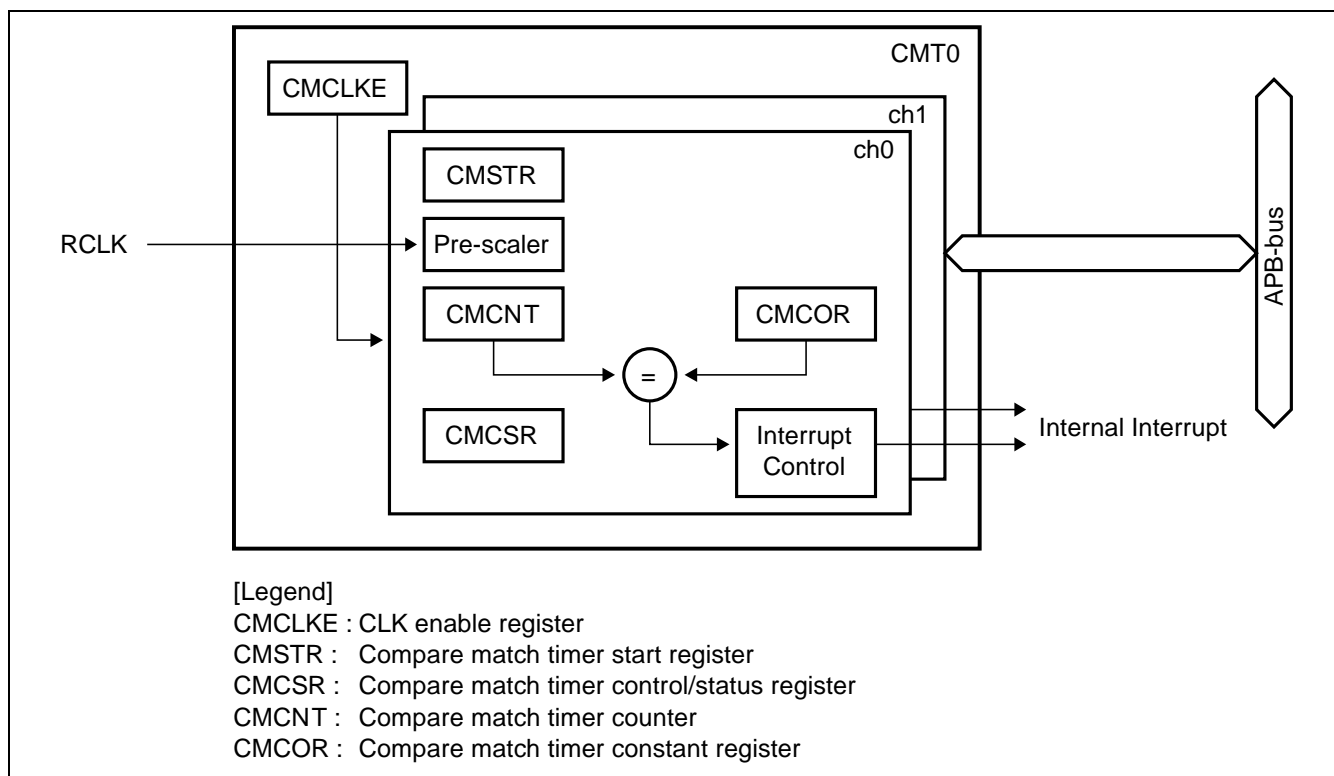


Figure 67.1 Block Diagram of CMT0

67.1.3 Register Configuration

Table 67.1 shows the CMT0 register configuration. Table 67.2 shows the register states in each operating mode. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined. Access the register as a longword (32 bits). Operation cannot be guaranteed if the register is not accessed as a longword.

Table 67.1 Register Configuration

| Register Name | Abbreviation | R/W | Address | Access Size | Second Generation RZ/G Series Products | | | | |
|---|--------------|------|-------------|-------------|--|----------------------------|--------|--------|--|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| CLK enable register | CMCLKE | R/W | H'E60F_1000 | 32 | √ | √ | √ | √ | |
| Compare match timer start register 0 | CMSTR0 | R/W | H'E60F_0500 | 32 | √ | √ | √ | √ | |
| Compare match timer control/status register 0 | CMCSR0 | R/W* | H'E60F_0510 | 32 | √ | √ | √ | √ | |
| Compare match timer counter 0 | CMCNT0 | R/W | H'E60F_0514 | 32 | √ | √ | √ | √ | |
| Compare match timer constant register 0 | CMCOR0 | R/W | H'E60F_0518 | 32 | √ | √ | √ | √ | |
| Compare match timer start register 1 | CMSTR1 | R/W | H'E60F_0600 | 32 | √ | √ | √ | √ | |
| Compare match timer control/status register 1 | CMCSR1 | R/W* | H'E60F_0610 | 32 | √ | √ | √ | √ | |
| Compare match timer counter 1 | CMCNT1 | R/W | H'E60F_0614 | 32 | √ | √ | √ | √ | |
| Compare match timer constant register 1 | CMCOR1 | R/W | H'E60F_0618 | 32 | √ | √ | √ | √ | |

Note: * Except for bit [15:14] that are R/WC0 register.

Table 67.2 Register States in Each Operating Mode

| Register Abbreviation | Power-on Reset/Software Reset | Module Standby |
|-----------------------|-------------------------------|----------------|
| CMCLKE | Initialized | Retained |
| CMSTR0 | Initialized | Retained |
| CMCSR0 | Initialized | Retained |
| CMCNT0 | Initialized | Retained |
| CMCOR0 | Initialized | Retained |
| CMSTR1 | Initialized | Retained |
| CMCSR1 | Initialized | Retained |
| CMCNT1 | Initialized | Retained |
| CMCOR1 | Initialized | Retained |

67.2 Register Description

[Legend for Register Description]

Initial value: Register value after a reset.

—: Undefined value

R/W: Bit or field is readable and writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R: Read-only. The write value should always be initial value.

All access to register is made in longword units.

67.2.1 CLK Enable Register (CMCLKE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMCLKE is a 32-bit register, which specifies clock supply to each channel. When there are unused channels, set 0 to this register, for stop supplying clock to the channel. It is prohibited to stop clock supply, while counter is working.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----------|----------|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | Ch1 clke | Ch0 clke | — | — | — | — | — |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 7 | — | All 1 | R | Reserved These bits are always read as 1. The write value should always be 1. |
| 6 | Ch1clke | B'1 | R/W | 0: Clock isn't supplied to ch1. 1: Clock is supplied to ch1. |
| 5 | Ch0clke | B'1 | R/W | 0: Clock isn't supplied to ch0. 1: Clock is supplied to ch0. |
| 4 to 0 | — | All 1 | R | Reserved These bits are always read as 1. The write value should always be 1. |

67.2.2 Compare Match Timer Start Registers 0 and 1 (CMSTR0, CMSTR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMSTRn (n = 0 and 1) is a 32-bit register which specifies the operation of compare match timer counters (CMCNTn).

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | STR0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | STR0 | B'1 | R/W | Count Start These bits specify start/halt of compare match timer counter (CMCNTn). 0: CMCNTn halts 1: CMCNTn starts counting |

67.2.3 Compare Match Timer Control/Status Registers 0 and 1 (CMCSR0, CMCSR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMCSRn (n = 0 and 1) is a 32-bit register that indicates the occurrence of compare matches, enables interrupts, and sets the counter input clocks.

Do not change bits other than the CMF and OVF bits, while compare match timer counter (CMCNTn) is under counting.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|----|----|----|-----|-----|----|----|----------|---------|----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CMF | OVF | WRFLG | — | — | — | CMS | CMM | — | — | CMR[1:0] | DBGIV D | CKS[2:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| R/W: | R/WC0 | R/WC0 | R | R | R | R | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | CMF | B'0 | R/WC0 | Compare Match Flag This flag indicates whether values of the compare match timer counter (CMCNTn) and compare match timer constant register (CMCORn) have matched or not. Software cannot write 1 to the bit. When one-shot is selected for the counter operation, counting resumes by clearing this bit. 0: CMCNTn and CMCORn values have not matched. [Clearing condition] Write 0 to CMF 1: CMCNTn and CMCORn values have matched. Note: Only 0 can be written to clear the flag. |
| 14 | OVF | B'0 | R/WC0 | Overflow Flag This flag indicates whether the compare match timer counter (McCants) has overflowed or not. Software cannot write 1 to this bit. 0: CMCNTn has not overflowed. [Clearing condition] Write 0 to OVF 1: CMCNTn has overflowed. Note: Only 0 can be written to clear the flag. |
| 13 | WRFLG | B'0 | R | Write State Flag Write access to CMCNTn is prohibited, while this bit is 1. Further behavior is not guaranteed, if data is written while this bit is 1. This bit indicates CMCNTn is in synchronization period for setting previously written data. Confirm that this flag is 0, before writing to CMCNTn. |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 12 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | CMS | B'0 | R/W | Compare Match Timer Counter Size Specify whether the compare match timer counter (CMCNTn) is used as a 16-bit counter or a 32-bit counter. This bit specifies the valid size of compare match timer constant register (CMCORn). 0: Operates as a 32-bit counter. 1: Operates as a 16-bit counter. Note: Refer to section 67.3.5, CMT0 Register Access, for the note regarding writing to or reading from this bit. |
| 8 | CMM | B'1 | R/W | Compare Match Mode Specify counter operation mode. 0: One-shot operation 1: Free-running operation Note: Refer to section 67.3.5, CMT0 Register Access, for the note regarding writing to or reading from this bit. |
| 7, 6 | — | All 0 | R | Reserved This bit is always read as 0. The write value should always be 0. |
| 5, 4 | CMR[1:0] | B'00 | R/W | Compare Match Request These bits enable or disable an internal interrupt request in a compare match. B'00: Disables an internal interrupt request. B'01: Setting prohibited B'10: Enables an internal interrupt request. B'11: Setting prohibited Note: Refer to section 67.3.5, CMT0 Register Access, for the note regarding writing to or reading from this bit. |
| 3 | DBGIVD | B'1 | R/W | Debug Mode Operation Select Sets the counter operation in debug mode. 0: Stops the counter operation in debug mode. 1: Enables the counter operation even in debug mode. Note: Refer to section 67.3.5, CMT0 Register Access, for the note regarding writing to or reading from this bit. |
| 2 to 0 | CKS[2:0] | B'111 | R/W | Clock Select These bits select the input clock to CMCNTn. When the count start bit (STR0) in CMSTRn is set to 1, CMCNTn begins incrementing with the clock selected by these bits. B'000: Setting prohibited B'001: Setting prohibited B'010: Setting prohibited B'011: Setting prohibited B'100: RCLK/8 B'101: RCLK/32 B'110: RCLK/128 B'111: RCLK/1 Note: Refer to section 67.3.5, CMT0 Register Access, for the note regarding writing to or reading from this bit. |

67.2.4 Compare Match Timer Counters 0 and 1 (CMCNT0, CMCNT1)

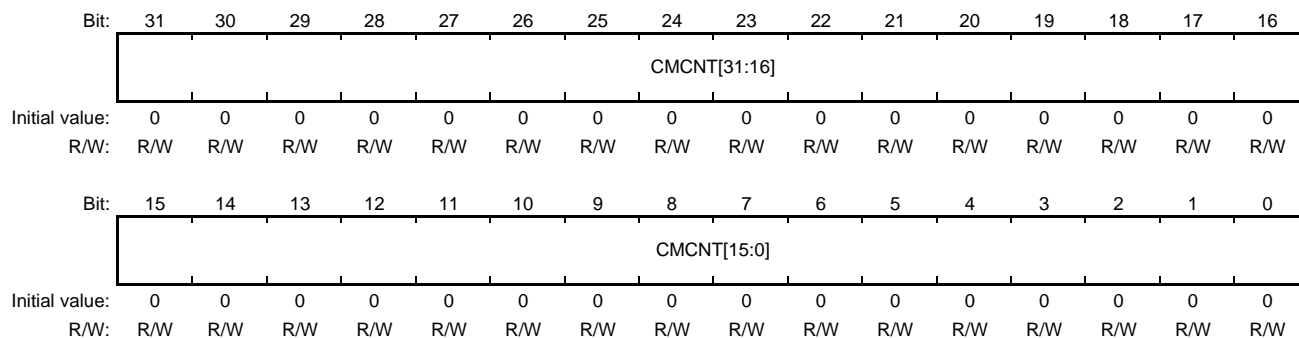
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMCNTn (n = 0 and 1) is a 32-bit register that is used as an up-counter.

To specify counter operation, set compare match timer control/status register n (CMCSRn), before starting operation of corresponding channel.

When the 16-bit counter operation is selected by the CMS bit, bits 31 to 16 of this register become invalid. When data is written to this register in 16-bit mode, write H'0000 to the upper 16 bits.

When CMCNTn is read during the counter operation, the read value may be wrong because different clock is used between counter and bus-interface. For exact value, read this register continuously, until same values are read from this register.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|--|
| 31 to 0 | CMCNT [31:0] | H'0000_0000 | R/W | Compare match timer counter bit31 to 0 Note: Refer to section 67.3.5, CMT0 Register Access, for the note regarding writing to or reading from this bit. |

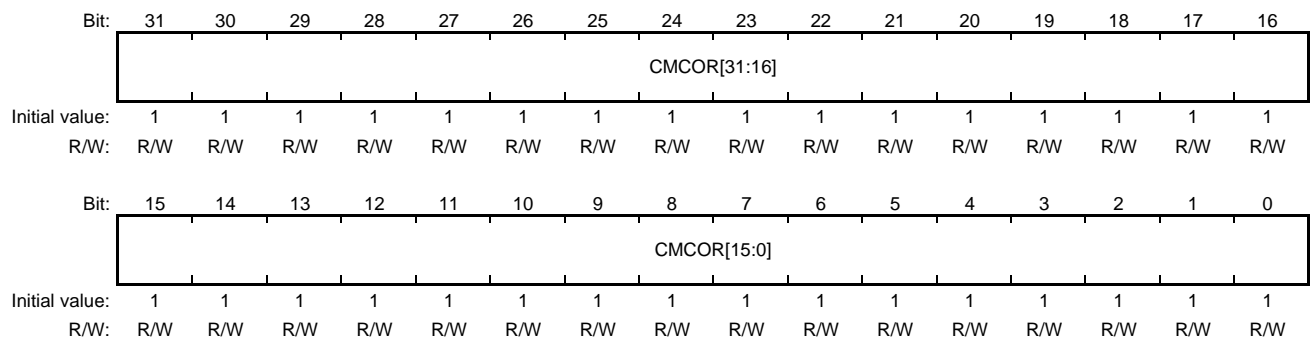
67.2.5 Compare Match Timer Constant Registers 0 and 1 (CMCOR0, CMCOR1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMCORn (n = 0 and 1) is a 32-bit register that sets the compare match period with CMCNTn.

When the 16-bit counter operation is selected by the CMS bit in CMCSRn, bits 15 to 0 of this register become valid. Write H'0000 to upper 16bits in 16-bit counter operation.

An overflow is detected when CMCNTn is cleared to 0 and this register is H'FFFF_FFFF.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|--|
| 31 to 0 | CMCOR [31:0] | H'FFFF_FFFF | R/W | Compare match timer constant register bit31 to 0 Note: Refer to section 67.3.5, CMT0 Register Access, for the note regarding writing to or reading from this bit. |

67.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

67.3.1 Counter Operation

The CMT0 starts the operation of the counter by writing 1 to the STR0 bit in CMSTRn after each register has been set. Complete all of the settings before starting the operation. Do not change the register settings other than clearing flag bits, during the compare match timer n (CMCNTn) is under operation.

The counter operates in one of two ways.

- One-Shot Operation

One-shot operation is selected by setting the CMM bit in CMCSRn to 0. When the value in CMCNTn matches the value in CMCORn, the value in CMCNTn is cleared to H'0000_0000 and the CMF bit in CMCSRn is set to 1. Counting by CMCNTn stops after it has been cleared.

To detect an overflow interrupt, set the value in CMCORn to H'FFFF_FFFF. When the value in CMCNTn matches the value in CMCORn, CMCNTn is cleared to H'0000_0000 and the CMF and OVF bits in CMCSRn are set to 1.

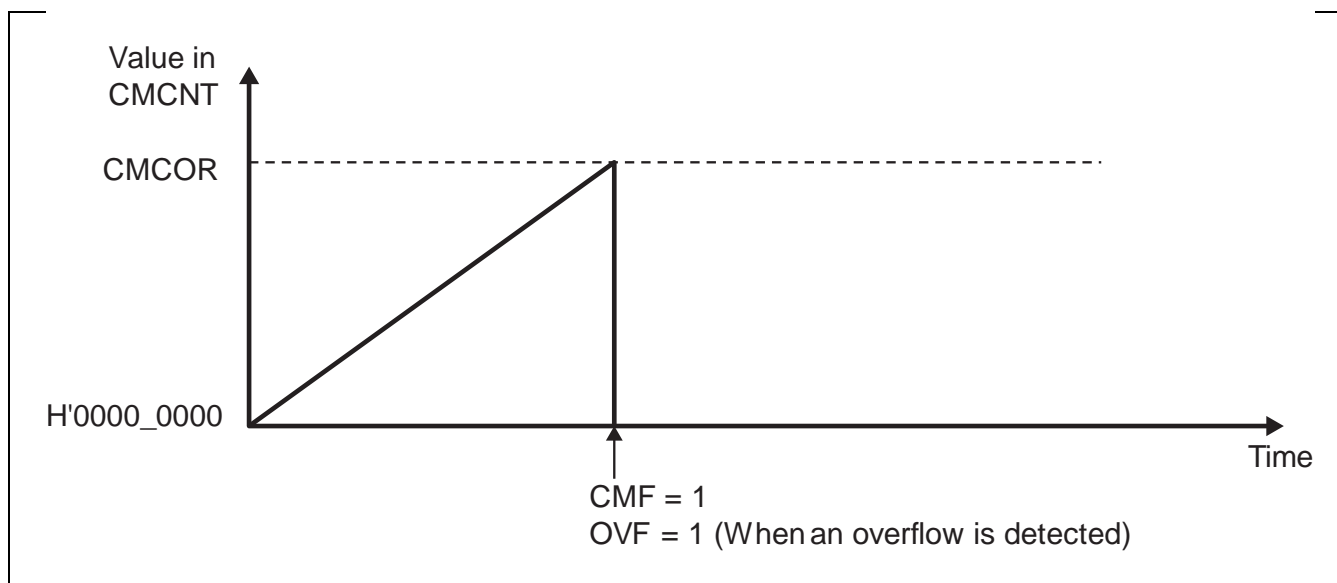


Figure 67.2 Counter Operation (One-Shot Operation)

- Free-Running Operation

Free-running operation is selected by setting the CMM bit in CMCSRn to 1. When the value in CMCNTn matches the value in CMCORn, CMCNTn is cleared to H'0000_0000 and the CMF bit in CMCSRn is set to 1. CMCNTn resumes counting-up after it has been cleared.

To detect an overflow interrupt, set CMCORn to H'FFFF_FFFF. When the values in CMCNTn and CMCORn match, CMCNTn is cleared to H'0000_0000 and the CMF and OVF bits in CMCSRn are set to 1.

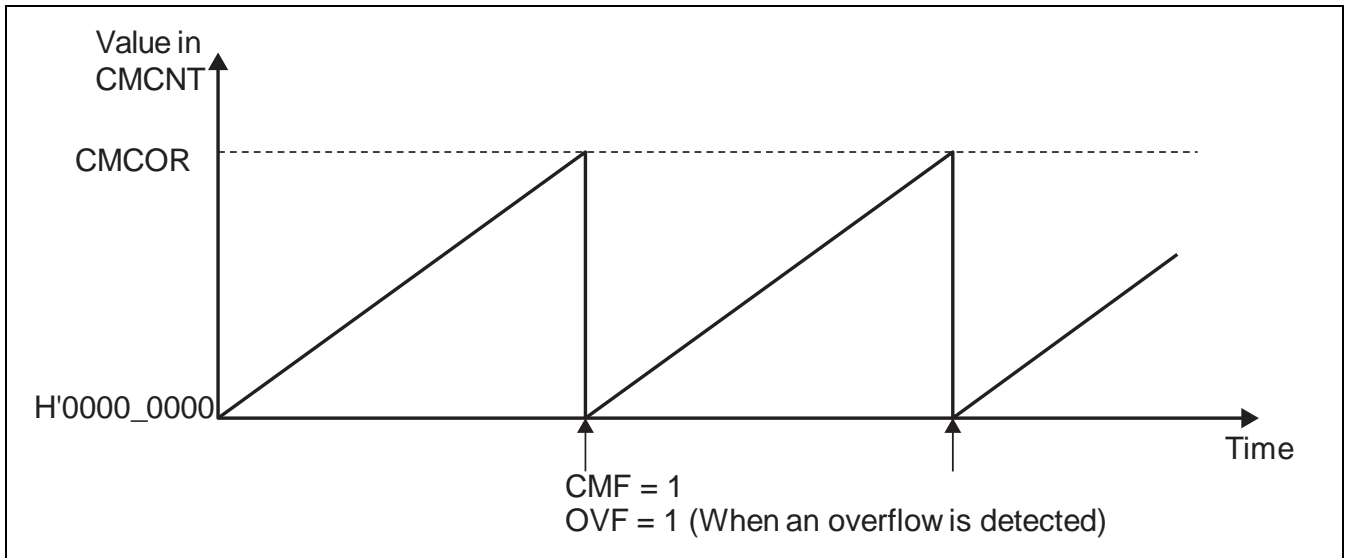


Figure 67.3 Counter Operation (Free-Running Operation)

67.3.2 Counter Size

In this module, the size of the counter can be selected from 16, or 32 bits. This is selected by the CMS bit in CMCSRn.

When the 16-bit size is selected, H'0000 should be used as upper 16 bits of write data to CMCORn. To detect an overflow interrupt, CMCORn must be set to H'0000_FFFF.

67.3.3 Timing for Counting by CMCNTn

In this module, the clock for the counter can be selected from among the following:

- RCLK: 1/1, 1/8, 1/32, and 1/128

The clock for the counter is selected by the CKS bits in CMCSRn. CMCNTn is incremented at the rising edge of the selected clock.

67.3.4 Internal Interrupt Request to CPU

By CMR bits in CMCSRn, internal interrupt request to the CPU at a compare match can be asserted.

To clear an internal interrupt request to the CPU in Free-Running Operation, the CMF bit should be set to 0. Set the CMF bit to 0 in the handling routine for the CMT0 interrupt.

To clear an internal interrupt request to the CPU with resuming count-up in One-Shot Operation, the CMF bit should be set to 0 while keeping 1 in STR0 bit in CMSTRn. Set the CMF bit to 0 while keeping 1 in STR0 bit in the handling routine for the CMT0 interrupt.

To clear an internal interrupt request to the CPU without resuming count-up in One-Shot Operation, the CMF bit should be set to 0 after setting STR0 bit in CMSTRn to 0. Set the CMF bit to 0 after setting STR0 bit to 0 in the handling routine for the CMT0 interrupt.

67.3.5 CMT0 Register Access

After writing to following registers, written data can be read 2 cycles in counter input clock (RCLK) after writing has finished. And it takes 2 cycles in counter input clock (RCLK), for reflecting written data to counter behavior.

CMCSRn: Bits CKS, CMM, CMS, CMR, DBGIVD

CMCORn: Bits 31 to 0

CMSTRn: Bit STR0

After writing to following registers, written data can be read 2 cycles in counter input clock (RCLK) after writing has finished. And it takes 2 cycles in counter input clock (RCLK), for reflecting written data to counter behavior.

And for following registers, write access is prohibited, while previously written data is under writing. Do not perform next write access, while CMCSRn.WRFLG is 1.

CMCNT*: Bits 31 to 0

67.3.6 Compare Match Flag Set/Clear Timing

The CMF bit in CMCSRn is set to 1 by the compare match signal generated when CMCORn and CMCNTn match. The compare match signal is generated upon the final state of the match (timing at which the CMCNTn value is updated to 0). Consequently, after CMCORn and CMCNTn match, a compare match signal will not be generated until a CMCNTn counter clock is input. Figure 67.4 shows the set timing of the CMF bit.

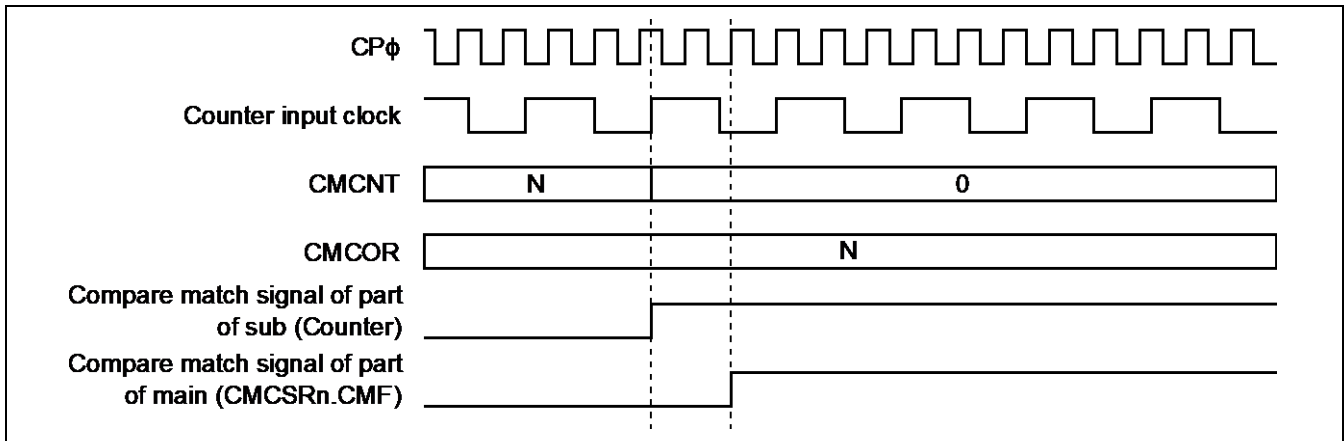


Figure 67.4 CMF Set Timing

Set the CMF bit as 0. The CMF flag is cleared immediately.

67.3.7 CMT0 Usage

Take the following steps to use the CMT0.

1. Clear the STR0 bit in CMSTRn to 0 to temporarily stop counting. (Confirm that RCLK was input more than 2 cycles.)
2. Write H'0000_0000 on CMCNTn.
3. Set counter size, compare match mode, type of counter clock and interrupt request and clear the bit of OVF and CMF in CMCSRn.
4. Set the value on CMCORn.
5. Confirm that CMCSRn.WRFLG is 0. If WRFLG is 1, wait until it'll be 0.
6. Start the counting by setting the STR0 bit in CMSTRn to 1. (Refer to section 67.3.5, CMT0 Register Access)

67.3.8 Module Stop/ CMCLKE Setting

When counter is not used, clock of this module can be stopped by CMCLKE register. Confirm that, following conditions are satisfied, before stopping counter's clock.

- CMSTRn.STR0 bit is 0
- CMCSRn.WRFLG is 0
- 3 or more cycles have passed in RCLK, since the last Register access

68. Compare Match Timer Type1 (CMT1)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

68.1 Overview

This section is written to introduce about CMT1/2/3, total include 24 channels, CMT2/3 are exactly same as CMT1, CMT1's usage are introduced here in, refer to 68.3 for CMT2/3 register address.

68.1.1 Features

- Eight channels
- 16 bits/32 bits/48 bits can be selected as counter size (bit-width).
- 48-bit constant registers and 48-bit up counters that can be written or read at any time.
- For channel 0 to 4, following twelve clocks can be selected as counter clock.
 - CPEX ϕ : 1/8, 1/32, 1/128, and 1/1
 - RCLK: 1/1, 1/8, 1/32, and 1/128
 - Pseudo 32 kHz: 1/1, 1/8, 1/32, and 1/128
- For channel 5 to 7, following four clocks can be selected as counter clock.
 - RCLK: 1/1, 1/8, 1/32, and 1/128
- One-shot operation or free-running operations are selectable.
- Compare match can be used as interrupt source.
- Support Module standby mode.
- RCLK-synchronous counter start/stop mode for channel 0
- Counter operation can be enabled or disabled at the time of debugging of CPU core using the debugging mode operation selector.

68.1.2 Block Diagram

Figure 68.1 shows a block diagram of the CMT1.

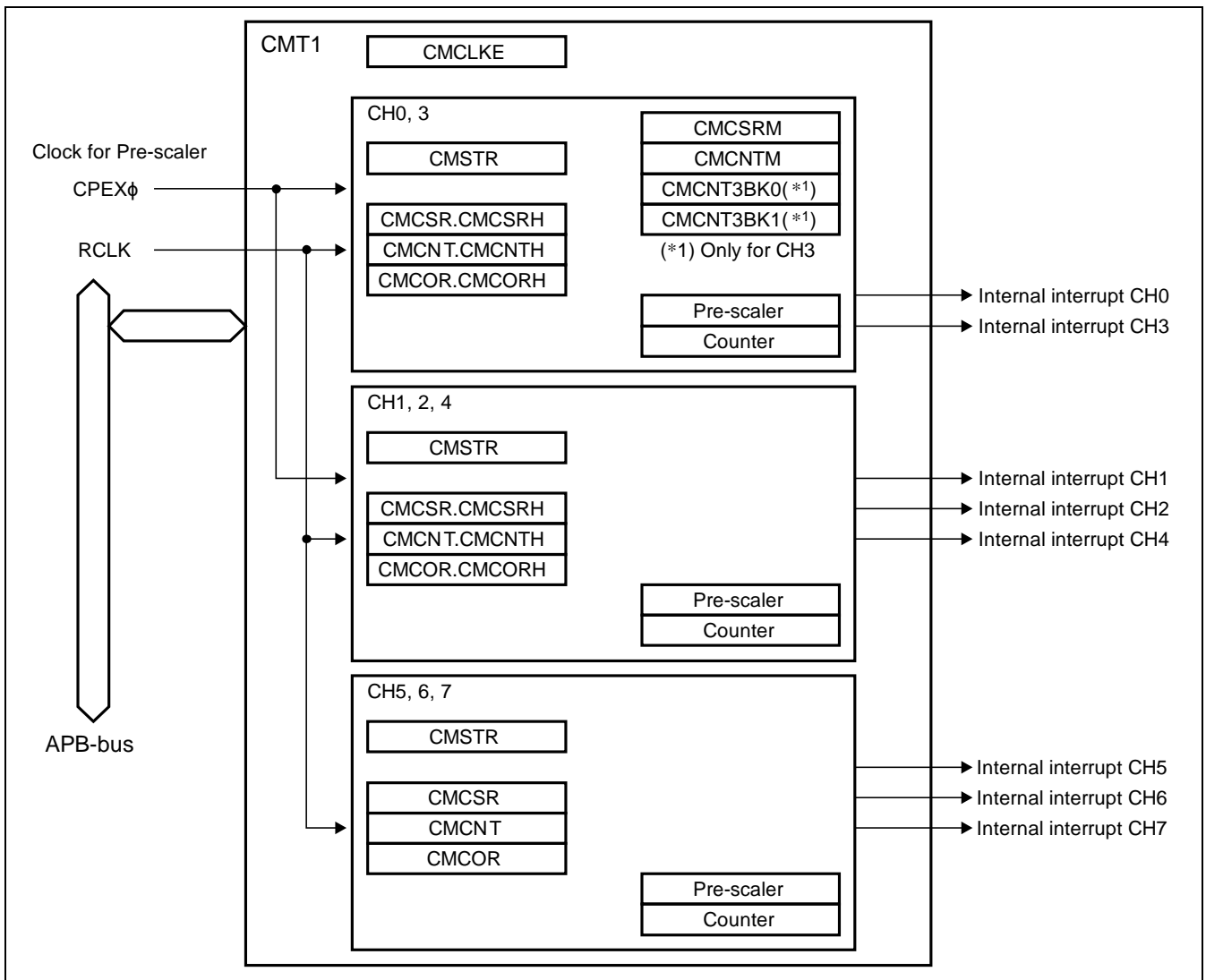


Figure 68.1 Block Diagram of CMT1

68.1.3 Register Configuration

Table 68.1 shows the CMT1/2/3 register configuration. Table 68.2 shows the register states in each operating mode. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined. Access the register as a longword (32 bits). Operation cannot be guaranteed if the register is not accessed as a longword.

Table 68.1 Register Configuration

| Register Name | CMT1/2/3 | Abbreviation | R/W | Address | Access Size |
|---|----------|--------------|-------|-------------|-------------|
| CLK enable register | CMT1 | CMCLKE | R/W | H'E613_1000 | 32 |
| | CMT2 | | | H'E614_1000 | |
| | CMT3 | | | H'E614_9000 | |
| Compare match timer start register 0 | CMT1 | CMSTR0 | R/W | H'E613_0000 | 32 |
| | CMT2 | | | H'E614_0000 | |
| | CMT3 | | | H'E614_8000 | |
| Compare match timer control/status register 0 | CMT1 | CMCSR0 | R/W*1 | H'E613_0010 | 32 |
| | CMT2 | | | H'E614_0010 | |
| | CMT3 | | | H'E614_8010 | |
| Compare match timer counter 0 | CMT1 | CMCNT0 | R/W | H'E613_0014 | 32 |
| | CMT2 | | | H'E614_0014 | |
| | CMT3 | | | H'E614_8014 | |
| Compare match timer constant register 0 | CMT1 | CMCOR0 | R/W | H'E613_0018 | 32 |
| | CMT2 | | | H'E614_0018 | |
| | CMT3 | | | H'E614_8018 | |
| Compare match timer control/status register H0 | CMT1 | CMCSRH0 | R/W | H'E613_0020 | 32 |
| | CMT2 | | | H'E614_0020 | |
| | CMT3 | | | H'E614_8020 | |
| Compare match timer counter H0 | CMT1 | CMCNTH0 | R/W | H'E613_0024 | 32 |
| | CMT2 | | | H'E614_0024 | |
| | CMT3 | | | H'E614_8024 | |
| Compare match timer constant register H0 | CMT1 | CMCORH0 | R/W | H'E613_0028 | 32 |
| | CMT2 | | | H'E614_0028 | |
| | CMT3 | | | H'E614_8028 | |
| Compare match timer match control/status register 0 | CMT1 | CMCSRM0 | R/W | H'E613_0040 | 32 |
| | CMT2 | | | H'E614_0040 | |
| | CMT3 | | | H'E614_8040 | |
| Compare match timer match counter 0 | CMT1 | CMCNTM0 | R | H'E613_0044 | 32 |
| | CMT2 | | | H'E614_0044 | |
| | CMT3 | | | H'E614_8044 | |
| Compare match timer start register 1 | CMT1 | CMSTR1 | R/W | H'E613_0100 | 32 |
| | CMT2 | | | H'E614_0100 | |
| | CMT3 | | | H'E614_8100 | |
| Compare match timer control/status register 1 | CMT1 | CMCSR1 | R/W*2 | H'E613_0110 | 32 |
| | CMT2 | | | H'E614_0110 | |
| | CMT3 | | | H'E614_8110 | |

| Register Name | CMT1/2/3 | Abbreviation | R/W | Address | Access Size |
|--|----------|--------------|-------|-------------|-------------|
| Compare match timer counter 1 | CMT1 | CMCNT1 | R/W | H'E613_0114 | 32 |
| | CMT2 | | | H'E614_0114 | |
| | CMT3 | | | H'E614_8114 | |
| Compare match timer constant register 1 | CMT1 | CMCOR1 | R/W | H'E613_0118 | 32 |
| | CMT2 | | | H'E614_0118 | |
| | CMT3 | | | H'E614_8118 | |
| Compare match timer control/status register H1 | CMT1 | CMCSRH1 | R/W | H'E613_0120 | 32 |
| | CMT2 | | | H'E614_0120 | |
| | CMT3 | | | H'E614_8120 | |
| Compare match timer counter H1 | CMT1 | CMCNTH1 | R/W | H'E613_0124 | 32 |
| | CMT2 | | | H'E614_0124 | |
| | CMT3 | | | H'E614_8124 | |
| Compare match timer constant register H1 | CMT1 | CMCORH1 | R/W | H'E613_0128 | 32 |
| | CMT2 | | | H'E614_0128 | |
| | CMT3 | | | H'E614_8128 | |
| Compare match timer start register 2 | CMT1 | CMSTR2 | R/W | H'E613_0200 | 32 |
| | CMT2 | | | H'E614_0200 | |
| | CMT3 | | | H'E614_8200 | |
| Compare match timer control/status register 2 | CMT1 | CMCSR2 | R/W*2 | H'E613_0210 | 32 |
| | CMT2 | | | H'E614_0210 | |
| | CMT3 | | | H'E614_8210 | |
| Compare match timer counter 2 | CMT1 | CMCNT2 | R/W | H'E613_0214 | 32 |
| | CMT2 | | | H'E614_0214 | |
| | CMT3 | | | H'E614_8214 | |
| Compare match timer constant register 2 | CMT1 | CMCOR2 | R/W | H'E613_0218 | 32 |
| | CMT2 | | | H'E614_0218 | |
| | CMT3 | | | H'E614_8218 | |
| Compare match timer control/status register H2 | CMT1 | CMCSRH2 | R/W | H'E613_0220 | 32 |
| | CMT2 | | | H'E614_0220 | |
| | CMT3 | | | H'E614_8220 | |
| Compare match timer counter H2 | CMT1 | CMCNTH2 | R/W | H'E613_0224 | 32 |
| | CMT2 | | | H'E614_0224 | |
| | CMT3 | | | H'E614_8224 | |
| Compare match timer constant register H2 | CMT1 | CMCORH2 | R/W | H'E613_0228 | 32 |
| | CMT2 | | | H'E614_0228 | |
| | CMT3 | | | H'E614_8228 | |
| Compare match timer start register 3 | CMT1 | CMSTR3 | R/W | H'E613_0300 | 32 |
| | CMT2 | | | H'E614_0300 | |
| | CMT3 | | | H'E614_8300 | |

| Register Name | CMT1/2/3 | Abbreviation | R/W | Address | Access Size |
|---|----------|--------------|-------|-------------|-------------|
| Compare match timer control/status register 3 | CMT1 | CMCSR3 | R/W*2 | H'E613_0310 | 32 |
| | CMT2 | | | H'E614_0310 | |
| | CMT3 | | | H'E614_8310 | |
| Compare match timer counter 3 | CMT1 | CMCNT3 | R/W | H'E613_0314 | 32 |
| | CMT2 | | | H'E614_0314 | |
| | CMT3 | | | H'E614_8314 | |
| Compare match timer constant register 3 | CMT1 | CMCOR3 | R/W | H'E613_0318 | 32 |
| | CMT2 | | | H'E614_0318 | |
| | CMT3 | | | H'E614_8318 | |
| Compare match timer control/status register H3 | CMT1 | CMCSRH3 | R/W | H'E613_0320 | 32 |
| | CMT2 | | | H'E614_0320 | |
| | CMT3 | | | H'E614_8320 | |
| Compare match timer counter H3 | CMT1 | CMCNTH3 | R/W | H'E613_0324 | 32 |
| | CMT2 | | | H'E614_0324 | |
| | CMT3 | | | H'E614_8324 | |
| Compare match timer constant register H3 | CMT1 | CMCORH3 | R/W | H'E613_0328 | 32 |
| | CMT2 | | | H'E614_0328 | |
| | CMT3 | | | H'E614_8328 | |
| Compare match timer counter 3 backup 0 | CMT1 | CMCNT3BK0 | R | H'E613_0330 | 32 |
| | CMT2 | | | H'E614_0330 | |
| | CMT3 | | | H'E614_8330 | |
| Compare match timer counter 3 backup 1 | CMT1 | CMCNT3BK1 | R | H'E613_0334 | 32 |
| | CMT2 | | | H'E614_0334 | |
| | CMT3 | | | H'E614_8334 | |
| Compare match timer match control/status register 3 | CMT1 | CMCSR3M3 | R/W | H'E613_0340 | 32 |
| | CMT2 | | | H'E614_0340 | |
| | CMT3 | | | H'E614_8340 | |
| Compare match timer match counter 3 | CMT1 | CMCNTM3 | R | H'E613_0344 | 32 |
| | CMT2 | | | H'E614_0344 | |
| | CMT3 | | | H'E614_8344 | |
| Compare match timer start register 4 | CMT1 | CMSTR4 | R/W | H'E613_0400 | 32 |
| | CMT2 | | | H'E614_0400 | |
| | CMT3 | | | H'E614_8400 | |
| Compare match timer control/status register 4 | CMT1 | CMCSR4 | R/W*2 | H'E613_0410 | 32 |
| | CMT2 | | | H'E614_0410 | |
| | CMT3 | | | H'E614_8410 | |
| Compare match timer counter 4 | CMT1 | CMCNT4 | R/W | H'E613_0414 | 32 |
| | CMT2 | | | H'E614_0414 | |
| | CMT3 | | | H'E614_8414 | |

| Register Name | CMT1/2/3 | Abbreviation | R/W | Address | Access Size |
|--|----------|--------------|-------|-------------|-------------|
| Compare match timer constant register 4 | CMT1 | CMCOR4 | R/W | H'E613_0418 | 32 |
| | CMT2 | | | H'E614_0418 | |
| | CMT3 | | | H'E614_8418 | |
| Compare match timer control/status register H4 | CMT1 | CMCSRH4 | R/W | H'E613_0420 | 32 |
| | CMT2 | | | H'E614_0420 | |
| | CMT3 | | | H'E614_8420 | |
| Compare match timer counter H4 | CMT1 | CMCNTH4 | R/W | H'E613_0424 | 32 |
| | CMT2 | | | H'E614_0424 | |
| | CMT3 | | | H'E614_8424 | |
| Compare match timer constant register H4 | CMT1 | CMCORH4 | R/W | H'E613_0428 | 32 |
| | CMT2 | | | H'E614_0428 | |
| | CMT3 | | | H'E614_8428 | |
| Compare match timer start register 5 | CMT1 | CMSTR5 | R/W | H'E613_0500 | 32 |
| | CMT2 | | | H'E614_0500 | |
| | CMT3 | | | H'E614_8500 | |
| Compare match timer control/status register 5 | CMT1 | CMCSR5 | R/W*2 | H'E613_0510 | 32 |
| | CMT2 | | | H'E614_0510 | |
| | CMT3 | | | H'E614_8510 | |
| Compare match timer counter 5 | CMT1 | CMCNT5 | R/W | H'E613_0514 | 32 |
| | CMT2 | | | H'E614_0514 | |
| | CMT3 | | | H'E614_8514 | |
| Compare match timer constant register 5 | CMT1 | CMCOR5 | R/W | H'E613_0518 | 32 |
| | CMT2 | | | H'E614_0518 | |
| | CMT3 | | | H'E614_8518 | |
| Compare match timer start register 6 | CMT1 | CMSTR6 | R/W | H'E613_0600 | 32 |
| | CMT2 | | | H'E614_0600 | |
| | CMT3 | | | H'E614_8600 | |
| Compare match timer control/status register 6 | CMT1 | CMCSR6 | R/W*2 | H'E613_0610 | 32 |
| | CMT2 | | | H'E614_0610 | |
| | CMT3 | | | H'E614_8610 | |
| Compare match timer counter 6 | CMT1 | CMCNT6 | R/W | H'E613_0614 | 32 |
| | CMT2 | | | H'E614_0614 | |
| | CMT3 | | | H'E614_8614 | |
| Compare match timer constant register 6 | CMT1 | CMCOR6 | R/W | H'E613_0618 | 32 |
| | CMT2 | | | H'E614_0618 | |
| | CMT3 | | | H'E614_8618 | |
| Compare match timer start register 7 | CMT1 | CMSTR7 | R/W | H'E613_0700 | 32 |
| | CMT2 | | | H'E614_0700 | |
| | CMT3 | | | H'E614_8700 | |

| Register Name | CMT1/2/3 | Abbreviation | R/W | Address | Access Size |
|---|----------|--------------|-------|-------------|-------------|
| Compare match timer control/status register 7 | CMT1 | CMCSR7 | R/W*2 | H'E613_0710 | 32 |
| | CMT2 | | | H'E614_0710 | |
| | CMT3 | | | H'E614_8710 | |
| Compare match timer counter 7 | CMT1 | CMCNT7 | R/W | H'E613_0714 | 32 |
| | CMT2 | | | H'E614_0714 | |
| | CMT3 | | | H'E614_8714 | |
| Compare match timer constant register 7 | CMT1 | CMCOR7 | R/W | H'E613_0718 | 32 |
| | CMT2 | | | H'E614_0718 | |
| | CMT3 | | | H'E614_8718 | |

Notes: 1. Except for bit [15:14] / [12:11] that are R/WC0 register.

2. Except for bit [15:14] that are R/WC0 register.

Table 68.2 Register States in Each Operating Mode

| Register Abbreviation | Power-on Reset/Software Reset | Module Standby |
|------------------------------|--------------------------------------|-----------------------|
| CMCLKE | Initialized | Retained |
| CMSTR0 | Initialized | Retained |
| CMCSR0 | Initialized | Retained |
| CMCNT0 | Initialized | Retained |
| CMCOR0 | Initialized | Retained |
| CMCSRH0 | Initialized | Retained |
| CMCNTH0 | Initialized | Retained |
| CMCORH0 | Initialized | Retained |
| CMCSRMO | Initialized | Retained |
| CMCNTMO | Initialized | Retained |
| CMSTR1 | Initialized | Retained |
| CMCSR1 | Initialized | Retained |
| CMCNT1 | Initialized | Retained |
| CMCOR1 | Initialized | Retained |
| CMCSRH1 | Initialized | Retained |
| CMCNTH1 | Initialized | Retained |
| CMCORH1 | Initialized | Retained |
| CMSTR2 | Initialized | Retained |
| CMCSR2 | Initialized | Retained |
| CMCNT2 | Initialized | Retained |
| CMCOR2 | Initialized | Retained |
| CMCSRH2 | Initialized | Retained |
| CMCNTH2 | Initialized | Retained |
| CMCORH2 | Initialized | Retained |
| CMSTR3 | Initialized | Retained |
| CMCSR3 | Initialized | Retained |
| CMCNT3 | Initialized | Retained |
| CMCOR3 | Initialized | Retained |
| CMCSRH3 | Initialized | Retained |
| CMCNTH3 | Initialized | Retained |
| CMCORH3 | Initialized | Retained |
| CMCNT3BK0 | Initialized | Retained |
| CMCNT3BK1 | Initialized | Retained |
| CMCSRMO3 | Initialized | Retained |
| CMCNTMO3 | Initialized | Retained |
| CMSTR4 | Initialized | Retained |
| CMCSR4 | Initialized | Retained |
| CMCNT4 | Initialized | Retained |
| CMCOR4 | Initialized | Retained |
| CMCSRH4 | Initialized | Retained |
| CMCNTH4 | Initialized | Retained |
| CMCORH4 | Initialized | Retained |

| Register Abbreviation | Power-on Reset/Software Reset | Module Standby |
|------------------------------|--------------------------------------|-----------------------|
| CMSTR5 | Initialized | Retained |
| CMCSR5 | Initialized | Retained |
| CMCNT5 | Initialized | Retained |
| CMCOR5 | Initialized | Retained |
| CMSTR6 | Initialized | Retained |
| CMCSR6 | Initialized | Retained |
| CMCNT6 | Initialized | Retained |
| CMCOR6 | Initialized | Retained |
| CMSTR7 | Initialized | Retained |
| CMCSR7 | Initialized | Retained |
| CMCNT7 | Initialized | Retained |
| CMCOR7 | Initialized | Retained |

68.2 Register Description

[Legend for Register Description]

Initial value: Register value after a reset.

—: Undefined value

R/W: Bit or field is readable and writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R: Read-only. The write value should always be initial value.

All access to register is made in longword units.

68.2.1 CLK Enable Register (CMCLKE)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMCLKE is a 32bits register, which specify clock supply to each channel. When there are unused channels, set '0' as this register, to stop supplying clock to the channel. It is prohibited to stop clock supply, while counter is working.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----------|----------|----------|----------|----------|----------|----------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | Ch7 clke | Ch6 clke | Ch5 clke | Ch4 clke | Ch3 clke | Ch2 clke | Ch1 clke | Ch0 clke |
| Initial value: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 8 | — | All 1 | R | Reserved These bits are always read as 1. The write value should always be 1. |
| 7 | Ch7clke | B'1 | R/W | 0: Clock isn't supplied to ch7 1: Clock is supplied to ch7 |
| 6 | Ch6clke | B'1 | R/W | 0: Clock isn't supplied to ch6 1: Clock is supplied to ch6 |
| 5 | Ch5clke | B'1 | R/W | 0: Clock isn't supplied to ch5 1: Clock is supplied to ch5 |
| 4 | Ch4clke | B'1 | R/W | 0: Clock isn't supplied to ch4 1: Clock is supplied to ch4 |
| 3 | Ch3clke | B'1 | R/W | 0: Clock isn't supplied to ch3 1: Clock is supplied to ch3 |
| 2 | Ch2clke | B'1 | R/W | 0: Clock isn't supplied to ch2 1: Clock is supplied to ch2 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | Ch1cke | B'1 | R/W | 0: Clock isn't supplied to ch1 1: Clock is supplied to ch1 |
| 0 | Ch0cke | B'1 | R/W | 0: Clock isn't supplied to ch0 1: Clock is supplied to ch0 |

68.2.2 Compare Match Timer Start Registers 0 to 7 (CMSTRn)

CMSTRn (n = 0 to 7) is a 32-bit register which specify the operation of compare match timer counter (CMCNTn). Refer to section 68.3.5, Register Access, for register value update timing.

(1) CMSTR0

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|--------|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | STR0RS | — | — | — | — | — | — | — | STR0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 9 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 8 | STR0RS | B'0 | R/W | RCLK-Synchronous Counter Start/Stop Mode Select 0: Normal operation Channel 0 starts or stops counting, immediately after data is written to STR0. 1: RCLK-synchronous counter start/stop mode Channel 0 starts or stops counting on detecting an RCLK rising edge, after data is written to STR0. |
| 7 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | STR0 | B'0 | R/W | Count Start 0 These bits specify start/halt of compare match timer counter 0 (CMCNT0). 0: CMCNT0 halts 1: CMCNT0 start counting |

(2) CMSTRn (n = 1 to 4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | STR0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | STR0 | B'0 | R/W | Count Start 0 These bits specify start/halt of compare match timer counter (CMCNTn). 0: CMCNTn halts 1: CMCNTn starts counting |

(3) CMSTRn (n = 5 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | STR0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 31 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | STR0 | B'1 | R/W | Count Start 0 These bits specify start/halt of compare match timer counter (CMCNTn). 0: CMCNTn halts 1: CMCNTn starts counting |

68.2.3 Compare Match Timer Control/Status Registers 0 to 7 (CMCSRn)

CMCSRn (n = 0 to 7) is a 32-bit register that indicates the occurrence of compare match, enable interrupt and set the counter input clock.

Do not change bits other than the CMF and OVF bits, while compare match timer counter (CMCNTn) is under counting.

Note: * Refer to section 68.3.5, Register Access, for the note regarding writing to or reading from the following bits.

- CMCSR0.CH0SSIE
- CMCSRn.CMS
- CMCSRn.CMM
- CMCSRn.CMR [1:0]
- CMCSRn.DBGIVD
- CMCSRn.CKS [2:0]

(1) CMCSR0

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-----------|-------------|-------------|-------------|-----|-----|----|----|----------|------------|----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CMF | OVF | WRFL G | CH0 STTF | CH0 STPF | CH0 SSIE | CMS | CMM | — | — | CMR[1:0] | DBG IVD | CKS[2:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| R/W: | R/WC0 | R/WC0 | R | R/WC0 | R/WC0 | R/W | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | CMF | B'0 | R/WC0 | Compare Match Flag This flag indicates whether values of the compare match timer counter (CMCNT0) and compare match timer constant register (CMCOR0) have matched or not. Software cannot write 1 to the bit. When one-shot is selected for the counter operation, counting resumes by clearing this bit. 0: CMCNT0 and CMCOR0 values have not matched [Clearing conditions] <ul style="list-style-type: none"> • Write 0 to this bit 1: CMCNT0 and CMCOR0 values have matched Note: Only 0 can be written to clear the flag. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------|---|
| 14 | OVF | B'0 | R/WC0 | <p>Overflow Flag</p> <p>This flag indicates whether the compare match timer counter (CMCNT0) has overflowed or not. Software cannot write 1 to this bit.</p> <p>0: CMCNT0 has not overflowed [Clearing conditions]</p> <ul style="list-style-type: none"> Write 0 to this bit <p>1: CMCNT0 has overflowed</p> <p>Note: Only 0 can be written to clear the flag.</p> |
| 13 | WRFLG | B'0 | R | <p>Write State Flag</p> <p>Write access to CMCNT0 is prohibited, while this bit is 1. Further behavior is not guaranteed, if data is written while this bit is 1.</p> <p>This bit indicates CMCNT0 is in synchronization period for setting previously written data.</p> <p>Confirm that this flag is 0, before writing to CMCNT0.</p> |
| 12 | CH0STTF | B'0 | R/WC0 | <p>Channel 0 Start Flag</p> <p>When RCLK-synchronous channel 0 counter start/stop mode is selected, this flag indicates whether the counter in channel 0 started on detecting an RCLK rising edge after 1 was written to the STR0 bit in CMSR.</p> <p>0: Channel 0 counter has not started. [Clearing condition]</p> <ul style="list-style-type: none"> Write 0 to CH0STTF. <p>1: Channel 0 counter has started.</p> <p>Note: Only 0 can be written to clear the flag.</p> |
| 11 | CH0STPF | B'0 | R/WC0 | <p>Channel 0 Stop Flag</p> <p>When RCLK-synchronous channel 0 counter start/stop mode is selected, this flag indicates whether the counter in channel 0 stopped on detecting an RCLK rising edge after 1 was written to the STR0 bit in CMSR.</p> <p>0: Channel 0 counter has not stopped. [Clearing condition]</p> <ul style="list-style-type: none"> Write 0 to CH0STPF. <p>1: Channel 0 counter has stopped.</p> <p>Note: Only 0 can be written to clear the flag.</p> |
| 10 | CH0SSIE | B'0 | R/W | <p>Channel 0 Start/Stop Interrupt Enable</p> <p>When RCLK-synchronous channel 0 counter start/stop mode is selected, this bit enables or disables an interrupt due to the start or stop of the counter in channel 0.</p> <p>0: Disables an interrupt due to start or stop of channel 0 counter. 1: Enables an interrupt due to start or stop of channel 0 counter.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 9 | CMS | B'0 | R/W | <p>Compare Match Timer Counter Size</p> <p>This bit and CMCSRH0.CMSH select whether the compare match timer counter 0 (CMCNTH0 [15:0], and CMCNT0 [31:0]) is used as a 16-bit counter, a 32-bit counter, or a 48-bit counter.</p> <p>This bit also specifies valid size of the compare match timer constant register 0 (CMCORH0 [15:0], and CMCOR0 [31:0]). CMCSRH0[9].CMSH, and CMCSR0[9].CMS:</p> <p>B'1x: Operates as a 48-bit counter B'00: Operates as a 32-bit counter B'01: Operates as a 16-bit counter</p> |
| 8 | CMM | B'1 | R/W | <p>Compare Match Mode</p> <p>Specify operation mode of the counter.</p> <p>0: One-shot operation 1: Free-running operation</p> |
| 7 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 6 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 5, 4 | CMR[1:0] | B'00 | R/W | <p>Compare Match Request</p> <p>These bits enable or disable internal interrupt request in a compare match.</p> <p>B'00: Disables internal interrupt request B'01: Setting prohibited B'10: Enables an internal interrupt request B'11: Setting prohibited</p> |
| 3 | DBGIVD | B'1 | R/W | <p>Debug Mode Operation Select</p> <p>Sets the counter operation in debugging mode.</p> <p>0: Stops the counter operation in debugging mode. 1: Continues the counter operation even in debugging mode.</p> |
| 2 to 0 | CKS[2:0] | B'111 | R/W | <p>Clock Select</p> <p>These bits and CMCSRH0.CKSH select the clock input to CMCNT0. When the count start bit (STR0) for the corresponding channel is set to 1, CMCNT0 begins incrementing with the clock selected by these bits.</p> <p>CMCSRH0.CKSH[0]+CMCSR0.CKS[2:0]:</p> <p>B'x000: CPEXϕ/8 B'x001: CPEXϕ/32 B'x010: CPEXϕ/128 B'x011: CPEXϕ/1 B'0100: RCLK/8 B'0101: RCLK/32 B'0110: RCLK/128 B'0111: RCLK/1 B'1100: Pseudo 32 kHz/8 B'1101: Pseudo 32 kHz/32 B'1110: Pseudo 32 kHz/128 B'1111: Pseudo 32 kHz/1</p> |

(2) CMCSRn (n = 1 to 4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|----|----|----|-----|-----|----|----|----------|---------|----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CMF | OVF | WRFLG | — | — | — | CMS | CMM | — | — | CMR[1:0] | DBG IVD | CKS[2:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| R/W: | R/WC0 | R/WC0 | R | R | R | R | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | CMF | B'0 | R/WC0 | Compare Match Flag This flag indicates whether values of the compare match timer counter (CMCNTn) and compare match timer constant register (CMCORn) have matched or not. Software cannot write 1 to the bit. When one-shot is selected for the counter operation, counting resumes by clearing this bit. 0: CMCNTn and CMCORn values have not matched [Clearing conditions] <ul style="list-style-type: none"> Write 0 to this bit 1: CMCNTn and CMCORn values have matched Note: Only 0 can be written to clear the flag. |
| 14 | OVF | B'0 | R/WC0 | Overflow Flag This flag indicates whether the compare match timer counter (CMCNTn) has overflowed or not. Software cannot write 1 to this bit. 0: CMCNTn has not overflowed [Clearing conditions] <ul style="list-style-type: none"> Write 0 to this bit 1: CMCNTn has overflowed Note: Only 0 can be written to clear the flag. |
| 13 | WRFLG | B'0 | R | Write State Flag Write access to CMCNTn is prohibited, while this bit is 1. Further behavior is not guaranteed, if data is written while this bit is 1. This bit indicates CMCNTn is in synchronization period for setting previously written data. Confirm that this flag is 0, before writing to CMCNTn. |
| 12 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 9 | CMS | B'0 | R/W | <p>Compare Match Timer Counter Size</p> <p>This bit and CMCSRn.CMSH specify whether the compare match timer counter (CMCNTHn[15:0] and CMCNTn[31:0]) is used as a 16-bit counter, a 32-bit counter, or a 48-bit counter.</p> <p>This setting becomes the valid size for the compare match timer constant register (CMCORHn[15:0], CMCORn[31:0]).</p> <p>CMCSRn[9].CMSH+CMCSRn[9].CMS:</p> <p>B'1x: Operates as a 48-bit counter B'00: Operates as a 32-bit counter B'01: Operates as a 16-bit counter</p> |
| 8 | CMM | B'1 | R/W | <p>Compare Match Mode</p> <p>Specify counter operation mode.</p> <p>0: One-shot operation 1: Free-running operation</p> |
| 7, 6 | — | All 0 | R | <p>Reserved</p> <p>These bits are read as 0. The write value should always be 0.</p> |
| 5, 4 | CMR[1:0] | B'00 | R/W | <p>Compare Match Request</p> <p>These bits enable or disable internal interrupt request in a compare match.</p> <p>B'00: Disables internal interrupt request B'01: Setting prohibited B'10: Enables an internal interrupt request B'11: Setting prohibited</p> |
| 3 | DBGIVD | B'1 | R/W | <p>Debug Mode Operation Select</p> <p>Sets the counter operation in debugging mode.</p> <p>0: Stops the counter operation in debugging mode. 1: Continues the counter operation even in debugging mode.</p> |
| 2 to 0 | CKS[2:0] | B'111 | R/W | <p>Clock Select</p> <p>These bits and CMCSRn.CKSH specify the input clock to CMCNTn. When the count start bit (STR0) for the corresponding channel is set to 1, CMCNTn begins incrementing with the clock selected by these bits.</p> <p>CMCSRn[0].CKSH[0], CMCSRn[2:0].CKS:</p> <p>B'x000: CPEXϕ/8 B'x001: CPEXϕ/32 B'x010: CPEXϕ/128 B'x011: CPEXϕ/1 B'0100: RCLK/8 B'0101: RCLK/32 B'0110: RCLK/128 B'0111: RCLK/1 B'1100: Pseudo 32 kHz/8 B'1101: Pseudo 32 kHz/32 B'1110: Pseudo 32 kHz/128 B'1111: Pseudo 32 kHz/1</p> |

(3) CMCSRn (n = 5 to 7)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|----|----|----|-----|-----|----|----|----------|---------|----------|-----|-----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CMF | OVF | WRFLG | — | — | — | CMS | CMM | — | — | CMR[1:0] | DBG IVD | CKS[2:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| R/W: | R/WC0 | R/WC0 | R | R | R | R | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | CMF | B'0 | R/WC0 | Compare Match Flag This flag indicates whether values of the compare match timer counter (CMCNTn) and compare match timer constant register (CMCORn) have matched or not. Software cannot write 1 to the bit. When one-shot is selected for the counter operation, counting resumes by clearing this bit. 0: CMCNTn and CMCORn values have not matched [Clearing conditions] <ul style="list-style-type: none"> Write 0 to this bit 1: CMCNTn and CMCORn values have matched Note: Only 0 can be written to clear the flag. |
| 14 | OVF | B'0 | R/WC0 | Overflow Flag This flag indicates whether the compare match timer counter (CMCNTn) has overflowed or not. Software cannot write 1 to this bit. 0: CMCNTn has not overflowed [Clearing conditions] <ul style="list-style-type: none"> Write 0 to this bit 1: CMCNTn has overflowed Note: Only 0 can be written to clear the flag. |
| 13 | WRFLG | B'0 | R | Write State Flag Write access to CMCNTn is prohibited, while this bit is 1. Further behavior is not guaranteed, if data is written while this bit is 1. This bit indicates CMCNTn is in synchronization period for setting previously written data. Confirm that this flag is 0, before writing to CMCNTn. |
| 12 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 9 | CMS | B'0 | R/W | <p>Compare Match Timer Counter Size</p> <p>Selects whether the compare match timer counter (CMCNTn) is used as a 16-bit counter or a 32-bit counter.</p> <p>This setting becomes the valid size for the compare match timer constant register (CMCORn).</p> <p>0: Operates as a 32-bit counter. 1: Operates as a 16-bit counter.</p> |
| 8 | CMM | B'1 | R/W | <p>Compare Match Mode</p> <p>Specify counter operation mode.</p> <p>0: One-shot operation 1: Free-running operation</p> |
| 7, 6 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 5, 4 | CMR[1:0] | B'00 | R/W | <p>Compare Match Request</p> <p>These bits enable or disable internal interrupt request in a compare match.</p> <p>B'00: Disables internal interrupt request B'01: Setting prohibited B'10: Enables an internal interrupt request B'11: Setting prohibited</p> |
| 3 | DBGIVD | B'1 | R/W | <p>Debug Mode Operation Select</p> <p>Sets the counter operation in debugging mode.</p> <p>0: Stops the counter operation in debugging mode. 1: Continues the counter operation even in debugging mode.</p> |
| 2 to 0 | CKS[2:0] | B'111 | R/W | <p>Clock Select</p> <p>These bits select the clock input to CMCNTn. When the count start bit (STR0) in CMSTRn is set to 1, CMCNTn begins incrementing with the clock selected by these bits.</p> <p>B'000: Setting prohibited B'001: Setting prohibited B'010: Setting prohibited B'011: Setting prohibited B'100: RCLK/8 B'101: RCLK/32 B'110: RCLK/128 B'111: RCLK/1</p> |

68.2.4 Compare Match Timer Control/Status Registers H0 to 4 (CMCSRHn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMCSRHn is a 32-bit register which specify the counter size and input clocks.

Note: * For write to or read from CMCSRHn.CMSH and CMCSRHn.CKSH, refer to section 68.3.5, Register Access.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|------|----|----|----|----|----|----|----|----|------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | CMSH | — | — | — | — | — | — | — | — | CKSH |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | CMSH | B'0 | R/W | Compare Match Timer Counter Size This bit and CMCSRn.CMS specify whether the compare match timer counter (CMCNTHn[15:0] and CMCNTn[31:0]) is used as a 16-bit counter, a 32-bit counter, or a 48-bit counter. This bit also specify valid size for the compare match timer constant register (CMCORHn [15:0], CMCORn [31:0]). CMCSRHn.CMSH[9], CMCSRn.CMS[9]: B'1x: Operates as a 48-bit counter B'00: Operates as a 32-bit counter B'01: Operates as a 16-bit counter |
| 8 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

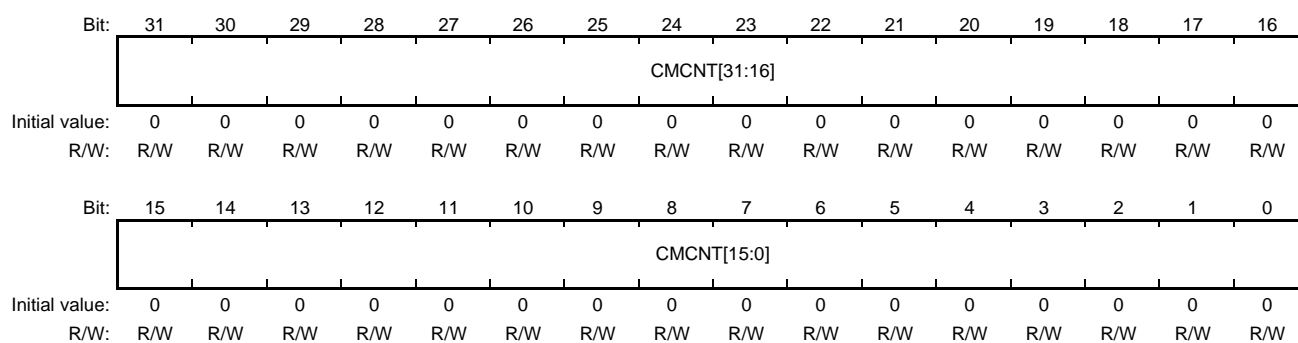
| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | CKSH | B'0 | R/W | <p>Clock Select</p> <p>This bit and CMCSRn.CKS [2:0] select the clock input to CMCNT. When the count start bit (STRO) for the corresponding channel is set to 1, CMCNTn begins incrementing with the clock selected by these bits.</p> <p>CMCSRHn.CKSH[0], CMCSRn.CKS[2:0]:</p> <p>B'x000: CPEXϕ/8</p> <p>B'x001: CPEXϕ/32</p> <p>B'x010: CPEXϕ/128</p> <p>B'x011: CPEXϕ/1</p> <p>B'0100: RCLK/8</p> <p>B'0101: RCLK/32</p> <p>B'0110: RCLK/128</p> <p>B'0111: RCLK/1</p> <p>B'1100: Pseudo 32kHz/8</p> <p>B'1110: Pseudo 32kHz/32</p> <p>B'1101: Pseudo 32kHz/128</p> <p>B'1111: Pseudo 32kHz/1</p> |

68.2.5 Compare Match Timer Counters 0 to 7 (CMCNTn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMCNTn (n = 0 to 7) is a 32-bit register which is used as an up-counter of each channel.

To specify counter operation, set compare match timer control/status register n (CMCSRn), before starting operation of corresponding channel. When the 16-bit counter operation is selected by the CMCSRH.CMSH and CMCSR.CMS bits, bits 31 to 16 of this register is invalid. When data is written to this register in 16-bit mode, write H'0000 to the upper 16 bit.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|---|
| 31 to 0 | CMCNT [31:0] | H'0000_0000 | R/W | <p>Compare match timer counter bit31 to 0</p> <p>Note: For access to this register, refer to section 68.3.5, Register Access.</p> |

68.2.6 Compare Match Timer Counters H 0 to 4 (CMCNTHn)

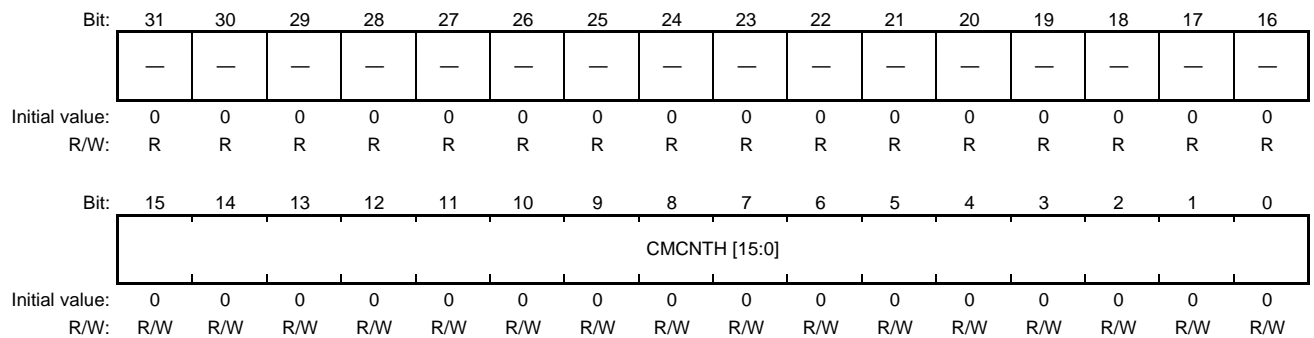
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMCNTHn (n = 0 to 4) is a 32-bit register which is used as an up-counter for channel0 to 4.

To specify counter operation, set compare match timer control/status register n (CMCSRn/CMCSRHn), before starting operation of corresponding channel. When the 48-bit counter operation is selected by the CMCSRHn.CMSH and CMCSRn.CMS bits, bits 15 to 0 of this register become valid as the upper 16-bit of 48-bit counter. The value written to this register is reflected to counter behavior, when lower data is written to CMCNTn. Write the upper data to this register first. The value is not reflected to counter’s behavior, if data is not written to CMCNTn later.

When reading the value of 48-bit counter, upper 16-bit can be read out by reading bits 15 to 0 of this register. At the same time of the reading, the value of 48-bit counter is stored to the read-buffer. After reading CMCNTHn, lower 32-bit can be read out by reading CMCNTn.

Note: If CMCNTn is read out before reading CMCNTHn, the read out value of 48-bit counter may not be correct.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | CMCNTH [15:0] | H'0000 | R/W | Compare match timer counter H bit15 to 0 Note: For access to this register, refer to section 68.3.5, Register Access. |

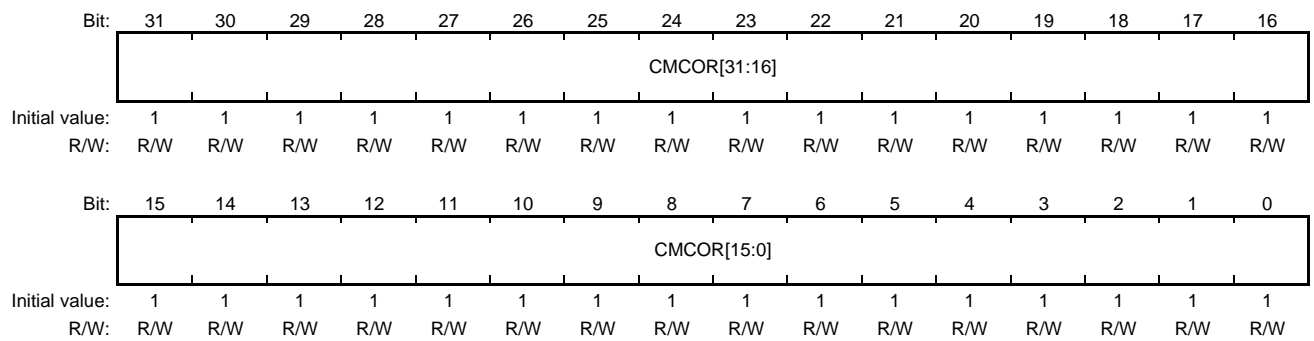
68.2.7 Compare Match Timer Constant Registers 0 to 7 (CMCORN)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMCORN is a 32-bit register which specify the compare match period of CMCNTn for each channel.

When the 16-bit counter operation is selected by the CMCSRn.CMSH and CMCSRn.CMS bits, bits 31 to 16 of this register is invalid. When data is written to this register in 16-bit operation mode, write H'0000 as upper 16bit.

An overflow is detected when CMCNTn is cleared to 0 and this register is H'FFFF_FFFF (when the 16-bit counter operation it is H'0000_FFFF).



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------|---------------|-----|--|
| 31 to 0 | CMCOR [31:0] | H'FFFF_FFFF | R/W | Compare match timer constant register bit31 to 0 Note: For access to this register, refer to section 68.3.5, Register Access. |

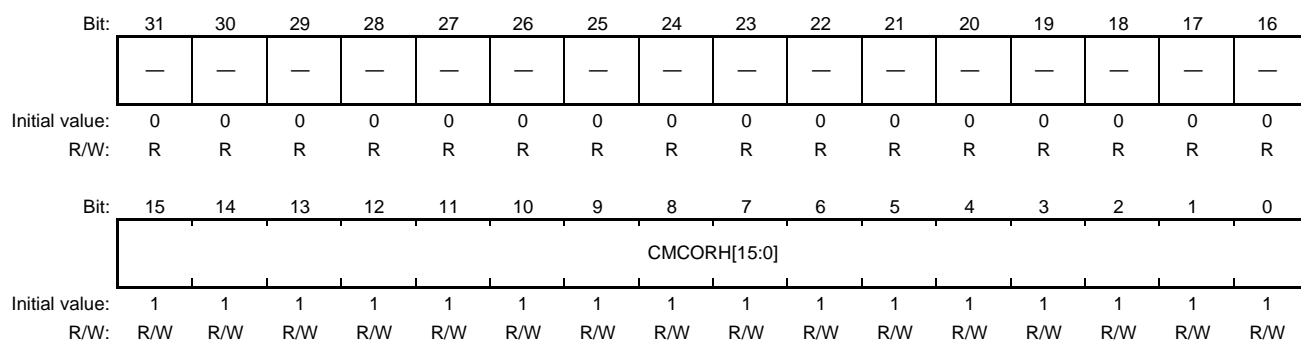
68.2.8 Compare Match Timer Constant Registers H0 to 4 (CMCORHn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMCORHn is a 32-bit register which specify the compare match period with CMCNTHn for channel0 to 4.

When the 48-bit counter operation is selected by the CMCSRn.CMSH and CMCSRn.CMS bits, bits 15 to 0 of this register become valid as the upper 16-bit of 48-bit counter.

An overflow is detected when CMCNTHn is cleared to 0 and this register is H'FFFF.



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 0 | CMCORH [15:0] | H'FFFF | R/W | Compare match timer constant register H bit15 to 0 Note: For access to this register, refer to section 68.3.5, Register Access. |

68.2.9 Compare Match Timer Counter 3 Backup 0 (CMCNT3BK0)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMCNT3BK0 is a 32-bit register which stores a copy of the CMCNT3 value immediately after the counter in channel 0 stops in RCLK-synchronous channel 0 counter start/stop mode.

| | | | | | | | | | | | | | | | | |
|----------------|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CMCNT3BK0[31:16] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CMCNT3BK0[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 0 | CMCNT3BK0 | H'0000_0000 | R | Compare match timer counter 3 backup 0 bit31 to 0 [31:0] |

68.2.10 Compare Match Timer Counter 3 Backup 1 (CMCNT3BK1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMCNT3BK1 is a 32-bit register which stores a copy of the CMCNT3 value immediately after the counter in channel 0 starts in RCLK-synchronous channel 0 counter start/stop mode.

| | | | | | | | | | | | | | | | | |
|----------------|------------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | CMCNT3BK1[31:16] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CMCNT3BK1[15:0] | | | | | | | | | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 0 | CMCNT3BK1 | H'0000_0000 | R | Compare match timer counter 3 backup 1 bit31 to 0 [31:0] |

68.2.11 Compare Match Timer Match Control/Status Registers 0 and 3 (CMCSRm)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMCSRm is a 32-bit register which resets to compare match timer match counter, and sets the counter start/halt. This register is only present in channels 0 and 3.

| | | | | | | | | | | | | | | | | |
|----------------|-------|----|----|----|----|----|----|----|----|----|----|----|----|----|--------|----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | WRFLG | — | — | — | — | — | — | — | — | — | — | — | — | — | CMPCLR | CMPSTART |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R/W | R/W |

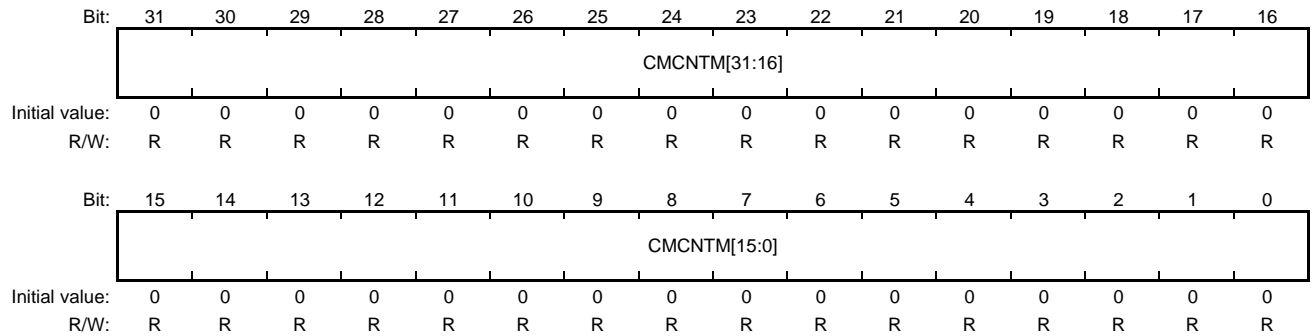
| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 | WRFLG | B'0 | R | Write state flag. When this Bit is 1, CMCNTMn cannot be cleared. After CMCNTMn is cleared, this bit become 1 for a while (*), for indicating synchronization period. Needs to confirm this flag is 0 or wait the period enough when continuously clearing. Note: * In the maximum, "Counter input clock 6 cycle " <ul style="list-style-type: none"> Counter input clock CMCSRn.CKS2 = 0: CPEXϕ CMCSRn.CKS2 = 1: RCLK |
| 14 to 2 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 1 | CMPCLR | B'0 | R/W | Counter Clear [When writing] 0: No operation 1: Clears counter When CMPCLR and CMPSTART are written 1 at the same time, the counter starts after clearing the counter. [When reading] This bit is always read as 0. |
| 0 | CMPSTART | B'0 | R/W | Count Start This bit specifies start/halt of the compare match timer match counter (CMCNTMn) of each channel. 0: CMCNTMn Halts 1: CMCNTMn Starts counting |

68.2.12 Compare Match Timer Match Counters 0 and 3 (CMCNTMn)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMCNTMn is a 32-bit register which is used as an up-counter. This register is only present in channels 0 and 3.

A counter operation is set by the compare match timer match control/status register (CMCSRm). After the counter CMCNTn starts, this counter increases whenever at a compare match.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 31 to 0 | CMCNTM [31:0] | H'0000_0000 | R | Compare match timer match counter bit31 to 0 |

68.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

68.3.1 Counter Operation

The CMT1 starts the operation of the counter by writing H'1 to the STR0 bit in CMSTRn after each register has been set. Complete all of the settings before starting the operation. Do not change the register settings other than clearing flag bits, while the compare match timer (CMCNTn) is under operation.

The counter operates in one of two ways.

- One-Shot Operation

One-shot operation is selected by setting the CMM bit in CMCSRn to H'0. When the value in CMCNTn matches the value in CMCORn, the value in CMCNTn is cleared to H'0000_0000 and the CMF bit in CMCSRn is set to H'1. Counting by CMCNTn stops after it has been cleared.

To detect an overflow interrupt, set the value in CMCORn to H'FFFF_FFFF. When the value in CMCNTn matches the value in CMCORn, CMCNTn is cleared to H'0000_0000 and the CMF and OVF bits in CMCSRn are set to H'1.

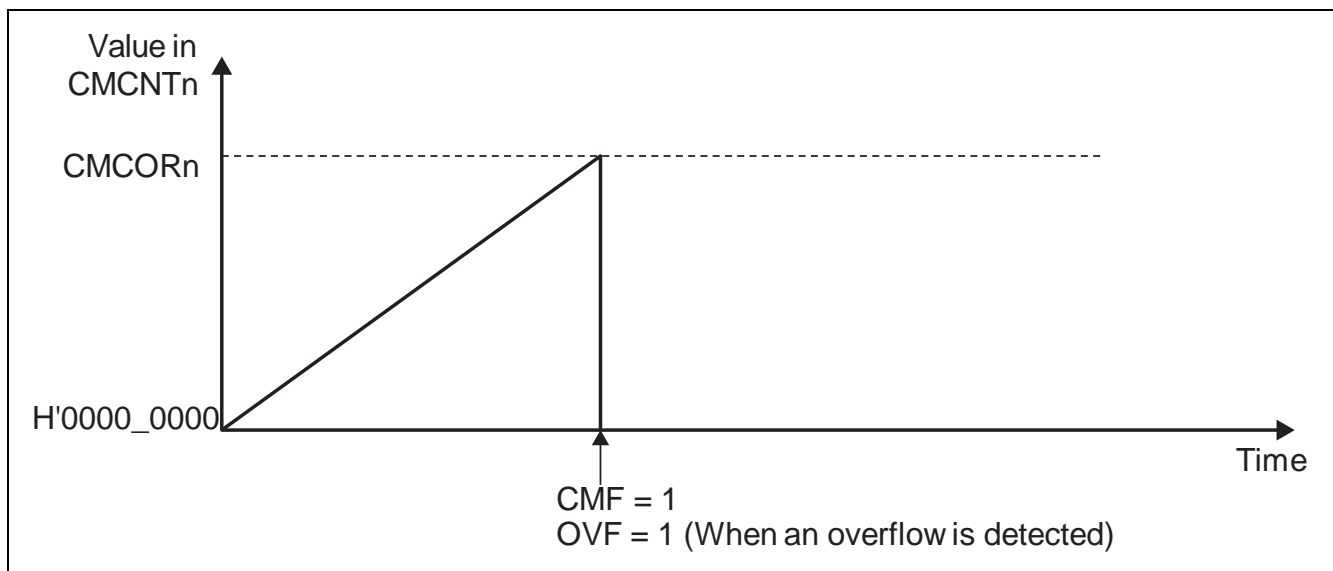


Figure 68.2 Counter Operation (One-Shot Operation)

- Free-Running Operation

Free-running operation is selected by setting the CMM bit in CMCSRn to H'1. When the value in CMCNTn matches the value in CMCORn, CMCNTn is cleared to H'0000_0000 and the CMF bit in CMCSRn is set to H'1. CMCNTn resumes counting-up after it has been cleared.

To detect an overflow interrupt, set CMCORn to H'FFFF_FFFF. When the values in CMCNTn and CMCORn match, CMCNTn is cleared to H'0000_0000 and the CMF and OVF bits in CMCSRn are set to H'1.

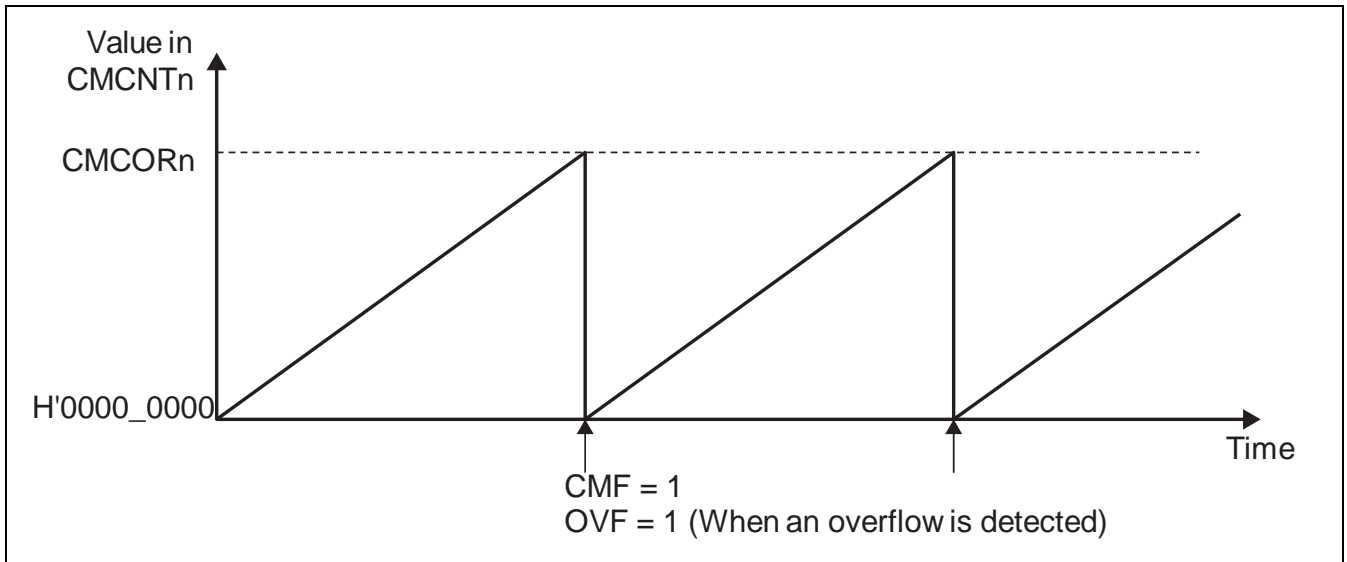


Figure 68.3 Counter Operation (Free-Running Operation)

68.3.2 Counter Size

In this module, the size of the counter can be selected from 16, 32, or 48 bits. This is selected by the CMS bit in CMCSRn and CMSH bit in CMCSRHn.

When the 16-bit/32-bit size is selected, upper 32 bits/16 bits of CMCORn is ignored. To detect an overflow interrupt, the value 1 must be set to valid bits of CMCORn.

(for example, H'0000_0000_FFFF, H'0000_FFFF_FFFF)

68.3.3 Timing for Counting by CMCNTn

In this module, the clock for the counter can be selected from among the following:

| | | |
|------------------|--------------------|---------------------------|
| Channels 0 to 4: | CPEX ϕ clock: | 1/1, 1/8, 1/32, and 1/128 |
| | RCLK: | 1/1, 1/8, 1/32, and 1/128 |
| | Pseudo 32-kHz: | 1/1, 1/8, 1/32 and 1/128 |
| Channel 5 to 7: | RCLK: | 1/1, 1/8, 1/32, and 1/128 |

For details of CPEX ϕ and RCLK, refer to section 11, Clock Pulse Generator (CPG).

Pseudo 32-kHz is a clock, which is generated inside this module. Refer to section 68.3.8, Pseudo 32-kHz Counter.

The clock for the counter is selected by the CKS bits in CMCSRn and CKSH bit in CMCSRHn. CMCNTn is incremented at the rising edge of the selected clock.

68.3.4 Internal Interrupt Request to CPU

By CMR bits in CMCSRn, internal interrupt request to the CPU at a compare match can be asserted.

To clear an internal interrupt request to the CPU in Free-Running Operation, the CMF bit should be set to H'0. Set the CMF bit to H'0 in the handling routine for the CMT1 interrupt.

To clear an internal interrupt request to the CPU with resuming count-up in One-Shot Operation, the CMF bit should be set to H'0 while keeping H'1 in STR0 bit in CMSTRn. Set the CMF bit to H'0 while keeping H'1 in STR0 bit in the handling routine for the CMT1 interrupt.

To clear an internal interrupt request to the CPU without resuming count-up in One-Shot Operation, the CMF bit should be set to H'0 after setting STR0 bit in CMSTRn to H'0. Set the CMF bit to H'0 after setting STR0 bit to H'0 in the handling routine for the CMT1 interrupt.

68.3.5 Register Access

After writing to following registers, written data can be read 2 cycles in counter input clock (RCLK) after writing has finished. And it takes 2 cycles in counter input clock (RCLK), for reflecting written data to counter behavior.

- CMCSRn: Bits CH0SSIE(ch0), CMS, CMM, CMR[1:0], DBGIVD, CKS[2:0]
- CMCORn, CMCORHn
- CMSTRn: Bit STR0RS, STR0
- CMCSRm: Bit CMPCLR, CMPSTART

After writing to following registers, written data can be read 2 cycles in counter input clock (RCLK or CPEX ϕ) after writing has finished. And it takes 2 cycles in counter input clock (RCLK or CPEX ϕ), for reflecting written data to counter behavior. After writing data to this register, next write access is prohibited, until CMCSRn.WRFLG become 0.

- CMCNTn, CMCNTHn

68.3.6 Compare Match Flag Set/Clear Timing

The CMF bit in CMCSRn is set to 1 by the compare match signal generated when CMCORn and CMCNTn match. The compare match signal is generated upon the final state of the match (timing at which the CMCNTn value is updated to H'0000). Consequently, after CMCORn and CMCNTn match, a compare match signal will not be generated until a CMCNTn counter clock is input. Figure 68.4 shows the set timing of the CMF bit.

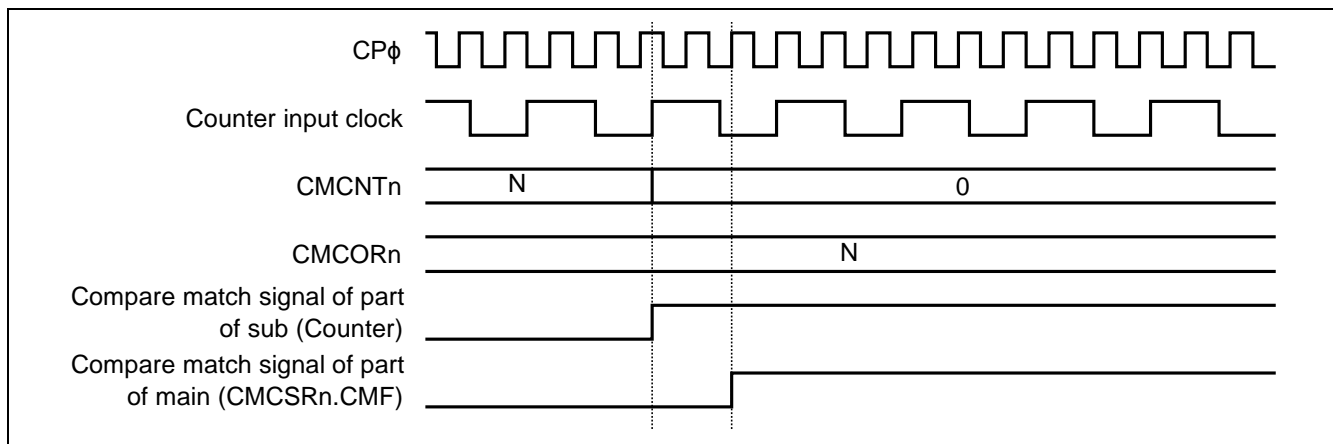


Figure 68.4 CMF Set Timing

Set the CMF bit to 0. The CMF flag is cleared immediately.

68.3.7 RCLK-Synchronous Counter Start/Stop Mode in Channel 0

The CMT1 provides the RCLK-synchronous counter start/stop mode in channel 0, in which channels 0 and 3 can be used together for time measurement.

Figure 68.5 shows a flow chart for preparation.

Figure 68.6 shows a flow chart for starting measurement.

Figure 68.7 shows a flow chart for stopping measurement.

Figure 68.8 shows a timing chart when measurement starts.

Figure 68.9 shows a timing chart when measurement stops.

Figure 68.10 shows how to calculate the time measured by the counters.

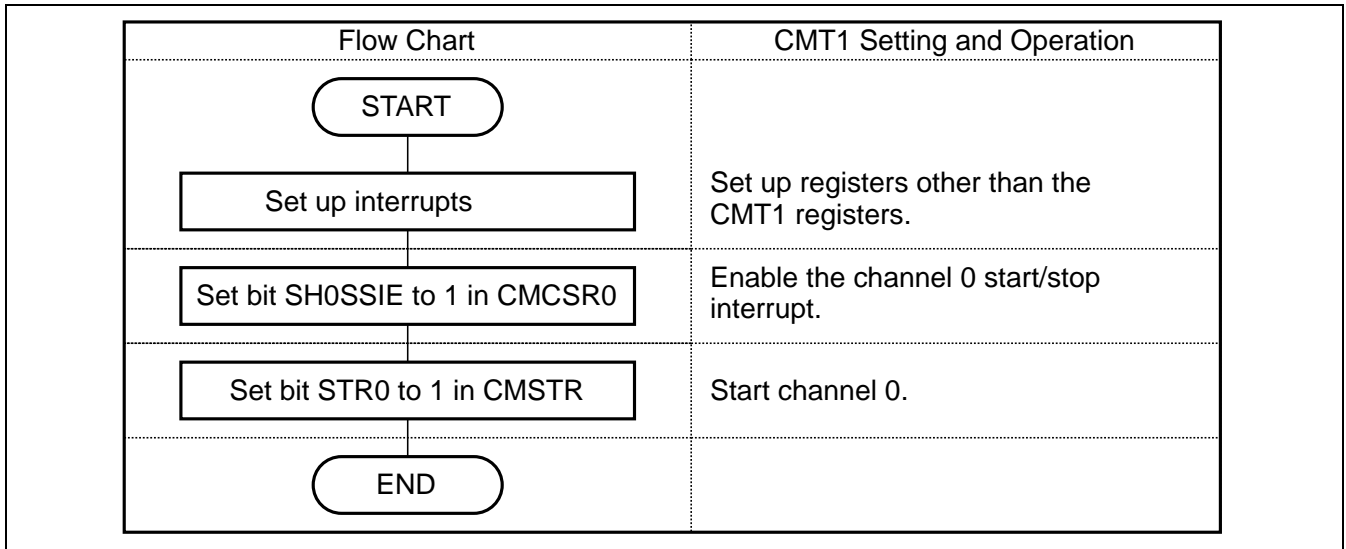


Figure 68.5 Flow Chart for Preparation

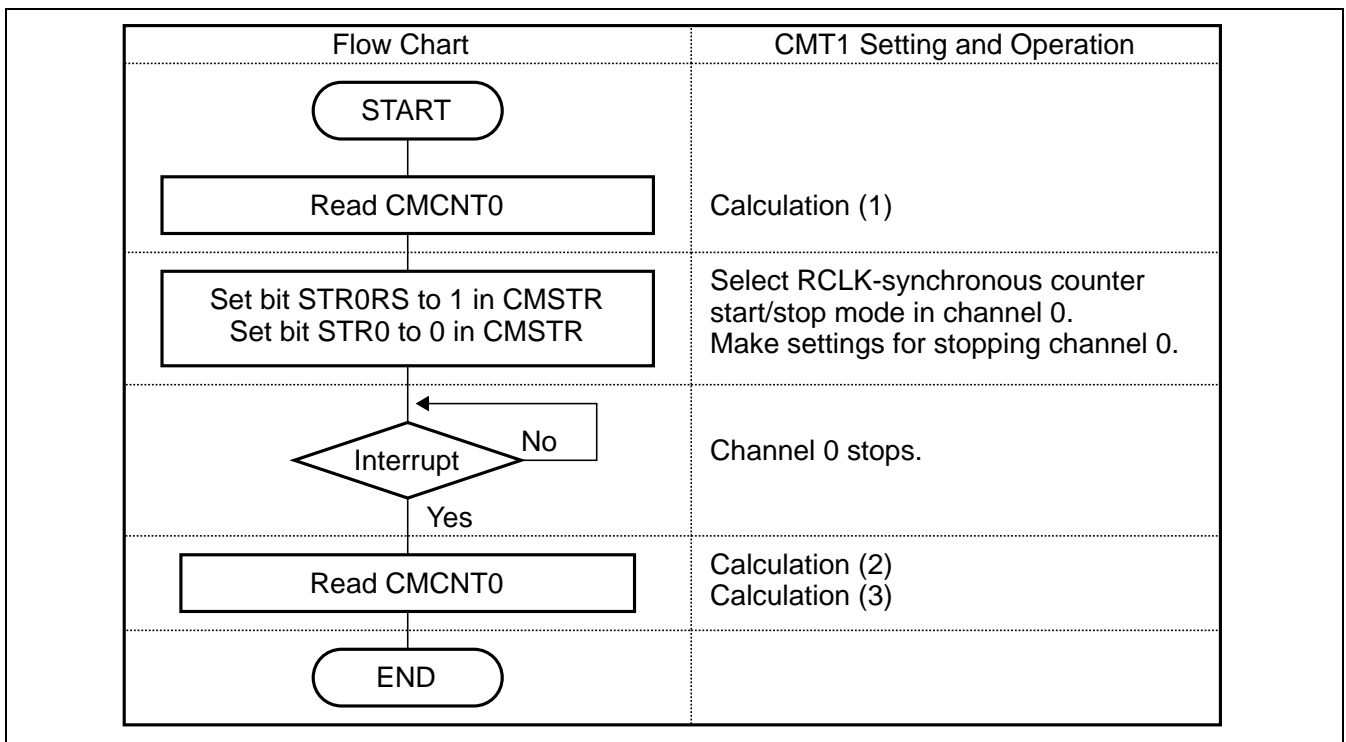


Figure 68.6 Flow Chart for Starting Measurement

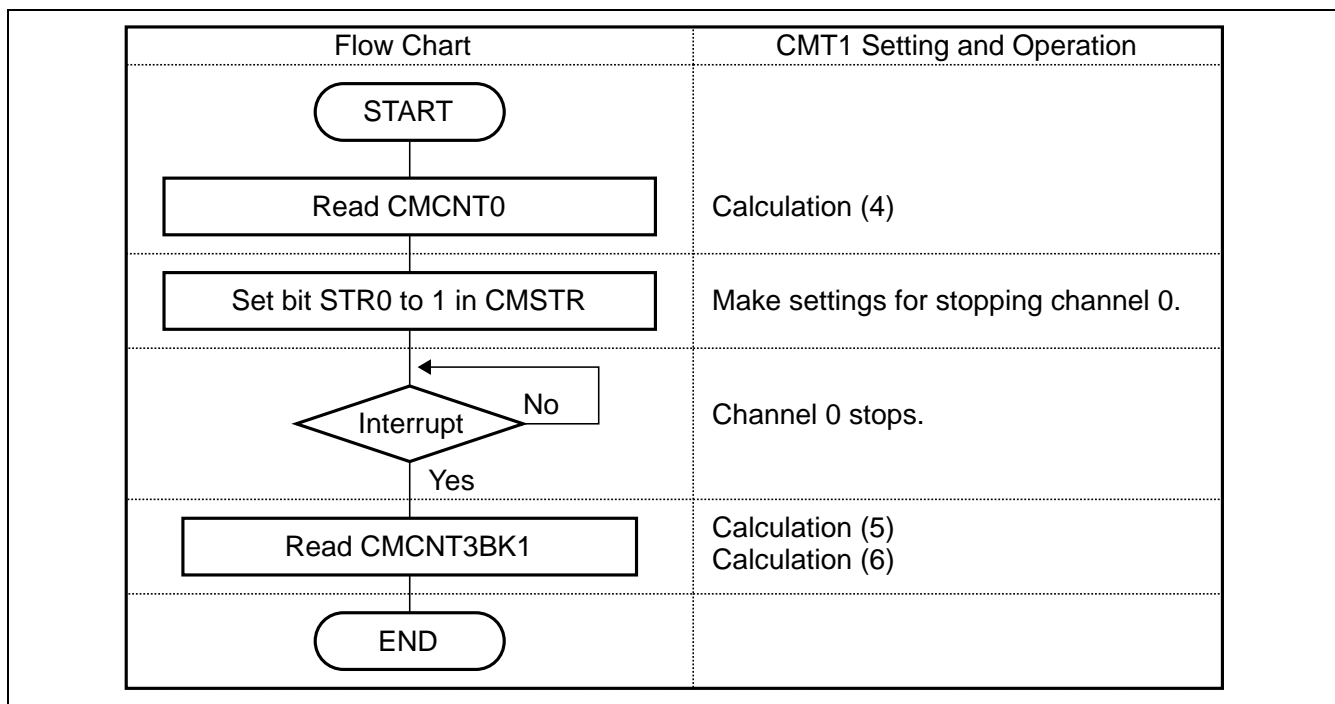


Figure 68.7 Flow Chart for Stopping Measurement

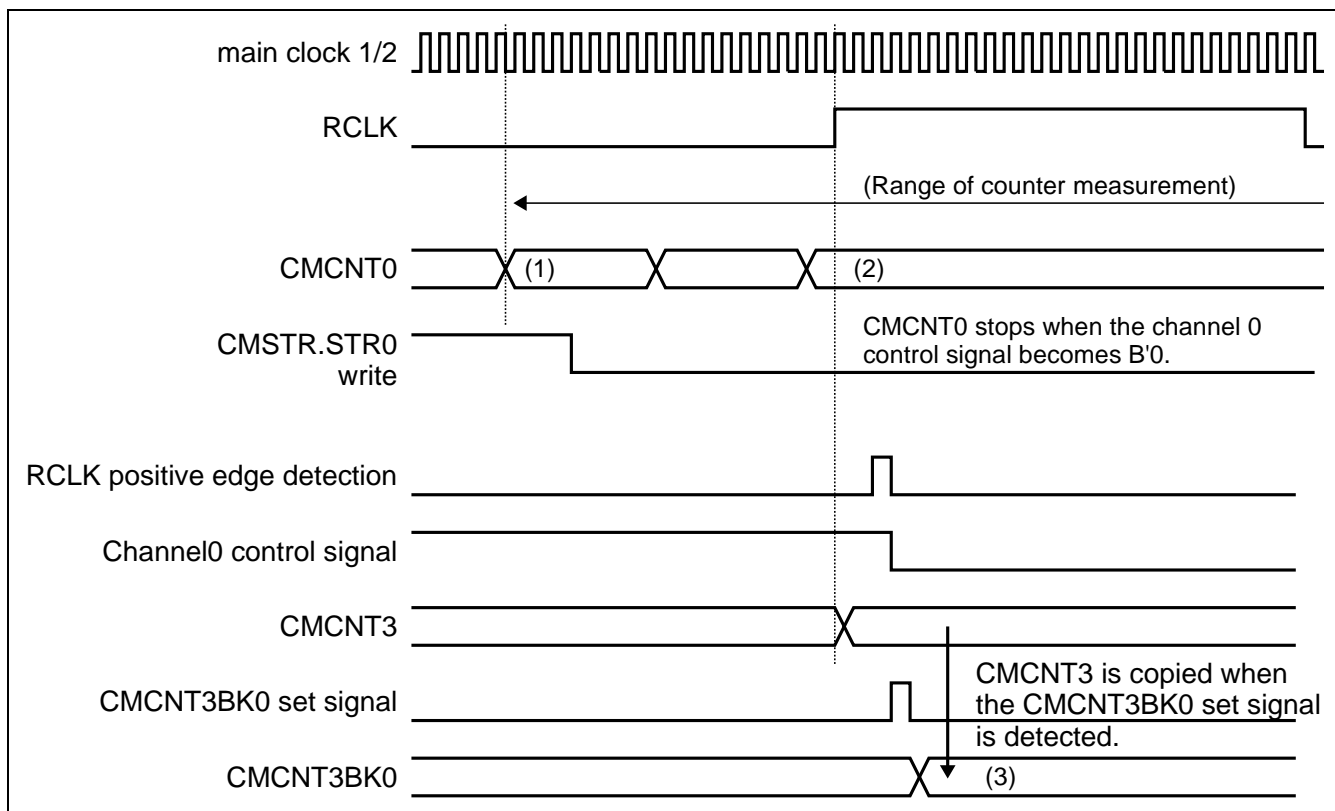


Figure 68.8 Timing Chart When Measurement Starts

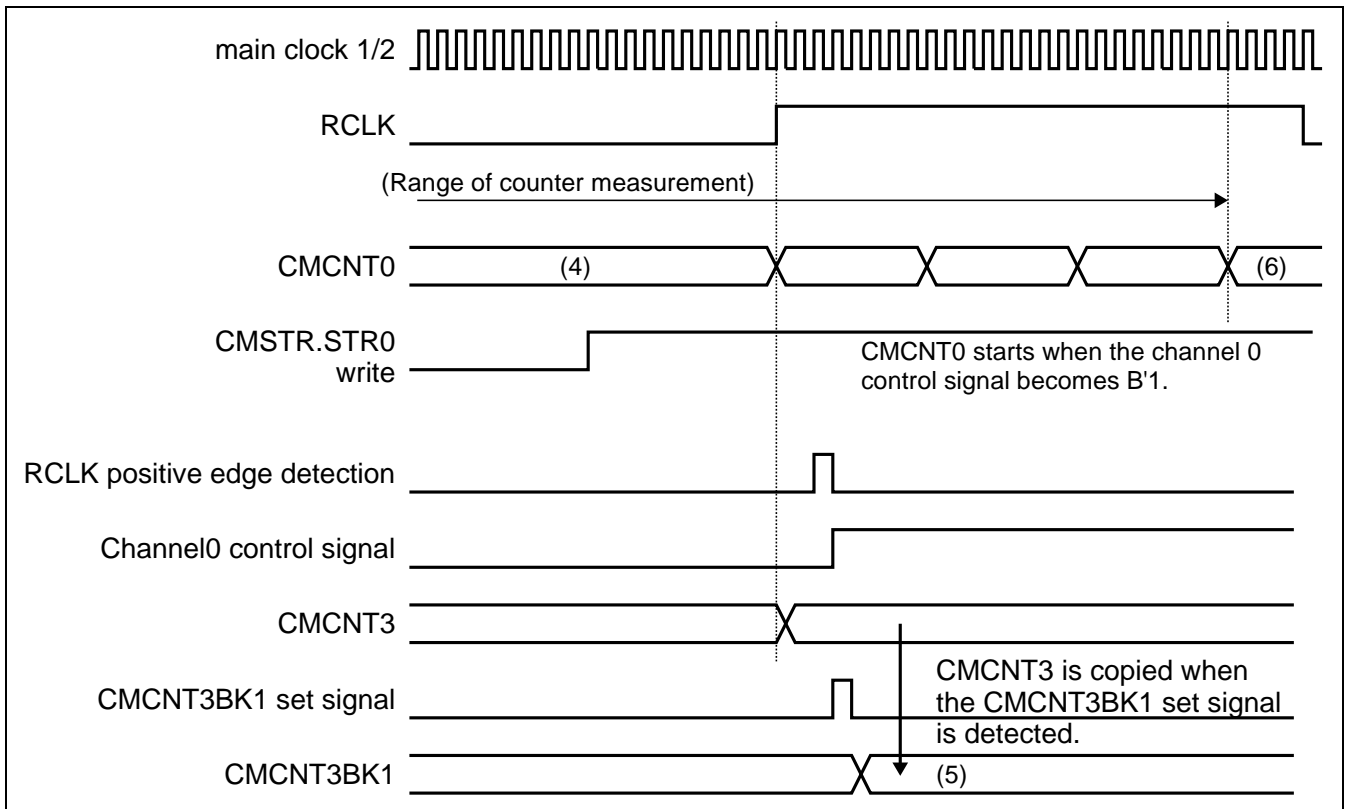


Figure 68.9 Timing Chart When Measurement Stops

$$(((2) - (1)) + ((6) - (4) - 1)) \times T_{apc} + ((5) - (3)) \times T_r$$

T_{apc} : Cycle of main clock 1/2
 T_r : Cycle of RCLK

Figure 68.10 Calculation of Time Measured by Counters

68.3.8 Pseudo 32-kHz Counter

Pseudo 32-kHz clock is created inside this module, based on RCLK by Pseudo 32-kHz Counter. By removing 3 edge of RCLK every 128 cycle, Pseudo 32-kHz clock is generated. By this edge removing, frequency of this clock becomes 128/131 of RCLK.

68.3.9 CMT1 Usage

Take the following steps to use the CMT1.

1. Clear the STR0 bit in CMSTRn to 0 to temporarily stop counting.
(Confirm that Counter input clock was input more than 2 cycles.)
2. Write H'0000_0000 on CMCNTn.
3. Set counter size, compare match mode, kind of count clock and interrupt request and clear the bit of OVF and CMF in CMCSRn.
4. Set the value on CMCORn.
5. Confirm that CMCSRn.WRFLG is 0.
If WRFLG is 1, wait until it'll be 0.
6. Start the counting by setting the STR0 bit in CMSTRn to 1 (refer to section 68.3.5, Register Access).

68.3.10 Module Stop/ CMCLKE Setting

When counter is not used, clock of this module can be stopped by CMCLKE register. Confirm that, following conditions are satisfied, before stopping counter's clock.

- CMSTRn.STR0 bit is 0
- CMCSRn.WRFLG is 0
- 3 or more cycles have passed in Counter input clock since the last Register access

69. Timer Unit (TMU)

RZ/G2H**RZ/G2M V1.3****RZ/G2M V3.0****RZ/G2N****RZ/G2E**

69.1 Overview

[RZ/G2H]

This LSI includes a 32-bit timer unit (TMU) with 15 channels (channels 0 to 14).

Channels 0 to 2, 3 to 5, 6 to 8, 9 to 11 and 12 to 14 are grouped into timers 0, 1, 2, 3, 4 respectively.

For channels 0 to 2, CP ϕ is provided as a base clock. For channels 3 to 11, S3D2 ϕ is provided. For channels 12 to 14, S0D6 ϕ is provided.

[RZ/G2M V1.3, RZ/G2M V3.0]

This LSI includes a 32-bit timer unit (TMU) with 15 channels (channels 0 to 14).

Channels 0 to 2, 3 to 5, 6 to 8, 9 to 11 and 12 to 14 are grouped into timers 0, 1, 2, 3, 4 respectively.

For channels 0 to 2, CP ϕ is provided as a base clock. For channels 3 to 11, S3D2 ϕ is provided. For channels 12 to 14, S0D6 ϕ is provided.

[RZ/G2N]

This LSI includes a 32-bit timer unit (TMU) with 15 channels (channels 0 to 14).

Channels 0 to 2, 3 to 5, 6 to 8, 9 to 11 and 12 to 14 are grouped into timers 0, 1, 2, 3, 4 respectively.

For channels 0 to 2, CP ϕ is provided as a base clock. For channels 3 to 11, S3D2 ϕ is provided. For channels 12 to 14, S0D6 ϕ is provided.

[RZ/G2E]

This LSI includes a 32-bit timer unit (TMU) with 15 channels (channels 0 to 14).

Channels 0 to 2, 3 to 5, 6 to 8, 9 to 11 and 12 to 14 are grouped into timers 0, 1, 2, 3, 4 respectively.

For channels 0 to 2, CP ϕ is provided as a base clock. For channels 3 to 11, S3D2C ϕ is provided. For channels 12 to 14, S0D6C ϕ is provided.

69.1.1 Features

TMU has the following features:

[RZ/G2H]

- Auto-reload type 32-bit down counter is provided for each channel.
- Input capture function provided: Channels 5 and 8.
- Selection of rising edge or falling edge as external clock input edge when external clock is selected or input capture function is used: Channels 3 to 8.
- 32-bit timer constant register for auto-reload use, readable/writable at any time, and 32-bit down-counter are provided for each channel.
- Selection of five counter input clocks: Channels 0 to 2.
Five peripheral clocks (CP ϕ /4, CP ϕ /16, CP ϕ /64, CP ϕ /256, and CP ϕ /1024).
- Selection of six counter input clocks: Channels 3 to 8.
External clock (TCLK1/2), and five peripheral clocks (S3D2 ϕ /4, S3D2 ϕ /16, S3D2 ϕ /64, S3D2 ϕ /256, and S3D2 ϕ /1024).
- Selection of five counter input clocks: Channels 9 to 11.
Five peripheral clocks (S3D2 ϕ /4, S3D2 ϕ /16, S3D2 ϕ /64, S3D2 ϕ /256, and S3D2 ϕ /1024).
- Selection of five counter input clocks: Channels 12 to 14.
Five peripheral clocks (S0D6 ϕ /4, S0D6 ϕ /16, S0D6 ϕ /64, S0D6 ϕ /256, and S0D6 ϕ /1024).
- Two interrupt sources.
One underflow source (each channel) and one input capture source (channels 5 and 8).

[RZ/G2M V1.3, RZ/G2M V3.0]

- Auto-reload type 32-bit down counter is provided for each channel.
- Input capture function provided: Channels 5 and 8.
- Selection of rising edge or falling edge as external clock input edge when external clock is selected or input capture function is used: Channels 3 to 8.
- 32-bit timer constant register for auto-reload use, readable/writable at any time, and 32-bit down-counter are provided for each channel.
- Selection of five counter input clocks: Channels 0 to 2.
Five peripheral clocks (CP ϕ /4, CP ϕ /16, CP ϕ /64, CP ϕ /256, and CP ϕ /1024).
- Selection of six counter input clocks: Channels 3 to 8.
External clock (TCLK1/2), and five peripheral clocks (S3D2 ϕ /4, S3D2 ϕ /16, S3D2 ϕ /64, S3D2 ϕ /256, and S3D2 ϕ /1024).
- Selection of five counter input clocks: Channels 9 to 11.
Five peripheral clocks (S3D2 ϕ /4, S3D2 ϕ /16, S3D2 ϕ /64, S3D2 ϕ /256, and S3D2 ϕ /1024).
- Selection of five counter input clocks: Channels 12 to 14.
Five peripheral clocks (S0D6 ϕ /4, S0D6 ϕ /16, S0D6 ϕ /64, S0D6 ϕ /256, and S0D6 ϕ /1024).
- Two interrupt sources.
One underflow source (each channel) and one input capture source (channels 5 and 8).

[RZ/G2N]

- Auto-reload type 32-bit down counter is provided for each channel.
- Input capture function provided: Channels 5 and 8.
- Selection of rising edge or falling edge as external clock input edge when external clock is selected or input capture function is used: Channels 3 to 8.
- 32-bit timer constant register for auto-reload use, readable/writable at any time, and 32-bit down-counter are provided for each channel.
- Selection of five counter input clocks: Channels 0 to 2.
Five peripheral clocks (CP ϕ /4, CP ϕ /16, CP ϕ /64, CP ϕ /256, and CP ϕ /1024).
- Selection of six counter input clocks: Channels 3 to 8.
External clock (TCLK1/2), and five peripheral clocks (S3D2 ϕ /4, S3D2 ϕ /16, S3D2 ϕ /64, S3D2 ϕ /256, and S3D2 ϕ /1024).
- Selection of five counter input clocks: Channels 9 to 11.
Five peripheral clocks (S3D2 ϕ /4, S3D2 ϕ /16, S3D2 ϕ /64, S3D2 ϕ /256, and S3D2 ϕ /1024).
- Selection of five counter input clocks: Channels 12 to 14.
Five peripheral clocks (S0D6 ϕ /4, S0D6 ϕ /16, S0D6 ϕ /64, S0D6 ϕ /256, and S0D6 ϕ /1024).
- Two interrupt sources.
One underflow source (each channel) and one input capture source (channels 5 and 8).

[RZ/G2E]

- Auto-reload type 32-bit down counter is provided for each channel.
- Input capture function provided: Channels 5 and 8.
- Selection of rising edge or falling edge as external clock input edge when external clock is selected or input capture function is used: Channels 3 to 8.
- 32-bit timer constant register for auto-reload use, readable/writable at any time, and 32-bit down-counter are provided for each channel.
- Selection of five counter input clocks: Channels 0 to 2.
Five peripheral clocks (CP ϕ /4, CP ϕ /16, CP ϕ /64, CP ϕ /256, and CP ϕ /1024).
- Selection of six counter input clocks: Channels 3 to 8.
External clock (TCLK1/2), and five peripheral clocks (S3D2C ϕ /4, S3D2C ϕ /16, S3D2C ϕ /64, S3D2C ϕ /256, and S3D2C ϕ /1024).
- Selection of five counter input clocks: Channels 9 to 11.
Five peripheral clocks (S3D2C ϕ /4, S3D2C ϕ /16, S3D2C ϕ /64, S3D2C ϕ /256, and S3D2C ϕ /1024).
- Selection of five counter input clocks: Channels 12 to 14.
Five peripheral clocks (S0D6C ϕ /4, S0D6C ϕ /16, S0D6C ϕ /64, S0D6C ϕ /256, and S0D6C ϕ /1024).
- Two interrupt sources.
One underflow source (each channel) and one input capture source (channels 5 and 8).

69.1.2 Block Diagram

Figure 69.1 show block diagrams of TMU.

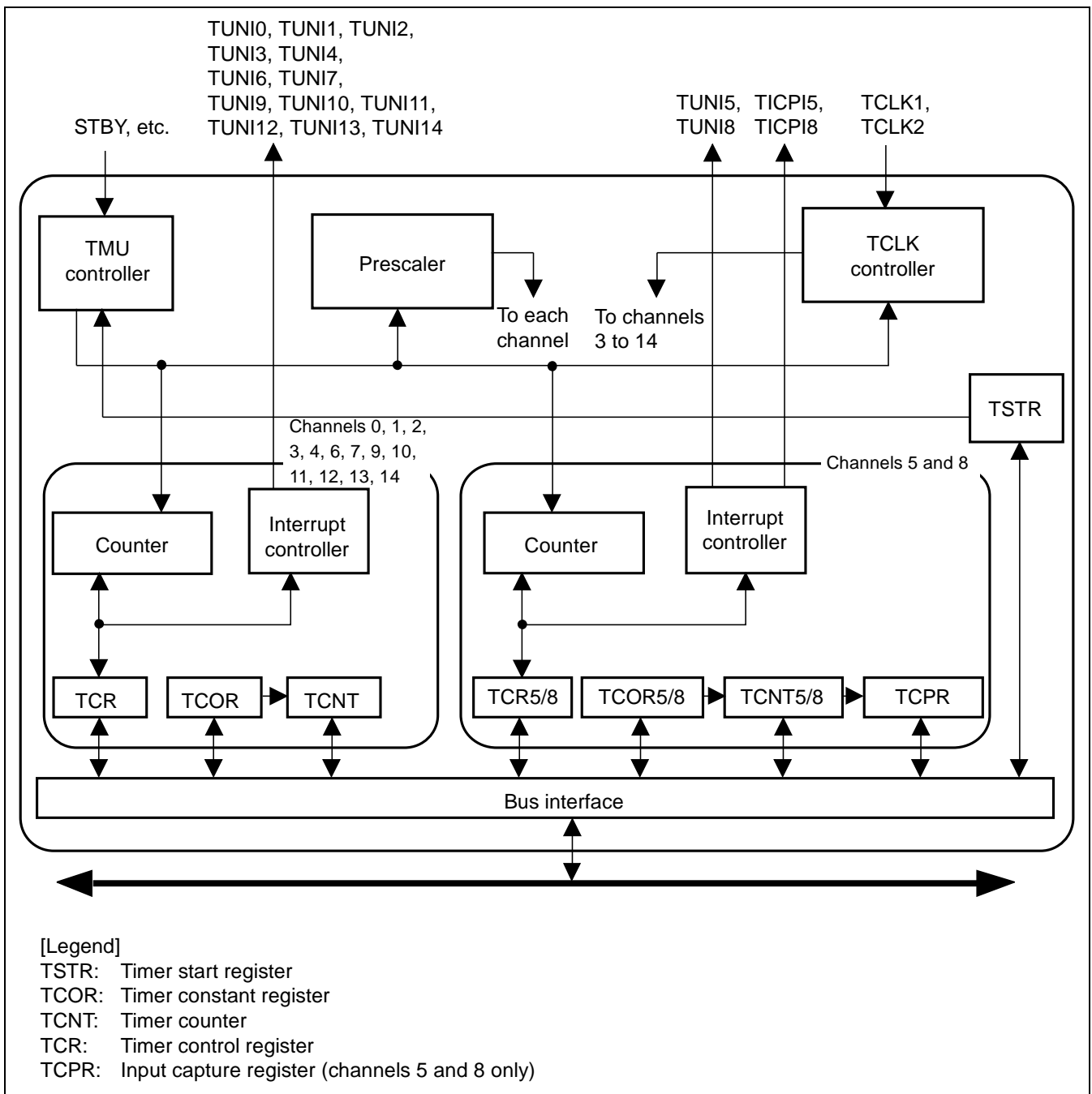


Figure 69.2 Block Diagram of TMU

69.1.3 External Pins

Table 69.1 shows the pin configuration of the TMU.

Table 69.1 Pin Configuration

| Name | Abbreviation | I/O | Function | Second Generation RZ/G Series Products | | | |
|---------------|--------------|-------|---|---|----------------------------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Clock input 1 | TCLK1 | Input | External clock input pin for channels 3 to 5. Input capture control input pin for channel 5. | √ | √ | √ | √ |
| Clock input 2 | TCLK2 | Input | External clock input pin for channels 6 to 8. Input capture control input pin for channel 8. | √ | √ | √ | √ |

69.1.4 Register Configuration

The TMU has the following registers. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined.

Table 69.2 Register Configurations (1)

| Channel | Name | Abbreviation | R/W | Address | Size | Second Generation RZ/G Series Products | | | |
|---------------------|---------------------------|--------------|-----|-------------|------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Common to 0 to 2 | Timer start register 0 | TSTR0 | R/W | H'E61E_0004 | 8 | √ | √ | √ | √ |
| 0 | Timer constant register 0 | TCOR0 | R/W | H'E61E_0008 | 32 | √ | √ | √ | √ |
| | Timer counter 0 | TCNT0 | R/W | H'E61E_000C | 32 | √ | √ | √ | √ |
| | Timer control register 0 | TCR0 | R/W | H'E61E_0010 | 16 | √ | √ | √ | √ |
| 1 | Timer constant register 1 | TCOR1 | R/W | H'E61E_0014 | 32 | √ | √ | √ | √ |
| | Timer counter 1 | TCNT1 | R/W | H'E61E_0018 | 32 | √ | √ | √ | √ |
| | Timer control register 1 | TCR1 | R/W | H'E61E_001C | 16 | √ | √ | √ | √ |
| 2 | Timer constant register 2 | TCOR2 | R/W | H'E61E_0020 | 32 | √ | √ | √ | √ |
| | Timer counter 2 | TCNT2 | R/W | H'E61E_0024 | 32 | √ | √ | √ | √ |
| | Timer control register 2 | TCR2 | R/W | H'E61E_0028 | 16 | √ | √ | √ | √ |
| Common to 3 to 5 | Timer start register 1 | TSTR1 | R/W | H'E6FC_0004 | 8 | √ | √ | √ | √ |
| 3 | Timer constant register 3 | TCOR3 | R/W | H'E6FC_0008 | 32 | √ | √ | √ | √ |
| | Timer counter 3 | TCNT3 | R/W | H'E6FC_000C | 32 | √ | √ | √ | √ |
| | Timer control register 3 | TCR3 | R/W | H'E6FC_0010 | 16 | √ | √ | √ | √ |
| 4 | Timer constant register 4 | TCOR4 | R/W | H'E6FC_0014 | 32 | √ | √ | √ | √ |
| | Timer counter 4 | TCNT4 | R/W | H'E6FC_0018 | 32 | √ | √ | √ | √ |
| | Timer control register 4 | TCR4 | R/W | H'E6FC_001C | 16 | √ | √ | √ | √ |
| 5 | Timer constant register 5 | TCOR5 | R/W | H'E6FC_0020 | 32 | √ | √ | √ | √ |
| | Timer counter 5 | TCNT5 | R/W | H'E6FC_0024 | 32 | √ | √ | √ | √ |
| | Timer control register 5 | TCR5 | R/W | H'E6FC_0028 | 16 | √ | √ | √ | √ |
| | Input capture register 5 | TCPR5 | R | H'E6FC_002C | 32 | √ | √ | √ | √ |

| | | | | | | Second Generation RZ/G Series Products | | | |
|-----------------------|----------------------------|--------------|-----|-------------|------|---|----------------------------|--------|--------|
| Channel | Name | Abbreviation | R/W | Address | Size | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Common to 6 to 8 | Timer start register 2 | TSTR2 | R/W | H'E6FD_0004 | 8 | √ | √ | √ | √ |
| 6 | Timer constant register 6 | TCOR6 | R/W | H'E6FD_0008 | 32 | √ | √ | √ | √ |
| | Timer counter 6 | TCNT6 | R/W | H'E6FD_000C | 32 | √ | √ | √ | √ |
| | Timer control register 6 | TCR6 | R/W | H'E6FD_0010 | 16 | √ | √ | √ | √ |
| 7 | Timer constant register 7 | TCOR7 | R/W | H'E6FD_0014 | 32 | √ | √ | √ | √ |
| | Timer counter 7 | TCNT7 | R/W | H'E6FD_0018 | 32 | √ | √ | √ | √ |
| | Timer control register 7 | TCR7 | R/W | H'E6FD_001C | 16 | √ | √ | √ | √ |
| 8 | Timer constant register 8 | TCOR8 | R/W | H'E6FD_0020 | 32 | √ | √ | √ | √ |
| | Timer counter 8 | TCNT8 | R/W | H'E6FD_0024 | 32 | √ | √ | √ | √ |
| | Timer control register 8 | TCR8 | R/W | H'E6FD_0028 | 16 | √ | √ | √ | √ |
| | Input capture register 8 | TCPR8 | R | H'E6FD_002C | 32 | √ | √ | √ | √ |
| Common to 9 to 11 | Timer start register 3 | TSTR3 | R/W | H'E6FE_0004 | 8 | √ | √ | √ | √ |
| 9 | Timer constant register 9 | TCOR9 | R/W | H'E6FE_0008 | 32 | √ | √ | √ | √ |
| | Timer counter 9 | TCNT9 | R/W | H'E6FE_000C | 32 | √ | √ | √ | √ |
| | Timer control register 9 | TCR9 | R/W | H'E6FE_0010 | 16 | √ | √ | √ | √ |
| 10 | Timer constant register 10 | TCOR10 | R/W | H'E6FE_0014 | 32 | √ | √ | √ | √ |
| | Timer counter 10 | TCNT10 | R/W | H'E6FE_0018 | 32 | √ | √ | √ | √ |
| | Timer control register 10 | TCR10 | R/W | H'E6FE_001C | 16 | √ | √ | √ | √ |
| 11 | Timer constant register 11 | TCOR11 | R/W | H'E6FE_0020 | 32 | √ | √ | √ | √ |
| | Timer counter 11 | TCNT11 | R/W | H'E6FE_0024 | 32 | √ | √ | √ | √ |
| | Timer control register 11 | TCR11 | R/W | H'E6FE_0028 | 16 | √ | √ | √ | √ |
| Common to 12 to 14 | Timer start register 4 | TSTR4 | R/W | H'FFC0_0004 | 8 | √ | √ | √ | √ |
| 12 | Timer constant register 12 | TCOR12 | R/W | H'FFC0_0008 | 32 | √ | √ | √ | √ |
| | Timer counter 12 | TCNT12 | R/W | H'FFC0_000C | 32 | √ | √ | √ | √ |
| | Timer control register 12 | TCR12 | R/W | H'FFC0_0010 | 16 | √ | √ | √ | √ |
| 13 | Timer constant register 13 | TCOR13 | R/W | H'FFC0_0014 | 32 | √ | √ | √ | √ |
| | Timer counter 13 | TCNT13 | R/W | H'FFC0_0018 | 32 | √ | √ | √ | √ |
| | Timer control register 13 | TCR13 | R/W | H'FFC0_001C | 16 | √ | √ | √ | √ |
| 14 | Timer constant register 14 | TCOR14 | R/W | H'FFC0_0020 | 32 | √ | √ | √ | √ |
| | Timer counter 14 | TCNT14 | R/W | H'FFC0_0024 | 32 | √ | √ | √ | √ |
| | Timer control register 14 | TCR14 | R/W | H'FFC0_0028 | 16 | √ | √ | √ | √ |

Table 69.3 Register Configurations (2)

| Channel | Name | Abbreviation | Power-on Reset/ Software Reset | Module Standby | Second Generation RZ/G Series Products | | | |
|---------------------|---------------------------|--------------|--------------------------------------|-------------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Common to 0 to 2 | Timer start register 0 | TSTR0 | H'00 | Retained | √ | √ | √ | √ |
| 0 | Timer constant register 0 | TCOR0 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 0 | TCNT0 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 0 | TCR0 | H'0000 | Retained | √ | √ | √ | √ |
| 1 | Timer constant register 1 | TCOR1 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 1 | TCNT1 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 1 | TCR1 | H'0000 | Retained | √ | √ | √ | √ |
| 2 | Timer constant register 2 | TCOR2 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 2 | TCNT2 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 2 | TCR2 | H'0000 | Retained | √ | √ | √ | √ |
| Common to 3 to 5 | Timer start register 1 | TSTR1 | H'00 | Retained | √ | √ | √ | √ |
| 3 | Timer constant register 3 | TCOR3 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 3 | TCNT3 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 3 | TCR3 | H'0000 | Retained | √ | √ | √ | √ |
| 4 | Timer constant register 4 | TCOR4 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 4 | TCNT4 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 4 | TCR4 | H'0000 | Retained | √ | √ | √ | √ |
| 5 | Timer constant register 5 | TCOR5 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 5 | TCNT5 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 5 | TCR5 | H'0000 | Retained | √ | √ | √ | √ |
| | Input capture register 5 | TCPR5 | Undefined | Retained | √ | √ | √ | √ |

| Channel | Name | Abbreviation | Power-on Reset/ Software Reset | Module Standby | Second Generation RZ/G Series Products | | | |
|----------------------|----------------------------|--------------|--------------------------------------|-------------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Common to 6 to 8 | Timer start register 2 | TSTR2 | H'00 | Retained | √ | √ | √ | √ |
| 6 | Timer constant register 6 | TCOR6 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 6 | TCNT6 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 6 | TCR6 | H'0000 | Retained | √ | √ | √ | √ |
| 7 | Timer constant register 7 | TCOR7 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 7 | TCNT7 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 7 | TCR7 | H'0000 | Retained | √ | √ | √ | √ |
| 8 | Timer constant register 8 | TCOR8 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 8 | TCNT8 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 8 | TCR8 | H'0000 | Retained | √ | √ | √ | √ |
| | Input capture register 8 | TCPR8 | Undefined | Retained | √ | √ | √ | √ |
| Common to 9 to 11 | Timer start register 3 | TSTR3 | H'00 | Retained | √ | √ | √ | √ |
| 9 | Timer constant register 9 | TCOR9 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 9 | TCNT9 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 9 | TCR9 | H'0000 | Retained | √ | √ | √ | √ |
| 10 | Timer constant register 10 | TCOR10 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 10 | TCNT10 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 10 | TCR10 | H'0000 | Retained | √ | √ | √ | √ |
| 11 | Timer constant register 11 | TCOR11 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 11 | TCNT11 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 11 | TCR11 | H'0000 | Retained | √ | √ | √ | √ |

| Channel | Name | Abbreviation | Power-on Reset/ Software Reset | Module Standby | Second Generation RZ/G Series Products | | | |
|-----------------------|----------------------------|--------------|--------------------------------------|-------------------|---|----------------------------|--------|--------|
| | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Common to 12 to 14 | Timer start register 4 | TSTR4 | H'00 | Retained | √ | √ | √ | √ |
| 12 | Timer constant register 12 | TCOR12 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 12 | TCNT12 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 12 | TCR12 | H'0000 | Retained | √ | √ | √ | √ |
| 13 | Timer constant register 13 | TCOR13 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 13 | TCNT13 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 13 | TCR13 | H'0000 | Retained | √ | √ | √ | √ |
| 14 | Timer constant register 14 | TCOR14 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer counter 14 | TCNT14 | H'FFFF_FFFF | Retained | √ | √ | √ | √ |
| | Timer control register 14 | TCR14 | H'0000 | Retained | √ | √ | √ | √ |

69.2 Register Description

Registers in the TMU are allocated to and arranged in the address space of the internal bus.

Legend for Register Description

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R: Read-only. When the bit is reserved, the write value should always be 0. Writing 1 to these bits can cause a malfunction of TMU.

—/W: Write-only. The read value is undefined.

69.2.1 Timer Start Registers (TSTRn) (n = 0 to 4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

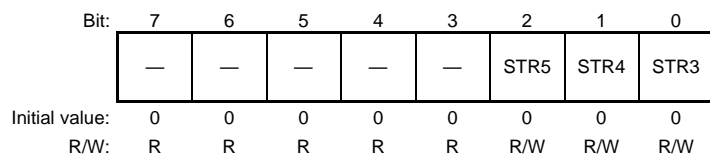
TSTR are 8-bit readable/writable registers that select whether to run or halt the TCNT.

- (TSTR0)

| | | | | | | | | |
|----------------|---|---|---|---|---|------|------|------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | STR2 | STR1 | STR0 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W |

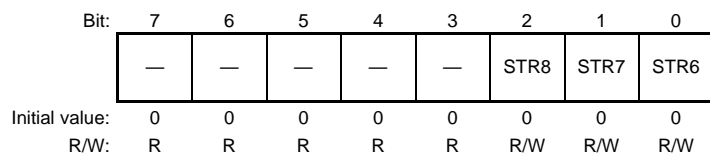
| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | STR2 | B'0 | R/W | Counter Start 2 Selects whether to run or halt TCNT2. 0: TCNT2 count halted 1: TCNT2 counts |
| 1 | STR1 | B'0 | R/W | Counter Start 1 Selects whether to run or halt TCNT1. 0: TCNT1 count halted 1: TCNT1 counts |
| 0 | STR0 | B'0 | R/W | Counter Start 0 Selects whether to run or halt TCNT0. 0: TCNT0 count halted 1: TCNT0 counts |

• (TSTR1)



| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | STR5 | B'0 | R/W | Counter Start 5 Selects whether to run or halt TCNT5. 0: TCNT5 count halted 1: TCNT5 counts |
| 1 | STR4 | B'0 | R/W | Counter Start 4 Selects whether to run or halt TCNT4. 0: TCNT4 count halted 1: TCNT4 counts |
| 0 | STR3 | B'0 | R/W | Counter Start 3 Selects whether to run or halt TCNT3. 0: TCNT3 count halted 1: TCNT3 counts |

• (TSTR2)



| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | STR8 | B'0 | R/W | Counter Start 8 Selects whether to run or halt TCNT8. 0: TCNT8 count halted 1: TCNT8 counts |
| 1 | STR7 | B'0 | R/W | Counter Start 7 Selects whether to run or halt TCNT7. 0: TCNT7 count halted 1: TCNT7 counts |
| 0 | STR6 | B'0 | R/W | Counter Start 6 Selects whether to run or halt TCNT6. 0: TCNT6 count halted 1: TCNT6 counts |

• (TSTR3)

| | | | | | | | | |
|----------------|---|---|---|---|---|-------|-------|------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | STR11 | STR10 | STR9 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | STR11 | B'0 | R/W | Counter Start 11 Selects whether to run or halt TCNT11. 0: TCNT11 count halted 1: TCNT11 counts |
| 1 | STR10 | B'0 | R/W | Counter Start 10 Selects whether to run or halt TCNT10. 0: TCNT10 count halted 1: TCNT10 counts |
| 0 | STR9 | B'0 | R/W | Counter Start 9 Selects whether to run or halt TCNT9. 0: TCNT9 count halted 1: TCNT9 counts |

• (TSTR4)

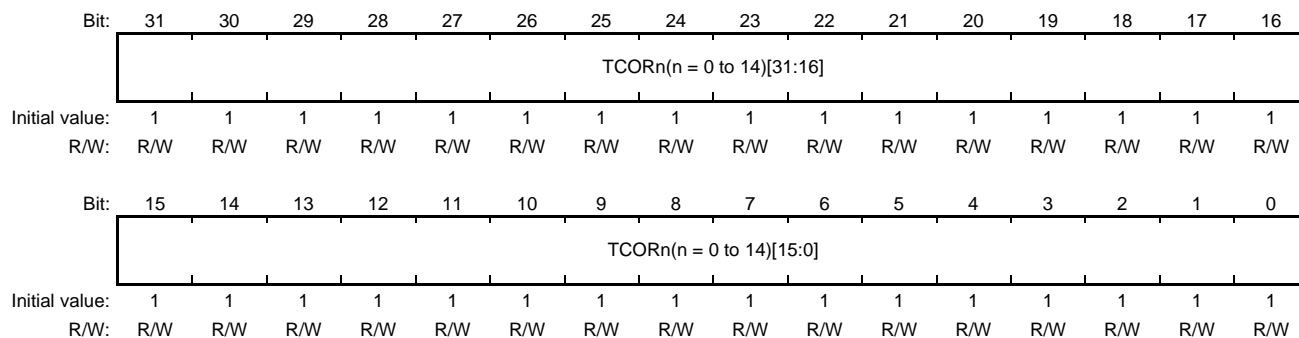
| | | | | | | | | |
|----------------|---|---|---|---|---|-------|-------|-------|
| Bit: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | STR14 | STR13 | STR12 |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 3 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 2 | STR14 | B'0 | R/W | Counter Start 14 Selects whether to run or halt TCNT14. 0: TCNT14 count halted 1: TCNT14 counts |
| 1 | STR13 | B'0 | R/W | Counter Start 13 Selects whether to run or halt TCNT13. 0: TCNT13 count halted 1: TCNT13 counts |
| 0 | STR12 | B'0 | R/W | Counter Start 12 Selects whether to run or halt TCNT12. 0: TCNT12 count halted 1: TCNT12 counts |

69.2.2 Timer Constant Registers (TCORn) (n = 0 to 14)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TCOR are 32-bit readable/writable registers. After underflow has been generated according to the result of the TCNT countdown, the value of TCOR is set to TCNT and TCNT continues countdown from the value.

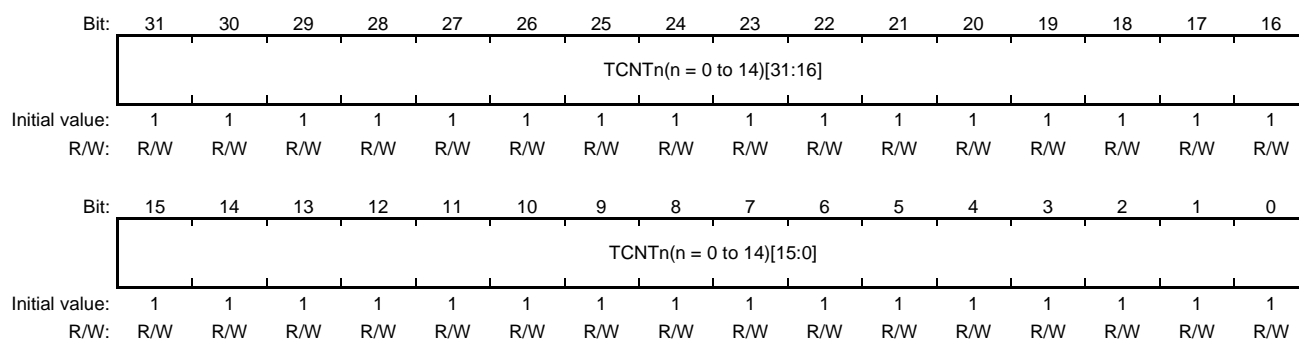


69.2.3 Timer Counters (TCNTn) (n = 0 to 14)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TCNT are 32-bit readable/writable registers that count down upon the input clock selected using the bits TPSC2 to TPSC0 in TCR.

When a TCNT countdown results in an underflow, the UNF in TCR of corresponding channel is set. At the same time, the value of TCOR is set to TCNT and TCNT continues countdown from that value.



69.2.4 Timer Control Registers (TCRn) (n = 0 to 14)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TCR are 16-bit readable/writable registers that select a count clock and edge when an external clock is selected, and control an interrupt generation when the flag that indicates the generation of a TCNT is set to 1.

TCR of channels 5 and 8 control the input capture function and generation of an interrupt during the input capture.

- Bit assignment for non input capture function

TCR0, TCR1, TCR2, TCR3, TCR4, TCR6, TCR7, TCR9, TCR10, TCR11, TCR12, TCR13, TCR14

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|-----|---|---|------|------|-----|------|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | UNF | — | — | UNIE | CKEG | | TPSC | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

- Bit assignment for input capture function

TCR5, TCR8

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|------|-----|------|-----|------|------|-----|------|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | ICPF | UNF | ICPE | | UNIE | CKEG | | TPSC | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|---|
| 15 to 10 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 9 | ICPF*1 | B'0 | R/W | Input Capture Interrupt Flag Status flag, provided in channels 5 and 8 which indicates the occurrence of input capture. 0: No input capture has occurred [Clearing condition] When 0 is written to ICPF 1: Input capture has occurred [Setting condition] When an input capture occurs*2 |
| 8 | UNF | B'0 | R/W | Underflow Flag Status flag which indicates the occurrence of a TCNT underflow. 0: TCNT has not underflowed [Clearing condition] When 0 is written to UNF 1: TCNT has underflowed [Setting condition] When TCNT underflows*2 |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|--------------------|---------------|-----|--|
| 7, 6 | ICPE* ¹ | B'00 | R/W | <p>Input Capture Control</p> <p>A function of channels 5, 8: determines whether the input capture function can be used, and when used, whether or not to enable interrupts.</p> <p>Use the CKEG bits to designate use of either the rising or falling edge of the TCLK pin to set the values of TCNT5, TCNT8 to TCPR5, TCPR8 respectively.</p> <p>Only when the ICPF bits in TCR5, TCR8 are 0, the values of TCNT5, TCNT8 are set to TCPR5, TCPR8. When the ICPF bit is set to 1, neither TCPR5, TCPR8 is set even when input capture is generated.</p> <p>B'00: Input capture function is not used. B'01: Reserved (setting prohibited). B'10: Input capture function is used. Interrupt due to input capture (TICPI5, TICPI8) is not enabled. B'11: Input capture function is used. Interrupt due to input capture (TICPI5, TICPI8) is enabled.</p> |
| 5 | UNIE | B'0 | R/W | <p>Underflow Interrupt Control</p> <p>Controls enabling of interrupt generation when the status flag (UNF) indicating TCNT underflow has been set to 1.</p> <p>0: Interrupt due to underflow (TUNI) is not enabled 1: Interrupt due to underflow (TUNI) is enabled</p> |
| 4, 3 | CKEG | B'00 | R/W | <p>Clock Edge</p> <p>Select an input edge of the external clock when the external clock is selected, or when the input capture function is used.</p> <p>B'00: Count/capture register set on rising edge B'01: Count/capture register set on falling edge Other: Count/capture register set on both rising and falling edge</p> |
| 2 to 0 | TPSC | B'000 | R/W | <p>Timer Pre-scaler 2 to 0</p> <p>Select the TCNT count clock.</p> <p>B'000: Count on (input-clock)/4 B'001: Count on (input-clock)/16 B'010: Count on (input-clock)/64 B'011: Count on (input-clock)/256 B'100: Count on (input-clock)/1024 B'101: Setting prohibited B'110: Setting prohibited B'111: Count on external clock</p> <p>Channels 3 to 5 TCLK1, channels 6 to 8 TCLK2. Not usable in channels 0, 1, 2, 9, 10, 11, 12, 13, 14.</p> <p>Note: Input-clock is CPϕ for channel 0/1/2, S3D2ϕ for channel 3/4/5/6/7/8/9/10/11, S0D6ϕ for channel 12/13/14 in the RZ/G2H.</p> <p>Input-clock is CPϕ for channel 0/1/2, S3D2ϕ for channel 3/4/5/6/7/8/9/10/11, S0D6ϕ for channel 12/13/14 in the RZ/G2M V1.3, RZ/G2M V3.0.</p> <p>Input-clock is CPϕ for channel 0/1/2, S3D2ϕ for channel 3/4/5/6/7/8/9/10/11, S0D6ϕ for channel 12/13/14 in the RZ/G2N.</p> <p>Input-clock is CPϕ for channel 0/1/2, S3D2Cϕ for channel 3/4/5/6/7/8/9/10/11, S0D6Cϕ for channel 12/13/14 in the RZ/G2E.</p> |

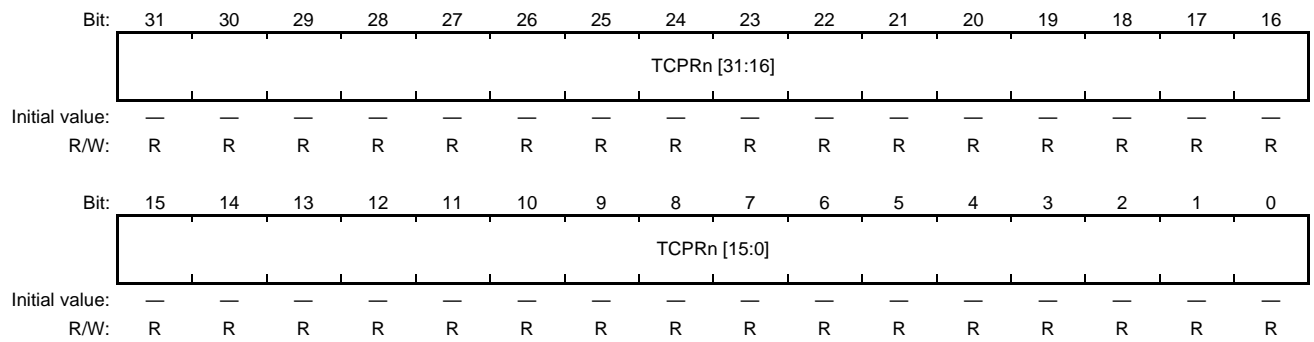
- Notes:
1. Reserved in channels 0, 1, 2, 3, 4, 6, 7, 9, 10, 11, 12, 13 and 14 (initial value is 0 and read-only).
 2. Writing 1 does not change the value.

69.2.5 Input Capture Registers n (TCPRn)

Note: n = 5 and 8

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

TCPR5, TCPR8 are read-only 32-bit registers used for the input capture function provided only in channels 5, 8. The ICPE and CKEG bits in TCR5, TCR8 control the input capture function. When an input capture occurs, the value of TCNT5 is copied to TCPR5, the value of TCNT8 is copied to TCPR8. The values of TCNT5, TCNT8 are set in TCPR5, TCPR8, respectively, only when the ICPF bits in TCR5, TCR8 are 0.



69.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Each channel has a 32-bit timer counter (TCNT) and 32-bit timer constant register (TCOR). TCNT counts down. The auto-reload function enables synchronized counting and external-event counting. Channels 5, 8 have the input capture function.

69.3.1 Counter Operation

When the bits STR0 to STR14 in TSTR0 to TSTR4 are set to 1, TCNT of corresponding channel starts counting. When TCNT underflows, the UNF flag of corresponding TCR is set. In this case, if the UNIE bit in TCR is set to 1, an interrupt request is sent to the CPU. Also, the value is copied from TCOR to TCNT and the down-count operation is continued (Auto reload function).

(1) Procedure for setting count operation

Figure 69.3 shows an example of the procedure for setting the count operation.

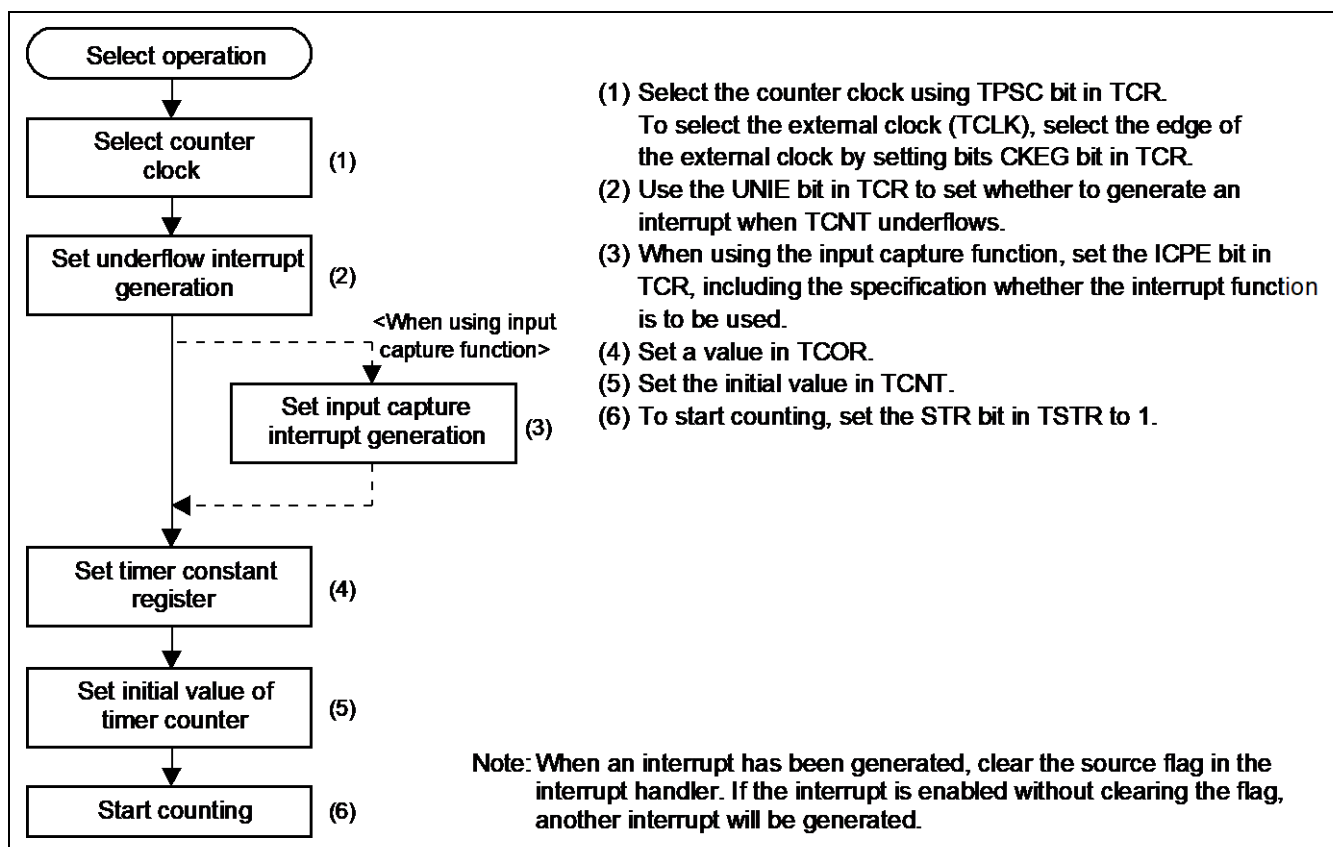


Figure 69.3 Procedure for Setting Count Operation

(2) Auto-reload count operation

Figure 69.4 shows the TCNT auto-reload operation.

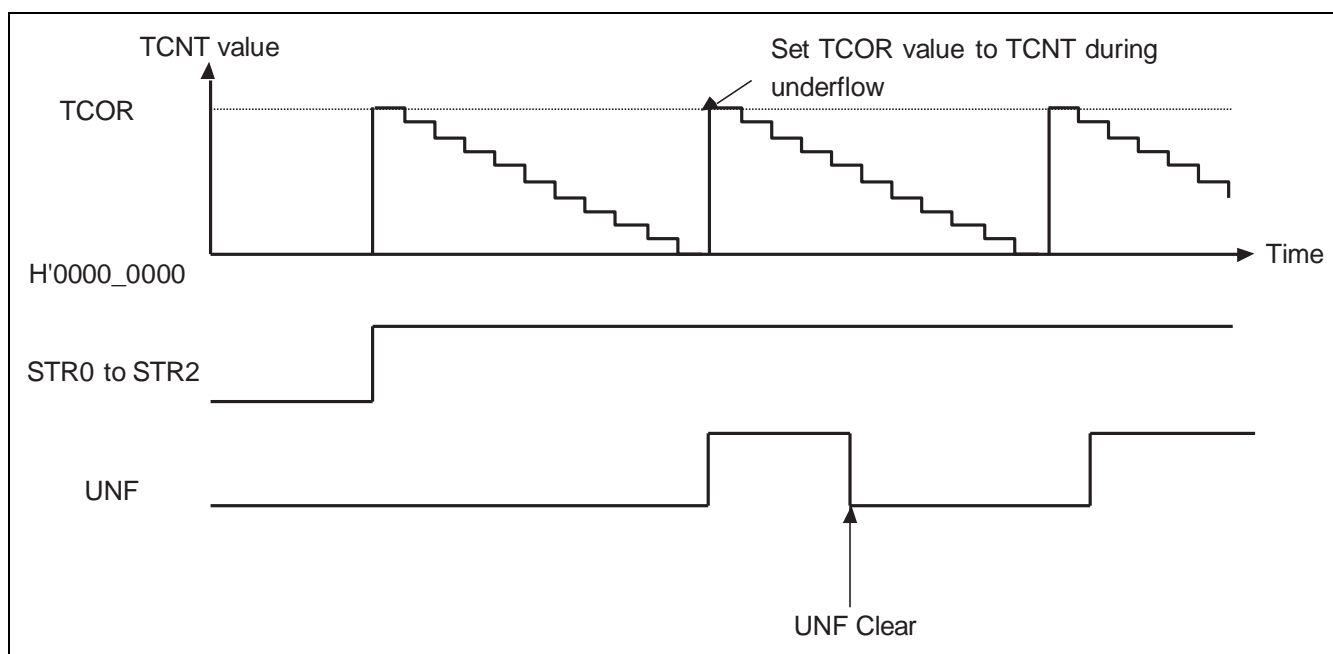


Figure 69.4 TCNT Auto-Reload Operation

(3) TCNT count timing

- Internal clock operation

[RZ/G2H]:

Five clocks (S3D2 ϕ /4, S3D2 ϕ /16, S3D2 ϕ /64, S3D2 ϕ /256, S3D2 ϕ /1024) and five clocks (CP ϕ /4, CP ϕ /16, CP ϕ /64, CP ϕ /256, CP ϕ /1024) and five clocks (S0D6 ϕ /4, S0D6 ϕ /16, S0D6 ϕ /64, S0D6 ϕ /256, S0D6 ϕ /1024) that are created by dividing peripheral clocks are selected as count clocks by setting bits TPSC2 to TPSC0 in TCR.

[RZ/G2M V1.3, RZ/G2M V3.0]:

Five clocks (S3D2 ϕ /4, S3D2 ϕ /16, S3D2 ϕ /64, S3D2 ϕ /256, S3D2 ϕ /1024) and five clocks (CP ϕ /4, CP ϕ /16, CP ϕ /64, CP ϕ /256, CP ϕ /1024) and five clocks (S0D6 ϕ /4, S0D6 ϕ /16, S0D6 ϕ /64, S0D6 ϕ /256, S0D6 ϕ /1024) that are created by dividing peripheral clocks are selected as count clocks by setting bits TPSC2 to TPSC0 in TCR.

[RZ/G2N]:

Five clocks (S3D2 ϕ /4, S3D2 ϕ /16, S3D2 ϕ /64, S3D2 ϕ /256, S3D2 ϕ /1024) and five clocks (CP ϕ /4, CP ϕ /16, CP ϕ /64, CP ϕ /256, CP ϕ /1024) and five clocks (S0D6 ϕ /4, S0D6 ϕ /16, S0D6 ϕ /64, S0D6 ϕ /256, S0D6 ϕ /1024) that are created by dividing peripheral clocks are selected as count clocks by setting bits TPSC2 to TPSC0 in TCR.

[RZ/G2E]:

Five clocks (S3D2C ϕ /4, S3D2C ϕ /16, S3D2C ϕ /64, S3D2C ϕ /256, S3D2C ϕ /1024) and five clocks (CP ϕ /4, CP ϕ /16, CP ϕ /64, CP ϕ /256, CP ϕ /1024) and five clocks (S0D6C ϕ /4, S0D6C ϕ /16, S0D6C ϕ /64, S0D6C ϕ /256, S0D6C ϕ /1024) that are created by dividing peripheral clocks are selected as count clocks by setting bits TPSC2 to TPSC0 in TCR.

Figure 69.6, 69.7, 69.8 and 69.9 show the timing.

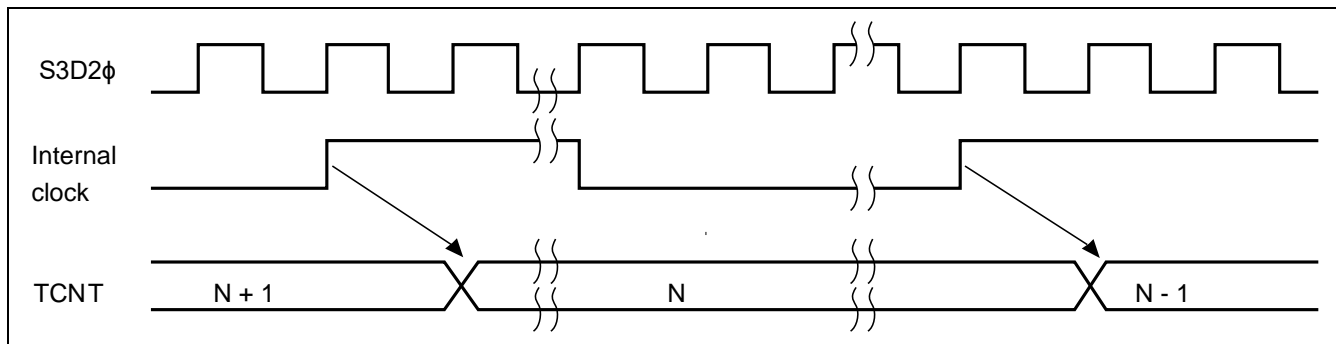


Figure 69.6 Count Timing when Internal Clock is Operating for , RZ/G2H

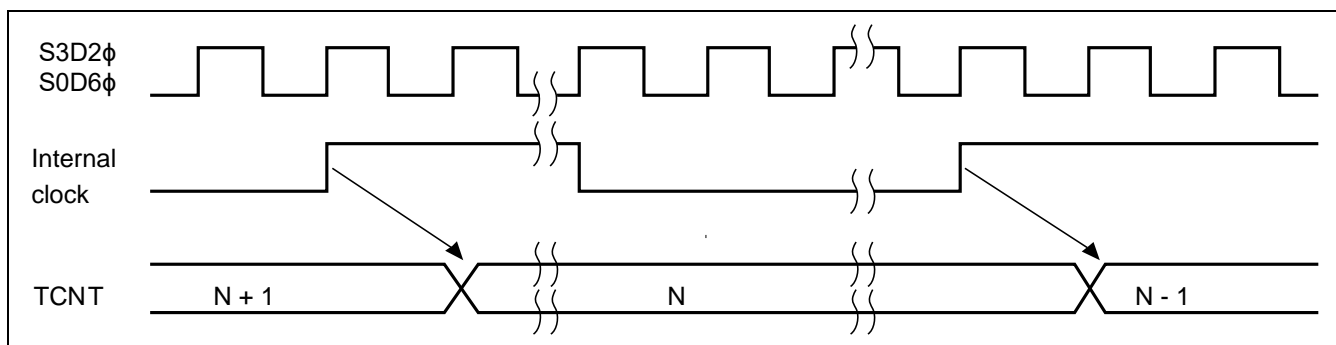


Figure 69.7 Count Timing when Internal Clock is Operating for RZ/G2M V1.3, RZ/G2M V3.0

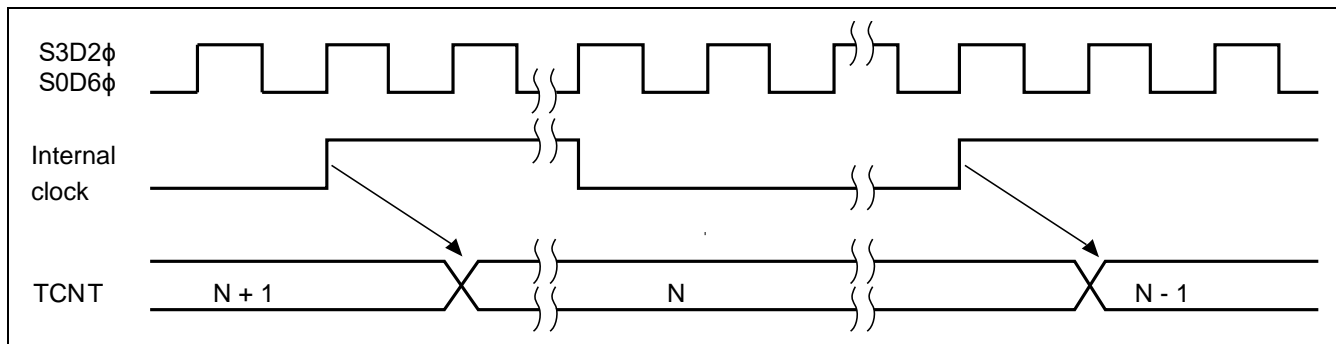


Figure 69.8 Count Timing when Internal Clock is Operating for RZ/G2N

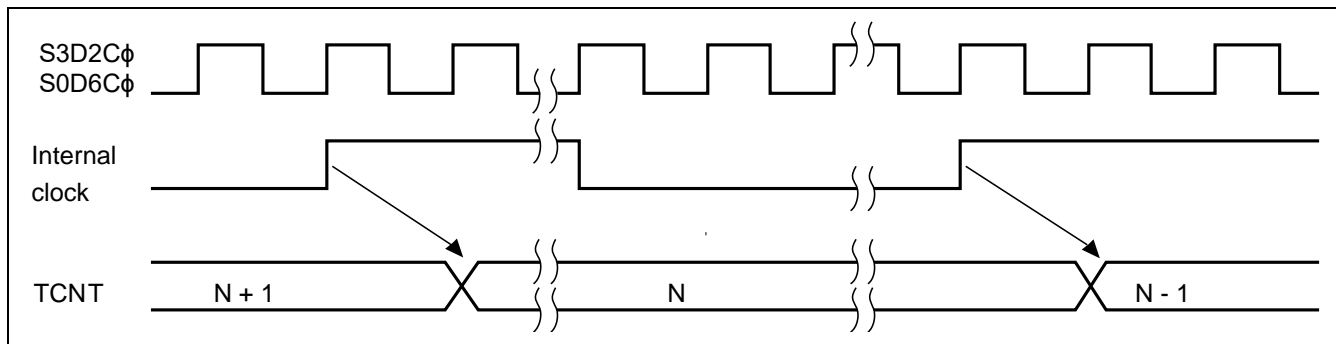


Figure 69.9 Count Timing when Internal Clock is Operating for RZ/G2E

- External clock operation

Set the bits TPSC2 to TPSC0 in TCR to select the external clock pin (TCLK) as the timer clock. Use the bits CKEG1 and CKEG0 in TCR to select the detection edge. Rise, fall, or both can be selected.

Figure 69.10, 69.11, 69.12 and 69.13 show the timing for both-edge detection.

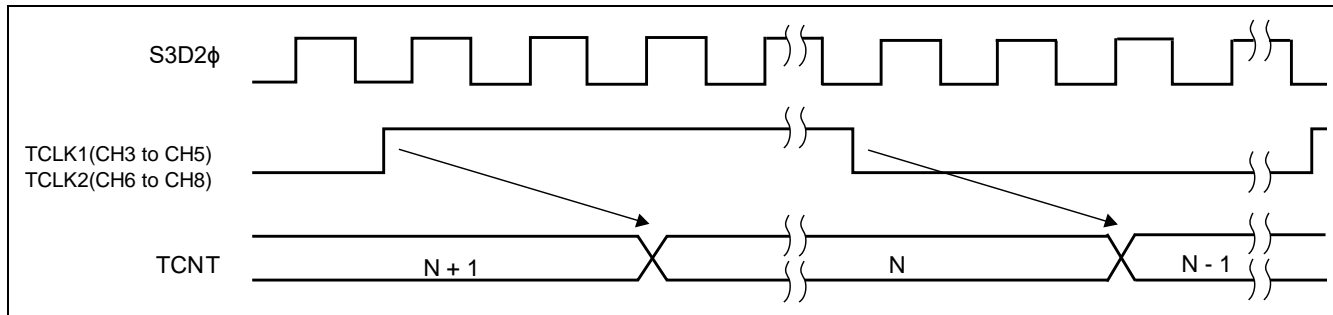


Figure 69.10 Count Timing when External Clock is Operating for , RZ/G2H

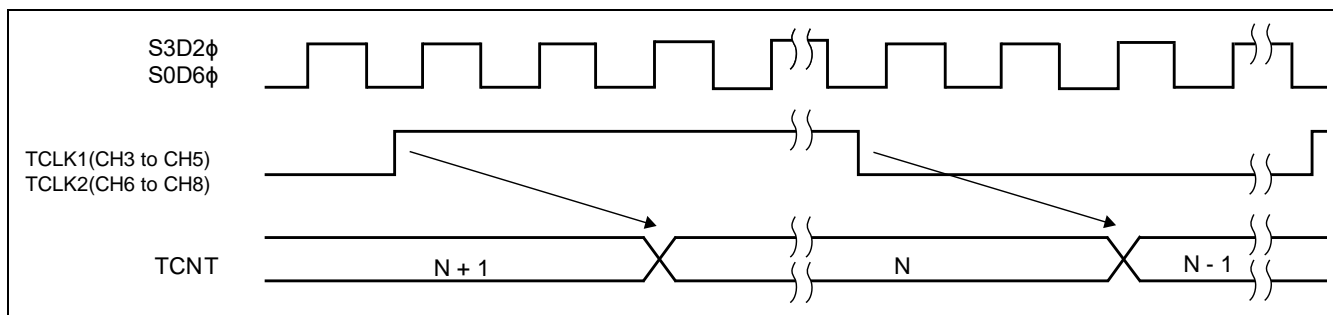


Figure 69.11 Count Timing when External Clock is Operating for RZ/G2M V1.3, RZ/G2M V3.0

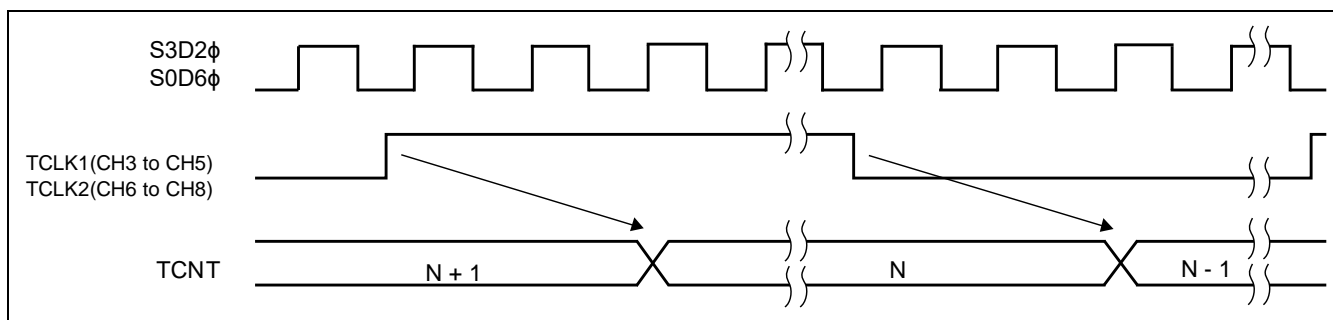


Figure 69.12 Count Timing when External Clock is Operating for RZ/G2N

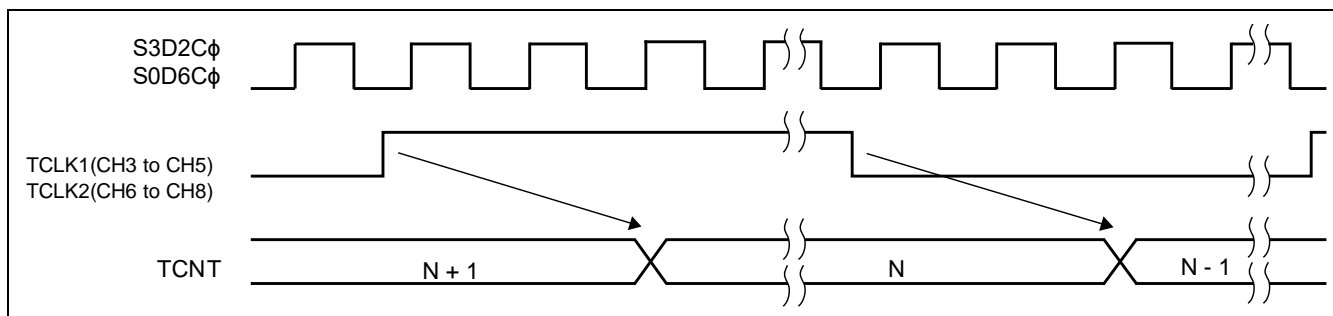


Figure 69.13 Count Timing when External Clock is Operating for RZ/G2E

69.3.2 Input Capture Function

Channels 5 and 8 have the input capture function.

1. Set the timer operating clock as an internal clock with bits TPSC2 to TPSC0 in TCR.
2. Set use of the input capture function and whether to generate an interrupt on using it with bits ICPE1 and ICPE0 in TCR.
3. Specify either rising edge or falling edge of the TCLK1 (channel 5), TCLK2 (channel 8) pin to be used to set the value of TCNT to TCPR5, TCPR8 with bits CKEG1 and CKEG0 in TCR.

Only when an input capture is occurred and the ICPF bits in TCR5, TCR8 are 0, the values of TCNT5, TCNT8 are set in TCPR5, TCPR8 respectively.

Figure 69.14 shows the operating timing when the input capture function is used (the rising edge of TCLK1/2 is used).

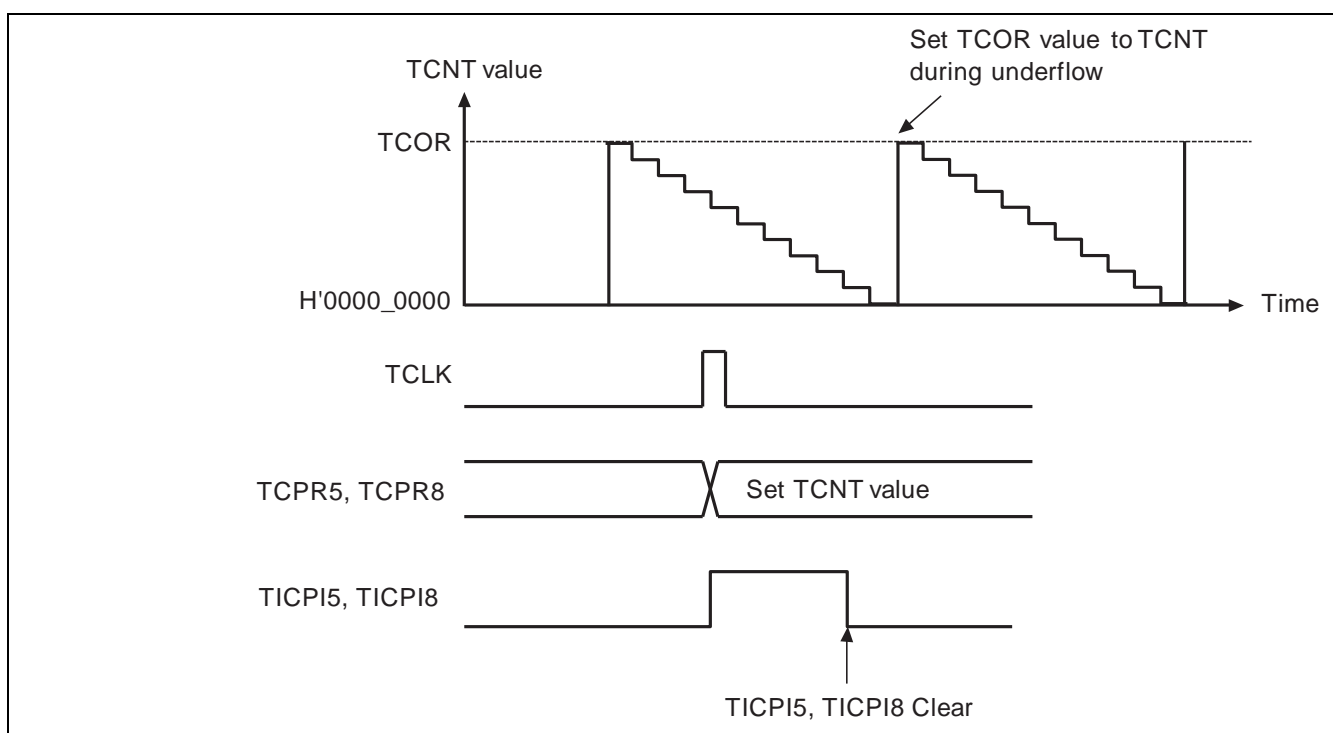


Figure 69.14 Operating Timing when Using Input Capture Function

69.3.3 Interrupt

The TMU interrupt sources are underflow interrupt or input capture interrupt when the input capture function is used. The underflow interrupt is generated at each channel. The input capture interrupt is generated at channels 5, 8 only.

An underflow interrupt request is generated (for each channel) when both the UNF bit and the interrupt enable bit for that channel are set to 1.

When the input capture function is used and the input capture request is generated, an interrupt request is generated if the ICPF bit in TCR5 or TCR8 is 1 and the input capture control bits ICPE1 and ICPE0 in TCR5 or TCR8 are 11.

Table 69.4 shows the TMU interrupt sources.

Table 69.4 TMU Interrupt Sources

| | | | Second Generation RZ/G Series Products | | | |
|---------|-------------------|---------------------------|---|----------------------------|--------|--------|
| Channel | Interrupt Sources | Description | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| 0 | TUNI0 | Underflow interrupt 0 | √ | √ | √ | √ |
| 1 | TUNI1 | Underflow interrupt 1 | √ | √ | √ | √ |
| 2 | TUNI2 | Underflow interrupt 2 | √ | √ | √ | √ |
| 3 | TUNI3 | Underflow interrupt 3 | √ | √ | √ | √ |
| 4 | TUNI4 | Underflow interrupt 4 | √ | √ | √ | √ |
| 5 | TUNI5 | Underflow interrupt 5 | √ | √ | √ | √ |
| | TICPI5 | Input capture interrupt 5 | √ | √ | √ | √ |
| 6 | TUNI6 | Underflow interrupt 6 | √ | √ | √ | √ |
| 7 | TUNI7 | Underflow interrupt 7 | √ | √ | √ | √ |
| 8 | TUNI8 | Underflow interrupt 8 | √ | √ | √ | √ |
| | TICPI8 | Input capture interrupt 8 | √ | √ | √ | √ |
| 9 | TUNI9 | Underflow interrupt 9 | √ | √ | √ | √ |
| 10 | TUNI10 | Underflow interrupt 10 | √ | √ | √ | √ |
| 11 | TUNI11 | Underflow interrupt 11 | √ | √ | √ | √ |
| 12 | TUNI12 | Underflow interrupt 12 | √ | √ | √ | √ |
| 13 | TUNI13 | Underflow interrupt 13 | √ | √ | √ | √ |
| 14 | TUNI14 | Underflow interrupt 14 | √ | √ | √ | √ |

69.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

69.4.1 Writing to Registers

When writing to the TMU registers, clear the start bits (STR14 to STR0) of the corresponding TSTR channel to stop the timer counting.

Writing to TSTR and clearing bits UNF and ICPF in TCR can be executed during counting. To clear flags UNF and ICPF during counting, do not change the values of bits other than those to be cleared.

69.4.2 Reading TCNT Register

Reading from TCNT is performed synchronously with the timer count operation. When timer counting and register read processing are performed simultaneously, the value before TCNT counting down is read.

69.4.3 External Clock Frequency

The frequency of external clock (TCLK1/2) for each channel should be $S3D2\phi/4$ or less in the RZ/G2H.

The frequency of external clock (TCLK1/2) for each channel should be $S3D2\phi/4$ or less in the RZ/G2M V1.3, RZ/G2M V3.0.

The frequency of external clock (TCLK1/2) for each channel should be $S3D2\phi/4$ or less in the RZ/G2N.

The frequency of external clock (TCLK1/2) for each channel should be $S3D2C\phi/4$ or less in the RZ/G2E.

70. System Timer (SCMT)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

70.1 Overview

System Timer is a 32-bit compare match timer of one channel.

70.1.1 Features

- One channel is implemented.
- 16 bits/32 bits can be selected.
- Provided with 32-bit constant registers and 32-bit up-counters that can be written or read at any time.
- Allows selection among four counter input clocks
 - OSCCLK: 1/1, 1/8, 1/32, and 1/128
- One-shot operation or free-running operation is selectable.
- Compare match or overflow can be selected as interrupt source.
- Counter operation can be enabled or disabled at the time of debugging of CPU core using the debugging mode operation selector.

70.1.2 Block Diagram

Figure 70.1 shows a block diagram of the System Timer.

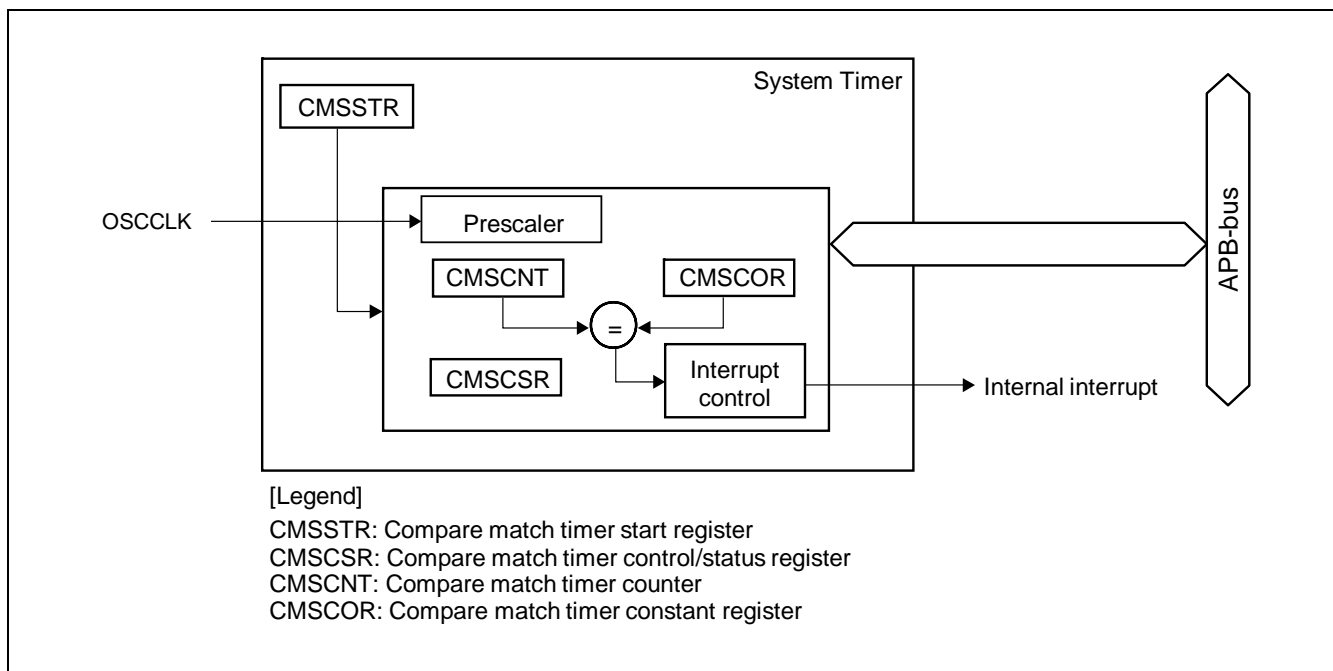


Figure 70.1 Block Diagram of System Timer

70.1.3 Register Configuration

Table 70.1 shows the System Timer register configuration. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined. Access CMSSTR and CMSCSR as a word (16 bits). Operation cannot be guaranteed if CMSSTR and CMSCSR are not accessed as a word. Access CMSCNT and CMSCOR as a longword (32 bits). Operation cannot be guaranteed if CMSCNT and CMSCOR are not accessed as a longword.

Table 70.1 Register Configurations

| Register Name | Abbreviation | R/W | Address | Value after Power-On Reset | Size | Second Generation RZ/G Series Products | | | | |
|---|--------------|------|-------------|----------------------------|------|--|----------------------------|--------|--------|--|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Compare match timer start register | CMSSTR | R/W | H'E604_0000 | H'0000 | 16 | √ | √ | √ | √ | |
| Compare match timer control/status register | CMSCSR | R/W* | H'E604_0040 | H'012F | 16 | √ | √ | √ | √ | |
| Compare match timer counter | CMSCNT | R/W | H'E604_0044 | H'0000_0000 | 32 | √ | √ | √ | √ | |
| Compare match timer constant register | CMSCOR | R/W | H'E604_0048 | H'0144_3FCF | 32 | √ | √ | √ | √ | |

Note: * Except for bit [15:14] that are R/WC0 register.

Table 70.2 shows the register states in each operating mode.

Table 70.2 Register States in Each Operating Mode

| Register Abbreviation | Power-on Reset/Software Reset * |
|------------------------------|--|
| CMSSTR | Initialized |
| CMSCSR | Initialized |
| CMSCNT | Initialized |
| CMSCOR | Initialized |

Note: * For details, refer to 'General Part Section 17 Reset (RST) and '12. Module Standby, Software Reset.

70.2 Register Description

[Legend for Register Description]

Initial value: Register value after a reset.

—: Undefined value

R/W: Bit or field is readable and writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R: Read-only. The write value should always be initial value.

70.2.1 Compare Match Timer Start Register (CMSSTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMSSTR is a 16-bit register which specify whether the compare match timer counter (CMSCNT) is operated or halted.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|------|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | STR5 | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 15 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 | STR5 | B'0 | R/W | Count Start Specify halt/counting of compare match timer counter (CMSCNT). 0: CMSCNT Halts. 1: CMSCNT starts counting. |
| 4 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

70.2.2 Compare Match Timer Control/Status Register (CMSCSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMSCSR is a 16-bit register that indicates the occurrence of compare matches, enables interrupts, and sets the counter input clocks.

Do not change bits other than the CMF and OVF bits, while the compare match timer counter (CMSCNT) is under operating.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|----|----|----|-----|-----|---|---|----------|---------|----------|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CMF | OVF | WRFLG | — | — | — | CMS | CMM | — | — | CMR[1:0] | DBGIV D | CKS[2:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 |
| R/W: | R/WC0 | R/WC0 | R | R | R | R | R/W | R/W | R | R | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|---|
| 15 | CMF | B'0 | R/WC0 | <p>Compare Match Flag</p> <p>This flag indicates whether the values of the compare match timer counter (CMSCNT) and the compare match timer constant register (CMSCOR) have matched or not.</p> <p>Software cannot write 1 to the bit. When one-shot is selected for the counter operation, counting resumes by clearing this bit.</p> <p>0: CMSCNT and CMSCOR values have not matched. [Clearing condition]</p> <ul style="list-style-type: none"> Write 0 to CMF <p>1: CMSCNT and CMSCOR values have matched.</p> <p>* Only 0 can be written, for flag clearing.</p> |
| 14 | OVF | B'0 | R/WC0 | <p>Overflow Flag</p> <p>This flag indicates whether the compare match timer counter (CMSCNT) has overflowed or not. Software cannot write 1 to this bit.</p> <p>0: CMSCNT has not overflowed. [Clearing condition]</p> <ul style="list-style-type: none"> Write 0 to OVF <p>1: CMSCNT has overflowed.</p> <p>* Only 0 can be written, for flag clearing.</p> |
| 13 | WRFLG | B'0 | R | <p>Write State Flag</p> <p>When this bit is 1, write access to CMSCNT is prohibited. If it isn't maintained, operation isn't secured. This bit indicates the period that the writing to CMSCNT is masked for synchronization after the writing to CMSCNT. Confirm that this flag is 0 when CMSCNT is written to continuously.</p> |
| 12 to 10 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 9 | CMS | B'0 | R/W | <p>Compare Match Timer Counter Size</p> <p>This bit selects whether the compare match timer counter (CMSCNT) is used as a 16-bit counter or 32-bit counter. This register setting becomes the valid bit-size for the compare match timer constant register (CMSCOR).</p> <p>0: Operates as a 32-bit counter. 1: Operates as a 16-bit counter.</p> <p>Note: When Write or Read in this bit, refer to 70.3.5 Register Access.</p> |
| 8 | CMM | B'1 | R/W | <p>Compare Match Mode</p> <p>Specify operation mode of the counter.</p> <p>0: One-shot operation 1: Free-running operation</p> <p>Note: When Write or Read in this bit, refer to 70.3.5 Register Access.</p> |
| 7,6 | — | B'00 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |
| 5, 4 | CMR[1:0] | B'10 | R/W | <p>Compare Match Request</p> <p>These bits enable or disable an internal interrupt request at a compare match.</p> <p>B'00: Disables an internal interrupt request. B'01: Setting prohibited B'10: Enables an internal interrupt request. B'11: Setting prohibited</p> |
| 3 | DBGIVD | B'1 | R/W | <p>Debug Mode Operation Select</p> <p>This bit sets counter operation when in debug mode.</p> <p>0: Stops counter operation when in debug mode. 1: Enables counter operation even when in debug mode.</p> <p>Note: When Write or Read in this bit, refer to 70.3.5 Register Access.</p> |
| 2 to 0 | CKS[2:0] | B'111 | R/W | <p>Clock Select</p> <p>These bits select the clock input to CMSCNT. When the count start bit (STR5) in CMSSTR is set to 1, CMSCNT begins incrementing with the clock selected by these bits.</p> <p>B'000: Setting prohibited B'001: Setting prohibited B'010: Setting prohibited B'011: Setting prohibited B'100: clock/8 B'101: clock/32 B'110: clock/128 B'111: clock/1</p> <p>Note: When Write or Read in this bit, refer to 70.3.5 Register Access.</p> |

70.2.3 Compare Match Timer Counter (CMSCNT)

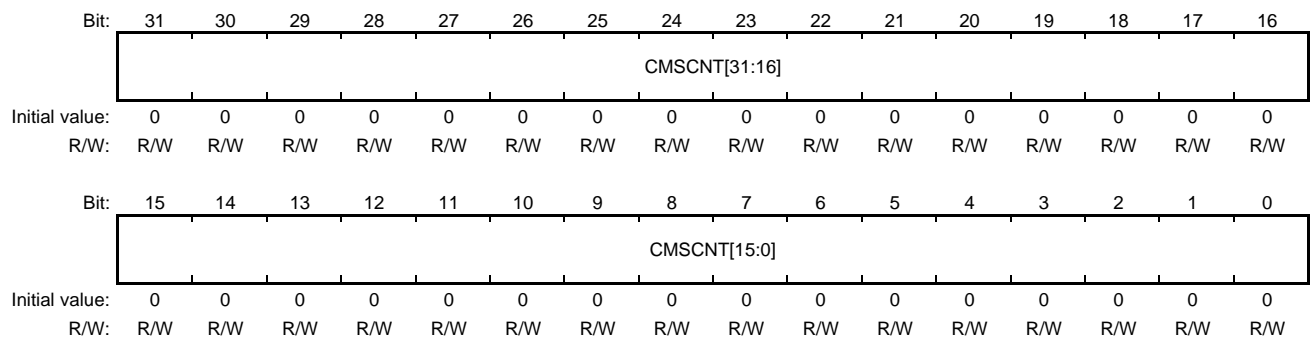
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMSCNT is a 32-bit register which is used as an up-counter.

To specify counter operation, set compare match timer control/status register (CMSCSR), before starting operation.

When the 16-bit counter operation is selected by the CMS bit, bits 31 to 16 of this register is invalid. When data is written in 16-bit mode, write H'0000 as upper 16-bits data.

When CMSCNT is read during the counter operation, the read value may be wrong because of an asynchronous clock between counter and bus-interface. For exact value, read this register continuously, until same values are read from this register.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|---|
| 31 to 0 | CMSCNT [31:0] | H'0000_0000 | R/W | Compare match timer counter bit 31 to 0 Note: When Write or Read in this bit, refer to 70.3.5 Register Access. |

70.2.4 Compare Match Timer Constant Register (CMSCOR)

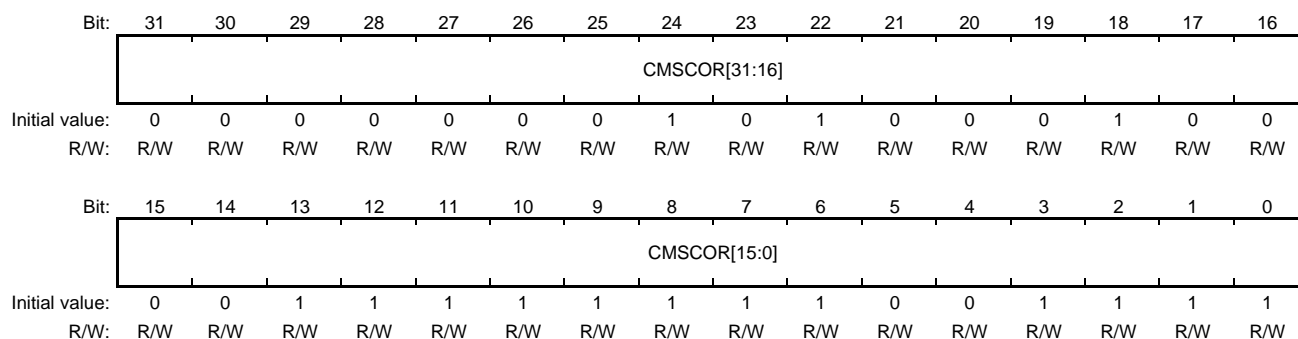
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMSCOR is a 32-bit register which specify the compare match period with the compare match timer counter (CMSCNT).

The initial value of this register is H'0144_3FCF.

When the 16-bit counter operation is selected by the CMS bit in the compare match timer control/status register (CMSCSR), bits 15 to 0 of this register become valid. When data is written to this register in 16-bit mode, write H'0000 as upper 16-bits data.

An overflow is detected when CMSCNT is cleared to 0 and this register is set to H'FFFF_FFFF.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 31 to 0 | CMSCOR [31:0] | H'0144_3FCF | R/W | Compare match timer constant register bit31 to 0 Note: When Write or Read in this bit, refer to 70.3.5 Register Access. |

70.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

70.3.1 Counter Operation

System Timer starts the operation of the counter by writing 1 to the STR5 bit in CMSSTR after each register has been set. Complete all of the settings before starting the operation. Do not change the register settings other than by clearing flag bits during the compare match timer (CMSCNT) operation.

The counter operates in one of two ways.

- One-Shot Operation

One-shot operation is selected by setting the CMM bit in CMSCSR to 0. When the value in CMSCNT matches the value in CMSCOR, the value in CMSCNT is cleared to H'0000_0000, and the CMF bit in CMSCSR is set to 1. Counting by CMSCNT stops after it has been cleared.

To detect an overflow interrupt, set the value in CMSCOR to H'FFFF_FFFF. When the value in CMSCNT matches the value in CMSCOR, CMSCNT is cleared to H'0000_0000 and the CMF and OVF bits in CMSCSR are set to 1

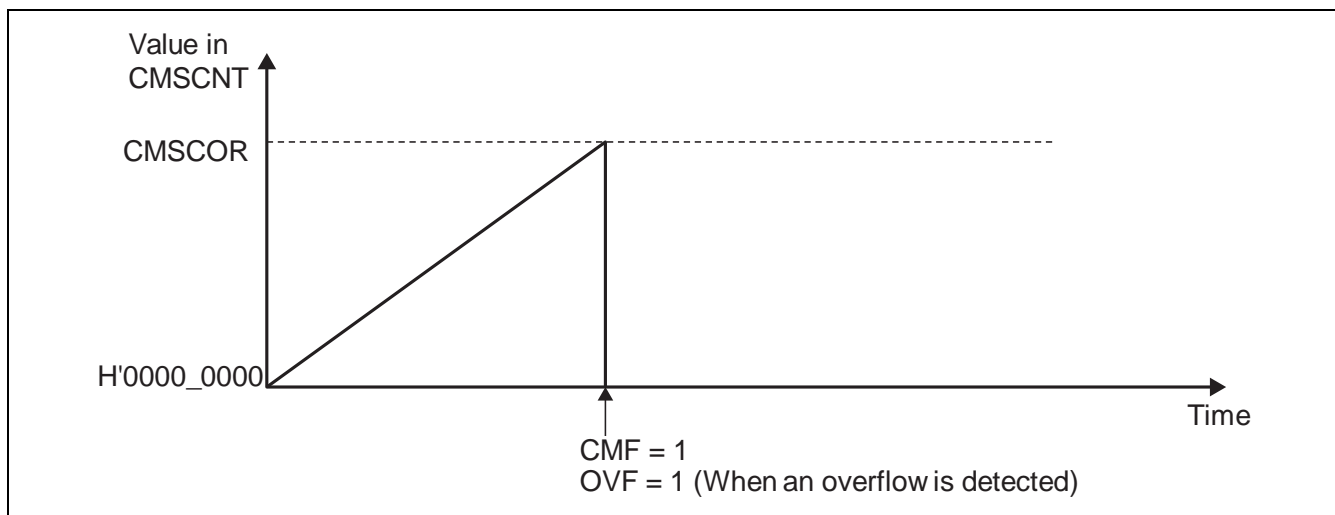


Figure 70.2 Counter Operation (One-Shot Operation)

- Free-Running Operation

Free-running operation is selected by setting the CMM bit in CMSCSR to 1. When the value in CMSCNT matches the value in CMSCOR, CMSCNT is cleared to H'0000_0000 and the CMF bit in CMSCSR is set to 1. CMSCNT resumes counting-up after it has been cleared.

To detect an overflow interrupt, set CMSCOR to H'FFFF_FFFF. When the values in CMSCNT and CMSCOR match, CMSCNT is cleared to H'0000_0000 and the CMF and OVF bits in CMSCSR are set to 1.

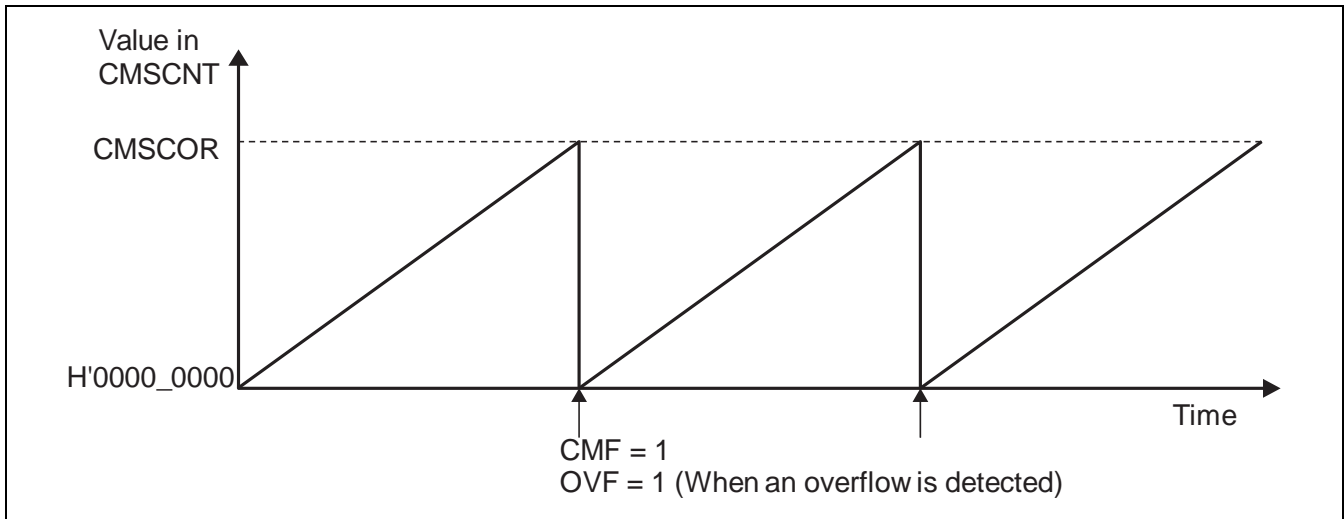


Figure 70.3 Counter Operation (Free-Running Operation)

70.3.2 Counter Size

In this module, the size of the counter can be selected from 16 or 32 bits. This is selected by the CMS bit in CMSCSR.

When the 16-bit size is selected, use H'0000 for upper 16-bit of write data for CMSCOR. To detect an overflow interrupt, CMSCOR must be set to H'0000_FFFF.

70.3.3 Timing for Counting by CMSCNT

In this module, the clock for the CMSCNT counter can be selected from following clocks:

- OSCCLK: 1/1, 1/8, 1/32, and 1/128

The clock for the counter is selected by the CKS bits in CMSCSR. CMSCNT is incremented at the rising edge of the selected clock.

70.3.4 Internal Interrupt Request to CPU

The setting of the CMR bits in CMSCSR selects the sending of an internal interrupt request to the CPU at a compare match.

To clear an internal interrupt request to the CPU in Free-Running Operation, the CMF bit should be set to b'0. Set the CMF bit to b'0 in the handling routine for the System Timer interrupt.

To clear an internal interrupt request to the CPU with resuming count-up in One-Shot Operation, the CMF bit should be set to b'0 while keeping b'1 in STR5 bit in CMSSTR. Set the CMF bit to b'0 while keeping b'1 in STR5 bit in the handling routine for the System Timer interrupt.

To clear an internal interrupt request to the CPU without resuming count-up in One-Shot Operation, the CMF bit should be set to b'0 after setting STR5 bit in CMSSTR to b'0. Set the CMF bit to b'0 after setting STR5 bit to b'0 in the handling routine for the System Timer interrupt.

70.3.5 Register Access

After writing to following registers, written data can be read 2 cycles in counter input clock (OSCCLK) after writing has finished. And it takes 2 cycles in counter input clock (OSCCLK), for reflecting written data to counter behavior.

CMSCSR: Bits CKS, CMM, CMS, CMR, DBGIVD

CMSCOR: Bits 31 to 0

CMSSTR: Bit STR5

After writing to following registers, written data can be read 2 cycles in counter input clock (OSCCLK) after writing has finished. And it takes 2 cycles in counter input clock (OSCCLK), for reflecting written data to counter behavior.

And for following registers, write access is prohibited, while previously written data is under writing. Do not perform next write access, while CMSCSR.WRFLG is 1.

CMSCNT: Bits 31 to 0

70.3.6 Compare Match Flag Set/Clear Timing

The CMF bit in CMSCSR is set to 1 by the compare match signal generated when CMSCOR and CMSCNT match. The compare match signal is generated upon the final state of the match (timing at which the CMSCNT value is updated to H'0000_0000). Consequently, after CMSCOR and CMSCNT match, a compare match signal will not be generated until a CMSCNT counter clock is input. Figure 70.4 shows the set timing of the CMF bit.

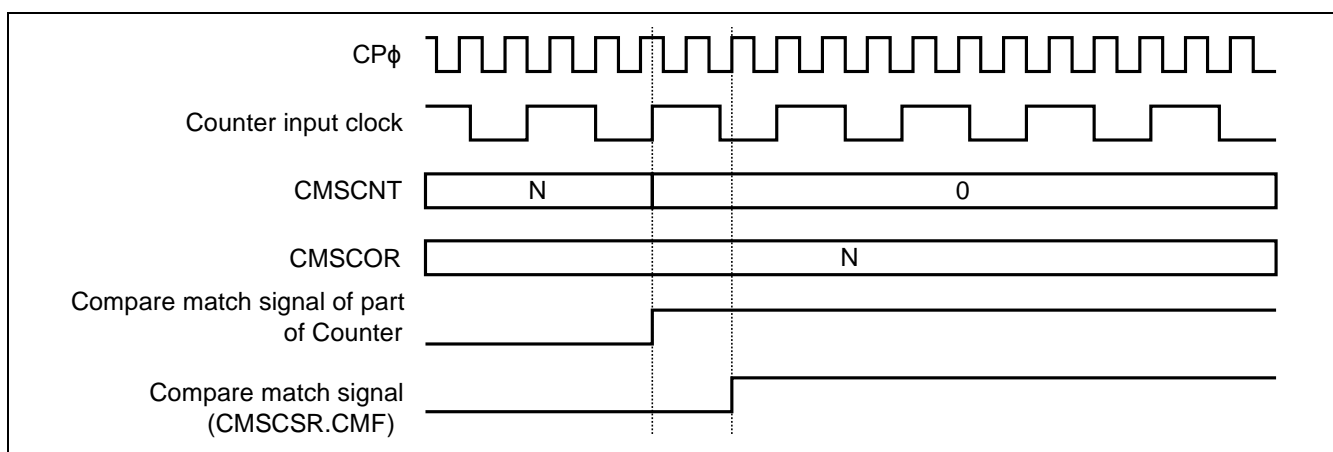


Figure 70.4 CMF Set Timing

Set the CMF bit as 0. The CMF flag is cleared immediately.

70.3.7 System Timer Usage

Take the following steps to use the System Timer.

1. Clear the STR5 bit in CMSSTR to 0 to temporarily stop counting.
(Confirm that OSCCLK was input more than 2 cycles.)
2. Write H'0000_0000 on CMSCNT.
3. Set counter size, compare match mode, kind of count clock and interrupt request and clear the bit of OVF and CMF in CMSCSR.
4. Set the value on CMSCOR.
5. Confirm that CMSCSR.WRFLG becomes 0.
If WRFLG is 1, wait until it'll be 0.
6. Start the counting by setting the STR5 bit in CMSSTR to 1.
(Refer to 70.3.5 Register Access)

71. System Up-Time Clock (SUCMT)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

71.1 Overview

System Up-Time Clock is an internal 32-bit compare match timer of one channel.

71.1.1 Features

- One channel is installed.
- 16 bits/32 bits can be selected as counter size (bit-width).
- Provided with 32-bit constant registers and 32-bit up-counters that can be written or read at any time.
- Following four clocks can be selected as counter clocks.
 - RCLK: 1/1, 1/8, 1/32, and 1/128
- One-shot operation or free-running operation is selectable.
- Compare match or overflow can be selected as interrupt source.
- Module standby mode can be set.
- Counter operation can be enabled or disabled at the time of debugging of CPU core using the debugging mode operation selector.

71.1.2 Block Diagram

Figure 71.1 shows a block diagram of the System Up-Time Clock.

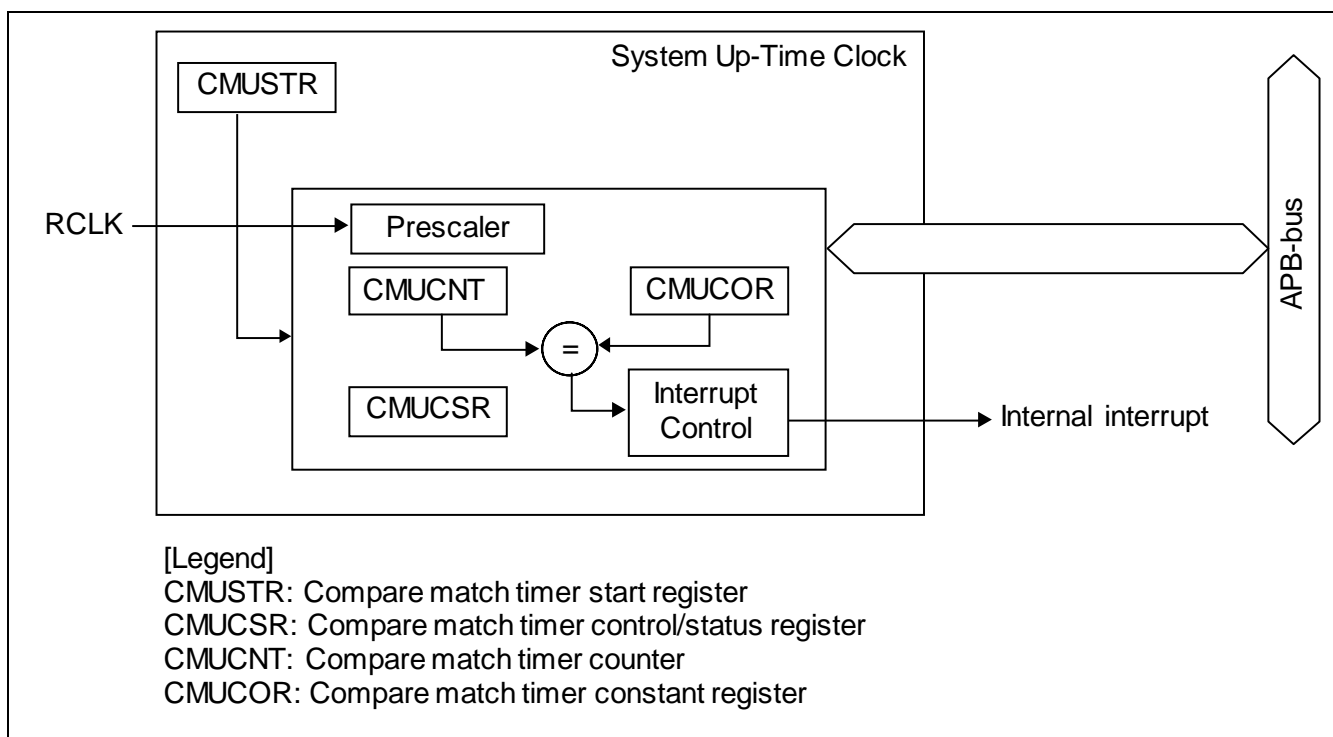


Figure 71.1 Block Diagram of System Up-Time Clock

71.1.3 Register Configuration

Table 71.1 shows the System Up-Time Clock register configuration. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined. Access CMUSTR and CMUCSR as a word (16 bits). Operation cannot be guaranteed if CMUSTR and CMUCSR are not accessed as a word. Access CMUCNT and CMUCOR as a longword (32 bits). Operation cannot be guaranteed if CMUCNT and CMUCOR are not accessed as a longword.

Table 71.1 Register Configurations

| Register Name | Abbreviation | R/W | Address | Value after Power-On Reset | Access Size | Second Generation RZ/G Series Products | | | |
|---|--------------|------|-------------|----------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Compare match timer start register | CMUSTR | R/W | H'E61D_0000 | H'0000 | 16 | √ | √ | √ | √ |
| Compare match timer control/status register | CMUCSR | R/W* | H'E61D_0040 | H'010F | 16 | √ | √ | √ | √ |
| Compare match timer counter | CMUCNT | R/W | H'E61D_0044 | H'0000_0000 | 32 | √ | √ | √ | √ |
| Compare match timer constant register | CMUCOR | R/W | H'E61D_0048 | H'FFFF_FFFF | 32 | √ | √ | √ | √ |

Note: * Except for bit [15:14] that are R/WC0 register.

Table 71.2 shows the register states in each operating mode.

Table 71.2 Register States in Each Operating Mode

| Register Abbreviation | Power-on Reset/Software Reset * | Module Standby |
|------------------------------|--|-----------------------|
| CMUSTR | Initialized | Retained |
| CMUCSR | Initialized | Retained |
| CMUCNT | Initialized | Retained |
| CMUCOR | Initialized | Retained |

Note: * For details, refer to 'General Part section 17 Reset (RST) and '12. Module Standby, Software Reset.

71.2 Register Description

[Legend for Register Description]

Initial value: Register value after a reset.

—: Undefined value

R/W: Bit or field is readable and writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R: Read-only. The write value should always be initial value.

71.2.1 Compare Match Timer Start Register (CMUSTR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMUSTR is a 16-bit register that selects whether the compare match timer counter (CMUCNT) is operated or halted.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|---|---|---|---|------|---|---|---|---|---|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | STR5 | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R/W | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 15 to 6 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 | STR5 | B'0 | R/W | Count Start These bits specify start/halt of compare match timer counter (CMUCNT). 0: CMUCNT halts. 1: CMUCNT starts counting. |
| 4 to 0 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |

71.2.2 Compare Match Timer Control/Status Register (CMUCSR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMUCSR is a 16-bit register which indicates the occurrence of compare matches, enables interrupts, and sets the counter input clocks.

Do not change bits other than the CMF and OVF bits, while the compare match timer counter (CMUCNT) is under operation.

| | | | | | | | | | | | | | | | | |
|----------------|-------|-------|-------|------|----|----|-----|-----|---|-----|----------|------------|----------|-----|-----|-----|
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | CMF | OVF | WRFLG | WERR | — | — | CMS | CMM | — | WER | CMR[1:0] | DBGI VD | CKS[2:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| R/W: | R/WC0 | R/WC0 | R | R | R | R | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-------|--|
| 15 | CMF | B'0 | R/WC0 | <p>Compare Match Flag</p> <p>This flag indicates whether the values of the compare match timer counter (CMUCNT) and the compare match timer constant register (CMUCOR) have matched or not.</p> <p>Software cannot write 1 to the bit. When one-shot is selected for the counter operation, counting resumes by clearing this bit.</p> <p>0: CMUCNT and CMUCOR values have not matched. [Clearing condition]</p> <ul style="list-style-type: none"> Write 0 to CMF <p>1: CMUCNT and CMUCOR values have matched.</p> |
| 14 | OVF | B'0 | R/WC0 | <p>Overflow Flag</p> <p>This flag indicates whether the compare match timer counter (CMUCNT) has overflowed or not. Software cannot write 1 to this bit.</p> <p>0: CMUCNT has not overflowed. [Clearing condition]</p> <ul style="list-style-type: none"> Write 0 to OVF <p>1: CMUCNT has overflowed.</p> |
| 13 | WRFLG | B'0 | R | <p>Write State Flag</p> <p>When this bit is 1, write access to CMUCNT is prohibited. If it isn't maintained, operation isn't secured. This bit indicates the period that the writing to CMUCNT is masked for synchronization after the writing to CMUCNT. Confirm that this flag is 0 when CMUCNT is written to continuously.</p> |
| 12 | WERR | B'0 | R | <p>Write Access Error Flag</p> <p>This flag indicates whether or not the write access from the public domain. Software cannot write 1 to this bit.</p> <p>0: Public domain has not Write access [Clearing condition]</p> <ul style="list-style-type: none"> Read register <p>1: Public domain has Write access</p> |
| 11 to 10 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 9 | CMS | B'0 | R/W | <p>Compare Match Timer Counter Size</p> <p>This bit selects whether the compare match timer counter (CMUCNT) is used as a 16-bit counter or 32-bit counter. This register setting becomes the valid bit-size for the compare match timer constant register (CMUCOR).</p> <p>0: Operates as a 32-bit counter. 1: Operates as a 16-bit counter.</p> <p>Note: When Write or Read in this bit, refer to 71.3.5 Register Access.</p> |
| 8 | CMM | B'1 | R/W | <p>Compare Match Mode</p> <p>This bit selects one-shot operation or free-running operation of the counter.</p> <p>0: One-shot operation 1: Free-running operation</p> <p>Note: When Write or Read in this bit, refer to 71.3.5 Register Access.</p> |
| 7 | — | B'0 | R | <p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p> |
| 6 | WER | B'0 | R/W | <p>Write Access Error Request</p> <p>This bit Write Access Error interrupt request at the public domain.</p> <p>0: Disables an internal interrupt request. 1: Enables an internal interrupt request.</p> <p>Note: When Write or Read in this bit, refer to 71.3.5 Register Access.</p> |
| 5, 4 | CMR[1:0] | B'00 | R/W | <p>Compare Match Request</p> <p>These bits enable or disable an internal interrupt request at a compare match.</p> <p>B'00: Disables an internal interrupt request. B'01: Setting prohibited B'10: Enables an internal interrupt request. B'11: Setting prohibited</p> <p>Note1: When various standby modes are canceled by the System Up-Time Clock, these bits should be set to B'10. Note2: When Write or Read in this bit, refer to 71.3.5 Register Access.</p> |
| 3 | DBGIVD | B'1 | R/W | <p>Debug Mode Operation Select</p> <p>This bit sets counter operation when in debug mode.</p> <p>0: Stops counter operation when in debug mode. 1: Enables counter operation even when in debug mode.</p> <p>Note: When Write or Read in this bit, refer to 71.3.5 Register Access.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 2 to 0 | CKS[2:0] | B'111 | R/W | <p>Clock Select</p> <p>These bits select the clock input to CMUCNT. When the count start bit (STR5) in CMUSTR is set to 1, CMUCNT begins incrementing with the clock selected by these bits.</p> <p>B'000: Setting prohibited B'001: Setting prohibited B'010: Setting prohibited B'011: Setting prohibited B'100: RCLK/8 B'101: RCLK/32 B'110: RCLK/128 B'111: RCLK/1</p> <p>Note: When Write or Read in this bit, refer to 71.3.5 Register Access.</p> |

Note: * Only 0 can be written, for flag clearing.

71.2.3 Compare Match Timer Counter (CMUCNT)

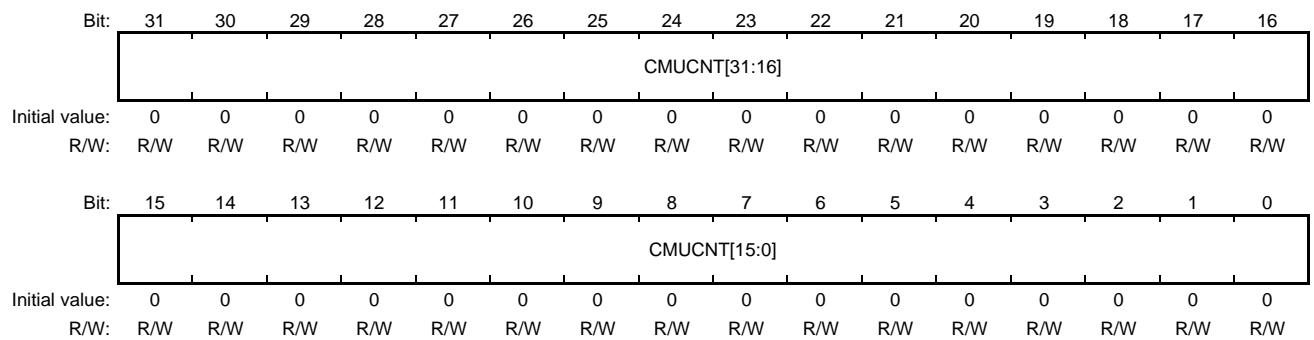
| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMUCNT is a 32-bit register which is used as an up-counter.

To specify counter operation, set compare match timer control/status register n (CMUCSR), before starting operation of corresponding channel.

When the 16-bit counter operation is selected by the CMS bit, bits 31 to 16 of this register become invalid. When data is written to this register in 16-bit mode, write H'0000 as upper 16bits

When CMUCNT is read during counter operation, the read value may be wrong because different clock is used between counter and bus-interface. For exact value, read this register continuously, until same values are read from this register.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 31 to 0 | CMUCNT [31:0] | H'0000_0000 | R/W | Compare match timer counter bit31 to 0 Note: When Write or Read in this bit, refer to 71.3.5 Register Access. |

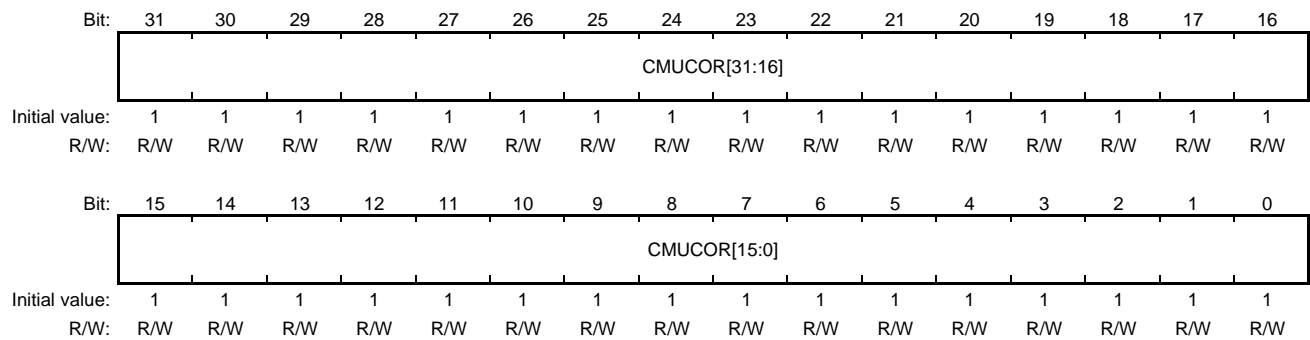
71.2.4 Compare Match Timer Constant Register (CMUCOR)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

CMUCOR is a 32-bit register which specify the compare match period with the compare match timer counter (CMUCNT).

When the 16-bit counter operation is selected by the CMS bit in the compare match timer control/status register (CMUCSR), bits 15 to 0 of this register become valid. Write H'0000 to upper 16bits in 16-bit counter operation.

An overflow is detected when CMUCNT is cleared to 0 and this register is set to H'FFFF_FFFF.



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------|---------------|-----|--|
| 31 to 0 | CMUCOR [31:0] | H'FFFF_FFFF | R/W | Compare match timer counter bit31 to 0 Note: When Write or Read in this bit, refer to 71.3.5 Register Access. |

71.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

71.3.1 Counter Operation

System Up-Time Clock starts the operation of the counter by writing 1 to the STR5 bit in CMUSTR after each register has been set. Complete all of the settings before starting the operation. Do not change the register settings other than by clearing flag bits during the compare match timer (CMUCNT) operation.

The counter operates in one of two ways.

- One-Shot Operation

One-shot operation is selected by setting the CMM bit in CMUCSR to H'0. When the value in CMUCNT matches the value in CMUCOR, the value in CMUCNT is cleared to H'0000_0000, and the CMF bit in CMUCSR is set to 1. Counting by CMUCNT stops after it has been cleared.

To detect an overflow interrupt, set the value in CMUCOR to H'FFFF_FFFF. When the value in CMUCNT matches the value in CMUCOR, CMUCNT is cleared to H'0000_0000 and the CMF and OVF bits in CMUCSR are set to 1

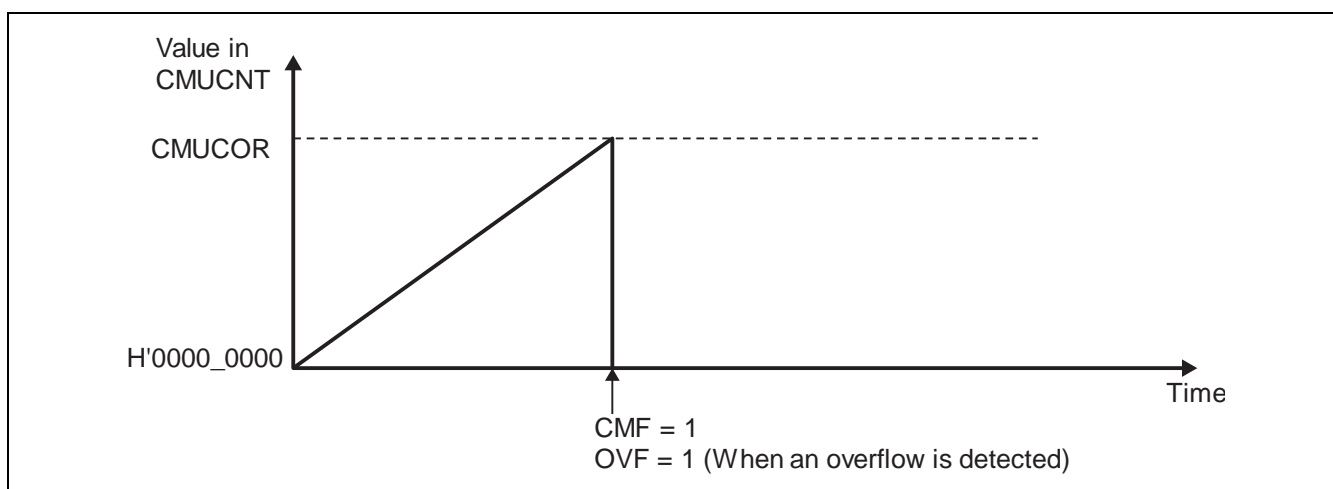


Figure 71.2 Counter Operation (One-Shot Operation)

- Free-Running Operation

Free-running operation is selected by setting the CMM bit in CMUCSR to 1. When the value in CMUCNT matches the value in CMUCOR, CMUCNT is cleared to H'0000_0000 and the CMF bit in CMUCSR is set to 1. CMUCNT resumes counting-up after it has been cleared.

To detect an overflow interrupt, set CMUCOR to H'FFFF_FFFF. When the values in CMUCNT and CMUCOR match, CMUCNT is cleared to H'0000_0000 and the CMF and OVF bits in CMUCSR are set to 1.

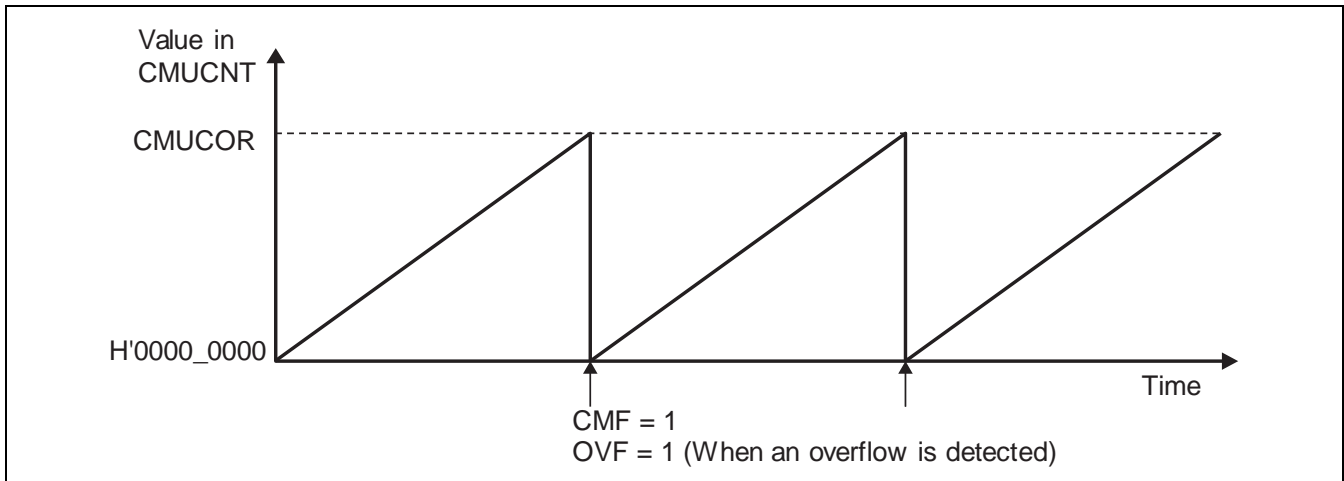


Figure 71.3 Counter Operation (Free-Running Operation)

71.3.2 Counter Size

In this module, the size of the counter can be selected from either 16, or 32 bits. This is selected by the CMS bit in CMUCSR.

When the 16-bit size is selected, H'0000 should be used as upper 16bit of write data to CMUCOR. To detect an overflow interrupt, CMUCOR must be set to H'0000_FFFF.

71.3.3 Timing for Counting by CMUCNT

In this module, the clock for the CMUCNT counter can be selected from following clocks:

- RCLK: 1/1, 1/8, 1/32, and 1/128

The clock for the counter is selected by the CKS bits in CMUCSR. CMUCNT is incremented at the rising edge of the selected clock.

71.3.4 Internal Interrupt Request to CPU

By CMR bits in CMUCSR, interrupt request to CPU can be asserted, at a compare match.

To clear an internal interrupt request to the CPU in Free-Running Operation, the CMF bit should be set to 0. Set the CMF bit to 0 in the handling routine for the System Up-Time Clock interrupt.

To clear an internal interrupt request to the CPU with resuming count-up in One-Shot Operation, the CMF bit should be set to 0 while keeping 1 in STR5 bit in CMUSTR. Set the CMF bit to 0 while keeping 1 in STR5 bit in the handling routine for the System Up-Time Clock interrupt.

To clear an internal interrupt request to the CPU without resuming count-up in One-Shot Operation, the CMF bit should be set to 0 after setting STR5 bit in CMUSTR to 0. Set the CMF bit to 0 after setting STR5 bit to 0 in the handling routine for the System Up-Time Clock interrupt.

71.3.5 Register Access

After writing to following registers, written data can be read 2 cycles in counter input clock (RCLK) after writing has finished. And it takes 2 cycles in counter input clock (RCLK), for reflecting written data to counter behavior.

CMUCSR: Bits CKS, CMM, CMS, WER, CMR, DBGIVD

CMUCOR: Bits 31 to 0

CMUSTR: Bit STR5

After writing to following registers, written data can be read 2 cycles in counter input clock (RCLK) after writing has finished. And it takes 2 cycles in counter input clock (RCLK), for reflecting written data to counter behavior.

And for following registers, write access is prohibited, while previously written data is under writing. Do not perform next write access, while CMUCSR.WRFLG is 1.

CMUCNT: Bits 31 to 0

71.3.6 Compare Match Flag Set/Clear Timing

The CMF bit in CMUCSR is set to 1 by the compare match signal generated when CMUCOR and CMUCNT match. The compare match signal is generated upon the final state of the match (timing at which the CMUCNT value is updated to H'0). Consequently, after CMUCOR and CMUCNT match, a compare match signal will not be generated until a CMUCNT counter clock is input. Figure 71.4 shows the set timing of the CMF bit.

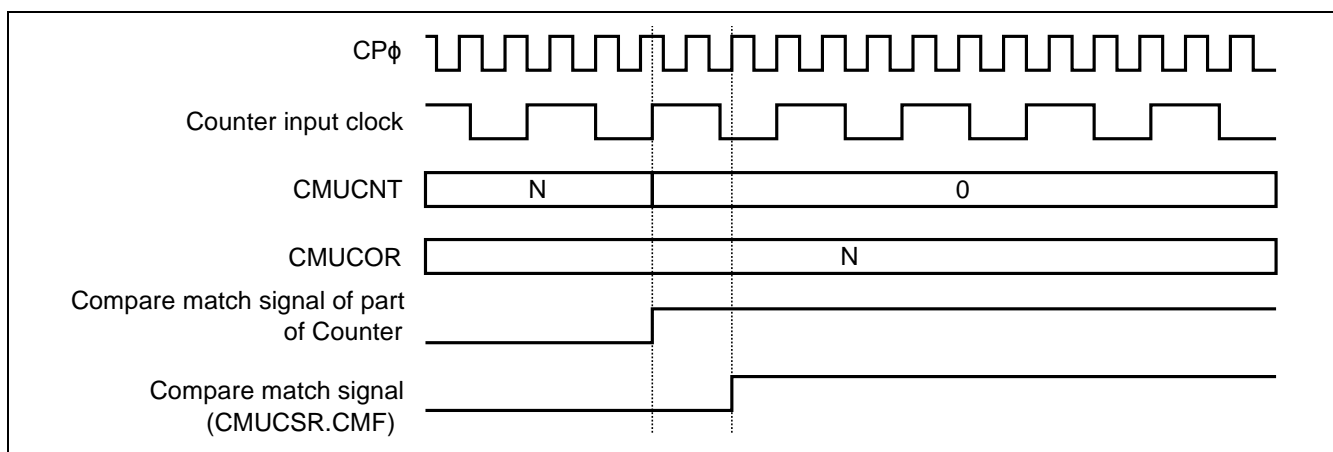


Figure 71.4 CMF Set Timing

Set the CMF bit to 0. The CMF flag is cleared immediately.

71.3.7 System Up-Tim Clock Usage

Take the following steps to use the System Up-Time Clock.

1. Clear the STR5 bit in CMUSTR to 0 to temporarily stop counting.
(Confirm that RCLK was input more than 2 cycles.)
2. Write H'0000_0000 on CMUCNT.
3. Set counter size, compare match mode, kind of count clock and interrupt request and clear the bit of OVF and CMF in CMUCSR.
4. Set the value on CMUCOR.
5. Confirm that CMUCSR.WRFLG becomes 0.
If WRFLG is 1, wait until it'll be 0.
6. Start the counting by setting the STR5 bit in CMUSTR to 1.
(Refer to 71.3.5 Register Access)

72. Debug and Trace

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

72.1 Overview

This module controls debug function of Cortex-A57 and Cortex-A53 via JTAG ports.

72.1.1 Features

72.1.1.1 RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 features

- JTAG interface
 - Supports dedicated 5-pin JTAG and SWD
- The cross trigger function between following IPs.
 - Cortex-A57 CPU_n
 - Cortex-A53 CPU_n.
 - CoreSight
- Tracing function
 - A maximum of 16 bits × 266 Mbps (133MHz DDR) trace*¹ data pin output
 - A maximum of 4 bits × 266 Mbps (133MHz DDR) trace*¹ data pin output
 - 16-Kbyte embedded trace FIFO each Cortex-A57 and Cortex-A53 cluster
 - 4-Kbyte embedded trace FIFO for CoreSight

Notes: 1. Trace bandwidth of internal logic. This is limited by I/O buffer performance.

72.1.1.2 RZ/G2N features

- JTAG interface
 - Supports dedicated 5-pin JTAG and SWD
- The cross trigger function between following IPs.
 - Cortex-A57 CPUn
 - CoreSight
- Tracing function
 - A maximum of 16 bits × 266 Mbps (133MHz DDR) trace*¹ data pin output
 - A maximum of 4 bits × 266 Mbps (133MHz DDR) trace*¹ data pin output
 - 16-Kbyte embedded trace FIFO for Cortex-A57
 - 4-Kbyte embedded trace FIFO for CoreSight

Notes: 1. Trace bandwidth of internal logic. This is limited by I/O buffer performance.

72.1.1.3 RZ/G2E features

- JTAG interface
 - Supports dedicated 5-pin JTAG and SWD
- The cross trigger function between following IPs.
 - Cortex-A53 CPU_n
 - CoreSight
- Tracing function
 - A maximum of 16 bits × 266 Mbps (133MHz DDR) trace*¹ data pin output
 - A maximum of 4 bits × 266 Mbps (133MHz DDR) trace*¹ data pin output
 - 16-Kbyte embedded trace FIFO for Cortex-A57
 - 4-Kbyte embedded trace FIFO for CoreSight

Notes: 1 Trace bandwidth of internal logic. This is limited by I/O buffer performance.

72.1.2 Block Diagram

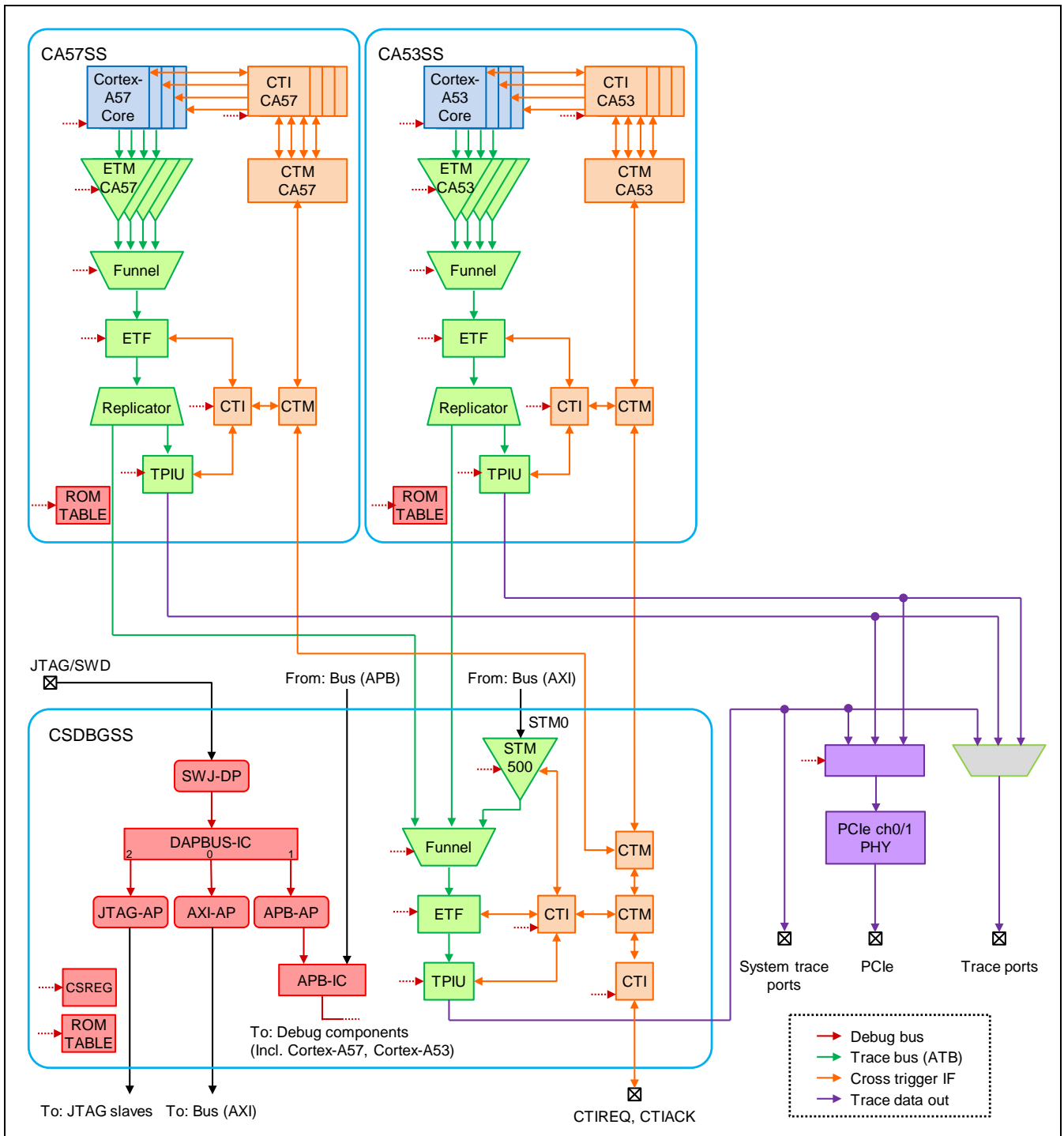


Figure 72.1 CoreSight Block Diagram for RZ/G2H

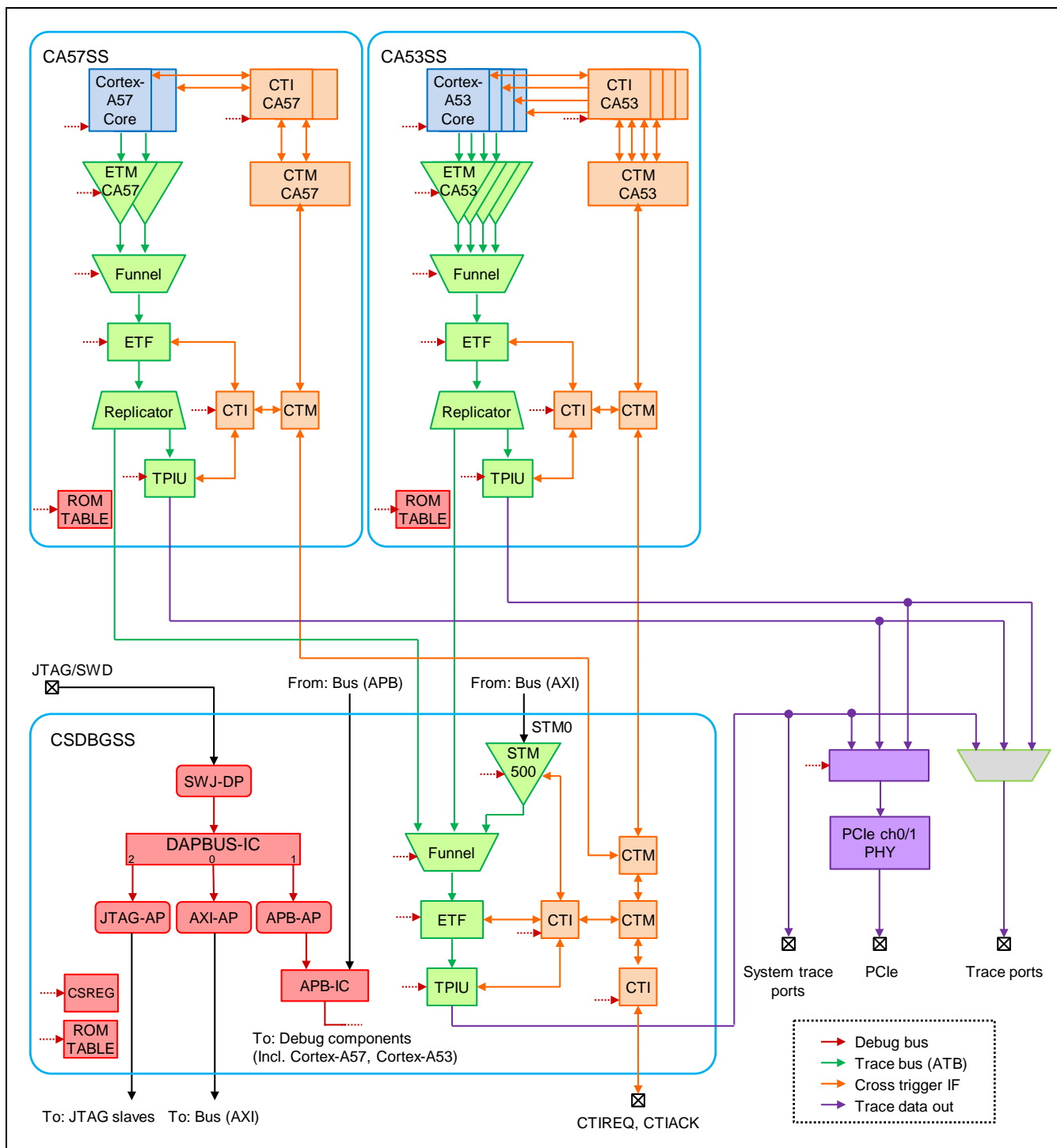


Figure 72.2 CoreSight Block Diagram for RZ/G2M V1.3, RZ/G2M V3.0

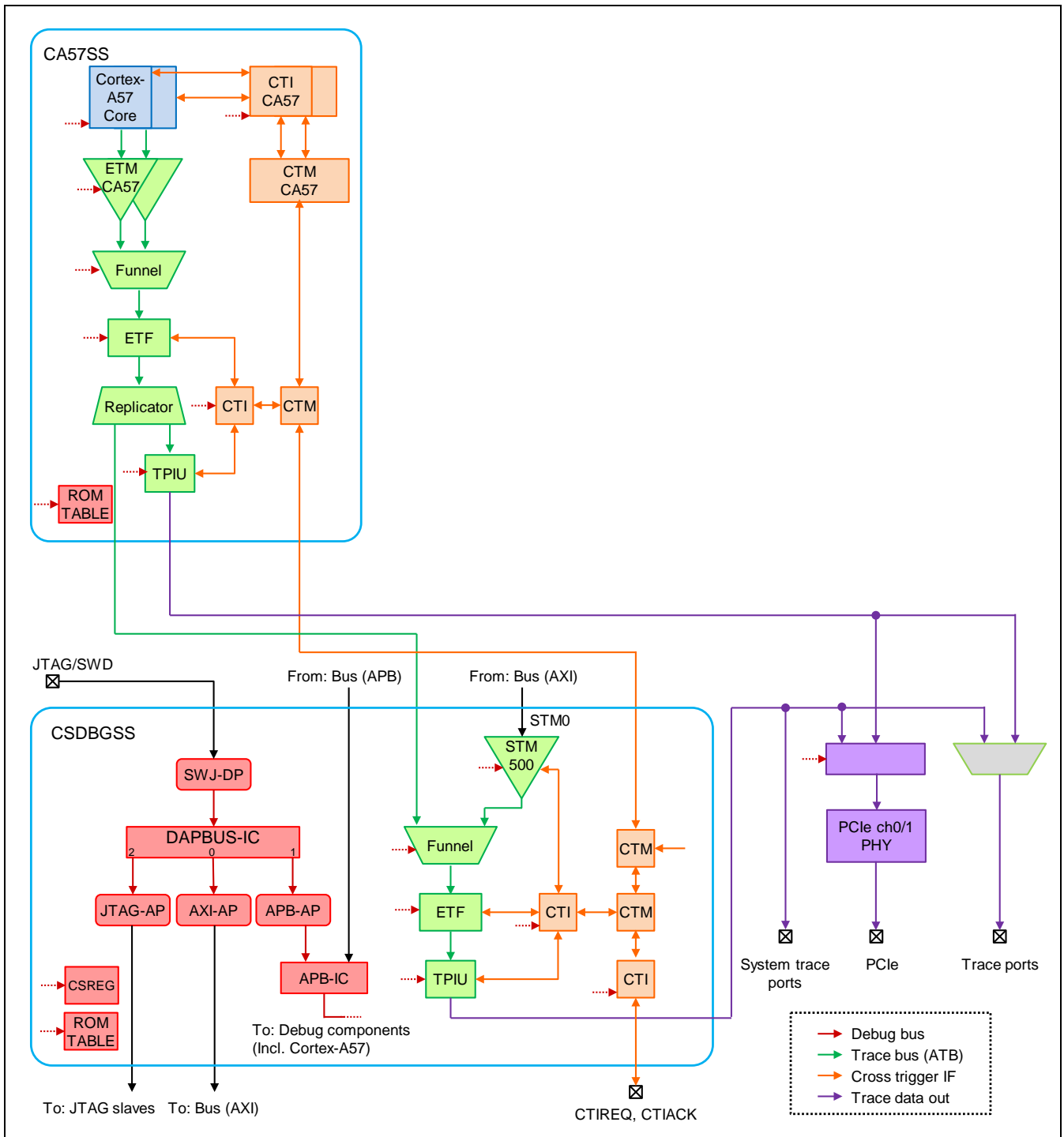


Figure 72.3 CoreSight Block Diagram for RZ/G2N

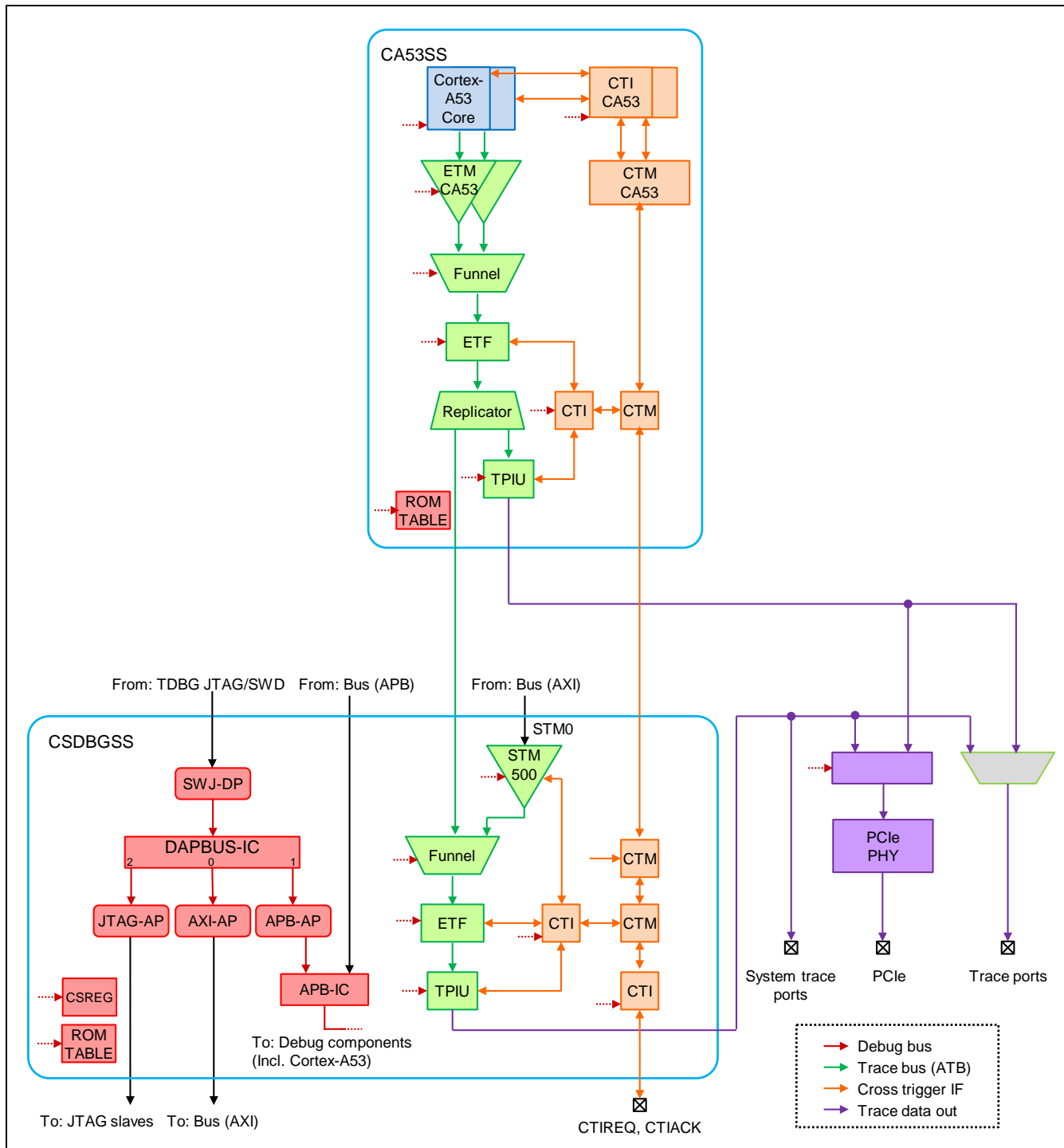


Figure 72.4 CoreSight Block Diagram for RZ/G2E

72.1.3 External Pins

72.1.3.1 JTAG Pins

5-pin JTAG and SWD are supported as dedicated JTAG port. JTAG ports support 1.8V only. Each JTAG port can be connected to different TAP controller through MD pins. Refer to JTAG selection tables in Table 72.12 and Table 72.13.

Table 72.1 JTAG pin for RZ/G2H

| IF | Pin name | I/O | Function * | |
|------|----------|--------|------------|------------------------------------|
| | | | JTAG IF | Serial Wire Debug IF for CoreSight |
| JTAG | TRST# | Input | TRST# | — |
| | TCK | Input | TCK | SWCLK |
| | TMS | I/O | TMS | SWDIO |
| | TDI | Input | TDI | — |
| | TDO | Output | TDO | SWO |

Note: * The pin operates as JTAG IF from startup period. It can be switched to Serial Wire Debug IF by control from the debugger side. (It cannot be switched by pin setting)

Table 72.2 JTAG pin for RZ/G2M V1.3, RZ/G2M V3.0

| IF | Pin name | I/O | Function * | |
|------|----------|--------|------------|------------------------------------|
| | | | JTAG IF | Serial Wire Debug IF for CoreSight |
| JTAG | TRST# | Input | TRST# | — |
| | TCK | Input | TCK | SWCLK |
| | TMS | I/O | TMS | SWDIO |
| | TDI | Input | TDI | — |
| | TDO | Output | TDO | SWO |

Note: * The pin operates as JTAG IF from startup period. It can be switched to Serial Wire Debug IF by control from the debugger side. (It cannot be switched by pin setting)

Table 72.3 JTAG pin for RZ/G2N

| IF | Pin name | I/O | Function * | |
|------|----------|--------|------------|------------------------------------|
| | | | JTAG IF | Serial Wire Debug IF for CoreSight |
| JTAG | TRST# | Input | TRST# | — |
| | TCK | Input | TCK | SWCLK |
| | TMS | I/O | TMS | SWDIO |
| | TDI | Input | TDI | — |
| | TDO | Output | TDO | SWO |

Note: * The pin operates as JTAG IF from startup period. It can be switched to Serial Wire Debug IF by control from the debugger side. (It cannot be switched by pin setting)

Table 72.4 JTAG pin for RZ/G2E

| IF | Pin name | I/O | Function * | |
|------|----------|--------|------------|------------------------------------|
| | | | JTAG IF | Serial Wire Debug IF for CoreSight |
| JTAG | TRST# | Input | TRST# | — |
| | TCK | Input | TCK | SWCLK |
| | TMS | I/O | TMS | SWDIO |
| | TDI | Input | TDI | — |
| | TDO | Output | TDO | SWO |

Note: * The pin operates as JTAG IF from startup period. It can be switched to Serial Wire Debug IF by control from the debugger side. (It cannot be switched by pin setting)

72.1.3.2 Trace Pins

A maximum of 16-bit trace ports are supported. Trace ports support 1.8V only.

Table 72.5 Arm Trace Pins

| Name | Pin Name | I/O | Function | Second Generation RZ/G Series Products | | | |
|------------------|----------|--------|--|---|----------------------------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| ARM_TRACEDATA_15 | TMONDT19 | Output | Trace Data 15 | √ | √ | √ | √ |
| ARM_TRACEDATA_14 | TMONDT18 | Output | Trace Data 14 | √ | √ | √ | √ |
| ARM_TRACEDATA_13 | TMONDT17 | Output | Trace Data 13 | √ | √ | √ | √ |
| ARM_TRACEDATA_12 | TMONDT16 | Output | Trace Data 12 | √ | √ | √ | √ |
| ARM_TRACEDATA_11 | TMONDT15 | Output | Trace Data 11 | √ | √ | √ | √ |
| ARM_TRACEDATA_10 | TMONDT14 | Output | Trace Data 10 | √ | √ | √ | √ |
| ARM_TRACEDATA_9 | TMONDT13 | Output | Trace Data 9 | √ | √ | √ | √ |
| ARM_TRACEDATA_8 | TMONDT12 | Output | Trace Data 8 | √ | √ | √ | √ |
| TRACECTL | TMONDT11 | Output | Trace Control Output | √ | √ | √ | √ |
| ARM_TRACEDATA_0 | TMONDT10 | Output | Trace Data 0 | √ | √ | √ | √ |
| ARM_TRACEDATA_7 | TMONDT9 | Output | Trace Data 7 | √ | √ | √ | √ |
| ARM_TRACEDATA_6 | TMONDT8 | Output | Trace Data 6 | √ | √ | √ | √ |
| ARM_TRACEDATA_5 | TMONDT7 | Output | Trace Data 5 | √ | √ | √ | √ |
| ARM_TRACEDATA_4 | TMONDT6 | Output | Trace Data 4 | √ | √ | √ | √ |
| ARM_TRACEDATA_3 | TMONDT5 | Output | Trace Data 3 | √ | √ | √ | √ |
| ARM_TRACEDATA_2 | TMONDT4 | Output | Trace Data 2 | √ | √ | √ | √ |
| ARM_TRACEDATA_1 | TMONDT3 | Output | Trace Data 1 | √ | √ | √ | √ |
| CTIACK | TMONDT2 | Input | Cross Trigger Interface acknowledge | √ | √ | √ | √ |
| CTIREQ | TMONDT1 | Output | Cross Trigger Interface Request | √ | √ | √ | √ |
| TRACECLK | TMONDT0 | Output | Trace Clock output | √ | √ | √ | √ |

72.1.3.3 System Trace Pins

2 ports are supported for system trace. System trace ports support 1.8V only.

Table 72.6 System Trace Pins

| Function Name | Pin Name | I/O | Second Generation RZ/G Series Products | | | |
|---------------|------------|--------|---|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| STMCLK | STMCLK_1 | Output | √ | √ | √ | √ |
| STMDATA3 | STMDATA3_1 | Output | √ | √ | √ | √ |
| STMDATA2 | STMDATA2_1 | Output | √ | √ | √ | √ |
| STMDATA1 | STMDATA1_1 | Output | √ | √ | √ | √ |
| STMDATA0 | STMDATA0_1 | Output | √ | √ | √ | √ |
| STMCLK | STMCLK_2 | Output | √ | √ | √ | √ |
| STMDATA3 | STMDATA3_2 | Output | √ | √ | √ | √ |
| STMDATA2 | STMDATA2_2 | Output | √ | √ | √ | √ |
| STMDATA1 | STMDATA1_2 | Output | √ | √ | √ | √ |
| STMDATA0 | STMDATA0_2 | Output | √ | √ | √ | √ |

72.1.4 Register Configuration

Do not write to addresses other than those listed below, otherwise normal operation cannot be guaranteed. Values read from other addresses are undefined.

[Legend]

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R: Read-only. The write value should always be the initial value.

—/W: Write-only. The read value is undefined.

72.1.4.1 Debug resource configurations

Table 72.7 CoreSight Address Map

| | | Component | | Base Address | | Second Generation RZ/G Series Products | | | | |
|---------------|-------------|--------------|---------------------|--------------------------|--------------------------------|---|----------------------------|--------|--------|--|
| Module | Name | Abbreviation | CPU view Address | Debugger view Address | Offset from Base Address | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| CSDBGSS | ROM Table | CSD-ROM | H'00 EA00_0000 | H'8000_0000 | H'00_0000 | √ | √ | √ | √ | |
| | STM0 | CSD-STM0 | | | H'01_0000 | √ | √ | √ | √ | |
| | ATB Funnel | CSD-FUN | | | H'02_0000 | √ | √ | √ | √ | |
| | ETF | CSD-ETF | | | H'03_0000 | √ | √ | √ | √ | |
| | TPIU | CSD-TPI | | | H'04_0000 | √ | √ | √ | √ | |
| | CTI0 | CSD-CTI0 | | | H'08_0000 | √ | √ | √ | √ | |
| | CTI1 | CSD-CTI1 | | | H'09_0000 | √ | √ | √ | √ | |
| | CSREG | CSD-REG | | | H'0F_0000 | √ | √ | √ | √ | |
| CA57SS GCA | ROM Table | A57CA-ROM | H'00 EA20_0000 | H'8020_0000 | H'00_0000 | √ | √ | √ | — | |
| | ETF | A57CA-ETF | | | H'01_0000 | √ | √ | √ | — | |
| | CTI | A57CA-CTI | | | H'02_0000 | √ | √ | √ | — | |
| | TPIU | A57CA-TPI | | | H'03_0000 | √ | √ | √ | — | |
| | ATB Funnel | A57CA-FUN | | | H'04_0000 | √ | √ | √ | — | |
| MP Core | ROM Table | A57MP-ROM | H'00 EA40_0000 | H'8040_0000 | H'00_0000 | √ | √ | √ | — | |
| | CPU 0 Debug | A57MP-DBG0 | | | H'01_0000 | √ | √ | √ | — | |
| | CPU 0 CTI | A57MP-CTI0 | | | H'02_0000 | √ | √ | √ | — | |
| | CPU 0 PMU | A57MP-PMU0 | | | H'03_0000 | √ | √ | √ | — | |
| | CPU 0 ETM | A57MP-ETM0 | | | H'04_0000 | √ | √ | √ | — | |
| | CPU 1 Debug | A57MP-DBG1 | | | H'11_0000 | √ | √ | √ | — | |
| | CPU 1 CTI | A57MP-CTI1 | | | H'12_0000 | √ | √ | √ | — | |
| | CPU 1 PMU | A57MP-PMU1 | | | H'13_0000 | √ | √ | √ | — | |
| | CPU 1 ETM | A57MP-ETM1 | | | H'14_0000 | √ | √ | √ | — | |
| | CPU 2 Debug | A57MP-DBG2 | | | H'21_0000 | √ | — | — | — | |
| | CPU 2 CTI | A57MP-CTI2 | | | H'22_0000 | √ | — | — | — | |
| | CPU 2 PMU | A57MP-PMU2 | | | H'23_0000 | √ | — | — | — | |
| | CPU 2 ETM | A57MP-ETM2 | | | H'24_0000 | √ | — | — | — | |
| | CPU 3 Debug | A57MP-DBG3 | | | H'31_0000 | √ | — | — | — | |
| | CPU 3 CTI | A57MP-CTI3 | | | H'32_0000 | √ | — | — | — | |
| CPU 3 PMU | A57MP-PMU3 | | | H'33_0000 | √ | — | — | — | | |
| CPU 3 ETM | A57MP-ETM3 | | | H'34_0000 | √ | — | — | — | | |

| | | Component | | Base Address | | Second Generation RZ/G Series Products | | | | |
|------------|-------------|--------------|-----------|----------------|--------------------------|--|----------------------------|--------|--------|---|
| Module | Name | Abbreviation | CPU view | Debugger | Offset from Base Address | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| | | | Address | view Address | | | | | | |
| CA53SS | CSDB GCA | ROM Table | A53CA-ROM | H'00 EAA0_0000 | H'80A0_0000 | H'00_0000 | √ | √ | — | √ |
| | | ETF | A53CA-ETF | | | H'01_0000 | √ | √ | — | √ |
| | | CTI | A53CA-CTI | | | H'02_0000 | √ | √ | — | √ |
| | | TPIU | A53CA-TPI | | | H'03_0000 | √ | √ | — | √ |
| | | ATB Funnel | A53CA-FUN | | | H'04_0000 | √ | √ | — | √ |
| MP Core | ROM Table | A53MP-ROM | H'00 | | H'80C0_0000 | H'00_0000 | √ | √ | — | √ |
| | CPU 0 Debug | A53MP-DBG0 | EAC0_0000 | | | H'01_0000 | √ | √ | — | √ |
| | CPU 0 CTI | A53MP-CTI0 | | | | H'02_0000 | √ | √ | — | √ |
| | CPU 0 PMU | A53MP-PMU0 | | | | H'03_0000 | √ | √ | — | √ |
| | CPU 0 ETM | A53MP-ETM0 | | | | H'04_0000 | √ | √ | — | √ |
| | CPU 1 Debug | A53MP-DBG1 | | | | H'11_0000 | √ | √ | — | √ |
| | CPU 1 CTI | A53MP-CTI1 | | | | H'12_0000 | √ | √ | — | √ |
| | CPU 1 PMU | A53MP-PMU1 | | | | H'13_0000 | √ | √ | — | √ |
| | CPU 1 ETM | A53MP-ETM1 | | | | H'14_0000 | √ | √ | — | √ |
| | CPU 2 Debug | A53MP-DBG2 | | | | H'21_0000 | √ | √ | — | — |
| | CPU 2 CTI | A53MP-CTI2 | | | | H'22_0000 | √ | √ | — | — |
| | CPU 2 PMU | A53MP-PMU2 | | | | H'23_0000 | √ | √ | — | — |
| | CPU 2 ETM | A53MP-ETM2 | | | | H'24_0000 | √ | √ | — | — |
| | CPU 3 Debug | A53MP-DBG3 | | | | H'31_0000 | √ | √ | — | — |
| | CPU 3 CTI | A53MP-CTI3 | | | | H'32_0000 | √ | √ | — | — |
| CPU 3 PMU | A53MP-PMU3 | | | | H'33_0000 | √ | √ | — | — | |
| CPU 3 ETM | A53MP-ETM3 | | | | H'34_0000 | √ | √ | — | — | |

Note: Supported CPU in each product is defined as below:

| | |
|----------------------------|--|
| [RZ/G2H] | Cortex-A53 CPU0/CPU1/CPU2/CPU3 Cortex-A57 CPU0/CPU1/CPU2/CPU3 |
| [RZ/G2M V1.3, RZ/G2M V3.0] | Cortex-A53 CPU0/CPU1/CPU2/CPU3 Cortex-A57 CPU0/CPU1 |
| [RZ/G2N] | Cortex-A57 CPU0/CPU1 |
| [RZ/G2E] | Cortex-A53 CPU0/CPU1 |

72.1.4.2 CSreg registers configurations

CSreg is the collection of control registers for chip-specific debug features. These registers can be accessed from both Debugger and CPU via Debug-APB.

Table 72.8 CSreg register configurations

| Name of Register | Abbreviation | R/W | CPU view Address | Debugger view Address | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|------------------|--------------|-----|------------------|-----------------------|---------------|-------------|--|----------------------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| ATCLKCR_CA57 | — | R/W | H'EA0F_0100 | H'800F_0100 | H'0000_0101 | 32 | √ | √ | √ | — |
| CSREG_E00 | — | R/W | H'EA0F_0E00 | H'800F_0E00 | H'0000_0000 | 32 | — | — | — | √ |
| LOCKACCESS | — | —/W | H'EA0F_0FB0 | H'800F_0FB0 | — | 32 | √ | √ | √ | √ |
| LOCKSTATUS | — | R | H'EA0F_0FB4 | H'800F_0FB4 | H'0000_0003 | 32 | √ | √ | √ | √ |
| Peripheral ID4 | — | R | H'EA0F_0FD0 | H'800F_0FD0 | H'0000_0004 | 32 | √ | √ | √ | √ |
| Peripheral ID0 | — | R | H'EA0F_0FE0 | H'800F_0FE0 | H'0000_00AA | 32 | √ | √ | √ | √ |
| Peripheral ID1 | — | R | H'EA0F_0FE4 | H'800F_0FE4 | H'0000_003F | 32 | √ | √ | √ | √ |
| Peripheral ID2 | — | R | H'EA0F_0FE8 | H'800F_0FE8 | H'0000_000A | 32 | √ | √ | √ | √ |
| Peripheral ID3 | — | R | H'EA0F_0FEC | H'800F_0FEC | H'0000_0000 | 32 | √ | √ | √ | √ |
| Component ID0 | — | R | H'EA0F_0FF0 | H'800F_0FF0 | H'0000_000D | 32 | √ | √ | √ | √ |
| Component ID1 | — | R | H'EA0F_0FF4 | H'800F_0FF4 | H'0000_00F0 | 32 | √ | √ | √ | √ |
| Component ID2 | — | R | H'EA0F_0FF8 | H'800F_0FF8 | H'0000_0005 | 32 | √ | √ | √ | √ |
| Component ID3 | — | R | H'EA0F_0FFC | H'800F_0FFC | H'0000_00B1 | 32 | √ | √ | √ | √ |

72.1.4.3 TDBG registers configurations

TDBG is the collection of trace pin control registers and debugging status registers for chip-specific debug features. Only after write operation to the KEY bit in DBGREG9 is done, it is possible to write to DBGREG1. Note that asserting KEY bit in DBGREG9 register requires two phases of writing as describing in section 72.2.3.3.

Table 72.9 TDBG register configurations

| Name of Register | Abbreviation | R/W | Address (CPU/ Debugger View) | Initial Value | Access Size | Second Generation RZ/G Series Products | | | |
|------------------|--------------|-----|---------------------------------------|---------------|----------------|---|----------------------------|--------|--------|
| | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Debug Register 1 | DBGREG1 | R/W | H'E610_0020 | H'x00x_0000 * | 32 | √ | √ | √ | √ |
| Debug Register 4 | DBGREG4 | R | H'E610_002C | H'0000_0xxx * | 32 | √ | √ | √ | √ |
| Debug Register 9 | DBGREG9 | R/W | H'E610_0040 | H'0000_0000 | 32 | √ | √ | √ | √ |

Note: * The initial value is determined by input of LSI pin.

72.1.5 Connected module

Table 72.10 Connected module

| Module name | Connected module name | Function of connected module | Second Generation RZ/G Series Products | | | |
|-------------|--------------------------|---------------------------------|---|----------------------------|--------|--------|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| CoreSight | AP-System Core | Processor | √ | √ | √ | √ |
| | CPG | Clock Pulse Generator | √ | √ | √ | √ |
| | PFC | Pin function | √ | √ | √ | √ |

72.1.6 Part number and TARGETID

Part number is a product identification. This number is stored in Peripheral Identification Registers of ROM Table and TARGETID Register. Debugger reads those registers to identify the product. Each product has unique value of part number as shown in Table 72.11:

Note: Refer to Arm CoreSight Architecture Specification v2.0 for more detail information.

Table 72.11 Part number and TARGETID

| Second Generation of RZ/G Series Products | Part number | TARGETID |
|--|--------------------|-----------------|
| RZ/G2H | H'008 | H'0008_0447 |
| RZ/G2M V1.3, RZ/G2M V3.0 | H'00D | H'000D_0447 |
| RZ/G2N | H'021 | H'0021_0447 |
| RZ/G2E | H'026 | H'0026_0447 |

72.2 Register Description

72.2.1 CoreSight and Debug registers

Refer to following manuals for generic CoreSight components.

- Arm Architecture Reference Manual Armv8-A
- Arm Architecture Reference Manual Armv7-A and Armv7-R edition
- Arm Cortex-A53 MPCore Processor Technical Reference Manual
- Arm Cortex-A57 MPCore Processor Technical Reference Manual
- Arm CoreSight SoC-400 Technical Reference Manual
- Arm CoreSight Trace Memory Controller Technical Reference Manual
- Arm CoreSight STM-500 System Trace Macrocell Technical Reference Manual

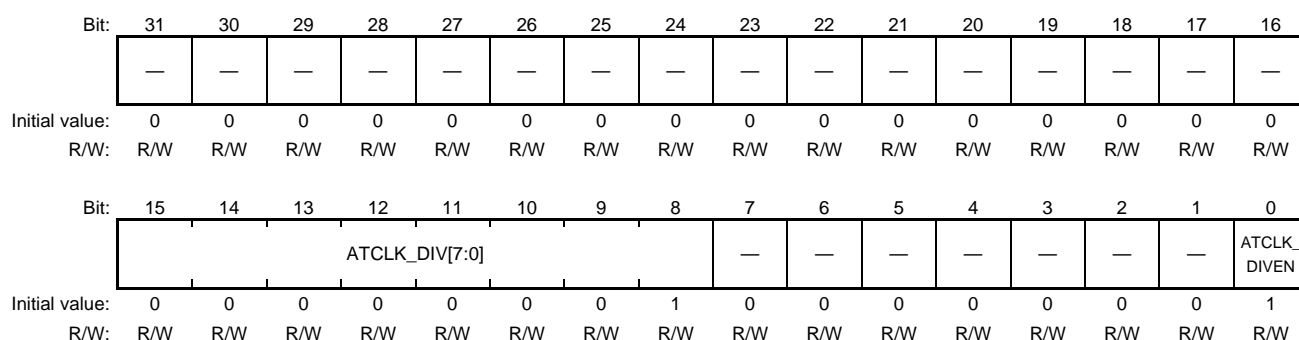
72.2.2 CSreg registers

CSreg is the collection of control registers for chip-specific debug features. These registers can be accessed from both Debugger and CPU via Debug-APB.

72.2.2.1 ATCLKCR_CA57 Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | — |

Address CPU view: H' EA0F_0100 Debugger view: H' 800F_0100



| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------------|---------------|-----|---|
| 31 to 16 | Reserved | All 0 | R/W | Reserved The write value should be the same one that has been read immediately before writing. |
| 15 to 8 | ATCLK_DIV[7:0] | H'01 | R/W | Control the clock ratio CLK:ATCLK in Cortex-A57. The clock ratio is specified by ATCLK_DIV + 1. |
| 7 to 1 | Reserved | All 0 | R/W | Reserved The write value should be the same one that has been read immediately before writing. |
| 0 | ATCLK_DIVEN | B'1 | R/W | Enable ATCLK_DIV: 0: ATCLK_DIV is invalid. The clock ratio CLK:ATCLK becomes 1. 1: ATCLK_DIV is valid. |

72.2.2.2 CSREG_E00 Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| — | — | — | √ |

Address CPU view: H' EA0F_0E00 Debugger view: H' 800F_0E00

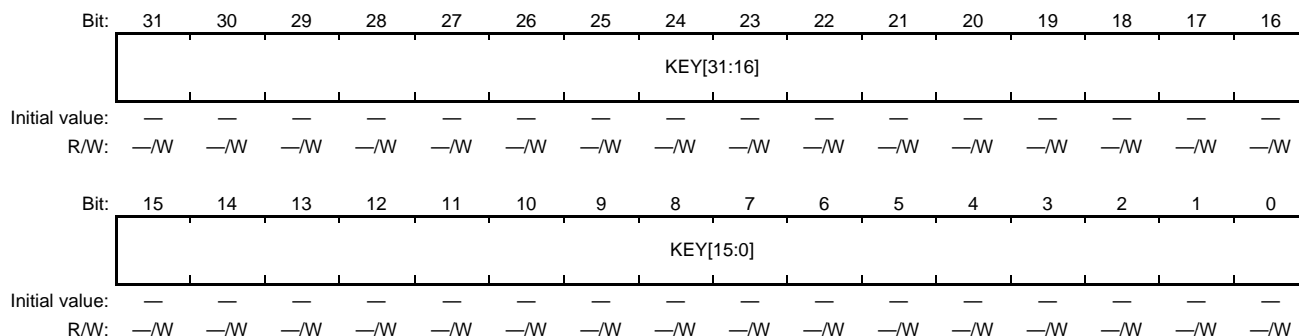
| | | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---------------------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | STM0 FIXED AWBURST enable |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------------|---------------|-----|---|
| 31 to 1 | — | All 0 | R/W | Reserved The write value should be the same one that has been read immediately before writing. |
| 0 | STM0 FIXED AWBURST enable | B'0 | R/W | Override AWBURST at input of STM0 to FIXED type. 0: disable 1: enable |

72.2.2.3 LOCKACCESS Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address CPU view: H' EA0F_0FB0 Debugger view: H' 800F_0FB0

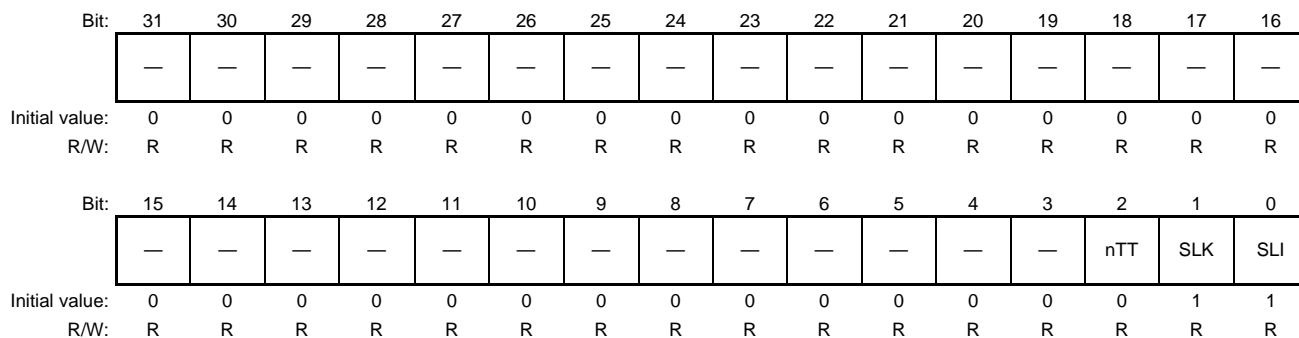


| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|--|
| 31 to 0 | KEY[31:0] | — | —/W | Lock Access Register. See CoreSight Architecture Specification |

72.2.2.4 LOCKSTATUS Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address CPU view: H' EA0F_0FB4 Debugger view: H' 800F_0FB4



| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|--|
| 31 to 3 | — | All 0 | R | Reserved |
| 2 | nTT | B'0 | R | Lock Status Register. See CoreSight Architecture Specification |
| 1 | SLK | B'1 | R | |
| 0 | SLI | B'1 | R | |

72.2.2.5 Peripheral ID4 Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address CPU view: H' EA0F_0FD0 Debugger view: H' 800F_0FD0

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----------------|----|----|----|-------------------------------|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | 4KB count[3:0] | | | | JEP106 continuation code[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|--------------------------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved |
| 7 to 4 | 4KB count | H'0 | R | Indicates that the device only occupies 4KB of memory |
| 3 to 0 | JEP106 continuation code | H'4 | R | JEDEC code |

72.2.2.6 Peripheral ID0 Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address CPU view: H' EA0F_0FE0 Debugger view: H' 800F_0FE0

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | Part No.0[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------|---------------|-----|------------------------------|
| 31 to 8 | — | All 0 | R | Reserved |
| 7 to 0 | Part No.0 | H'AA | R | part number of the component |

72.2.2.7 Peripheral ID1 Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address CPU view: H' EA0F_0FE4 Debugger view: H' 800F_0FE4

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------------------|----|----|----------------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | JEP106 ID code[3:0] | | | Part No.1[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------|---------------|-----|------------------------------|
| 31 to 8 | — | All 0 | R | Reserved |
| 7 to 4 | JEP106 ID code[3:0] | H'3 | R | JEDEC code |
| 3 to 0 | Part No.1 | H'F | R | part number of the component |

72.2.2.8 Peripheral ID2 Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address CPU view: H' EA0F_0FE8 Debugger view: H' 800F_0FE8

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----------|----|----|----|---------------------|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | Rev[3:0] | | | — | JEP106 ID code[6:4] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|---------------------|---------------|-----|-------------|
| 31 to 8 | — | All 0 | R | Reserved |
| 7 to 4 | Rev | H'0 | R | Revision |
| 3 | — | B'1 | R | Reserved |
| 2 to 0 | JEP106 ID code[6:4] | B'010 | R | JEDEC code |

72.2.2.9 Peripheral ID3 Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address CPU view: H' EA0F_0FEC Debugger view: H' 800F_0FEC

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|-------------|----|----|-----------------------|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | RevAnd[3:0] | | | CustomerModified[3:0] | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|------------------|---------------|-----|---|
| 31 to 8 | — | All 0 | R | Reserved |
| 7 to 4 | RevAnd | H'0 | R | Indicates that there are no errata fixes to this component |
| 3 to 0 | CustomerModified | H'0 | R | Indicates that the customer has not modified this component |

72.2.2.10 Component ID0 Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address CPU view: H' EA0F_0FF0 Debugger view: H' 800F_0FF0

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | Preamble[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|-------------------------------|
| 31 to 8 | — | All 0 | R | Reserved |
| 7 to 0 | Preamble | H'0D | R | component identification code |

72.2.2.11 Component ID1 Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address CPU view: H' EA0F_0FF4 Debugger view: H' 800F_0FF4

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----------------------|----|----|----|---------------|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | Component class[3:0] | | | | Preamble[3:0] | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|-----------------|---------------|-----|-------------------------------|
| 31 to 8 | — | All 0 | R | Reserved |
| 7 to 4 | Component class | H'F | R | Class of the component |
| 3 to 0 | Preamble | H'0 | R | component identification code |

72.2.2.12 Component ID2 Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address CPU view: H' EA0F_0FF8 Debugger view: H' 800F_0FF8

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------------|----|----|----|----|----|----|----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | — | — | — | Preamble[7:0] | | | | | | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|-------------------------------|
| 31 to 8 | — | All 0 | R | Reserved |
| 7 to 0 | Preamble | H'05 | R | component identification code |

72.2.2.13 Component ID3 Register

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

Address CPU view: H' EA0F_0FFC Debugger view: H' 800F_0FFC

| | | | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|---------------|----|----|----|----|----|----|----|---|---|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | | |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | | |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
| | — | — | — | — | — | — | — | — | Preamble[7:0] | | | | | | | | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | | |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|-------------------------------|
| 31 to 8 | — | All 0 | R | Reserved |
| 7 to 0 | Preamble | H'B1 | R | component identification code |

72.2.3 TDBG registers

72.2.3.1 Debug register 1 (DBGREG1)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register is intended to reflect the values entered into external pins. The values for external pins are set when initialization is performed.

| | | | | | | | | | | | | | | | | |
|----------------|-----------|-----------|-----------|-----------|---------|---------|-----|-----|-----|-----|-------------|-----|-----|-----|-----------|-----------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | MD11 | MD10 | MD21 | MD20 | — | — | — | — | — | — | STMSEL[1:0] | — | — | — | — | MDT[1:0] |
| Initial value: | Pin Input | Pin Input | Pin Input | Pin Input | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Pin Input | Pin Input |
| R/W: | R | R | R | R | R | R | R | R | R | R/W | R/W | R/W | R | R | R/W | R/W |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | TRC[2:0] | | | — | DMON2EN | DMON1EN | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|---|
| 31 | MD11 | Pin Input | R | The value of the MD11 pin input is reflected in the initial value. |
| 30 | MD10 | Pin Input | R | The value of the MD10 pin input is reflected in the initial value. |
| 29, 28 | MD21, MD20 | Pin Input | R | The value of MD21 and MD20 pin input is reflected in the initial value. |
| 27 to 23 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 22 | — | B'0 | R/W | Reserved. The write value should be 0. |
| 21 | STMSEL[1] | B'0 | R/W | Select STM PORT2 0: Normal function pins 1: Select STM PORT2 (STMCLK_2, STMDATAx_2 (x: 0 to 3)) |
| 20 | STMSEL[0] | B'0 | R/W | Select STM PORT1 0: Normal function pins 1: Select STM PORT1 (STMCLK_1, STMDATAx_1 (x: 0 to 3)) |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 19, 18 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 17, 16 | MDT[1:0] | Pin Input | R/W | The value of the MDT1 and MDT0 pin input is reflected in the initial value. The write value must be read value. |
| 15 to 13 | TRC[2:0] | B'000 | R/W | Selects the trace function. B'000: Normal (no select) B'001: Cortex-A57 trace data (RZ/G2E: not support). B'010: CoreSight(SYS) trace data B'011: Setting prohibited B'100: Setting prohibited B'101: Cortex-A53 trace data (RZ/G2N: not support). B'110: Setting prohibited B'111: Setting prohibited |
| 12 | — | B'0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 11, 10 | DMON2EN | B'00 | R/W | (RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N) Bit11 0: Disable PCIe ch1 trace output 1: Enable PCIe ch1 trace output Bit10 0: Disable PCIe ch0 trace output 1: Enable PCIe ch0 trace output (RZ/G2E) Reserved These bits are always read as 0. The write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 9, 8 | DMON1EN | B'00 | R/W | Bit9 – TMONDT19 to TMONDT12 enable Bit8 – TMONDT11 to TMONDT0 enable 0: Enable other function depending on the PFC setting 1: Enable TMONDTn Pins |
| 7 to 0 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |

72.2.3.2 Debug register 4 (DBGREG4)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This is a status register showing debug features are enable or disable. It is possible to perform read operations only from the CPU.

| | | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|--------------|-----------------|----------------|---------------|---------------|----|----|-----------------|----------------|---------------|---------------|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| | | | | | | | | | | | | | | | | |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | — | — | — | — | — | CS_D EVEN | CS_ SPNI DEN | CS_ SPID EN | CS_ NI DEN | CS_ D BGEN | — | — | CA_ SPNI DEN | CA_ SPID EN | CA_ NI DEN | CA_ D BGEN |
| Initial value: | 0 | 0 | 0 | 0 | * | * | * | * | * | * | * | * | * | * | * | * |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|------------|---------------|-----|--|
| 31 to 12 | — | All 0 | R | Reserved These bits are always read as 0. |
| 11 | Reserved | * | R | Reserved |
| 10 | CS_DEVEN | * | R | Indicates disable or enable of the DAP_DEVICEEN of CoreSight. 0: Disable 1: Enable |
| 9 | CS_SPNIDEN | * | R | Indicates disable or enable of the SPNIDEN of CoreSight. 0: Disable 1: Enable |
| 8 | CS_SPIDEN | * | R | Indicates disable or enable of the SPIDEN of CoreSight. 0: Disable 1: Enable |
| 7 | CS_NIDEN | * | R | Indicates disable or enable of the NIDEN of CoreSight. 0: Disable 1: Enable |
| 6 | CS_DBGEN | * | R | Indicates disable or enable of the DBGEN of CoreSight. 0: Disable 1: Enable |
| 5 | CR_NIDEN | * | R | Reserved |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|------------|---------------|-----|---|
| 4 | CR_DBGGEN | * | R | Reserved |
| 3 | CA_SPNIDEN | * | R | Indicates disable or enable of the SPNIDEN of Cortex-A57/ Cortex-A53. 0: Disable 1: Enable (RZ/G2E: only support Cortex-A53) (RZ/G2N: only support Cortex-A57) |
| 2 | CA_SPIDEN | * | R | Indicates disable or enable of the SPIDEN of Cortex-A57/ Cortex-A53. 0: Disable 1: Enable (RZ/G2E: only support Cortex-A53) (RZ/G2N: only support Cortex-A57) |
| 1 | CA_NIDEN | * | R | Indicates disable or enable of the NIDEN of Cortex-A57/ Cortex-A53. 0: Disable 1: Enable (RZ/G2E: only support Cortex-A53) (RZ/G2N: only support Cortex-A57) |
| 0 | CA_DBGGEN | * | R | Indicates disable or enable of the DBGGEN of Cortex-A57/ Cortex-A53. 0: Disable 1: Enable (RZ/G2E: only support Cortex-A53) (RZ/G2N: only support Cortex-A57) |

Note: * Depends on the setting of the MD21, MD20, MD11, MD10, MDT[1:0] pins.

72.2.3.3 Debug register 9 (DBGREG9)

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |

This register has a key role in preventing invalid access. It is possible to perform read/write operations only from the CPU. DBGREG1 can only be written when the KEY bit in DBGREG9 is updated as below procedure.

First, write H'A500 to DBGREG9[15:0]. This enables to write to DBGREG9[0]. Then, write H'A501 to DBGREG9[15:0]. After this writing is done, DBGREG1 can be written.

It is recommended to clear the KEY bit in DBGREG9 after finishing writing to DBGREG1. To clear KEY bit in DBGREG9, write H'0000_0000 to DBGREG9.

| | | | | | | | | | | | | | | | | |
|----------------|----------|-----|-----|-----|-----|-----|-----|-----|----|----|----|----|----|----|----|-----|
| Bit: | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — | — |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R | R |
| Bit: | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | AID[7:0] | | | | | | | | — | — | — | — | — | — | — | KEY |
| Initial value: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R/W: | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R | R | R | R | R | R | R | R/W |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---------------|-----|--|
| 31 to 16 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 15 to 8 | AID[7:0] | H'00 | R/W | KEY Bit Write Aid Use this bit to write a bit pattern (H'A5) to be written to the KEY bit. |
| 7 to 1 | — | All 0 | R | Reserved These bits are always read as 0. The write value should always be 0. |
| 0 | KEY | B'0 | R/W | DBGREG1 access control Provides access control to the DBGREG1. 0: It is not possible to perform write operations to the DBGREG1. 1: It is possible to perform write operations to the DBGREG1. When writing 1 to this bit, be sure to write H'A5 to the AID bits at the same time. |

72.3 Operation

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

72.3.1 JTAG selection

JTAG function can be selected through MD pins setting below.

If CoreSight is selected through MD pins, internal debug enable signals for Cortex-A57/Cortex-A53 are asserted automatically.

Table 72.12 JTAG selection table for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N

| MD pin | | | | |
|--------|---------|------|----------|-----------|
| MD10 | MD21,20 | MD11 | MDT[1:0] | JTAG |
| B'0 | B'00 | * | * | — |
| | B'10 | B'0 | * | CoreSight |

*: don't care, —: No connection with TAP controller

Note : Other than above, setting prohibited.

Table 72.13 JTAG selection table for RZ/G2E

| MD pin | | | | |
|--------|---------|------|----------|-----------|
| MD10 | MD21,20 | MD11 | MDT[1:0] | JTAG |
| B'0 | B'00 | * | * | — |
| | B'10 | B'0 | * | CoreSight |

*: don't care, —: No connection with TAP controller
 Note : Other than above, setting prohibited.

72.3.2 Trace output selection

Trace Data source can be selected by DBGREG1.TRC[2:0] setting. Cortex-A57, CoreSight, and Cortex-A53 can be selected.

DBGREG1.DMONIEN[1:0] can select pin function of trace data bit width. The trace clock is ZTR ϕ .

72.3.3 STM output selection

DBGREG1.STMSEL bits can select STM Port output. System Trace clock is ZTR ϕ .

72.3.4 STM Master ID mapping

STM-500 is used for system trace to output some information from CPU and monitor the contents stored in memories such as DDR memories and internal SRAMs (e.g. System RAM) by utilizing DMA transfer. Following table lists of Master ID value (output of STM) to identify the master module sending the transaction to STM-500 in each product.

Table 72.14 STM Master ID mapping

—: no use

| Master name | STM number | Secure Access | Master_ID | Second Generation RZ/G Series Products | | | |
|-------------------|------------|---------------|-------------|---|----------------------------|--------|--------|
| | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Cortex-A57 Core#0 | STM0 | Secure | H'20 | √ | √ | √ | — |
| Cortex-A57 Core#1 | STM0 | Secure | H'21 | √ | √ | √ | — |
| Cortex-A57 Core#2 | STM0 | Secure | H'22 | √ | — | — | — |
| Cortex-A57 Core#3 | STM0 | Secure | H'23 | √ | — | — | — |
| Cortex-A53 Core#0 | STM0 | Secure | H'24 | √ | √ | — | √ |
| Cortex-A53 Core#1 | STM0 | Secure | H'25 | √ | √ | — | √ |
| Cortex-A53 Core#2 | STM0 | Secure | H'26 | √ | √ | — | — |
| Cortex-A53 Core#3 | STM0 | Secure | H'27 | √ | √ | — | — |
| SYS-DMAC * | STM0 | Secure | H'30 – H'3F | √ | √ | √ | √ |
| Cortex-A57 Core#0 | STM0 | Non-secure | H'60 | √ | √ | √ | — |
| Cortex-A57 Core#1 | STM0 | Non-secure | H'61 | √ | √ | √ | — |
| Cortex-A57 Core#2 | STM0 | Non-secure | H'62 | √ | — | — | — |
| Cortex-A57 Core#3 | STM0 | Non-secure | H'63 | √ | — | — | — |
| Cortex-A53 Core#0 | STM0 | Non-secure | H'64 | √ | √ | — | √ |
| Cortex-A53 Core#1 | STM0 | Non-secure | H'65 | √ | √ | — | √ |
| Cortex-A53 Core#2 | STM0 | Non-secure | H'66 | √ | √ | — | — |
| Cortex-A53 Core#3 | STM0 | Non-secure | H'67 | √ | √ | — | — |
| SYS-DMAC * | STM0 | Non-secure | H'70 – H'7F | √ | √ | √ | √ |

Note: * SYS-DMAC are not distinguishable. Either one DMA Controller is expected to use at the same time.

72.3.5 STM AXI Slave Interface

Each address in stimulus port of STM is corresponding to the types of trace packet to be generated; therefore, for choosing the type of trace packet, issue AXI transactions with the corresponding addresses to STM.

STM has a 64-bit (8-byte) AXI write slave interface which can accept a single 8-byte write. Sending more than 8-byte in a burst access to STM is prohibited. Otherwise, the AXI addresses are incremental and unintended types of trace packets are generated.

72.3.5.1 STM FIXED AWBURST [RZ/G2E]

By setting 1 to "FIXED AWBURST enable" bit in CSREG_E00 register, RZ/G2E can send more than 8-byte to STM in a burst access. In that case, the AXI addresses become the same for every transfer in the burst access by overriding the AXI burst type to the "FIXED" type.

72.3.6 Cross Trigger Connection

Cross trigger connections shows below.

Table 72.15 CSD-CTI0 Connections

| IF | Port | Wire | Second Generation RZ/G Series Products | | | | | |
|------------|------|-----------------|---|----------------------------|--------|--------|---|---|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | | |
| Trigger IF | | in | inack | | | | | |
| | in | 0 | ETF FULL | — | √ | √ | √ | √ |
| | | 1 | ETF ACQCOMP | — | √ | √ | √ | √ |
| | | 2 | — | — | — | — | — | — |
| | | 3 | — | — | — | — | — | — |
| | | 4 | STM TRIGOUTSPTE | — | √ | √ | √ | √ |
| | | 5 | STM TRIGOUTSW | — | √ | √ | √ | √ |
| | | 6 | STM TRIGOUTHETE | — | √ | √ | √ | √ |
| | | 7 | STM ASYNCOUT | — | √ | √ | √ | √ |
| | | Out | outack | | | | | |
| | out | 0 | ETF FLUSHIN | — | √ | √ | √ | √ |
| | | 1 | ETF TRIGIN | — | √ | √ | √ | √ |
| | | 2 | TPIU FLUSHIN | TPIU FLUSHINACK | √ | √ | √ | √ |
| | | 3 | TPIU TRIGIN | TPIU TRIGINACK | √ | √ | √ | √ |
| | | 4 | — | — | — | — | — | — |
| 5 | | — | — | — | — | — | — | |
| 6 | | — | — | — | — | — | — | |
| 7 | | CSWAKEUP wakeup | — | √ | √ | √ | √ | |

—: no use

Table 72.16 CSD-CTH Connections

| | | | Second Generation RZ/G Series Products | | | | | |
|------------|------|-------------------------------|---|---------------|----------------------------|--------|--------|---|
| IF | Port | Wire | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| Trigger IF | | in | inack | | | | | |
| | in | 0 | — | — | — | — | — | |
| | | 1 | — | — | — | — | — | |
| | | 2 | mtirt overflow error | — | — | — | — | |
| | | 3 | — | — | — | — | — | |
| | | 4 | Reserved | support | √ | √ | √ | √ |
| | | 5 | Reserved | support | — | — | — | — |
| | | 6 | Reserved | support | — | — | — | — |
| | | 7 | External signal from CTIACK pin | — | √ | √ | √ | √ |
| | | | Out | outack | | | | |
| | out | 0 | — | — | — | — | — | — |
| | | 1 | — | — | — | — | — | — |
| | | 2 | — | — | — | — | — | — |
| | | 3 | — | — | — | — | — | — |
| | | 4 | Reserved | — | √ | √ | √ | √ |
| | | 5 | Reserved | support | — | — | — | — |
| | 6 | Reserved | support | — | — | — | — | |
| | 7 | External signal to CTIREQ pin | — | √ | √ | √ | √ | |

—: no use

Table 72.17 CSD-CTI2 Connections

| | | | | Second Generation RZ/G Series Products | | | | | |
|------------|------|-----------|------------------|---|-------------|-------------|--------|--------|---|
| IF | Port | Wire | | RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 | RZ/G2N | RZ/G2E | |
| | | | | | | | | | |
| Trigger IF | | in | inack | | | | | | |
| | i | 0 | STM2 TRIGOUTSPTE | — | — | — | — | — | |
| | | 1 | STM2 TRIGOUTSW | — | — | — | — | — | |
| | | 2 | STM2 TRIGOUTHETE | — | — | — | — | — | |
| | | 3 | STM2 ASYNCOU | — | — | — | — | — | |
| | | 4 | STM1 TRIGOUTSPTE | — | — | — | — | — | |
| | | 5 | STM1 TRIGOUTSW | — | — | — | — | — | |
| | | 6 | STM1 TRIGOUTHETE | — | — | — | — | — | |
| | | 7 | STM1 ASYNCOU | — | — | — | — | — | |
| | | | out | outack | | | | | |
| | | o | 0 | — | — | — | — | — | — |
| | | | 1 | — | — | — | — | — | — |
| | | | 2 | — | — | — | — | — | — |
| | | | 3 | — | — | — | — | — | — |
| | | | 4 | — | — | — | — | — | — |
| | | 5 | — | — | — | — | — | — | |
| | | 6 | — | — | — | — | — | — | |
| | | 7 | — | — | — | — | — | — | |

—: no use

Table 72.18 A57CA-CTH1 Connections

| IF | Port | Wire | Second Generation RZ/G Series Products | | | | | |
|------------|------|------------|---|----------------------------|--------|--------|---|---|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | | |
| Trigger IF | in | in | inack | | | | | |
| | | 0 | — | — | √ | √ | √ | — |
| | | 1 | — | — | √ | √ | √ | — |
| | | 2 | ETF FULL | — | √ | √ | √ | — |
| | | 3 | ETF ACQCOMP | — | √ | √ | √ | — |
| | | 4 | — | — | √ | √ | √ | — |
| | | 5 | — | — | √ | √ | √ | — |
| | | 6 | — | — | √ | √ | √ | — |
| | 7 | — | — | √ | √ | √ | — | |
| | out | Out | outack | | | | | |
| | | 0 | ETF FLUSHIN | — | √ | √ | √ | — |
| | | 1 | ETF TRIGIN | — | √ | √ | √ | — |
| | | 2 | TPIU FLUSHIN | TPIU FLUSHINACK | √ | √ | √ | — |
| | | 3 | TPIU TRIGIN | TPIU TRIGINACK | √ | √ | √ | — |
| | | 4 | — | — | √ | √ | √ | — |
| 5 | | — | — | √ | √ | √ | — | |
| 6 | | — | — | √ | √ | √ | — | |
| 7 | — | — | √ | √ | √ | — | | |

—: no use

Table 72.19 A53CA-CTH1 Connections

| IF | Port | Wire | Second Generation RZ/G Series Products | | | | | |
|---------------|------|------------|---|----------------------------|--------|--------|---|---|
| | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E | | |
| Trigger IF | in | in | Inack | | | | | |
| | | 0 | — | — | √ | √ | — | √ |
| | | 1 | — | — | √ | √ | — | √ |
| | | 2 | ETF FULL | — | √ | √ | — | √ |
| | | 3 | ETF ACQCOMP | — | √ | √ | — | √ |
| | | 4 | — | — | √ | √ | — | √ |
| | | 5 | — | — | √ | √ | — | √ |
| | | 6 | — | — | √ | √ | — | √ |
| | 7 | — | — | √ | √ | — | √ | |
| | out | Out | outack | | | | | |
| | | 0 | ETF FLUSHIN | — | √ | √ | — | √ |
| | | 1 | ETF TRIGIN | — | √ | √ | — | √ |
| | | 2 | TPIU FLUSHIN | TPIU FLUSHINACK | √ | √ | — | √ |
| | | 3 | TPIU TRIGIN | TPIU TRIGINACK | √ | √ | — | √ |
| | | 4 | — | — | √ | √ | — | √ |
| 5 | | — | — | √ | √ | — | √ | |
| 6 | | — | — | √ | √ | — | √ | |
| 7 | — | — | √ | √ | — | √ | | |

—: no use

72.3.7 ATB Funnel information

CSDBGSS ATB Funnel Port connection for each product is as follows:

Table 72.20 ATB Funnel Port connection

| Module | Connection | | Second Generation RZ/G Series Products | | | |
|----------------|-----------------|--------------------------|---|----------------------------|--------|--------|
| | ATB Funnel Port | Abbreviation | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| CSDBGSS | Port0 | A57 | √ | √ | √ | — |
| | Port1 | A53 | √ | √ | — | √ |
| | Port2 | — | — | — | — | — |
| | Port3 | CSD-STM0 | √ | √ | √ | √ |
| | Port4 | CSD-STM1 | — | — | — | — |
| | Port5 | CSD-STM2 | — | — | — | — |
| CA57SS CSDBGCA | Port0 | CPU0 | √ | √ | √ | — |
| | Port1 | CPU1 | √ | √ | √ | — |
| | Port2 | CPU2 | √ | — | — | — |
| | Port3 | CPU3 | √ | — | — | — |
| CA53SS CSDBGCA | Port0 | CPU0 | √ | √ | — | √ |
| | Port1 | CPU1 | √ | √ | — | √ |
| | Port2 | CPU2 | √ | √ | — | — |
| | Port3 | CPU3 | √ | √ | — | — |
| CR7SS CSDBGCA | Port0 | Instruction trace stream | √ | √ | √ | √ |
| | Port1 | — | — | — | — | — |
| | Port2 | — | — | — | — | — |
| | Port3 | — | — | — | — | — |
| | Port4 | Data trace stream | √ | √ | √ | √ |

72.4 Usage Notes

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

72.4.1 EDPRSR.PU polling

For Cortex-A53 and Cortex-A57, all register settings in Core Power Domain need to be done after EDPRSR.PU in Debug Power Domain becomes one.

Figure 72.5 is an example for Cortex-A53 with 2 CPUs. The number of CPU is dependent on product.

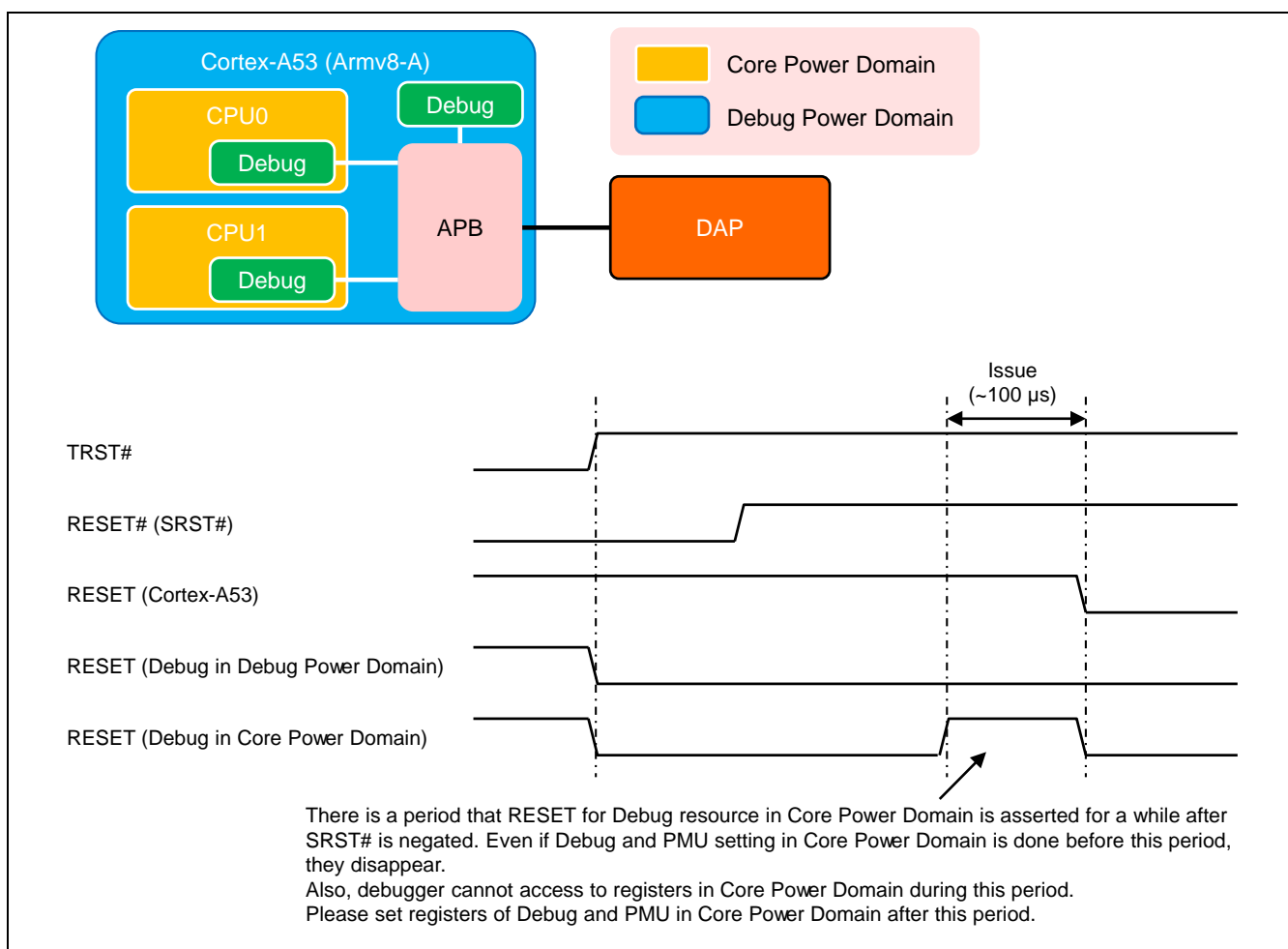


Figure 72.5 Reset timing in Core Power domain

72.4.2 TRST control in no JTAG selection

If the JTAG Ports are not selected by debug related MD pins (MD[11:10], MD[21:20], MDT[1:0] pins are all low level), TRST# pin of JTAG should be set to low level or driven with the same level of PRESET# pin.

72.4.3 Pseudo Power down

Pseudo Power down is not supported at debugging.

72.4.4 Important notice for soft power-on reset

In debug mode*¹, soft power-on reset (WDT reset or SRESCR.SPRES = 1) cannot be issued. Do not change the following register settings.

- RWDT_RSTMSK bit (bit 0 in the WDTRSTCR register): Initial value 1.
- SWDT_RSTMSK bit (bit 1 in the WDTRSTCR register): Initial value 1.
- SPRES bit (bit 15 in the SRESCR register): Initial value 0.

Note: 1. LSI is in debug mode when read value of MD21 bit (bit 29 in the DBGREG1 register) is 1.

73. Electrical Characteristics

73.1 Absolute Maximum Ratings

Table 73.1.1 Absolute Maximum Ratings for [RZ/G2H] [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N]

| Item | Value | Unit | Remarks | RZ/G2H | |
|---|------------------------------|------|---------|-------------|-------------|
| | | | | RZ/G2M V1.3 | RZ/G2M V3.0 |
| | | | | RZ/G2N | RZ/G2E |
| Power supply voltage (3.3 V) (VDDQ33, VDDQVA_SDn (n = 0 to 3) (except during operation through SDHI (SDR50, SDR104)), VDD33_SATA, VDD33_PCIEn (n = 0, 1), VDDQ33_USB3HSHn (n = 0, 1), VDDQ33_USB3HSn (n = 0, 1), VDD33_USB3n (n = 0, 1), VDDQ33_USB2Hn (n = 0, 1, 2), VDDQ33_USB2) | -0.3 to +3.84 | V | | | |
| Power supply voltage (2.5 V) (VDDQ25_ETH) | -0.3 to +2.9 | V | | | |
| Power supply voltage (1.8 V) (VDDQ18, VDD18_CPGPLLn (n = 0 to 4), VDDQ18_HDMIIn (n = 0, 1), VDDQ18_LVDS, VDD18_LVDSPLL1, VDDQ18_CSIn (n = 0 to 3), VDD18_MLBPLL, VDD18_DUPLLn (n = 0, 1), VDDQVA_SDn (n = 0 to 3) (SDR50, SDR104)) | -0.3 to +2.34 | V | | | |
| Power supply voltage (1.1 V) (VDDQVA_DDRn (n = 0, 1)) | -0.3 to +1.5 | V | LPDDR4 | | |
| Power supply voltage (0.9 V) (VDD09_DDRPLLn (n = 0, 1), VDD09_SATA, VDD09_PCIEn (n = 0, 1), VDD09_USB3n (n = 0, 1), VDD09_USB3HSn (n = 0, 1), VDD09_USB2n (n = 0, 1, 2), VDDQ09_HDMIIn (n = 0, 1), VDDQ09_LVDS, VDD09_CSIn (n = 0 to 3)) VDD09_LVDSPLL2 | -0.3 to + 1.12 | V | | | |
| Power supply voltage (0.9 V) (VDD) | -0.3 to + 1.12 | V | | | |
| Power supply voltage (0.9 V) (VDD_DVFS) | -0.3 to + 1.12 | V | | | |
| Input voltage (3.3-V I/O) | -0.3 to VDDQ33 + 0.3 | V | *1 | | |
| Input voltage (3.3-V I/O [SDHI]) | -0.3 to VDDQVA_SDn + 0.3 | V | *1 | | |
| Input voltage (3.3-V I/O [SATA]) | -0.3 to VDD33_SATA + 0.3 | V | *1 | | |
| Input voltage (3.3-V I/O [PCIE]) | -0.3 to VDD33_PCIEn + 0.3 | V | *1 | | |
| Input voltage (3.3-V I/O [USB3]) | -0.3 to VDD33_USB3n + 0.3 | V | *1 | | |
| Input voltage (3.3-V I/O [USB3HSH]) | -0.3 to VDD33_USB3HSHn + 0.3 | V | *1 | | |
| Input voltage (3.3-V I/O [USB3HS]) | -0.3 to VDD33_USB3HSn + 0.3 | V | *1 | | |

| Item | Value | Unit | Remarks |
|--|------------------------------|------|-------------------|
| Input voltage (3.3-V I/O [USB2H]) | -0.3 to VDDQ33_USB2Hn + 0.3 | V | *1 |
| Input voltage (3.3-V I/O [USB2]) | -0.3 to VDDQ33_USB2 + 0.3 | V | *1 |
| Input voltage (2.5-V I/O [ETH]) | -0.3 to VDDQ25_ETH + 0.3 | V | *2 |
| Input voltage (1.8-V I/O) | -0.3 to VDDQ18 + 0.3 | V | *3 |
| Input voltage (1.8-V I/O [HDMI]) | -0.3 to VDDQ18_HDMIIn + 0.3 | V | *3 |
| Input voltage (1.8-V I/O [VDDQ18_LVDS]) | -0.3 to VDDQ18_LVDS + 0.3 | V | *3 |
| Input voltage (1.8-V I/O [VDDQ18_CSIn]) | -0.3 to VDDQ18_CSIn + 0.3 | V | *3 |
| Output voltage (1.8-V I/O [VDDQ18_PCIEIn]) | -0.3 to VDDQ18_PCIEIn + 0.3 | V | *3 |
| Input voltage (1.8-V I/O [VDDQVA_SDn]) | -0.3 to VDDQVA_SDn + 0.3 | V | *3, SDR50, SDR104 |
| Input voltage (1.1-V I/O [VDDQVA_DDRn]) | -0.3 to VDDQVA_DDRn + 0.3 | V | *4, LPDDR4 |
| Input voltage (0.9-V I/O [VDD09_SATA]) | -0.3 to VDD09_SATA + 0.3 | V | *5 |
| Input voltage (0.9-V I/O [VDD09_PCIEIn]) | -0.3 to VDD09_PCIEIn + 0.3 | V | *5 |
| Input voltage (0.9-V I/O [VDD09_USB3n]) | -0.3 to VDD09_USB3n + 0.3 | V | *5 |
| Input voltage (0.9-V I/O [VDD09_USB3HSn]) | -0.3 to VDD09_USB3HSn + 0.3 | V | *5 |
| Input voltage (0.9-V I/O [VDD09_USB2n]) | -0.3 to VDD09_USB2n + 0.3 | V | *5 |
| Input voltage (0.9-V I/O [VDDQ09_HDMIIn]) | -0.3 to VDDQ09_HDMIIn + 0.3 | V | *5 |
| Input voltage (0.9-V I/O [VDDQ09_LVDS]) | -0.3 to VDDQ09_LVDS + 0.3 | V | *5 |
| Input voltage (0.9-V I/O [VDD09_CSIn]) | -0.3 to VDD09_CSIn + 0.3 | V | *5 |
| Output voltage (3.3-V I/O) | -0.3 to VDDQ33 + 0.3 | V | *1 |
| Output voltage (3.3-V I/O [SDHI]) | -0.3 to VDDQVA_SDn + 0.3 | V | *1 |
| Output voltage (3.3-V I/O [SATA]) | -0.3 to VDD33_SATA + 0.3 | V | *1 |
| Output voltage (3.3-V I/O [PCIE]) | -0.3 to VDD33_PCIEIn + 0.3 | V | *1 |
| Output voltage (3.3-V I/O [USB3]) | -0.3 to VDD33_USB3n + 0.3 | V | *1 |
| Output voltage (3.3-V I/O [USB3HSH]) | -0.3 to VDD33_USB3HSHn + 0.3 | V | *1 |
| Output voltage (3.3-V I/O [USB3HS]) | -0.3 to VDD33_USB3HSn + 0.3 | V | *1 |
| Output voltage (3.3-V I/O [USB2H]) | -0.3 to VDDQ33_USB2Hn + 0.3 | V | *1 |
| Output voltage (3.3-V I/O [USB2]) | -0.3 to VDDQ33_USB2 + 0.3 | V | *1 |
| Output voltage (3.3-V I/O [GE]) | -0.3 to VDDQ25_GE + 0.3 | V | *1 |
| Output voltage (3.3-V I/O [AVB]) | -0.3 to VDDQ25_AVB + 0.3 | V | *1 |
| Output voltage (3.3-V I/O [VDDQ_DU]) | -0.3 to VDDQ_DU + 0.3 | V | *1 |
| Output voltage (3.3-V I/O [VDDQ_VIN01]) | -0.3 to VDDQ_VIN01 + 0.3 | V | *1 |
| Output voltage (2.5-V I/O [ETH]) | -0.3 to VDDQ25_ETH + 0.3 | V | *2 |
| Output voltage (2.5-V I/O [GE]) | -0.3 to VDDQ25_GE + 0.3 | V | *2 |
| Output voltage (2.5-V I/O [AVB]) | -0.3 to VDDQ25_AVB + 0.3 | V | *2 |
| Output voltage (1.8-V I/O) | -0.3 to VDDQ18 + 0.3 | V | *3 |
| Output voltage (1.8-V I/O [HDMI]) | -0.3 to VDDQ18_HDMIIn + 0.3 | V | *3 |
| Output voltage (1.8-V I/O [VDDQ18_LVDS]) | -0.3 to VDDQ18_LVDS + 0.3 | V | *3 |
| Output voltage (1.8-V I/O [VDDQ18_CSIn]) | -0.3 to VDDQ18_CSIn + 0.3 | V | *3 |
| Output voltage (1.8-V I/O [VDDQVA_SDn]) | -0.3 to VDDQVA_SDn + 0.3 | V | *3, SDR50, SDR104 |
| Output voltage (1.8-V I/O [VDDQ18_DIGRF]) | -0.3 to VDDQ18_DIGRF + 0.3 | V | *3 |
| Output voltage (1.8-V I/O [VDDQ18_DU]) | -0.3 to VDDQ_DU + 0.3 | V | *3 |
| Output voltage (1.8-V I/O [VDDQ_VIN01]) | -0.3 to VDDQ_VIN01 + 0.3 | V | *3 |
| Output voltage (1.1-V I/O [VDDQVA_DDRn]) | -0.3 to VDDQVA_DDRn + 0.3 | V | *4, LPDDR4 |
| Output voltage (0.9-V I/O [VDD09_SATA]) | -0.3 to VDD09_SATA + 0.3 | V | *5 |

| Item | Value | Unit | Remarks |
|--|-----------------------------|------|---------------|
| Output voltage (0.9-V I/O [VDD09_PCIEn]) | -0.3 to VDD09_PCIEn + 0.3 | V | *5 |
| Output voltage (0.9-V I/O [VDD09_USB3n]) | -0.3 to VDD09_USB3n + 0.3 | V | *5 |
| Output voltage (0.9-V I/O [VDD09_USB3HSn]) | -0.3 to VDD09_USB3HSn + 0.3 | V | *5 |
| Output voltage (0.9-V I/O [VDD09_USB2n]) | -0.3 to VDD09_USB2n + 0.3 | V | *5 |
| Output voltage (0.9-V I/O [VDDQ09_HDMIIn]) | -0.3 to VDDQ09_HDMIIn + 0.3 | V | *5 |
| Output voltage (0.9-V I/O [VDDQ09_LVDS]) | -0.3 to VDDQ09_LVDS + 0.3 | V | *5 |
| Output voltage (0.9-V I/O [VDD09_CSIn]) | -0.3 to VDD09_CSIn + 0.3 | V | *5 |
| Operating temperature | -40 to +85 | °C | Ta (ambient) |
| | -40 to +115 | °C | Tj (junction) |
| Maximum rating temperature | -40 to +125 | °C | Tj (junction) |
| Storage temperature | -55 to +125 | °C | Ta (ambient) |

Notes: Permanent damage to the LSI may result if absolute maximum ratings are exceeded. In normal operation, this LSI should be used within the specifications described in the following descriptions. If this LSI is not used within the specifications, the reliability of this LSI may lower.

Voltages are referenced at GND = VSS = AVSS = 0 V.

Abbreviations in the tables have the following meanings.

SDHI: SDHI does not operate through SDR50 and SDR104.

SDHI (SDR50, SDR104): SDHI operates through SDR50 and SDR104.

For details of the number of channels or available modules, refer to each manual; SDHI, DU/LVDS, SATA/PCIEC/USB3.0, DBSC4 (DDR), USB2.0, HDMI and CSI and Pin Information of Individual RZ/G Series Product.

1. Do not exceed 3.84 V.
2. Do not exceed 2.90 V.
3. Do not exceed 2.34 V.
4. Do not exceed 1.50 V.
5. Do not exceed 1.12 V.

Table 73.1.2 Absolute Maximum Ratings for [RZ/G2E]



| Item | Value | Unit | Remarks |
|---|---|------|---------------|
| Power supply voltage (3.3 V) (VDDQ33, VDDQ25_AVB0, VDDQ_SDn (n = 0, 1, 3), VDDQ_QSPI, VDDQ33_USB2) | -0.3 to +4.6 | V | *1 |
| Power supply voltage (2.5 V) (VDDQ25_AVB0) | -0.3 to +3.3 | V | *2 |
| Power supply voltage (1.8 V) (VDDQ18, VDD18_CPGPLLn (n = 0, 1, 3) VDDQ18_LVDS, VDD18_LVDSnPLL (n = 0, 1), VDDQ18_CSIO, VDDQ18_USB20, VDDQ18_USB30, VDDQ18_PCIE0, VDDQ_MAPLL, VDDQ_SD0, VDDQ_SD1, VDDQ_SD3, VDDQ_QSPI) | -0.3 to +2.5 | V | *3 |
| Power supply voltage (1.5 V) (VDDQ_CK, VDDQ_DDR) | -0.3 to + 1.875 | V | *4 |
| Power supply voltage (1.35 V) (VDDQ_CK, VDDQ_DDR) | -0.3 to + 1.75 | V | *5 |
| Power supply voltage (1.0 V) (VDD, VDDD_USB30, VDDD_PCIE0) | -0.3 to + 1.26 | V | |
| Input voltage (3.3-V I/O) | -0.3 to VDDQ33 + 0.3 | V | *1 |
| Input voltage (3.3-V I/O [AVB0]) | -0.3 to VDDQ25_AVB0 + 0.3 | V | *1 |
| Input voltage (3.3-V I/O [VDDQ_SD]) | -0.3 to VDDQ_SDn + 0.3 (n = 0, 1, 3) | V | *1 |
| Input voltage (3.3-V I/O [VDDQ_QSPI]) | -0.3 to VDDQ_QSPI + 0.3 | V | *1 |
| Input voltage (2.5-V I/O [AVB0]) | -0.3 to VDDQ25_AVB0 + 0.3 | V | *2 |
| Input voltage (1.8-V I/O) | -0.3 to VDDQ18 + 0.3 | V | *3 |
| Input voltage (1.8-V I/O [VDDQ_SD]) | -0.3 to VDDQ_SDn + 0.3 (n = 0, 1, 3) | V | *1 |
| Input voltage (1.8-V I/O [VDDQ_QSPI]) | -0.3 to VDDQ_QSPI + 0.3 | V | *1 |
| Input voltage (1.35-V I/O [VDDQ_DDR]) | -0.3 to VDDQ_DDR + 0.3 | V | *5 |
| Output voltage (3.3-V I/O) | -0.3 to VDDQ33 + 0.3 | V | *1 |
| Output voltage (3.3-V I/O [AVB0]) | -0.3 to VDDQ25_AVB0 + 0.3 | V | *1 |
| Output voltage (2.5-V I/O [AVB0]) | -0.3 to VDDQ25_AVB0 + 0.3 | V | *2 |
| Output voltage (1.8-V I/O) | -0.3 to VDDQ18 + 0.3 | V | *3 |
| Output voltage (1.8-V I/O [VDDQ18_LVDS]) | -0.3 to VDDQ18_LVDS + 0.3 | V | *3 |
| Output voltage (1.8-V I/O [VDDQ_SD]) | -0.3 to VDDQ_SDn + 0.3 (n = 0, 1, 3) | V | *1 |
| Output voltage (1.8-V I/O [VDDQ_QSPI]) | -0.3 to VDDQ_QSPI + 0.3 | V | *1 |
| Output voltage (1.35-V I/O [VDDQ_DDR]) | -0.3 to VDDQ_DDR + 0.3 | V | *5 |
| Operating temperature | -40 to +115 | °C | Tj (junction) |
| Maximum rating temperature | -40 to +125 | °C | Tj (junction) |
| Storage temperature | -55 to +125 | °C | Ta (ambient) |

Notes: Permanent damage to the LSI may result if absolute maximum ratings are exceeded. In normal operation, this LSI should be used within the specifications described in the following descriptions. If this LSI is not used within the specifications, the reliability of this LSI may lower.

Voltages are referenced at GND = VSS = AVSS = 0 V.

1. Do not exceed 4.6 V
2. Do not exceed 3.3 V
3. Do not exceed 2.5 V
4. Do not exceed 1.875 V
5. Do not exceed 1.75 V

73.2 Power Supply

Table 73.2.1 Power Supply for [RZ/G2H] [RZ/G2M V1.3] [RZ/G2M V3.0][RZ/G2N]

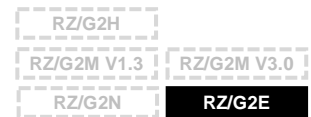
| | |
|--------------------|--------------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Voltage | | | Unit | Remarks |
|---|------------------------------|---------|------|------|------|---|
| | | Min. | Typ. | Max. | | |
| Power supply (Internal) | VDD | 0.75 | 0.82 | 0.88 | V | VDD for power supply and VSS for ground |
| Power supply (Internal) | VDD_DVFS | 0.76 | 0.83 | 0.88 | V | VDD_DVFS for power supply and VSS for ground. |
| Power supply (3.3-V I/O) | VDDQ33 | 3.1 | 3.3 | 3.5 | V | VDDQ33 for power supply and VSS for ground |
| Power supply (3.3-V I/O [SDHI]) | VDDQVA_SDn (n = 0 to 3) | 3.1 | 3.3 | 3.5 | V | VDDQVA_SDn for power supply and VSS for ground |
| Power supply (3.3-V I/O [SATA]) | VDD33_SATA*1 | 3.1 | 3.3 | 3.5 | V | VDD33_SATA for power supply and VSS for ground |
| Power supply (3.3-V I/O [PCIE]) | VDD33_PCIE n (n = 0, 1) | 3.1 | 3.3 | 3.5 | V | VDD33_PCIE n (n = 0, 1) for power supply and VSS for ground |
| Power supply (3.3-V I/O [USB3HSH]) | VDDQ33_USB3H SHn (n = 0, 1) | 3.1 | 3.3 | 3.5 | V | VDDQ33_USB3HSHn (n = 0, 1) for power supply and VSS for ground |
| Power supply (3.3-V I/O [USB3HS]) | VDDQ33_USB3H S n (n = 0, 1) | 3.1 | 3.3 | 3.5 | V | VDDQ33_USB3HSn (n = 0, 1) for power supply and VSS for ground |
| Power supply (3.3-V I/O [USB3]) | VDD33_USB3n (n = 0, 1) | 3.1 | 3.3 | 3.5 | V | VDD33_USB3n (n = 0, 1) for power supply and VSS for ground |
| Power supply (3.3-V I/O [USB2H]) | VDDQ33_USB2H n (n = 0, 1, 2) | 3.1 | 3.3 | 3.5 | V | VDDQ33_USB2Hn (n = 0, 1, 2) for power supply and VSS for ground |
| Power supply (3.3-V I/O [USB2]) | VDDQ33_USB2 | 3.1 | 3.3 | 3.5 | V | VDDQ33_USB2 for power supply and VSS for ground |
| Power supply (2.5-V I/O [ETH]) | VDDQ25_ETH | 2.4 | 2.5 | 2.6 | V | VDDQ25_ETH for power supply and VSS for ground |
| Power supply (1.8-V I/O [VDDQ18]) | VDDQ18 | 1.7 | 1.8 | 1.9 | V | VDDQ18 for power supply and VSS for ground |
| Power supply (1.8-V I/O [HDMI]) | VDDQ18_HDMI n (n = 0, 1) | 1.7 | 1.8 | 1.9 | V | VDDQ18_HDMI n (n = 0, 1) for power supply and VSS for ground |
| Power supply (1.8-V I/O [LVDS]) | VDDQ18_LVDS | 1.7 | 1.8 | 1.9 | V | VDDQ18_LVDS for power supply and VSS for ground |
| Power supply (1.8-V I/O [CSI]) | VDDQ18_CSIn (n = 0 to 3) | 1.7 | 1.8 | 1.9 | V | VDDQ18_CSIn (n = 0 to 3) for power supply and VSS for ground |
| Power supply (1.8-V I/O [SDHI (SDR50, SDR104)]) | VDDQVA_SDn (n = 0 to 3) | 1.7 | 1.8 | 1.9 | V | VDDQVA_SDn (n = 0 to 3) for power supply and VSS for ground |
| Power supply (1.1-V I/O [DDR (LPDDR4)]) | VDDQVA_DDRn (n = 0 to 3) | 1.06 | 1.10 | 1.17 | V | VDDQVA_DDRn (n = 0 to 3) for power supply and VSS for ground |
| Power supply (0.9-V I/O [SATA]) | VDD09_SATA*1 | 0.75 | 0.82 | 0.88 | V | VDD09_SATA for power supply and VSS for ground |
| Power supply (0.9-V I/O [PCIE]) | VDD09_PCIE n (n = 0, 1) | 0.75 | 0.82 | 0.88 | V | VDD09_PCIE n (n = 0, 1) for power supply and VSS for ground |
| Power supply (0.9-V I/O [USB3]) | VDD09_USB3n (n = 0, 1) | 0.75 | 0.82 | 0.88 | V | VDD09_USB3n (n = 0, 1) for power supply and VSS for ground |

| Item | Symbol | Voltage | | | Unit | Remarks |
|------------------------------------|-------------------------------|---------|------|------|------|--|
| | | Min. | Typ. | Max. | | |
| Power supply (0.9-VI/O[USB3HS]) | VDD09_USB3HSn (n = 0, 1) | 0.75 | 0.82 | 0.88 | V | VDD09_USB3HSn (n = 0, 1) for power supply and VSS for ground |
| Power supply (0.9-VI/O[USB2]) | VDD09_USB2n (n = 0, 1, 2) | 0.75 | 0.82 | 0.88 | V | VDD09_USB2n (n = 0, 1, 2) for power supply and VSS for ground |
| Power supply (0.9-VI/O[HDMI]) | VDDQ09_HDMI (n = 0, 1) | 0.75 | 0.82 | 0.88 | V | VDDQ09_HDMI (n = 0, 1) for power supply and VSS for ground |
| Power supply (0.9-VI/O[LVDS]) | VDDQ09_LVDS | 0.75 | 0.82 | 0.88 | V | VDDQ09_LVDS for power supply and VSS for ground |
| Power supply (0.9-VI/O[CSI]) | VDD09_CSIn (n = 0 to 3) | 0.75 | 0.82 | 0.88 | V | VDD09_CSIn (n = 0 to 3) for power supply and VSS for ground |
| Power supply (PLL[LVDS]) | VDD09_LVDSPLL 2 | 0.75 | 0.82 | 0.88 | V | VDD09_LVDSPLL2 for power supply and VSS for ground respectively |
| Power supply (PLL[CPG]) | VDD18_CPGPLLn (n = 0 to 4) | 1.7 | 1.8 | 1.9 | V | VDD18_CPGPLLn (n = 0 to 4) for power supply and VSS_CPGPLLn (n = 0 to 4) for ground |
| Power supply (PLL[MLB]) | VDD18_MLBPLL | 1.7 | 1.8 | 1.9 | V | VDD18_MLBPLL for power supply and VSS_MLBPLL for ground respectively |
| Power supply (PLL[DU]) | VDD18_DUPLLn (n = 0, 1) | 1.7 | 1.8 | 1.9 | V | VDD18_DUPLLn (n = 0, 1) for power supply and VSS_DUPLLn (n = 0, 1) for ground respectively |
| Power supply (PLL[LVDS]) | VDD18_LVDSPLL 1 | 1.7 | 1.8 | 1.9 | V | VDD18_LVDSPLL 1 for power supply and VSS for ground respectively |
| Power supply (PLL[DDR]) | VDD09_DDRPLLn (n = 0 to 3) | 0.75 | 0.82 | 0.88 | V | VDD09_DDRPLLn (n = 0 to 3) for power supply and VSS for ground respectively |

Note: 1. Only for RZ/G2H, RZ/G2N.

Table 73.2.2 Power Supply for [RZ/G2E]



| Item | Symbol | Voltage | | | Unit | Remarks |
|---------------------------------------|--------------------------------|---------|------|------|------|--|
| | | Min. | Typ. | Max. | | |
| Power supply (Internal) | VDD | 0.98 | 1.03 | 1.08 | V | VDD for power supply and VSS for ground |
| Power supply (PLL[CPG]) | VDD18_CPGPLLn (n = 0, 1, 3) | 1.7 | 1.8 | 1.9 | V | VDD18_CPGPLLn (n = 0, 1, 3) for power supply and VSS_CPGPLLn (n = 0, 1, 3) |
| Power supply (PLL[MLB]) | VDD18_MLBPLL | 1.7 | 1.8 | 1.9 | V | VDD18_MLBPLL for power supply and VSS_MLBPLL for ground |
| Power supply (1.8-V I/O[VDDQ18]) | VDDQ18 | 1.7 | 1.8 | 1.9 | V | VDDQ18 for power supply and VSS for ground |
| Power supply (1.8-V I/O [VDDQ18_ISO]) | VDDQ18_ISO | 1.7 | 1.8 | 1.9 | V | VDDQ18_ISO for power supply and VSS for ground |
| Power supply (1.8-V I/O[CSI]) | VDDQ18_CSI0 | 1.7 | 1.8 | 1.9 | V | VDDQ18_CSI0 for power supply and VSS for ground |
| Power supply (1.8-V I/O[LVDS]) | VDDQ18_LVDS | 1.7 | 1.8 | 1.9 | V | VDDQ18_LVDS for power supply and VSS for ground |
| Power supply (PLL[LVDS]) | VDD18_LVDSnPL L (n = 0, 1) | 1.7 | 1.8 | 1.9 | V | VDD18_LVDSnPLL (n = 0, 1) for power supply and VSS_LVDSnPLL (n = 0, 1) for ground respectively |
| Power supply (2.5-V I/O[AVB0]) | VDDQ25_AVB0 | 2.3 | 2.5 | 2.7 | V | VDDQ25_AVB0 for power supply and VSS for ground |
| Power supply (3.3-V I/O[AVB0]) | VDDQ25_AVB0 | 3.0 | 3.3 | 3.6 | V | VDDQ25_AVB0 for power supply and VSS for ground |
| Power supply (3.3-V I/O) | VDDQ33 | 3.0 | 3.3 | 3.6 | V | VDDQ33 for power supply and VSS for ground |
| Power supply (1.35-V I/O[DDR]) | VDDQ_CK, VDDQ_DDR | 1.283 | 1.35 | 1.45 | V | VDDQ_CK, VDDQ_DDR for power supply and VSS for ground |
| Power supply (1.8-V I/O[SD]) | VDDQ_SDn (n = 0, 1, 3) | 1.7 | 1.8 | 1.9 | V | VDDQ_SDn (n = 0, 1, 3) for power supply and VSS for ground respectively |
| Power supply (1.8-V I/O[QSPI]) | VDDQ_QSPI | 1.7 | 1.8 | 1.9 | V | VDDQ_QSPI for power supply and VSS for ground respectively |
| Power supply (3.3-V I/O[SD]) | VDDQ_SDn (n = 0, 1, 3) | 3.0 | 3.3 | 3.6 | V | VDDQ_SDn (n = 0, 1, 3) for power supply and VSS for ground respectively |
| Power supply (3.3-V I/O[QSPI]) | VDDQ_QSPI | 3.0 | 3.3 | 3.6 | V | VDDQ_QSPI for power supply and VSS for ground respectively |
| Power supply (PLL) | VDDQ_MAPLL | 1.7 | 1.8 | 1.9 | V | VDDQ_MAPLL for power supply and VSS for ground respectively |
| Power supply (PLL) | VDDQ_MDPLLn (n = 0, 1) | 1.7 | 1.8 | 1.9 | V | VDDQ_MDPLLn (n = 0, 1) for power supply and VSS for ground respectively |

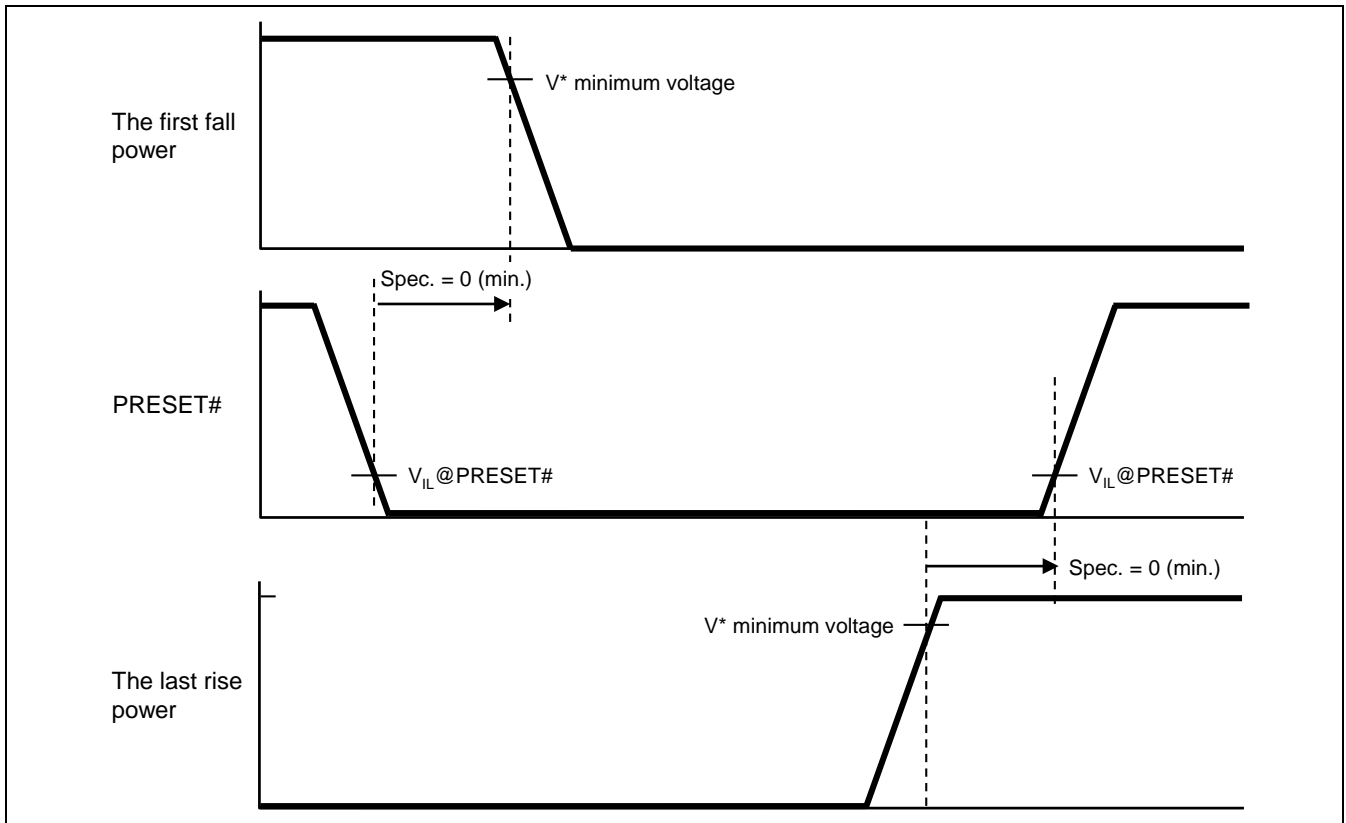
| Item | Symbol | Voltage | | | Unit | Remarks |
|-----------------------|--------------|---------|------|------|------|---|
| | | Min. | Typ. | Max. | | |
| Power supply (USB) | AVDD_USB | 1.7 | 1.8 | 1.9 | V | AVDD_USB for power supply and AVSS for ground respectively |
| | VDDQ33_USB2 | 3.0 | 3.3 | 3.6 | V | VDDQ33_USB2 for power supply and VSS for ground respectively |
| | VDDQ18_USB20 | 1.7 | 1.8 | 1.9 | V | VDDQ18_USB20 for power supply and VSS for ground respectively |
| Power supply (USB3.0) | VDDD_USB30 | 0.98 | 1.03 | 1.08 | V | VDDD_USB30 for power supply and VSS for ground |
| | VDDQ18_USB30 | 1.7 | 1.8 | 1.9 | V | VDDQ18_USB30 for power supply and VSS for ground |
| Power supply (PCIe) | VDDD_PCIE0 | 0.98 | 1.03 | 1.08 | V | VDDD_PCIE0 for power supply and VSS for ground |
| | VDDQ18_PCIE0 | 1.7 | 1.8 | 1.9 | V | VDDQ18_PCIE0 for power supply and VSS for ground |

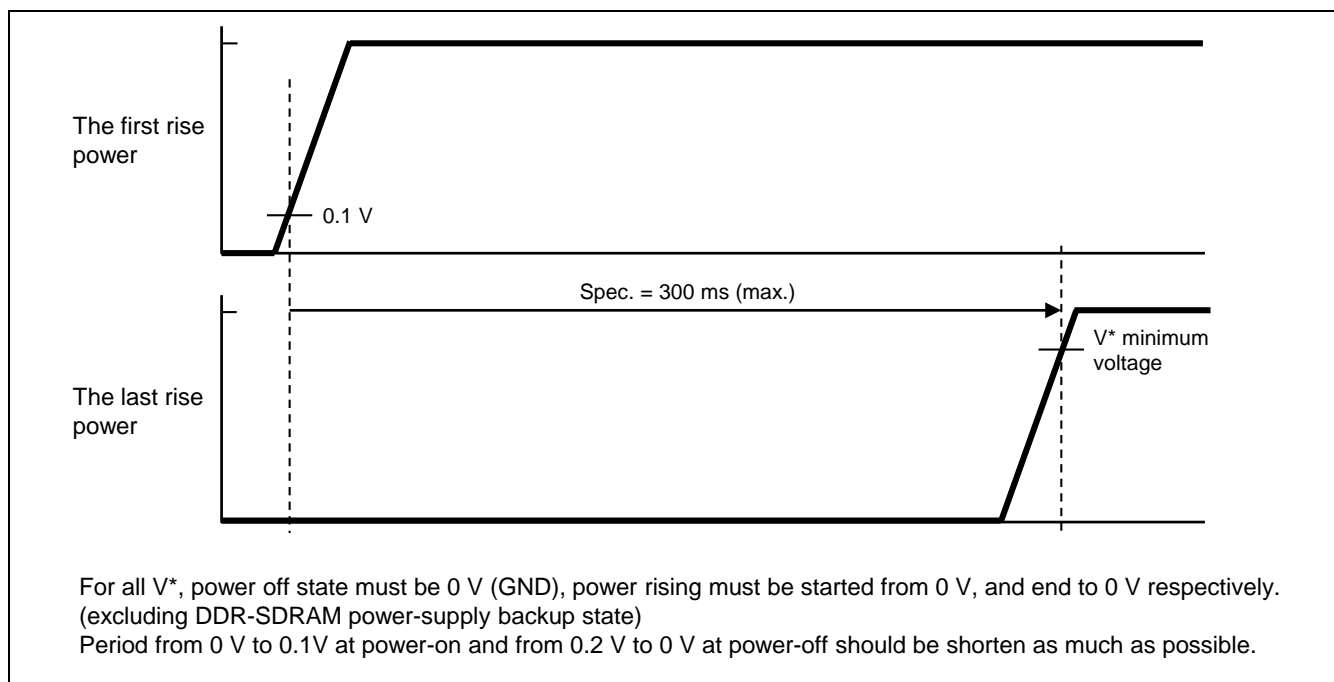
73.3 Sequence of Turning On/Off Power Supplies

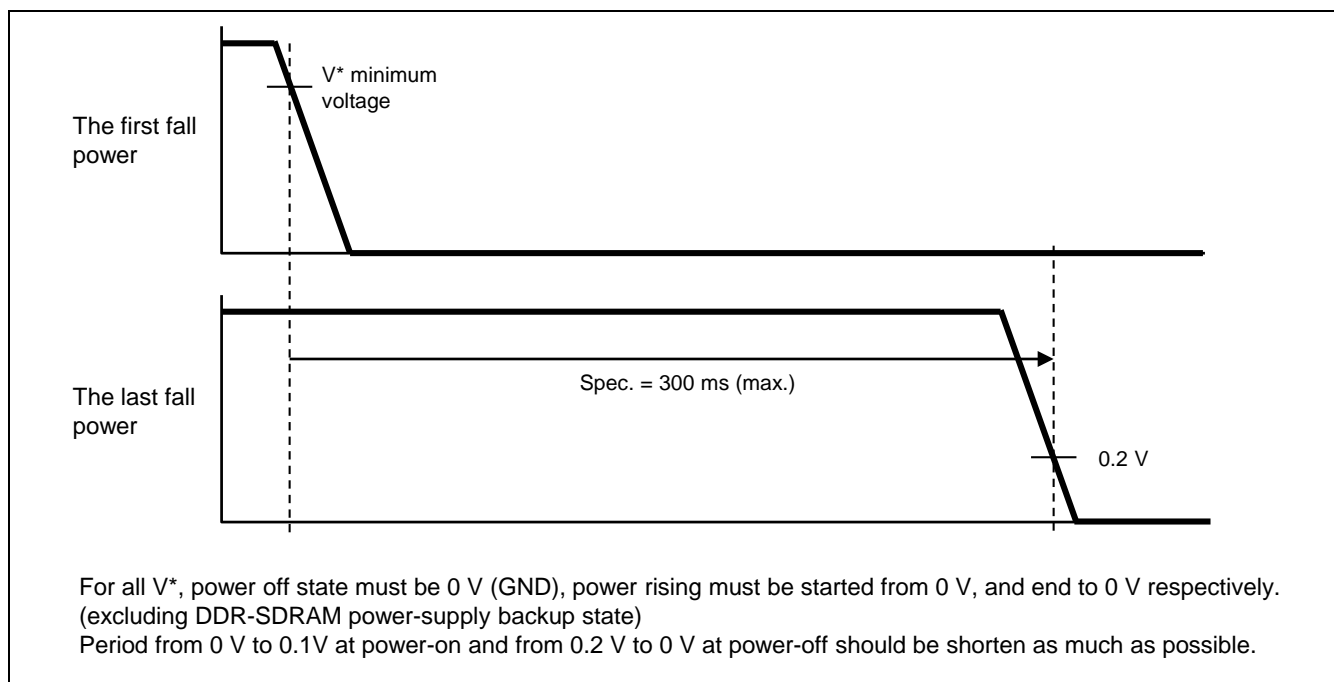
73.3.1 Sequence of Turning On/Off Power Supplies for [RZ/G2H] [RZ/G2M V1.3] [RZ/G2M V3.0] [RZ/G2N]

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

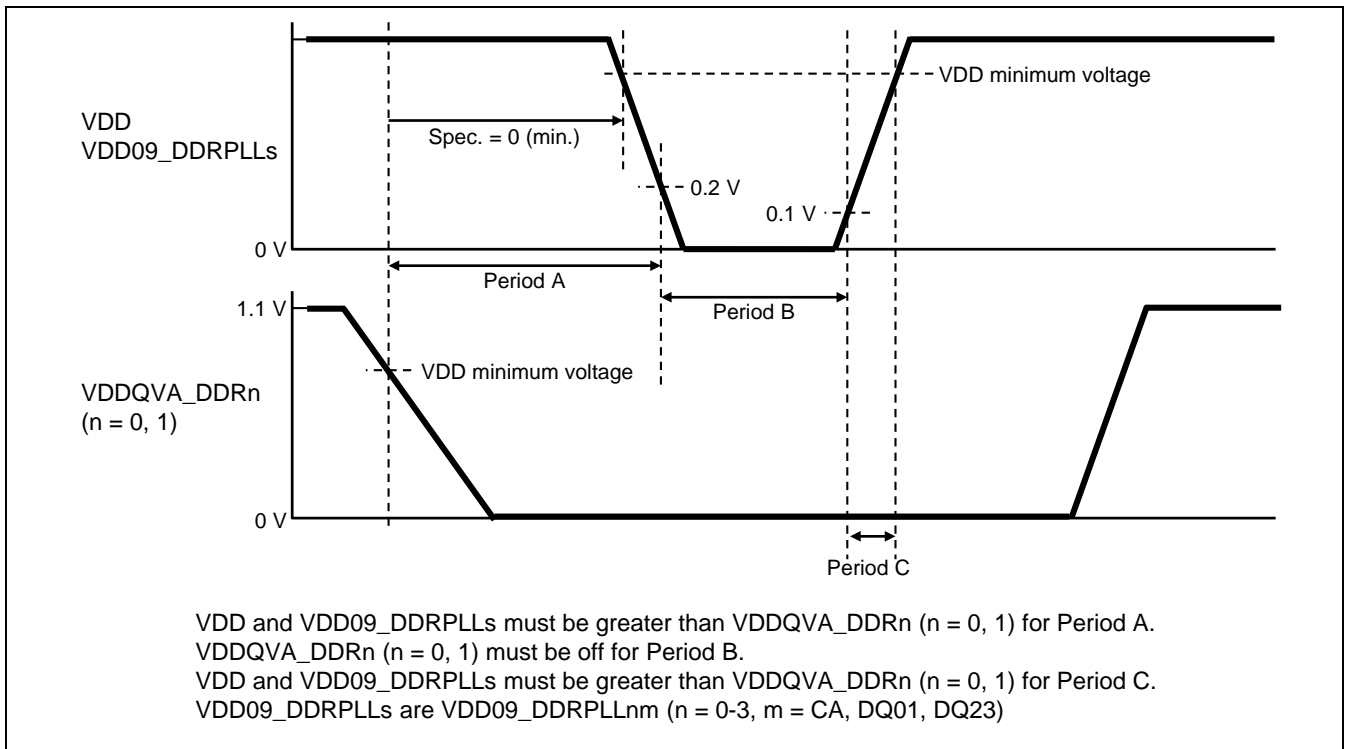
(1) PRESET# VS. Power



(2) Period for Power Rise for RZ/G2H, RZ/G2M V1.3, RZ/G2N

(3) Period for Power Fall for RZ/G2H, RZ/G2M V1.3, RZ/G2N

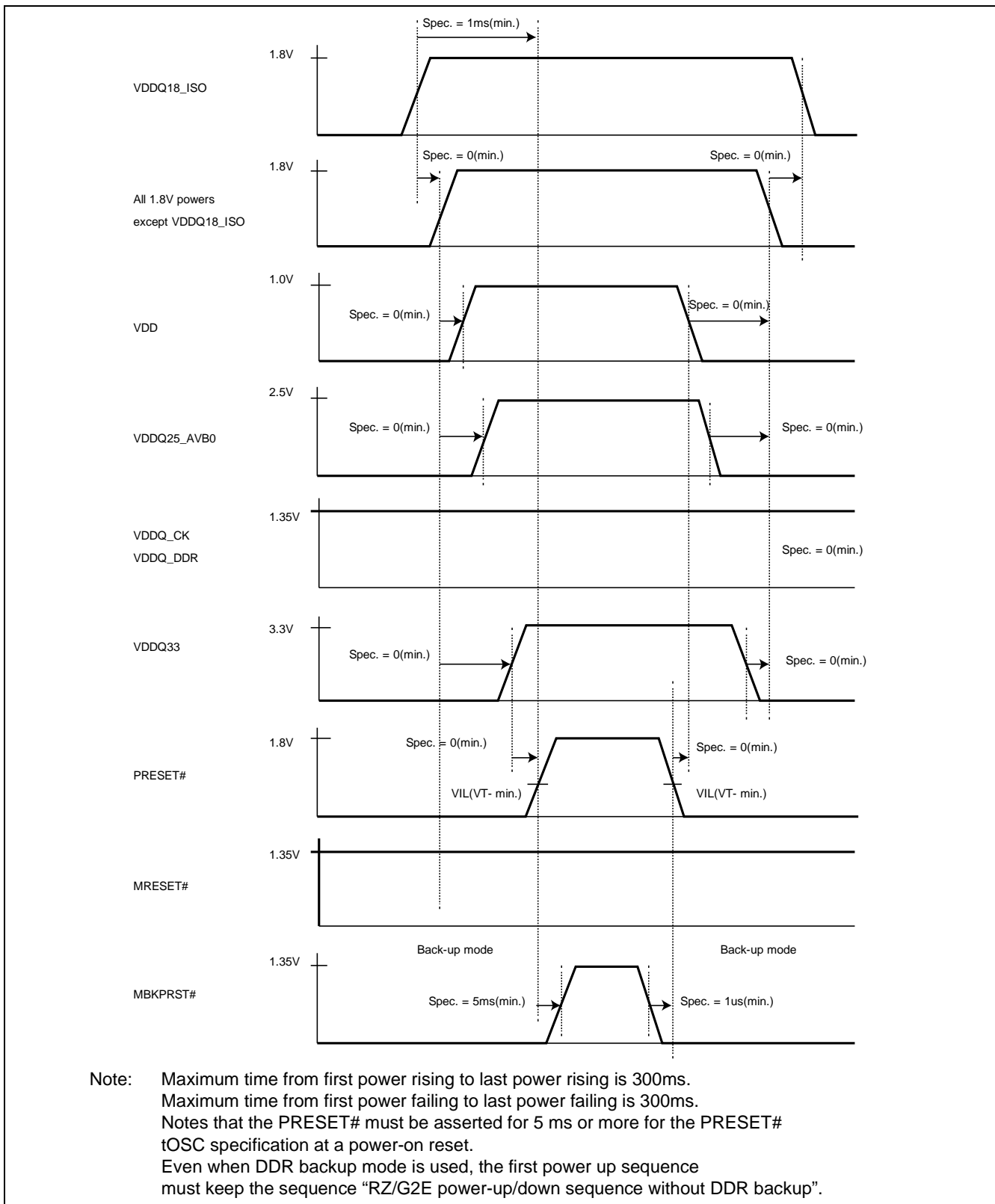
(4) VDD and VDD09_DDRPLLs vs VDDQVA_DDRn (n = 0, 1)



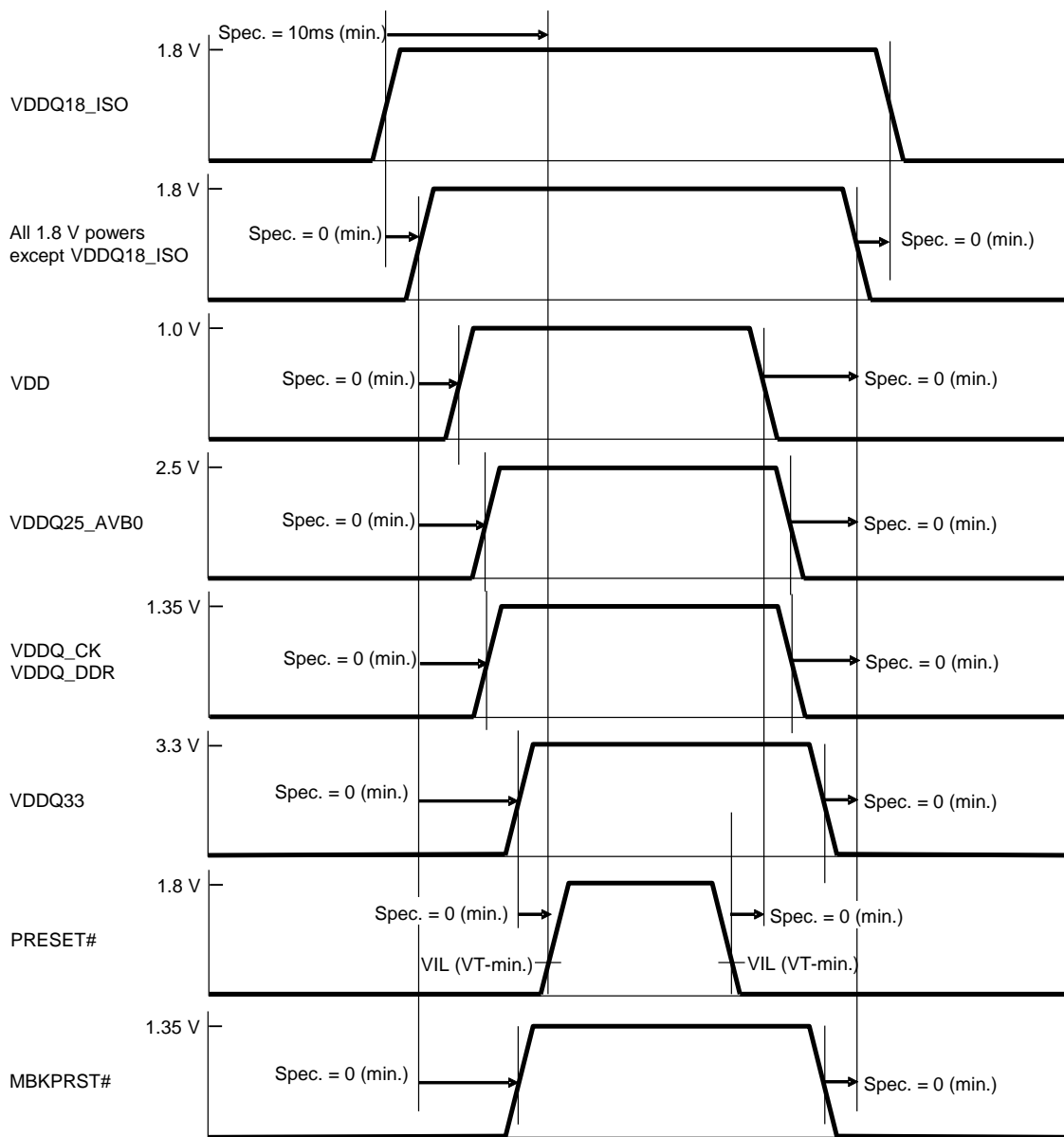
73.3.2 Sequence of Turning On/Off Power Supplies for [RZ/G2E]

| | |
|-------------|---------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

(1) RZ/G2E power-up/down sequence with DDR backup



(2) RZ/G2E power-up/down sequence without DDR backup

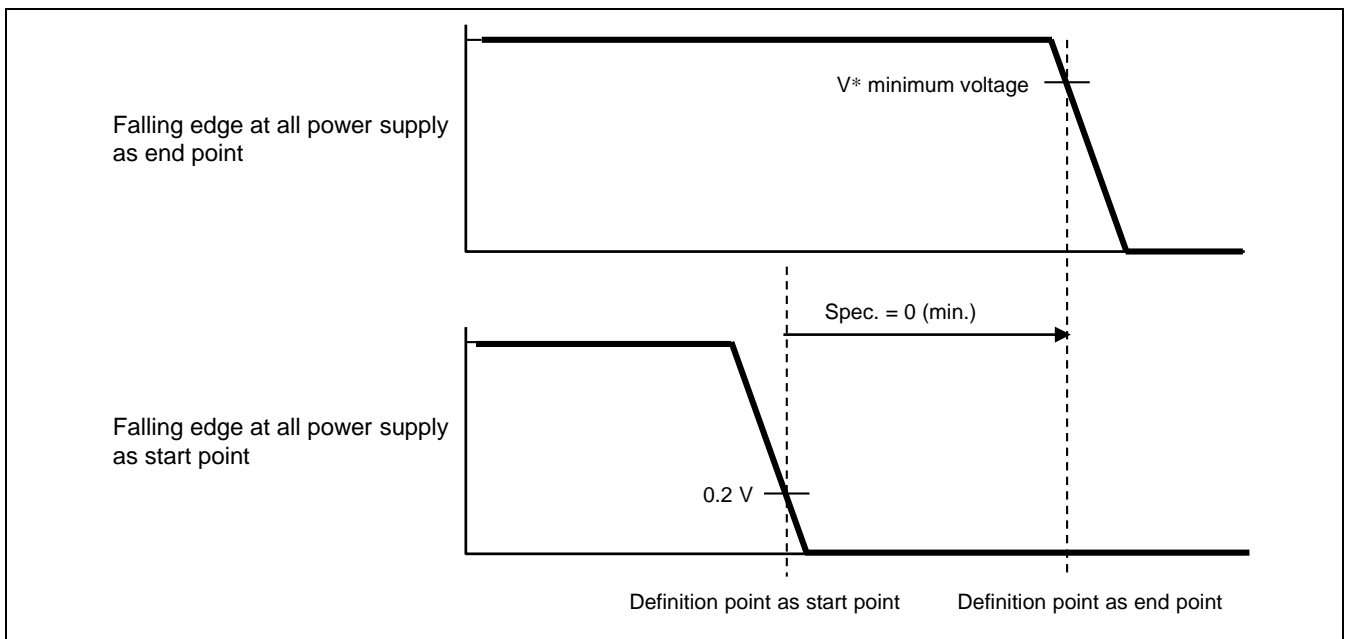
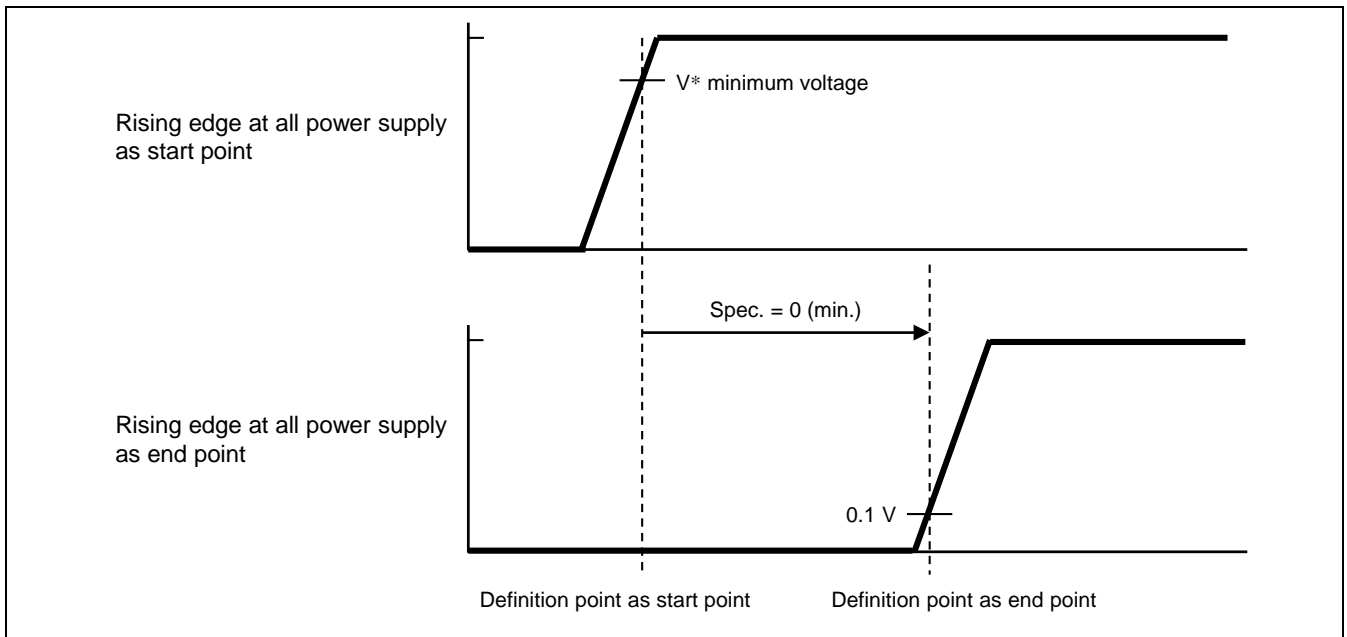


- Notes: 1. Maximum time from first power rising to last power rising is 300ms.
 Maximum time from first power failing to last power failing is 300ms.
 Notes that the PRESET# must be asserted for 5 ms or more for the PRESET# tOSC specification at a power-on reset.
- 2. MBKPRST# input must track VDDQ_DDR/VDDQ_CK during VDDQ_DDR/VDDQ_CK power-up.

73.3.3 Wave form definition for power sequence

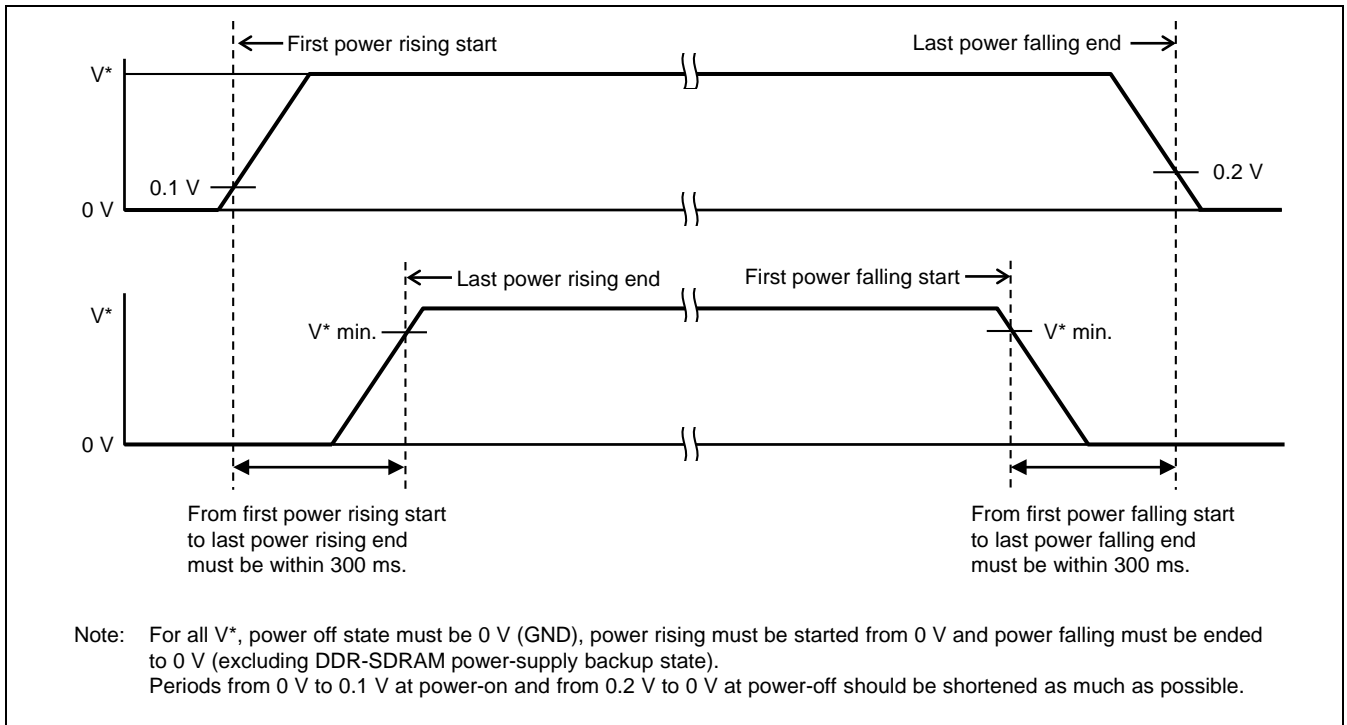
| | |
|-------------|---------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

This definition is for the different voltage power supply for example between 1.8-V and 3.3-V, it is possible to turn on or off at the same time for the same voltage power supply.



73.3.4 Power On and Power Off Wave Form

| | |
|-------------|---------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |



73.4 DC Characteristics

Table 73.4.1 Supply Current for LSI

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Power | | | Unit | Remarks |
|--------------------------------------|-----------------------------|-------|------|--------|------|--|
| | | Min. | Typ. | Max. | | |
| Supply current (Internal) | IDD | — | — | 8700 | mA | VDD = 0.88 V [RZ/G2H] |
| | | | | 6846.0 | mA | VDD = 0.88 V [RZ/G2M V1.3] |
| | | | | 6908.0 | mA | VDD = 0.88 V [RZ/G2M V3.0] |
| | | | | 3554.9 | mA | VDD = 0.88 V [RZ/G2N] |
| | | | | | | Condition: Cortex-A53: Dhrystone-MAX, |
| Supply current (Internal) | IDD_DVFS | — | — | 12000 | mA | VDD_DVFS = 0.88 V [RZ/G2H] |
| | | | | 7632.9 | mA | VDD_DVFS = 0.88 V [RZ/G2M V1.3] |
| | | | | 8351.0 | mA | VDD_DVFS = 0.88 V [RZ/G2M V3.0] |
| | | | | 5554.6 | mA | VDD_DVFS = 0.88 V [RZ/G2N] |
| | | | | | | Condition: Cortex-A57: Dhrystone-MAX, GPU: Polygon fill-MAX |
| Supply current (3.3-V I/O) | IDDQ33 | — | — | 94 | mA | VDDQ33 = 3.5 V [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (3.3-V I/O [SDHI]) | IDDQVA_SDn (n = 0 to 3) | — | — | 31 | mA | VDDQVA_SDn = 3.5 V (n = 0 to 3) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (3.3-V I/O [SATA]) | IDD33_SATA | — | — | 57.6 | mA | VDD33_SATA = 3.5 V [RZ/G2H], [RZ/G2N] |
| Supply current (3.3-V I/O [PCIE]) | IDD33_PCIEn (n = 0, 1) | — | — | 57.6 | mA | VDD33_PCIEn = 3.5 V (n = 0, 1) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (3.3-V I/O [USB3HSH]) | IDDQ33_USB3 HSHn (n = 0) | — | — | 72 | mA | VDDQ33_USB3HSHn = 3.5 V (n = 0) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (3.3-V I/O [USB3HS]) | IDDQ33_USB3 HSn (n = 0) | — | — | 72 | mA | VDDQ33_USB3HSn = 3.5 V (n = 0) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (3.3-V I/O [USB3]) | IDD33_USB3n (n = 0) | — | — | 57.6 | mA | VDD33_USB3n = 3.5 V (n = 0) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (3.3-V I/O [USB2H]) | IDDQ33_USB2 Hn (n = 0 to 3) | — | — | 72 | mA | VDDQ33_USB2Hn = 3.5 V (n = 0, 1) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (3.3-V I/O [USB2]) | IDDQ33_USB2 | — | — | 216 | mA | VDDQ33_USB2 = 3.5 V [RZ/G2H] |
| | | | | 144 | mA | VDDQ33_USB2 = 3.5 V [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (2.5-V I/O [ETH]) | IDDQ25_ETH | — | — | 40 | mA | VDDQ25_ETH = 2.6 V [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (1.8-V I/O [VDDQ18]) | IDDQ18 | — | — | 50 | mA | VDDQ18 = 1.9 V [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (1.8-V I/O [HDMI]) | IDDQ18_HDMI n (n = 0, 1) | — | — | 10.5 | mA | VDDQ18_HDMI n = 1.9 V (n = 0) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (1.8-V I/O [LVDS]) | IDDQ18_LVDS | — | — | 110.2 | mA | VDDQ18_LVDS = 1.9 V [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |

| Item | Symbol | Power | | | Unit | Remarks |
|--|--------------------------------------|----------------------------------|------|------|--|--|
| | | Min. | Typ. | Max. | | |
| Supply current (1.8-V I/O [CSI]) | IDDQ18_CSIn (n = 0 to 2) | — | — | 6.3 | mA | VDDQ18_CSIn = 1.9 V (n = 0, 2) [RZ/G2H] (n = 0) |
| | | | | 4.5 | mA | VDDQ18_CSIn = 1.9 V (n = 1) [RZ/G2H] |
| | | | | 6.3 | mA | VDDQ18_CSIn = 1.9 V (n = 0) [RZ/G2M V1.3], [RZ/G2N] |
| | | | | 4.5 | mA | VDDQ18_CSIn = 1.9 V (n = 1) [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (1.8-V I/O [SDHI(SDR50, SDR104)]) | IDDQVA_SDn (n = 0 to 3) | — | — | 35 | mA | VDDQVA_SDn = 1.9 V (n = 0 to 3) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (1.1-V I/O [DDR(LPDDR4)]) | Normal IDDQVA_DDR n (n = 0, 1) | — | — | 470 | mA | VDDQVA_DDRn = 1.17 V (n = 0, 1) [RZ/G2M V1.3], [RZ/G2M V3.0], [RZ/G2H] |
| | | | | 330 | mA | VDDQVA_DDRn = 1.17 V (n = 0) [RZ/G2N] |
| | | | | | | Condition: Driver impedance: 40Ω No termination Load capacitance: 2.5pF Write Read rate: 50%, 50% I/O Toggle rate: 100% |
| | | DDR power supply backup | — | — | 8 | mA |
| | | | 4 | mA | VDDQVA_DDRn = 1.17 V (n = 0, 1) [RZ/G2M V1.3] | |
| | | | 3 | mA | VDDQVA_DDRn = 1.17 V (n = 0) [RZ/G2N] | |
| Supply current (0.9-V I/O[SATA]) | IDD09_SATA | — | — | 162 | mA | VDD09_SATA = 0.88 V [RZ/G2H], [RZ/G2N] |
| Supply current (0.9-V I/O[PCIE]) | IDD09_PCIE n (n = 0, 1) | — | — | 162 | mA | VDD09_PCIE n = 0.88 V (n = 0, 1) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| | | | | 90 | | [RZ/G2M V1.3] |
| Supply current (0.9-V I/O[USB3]) | IDD09_USB3n (n = 0) | — | — | 162 | mA | VDD09_USB3n = 0.88 V (n = 0) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| | | | | 90 | | [RZ/G2M V1.3] |
| Supply current (0.9-V I/O[USB3HS]) | IDD09_USB3H Sn (n = 0) | — | — | 50.4 | mA | VDD09_USB3HSn = 0.88 V (n = 0) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| | | | | 28 | | [RZ/G2M V1.3] |
| Supply current (0.9-V I/O[USB2]) | IDD09_USB2n (n = 0 to 3) | — | — | 28 | mA | VDD09_USB2n = 0.88 V (n = 0, 1) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (0.9-V I/O[HDMI]) | IDDQ09_HDMI n (n = 0, 1) | — | — | 33.7 | mA | VDDQ09_HDMI n = 0.88 V (n = 0) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (0.9-V I/O[LVDS]) | IDDQ09_LVDS | — | — | 9 | mA | VDDQ09_LVDS = 0.88 V [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |

| Item | Symbol | Power | | | Unit | Remarks |
|--------------------------------|--------------------------------|-------|------|------|------|--|
| | | Min. | Typ. | Max. | | |
| Supply current (0.9-VI/O[CSI]) | IDD09_CSIn (n = 0 to 2) | — | — | 28.8 | mA | VDD09_CSIn = 0.88 V (n = 0) [RZ/G2H] |
| | | | | 18 | | VDD09_CSIn = 0.88 V (n = 1) [RZ/G2H] |
| | | | | 28.8 | | VDD09_CSIn = 0.88 V (n = 0) [RZ/G2M V1.3], [RZ/G2N] |
| | | | | 18 | | VDD09_CSIn = 0.88 V (n = 1) [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (PLL [CPG]) | IDD18_CPGPL Ln (n = 0 to 4) | — | — | 3.0 | mA | VDD18_CPGPLLn = 1.9 V (n = 0 to 4) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (PLL [MLB]) | IDD18_MLBPL L | — | — | 3.0 | mA | VDD18_MLBPLL = 1.9 V [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (PLL [DU]) | IDD18_DUPLL n (n = 0, 1) | — | — | 3.0 | mA | VDD18_DUPLLn = 1.9 V (n = 0) [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (PLL [LVDS]) | IDD18_LVDSP LL 1 | — | — | 12.6 | mA | VDD18_LVDSPLL1 = 1.9 V [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (PLL [LVDS]) | IDD09_LVDSP LL 2 | — | — | 18 | mA | VDD09_LVDSPLL2 = 0.88 V [RZ/G2H], [RZ/G2M V1.3], [RZ/G2N] |
| Supply current (PLL [DDR]) | IDD09_DDRPL L n (n = 0, 1) | — | — | 50 | mA | VDD09_DDRPLLn = 0.88 V (n = 0, 1) [RZ/G2M V1.3] VDD09_DDRPLLn = 0.88 V (n = 0) [RZ/G2N] |

Note: Tc(Tt) = 115 °C (ΘJc = 0.1°C/W) [RZ/G2H], [RZ/G2M V1.3]

Table 73.4.2 Supply current for [RZ/G2E]



| Item | Symbol | Power | | | Unit | Remarks |
|----------------------------------|------------------------------|-------|------|-------------|------|---|
| | | Min. | Typ. | Max. | | |
| Supply current (Internal) | IDD | — | — | 5649.1 | mA | VDD = 1.08V |
| Supply current (3.3V I/O) | IDDQ33 | — | — | 92.6 | mA | VDDQ33 = 3.6V |
| Supply current (3.3V I/O[QSPI]) | IDDQ_QSPI | — | — | 21.6 | mA | VDDQ_QSPI = 3.6V |
| Supply current (3.3V I/O[SDHI]) | IDDQ_SDn (n = 0, 1, 3) | — | — | 5.4 | mA | VDDQ_SDn (n = 0, 1, 3) = 3.6V, Highspeed mode |
| Supply current (3.3V I/O[AVB0]) | IDDQ25_AVB0 | — | — | 4.5 | mA | VDDQ25_AVB0 = 3.6V, 100Mbps |
| Supply current (3.3V I/O[USB2]) | IDDQ33_USB2 | — | — | 60 (30/ch) | mA | VDDQ33_USB2 = 3.6V |
| Supply current (2.5V I/O[AVB0]) | IDDQ25_AVB0 | — | — | 24.4 | mA | VDDQ25_AVB0 = 2.7V, 1Gbps |
| Supply current (1.8V I/O) | IDDQ18 | — | — | 25.2 | mA | VDDQ18 = 1.9V |
| Supply current (1.8V I/O[ISO]) | IDDQ18_ISO | — | — | 2 | mA | VDDQ18_ISO = 1.9V |
| Supply current (1.8V I/O[QSPI]) | IDDQ_QSPI | — | — | 25.7 | mA | VDDQ_QSPI = 1.9V |
| Supply current (1.8V I/O[SDHI]) | IDDQ_SDn (n = 0, 1, 3) | — | — | 11.4 | mA | VDDQ_SDn (n = 0, 1, 3) = 1.9V, SDR104 |
| Supply current (1.8V I/O[SDHI]) | IDDQ_SD3 | — | — | 67.8 | mA | VDDQ_SD3 = 1.9V, HS400 |
| Supply current (PLL[CPG]) | IDDQ18_CPGPLL0 | — | — | 4 | mA | VDD18_CPGPLL0 = 1.9V |
| Supply current (PLL[CPG]) | IDDQ18_CPGPLLn (n = 1, 3) | — | — | 2 | mA | VDD18_CPGPLLn (n = 1, 3) = 1.9V |
| Supply current (1.8V I/O[LVDS]) | IDDQ18_LVDS | — | — | 100 | mA | VDDQ18_LVDS = 1.9V |
| Supply current (PLL[LVDS]) | LVDSn_PLL_ICC (n = 0, 1) | — | — | 2 | mA | VDD18_LVDSnPLL (n = 0, 1) = 1.9V |
| Supply current (1.8V I/O[CSI]) | IDDQ18_CSI0 | — | — | 13 | mA | VDDQ18_CSI0 = 1.9V |
| Supply current (1.8V[USB2]) | IDD_AVDD0+IDDQ18_USB20, | — | — | 80, (40/ch) | mA | AVDD_USB = VDDQ18_USB20 = 1.9V |
| Supply current (1.8V I/O[USB30]) | IDDQ18_USB30 | — | — | 13 | mA | VDDQ18_USB30 = 1.9V |
| Supply current (1.8V I/O[PCIE0]) | IDDQ18_PCIE0 | — | — | 13 | mA | VDDQ18_PCIE0 = 1.9V |
| Supply current (PLL[MAPLL]) | IDDQ_MAPLL | — | — | 26.6 | mA | VDDQ_MAPLL = 1.9V |
| Supply current (PLL[MDPLL]) | IDDQ_MDPLLn (n = 0, 1) | — | — | 26.6 | mA | VDDQ_MDPLLn (n = 0, 1) = 1.9V |
| Supply current (PLL[MLB]) | IDDQ18_MLBPLL | — | — | 4 | mA | VDD18_MLBPLL = 1.9V |

| Item | Symbol | Power | | | Unit | Remarks | |
|--------------------------------------|-------------------------------|-----------------------|------|------|------|---------|--|
| | | Min. | Typ. | Max. | | | |
| Supply current (1.35V [DDR3L]) | Normal | IDDQ_CK + IDDQ_DDR | — | — | 959 | mA | VDDQ_CK = VDDQ_DDR = 1.45V (total value) |
| | DDR power supply backup | | — | — | 2.9 | mA | VDDQ_CK = VDDQ_DDR=1.45 V (total value) |
| | Normal | IDDQ_CK | — | — | 63.0 | mA | VDDQ_CK = 1.45V |
| Supply current (1.0V [PCIE]) | | IDDD_PCIE0 | — | — | 110 | mA | VDDD_PCIE0 = 1.08V |
| Supply current (1.0V [USB30]) | | IDDD_USB30 | — | — | 110 | mA | VDDD_USB30 = 1.08V |

Table 73.4.3 DC Characteristics (3.3-V I/O)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|------------------------|--------|------|------|--------------|------|---------------------------------------|--------------------|
| Input high voltage | VIH | 2.0 | — | VDDQ33 + 0.3 | V | VDDQ33 = 3.1 to 3.5 V | |
| Input low voltage | VIL | -0.3 | — | 0.8 | V | | |
| Output high voltage | VOH | 2.4 | — | — | V | VDDQ33 = 3.1 to 3.5 V | IOH = -4 mA |
| Output low voltage | VOL | — | — | 0.4 | V | | IOL = 4 mA |
| Pin capacitance | CL | — | — | 10 22 | pF | — | All pins |
| Input leakage current | ILI | — | — | 10 | μA | VDDQ33 = 3.1 to 3.5 V | All input pins |
| Output leakage current | ILO | — | — | 10 | μA | Without pull-up or pull-down resistor | Hi-Z output |
| Pull-up current | IPU | 40 | — | 115 | μA | VDDQ33 = 3.1 to 3.5 V | Vin = VSS |
| Pull-down current | IPD | 40 | — | 115 | μA | — | Vin = 3.1 to 3.5 V |

Table 73.4.4 DC Characteristics (3.3-V I/O)

| | |
|-------------|---------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|------------------------|--------|------|------|--------------|------|---------------------------------------|--------------------|
| Input high voltage | VIH | 2.0 | — | VDDQ33 + 0.3 | V | VDDQ33 = 3.0 to 3.6 V | |
| Input low voltage | VIL | -0.3 | — | 0.8 | V | | |
| Output high voltage | VOH | 2.4 | — | VDDQ33 + 0.3 | V | VDDQ33 = 3.0 to 3.6 V | IOH = -4 mA |
| Output low voltage | VOL | -0.3 | — | 0.4 | V | | IOL = 4 mA |
| Pin capacitance | CL | — | — | 10 | pF | — | All pins |
| Input leakage current | ILI | — | — | 1 | μA | VDDQ33 = 3.0 to 3.6 V | All input pins |
| Output leakage current | ILO | — | — | 1 | μA | Without pull-up or pull-down resistor | Hi-Z output |
| Pull-up current | IPU | -34 | — | -360 | μA | VDDQ33 = 3.0 to 3.6 V | Vin = VSS |
| Pull-down current | IPD | 34 | — | 360 | μA | — | Vin = 3.0 to 3.6 V |

Table 73.4.5 DC Characteristics (3.3-V I/O [SDHI])

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|------------------------|--------|-------------------|------|-------------------|------|---|--------------------------------|
| Input high voltage | VIH | 0.625 × VDDQVA_SD | — | VDDQ_SD + 0.3 | V | VDDQVA_SDn (n = 0 to 3) = 3.1 to 3.5 V, VSS = 0 V | — |
| Input low voltage | VIL | VSS – 0.3 | — | 0.25 × VDDQVA_SD | V | | — |
| Output high voltage | VOH | 0.75 × VDDQVA_SD | — | — | V | | IOH = -2 mA VDDQVA_S Dn min |
| Output low voltage | VOL | — | — | 0.125 × VDDQVA_SD | V | | IOL = 2 mA VDDQVA_S Dn min |
| Pin capacitance | CL | — | — | 10 | pF | | |
| Input leakage current | ILI | — | — | 10 | μA | VDDQVA_SDn (n = 0 to 3) = 3.1 to 3.5 V, VSS = 0 V | — |
| Output leakage current | ILO | — | — | 10 | μA | Without pull-up or pull-down resistor | Hi-Z output |

Table 73.4.6 DC Characteristics (3.3-V I/O [AVB0])

| | |
|-------------|---------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|-----------------------------|--------|------|------|--------------|------|---------------------------------------|--------------------|
| Input high voltage | VIH | 2.0 | — | VDDQ33 + 0.3 | V | VDDQ33 = 3.0 to 3.6 V | |
| Input low voltage | VIL | -0.3 | — | 0.8 | V | | |
| Output high voltage | VOH | 2.4 | — | VDDQ33 + 0.3 | V | VDDQ33 = 3.0 to 3.6 V | IOH = -4 mA |
| Output low voltage | VOL | -0.3 | — | 0.4 | V | | IOL = 4 mA |
| Pin capacitance | CL | — | — | 10 | pF | — | All pins |
| Input leakage current high | ILIH | — | — | 10 | μA | VDDQ33 = 3.0 to 3.6 V | All input pins |
| Input leakage current low | ILIL | -10 | — | — | μA | Without pull-up or pull-down resistor | |
| Output leakage current high | ILOH | — | — | 10 | μA | | Hi-Z output |
| Output leakage current low | ILOL | -10 | — | — | μA | | |
| Pull-up current | IPU | -34 | — | -360 | μA | VDDQ33 = 3.0 to 3.6 V | Vin = VSS |
| Pull-down current | IPD | 34 | — | 360 | μA | — | Vin = 3.0 to 3.6 V |

Table 73.4.7 DC Characteristics (2.5-V I/O [AVB])

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|-----------------------------|---------|------|------|--------------------|------|--|-----------------------|
| Input high voltage | VIH | 1.7 | — | VDDQ25_ETH+ 0.3 | V | VDDQ25_ETH = 2.4 to 2.6V | |
| Input low voltage | VIL | -0.3 | — | 0.7 | V | | |
| Output high voltage | VOH | 2.0 | — | — | V | VDDQ25_ETH = 2.4 to 2.6V | IOH= -1 mA |
| Output low voltage | VOL | — | — | 0.4 | V | | IOL = 1 mA |
| Pin capacitance | CL | — | — | 10 | pF | | |
| Input leakage current high | ILIH | — | — | 15 | μA | VDDQ25_ETH = 2.6 V, Vin = 2.5 V *1 | |
| Input leakage current low | ILIL | -15 | — | — | μA | VDDQ25_ETH = 2.6 V, Vin = 0.4 V *1 | |
| Output leakage current high | ILOH *1 | — | — | 10 | μA | — | Hi-Z output*1 |
| Output leakage current low | ILOL *1 | -10 | — | — | μA | — | Hi-Z output*1 |
| Pull-up current | IPU | 30 | — | 85 | μA | VDDQ25_ETH = 2.4 to 2.6 V | Vin = VSS |
| Pull-down current | IPD | 30 | — | 85 | μA | — | Vin = 2.4 to 2.6 V |

Note: *1 Without pull-up or pull-down resistor.

Table 73.4.8 DC Characteristics (2.5-V I/O [AVB0])

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|-----------------------------|--------|------|------|-------------------|------|------------------------------|----------------------|
| Input high voltage | VIH | 1.7 | — | VDDQ25_AVB0 + 0.3 | V | VDDQ25_AVB0 = 2.3 V to 2.7 V | |
| Input low voltage | VIL | -0.3 | — | 0.7 | V | | |
| Output high voltage | VOH | 2.0 | — | — | V | VDDQ25_AVB0 = 2.3 V to 2.7 V | IOH = -1 mA |
| Output low voltage | VOL | — | — | 0.4 | V | | IOL = 1 mA |
| Input leakage current high | ILIH | — | — | 10 | μA | — | |
| Input leakage current low | ILIL | -10 | — | — | μA | — | |
| Output leakage current high | ILOH | — | — | 10 | μA | — | |
| Output leakage current low | ILOL | -10 | — | — | μA | — | |
| Pull-up current | IPU | -26 | — | -270 | μA | VDDQ25_AVB0 = 2.3 V to 2.7 V | Vin = VSS |
| Pull-down current | IPD | 26 | — | 270 | μA | — | Vin = 2.3 V to 2.7 V |

Table 73.4.9 DC Characteristics
(1.8-V I/O [SDHI (SDR50, SDR104)])

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|------------------------|--------|-----------|------|---------------------|------|---|-------------|
| Input high voltage | VIH | 1.27 | — | VDDQVA_ SD + 0.3 | V | VSS = 0 V | — |
| Input low voltage | VIL | VSS – 0.3 | — | 0.58 | V | | — |
| Output high voltage | VOH | 1.4 | — | — | V | | IOH = -2 mA |
| Output low voltage | VOL | — | — | 0.45 | V | | IOL = 2 mA |
| Pin capacitance | CL | — | — | 10 | pF | | — |
| Input leakage current | ILI | — | — | 10 | μA | VSS = 0 V | — |
| Output leakage current | ILO | — | — | 10 | μA | Without pull-up or pull-down resistor | Hi-Z output |

Table 73.4.10 DC Characteristics (1.8-V I/O [MMC, SD])

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|------------------------|--------|-----------|------|-------------------|------|---|----------------------|
| Input high voltage | VIH | 1.27 | — | VDDQ_SDn + 0.3 | V | VDDQ_SDn (n = 0, 1, 3) = 1.7 to 1.9 V | — |
| Input low voltage | VIL | VSS – 0.3 | — | 0.58 | V | VSS = 0 V | — |
| Output high voltage | VOH | 1.4 | — | — | V | | IOH = -2 mA |
| Output low voltage | VOL | — | — | 0.45 | V | | IOL = 2 mA |
| Pin capacitance | CL | — | — | 10 | pF | | — |
| Pull-up current | IPU | -19 | — | -195 | μA | | Vin = VSS |
| Pull-down current | IPD | 19 | — | 195 | μA | | Vin = 1.7 to 1.9V |
| Input leakage current | ILI | — | — | 1 | μA | VSS = 0 V | — |
| Output leakage current | ILO | — | — | 1 | μA | Without pull-up or pull-down resistor | Hi-Z output |

Table 73.4.11 DC Characteristics (1.8-V I/O)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|-----------------------------------|--------|---------------|------|---------------|------|---------------------------------------|---------------------------|
| Schmitt input high voltage | VT+ | — | — | VDDQ18 × 0.7 | V | VDDQ18 = 1.7 to 1.9 V | PRESET#, BSMODE, NMI pins |
| Schmitt input low voltage | VT- | VDDQ18 × 0.3 | — | — | V | | |
| Input high voltage | VIH | VDDQ18 × 0.8 | — | VDDQ18 + 0.3 | V | — | EXTAL, USB_EXTAL pins |
| Input low voltage | VIL | -0.3 | — | VDDQ18 × 0.2 | V | | |
| Input high voltage (1.8-V LVCMOS) | VIH | 0.65 × VDDQ18 | — | VDDQ18 + 0.3 | V | VDDQ18 = 1.7 to 1.9 V | Other pins |
| Input low voltage (1.8-V LVCMOS) | VIL | -0.3 | — | 0.35 × VDDQ18 | V | | |
| Output high voltage | VOH | 0.7 × VDDQ18 | — | — | V | VDDQ18 = 1.7 to 1.9 V | IOH = - 4 mA |
| Output low voltage | VOL | — | — | 0.4 | V | | IOL = 4 mA |
| Pin capacitance | CL | — | — | 10 | pF | — | |
| Input leakage current | ILI | — | — | 10 | μA | VDDQ18 = 1.7 to 1.9 V | All input pins |
| Output leakage current | ILO | — | — | 10 | μA | Without pull-up or pull-down resistor | Hi-Z output |
| Pull-up current | IPU | 20 | — | 65 | μA | VDDQ18 = 1.7 to 1.9 V | Vin = VSS |
| Pull-down current | IPD | 20 | — | 65 | μA | — | Vin = 1.7 to 1.9 V |

Table 73.4.12 DC Characteristics (1.8-V I/O)

| | | |
|--------|-------------|-------------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E | |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|-----------------------------------|--------------|---------------|------|---------------|------|---------------------------------------|--|
| Schmitt input high voltage | VT+ | VDDQ18 × 0.4 | — | VDDQ18 × 0.7 | V | VDDQ18 = 1.7 to 1.9 V | PRESET#, BSMODE, NMI pins |
| Schmitt input low voltage | VT- | VDDQ18 × 0.3 | — | VDDQ18 × 0.6 | V | | $\Delta V = V_{out} - V_{in} \leq 0.1$ [V], Vout: IO voltage of external devices |
| Hysteresis voltage | ΔV_T | VDDQ18 × 0.1 | — | VDDQ18 × 0.4 | V | | Vin: VDDQ18 |
| Input high voltage | VIH | VDDQ18 × 0.8 | — | VDDQ18 + 0.3 | V | VDDQ18 = 1.7 to 1.9 V | EXTAL pin |
| Input low voltage | VIL | -0.3 | — | VDDQ18 × 0.2 | V | | |
| Input high voltage (1.8-V LVCMOS) | VIH | 0.65 × VDDQ18 | — | VDDQ18 + 0.3 | V | VDDQ18 = 1.7 to 1.9 V | Other pins |
| Input low voltage (1.8-V LVCMOS) | VIL | -0.3 | — | 0.35 × VDDQ18 | V | | |
| Output high voltage | VOH | 0.7 × VDDQ18 | — | VDDQ18 + 0.3 | V | VDDQ18 = 1.7 to 1.9 V | IOH = -4 mA |
| Output low voltage | VOL | -0.3 | — | 0.4 | V | | IOL = 4 mA |
| Pin capacitance | CL | — | — | 10 | pF | — | — |
| Input leakage current | ILI | — | — | 1 | μA | VDDQ18 = 1.7 to 1.9 V | All input pins |
| Output leakage current | ILO | — | — | 1 | μA | Without pull-up or pull-down resistor | Hi-Z output |
| Pull-up current | IPU | -19 | — | -195 | μA | VDDQ_DU = VDDQ_VIN01 = 1.7 to 1.9 V | Vin = VSS |
| Pull-down current | IPD | 19 | — | 195 | μA | — | Vin = 1.7 to 1.9 V |
| Pull-up current | IPU | -5 | — | -39 | μA | VDDQ18 = 1.7 to 1.9 V | All pins Vin = VSS |
| Pull-down current | IPD | 5 | — | 39 | μA | — | Other than PRESET# pin Vin = 1.7 to 1.9 V |
| PRESET# Pull-down current | IPD PRST | 12 | — | 30 | μA | VDDQ18 = 1.7 to 1.9 V | PRESET# pin Vin = 1.7 to 1.9 V |
| PRESET# Pull-down resistor | RPD PRST | 65 | — | 135 | kΩ | — | PRESET# pin |

Table 73.4.13 DC Characteristics (I2C open-drain 1.8-V I/O)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|--------------------------|--------|-----------------|------|-----------------|------|--------------------------|--|
| External pull-up voltage | VPU18 | 1.7 | 1.8 | 1.9 | V | — | I2Cn_SCL/S DA (I2C open-drain assigned channels) * |
| | VPU33 | 3.1 | 3.3 | 3.5 | V | | |
| Input high voltage | VIH | VDDQ18 x 0.7 | — | VPU18 + 0.3 | V | VDDQ18 = 1.7 to 1.9 V | HDMI0_SCL /SDA |
| | | VPU33 x 0.7 | — | VPU33 + 0.3 | V | | |
| Input low voltage | VIL | -0.3 | — | VDDQ18 x 0.3 | V | | I2C_PMIC_ SCL/SDA (only RZ/G2E) |
| Output low voltage | VOL | — | — | VDDQ18 x 0.2 | V | VDDQ18 = 1.7 to 1.9 V | IOL= 2mA |
| | | | | 0.4 | V | | IOL= 3mA |
| Output low current | IOL | 3 | — | — | mA | | VOL= 0.4V |

Notes: 1. Refer to section 52.1.2 in the RZ/G Gen2_HW_Users_Manual.

2. When using 1.8V I2C pins as 3.3 V tolerant, all the VPU33 power supply for the I2C of this LSI must keep the same power on/off sequence as the VDDQ33 of this LSI.

Table 73.4.14 DC Characteristics (I2C 3.3-V I/O)

| | |
|-------------|---------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|------------------------|--------|------|------|---------------------|------|--|----------------|
| Input high voltage | VIH | 2.0 | — | VDDQ_VIN01 + 0.3 | V | VDDQ_VIN01, VDDQ33 = VDDQ_SD0 = 3.0 to 3.6 V | |
| Input low voltage | VIL | -0.3 | — | 0.8 | V | | |
| Output low voltage | VOL | -0.3 | — | 0.4 | V | | IOL = 4 mA |
| Pin capacitance | CL | — | — | 10 | pF | — | All pins |
| Input leakage current | ILI | — | — | 1 | μA | VDDQ_VIN01, VDDQ33 = | All input pins |
| Output leakage current | ILO | — | — | 1 | μA | VDDQ_SD0 = 3.0 to 3.6 V Without pull-up or pull-down resistor | Hi-Z output |

Table 73.4.15 DC Characteristics (1.8-V I/O [CSI2])

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement condition | Remarks |
|-----------------------------------|-------------------|-------|------|------|----------|---|-------------|
| Input signal voltage range | V_{PIN} | -50 | — | 1350 | mV | | |
| Input leakage current | I_{LEAK} | -10 | — | 10 | uA | $V_{GNDSH(min)} \leq V_{PIN} \leq V_{GNDSH(max)} + V_{PIN(absmax)}$ Lane module in LP receive mode | |
| Ground shift | V_{GNDSH} | -50 | — | 50 | mV | | |
| transient pin voltage level | $V_{PIN(absmax)}$ | -0.15 | — | 1.45 | V | | |
| Differential input high threshold | V_{IDTH} | — | — | 70 | mV | | HS Receiver |
| Differential input low threshold | V_{IDTL} | -70 | — | — | mV | | HS Receiver |
| Single-ended input high voltage | V_{IHHS} | — | — | 460 | mV | | HS Receiver |
| Single-ended input low voltage | V_{ILHS} | -40 | — | — | mV | | HS Receiver |
| Input common mode voltage | $V_{CMRX(DC)}$ | 70 | — | 330 | mV | | HS Receiver |
| Differential input impedance | Z_{ID} | 80 | — | 125 | Ω | | HS Receiver |
| Input low voltage | V_{IL} | — | — | 550 | mV | | LP Receiver |
| Input high voltage | V_{IH} | 880 | — | — | mV | | LP Receiver |
| Input hysteresis | V_{HYST} | 25 | — | — | mV | | LP Receiver |

Table 73.4.16 DC Characteristics (1.8-V I/O [LVDS])

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement condition | Remarks |
|---------------------------------|--------|-------|------|-------|------|--------------------------|-------------|
| Output differential voltage | VOD | 250 | — | 450 | mV | Termination load = 100 Ω | TIA/EIA-644 |
| Output offset voltage | VOS | 1.125 | — | 1.375 | V | Termination load = 100 Ω | TIA/EIA-644 |
| Change in VOD between 0 and 1 | ΔVOD | — | — | 50 | mV | Termination load = 100 Ω | TIA/EIA-644 |
| Change in VOS between 0 and 1 | ΔVOS | — | — | 50 | mV | Termination load = 100 Ω | TIA/EIA-644 |

Table 73.4.17 DC Characteristics (Operating Conditions for HDMI Interface)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|----------------------------|----------------|------|------|------|------|------------------------|---|
| Termination supply Voltage | AVcc | 3.15 | 3.3 | 3.45 | V | | Minimum supply rise time for RX termination voltage > 2.5 μs Figure 73.4.1 |
| Termination resistance | R _T | 45 | 50 | 55 | Ω | | Figure 73.4.1 |

Table 73.4.18 DC Characteristics (Source DC characteristics at TP1 for HDMI Interface)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|--|-------------|---------------|------|---------------|------|-------------------------|---------------|
| Single-ended standby output voltage | V_{OFF} | $AV_{CC}-10$ | — | $AV_{CC}+10$ | mV | | Figure 73.4.2 |
| Single-ended output swing voltage | V_{SWING} | 400 | — | 600 | mV | | Figure 73.4.2 |
| Single-ended high level output voltage | V_H | $AV_{CC}-10$ | — | $AV_{CC}+10$ | mV | TMDSCCLK \leq 165 MHz | Figure 73.4.2 |
| | | $AV_{CC}-200$ | — | $AV_{CC}+10$ | mV | TMDSCCLK $>$ 165 MHz | |
| Single-ended low level output voltage | V_L | $AV_{CC}-600$ | — | $AV_{CC}-400$ | mV | TMDSCCLK \leq 165 MHz | Figure 73.4.2 |
| | | $AV_{CC}-700$ | — | $AV_{CC}-400$ | mV | TMDSCCLK $>$ 165 MHz | |

Table 73.4.19 DC Characteristics (Hot Plug Detect Signal for HDMI)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|--------------------|-------------------|------|------|------|------|------------------------|--|
| High voltage level | HPD _{VH} | 2.0 | — | 5.3 | V | | Minimum rise time for HPD voltage > 4 μs |
| Low voltage level | HPD _{VL} | 0 | — | 0.8 | V | | |

Please confirm HPD section of HDMI specification. (Note that many Sink device simply connect the HPD signal to the +5V Power signal through a 1000 Ω resistor.)

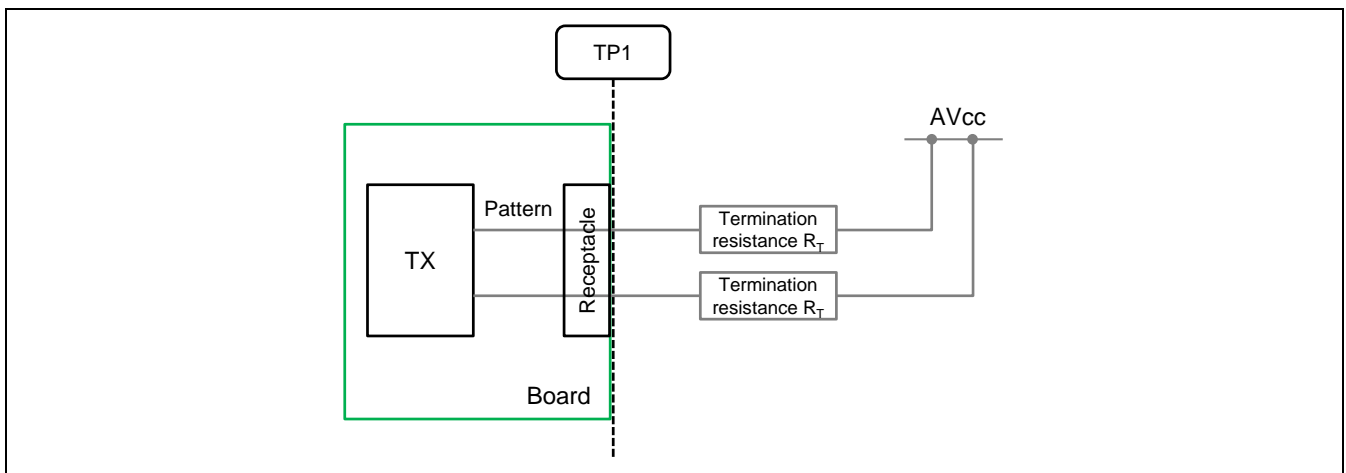


Figure 73.4.1 Balanced Source Test Load for HDMI

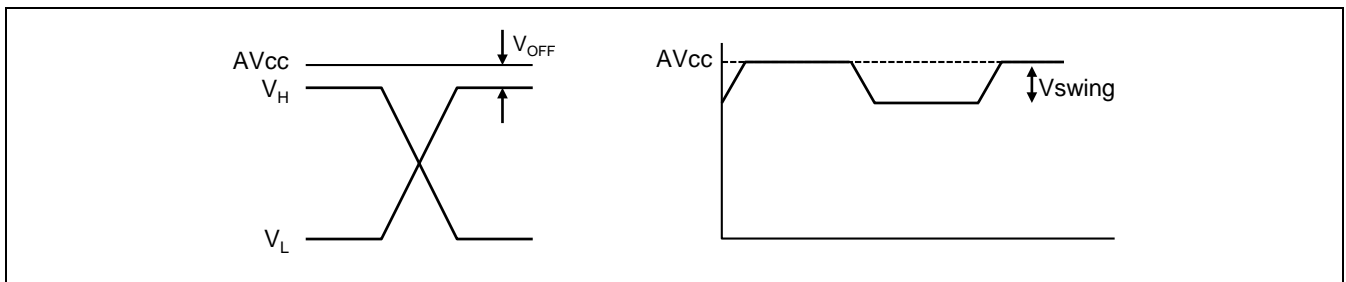


Figure 73.4.2 Single-ended Differential Signal for HDMI

Table 73.4.20 DC Characteristics (1.35-V I/O [DDR3L])

| | |
|-------------|---------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|--|-----------|-----------------------|-----------------|-----------------------|------|--|--------------------------------------|
| Input high voltage (MDQ pin) | VIH | VREF + 0.09 | — | — | V | VDDQ_DDR = VDDQ_CK = 1.283 to 1.450 V | MDQ, and MDQS pins |
| Input low voltage (MDQ pin) | VIL | — | — | VREF - 0.09 | V | | |
| Input high voltage | VIH | 0.7 × VDDQ_DDR | — | — | V | VDDQ_DDR = VDDQ_CK = 1.283 to 1.450 V | MBKPRST# pin |
| Input low voltage | VIL | — | — | 0.3 × VDDQ_DDR | V | | |
| Differential input reference voltage | VREF | 0.49 × VDDQ_DDR | 0.50 × VDDQ_DDR | 0.51 × VDDQ_DDR | V | VDDQ_DDR = VDDQ_CK = 1.283 to 1.450 V | *1 |
| DC differential input high voltage | VIHD | 0.18 | — | — | V | MDQS = H, VIN = VDDQ_DDR/2, VDD = 1.0 V _{typ} , VDDQ_DDR = VDDQ_CK = 1.283 to 1.450 V | MDQS pins |
| DC differential input low voltage | VILD | — | — | -0.18 | V | MDQS = L, VIN = VDDQ_DDR/2, VDD = 1.0 V _{typ} , VDDQ_DDR = VDDQ_CK = 1.283 to 1.450 V | |
| AC differential input high voltage | VIHD (AC) | 0.27 | — | — | V | VDD = 1.0 V _{typ} , VDDQ_DDR = VDDQ_CK = 1.283 to 1.450 V | MDQS pins |
| AC differential input low voltage | VILD (AC) | — | — | -0.27 | V | VDD = 1.0 V _{typ} , VDDQ_DDR = VDDQ_CK = 1.283 to 1.450 V | MDQS pins |
| Input high voltage | VIH (AC) | VREF + 0.135 | — | — | V | — | |
| Input low voltage | VIL (AC) | — | — | VREF - 0.135 | V | — | |
| AC differential input cross point voltage | VIX (AC) | 0.5 × VDDQ_DDR - 0.15 | — | 0.5 × VDDQ_DDR + 0.15 | V | VDDQ_DDR = VDDQ_CK = 1.283 to 1.450 V | MDQS pins *2 |
| AC differential output cross point voltage | VOX (AC) | VREF - 0.125 | — | VREF + 0.125 | V | VDDQ_DDR = VDDQ_CK = 1.450 V | MCK, and MDQS pins (PU, PD not used) |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|------------------------|----------------|------|------|------|------|------------------------------------|-------------------------------------|
| High Hi-Z leak current | IOZH | — | — | 4 | μA | VDDQ_DDR = VDDQ_CK = 1.450 V | Other than MZQ and MVREF pins |
| Low Hi-Z leak current | IOZL | -4 | — | — | μA | | |
| High Hi-Z leak current | IOZH | — | — | 5 | μA | | MZQ pin |
| Low Hi-Z leak current | IOZL | -5 | — | — | μA | | |
| High Hi-Z leak current | IOZH | — | — | 25 | μA | | MVREF pin |
| Low Hi-Z leak current | IOZL | -25 | — | — | μA | | |
| Pin capacitance | CL | — | — | 15 | pF | — | All pins* ³ |
| PLL power supply | * ⁴ | 1.7 | 1.8 | 1.9 | V | — | — |

- Notes: 1. Peak to peak ac noise on VREF may not exceed ± 2 % of VREF.
 2. The VIX (AC) indicates the voltage at which differential input signals cross each other. The typical value of VIX (AC) is expected to be 0.5 × VDDQ_DDR.
 3. Except power supply pins.
 4. VDDQ_MAPLL, VDDQ_MDPLL0, VDDQ_MDPLL1 pins.

Table 73.4.21 DDR3L Interface ODT Characteristics

| | |
|-------------|---------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|--------------------------|-----------------|----------------|------|----------------|------|---|---------|
| Termination voltage | V _{tt} | VREF – 0.04 | VREF | VREF + 0.04 | V | VDD = 1.0 V _{typ} , VDDQ_DDR = VDDQ_CK = 1.283 to 1.450 V | — |
| ODT resistance (60 Ω) | RTT60 | 54.0 | — | 96.0 | Ω | | — |
| ODT resistance (40 Ω) | RTT40 | 36.0 | — | 64.0 | Ω | | — |
| VM deviation | ΔVM | -7 | — | +7 | % | | * |

- Note: * VM is a voltage value measured with the ODT turned on without any load applied to this LSI chip. ΔVM is obtained by the following formula:

$$\Delta VM = (2 \times VM / VDDQ_DDR - 1) \times 100$$

Table 73.4.22 DC Characteristics (1.1-V I/O [LPDDR4])

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|------------------------------------|--------|--------------------------------|------|--------------------------------|------|---|-------------------------|
| Input high voltage (MDQ pin) | VIH | VREF + 0.09 | — | — | V | VDDQVA_DDRn = 1.06 to 1.17 V (n = 0 to 1) | MDQ, and MDQS pins |
| Input low voltage (MDQ pin) | VIL | — | — | VREF – 0.09 | V | VREF = VDDQVA_DDRn*(1/6) | |
| Input high voltage | VIH | 0.8 × VDDQVA_DDRn (n = 0 to 1) | — | — | V | VDDQVA_DDRn = 1.06 to 1.17 V (n = 0 to 1) | MBKUP pin |
| Input low voltage | VIL | — | — | 0.2 × VDDQVA_DDRn (n = 0 to 1) | V | | |
| DC differential input high voltage | VIHD | 0.09 | — | — | V | MDQS = H, VDDQVA_DDRn = 1.06 to 1.17 V (n = 0 to 1) | MDQS pins |
| DC differential input low voltage | VILD | — | — | -0.09 | V | MDQS = L, VDDQVA_DDRn = 1.06 to 1.17 V (n = 0 to 1) | |
| Output high voltage | VOH | 0.99 | — | — | V | VDDQVA_DDRn = 1.06 to 1.17 V (n = 0 to 1) | MBKUP pins IOH = 0mA |
| Output low voltage | VOL | — | — | 0.16 | V | VDDQVA_DDRn = 1.06 to 1.17 V (n = 0 to 1) | MBKUP pins IOL = 0mA |
| High Hi-Z leak current | IOZH | — | — | 10 | μA | VDDQVA_DDRn = 1.06 to 1.17 V (n = 0 to 1) | Other than MZQ |
| Low Hi-Z leak current | IOZL | -10 | — | — | μA | | |
| High Hi-Z leak current | IOZH | — | — | 10 | μA | | MZQ pin |
| Low Hi-Z leak current | IOZL | -10 | — | — | μA | | |
| Pin capacitance | CL | — | — | 15 | pF | — | All pins* |

Note: * Except power supply pins.

Table 73.4.23 LPDDR4 Interface ODT Characteristics

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement conditions | Remarks |
|-----------------------|--------|------|------|------|------|---|---------|
| ODT resistance (40 Ω) | RTT40 | 36 | — | 44 | Ω | VDDQVA_DDRn = 1.06 to 1.17 V (n = 0 to 1) | — |

73.4.1 Overload Condition (Injection Current)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The overload condition describes the behavior in case of current injection to the port pins.
Condition: $T_j = -40^{\circ}\text{C}$ to $T_j \text{ Max}$

Table 73.4.24 Overload Current [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, and RZ/G2N)

| Parameter | Symbol | Condition | Ratings*1 | Unit | |
|--|---|--|--|---------|----|
| Overload Current *2 $V_{in} > V_{CC}$ | Pins supplied by VDDQ33 | IINJPM | 1 pin | ± 2 | mA |
| | | IINJNM | Sum of all absolute IINJPM + IINJNM of pins supplied as group *3 | 50 | mA |
| $V_{in} < V_{SS}$ | Pins supplied by VDDQ18 | IINJPM | 1 pin | ± 2 | mA |
| | | IINJNM | Sum of all absolute IINJPM + IINJNM of pins supplied as group *3 | 50 | mA |
| | Pins supplied by VDDQVA_SDn (n = 0,1,2) | IINJPM | 1 pin | ± 2 | mA |
| | | IINJNM | Sum of all absolute IINJPM + IINJNM of pins supplied as group *3 | 12 | mA |
| | Pins supplied by VDDQVA_SDn (n = 3) | IINJPM | 1 pin | ± 2 | mA |
| | | IINJNM | Sum of all absolute IINJPM + IINJNM of pins supplied as group *3 | 22 | mA |
| Pins supplied by VDDQ25_ETH | IINJPM | 1 pin | ± 2 | mA | |
| | IINJNM | Sum of all absolute IINJPM + IINJNM of pins supplied as group *3 | 36 | mA | |

- Notes: 1. The Total current may be limited further by the total power dissipation.
2. Be sure not to exceed the absolute maximum ratings (Max value) of each supply voltage.
3. The total overload current must be within the output current.

Table 73.4.25 Overload Current [RZ/G2E]

| Parameter | | Symbol | Condition | Ratings*1 | Unit |
|-------------------------------------|------------------------------|--------|--|-----------|------|
| Overload Current *2 | Pins supplied by VDDQ33 | IINJPM | 1 pin | ±2 | mA |
| | | IINJNM | Sum of all absolute IINJPM + IINJNM of pins supplied as group *3 | 50 | mA |
| Vin > VCC | Pins supplied by VDDQ18 | IINJPM | 1 pin | ±2 | mA |
| | | IINJNM | Sum of all absolute IINJPM + IINJNM of pins supplied as group *3 | 25.2 | mA |
| Vin < VSS | Pins supplied by VDDQ25_AVB0 | IINJPM | 1 pin | ±2 | mA |
| | | IINJNM | Sum of all absolute IINJPM + IINJNM of pins supplied as group *3 | 24.4 | mA |
| Pins supplied by VDDQ_SDn (n = 0,1) | Pins supplied by VDDQ_SD3 | IINJPM | 1 pin | ±2 | mA |
| | | IINJNM | Sum of all absolute IINJPM + IINJNM of pins supplied as group *3 | 5.4 | mA |
| Pins supplied by VDDQ_QSPI | Pins supplied by VDDQ_QSPI | IINJPM | 1 pin | ±2 | mA |
| | | IINJNM | Sum of all absolute IINJPM + IINJNM of pins supplied as group *3 | 22 | mA |
| | | IINJPM | 1 pin | ±2 | mA |
| | | IINJNM | Sum of all absolute IINJPM + IINJNM of pins supplied as group *3 | 21.6 | mA |

- Notes: 1. The Total current may be limited further by the total power dissipation.
2. Be sure not to exceed the absolute maximum ratings (Max value) of each supply voltage.
3. The total overload current must be within the output current.

73.5 Clock and Reset Timings

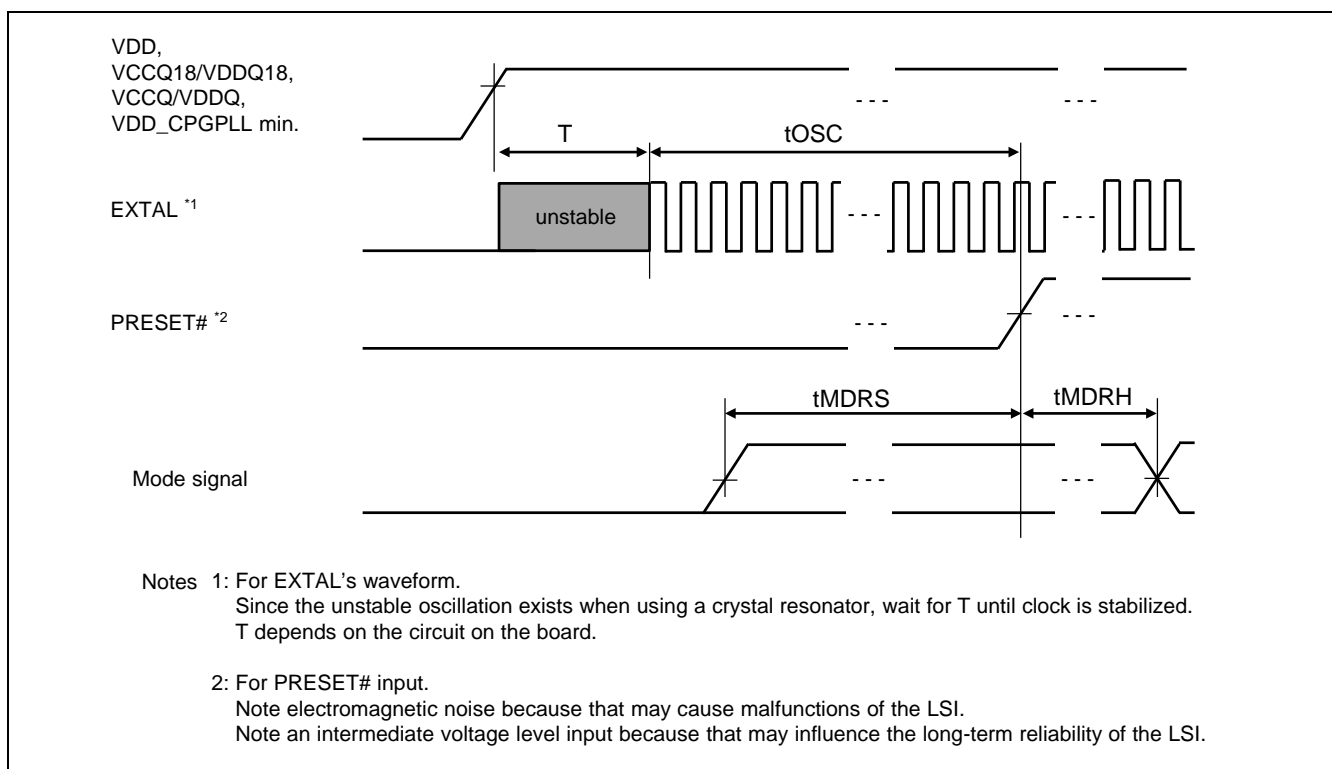
| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.5.1 Clock and Reset Timing

Conditions: VDDQ33 = 3.3 V ± 0.2 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] / ± 0.3 V [RZ/G2E],
 VDDQ18 = 1.8 V ± 0.1 V, GND = VSS = 0 V,
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3]
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Pin | Item | Symbol | Min. | Max. | Unit | Figures |
|-------------------|---|--------|------|------|------|---------------|
| PRESET#, EXTAL | Power-on oscillation settling time (Period from power and EXTAL stable clock input to PRESET# rise.) | tOSC | 5 | — | ms | Figure 73.5.1 |
| | | T | 0*2 | *3 | ms | |
| Mode signal*1 | MD reset setup time | tMDRS | 5 | — | ms | |
| Mode signal*1 | MD reset hold time | tMDRH | 3 | — | ns | |

- Notes: 1. MDn (n = 0, 1, 2 ...) and MDT [1:0]. For details of mode signals, refer to section 3, 4, 5 and 6, Mode Pin Settings.
 2: Using 0 stabilization time requires an external clock. For mode setting refer to Table 11.3 MD9 Settings.
 3: The oscillation stabilization time differs according the matching with the external resonator circuit. It is recommended to determine the oscillation stabilization time by a crystal resonator matching test. For details of the stabilization time "T", please contact the crystal manufacture.



- Notes 1: For EXTAL's waveform.
 Since the unstable oscillation exists when using a crystal resonator, wait for T until clock is stabilized. T depends on the circuit on the board.
 2: For PRESET# input.
 Note electromagnetic noise because that may cause malfunctions of the LSI.
 Note an intermediate voltage level input because that may influence the long-term reliability of the LSI.

Figure 73.5.1 Reset when Turning on Power Supply (PRESET# vs EXTAL and Mode signal)

73.6 EXTAL Clock Input/output Timing

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

73.6.1 EXTAL Clock Input Timing

Table 73.6.1 EXTAL Clock Input Timing

Conditions: VDDQ18 = 1.8 V ± 0.1 V, GND = VSS = 0 V, Tc = -40 to +115 °C[RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C[RZ/G2M V3.0, RZ/G2N], Tj = -40 to +115 °C[RZ/G2M V3.0, RZ/G2N]

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure | |
|--|----------------|--------|-------|-------|--------|--------|---------------|
| EXTAL clock input frequency | MD[14:13] = LL | fEX | 14.40 | 16.66 | 16.70 | MHz | — |
| | MD[14:13] = LH | | 17.30 | 20.00 | 20.04 | MHz | |
| | MD[14:13] = HL | | 21.62 | 25.00 | 25.06 | MHz | |
| | MD[14:13] = HH | | 28.80 | 33.33 | 33.40 | MHz | |
| EXTAL clock input cycle time | MD[14:13] = LL | tEXcyc | 59.88 | 60.02 | 69.44 | ns | Figure 73.6.1 |
| | MD[14:13] = LH | | 49.90 | 50.00 | 57.80 | ns | |
| | MD[14:13] = HL | | 36.90 | 40.00 | 46.25 | ns | |
| | MD[14:13] = HH | | 29.94 | 30.00 | 34.72 | ns | |
| EXTAL clock input duty cycle time | tEXduty | 0.4 | 0.5 | 0.6 | tEXcyc | | |
| EXTAL clock input low-level pulse width | tEXL | 5 | — | — | ns | | |
| EXTAL clock input high-level pulse width | tEXH | 5 | — | — | ns | | |
| EXTAL clock input rise time | tEXr | — | — | 4 | ns | | |
| EXTAL clock input fall time | tEXf | — | — | 4 | ns | | |

Note: Set the MD9 pin to L (low) during power-on reset when using the EXTAL pin as external clock input and the XTAL pin must be open.
 Not to exceed the specification of LPDDR4 operating frequency, input the lower frequency of EXATL or lower the multiplication rate of PLL.

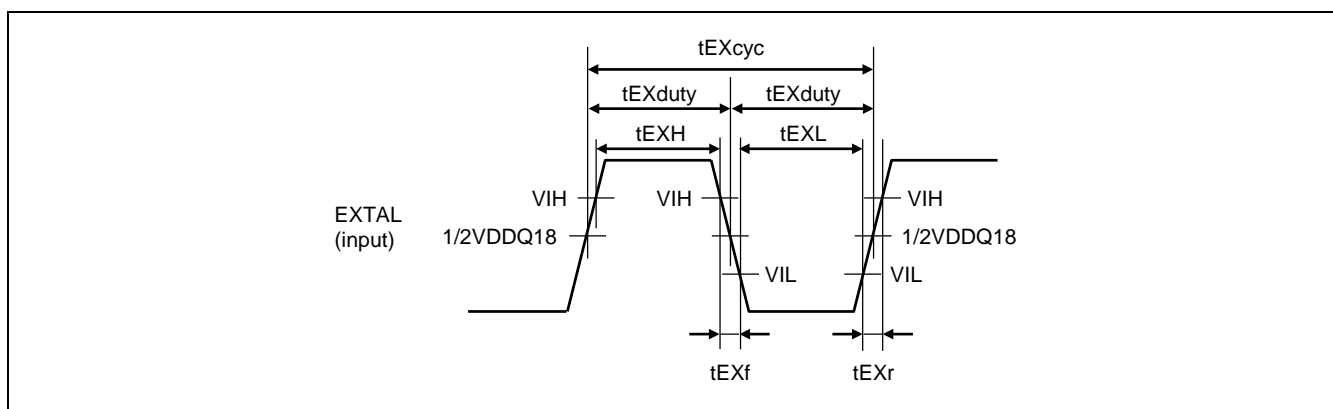


Figure 73.6.1 EXTAL Clock Input Timing

73.6.2 CLKOUT Clock Output Timing

Table 73.6.2 CLKOUT Clock Output Timing

Conditions: VDDQ18 = 1.8 V ± 0.1 V, GND = VSS = 0 V, Tc = -40 to +115 °C[RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C[RZ/G2M V3.0, RZ/G2N], Tj = -40 to +115 °C[RZ/G2M V3.0, RZ/G2N]

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure | |
|--------------------------------------|----------------------|---------------------|-------|-------|---------------------|--------|---------------|
| CLKOUT clock output frequency | MD18 = L | fCKO | 57.52 | 66.66 | 66.80 | MHz | — |
| | MD18 = H | | 38.35 | 44.44 | 44.53 | MHz | |
| CLKOUT clock output cycle time | MD18 = L | tCKO _{cyc} | 14.97 | 15.00 | 17.38 | ns | Figure 73.6.2 |
| | MD18 = H | | 22.45 | 22.50 | 26.07 | ns | |
| CLKOUT clock output duty cycle time | tCKO _{duty} | 0.4 | 0.5 | 0.6 | tCKO _{cyc} | | |
| CLKOUT clock output low pulse width | tCKOL | 4 | — | — | ns | | |
| CLKOUT clock output high pulse width | tCKOH | 4 | — | — | ns | | |
| CLKOUT clock output rise time | tCKOr | — | — | 4 | ns | | |
| CLKOUT clock output fall time | tCKOf | — | — | 4 | ns | | |

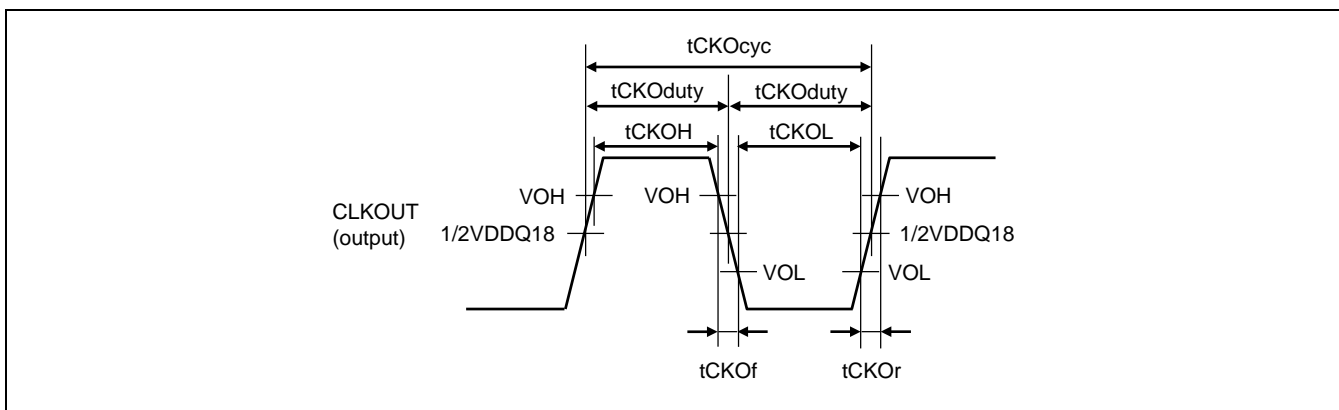


Figure 73.6.2 CLKOUT Clock Output Timing

73.7 EXTAL Clock Input/output Timing

| | |
|-------------|---------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

73.7.1 EXTAL Clock Input Timing

Table 73.7.1 EXTAL Clock Input Timing

Conditions: VDDQ18 = 1.8 V ± 0.1 V, GND = VSS = 0 V, Ta = -40 to +85 °C, Tj = -40 to +115 °C

| Item | Symbol | Min. | Typ. | Max. | Unit | Remarks | Figure |
|--|---------|------|-------|------|--------|---------------------------------------|--------|
| EXTAL clock input frequency | fEX | — | 48 | — | MHz | Frequency deviation: ±100 ppm or less | — |
| EXTAL clock input cycle time | tEXcyc | — | 20.83 | — | ns | | |
| EXTAL clock input duty cycle time | tEXduty | 0.4 | 0.5 | 0.6 | tEXcyc | | |
| EXTAL clock input low-level pulse width | tEXL | 5 | — | — | ns | | |
| EXTAL clock input high-level pulse width | tEXH | 5 | — | — | ns | | |
| EXTAL clock input rise time | tEXr | — | — | 4 | ns | | |
| EXTAL clock input fall time | tEXf | — | — | 4 | ns | | |

Note: Set the MD9 pin to L (low) during power-on reset when using the EXTAL pin as external clock input and the XTAL pin must be open.

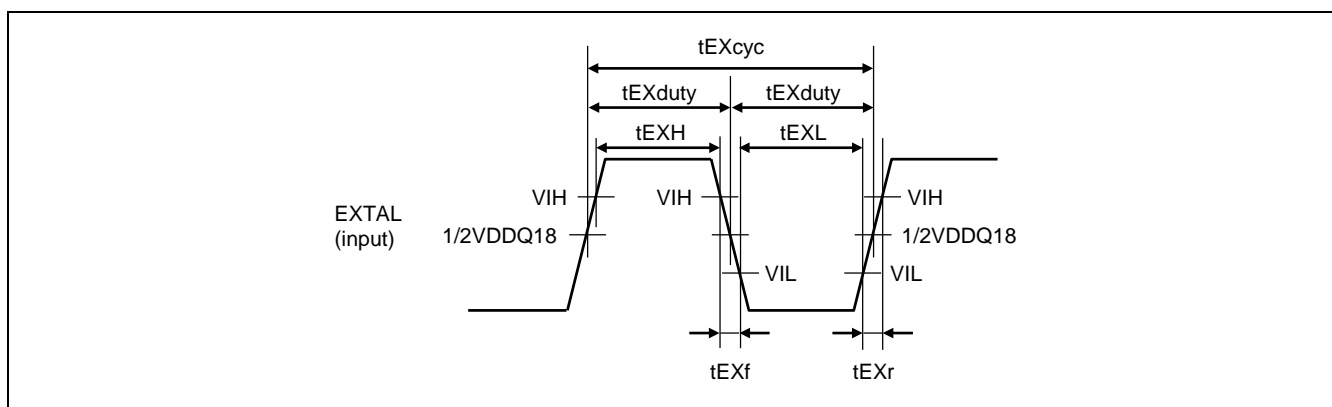


Figure 73.7.1 EXTAL Clock Input Timing

73.8 PLL Characteristics

| | |
|-------------|---------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.8.1 PLL1 SSCG Characteristics

Conditions: VDDQ18 = 1.8 V ± 0.1 V, GND = VSS = 0 V, Ta = -40 to +85 °C, Tj = -40 to +115 °C

| Item | Symbol | Min. | Typ. | Max. | Unit | Remarks |
|--|--------|-------|-------|-------|------|---------------|
| PLL1 modulation frequency | — | +4.02 | +4.03 | +4.04 | kHz | Figure 73.8.1 |
| PLL1 frequency dithering range (down-spread) | — | -3.0 | — | +0.0 | % | |

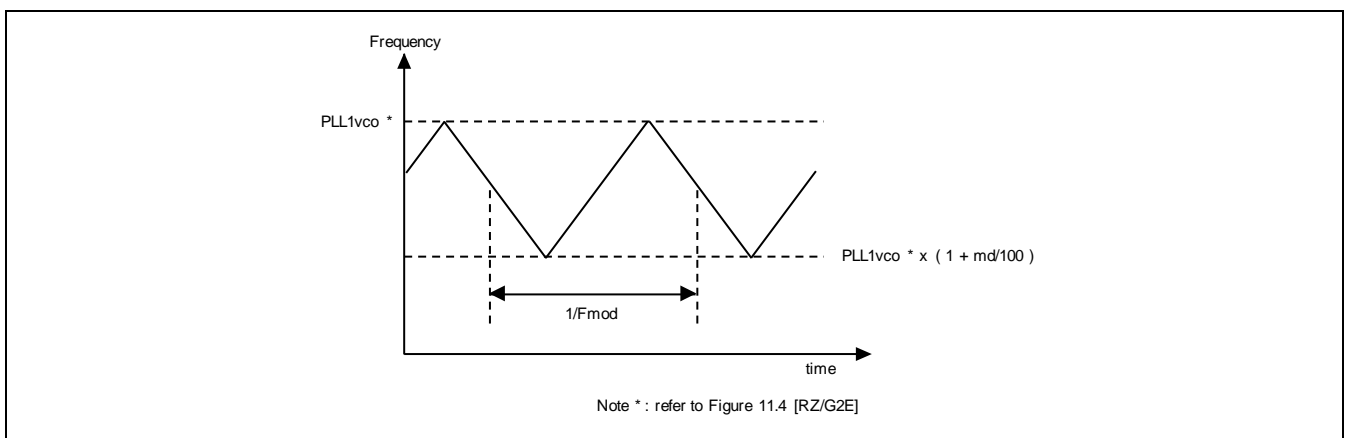


Figure 73.8.1 PLL1 SSCG Time trend of output clock

73.9 EXTALR Input Timing

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.9.1 EXTALR Characteristics for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N

Conditions: VDDQ18 = 1.8 V ± 0.1 V, GND = VSS = 0 V,
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3]
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N], Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N]

| Item | Symbol | Min. | Typ. | Max. | Unit | Remarks |
|---------------------------|--------|------|--------|------|------|---------|
| Input Frequency Range | — | 32.5 | 32.768 | 33 | kHz | |
| Input Frequency Tolerance | — | -200 | — | +200 | ppm | |

73.10 LBSC

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

73.10.1 Normal Read/Write, Burst ROM Read

Table 73.10.1 Normal Read/Write Access Timing [RZ/G2H/RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

Conditions: VDDQ33 = 3.3 ± 0.2 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] / ± 0.3 V [RZ/G2E],
 GND = VSSQ = 0 V, Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 CL = 40 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|---------------------------------|--------|------|------|------|------|----------------|
| Address output delay time | tDA | 0.0 | — | 6.0 | ns | Figure 73.10.1 |
| CS# output delay time | tDCS | 0.0 | — | 6.0 | ns | |
| BS# output delay time | tDBS | 0.0 | — | 6.0 | ns | |
| RD# output delay time | tDRD | 0.0 | — | 6.0 | ns | |
| RD/WR# output delay time | tDRW | 0.0 | — | 6.0 | ns | |
| Read data setup time | tSD | 11.0 | — | — | ns | |
| Read data hold time | tHD | 0.0 | — | — | ns | |
| WE# output delay time | tDWE | 0.0 | — | 6.0 | ns | |
| Write data output delay time | tDD | 0.0 | — | 6.0 | ns | |
| External wait signal setup time | tSEW | 11.0 | — | — | ns | |
| External wait signal hold time | tHEW | 0.0 | — | — | ns | |

Table 73.10.2 Burst ROM Read Access Timing

Conditions: VDDQ33 = 3.3 ± 0.2 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] / ± 0.3 V [RZ/G2E],
 GND = VSSQ = 0 V, Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +105 °C / -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], CL = 40 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|---------------------------|---------|------|------|------|------|----------------|
| Address output delay time | tDABST | 0.0 | — | 6.0 | ns | Figure 73.10.2 |
| CS# output delay time | tDCSBST | 0.0 | — | 6.0 | ns | |
| RD# output delay time | tDRDBST | 0.0 | — | 6.0 | ns | |
| Read data setup time | tSDBST | 11.0 | — | — | ns | |
| Read data hold time | tHDBST | 0.0 | — | — | ns | |

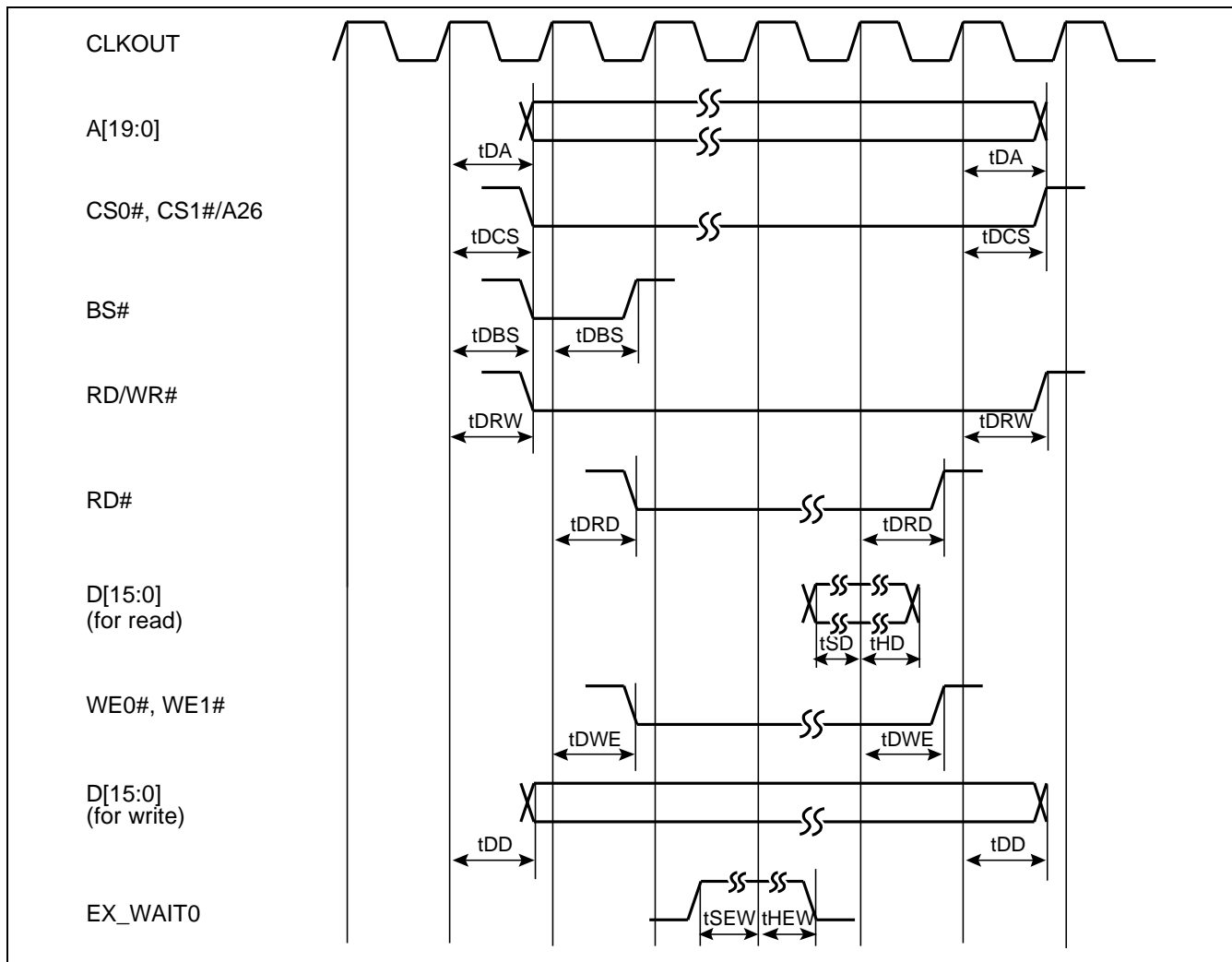


Figure 73.10.1 Normal Read/Write Access Timing

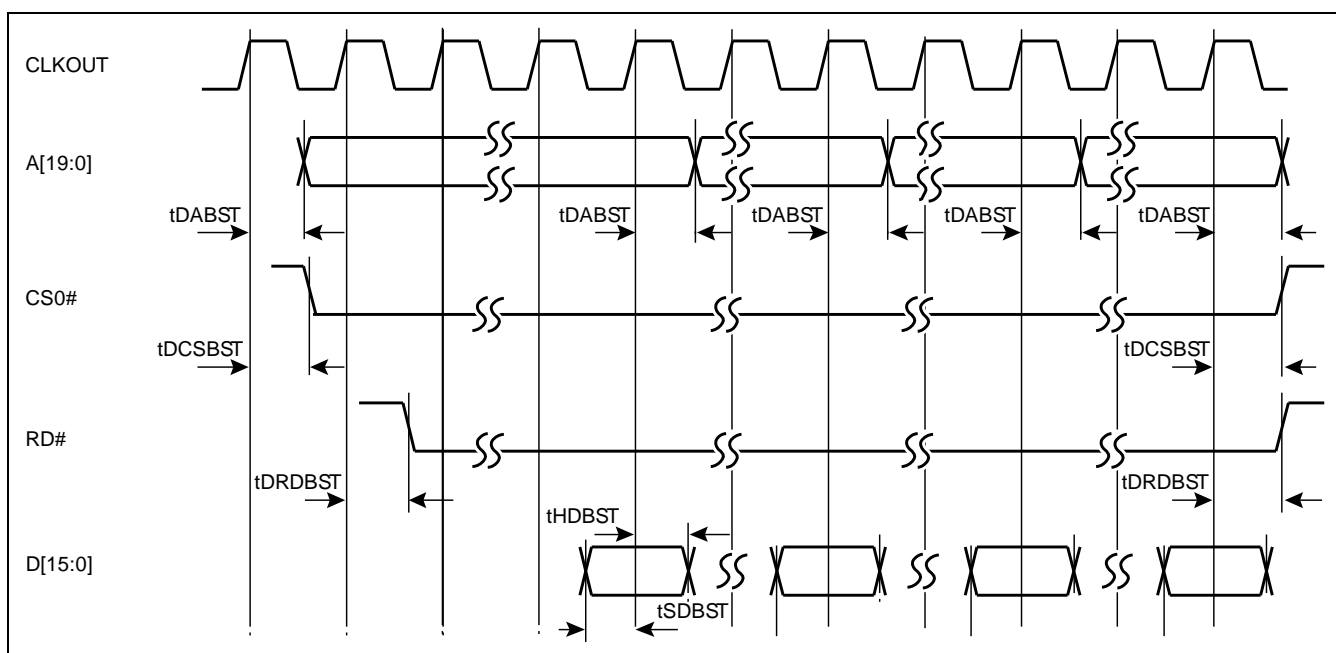


Figure 73.10.2 Burst ROM Read Access Timing

73.11 CSI2

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.11.1 AC Characteristics (1.8-V I/O [CSI2])

Conditions: VDDQ18_CSIn = 1.8 V ± 0.1 V, VDD09_CSIn = 0.82 V +0.06 V/ -0.07 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 VDDQ18_CSIO = 1.8 V ± 0.1 V [RZ/G2E]
 GND = VSS = 0 V,
 Tc = -40 to + 115°C [RZ/G2H, RZ/G2M V1.3]
 Ta = -40 to + 85°C [RZ/G2M V3.0, RZ/G2N, RZ/G2E]
 Tj = -40 to 115°C [RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement condition | Remarks |
|---|-----------------------|------|------|------|------|-----------------------|----------------|
| Common mode interference beyond 450 MHz | $\Delta V_{CMRX(HF)}$ | — | — | 100 | mVpp | — | HS Receiver |
| Common mode interference between 50 MHz and 450 MHz | $\Delta V_{CMRX(LF)}$ | -50 | — | 50 | mVpp | — | HS Receiver |
| Data to Clock Receiver Setup time | $t_{SETUP(RX)}$ | 0.20 | — | — | UI | — | HS Receiver |
| Data to Clock Receiver Hold time | $t_{HOLD(RX)}$ | 0.20 | — | — | UI | — | Figure 73.11.1 |

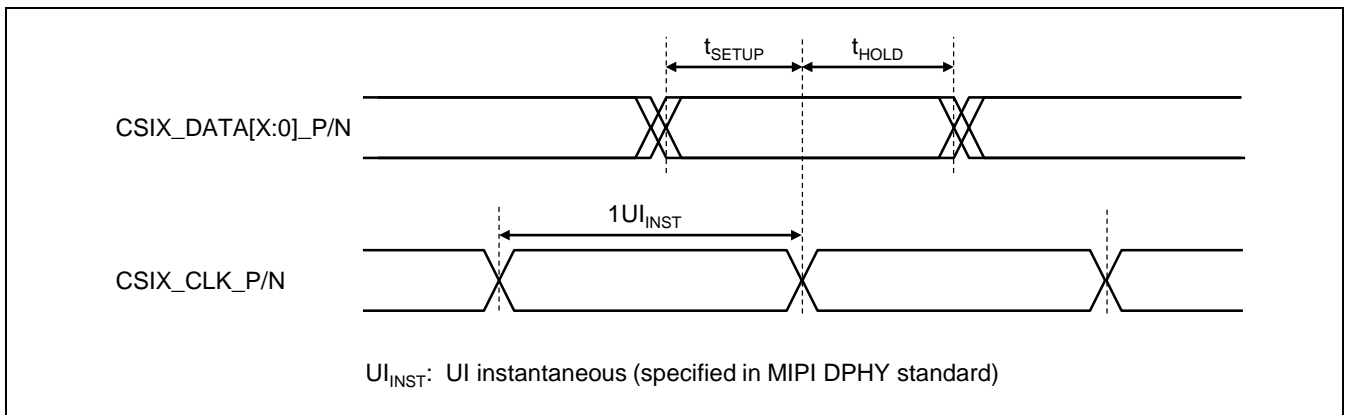


Figure 73.11.1 Data Receive timing

73.12 Video Input Module (VIN)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.12.1 VIN Signal Timing

Conditions: VDDQ33 = 3.3 V ± 0.2 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 VDDQ33 = 3.3V ± 0.3 V [RZ/G2E], GND = VSS = 0 V, Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|------------------------|-----------|------|------|------|--------|----------------|
| VI_CLK cycle time | tVCcyc | 10 | 37 | — | ns | Figure 73.12.1 |
| Data setup time | tVCDs | 3.5 | — | — | ns | |
| Data hold time | tVCDh | 1.5 | — | — | ns | |
| Sync signal setup time | tVCCEs | 3.5 | — | — | ns | |
| Sync signal hold time | tVCCEh | 1.5 | — | — | ns | |
| VI_HSYNC# hold cycle | tVCMcyc | 8* | — | — | tVCcyc | |
| VI_HSYNC# Low period | tVCHL | 300 | — | — | ns | |
| VI_VSYNC# Low period | tVCVL | 3 | — | — | Line | Figure 73.12.2 |
| VI_CLK clock duty | tVCdtyH/L | 45 | 50 | 55 | % | Figure 73.12.3 |

Note: * It is the case when VIn_CLKENB is input.

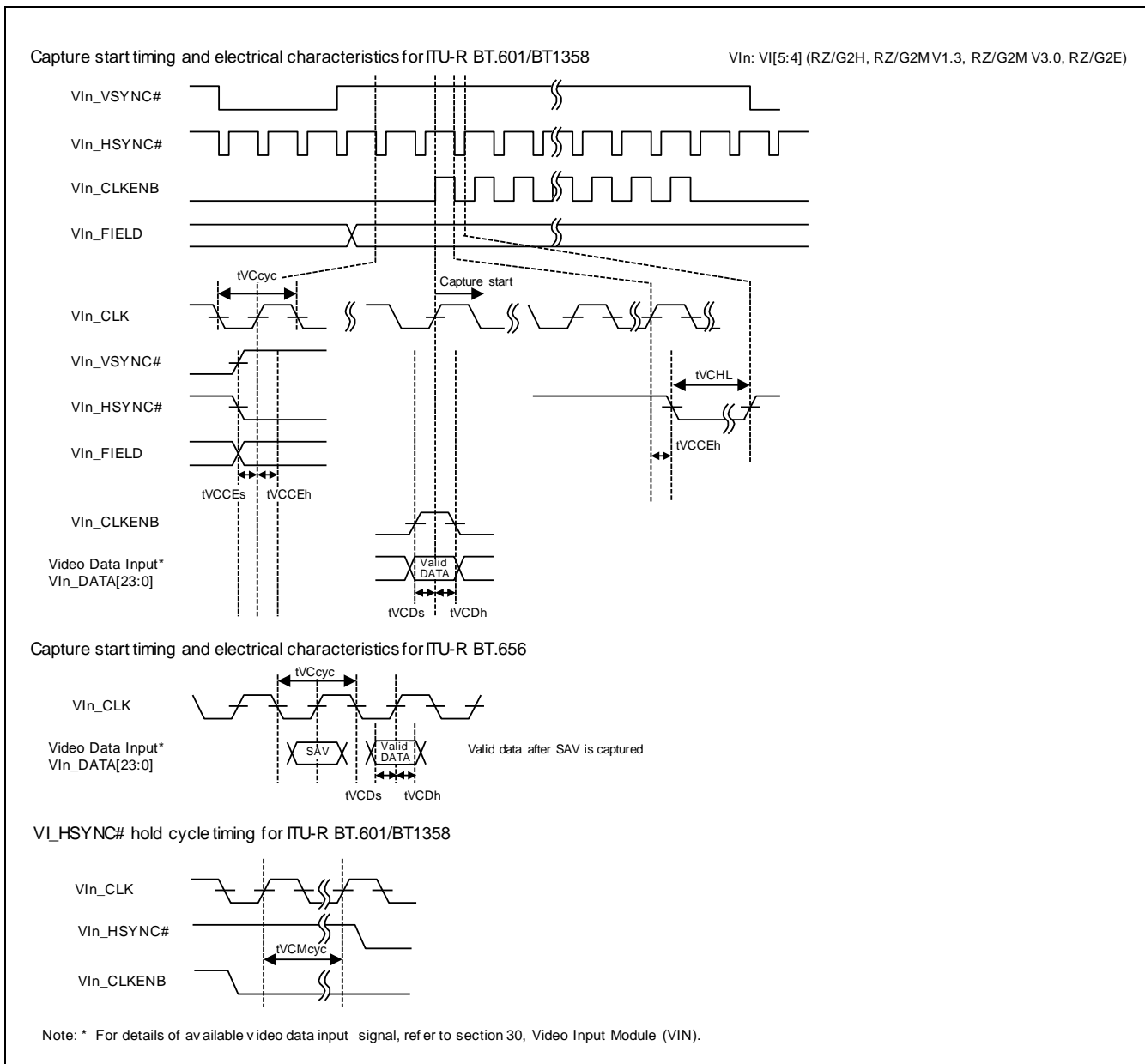


Figure 73.12.1 Capture start timing and electrical characteristics [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]

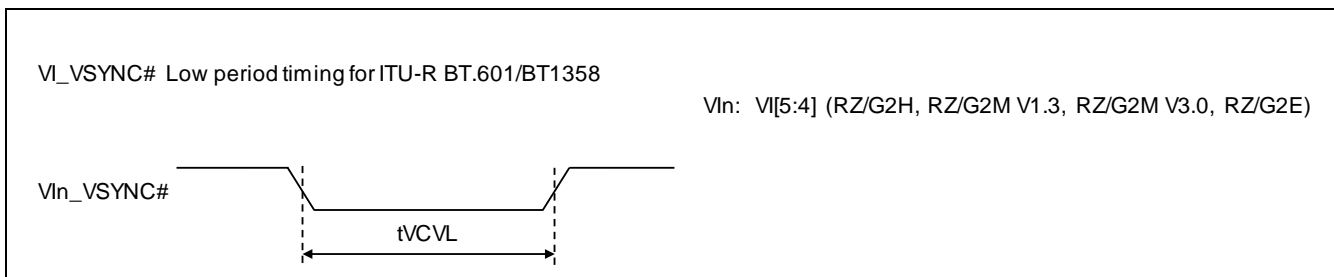


Figure 73.12.2 VI_VSYNC# [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2E]

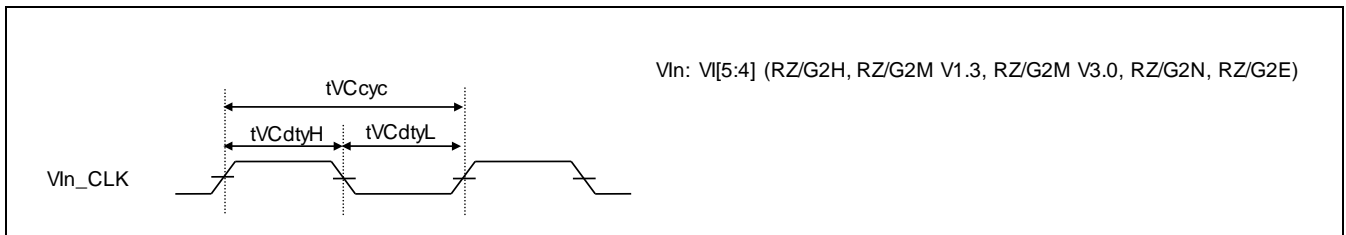


Figure 73.12.3 V_{In_CLK} Clock Duty [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

73.13 Display Unit (DU)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

This electrical characteristic apply to only DU3 output [RZ/G2H, RZ/G2N].

This electrical characteristic apply to only DU2 output [RZ/G2M V1.3, RZ/G2M V3.0].

Table 73.13.1 DOTCLKIN Timing

Conditions: VDDQ18 = 1.8 V ± 0.1 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 (DU_DOTCLKIN0), RZ/G2N] /
 VDDQ33 = 3.3V ± 0.3V [RZ/G2E], GND = VSS = 0 V,
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3], Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|--------------------------|--------|------|------|------|------|----------------|
| DOTCLKIN cycle time | tDICYC | 6.7 | — | 200 | ns | Figure 73.13.1 |
| DOTCLKIN High level time | tDCKIH | 3 | — | — | ns | |
| DOTCLKIN Low level time | tDCKIL | 3 | — | — | ns | |

Table 73.13.2 Display Signal Timing

Conditions: $V_{DDQ33} = 3.3 \text{ V} \pm 0.2 \text{ V}$ [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] / $\pm 0.3 \text{ V}$ [RZ/G2E],
 $GND = VSS = 0 \text{ V}$,
 $T_c = -40 \text{ to } +115 \text{ }^\circ\text{C}$, [RZ/G2H, RZ/G2M V1.3],
 $T_a = -40 \text{ to } +85 \text{ }^\circ\text{C}$ [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 $T_j = -40 \text{ to } +115 \text{ }^\circ\text{C}$ [RZ/G2M V3.0, RZ/G2N, RZ/G2E], $CL = 20 \text{ pF}$

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|--|--------|------|------|------|------|---|
| Display input control signal*1 setup time | tDS | 5 | — | — | ns | Figure 73.13.2 (relative to DOTCLKIN) |
| Display input control signal*1 hold time | tDH | 3 | — | — | ns | [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| DOTCLKOUT output cycle time | tDCYC | 20 | — | 200 | ns | Figure 73.12.3 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| | | 13.3 | — | 200 | ns | Figure 73.13.3 [RZ/G2E] |
| DOTCLKOUT output high level width | tDCKH | 5 | — | — | ns | Figure 73.13.3 |
| Display output control signal*1 output delay time (Relative to DOTCLKOUT rising edge*3) | tDD | 2 | — | 8.5 | ns | Figure 73.13.3 (relative to DOTCLKOUT) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| | | 2 | — | 11 | ns | Figure 73.13.3 (relative to DOTCLKOUT) [RZ/G2E] |
| Display output control signal*1 output delay time (Relative to DOTCLKOUT falling edge*4) | tDDf | 1.5 | — | 9.0 | ns | Figure 73.13.4 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| | | 1.5 | — | 11.5 | ns | Figure 73.13.4 [RZ/G2E] |
| Display digital data*1 output delay time (Relative to DOTCLKOUT rising edge*3) | tDD | 2 | — | 8.5 | ns | Figure 73.13.3 (relative to DOTCLKOUT) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| | | 2 | — | 11 | ns | Figure 73.13.3 (relative to DOTCLKOUT) [RZ/G2E] |

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|---|--------|--------------|------|------|-------|---|
| Display digital data*1 output delay time (Relative to DOTCLKOUT falling edge*4) | tDDf | 1.5 | — | 9.0 | ns | Figure 73.13.4 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| | | 1.5 | — | 11.5 | ns | Figure 73.13.4 [RZ/G2E] |
| Display output control signal*1 output delay time 3*5 (Relative to DOTCLKOUT falling edge) | tDD3f | 1.5 | — | 8.5 | ns | Figure 73.13.6 [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| Display digital data*1 output delay time 3*5 (Relative to DOTCLKOUT falling edge) | tDD3f | 1.5 | — | 8.5 | ns | Figure 73.13.6 [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| Display output control signal*1 output delay time 3*6 (Relative to DOTCLKOUT rising edge) | tDD3 | 1.5 | — | 9.0 | ns | Figure 73.13.7 [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| Display digital data*1 output delay time 3*6 (Relative to DOTCLKOUT rising edge) | tDD3 | 1.5 | — | 9.0 | ns | Figure 73.13.7 [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| EXHSYNC# input low level width | tEXHLW | 4tDCYC | — | — | ns | Figure 73.13.5 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| EXHSYNC# input high level width | tEXHHW | 4tDCYC | — | — | Ns | |
| EXVSYNC# input low level width | tEXVLW | 3HC | — | — | tDCYC | |
| ODDF# setup time 1*2 | tOD1 | (ys+yw) × HC | — | — | tDCYC | |
| ODDF# setup time 2*2 | tOD2 | 1HC | — | — | tDCYC | |

Notes: 1. For correspondence between these signals and pin names, refer to Table 73.9.4.

- ys, yw and HC in Min value of ODDF# setup time 1 and ODDF# setup time 2 (tOD1 and tOD2);
ys: From rise of VSYNC to display start position in the vertical direction of the display screen (unit: raster line)
yw: Vertical display period of display screen (unit: raster line)
HC: Horizontal scan period (unit: dot clock)
- Set bit 25 in ESCRn to B'0 (Initial value). (n = 3[RZ/G2H, RZ/G2N], n = 2[RZ/G2M V1.3, RZ/G2M V3.0], n = 0 and 1[RZ/G2E])
- Set bit 25 in ESCRn to B'1. (n = 3[RZ/G2H, RZ/G2N], n = 2[RZ/G2M V1.3, RZ/G2M V3.0], n = 0 and 1[RZ/G2E])
- Signal is output on the falling edge.
Set the falling edge in OTARn. (n = 3[RZ/G2N], n = 2[RZ/G2M V1.3, RZ/G2M V3.0])
Set bit 25 in ESCRn to B'0 (Initial value). (n = 3[RZ/G2N], n = 2[RZ/G2M V1.3, RZ/G2M V3.0])
- Data is output on the falling edge.
Set the falling edge in OTARn. (n = 3[RZ/G2N], n = 2[RZ/G2M V1.3, RZ/G2M V3.0])
Set bit 25 in ESCRn to B'1. (n = 3[RZ/G2N], n = 2[RZ/G2M V1.3, RZ/G2M V3.0])

Table 73.13.3 DOTCLKOUT output cycle time [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Conditions: $V_{DDQ33} = 3.3 \text{ V} \pm 0.2 \text{ V}$ [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 $GND = VSS = 0 \text{ V}$, $T_c = -40 \text{ to } +115 \text{ }^\circ\text{C}$ [RZ/G2H, RZ/G2M V1.3],
 $T_a = -40 \text{ to } +85 \text{ }^\circ\text{C}$ [RZ/G2M V3.0, RZ/G2N],
 $T_j = -40 \text{ to } +115 \text{ }^\circ\text{C}$ [RZ/G2M V3.0, RZ/G2N]

Load conditions: Characteristic impedance (50 - 60 Ω) + 10pF + Damping resistors (36 Ω)

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|------------------------------|--------|------|------|------|------|----------------|
| DOTCLKOUT output cycle time* | tDCYC | 10 | — | 200 | ns | Figure 73.13.3 |

Note: * Other items are the same as in Table 73.13.2.

Table 73.13.4 Correspondence between Signals in Notes and Pin Names

| Signal in Note | Pin Name |
|--------------------------------|---|
| Display input control signals | DU_EXVSYNC/DU_VSYNC [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| | DU_EXHSYNC/DU_HSYNC [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| | DU_EXODDF/DU_ODDF/DISP/CDE [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| Display output control signals | DU_EXVSYNC/DU_VSYNC [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| | DU_VSYNC [RZ/G2E] |
| | DU_EXHSYNC/DU_HSYNC [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| | DU_HSYNC [RZ/G2E] |
| | DU_EXODDF/DU_ODDF/DISP/CDE [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] |
| | DU_DISP_CDE [RZ/G2E] |
| | DU_DISP |
| | DU_CDE |
| Display digital data | DU_DR7 |
| | DU_DR6 |
| | DU_DR5 |
| | DU_DR4 |
| | DU_DR3 |
| | DU_DR2 |
| | DU_DR1 |
| | DU_DR0 |
| | DU_DG7 |
| | DU_DG6 |
| | DU_DG5 |
| | DU_DG4 |
| | DU_DG3 |
| | DU_DG2 |
| | DU_DG1 |
| | DU_DG0 |
| | DU_DB7 |
| | DU_DB6 |
| | DU_DB5 |
| | DU_DB4 |
| DU_DB3 | |
| DU_DB2 | |
| DU_DB1 | |
| DU_DB0 | |

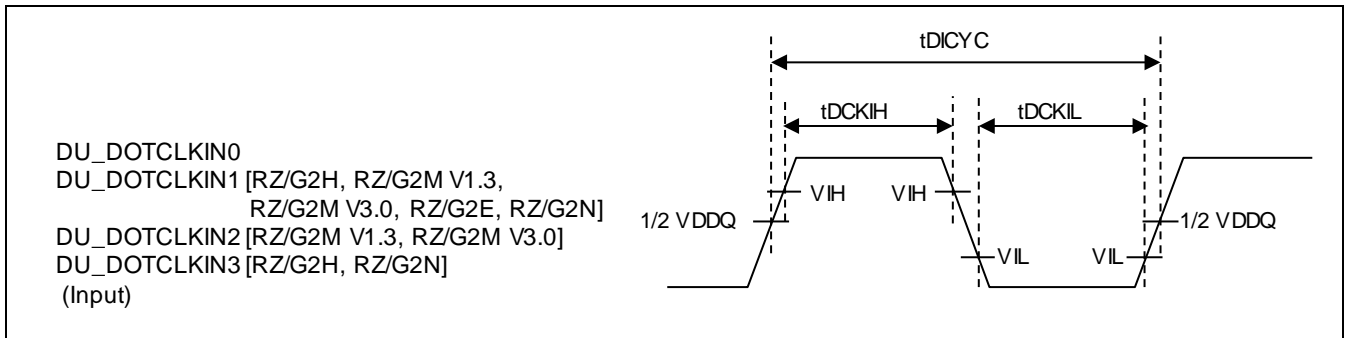


Figure 73.13.1 DOTCLKIN Clock Input Timing

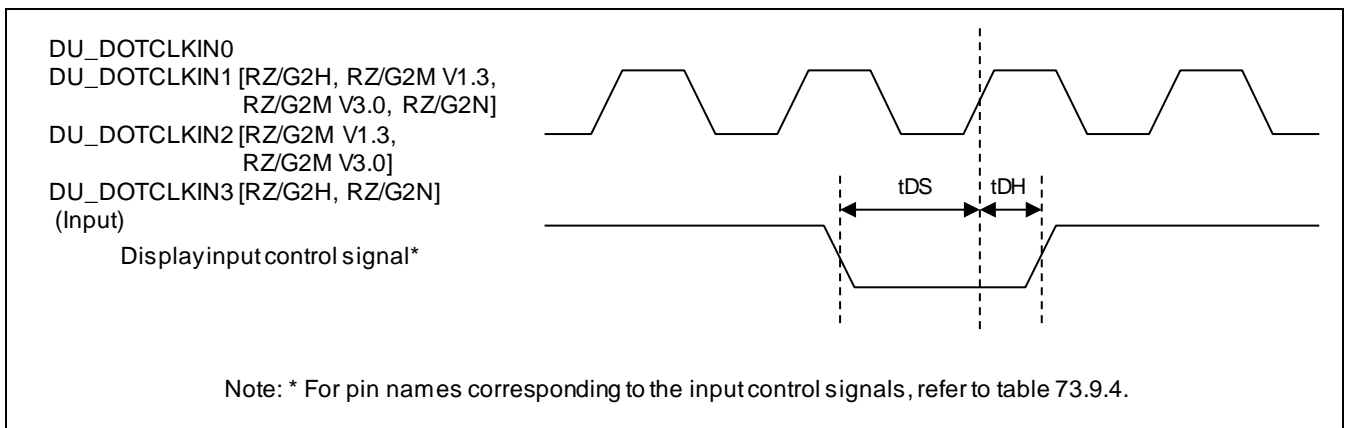


Figure 73.13.2 Display Signal Timing (Relative to DOTCLKIN)
 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

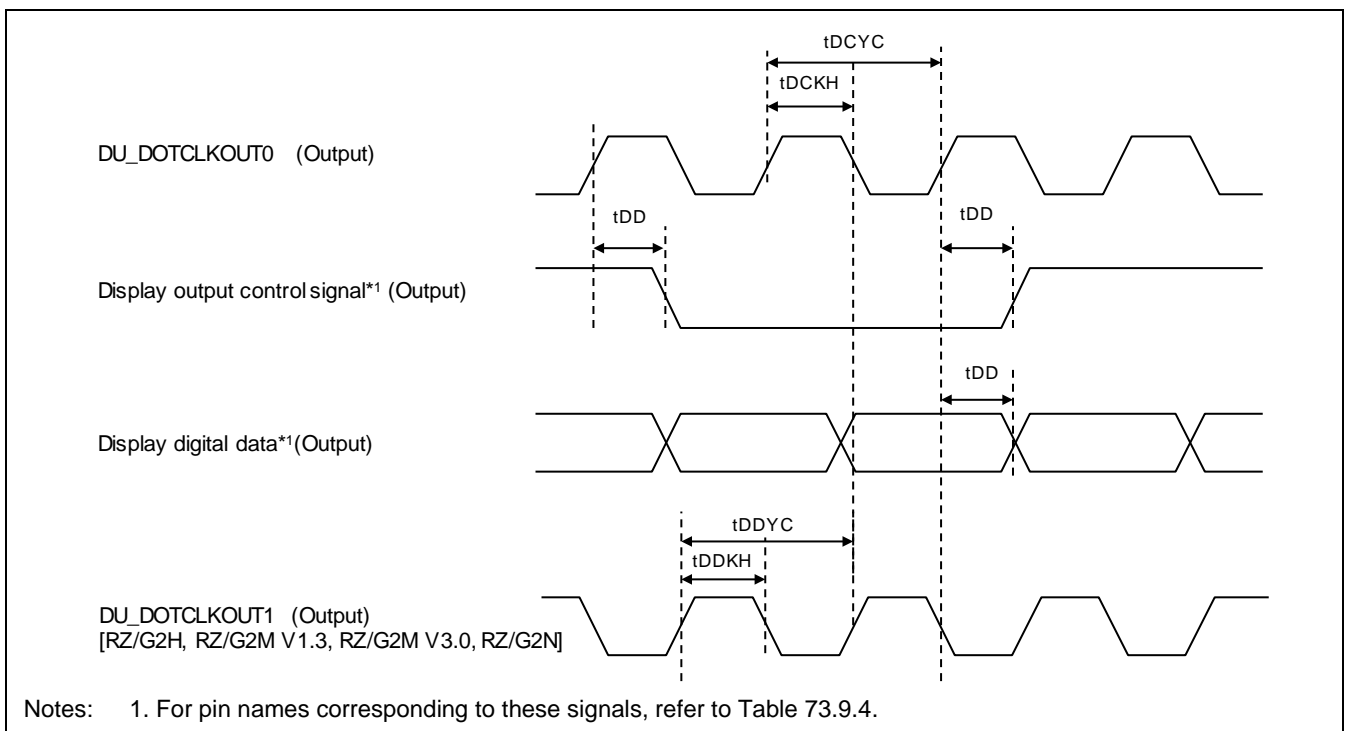


Figure 73.13.3 Display Signal Timing (Relative to DOTCLKOUT)

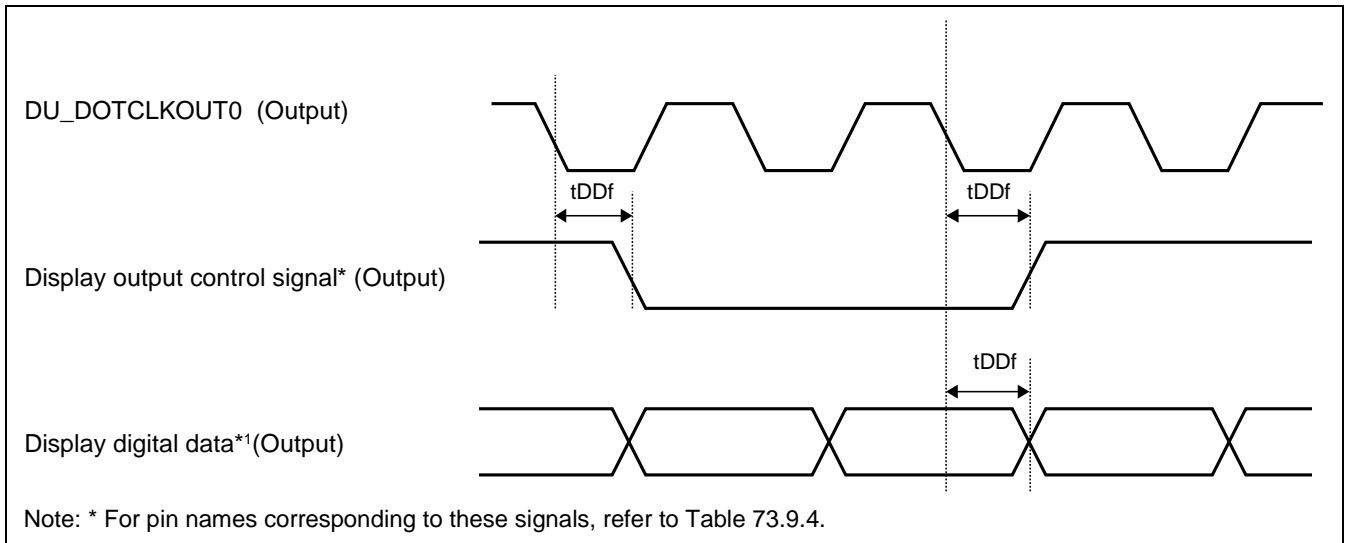


Figure 73.13.4 Display Signal Timing (Relative to DOTCLKOUT falling edge)
 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

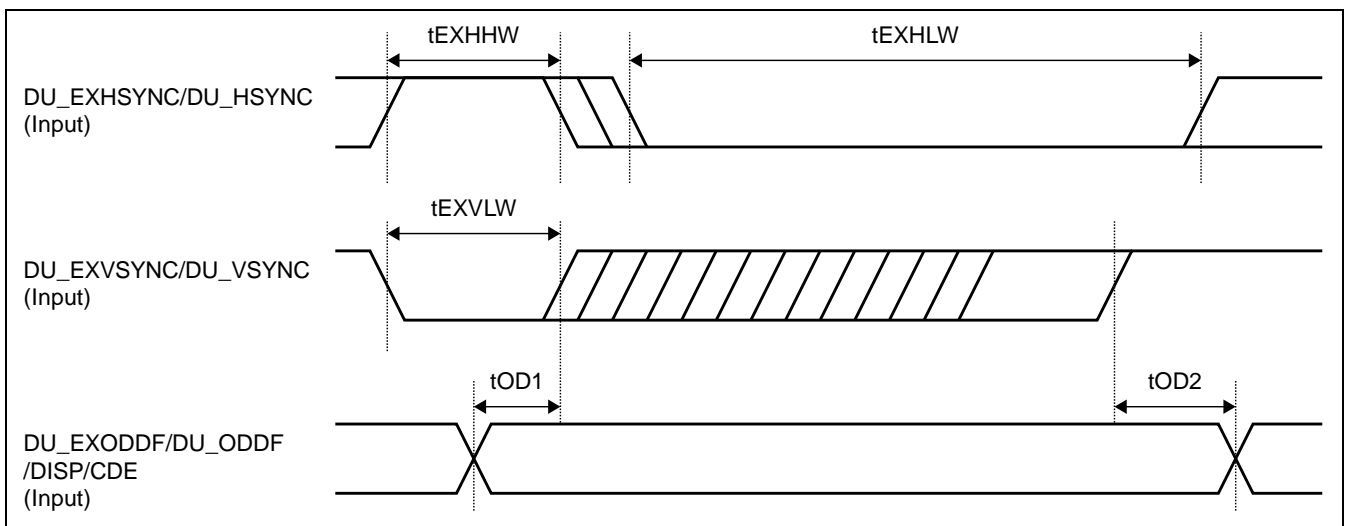


Figure 73.13.5 TV Sync Mode Display Signal Timing [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

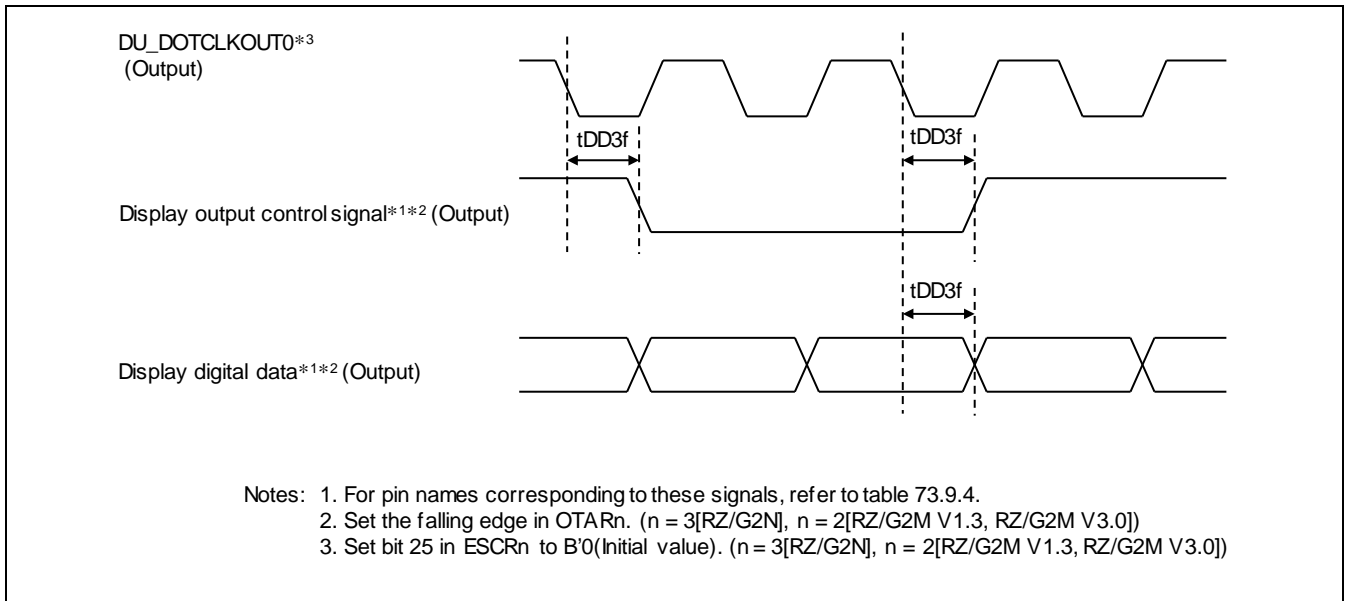


Figure 73.13.6 Display Signal Timing (Relative to DOTCLKOUT falling edge)
 [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

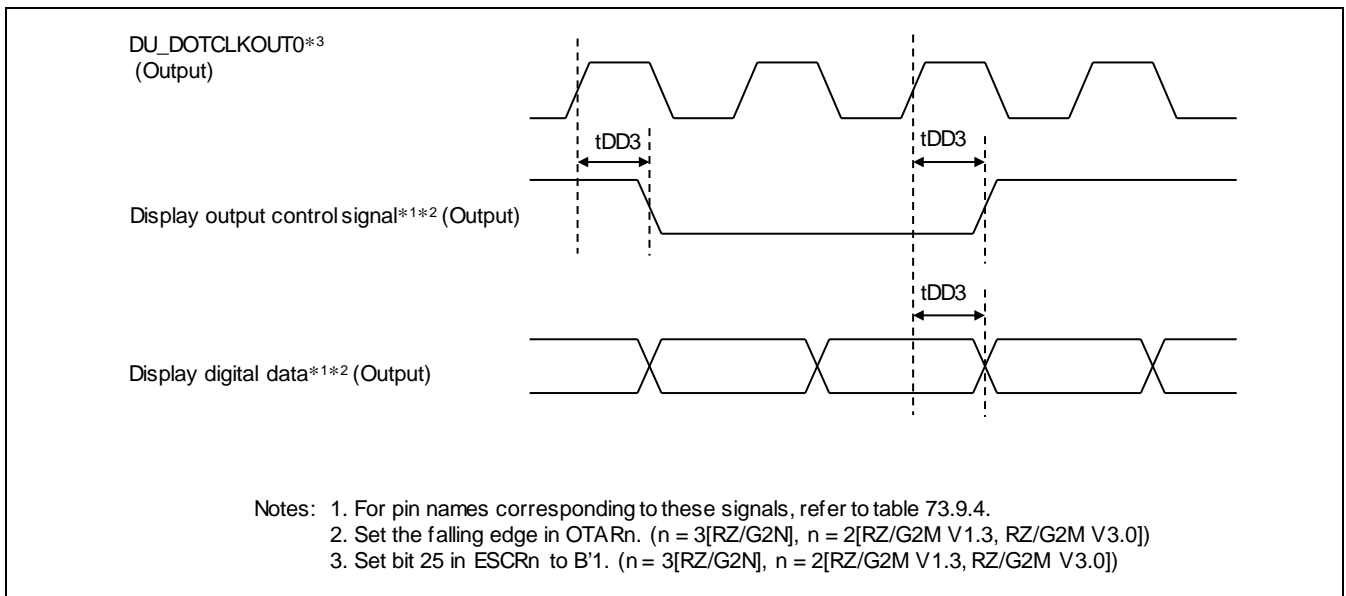


Figure 73.13.7 Display Signal Timing (Relative to DOTCLKOUT rising edge)
 [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

73.14 LVDS-IF

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.14.1 AC Characteristics (1.8-V I/O [LVDS]) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 (in Single-link)/RZ/G2N, RZ/G2E]

Conditions: VDDQ18_LVDS = VDD18_LVDSPLL1 = 1.8V ± 0.1V,
VDDQ09_LVDS = VDD09_LVDSPLL2 = 0.82V + 0.06V/-0.07V,
GND = VSS = 0V [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2H, RZ/G2N],
VDDQ18_LVDS = VDD18_LVDSnPLL (n = 0, 1) = 1.8V ± 0.1V,
GND = VSS = VSS_LVDSnPLL (n = 0, 1) = 0V [RZ/G2E]
Tc = -40 to +115°C [RZ/G2M V1.3, RZ/G2H],
Ta = -40 to +85°C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
Tj = -40 to +115°C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
CL = 3pF

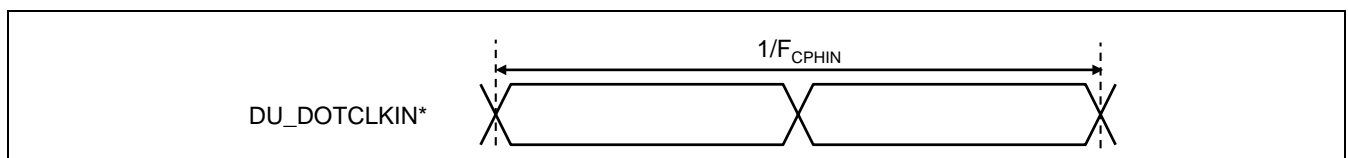
| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement condition | Remarks |
|--------------------------------|------------------------|---|---------------------|---|------|-----------------------|----------------|
| Input Clock Frequency of LVDS | F _{CPHIN} | 31.0 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] | — | 150.0 | MHz | — | Figure 73.14.1 |
| Output Clock Frequency of LVDS | F _{CLKP/CLKN} | 31.0 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 5.0 [RZ/G2E] | — | 150.0 | MHz | — | Figure 73.14.2 |
| Output Clock of LVDS | T _{CP} | 6.67 | — | 32.2 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 200.0 [RZ/G2E] | ns | — | |
| Output Data Rate of LVDS | F _{DAP/DAN} | 217.0 [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] 35.0 [RZ/G2E] | — | 1050.0 | Mbps | — | |
| LVDS Output Pulse Position 0 | T _{PPOS0} | -0.150 | 0 | 0.150 | ns | — | |
| LVDS Output Pulse Position 1 | T _{PPOS1} | (T _{CP} /7) -0.150 | T _{CP} /7 | (T _{CP} /7) +0.150 | ns | — | |
| LVDS Output Pulse Position 2 | T _{PPOS2} | 2(T _{CP} /7) -0.150 | 2T _{CP} /7 | 2(T _{CP} /7) +0.150 | ns | — | |
| LVDS Output Pulse Position 3 | T _{PPOS3} | 3(T _{CP} /7) -0.150 | 3T _{CP} /7 | 3(T _{CP} /7) +0.150 | ns | — | |
| LVDS Output Pulse Position 4 | T _{PPOS4} | 4(T _{CP} /7) -0.150 | 4T _{CP} /7 | 4(T _{CP} /7) +0.150 | ns | — | |

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement condition | Remarks |
|------------------------------|--------|-----------------|--------|-----------------|------|-----------------------|----------------|
| LVDS Output Pulse Position 5 | TPPOS5 | 5(TCP/7) -0.150 | 5TCP/7 | 5(TCP/7) +0.150 | ns | — | Figure 73.14.2 |
| LVDS Output Pulse Position 6 | TPPOS6 | 6(TCP/7) -0.150 | 6TCP/7 | 6(TCP/7) +0.150 | ns | — | |

Table 73.14.2 AC Characteristics in Dual-link (1.8-V I/O [LVDS]) [RZ/G2E]

Conditions: $VDDQ18_LVDS = VDD18_LVDSnPLL$ ($n = 0, 1$) = 1.8V \pm 0.1V [RZ/G2E],
 $GND = VSS = VSS_LVDSnPLL$ ($n = 0, 1$) = 0V [RZ/G2E],
 $T_a = -40$ to $+85^\circ\text{C}$ [RZ/G2E],
 $T_j = -40$ to $+115^\circ\text{C}$ [RZ/G2E]
 $CL = 3\text{pF}$

| Item | Symbol | Min. | Typ. | Max. | Unit | Measurement condition | Remarks |
|--------------------------------|------------------------|-----------------------------|---------------------|-----------------------------|------|-----------------------|----------------|
| Input Clock Frequency of LVDS | F _{CPHIN} | 12.0 | — | 150.0 | MHz | — | Figure 73.14.1 |
| Output Clock Frequency of LVDS | F _{CLKP/CLKN} | 5.0 | — | 75.0 | MHz | — | Figure 73.14.3 |
| Output Clock of LVDS | T _{CP} | 13.3 | — | 200.0 | ns | — | |
| Output Data Rate of LVDS | F _{DAP/DAN} | 35 | — | 525.0 | Mbps | — | |
| LVDS Output Pulse Position 0 | T _{PPOS0} | -0.200 | 0 | 0.200 | ns | — | |
| LVDS Output Pulse Position 1 | T _{PPOS1} | (T _{CP} /7)-0.200 | T _{CP} /7 | (T _{CP} /7)+0.200 | ns | — | |
| LVDS Output Pulse Position 2 | T _{PPOS2} | 2(T _{CP} /7)-0.200 | 2T _{CP} /7 | 2(T _{CP} /7)+0.200 | ns | — | |
| LVDS Output Pulse Position 3 | T _{PPOS3} | 3(T _{CP} /7)-0.200 | 3T _{CP} /7 | 3(T _{CP} /7)+0.200 | ns | — | |
| LVDS Output Pulse Position 4 | T _{PPOS4} | 4(T _{CP} /7)-0.200 | 4T _{CP} /7 | 4(T _{CP} /7)+0.200 | ns | — | |
| LVDS Output Pulse Position 5 | T _{PPOS5} | 5(T _{CP} /7)-0.200 | 5T _{CP} /7 | 5(T _{CP} /7)+0.200 | ns | — | |
| LVDS Output Pulse Position 6 | T _{PPOS6} | 6(T _{CP} /7)-0.200 | 6T _{CP} /7 | 6(T _{CP} /7)+0.200 | ns | — | |

**Figure 73.14.1 F_{CPHIN}**

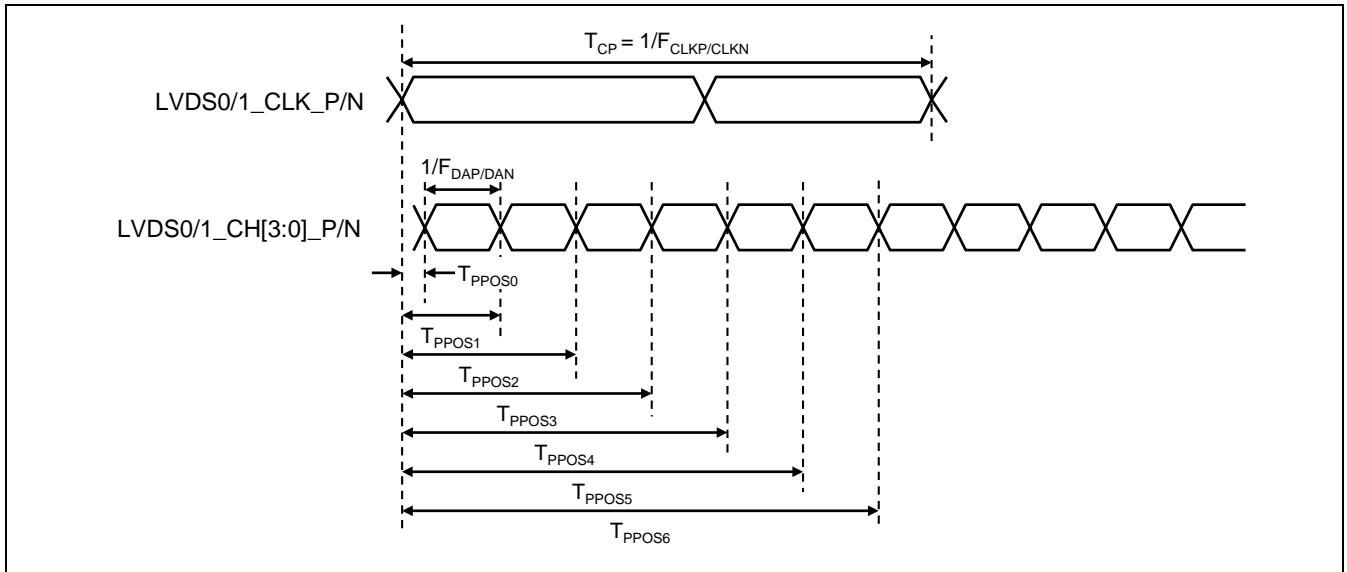


Figure 73.14.2 FCLKP/CLKN, TCP, FDAP/DAN, TPPOS in Single-link

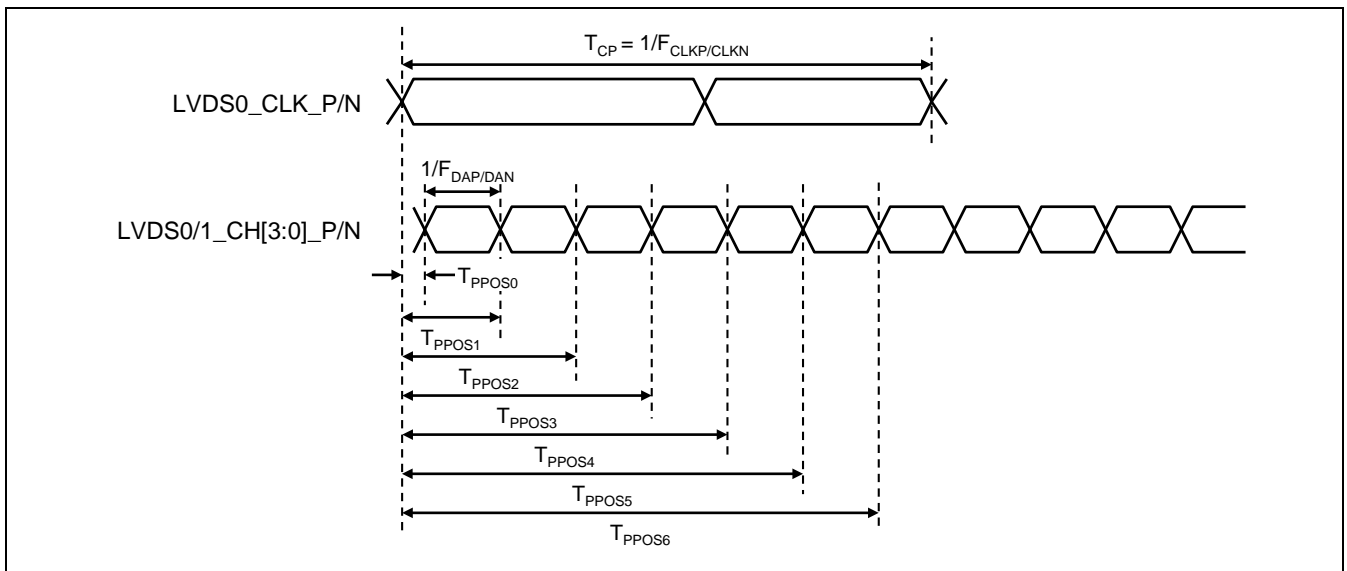


Figure 73.14.3 FCLKP/CLKN, TCP, FDAP/DAN, TPPOS in Dual-link

73.15 SSI Interface

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.15.1 SSI Interface Signal Timing

Conditions: VDDQ33 = 3.10 V – 3.50 V, GND = VSS = 0 V, Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N], Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N]
 CL = 30 pF (other than tRC: Rise-edge clock timing) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Conditions: VDDQ33 = 3.00 V – 3.60 V, GND = VSS = 0 V, Ta = -40 to +85 °C, Tj = -40 to +115 °C,
 CL = 30 pF (other than tRC: Rise-edge clock timing) [RZ/G2E]

| Item | Symbol | Min. | Typ. | Max. | Unit | Note | Figure |
|-------------------------|------------|-------|------|-------|--------|----------------|----------------------------------|
| Output clock cycle | tO | 80 | — | 15625 | ns | — | Figure 73.15.1 |
| Input clock cycle | tI | 66 | — | 15625 | ns | — | |
| Output clock high-cycle | tHC | 35 | — | — | ns | — | |
| Output clock low-cycle | tLC | 35 | — | — | ns | — | |
| Input clock high-cycle | tHC | 28 | — | — | ns | — | |
| Input clock low-cycle | tLC | 28 | — | — | ns | — | |
| Rise-edge clock timing | tRC | — | — | 20 | ns | Output (100pF) | |
| Output delay | tD | -5 | — | 19 | ns | — | Figure 73.15.2 to Figure 73.15.5 |
| | tD | — | — | 19 | ns | — | Figure 73.15.6 |
| Setup time | tS | 15 | — | — | ns | — | Figure 73.15.2 to Figure 73.15.5 |
| Hold time | tH | 5 | — | — | ns | — | |
| Audio clock frequency | fAUDIO | 3.072 | — | 25 | MHz | — | Figure 73.15.7 |
| Audio clock duty | fAUDIOduty | 0.45 | — | 0.55 | fAUDIO | — | |

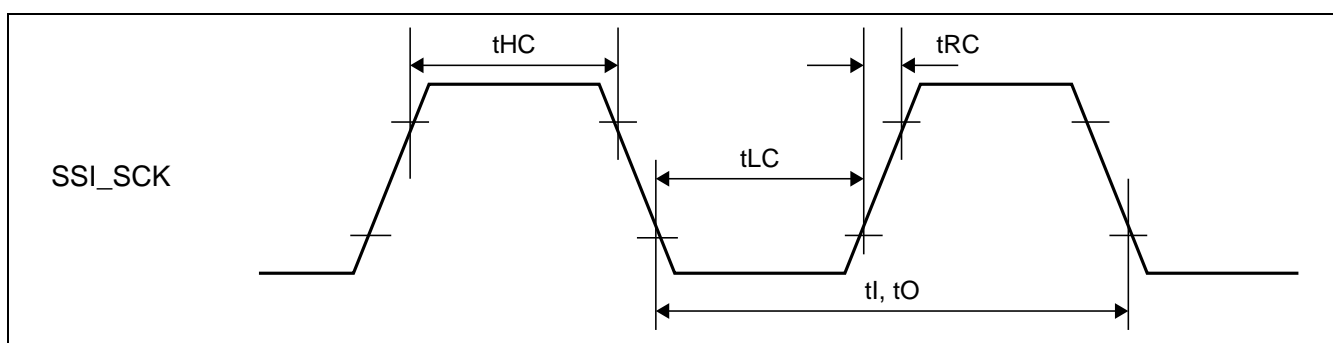


Figure 73.15.1 SCK Clock Input/output Timing

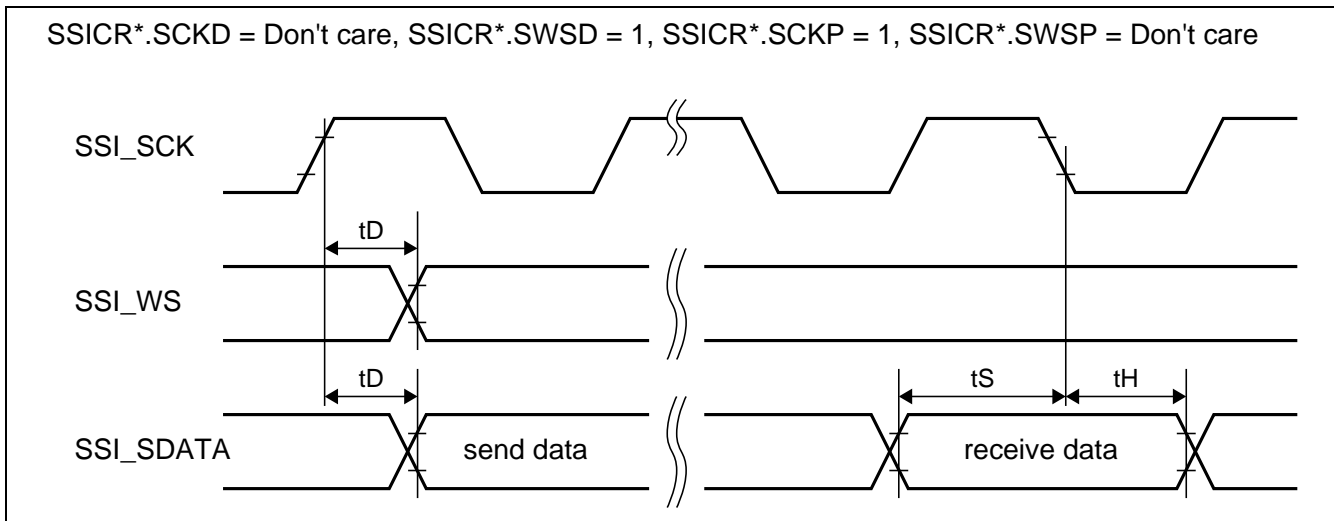


Figure 73.15.2 SSI Timing (1)

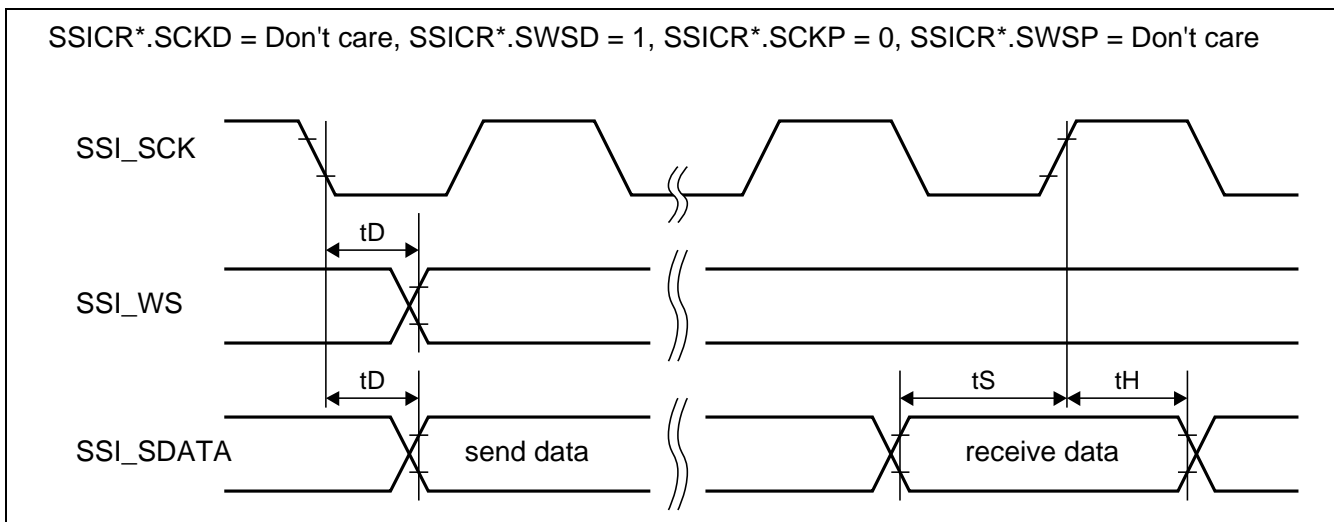


Figure 73.15.3 SSI Timing (2)

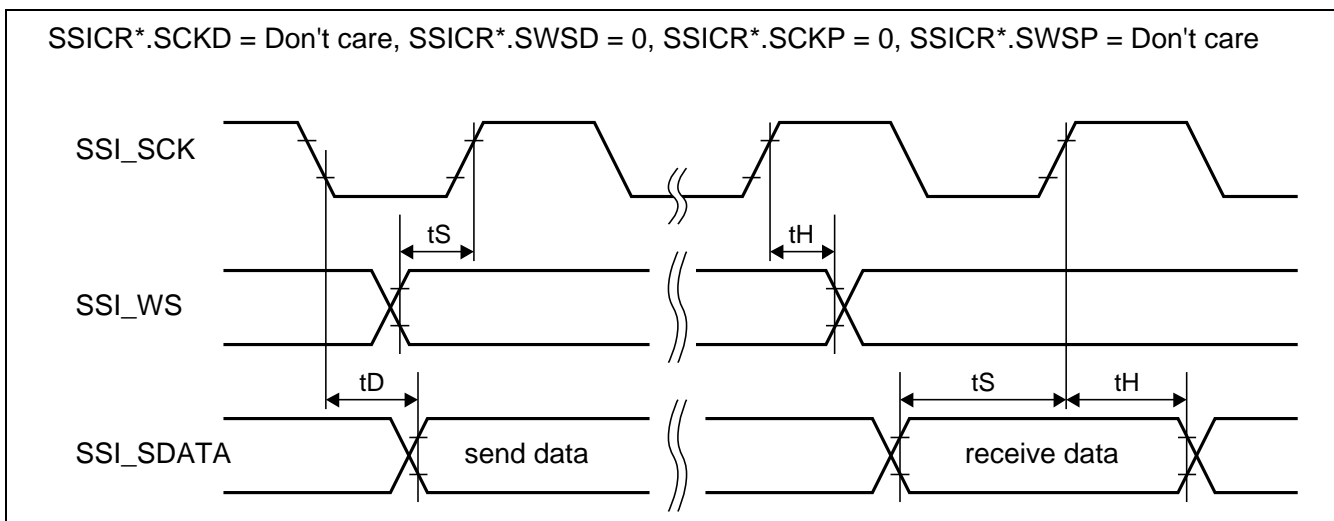


Figure 73.15.4 SSI Timing (3)

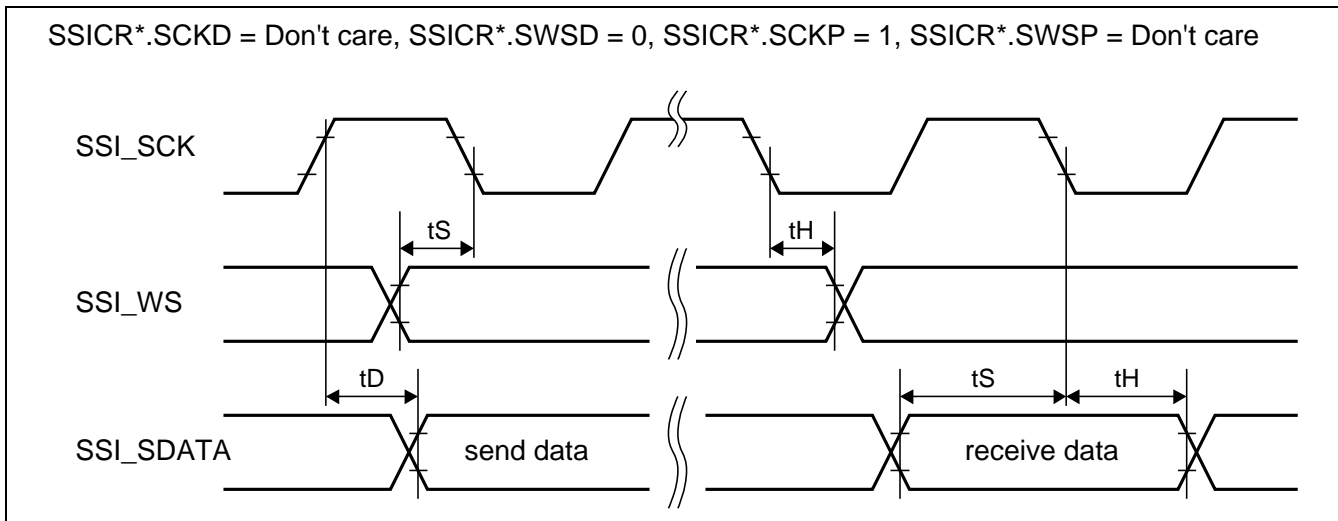


Figure 73.15.5 SSI Timing (4)

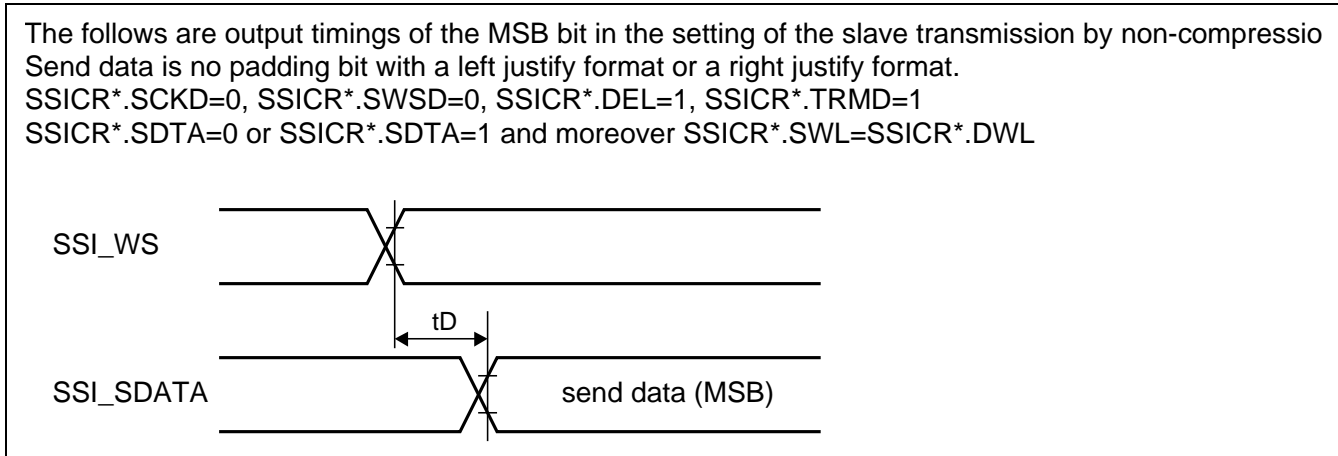


Figure 73.15.6 SSI Timing (5)

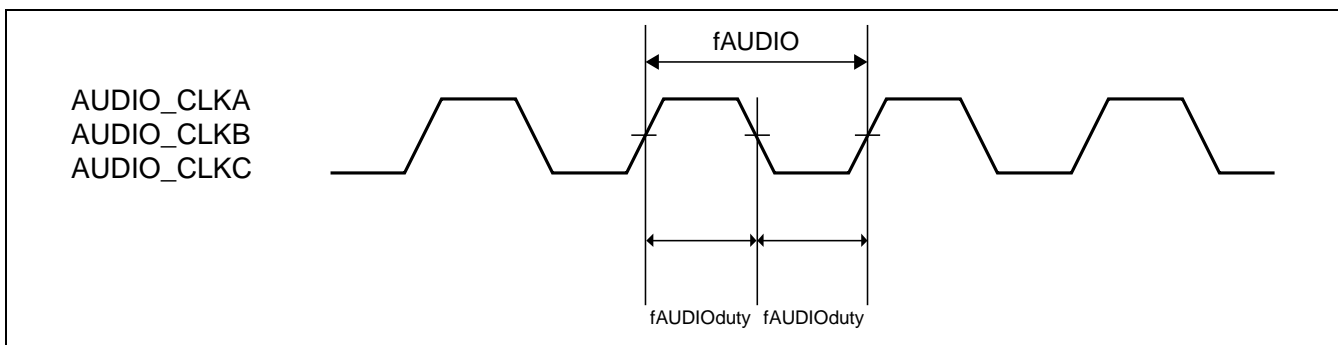


Figure 73.15.7 AUDIO_CLK Input Timing

73.16 Ethernet AVB-IF Module Signal Timing

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.16.1 Ethernet Control Timing (RGMII 100Mbps Mode)

Conditions: VDDQ33 = 3.3 ± 0.2 V, VDDQ25_ETH = 2.5 ± 0.1V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 VDDQ25_AVB0 = 2.5 ± 0.2V [RZ/G2E],
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], GND = VSS = 0 V, CL = 30 pF

Recommend: Connection of series 25 Ω as damping resistor to RGMII output buffer. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|----------------------------------|--------|--------------|------|--------------|------|----------------|
| AVB_TXCREFLCK cycle time | — | 125 – 50 ppm | — | 125 + 50 ppm | MHz | — |
| AVB_TXC cycle time | Tcyc | 36 | 40 | 44 | ns | |
| AVB_TXC duty cycle | Duty_T | 40 | 50 | 60 | % | |
| AVB_TX_CTL to clock output skew | TskewT | -500 | 0 | 500 | ps | Figure 73.16.1 |
| AVB_TD[3:0] to clock output skew | TskewT | -500 | 0 | 500 | ps | |
| AVB_RXC cycle time | Tcyc | 36 | 40 | 44 | ns | — |
| AVB_RXC duty cycle | Duty_T | 40 | 50 | 60 | % | |
| AVB_RX_CTL to clock input skew | TskewR | 1 | 1.8 | 17 | ns | Figure 73.16.2 |
| AVB_RD[3:0] to clock input skew | TskewR | 1 | 1.8 | 17 | ns | |

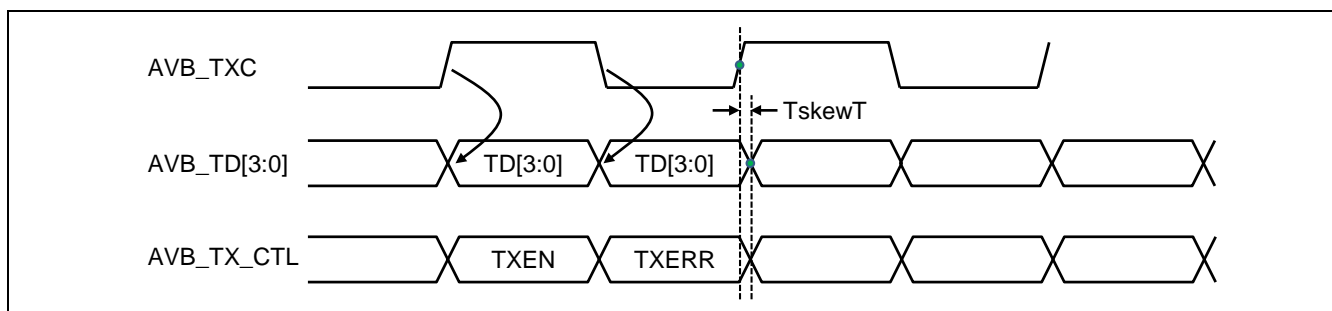


Figure 73.16.1 RGMII 100 Mbps Mode Transmission Timing

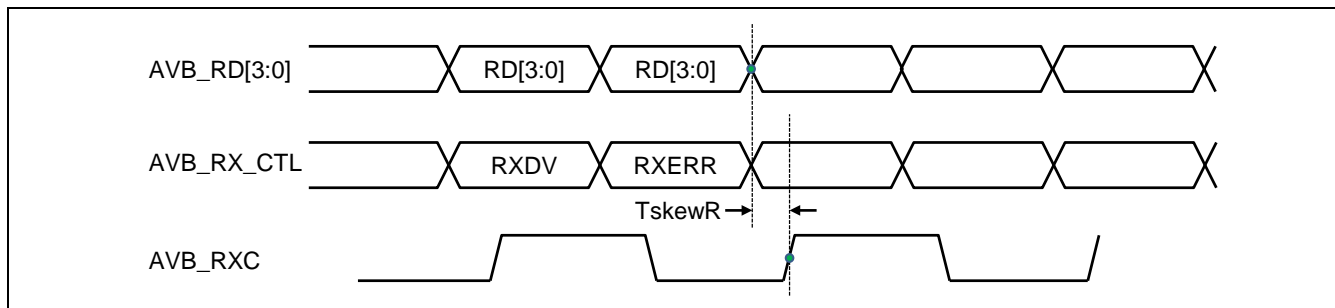


Figure 73.16.2 RGMII 100 Mbps Mode Reception Timing

Table 73.16.2 Ethernet Control Timing (RGMII 1Gbps Mode)

Conditions: $V_{DDQ33} = 3.3 \pm 0.2$ V, $V_{DDQ25_ETH} = 2.5 \pm 0.1$ V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 $V_{DDQ25_AVB0} = 2.5 \pm 0.2$ V [RZ/G2E],
 $T_c = -40$ to $+115$ °C [RZ/G2H, RZ/G2M V1.3],
 $T_a = -40$ to $+85$ °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 $T_j = -40$ to $+115$ °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], $GND = VSS = 0$ V, $CL = 30$ pF

Recommend: Connection of series 25 Ω as damping resistor to RGMII output buffer. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|----------------------------------|--------------------|--------------|------|--------------|------|----------------|
| AVB_TXCREFLCK cycle time | — | 125 – 50 ppm | — | 125 + 50 ppm | MHz | — |
| AVB_TXC cycle time | T _{cyc} | 7.2 | 8 | 8.8 | ns | — |
| AVB_TXC duty cycle | Duty_G | 45 | 50 | 55 | % | — |
| AVB_TX_CTL to clock output skew | T _{skewT} | -500 | 0 | 500 | ps | Figure 73.16.3 |
| AVB_TD[3:0] to clock output skew | T _{skewT} | -500 | 0 | 500 | ps | — |
| AVB_RXC cycle time | T _{cyc} | 7.2 | 8 | 8.8 | ns | — |
| AVB_RXC duty cycle | Duty_G | 45 | 50 | 55 | % | — |
| AVB_RX_CTL to clock input skew | T _{skewR} | 1 | 1.8 | 2.6 | ns | Figure 73.16.4 |
| AVB_RD[3:0] to clock input skew | T _{skewR} | 1 | 1.8 | 2.6 | ns | — |

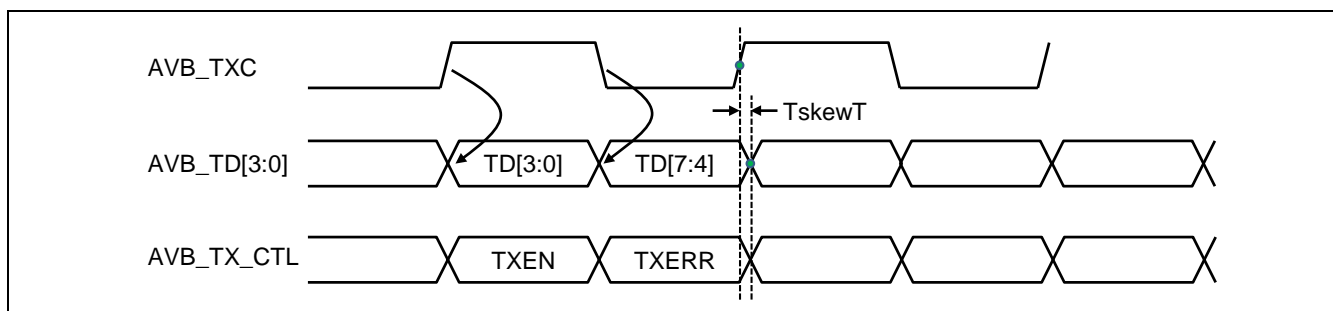


Figure 73.16.3 RGMII 1 Gbps Mode Transmission Timing

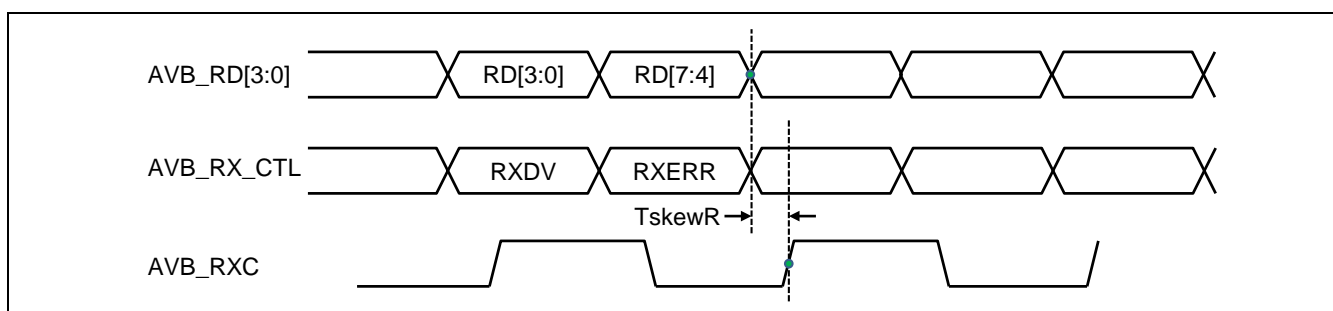


Figure 73.16.4 RGMII 1 Gbps Mode Reception Timing

Table 73.16.3 Ethernet Control Timing (RGMII Tx clock internal Delay Mode) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Conditions: VDDQ33 = 3.3 ± 0.2 V, VDDQ25_ETH = 2.5 ± 0.1V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N], GND = VSS = 0 V, CL = 30 pF

Recommend: Connection of series 25 Ω as damping resistor to RGMII output buffer. [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|----------------------------------|---------|--------------|------|--------------|------|----------------|
| AVB_TXCREFLCK cycle time | — | 125 – 50 ppm | — | 125 + 50 ppm | MHz | — |
| AVB_TXC cycle time* | Tcyc | 7.2 | 8 | 8.8 | ns | |
| AVB_TXC duty cycle (in 1 Gbps) | Duty_G | 45 | 50 | 55 | % | |
| AVB_TXC duty cycle (in 100 Mbps) | Duty_T | 40 | 50 | 60 | % | |
| AVB_TX_CTL setup time | TsetupT | 1.2 | 2.0 | — | ns | Figure 73.16.5 |
| AVB_TX_CTL hold time | TholdT | 1.2 | 2.0 | — | ns | |
| AVB_TD[3:0] setup time | TsetupT | 1.2 | 2.0 | — | ns | |
| AVB_TD[3:0] hold time | TholdT | 1.2 | 2.0 | — | ns | |

Note: * For 100 Mbps, Tcyc will scale to 40 ns +- 4 ns respectively.

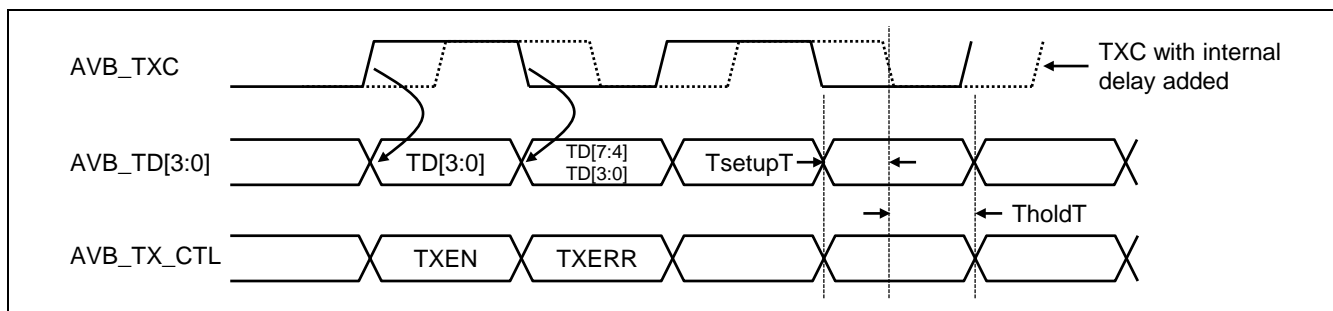


Figure 73.16.5 RGMII Delay Mode Transmission Timing

Table 73.16.4 Ethernet Control Timing (RGMII Rx clock internal Delay Mode)

Conditions: VDDQ25_ETH = 2.5 ± 0.1 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 VDDQ25_AVB0 = 2.5 ± 0.2 V [RZ/G2E],
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], GND = VSS = 0 V, CL = 30 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|----------------------------------|--------|------|------|------|------|-----------------|
| AVB_RXC cycle time* | Tcyc | 7.2 | 8 | 8.8 | ns | — |
| AVB_RXC duty cycle (in 1 Gbps) | Duty_G | 45 | 50 | 55 | % | |
| AVB_RXC duty cycle (in 100 Mbps) | Duty_T | 40 | 50 | 60 | % | |
| AVB_RX_CTL to clock input skew | TskewR | -500 | 0 | 500 | ps | Figure 73.16. 6 |
| AVB_RD[3:0] to clock input skew | TskewR | -500 | 0 | 500 | ps | |

Note: * For 100 Mbps, Tcyc will scale to 40 ns +/- 4 ns respectively.

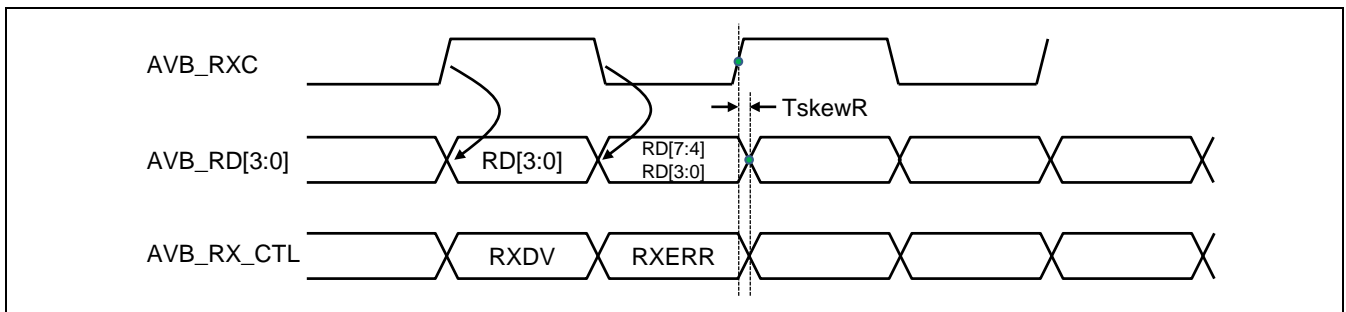


Figure 73.16.6 RGMII Delay Mode Reception Timing

Table 73.16.5 AVTP pins timing specification [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

Conditions: VDDQ33 = 3.3 ± 0.2 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 VDDQ33 = 3.3 ± 0.3 V [RZ/G2E],
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], GND = VSS = 0 V, CL = 30 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|------------------------------------|--------------|------|------|------|--------|----------------|
| AVB_AVTP_CAPTURE input pulse width | PWidth_capt | 3 | — | — | Cycle* | Figure 73.16.7 |
| AVB_AVTP_CAPTURE input period | Period_capt | 9 | — | — | Cycle* | |
| AVB_AVTP_MATCH output pulse width | PWidth_match | 1 | — | 4 | Cycle* | Figure 73.16.8 |
| AVB_AVTP_PPS output pulse width | PWidth_pps | — | 32 | — | Cycle* | Figure 73.16.9 |

Note: * 1 cycle = HPφ cycle.

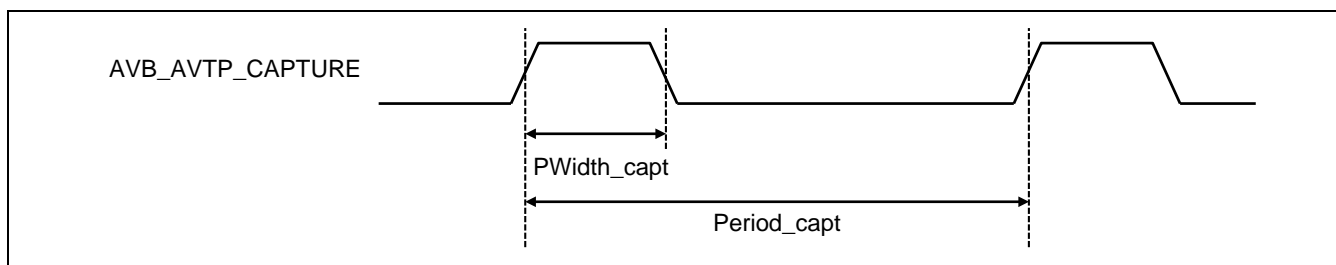


Figure 73.16.7 AVB_AVTP_CAPTURE input Timing

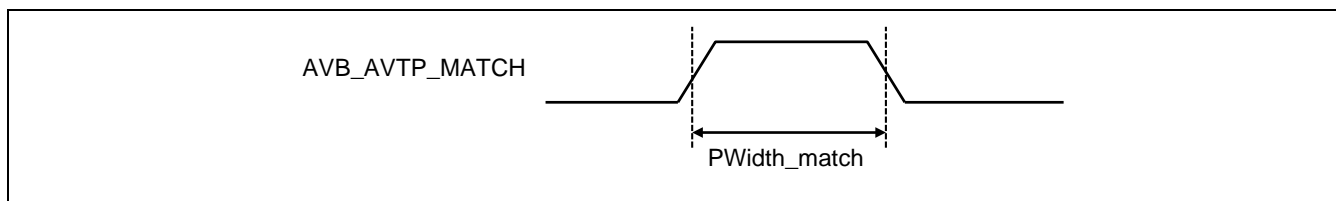


Figure 73.16.8 AVB_AVTP_MATCH output Timing

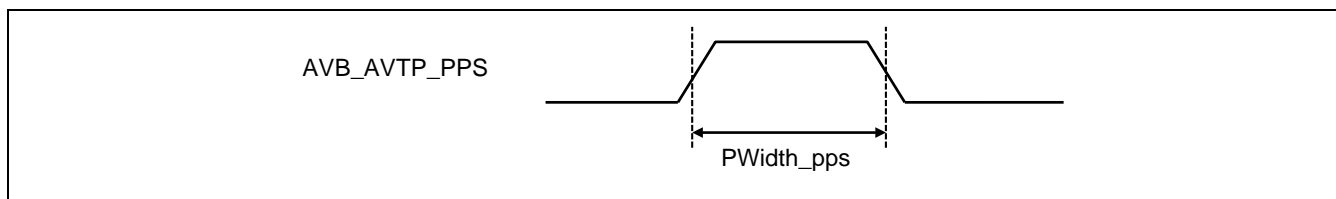


Figure 73.16.9 AVB_AVTP_PPS output Timing

Table 73.16.6 Sideband pins timing specification

Conditions: VDDQ33 = 3.3 ± 0.2 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 VDDQ25_AVB0 = 2.5 ± 0.2 V/3.3 ± 0.3 V [RZ/G2E],
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], GND = VSS = 0 V, CL = 30 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|-------------------------------|-------------|------|------|------|--------|-----------------|
| AVB_MDIO input pulse width | PWidth_side | 2 | — | — | Cycle* | Figure 73.16.10 |
| AVB_PHY_INT input pulse width | PWidth_side | 2 | — | — | Cycle* | |
| AVB_LINK input pulse width | PWidth_side | 2 | — | — | Cycle* | |

Note: * 1 cycle = HPφ cycle.

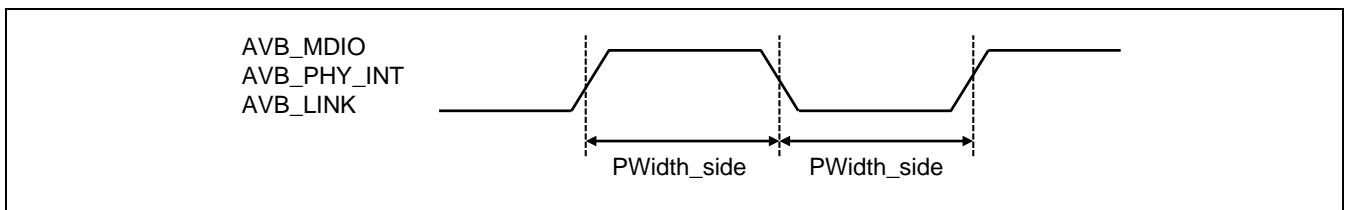


Figure 73.16.10 Sideband signals input Timing

Note: About AVB_MDC, in case of connecting to a device that does not have a built-in pull resistor, should add pull register at external.

73.17 CAN, CAN-FD

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.17.1 CAN Signal Timing

Conditions: VDDQ33 = 3.3 ± 0.2 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] / ± 0.3 V [RZ/G2E],
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 GND = VSS = 0 V, CL = 30 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|----------------------------------|--------|------|------|------|-------|----------------|
| Input clock cycle (asynchronous) | tSCYC | 20 | — | — | ns | Figure 73.17.1 |
| Input clock pulse width | tSCKW | 0.4 | — | 0.6 | tSCYC | |
| Input clock rise time | tSCKr | — | — | 0.2 | tCYC | |
| Input clock fall time | tSCKf | — | — | 0.2 | tCYC | |

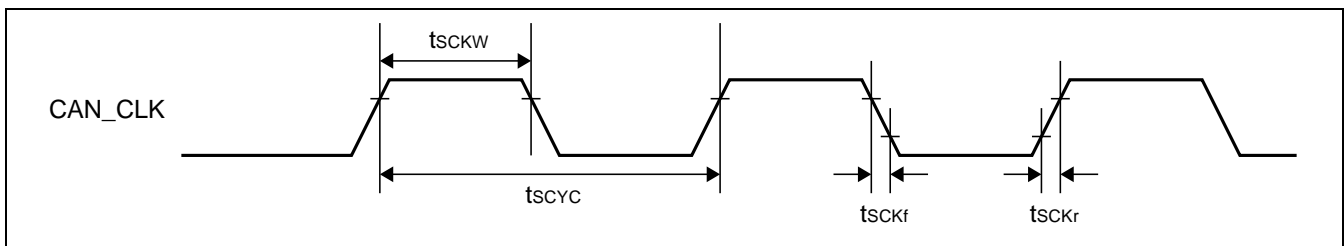


Figure 73.17.1 Input Clock Timing

73.18 PCIE Controller

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The description in this section is compliant with the following PCIe standard: 'PCI Express® Base Specification Revision 2.1, March 4, 2009'.

Table 73.18.1 PCIE Controller Characteristics [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

Conditions: VDDQ18_PCIE0 = 1.8 ± 0.1 V, VDDD_PCIE0 = 1.03 ± 0.05 V [RZ/G2E], VDD33_PCIEn (n = 0, 1) = 3.3 ± 0.2 V, VDD09_PCIEn (n = 0, 1) = 0.82 – 0.07 V / + 0.06 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N], GND = 0 V,
Tc = –40 to +115 °C [RZ/G2H, RZ/G2M V1.3], Ta = –40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
Tj = –40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Test Item | Symbol | Min. | Typ. | Max. | Unit | Remarks |
|---|----------------------|--------|------|--------|------|---------|
| Differential Input Peak to Peak Voltage (2.5 GT/s) | VRX-DIFFp-p | 0.175 | — | 1.200 | V | *1 |
| Differential Input Peak to Peak Voltage (5 GT/s) | VRX-DIFFp-p | 0.120 | — | 1.200 | V | *1 |
| Differential Peak to Peak Output Voltage (2.5 & 5 GT/s) | VTX-DIFFp-p | 0.800 | — | 1.2 | V | *2 |
| Absolute Delta of DC Common Mode Voltage between D+ and D- (2.5 & 5 GT/s) | VTX-CM-DC-LINE-DELTA | 0 | — | 25 | mV | *2 |
| Unit Interval (2.5 GT/s) | UI | 399.88 | — | 400.12 | ps | *3 |
| Unit Interval (5 GT/s) | UI | 199.94 | — | 200.06 | ps | *3 |
| DC Differential TX Impedance (2.5 GT/s) | ZTX-DIFF-DC | 80 | — | 120 | Ω | — |
| DC Differential TX Impedance (5 GT/s) | ZTX-DIFF-DC | — | — | 120 | Ω | — |
| DC Differential Input Impedance (2.5 GT/s) | ZRX-DIFF-DC | 80 | — | 120 | Ω | — |
| DC Input Impedance (2.5 & 5 GT/s) | ZRX-DC | 40 | — | 60 | Ω | — |

Notes: 1. RXP, RXN: DC test
2. TXP, TXN: DC test
3. Need Reference Clock (Low Voltage Swing, Differential Clocks). The nominal single-ended swing for each clock is 0 to 0.7 V and a nominal frequency of 100 MHz ± 100 PPM.

Table 73.18.2 PCIE Controller External Clock Accuracy

Conditions: VDD09_PCIE = 0.82 V – 0.07 V / + 0.06 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
VDDQ18_PCIE = 1.8 ± 0.1 V, VDDD_PCIE = 1.03 ± 0.05 V [RZ/G2E], GND = 0 V,
Tc = –40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
Ta = –40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
Tj = –40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Test Item | Symbol | Min. | Typ. | Max. | Unit | Remarks |
|--|--------|------|---------|------|------|--------------------------------------|
| External clock accuracy (PCIE0_CLK_P, PCIE0_CLK_M) | — | — | 100.000 | — | MHz | Frequency accuracy: ±100 ppm or less |

73.19 SCIF

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.19.1 SCIF Signal Timing

Conditions: VDDQ33 = VDDQVA_SDn = 3.3 ± 0.2 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 VDDQ33 = 3.3 ± 0.3 V [RZ/G2E],
 GND = VSS = 0 V, Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], CL = 30 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|---------------------------------------|--------|------|------|------|-------|----------------|
| Input clock cycle (asynchronous) | tSCYC | 4 | — | — | tCYC | Figure 73.19.1 |
| Input clock cycle (synchronous) | tSCYC | 8 | — | — | tCYC | |
| Input clock pulse width | tSCKW | 0.4 | — | 0.6 | tSCYC | |
| Input clock rise time | tSCKr | — | — | 0.8 | tCYC | |
| Input clock fall time | tSCKf | — | — | 0.8 | tCYC | |
| Transmit data delay time | tTXD | — | — | 4 | tCYC | Figure 73.19.2 |
| Receive data setup time (synchronous) | tRXS | 5 | — | — | tCYC | |
| Receive data hold time (synchronous) | tRXH | 2 | — | — | tCYC | |

Note: RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: tCYC is for one cycle of the S3D4φ clock.
 RZ/G2E: tCYC is for one cycle of the S3D4Cφ clock.

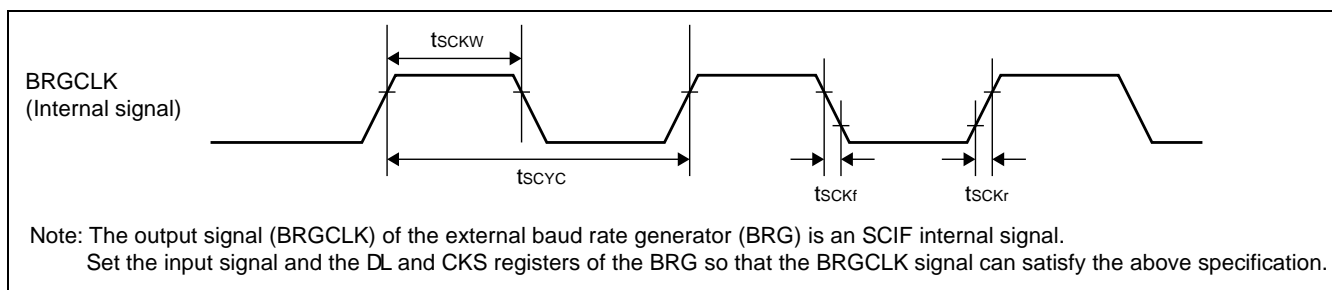


Figure 73.19.1 Input Clock Timing

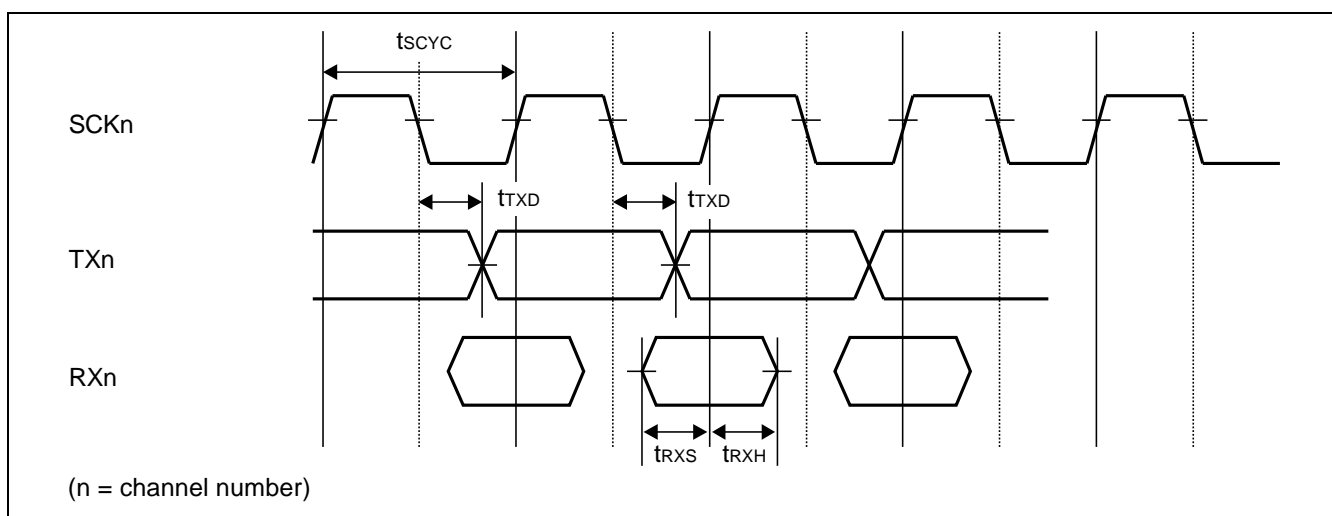


Figure 73.19.2 Input/output Timing in Synchronous Mode

73.20 HSCIF

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.20.1 HSCIF Signal Timing

Conditions: VDDQ33 = 3.3 ± 0.2 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 VDDQ33 = 3.3 ± 0.3 V [RZ/G2E],
 GND = VSS = 0 V, Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3], Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], CL = 30 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|----------------------------------|--------|------|------|------|-------|----------------|
| Input clock cycle (asynchronous) | tSCYC | 4 | — | — | tCYC | Figure 73.20.1 |
| Input clock pulse width | tSCKW | 0.4 | — | 0.6 | tSCYC | |
| Input clock rise time | tSCKr | — | — | 0.8 | tCYC | |
| Input clock fall time | tSCKf | — | — | 0.8 | tCYC | |

Note: RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N: tCYC is for one cycle of the S3D1φ clock.
 RZ/G2E: tCYC is for one cycle of the S3D1Cφ clock.

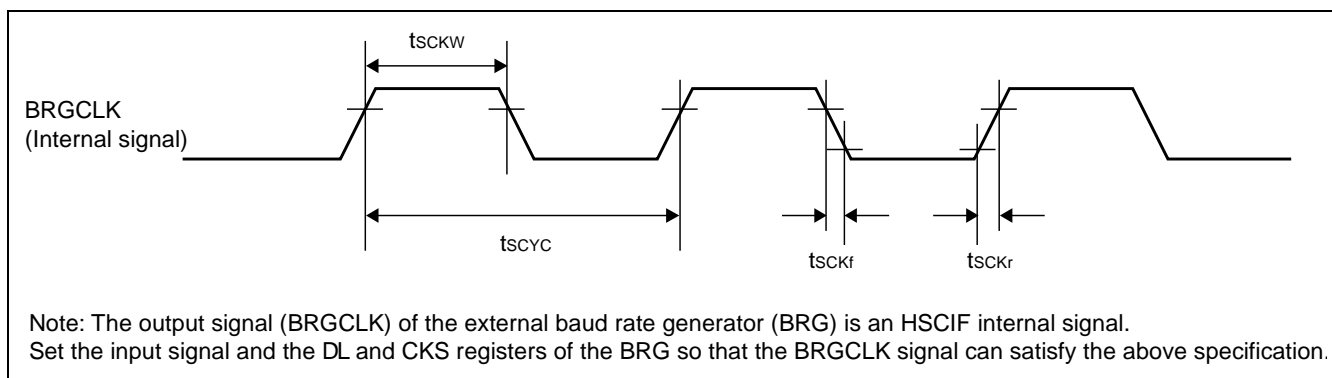


Figure 73.20.1 Input Clock Timing

73.21 I2C

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.21.1 I2C Signal Timing

Conditions: VDDQ33 = 3.3 ± 0.2 V, VDDQ18 = 1.8 V ± 0.1 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 VDDQ33 = 3.3 ± 0.3 V, VDDQ18 = 1.8 V ± 0.1 V, VDDQ_SDn = 3.3 ± 0.3 V / 1.8 V ± 0.1 V [RZ/G2E],
 GND = VSS = 0 V,
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3], Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], CL = 400 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|---|--------|------------------------|------|------|------|----------------|
| I2C_SCL frequency | tICYC | — | — | 400 | kHz | Figure 73.21.1 |
| I2C_SCL low level time | tLOW | 1/(2 * tICYC) - 100 | — | — | ns | |
| I2C_SCL high level time | tHIGH | 600 | — | — | ns | |
| I2C_SCL/I2C_SDA falling time | tICF | — | — | 250 | ns | |
| I2C_SDA bus free time | tICBF | 1300 | — | — | ns | |
| I2C_SCL start condition hold time | tICH | 600 | — | — | ns | |
| I2C_SCL repeat-start condition setup time | tICS | 600 | — | — | ns | |
| I2C_SDA stop condition setup time | tICST | 600 | — | — | ns | |
| I2C_SDA setup time | tDAS | 100 | — | — | ns | |
| I2C_SDA hold time | tICDH | 0 | — | 900 | ns | |

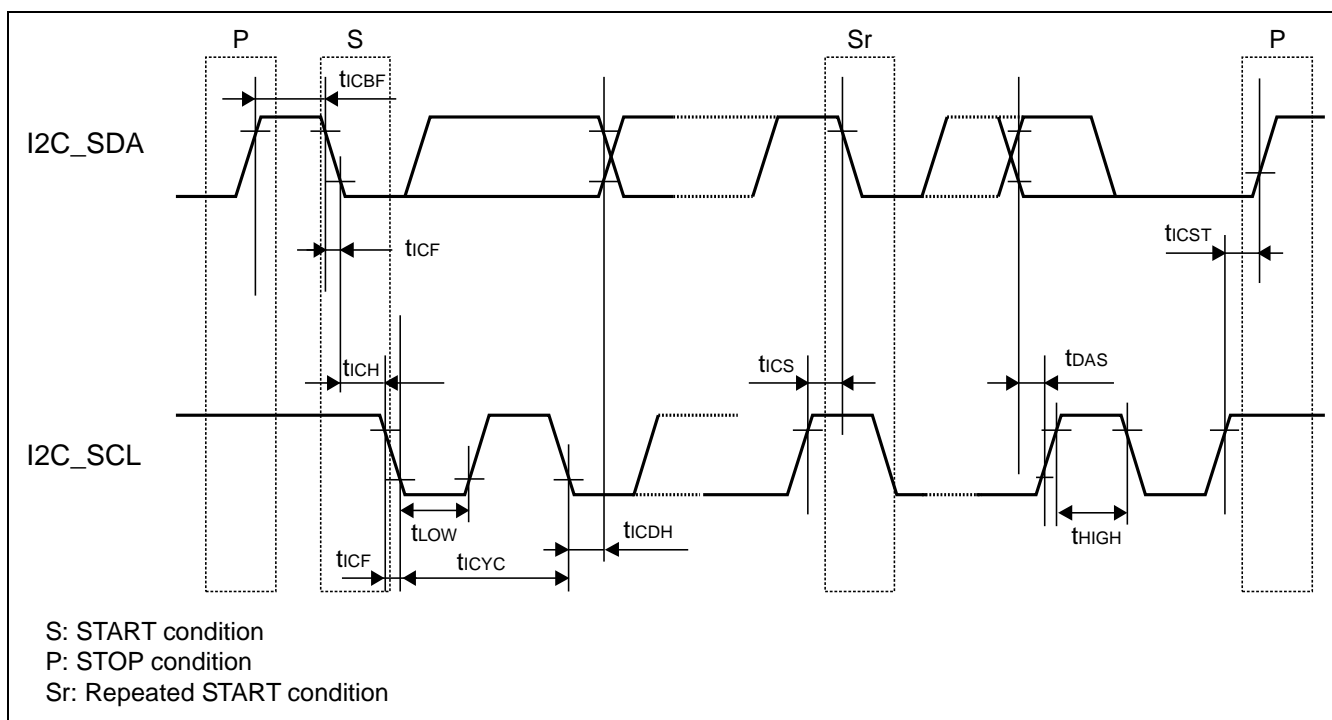


Figure 73.21.1 I2C Signal Timing

73.22 IIC Bus Interface

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.22.1 IIC Bus Interface Signal Timing

Condition: VDDQ33 = 3.3 V ± 0.2 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N] / ± 0.3 V [RZ/G2E] or
 VDDQ18 = 1.8 V ± 0.1 V, GND = VSS = 0 V,
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3], Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], CL = 30 pF

| Item | Symbol | Standard-Mode | | Fast-Mode | | Unit | Figure |
|--|---------|---------------|------|--------------|------|------|----------------|
| | | Min. | Max. | Min. | Max. | | |
| SCLclock frequency | fSCL | 0 | 100 | 0 | 400 | kHz | Figure 73.22.1 |
| Hold time (after repeat START condition, first clock pulse is generated) | tHD;STA | 4.0 | — | 0.6 | — | µs | |
| L period in SCL clock | tLOW | 4.7 | — | 1.3 | — | µs | |
| H period in SCL clock | tHIGH | 4.0 | — | 0.6 | — | µs | |
| Setup time for repeat START condition | tSU;STA | 4.7 | — | 0.6 | — | µs | |
| Date hold time | tHD;DAT | 0 | 3.45 | 0 | 0.9 | µs | |
| Data setup time | tSU;DAT | 250 | — | 100 | — | ns | |
| SDA and SCLsignal rise time | tr | — | 1000 | — | 300 | ns | |
| SDA and SCL signal fall time | tf | — | 300 | — | 300 | ns | |
| Setup time for STOP condition | tSU;STO | 4.0 | — | 0.6 | — | µs | |
| Bus free time between STOP and START conditions | tBUF | 4.7 | — | 1.3 | — | µs | |
| Noise margin at low level of each connected device (including hysteresis) | VnL | 0.1 × VDDQ*4 | — | 0.1 × VDDQ*4 | — | V | |
| Noise margin at high level of each connected device (including hysteresis) | VnH | 0.2 × VDDQ*4 | — | 0.2 × VDDQ*4 | — | V | |
| Spike pulse width suppressed by the input filter | tSP | — | — | 0 | 50 | ns | |

- Notes. 1. All values are referenced at VDDQ*4 × 0.3 and VDDQ*4 × 0.7 levels.
 2. To satisfy the I2C-bus specification, pull-up resistors (Rp) with the appropriate resistance must be included depending on the total of bus capacitive load of each line.
 Pull-up resistor range should be following, refer Table 73.21.2.
 $Rp(max) = tr / (0.8473 * Cb)$
 $Rp(min) = (VDDQ*4 - VOL(max)) / IOL$
 3. The total capacity (Cb) should be less than or equal to 100 pF.
 4. VDDQ means VDDQ33 or VDDQ18

Table 73.22.2 Pull-up Resistor Range of IIC Bus Interface

| Symbol | VCCQ Voltage | Standard-Mode | | Fast-Mode | | Unit | Remarks |
|---------------------|-----------------|---------------|-------|-----------|------|------|---------|
| | | Min. | Max. | Min. | Max. | | |
| Rp (Cb = 100 pF) | Vpullup = 1.8 V | 517 | 11802 | 517 | 3540 | Ω | — |
| | Vpullup = 3.3 V | 1067 | 11802 | 1067 | 3540 | Ω | — |

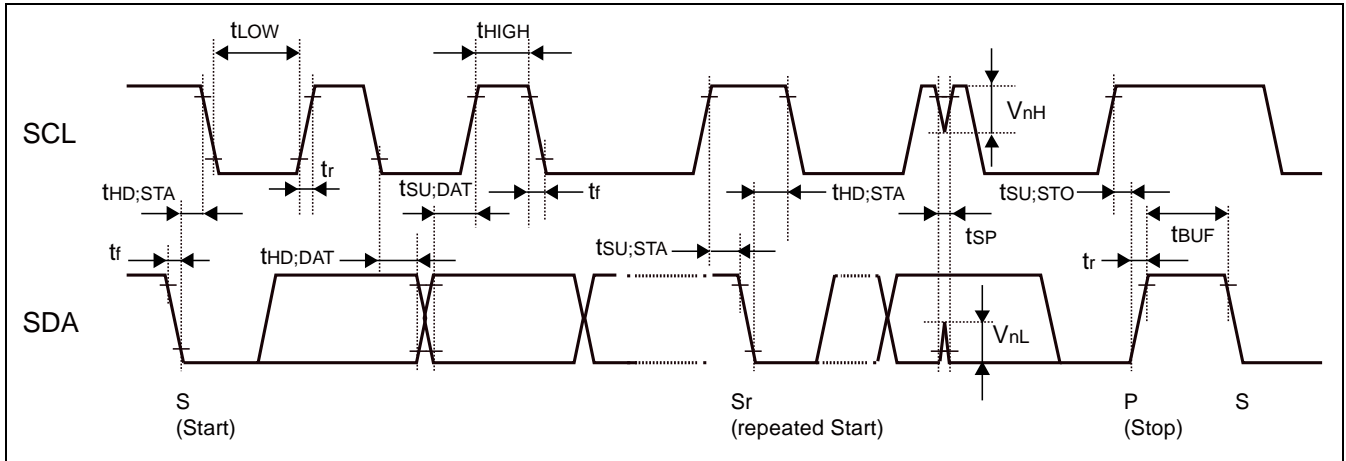


Figure 73.22.1 IIC Bus Interface Signal Timing

73.23 MSIOF

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.23.1 MSIOF Module Signal Timing

Conditions: VDDQ33 = 3.3 V ± 0.2 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 VDDQ33 = 3.3 V ± 0.3 V [RZ/G2E],
 GND = VSS = 0 V, Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], CL = 30 pF

[RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Item | Symbol | Min. | Max. | Unit | Figures |
|--|--------|--------------|------|------|-------------------------------|
| MSIOF_SCK clock cycle time (when master TX or slave RX) | tMSCYC | 4 × tpcyc* | — | ns | Figure 73.23.1, to 73.23.4 |
| MSIOF_SCK clock cycle time (when master RX or slave TX) | tMSCYC | 8 × tpcyc* | — | ns | |
| MSIOF_SCK output clock high-level width | tMSWHO | 0.4 × tMSCYC | — | ns | |
| MSIOF_SCK output clock low-level width | tMSWLO | 0.4 × tMSCYC | — | ns | |
| MSIOF_SCK input high-level width | tMSWHI | 0.4 × tMSCYC | — | ns | |
| MSIOF_SCK input low-level width | tMSWLI | 0.4 × tMSCYC | — | ns | |
| MSIOF_SYNC input setup time in slave mode | tTSFSS | 6 | — | ns | |
| MSIOF_SYNC input hold time in slave mode | tTSFSH | 2 | — | ns | |
| MSIOF_TXD output delay time1 in slave mode | tTSDD1 | 1 | 20 | ns | |
| MSIOF_TXD output delay time2 in slave mode | tTSDD2 | 1 | 16 | ns | |
| MSIOF_RXD input setup time in slave mode | tTSRDS | 6 | — | ns | |
| MSIOF_RXD input hold time in slave mode | tTSRDH | 5 | — | ns | |
| MSIOF_SYNC output delay time in master mode | tTMSFD | -2 | 8 | ns | |
| MSIOF_TXD output delay time in master mode | tTMDD | -2 | 8 | ns | |
| MSIOF_RXD input setup time in master mode | tTMRDS | 10 | — | ns | |
| MSIOF_RXD input hold time in master mode | tTMRDH | 2 | — | ns | |

Note: * tpcyc is a cycle time of peripheral clock (MSOφ).

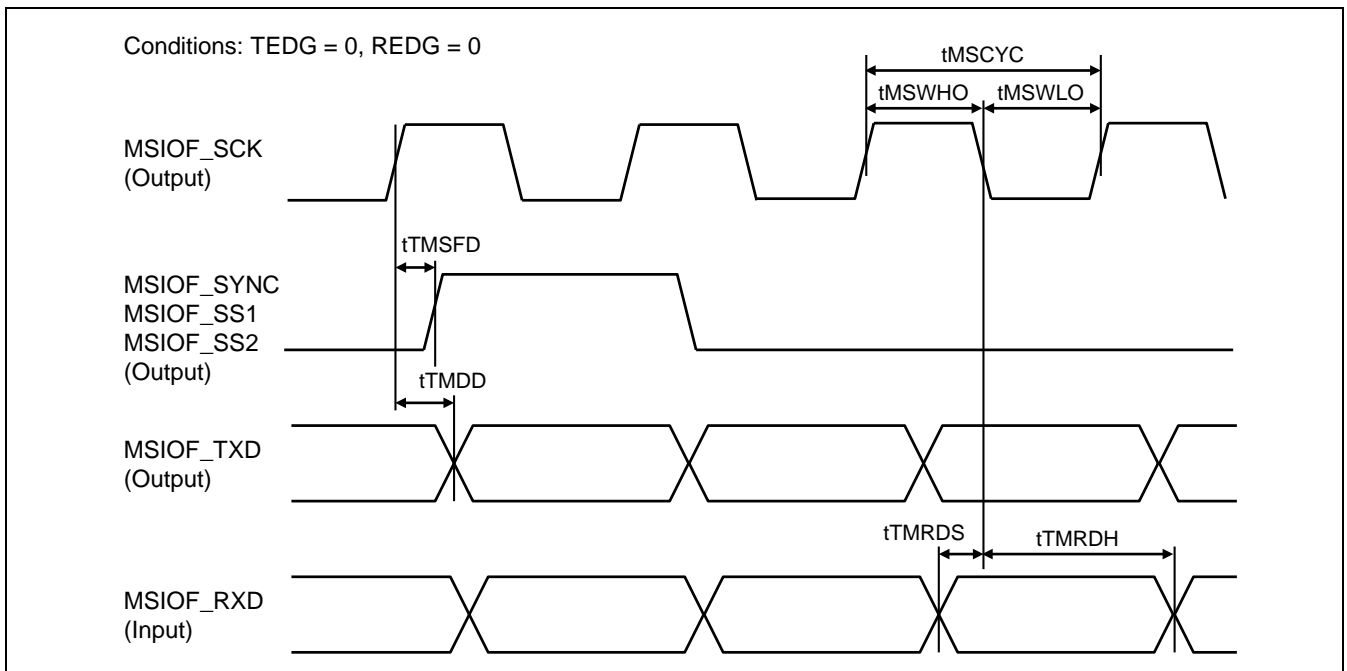


Figure 73.23.1 MSIOF Timing (Master Mode)

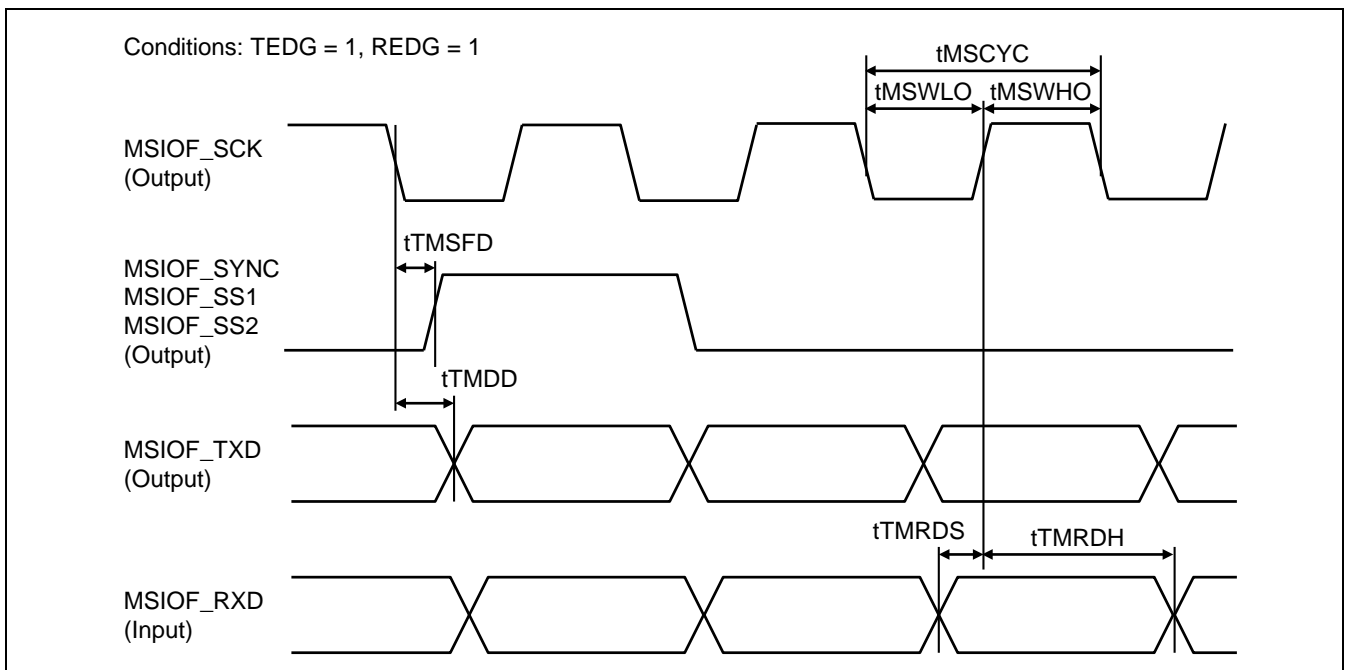


Figure 73.23.2 MSIOF Timing (Master Mode)

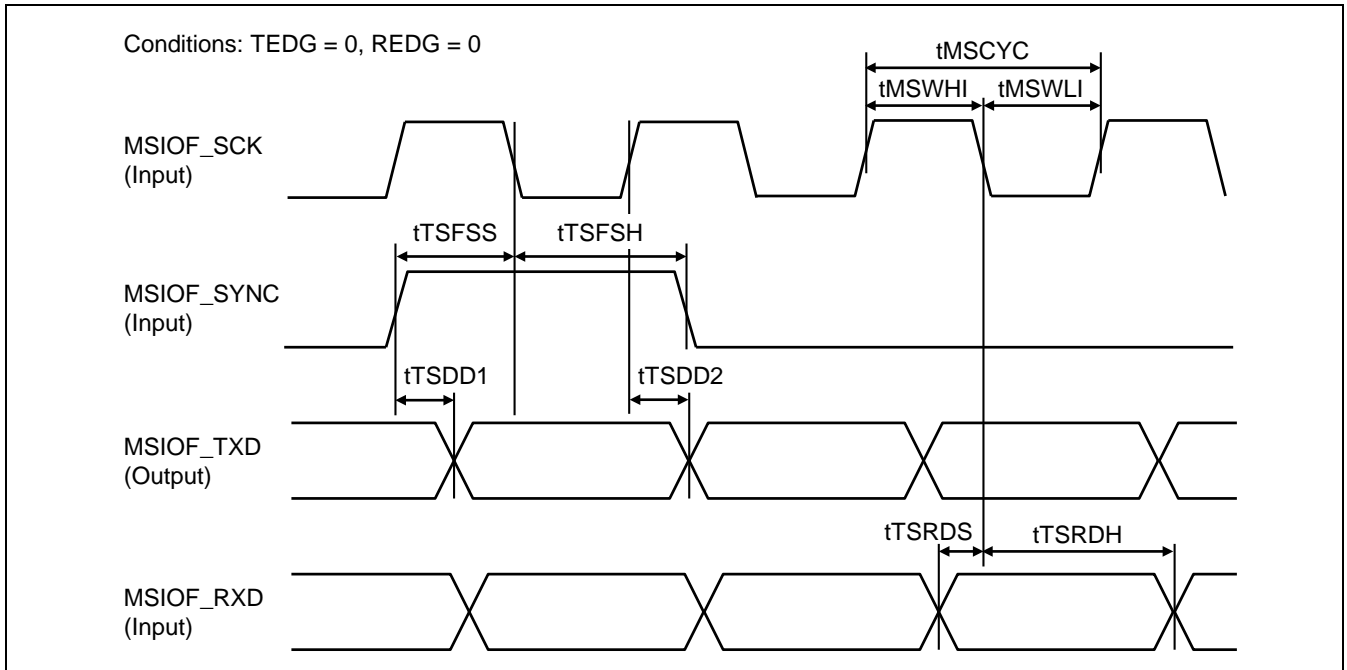


Figure 73.23.3 MSIOF Timing (Slave Mode)

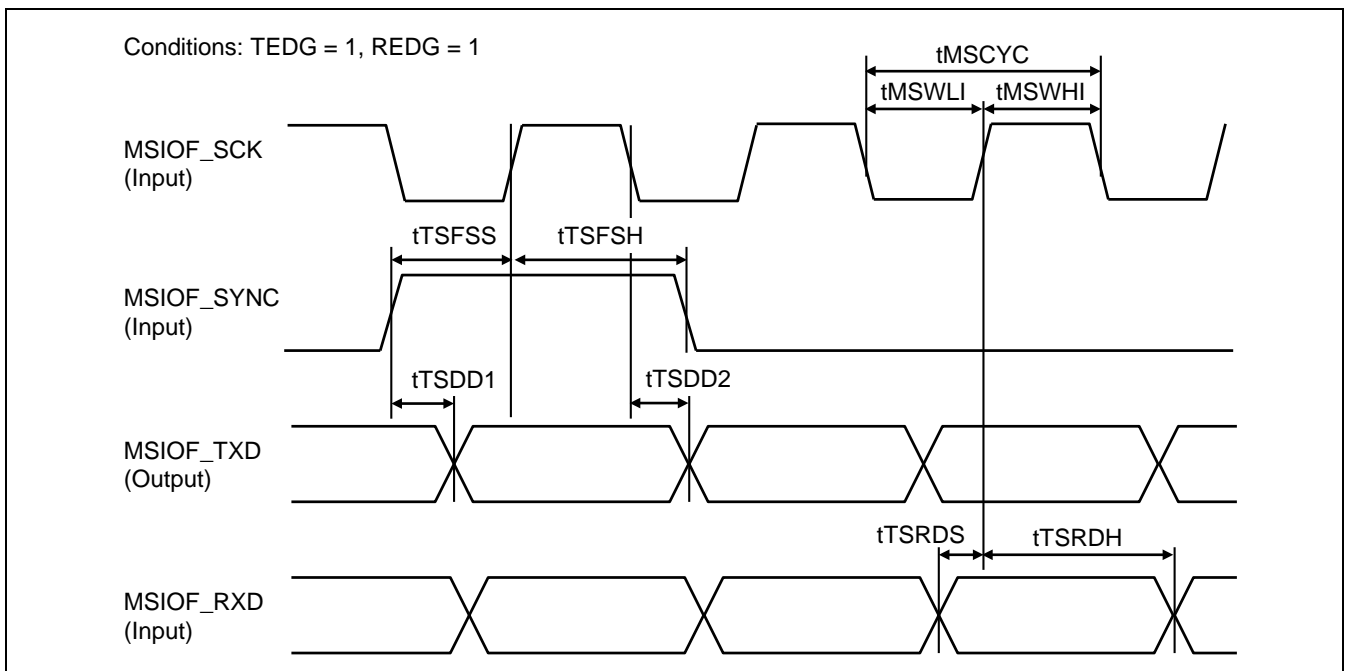


Figure 73.23.4 MSIOF Timing (Slave Mode)

73.24 RPC Interface

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.24.1 RPC Timing [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N]

Conditions: $V_{DDQ18} = 1.8V \pm 0.1V$, $GND = VSS = 0V$,
 $T_c = -40$ to $+115$ °C [RZ/G2M V1.3], $T_a = -40$ to $+85$ °C [RZ/G2M V3.0, RZ/G2N],
 $T_j = -40$ to $+115$ °C [RZ/G2M V3.0, RZ/G2N],
 $CL = 15$ pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|-------------------------------------|---------|----------------------------|------|----------------------------|--------|-----------------------------------|
| SPCLK clock cycle time (HyperFlash) | tSPCYC | 6.25 | — | — | ns | Figure 73.24.1 |
| SPCLK clock cycle time (QSPI Flash) | | 12.50 | — | — | ns | |
| SPCLK high pulse width | tSPWH | 0.45 | — | 0.55 | tSPCYC | |
| SPCLK low pulse width | tSPWL | 0.45 | — | 0.55 | tSPCYC | |
| SPCLK rise time | tSPR | — | — | 2.0 | ns | |
| SPCLK fall time | tSPF | — | — | 2.0 | ns | |
| SPCLK skew | tSPSKEW | — | — | 0.04 | tSPCYC | |
| SSL to setup time | tLEAD | $1 \times tSPCYC - 3$ | — | $8 \times tSPCYC + 3$ | ns | Figure 73.24.2, |
| SSL to hold time | tLAG | $5.5 \times tSPCYC - 3$ | — | $8.5 \times tSPCYC + 3$ | ns | Figure 73.24.3, Figure 73.24.4 |
| Data input valid setup time SDR | tVSS | — | — | $0.5 \times tSPCYC + 3$ | ns | Figure 73.24.2 |
| Data input valid hold time SDR | tVHS | 0 | — | — | ns | |
| Data output setup time | tOS | $0.5 \times tSPCYC - 2$ | — | — | ns | |
| Data output hold time | tOH | $0.5 \times tSPCYC - 2$ | — | — | ns | |
| Data input valid setup time DDR | tVSD | — | — | 6.5 | ns | Figure 73.24.3 |
| Data input valid hold time DDR | tVHD | $0.32 \times tSPCYC$ | — | — | ns | |
| Data to Data Strobe Skew | tDDSS | -0.65 | — | 0.65 | ns | Figure 73.24.4 |
| SPCLK to Data Strobe Skew | tCDSS | — | — | 7.0 | ns | |
| Output Data Setup time in DDR mode | tSDDR | $0.25 \times tSPCYC - 0.6$ | — | $0.25 \times tSPCYC + 0.6$ | ns | Figure 73.24.3, Figure 73.24.4 |
| Output Data Hold time in DDR mode | tHDDR | $0.25 \times tSPCYC - 0.6$ | — | $0.25 \times tSPCYC + 0.6$ | ns | |
| RPC_RESET assert time | tRPCRA | 0 | — | — | ns | Figure 73.24.5 |
| RPC_RESET negate time | tRPCRN | 10 | — | — | ns | Figure 73.24.5 |

Note: The RZ/G2M V1.3 does not support DDR operation with QSPI (1-/4-bit) and QSPIx2 (8-bit). Therefore, the following DDR related specifications with QSPI and QSPIx2 cannot be applied to the RZ/G2M V1.3.

Table 73.24.2 RPC Timing [RZ/G2E]

Conditions: $VDDQ_QSPI = 1.8V \pm 0.1 V$, $GND = VSS = 0 V$,
 $T_a = -40$ to $+85$ °C
 $T_j = -40$ to $+115$ °C
 $CL = 15$ pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|-------------------------------------|---------|----------------------------|------|----------------------------|--------|--|
| SPCLK clock cycle time (HyperFlash) | tSPCYC | 6.66 | — | — | ns | Figure 73.24.1 |
| SPCLK clock cycle time (QSPI Flash) | | 12.50 | — | — | ns | |
| SPCLK high pulse width | tSPWH | 0.45 | — | 0.55 | tSPCYC | |
| SPCLK low pulse width | tSPWL | 0.45 | — | 0.55 | tSPCYC | |
| SPCLK rise time | tSPR | — | — | 2.0 | ns | |
| SPCLK fall time | tSPF | — | — | 2.0 | ns | |
| SPCLK skew | tSPSKEW | — | — | 0.04 | tSPCYC | |
| SSL to setup time | tLEAD | $1 \times tSPCYC - 3$ | — | $8 \times tSPCYC + 3$ | ns | Figure 73.24.2, Figure 73.24.3, Figure 73.24.4 |
| SSL to hold time | tLAG | $5.5 \times tSPCYC - 3$ | — | $8.5 \times tSPCYC + 3$ | ns | |
| Data input valid setup time SDR | tVSS | — | — | $0.5 \times tSPCYC + 3$ | ns | Figure 73.24.2 |
| Data input valid hold time SDR | tVHS | 0 | — | — | ns | |
| Data output setup time | tOS | $0.5 \times tSPCYC - 2$ | — | — | ns | |
| Data output hold time | tOH | $0.5 \times tSPCYC - 2$ | — | — | ns | |
| Data input valid setup time DDR | tVSD | — | — | 6.5 | ns | Figure 73.24.3 |
| Data input valid hold time DDR | tVHD | $0.32 \times tSPCYC$ | — | — | ns | |
| Data to Data Strobe Skew | tDDSS | -0.65 | — | 0.65 | ns | Figure 73.24.4 |
| SPCLK to Data Strobe Skew | tCDSS | — | — | 7.0 | ns | |
| Output Data Setup time in DDR mode | tSDDR | $0.25 \times tSPCYC - 0.6$ | — | $0.25 \times tSPCYC + 0.6$ | ns | Figure 73.24.3, Figure 73.24.4 |
| Output Data Hold time in DDR mode | tHDDR | $0.25 \times tSPCYC - 0.6$ | — | $0.25 \times tSPCYC + 0.6$ | ns | |
| RPC_RESET assert time | tRPCRA | 0 | — | — | ns | Figure 73.24.5 |
| RPC_RESET negate time | tRPCRN | 10 | — | — | ns | Figure 73.24.5 |

Table 73.24.3 RPC Timing [RZ/G2E]

Conditions: VDDQ_QSPI = 3.3 V ± 0.3 V, GND = VSS = 0 V, Ta = -40 to +85 °C, Tj = -40 to +115 °C, CL = 15 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|-------------------------------------|---------|---------------------|------|---------------------|--------|-----------------------------------|
| SPCLK clock cycle time (HyperFlash) | tSPCYC | 10.00 | — | — | ns | Figure 73.24.1 |
| SPCLK clock cycle time (QSPI Flash) | | 12.50 | — | — | ns | |
| SPCLK high pulse width | tSPWH | 0.45 | — | 0.55 | tSPCYC | |
| SPCLK low pulse width | tSPWL | 0.45 | — | 0.55 | tSPCYC | |
| SPCLK rise time | tSPR | — | — | 2.0 | ns | |
| SPCLK fall time | tSPF | — | — | 2.0 | ns | |
| SPCLK skew | tSPSKEW | — | — | 0.04 | tSPCYC | |
| SSL to setup time | tLEAD | 1 × tSPCYC – 3 | — | 8 × tSPCYC + 3 | ns | Figure 73.24.2, |
| SSL to hold time | tLAG | 5.5 × tSPCYC – 3 | — | 8.5 × tSPCYC + 3 | ns | Figure 73.24.3, Figure 73.24.4 |
| Data input valid setup time SDR | tVSS | — | — | 0.5 × tSPCYC + 3 | ns | Figure 73.24.2 |
| Data input valid hold time SDR | tVHS | 0 | — | — | ns | |
| Data output setup time | tOS | 0.5 × tSPCYC – 2 | — | — | ns | |
| Data output hold time | tOH | 0.5 × tSPCYC – 2 | — | — | ns | |
| Data input valid setup time DDR | tVSD | — | — | 6.5 | ns | Figure 73.24.3 |
| Data input valid hold time DDR | tVHD | 0.32 × tSPCYC | — | — | ns | |
| Data to Data Strobe Skew | tDDSS | -0.65 | — | 0.65 | ns | Figure 73.24.4 |
| SPCLK to Data Strobe Skew | tCDSS | — | — | 7.0 | ns | |
| Output Data Setup time in DDR mode | tSDDR | 0.25 × tSPCYC – 0.6 | — | 0.25 × tSPCYC + 0.6 | ns | Figure 73.24.3, Figure 73.24.4 |
| Output Data Hold time in DDR mode | tHDDR | 0.25 × tSPCYC – 0.6 | — | 0.25 × tSPCYC + 0.6 | ns | |
| RPC_RESET assert time | tRPCRA | 0 | — | — | ns | Figure 73.23.5 |
| RPC_RESET negate time | tRPCRN | 10 | — | — | ns | Figure 73.24.5 |

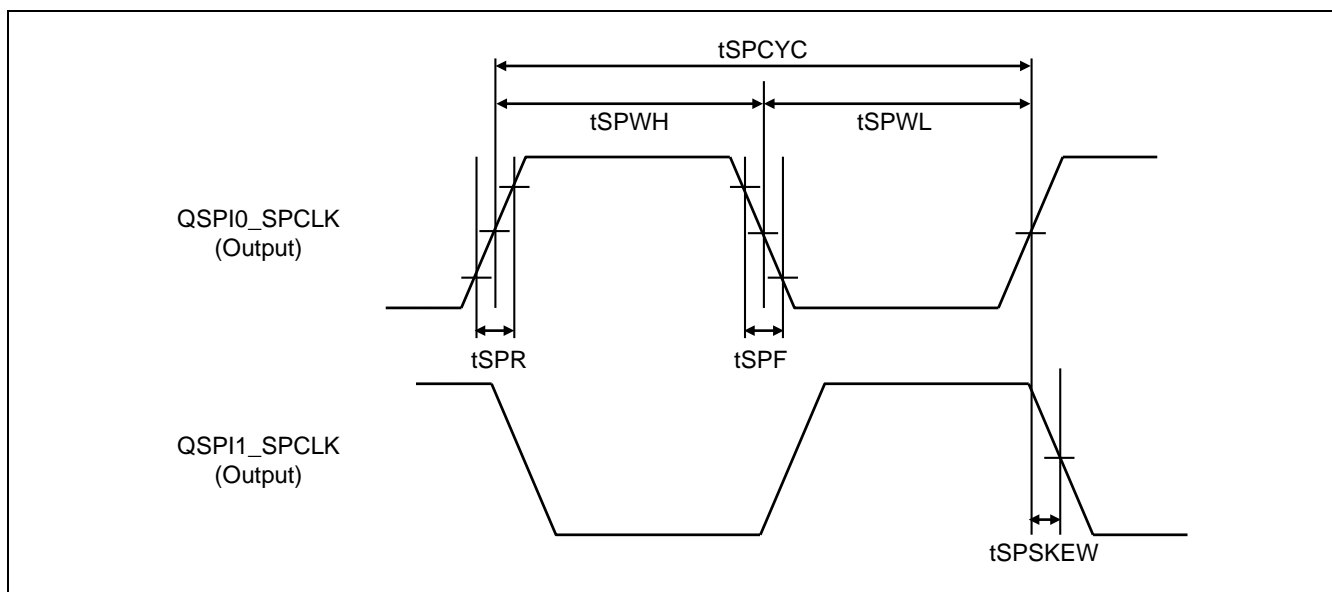


Figure 73.24.1 RPC Clock Timing

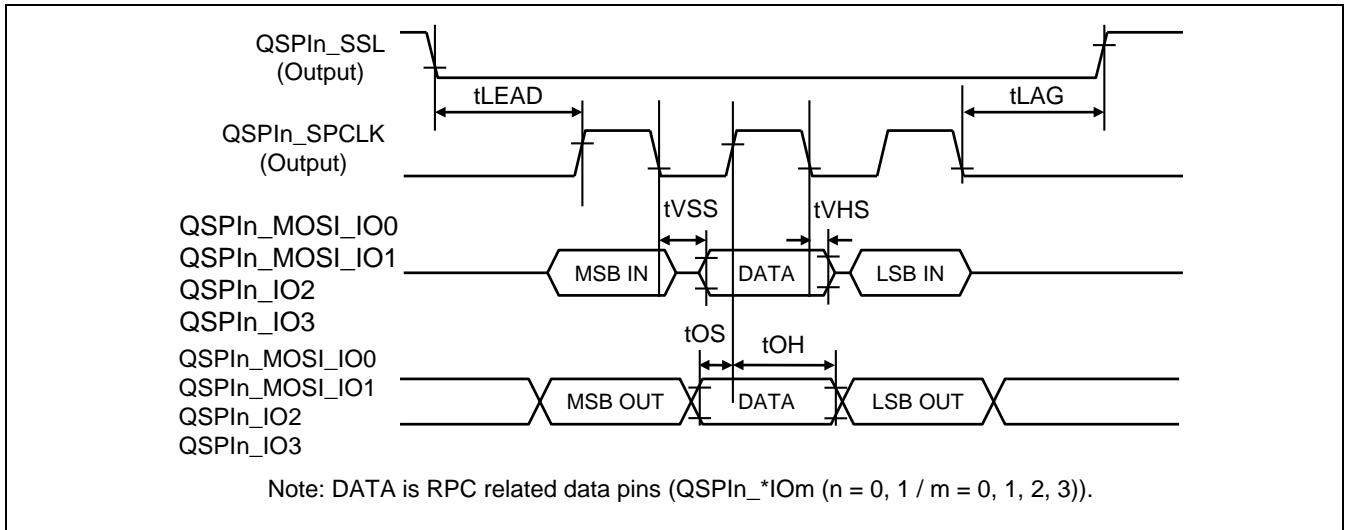


Figure 73.24.2 RPC SDR Operation Timing (QSPI Flash)

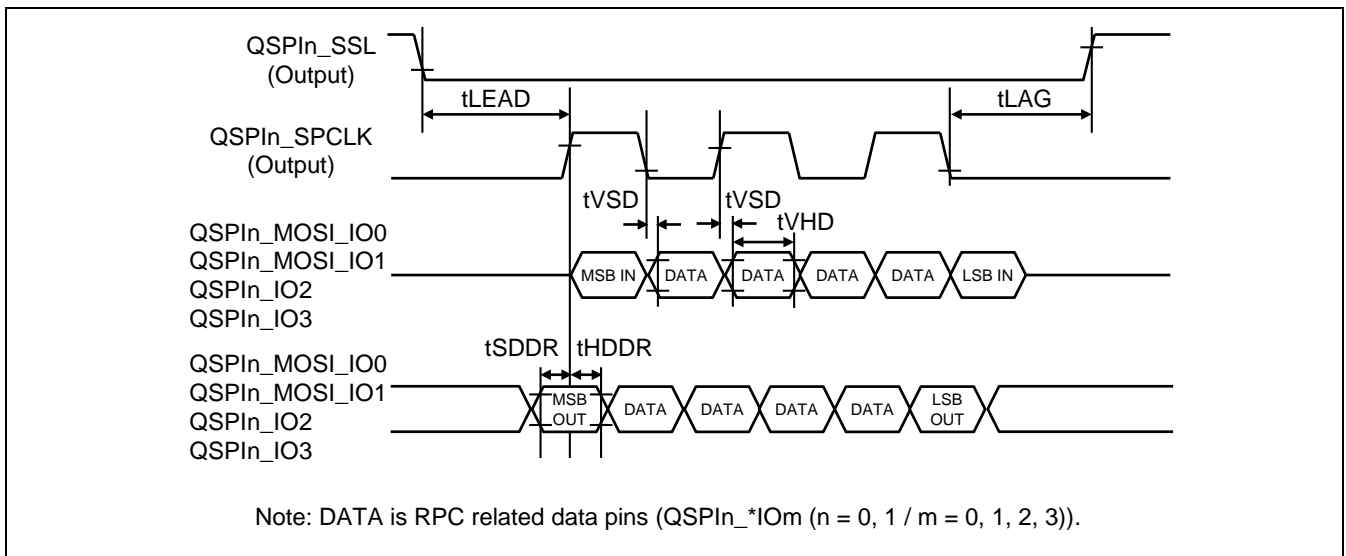


Figure 73.24.3 RPC DDR Operation Timing (QSPI Flash)

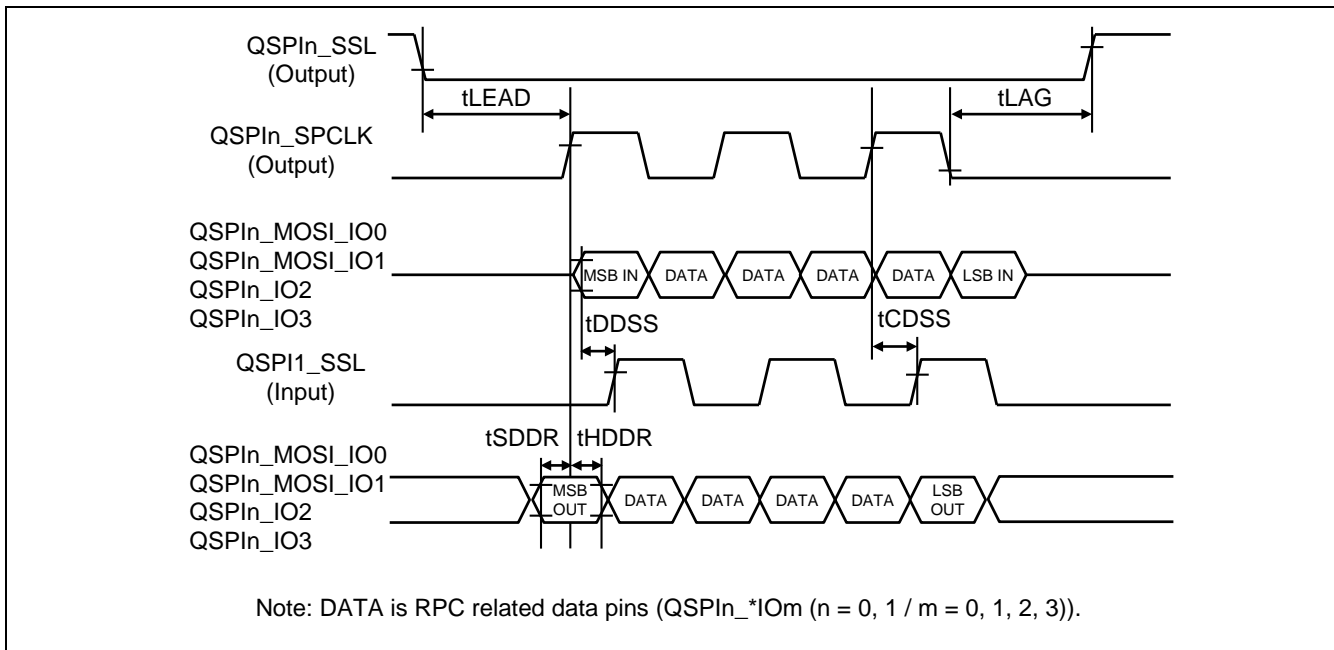


Figure 73.24.4 RPC DDR Operation Timing (HyperFlash)

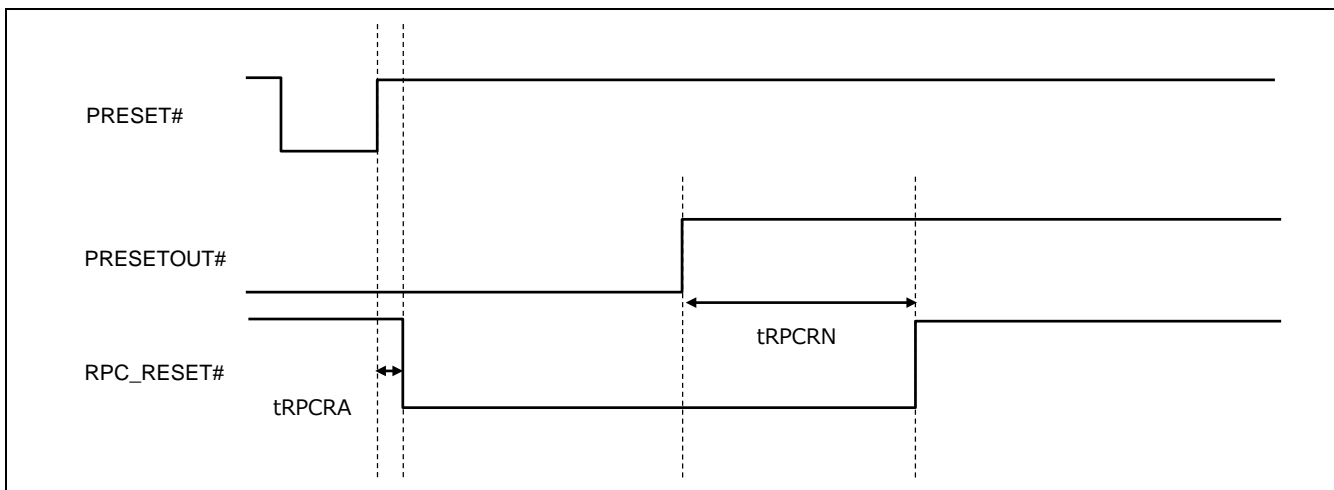


Figure 73.24.5 RPC_RESET# Operation Timing

73.25 SD Host Interface (SDHI)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

73.25.1 SDHI (3.3V) Electrical Characteristics

Table 73.25.1 SDHI Signal Timing (Default mode)

Conditions: VDDQVA_SDn (n = 0 to 3) = 3.3 V ± 0.2 V, GND = VSS = 0 V,
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3], Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N],
 VDDQ_SDn (n = 0, 1, 3) = 3.3V ± 0.3 V, GND = VSS = 0 V,
 Ta = -40 to + 85 °C [RZ/G2E], Tj = -40 to + 115 °C [RZ/G2E],
 CL = 40 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|------------------------------|----------|------|------|------|------|----------------|
| SDCLK clock cycle time | tSDCYC | 40.0 | — | — | ns | Figure 73.25.1 |
| SDCMD output data delay time | tSDCMD_S | — | — | 10.0 | ns | |
| | tSDCMD_H | 6.0 | — | — | ns | |
| SDDAT output data delay time | tSDDAD_S | — | — | 10.0 | ns | |
| | tSDDAD_H | 6.0 | — | — | ns | |
| SDCMD input data setup time | tSDCMS | 3.0 | — | — | ns | |
| SDCMD input data hold time | tSDCMH | 2.0 | — | — | ns | |
| SDDAT input data setup time | tSDDAS | 3.0 | — | — | ns | |
| SDDAT input data hold time | tSDDAH | 2.0 | — | — | ns | |

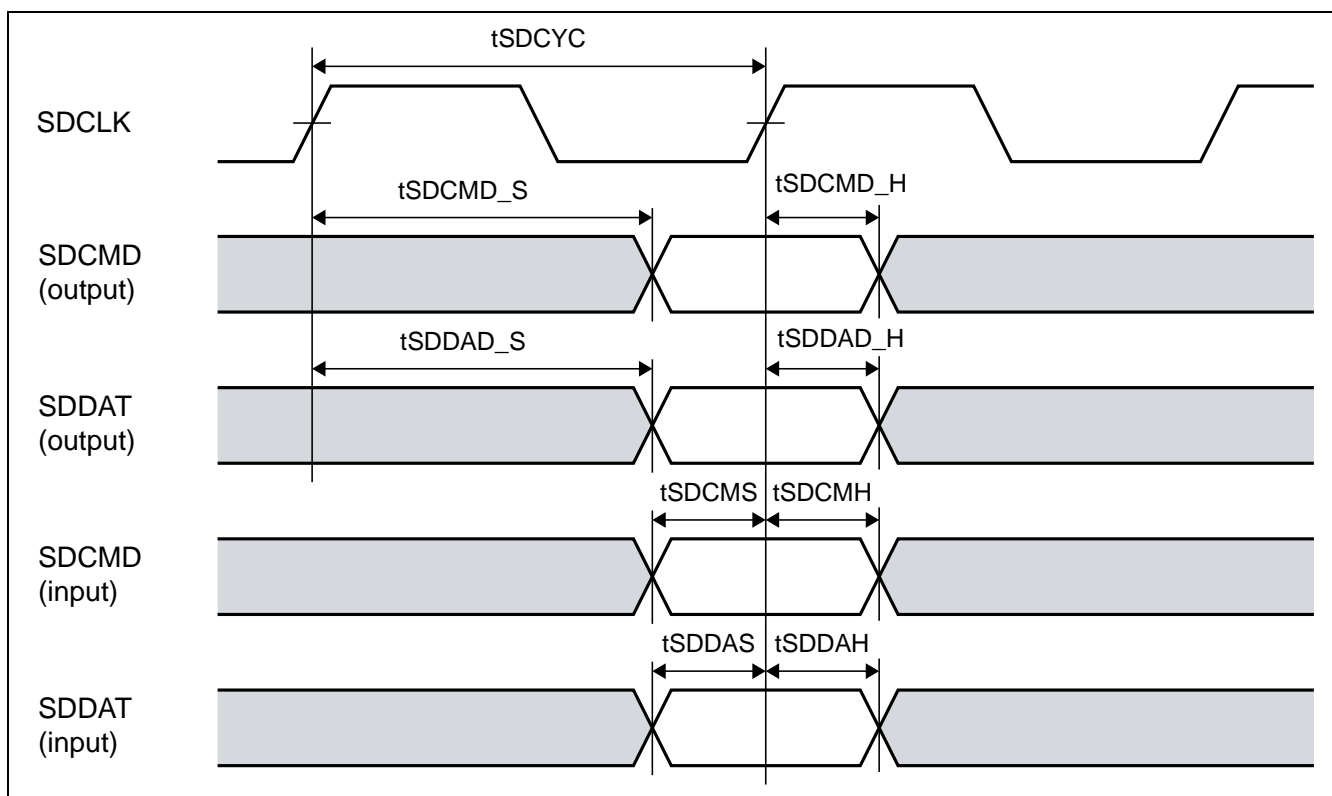


Figure 73.25.1 SDHI Signal Timing (Default mode)

Table 73.25.2 SDHI Signal Timing (High Speed mode)

Conditions: VDDQVA_SDn (n = 0 to 3) = 3.3 V ± 0.2 V, GND = VSS = 0 V,
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N], Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N],
 VDDQ_SDn (n = 0, 1, 3) = 3.3V ± 0.3 V, GND = VSS = 0 V,
 Ta = -40 to +85 °C [RZ/G2E], Tj = -40 to +115 °C [RZ/G2E],
 CL = 40 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|------------------------------|----------|------|------|--------------------|------|----------------|
| SDCLK clock cycle time | tSDCYC | 20.0 | — | — | ns | Figure 73.25.2 |
| SDCMD output data delay time | tSDCMD | 3.0 | — | 0.5 × tSDCYC + 3.0 | ns | |
| SDCMD output data hold time | tSDCMDOH | 3.0 | — | — | ns | |
| SDDAT output data delay time | tSDDAD | 3.0 | — | 0.5 × tSDCYC + 3.0 | ns | |
| SDDAT output data hold time | tSDDATOH | 3.0 | — | — | ns | |
| SDCMD input data setup time | tSDCMS | 3.0 | — | — | ns | |
| SDCMD input data hold time | tSDCMH | 2.0 | — | — | ns | |
| SDDAT input data setup time | tSDDAS | 3.0 | — | — | ns | |
| SDDAT input data hold time | tSDDAH | 2.0 | — | — | ns | |

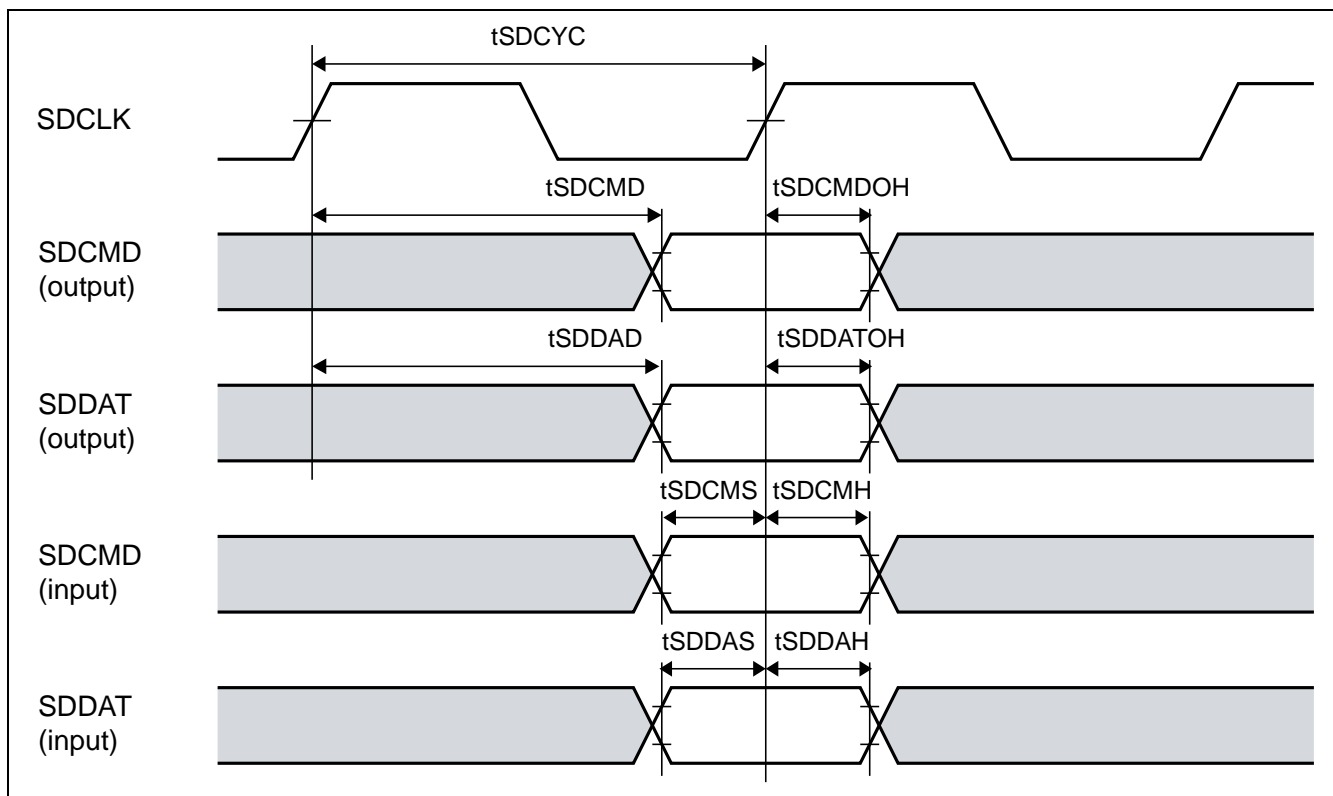


Figure 73.25.2 SDHI Signal Timing (High Speed mode)

73.25.2 SDHI (1.8V) Electrical Characteristics

Table 73.25.3 SDHI Signal Timing (SDR50 mode)

Conditions: V_{DDQVA_SDn} ($n = 0$ to 3) = $1.8\text{ V} \pm 0.1\text{ V}$, $GND = VSS = 0\text{ V}$,
 $T_c = -40$ to $+115\text{ }^\circ\text{C}$ [RZ/G2H, RZ/G2M V1.3],
 $T_a = -40$ to $+85\text{ }^\circ\text{C}$ [RZ/G2M V3.0, RZ/G2N], $T_j = -40$ to $+115\text{ }^\circ\text{C}$ [RZ/G2M V3.0, RZ/G2N],
 V_{DDQ_SDn} ($n = 0, 1, 3$) = $1.8\text{ V} \pm 0.1\text{ V}$, $GND = VSS = 0\text{ V}$,
 $T_a = -40$ to $+85\text{ }^\circ\text{C}$ [RZ/G2E], $T_j = -40$ to $+115\text{ }^\circ\text{C}$ [RZ/G2E],
 $CL = 30\text{ pF}$

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|------------------------------|--------|------|------|------------|------|----------------|
| SDCLK clock cycle time | tSDCYC | 10.0 | — | — | ns | Figure 73.25.3 |
| SDCMD output data delay time | tSDCMD | 1.3 | — | tSDCYC - 4 | ns | |
| SDDAT output data delay time | tSDDAD | 1.3 | — | tSDCYC - 4 | ns | |
| SDCMD input data setup time | tSDCMS | 2.0 | — | — | ns | |
| SDCMD input data hold time | tSDCMH | 1.5 | — | — | ns | |
| SDDAT input data setup time | tSDDAS | 2.0 | — | — | ns | |
| SDDAT input data hold time | tSDDAH | 1.5 | — | — | ns | |

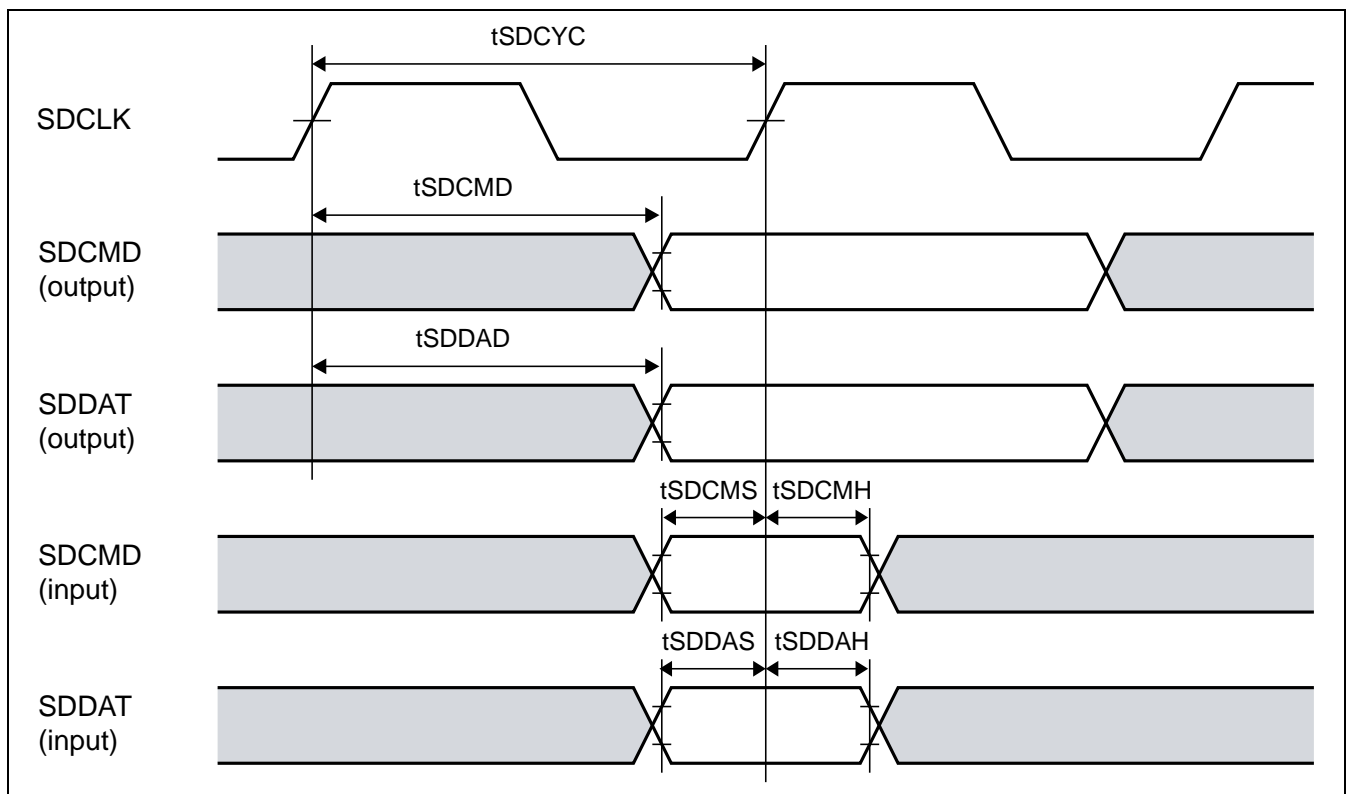


Figure 73.25.3 SDHI Signal Timing (SDR50 mode)

Table 73.25.4 SDHI Signal Timing (SDR104 mode)

Conditions: VDDQVA_SDn (n = 0 to 3) = 1.8 V ± 0.1 V, GND = VSS = 0 V,
 Tc = -20 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -20 to +85 °C [RZ/G2M V3.0, RZ/G2N], Tj = -20 to +115 °C [RZ/G2M V3.0, RZ/G2N]
 VDDQ_SDn (n = 0, 1, 3) = 1.8V ± 0.1 V, GND = VSS = 0 V,
 Ta = -40 to +85 °C [RZ/G2E], Tj = -40 to +115 °C [RZ/G2E], CL = 10 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|-------------------------------------|--------|------|------|------|------|----------------|
| SDCLK clock cycle time | tSDCYC | 5.0 | — | — | ns | Figure 73.25.4 |
| SDCMD output data delay time | tSDCMD | 1.30 | — | 3.50 | ns | |
| SDDAT output data delay time | tSDDAD | 1.30 | — | 3.50 | ns | |
| SDCMD input data width (for tuning) | tSDICW | 2.57 | 3.07 | — | ns | |
| (after tuning) | | 2.57 | — | — | ns | |
| SDDAT input data width (for tuning) | tSDIDW | 2.57 | 3.07 | — | ns | |
| (after tuning) | | 2.57 | — | — | ns | |

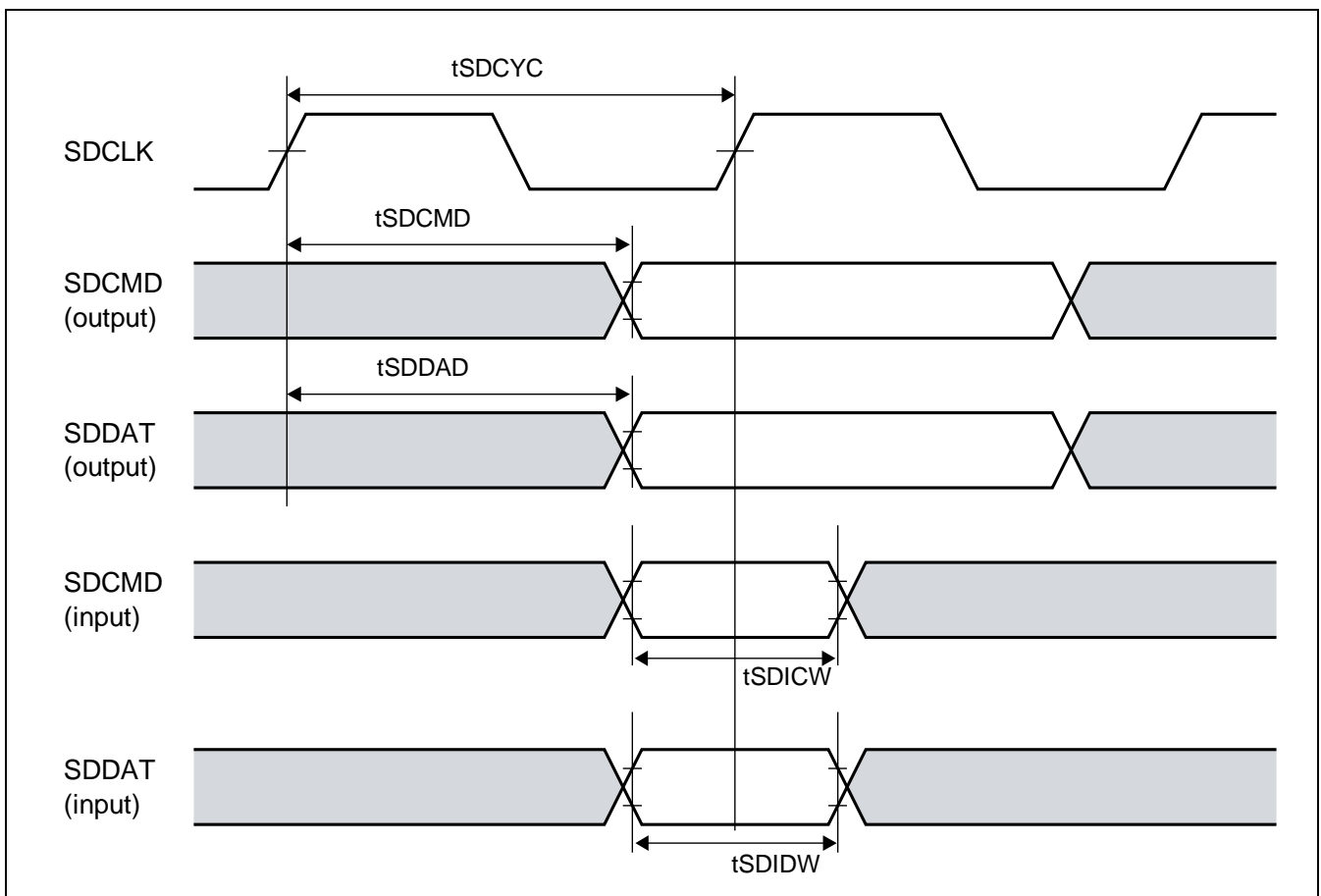


Figure 73.25.4 SDHI Signal Timing (SDR104 mode)

73.26 MMC Interface

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

73.26.1 MMC (3.3V) Electrical Characteristics [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E]

Table 73.26.1 MMC Signal Timing (High Speed mode)

Conditions: VDDQVA_SDn (n = 2, 3) = 3.3 V ± 0.2 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 VDDQ_SD3 = 3.3 V ± 0.3 V [RZ/G2E], GND = VSS = 0 V,
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3], Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], CL = 40 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|-------------------------------|-----------|------|------|---------------------|------|----------------|
| MMCCLK clock cycle time | tMMCCYC | 20.0 | — | — | ns | Figure 73.26.1 |
| MMCCMD output data delay time | tMMCCMD | 3.0 | — | 0.5 × tMMCCYC + 3.0 | ns | |
| MMCCMD output data hold time | tMMCCMDOH | 3.0 | — | — | ns | |
| MMCDAT output data delay time | tMMCDAD | 3.0 | — | 0.5 × tMMCCYC + 3.0 | ns | |
| MMCDAT output data hold time | tMMCDATOH | 3.0 | — | — | ns | |
| MMCCMD input data setup time | tMMCCMS | 3.0 | — | — | ns | |
| MMCCMD input data hold time | tMMCCMH | 2.0 | — | — | ns | |
| MMCDAT input data setup time | tMMCDAS | 3.0 | — | — | ns | |
| MMCDAT input data hold time | tMMCDAH | 2.0 | — | — | ns | |

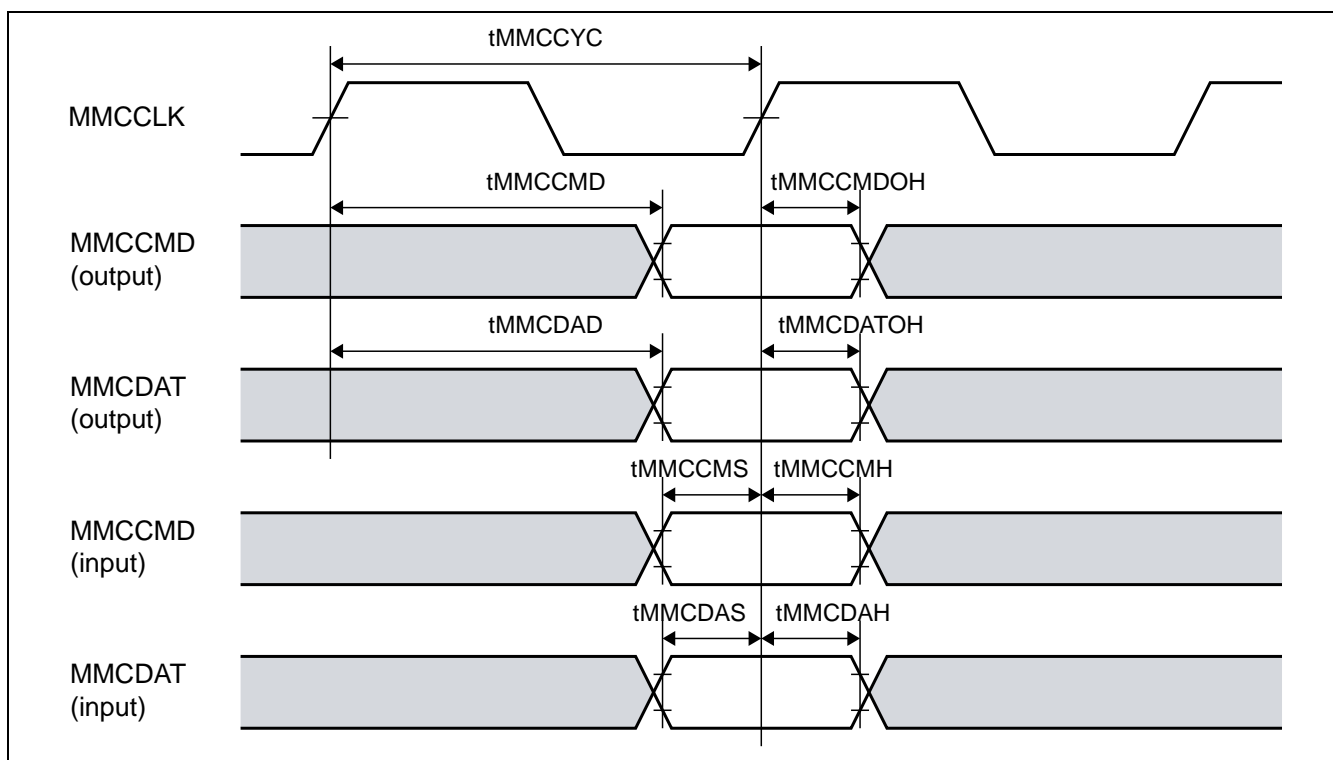


Figure 73.26.1 MMC Signal Timing (High Speed mode)

73.26.2 MMC (1.8V) Electrical Characteristics [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E]

Table 73.26.2 MMC Signal Timing (High Speed mode)

Conditions: VDDQVA_SDn (n = 2, 3) = 1.8 V ± 0.1 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 VDDQ_SD3 = 1.8 V ± 0.1 V [RZ/G2E], GND = VSS = 0 V,
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 CL = 30 pF [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E],
 GND = VSS = 0 V

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|-------------------------------|---------|------|------|---------------------|------|----------------|
| MMCCLK clock cycle time | tMMCCYC | 20.0 | — | — | ns | Figure 73.26.2 |
| MMCCMD output data delay time | tMMCCMD | 3.0 | — | 0.5 × tMMCCYC + 3.0 | ns | |
| MMCDAT output data delay time | tMMCDAD | 3.0 | — | 0.5 × tMMCCYC + 3.0 | ns | |
| MMCCMD input data setup time | tMMCCMS | 3.0 | — | — | ns | |
| MMCCMD input data hold time | tMMCCMH | 2.0 | — | — | ns | |
| MMCDAT input data setup time | tMMCDAS | 3.0 | — | — | ns | |
| MMCDAT input data hold time | tMMCDAH | 2.0 | — | — | ns | |

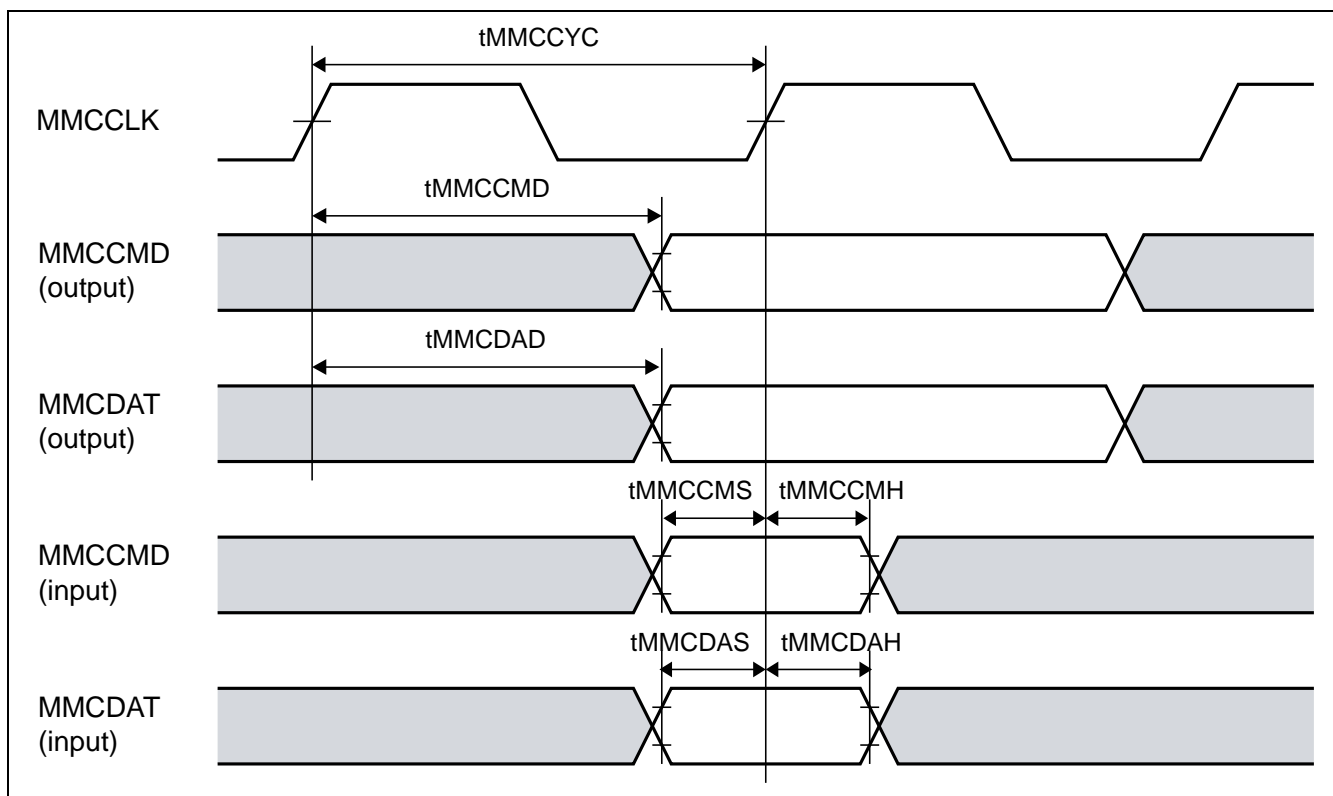


Figure 73.26.2 MMC Signal Timing

Table 73.26.3 MMC Signal Timing (HS200 mode) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E]

Conditions: VDDQVA_SDn (n = 2, 3) = 1.8 V ± 0.1 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 VDDQ_SD3 = 1.8 V ± 0.1 V [RZ/G2E], GND = VSS = 0 V,
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 CL = 10 pF [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E],
 GND = VSS = 0 V

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|---|---------|------|------|------|------|----------------|
| MMCCLK clock cycle time | tMMCCYC | 5.0 | — | — | ns | Figure 73.26.3 |
| MMCCMD output data delay time | tMMCCMD | 1.30 | — | 3.40 | ns | |
| MMCDAT output data delay time | tMMCDAD | 1.30 | — | 3.40 | ns | |
| MMCCMD input data width (for tuning) | tMMCICW | 2.57 | 3.07 | — | ns | |
| (after tuning) | | 2.57 | — | — | | |
| MMCDAT input data width (for tuning) | tMMCIDW | 2.57 | 3.07 | — | ns | |
| (after tuning) | | 2.57 | — | — | | |

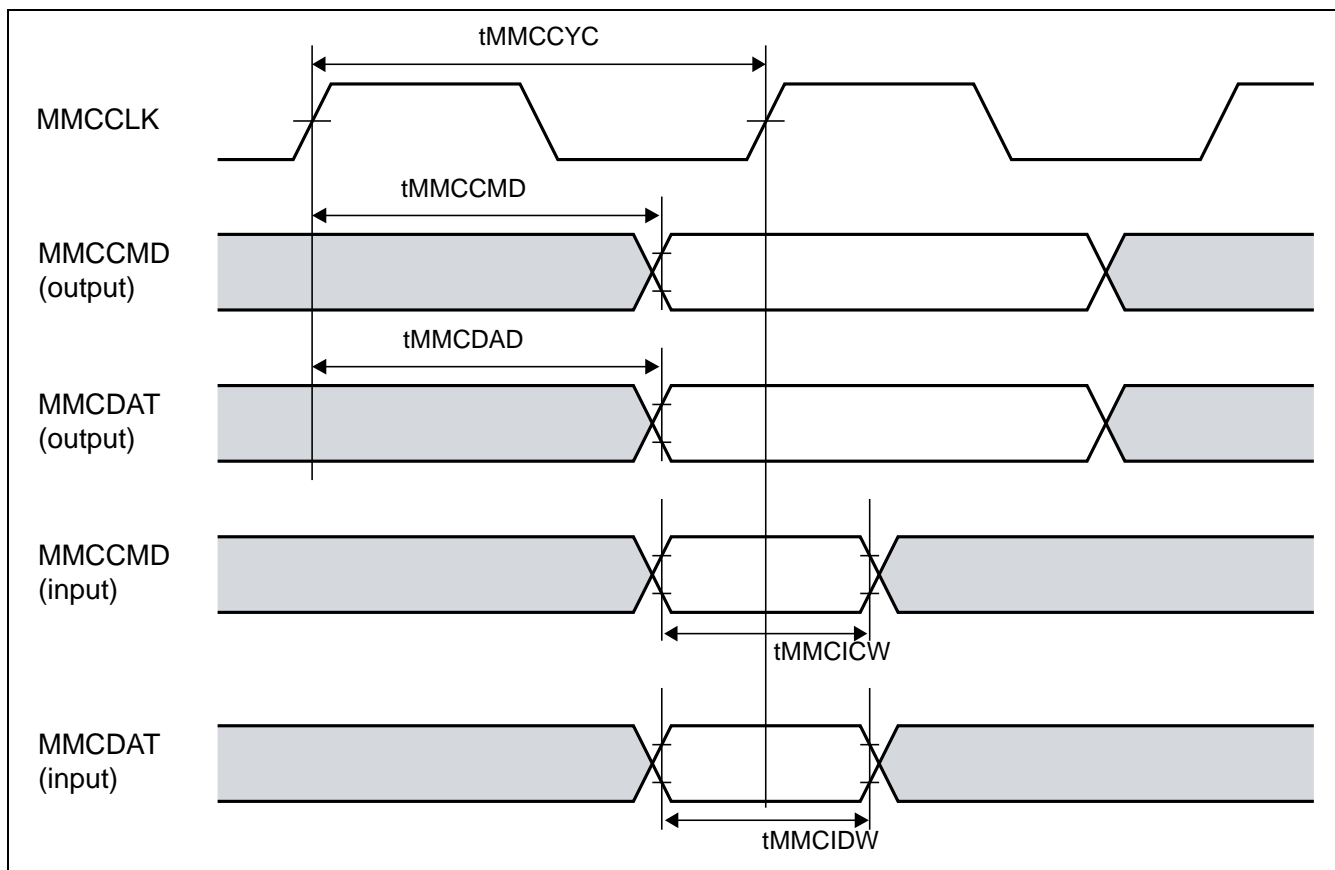


Figure 73.26.3 MMC Signal Timing (HS200 mode)

Table 73.26.4 MMC Signal Timing (HS400 mode) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N and RZ/G2E]

Conditions: VDDQVA_SDn (n = 2, 3) = 1.8 V ± 0.1 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 VDDQ_SD3 = 1.8 V ± 0.1 V [RZ/G2E], GND = VSS = 0 V,
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], CL = 10 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Remarks | Figure |
|-------------------------------|----------|------|------|------|------|--|----------------|
| MMCCLK clock cycle time | tMMCCYC | 5.0 | — | — | ns | | Figure 73.26.4 |
| MMCDS data strobe cycle time | tMMCDCYC | 5.0 | — | — | ns | | |
| MMCDS minimum pulse width | tDSW | 2.0 | — | — | ns | | |
| MMCDAT input data setup time | tMMCDIDS | — | — | 0.6 | ns | eMMC0-1 [RZ/G2H, RZ/G2M V3.0, RZ/G2N, RZ/G2E] eMMC0 [RZ/G2M V1.3] | |
| | | | | 0.5 | ns | eMMC1 [RZ/G2M V1.3] | |
| MMCDAT input data hold time | tMMCDIDH | 1.5 | — | — | ns | | |
| MMCDAT output data setup time | tMMCDWDS | 0.6 | — | — | ns | | |
| MMCDAT output data hold time | tMMCDWDH | 0.6 | — | — | ns | | |

The MMCCMD input timing for HS400 mode is the same as CMD timing for HS200 mode.

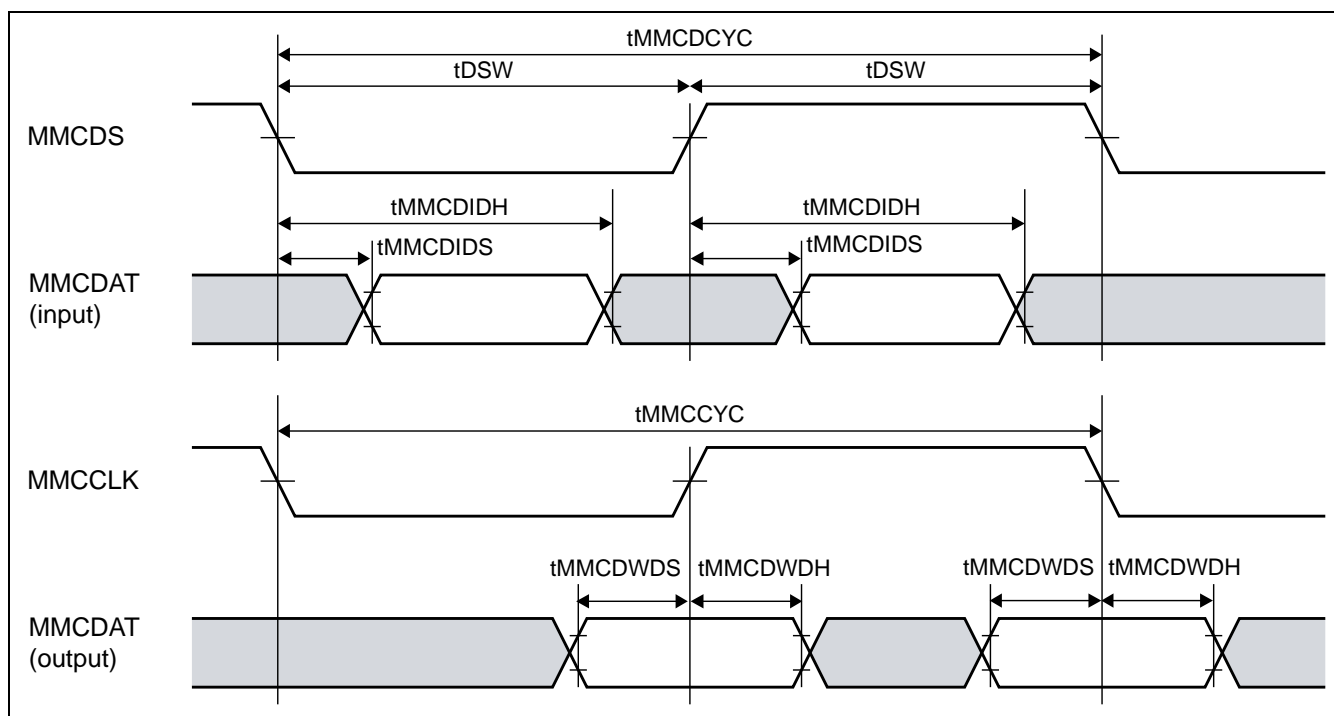


Figure 73.26.4 MMC Signal Timing (HS400 mode)

73.27 Serial ATA (Gen3)

| | | |
|--------|-------------|-------------|
| RZ/G2H | RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E | |

The description in this section is compliant with the following Serial ATA standard: 'Serial ATA International Organization: Serial ATA Revision 3.2, August 7, 2013'.

Table 73.27.1 Serial ATA (Gen3) Interface Characteristics [RZ/G2H, RZ/G2N]

Conditions: VDD33_SATA = VDD33_PCIE = 3.3 ± 0.3 V, VDD09_SATA = VDD09_PCIE = 0.82 -0.07 V / + 0.06 V, GND = 0 V, Tc = -40 to +115 °C [RZ/G2H], Ta = -40 to +85 °C [RZ/G2N], Tj = -40 to +115 °C [RZ/G2N]

| Test Item | Symbol | Min. | Typ. | Max. | Unit | Remarks |
|--|-----------|----------|----------|----------|-------|---------|
| RX Differential Input Voltage (Gen1i) | Vdiff RX | 325 | — | 600 | mVppd | *1 |
| RX Differential Input Voltage (Gen2i) | Vdiff RX | 275 | — | 750 | mVppd | *1 |
| RX Differential Input Voltage (Gen3i) | Vdiff RX | 200 | — | 900 | mVppd | *1 |
| TX Differential Output Voltage (Gen1i) | Vdiff TX | 400 | — | 600 | mVppd | *2 |
| TX Differential Output Voltage (Gen2i) | Vdiff TX | 400 | — | 700 | mVppd | *2 |
| TX Differential Output Voltage (Gen3i) | Vdiff TX | 240 | — | 1000 | mVppd | *2 |
| DC Coupled Common Mode Voltage (Gen1i) | Vcm dc | 200 | — | 450 | mV | *2 |
| Unit Interval (Gen1i) | TUI | 666.4333 | 666.6667 | 670.2333 | ps | *3 |
| Unit Interval (Gen2i) | TUI | 333.2167 | 333.3333 | 335.1167 | ps | *3 |
| Unit Interval (Gen3i) | TUI | 166.6083 | 166.6667 | 167.5583 | ps | *3 |
| COMRESET Transmit Gap Length (Gen1i and Gen2i and Gen3i) | TScmreset | — | 480 | — | UIOOB | *4 |
| COMWAKE Transmit Gap Length (Gen1i and Gen2i and Gen3i) | TScmwake | — | 160 | — | UIOOB | *4 |
| TX Differential Impedance (Gen1i) | ZdiffTX | 85 | — | 115 | Ω | *2 |
| TX Single-Ended Impedance (Gen1i) | Zs-eTX | 40 | — | — | Ω | *2 |
| RX Differential Impedance (Gen1i) | ZdiffRX | 85 | — | 115 | Ω | *1 |
| RX Single-Ended Impedance (Gen1i) | Zs-eRX | 40 | — | — | Ω | *1 |

- Notes: 1. RXP, RXN: DC test
 2. TXP, TXN: DC test
 3. Need Reference Clock (Low Voltage Swing, Differential Clocks). The nominal single-ended swing for each clock is 0 to 0.7 V and a nominal frequency of 100 MHz ± 100 PPM.
 4. TXP, TXN: UIOOB (UI During OOB Signaling) = 646.67 to 686.67 ps, Not tested.

Table 73.27.2 Serial ATA (Gen3) External Clock Accuracy

Conditions: VDD09_PCIE = 0.82 V - 0.07 V / + 0.06 V, GND = 0 V, Tc = -40 to +115 °C [RZ/G2H], Ta = -40 to +85 °C [RZ/G2N], Tj = -40 to +115 °C [RZ/G2N]

| Test Item | Symbol | Min. | Typ. | Max. | Unit | Remarks |
|--|--------|------|---------|------|------|--------------------------------------|
| External clock accuracy (PCIE1_CLK_P, PCIE1_CLK_M) | — | — | 100.000 | — | MHz | Frequency accuracy: ±100 ppm or less |

73.28 USB Signal Timing

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The descriptions in this section are compliant with the following USB 2.0 standards: "Universal Serial Bus Specification Revision 2.0" and "On-The-Go and Embedded Host Supplement to the USB Revision 2.0 Specification".

Table 73.28.1 USB High-speed Signal Timing

Conditions: VDDQ33 = VDDQ33_USB2* = 3.3 ± 0.2 V, VDDQ18 = 1.8 ± 0.1 V, VDD09_USB2* = 0.82 – 0.07 V / + 0.06 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N], VDDQ33 = VDDQ33_USB2* = 3.3 ± 0.3 V, VDDQ18 = VDDQ18_USB20 = 1.8 ± 0.1 V [RZ/G2E], VSS = 0 V,
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|----------------------|---------|--------|------|--------|------|--------|
| High-speed data rate | THSDRAT | 479.76 | 480 | 480.24 | Mb/s | — |

Table 73.28.2 USB Low-/full-speed Signal Timing

Conditions: VDDQ33 = VDDQ33_USB2* = 3.3 ± 0.2 V, VDDQ18 = 1.8 ± 0.1 V, VDD09_USB2* = 0.82 - 0.07 V / + 0.06 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N],
 VDDQ33 = VDDQ33_USB2* = 3.3 ± 0.3 V, VDDQ18 = VDDQ18_USB20 = 1.8 ± 0.1 V [RZ/G2E],
 VSS = 0 V,
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], CL = 50 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|------------|--|-------|------|------|--------|--------|
| Low-speed* | Rise time | tR | 75 | — | 300 | ns |
| | Fall time | tF | 75 | — | 300 | ns |
| | Differential Rise and Fall Time Matching (tR/tF) | tRFM1 | 80 | — | 125 | % |
| Full-speed | Rise time | tR | 4 | — | 20 | ns |
| | Fall time | tF | 4 | — | 20 | ns |
| | Differential Rise and Fall Time Matching (tR/tF) | tRFM | 90 | — | 111.11 | % |

Note: * The USB 2.0 Function module does not support Low-speed.

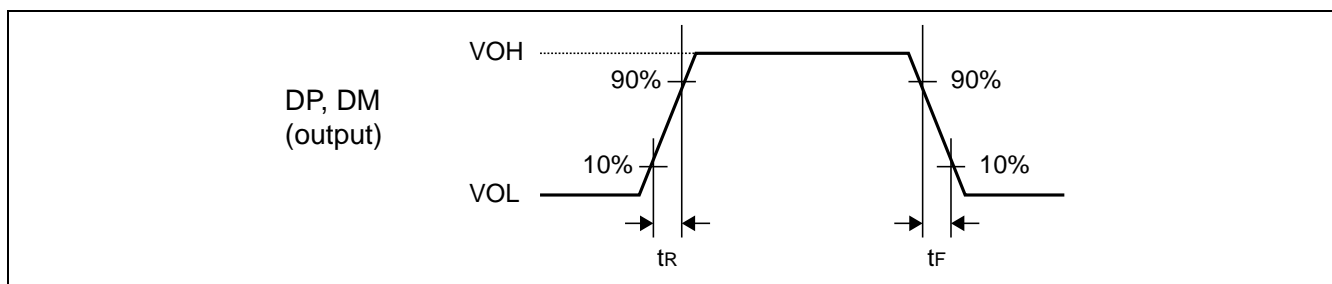


Figure 73.28.1 USB Low-/full-speed Signal Timing

Table 73.28.3 USB 2.0 External Clock Accuracy

Conditions: VDDQ18 = 1.8 V ± 0.1V,

VSS = 0 V,

Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3], Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],

Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Item | Symbol | Min. | Typ. | Max. | Unit | Remarks |
|--|--------|------|--------|------|------|--|
| External clock accuracy (USB_XTAL, USB_EXTAL) | — | — | 50.000 | — | MHz | Frequency deviation: ±100 ppm or less |

Note: USB2.0 External Clock is not supported RZ/G2E.

73.29 USB 3.0 (Super-Speed)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The description in this section is compliant with the following USB 3.0 standard: 'Universal Serial Bus 3.0 Specification Revision 1.0, Jun 6, 2011'.

Table 73.29.1 USB 3.0 (Super-Speed 5 GT/s) Interface Characteristics

Conditions: VDDQ33_USB3* = 3.3 ± 0.3 V, VDD09_USB3* = 0.82 V – 0.07 V / + 0.06 V,
 VSS = 0 V, Tc = –40 to +115 °C [RZ/G2H, RZ/G2M V1.3], Ta = –40 to +85 °C [RZ/G2M V3.0, RZ/G2N],
 Tj = –40 to +115 °C [RZ/G2M V3.0, RZ/G2N]
 VDDQ18_USB30 = 1.8 ± 0.1 V, VDDD_USB30 = 1.03 ± 0.05 V
 Ta = –40 to +85 °C, Tj = –40 to + 115 °C [RZ/G2E]

| Test Item | Symbol | Min. | Typ. | Max. | Unit | Remarks |
|---|----------------------------|---------|-------|---------|-------|---------|
| Differential Rx peak to peak voltage | VRX-DIFFp-p | 0.030 | — | — | V | *1 |
| Differential p-p Tx voltage swing | VTX-DIFFp-p | 0.8 | — | 1.2 | V | *2 |
| Absolute DC Common Mode Voltage between U1 and U0 | VTX-CM-DC-ACTIVEIDLE-DELTA | — | — | 200 | mV | *2 |
| Unit Interval | UI | 199.94 | — | 200.06 | ps | *3 |
| DC Differential TX Impedance | RTX-DIFF-DC | 72 | — | 120 | Ω | — |
| DC Differential Input Impedance | RRX-DIFF-DC | 72 | — | 120 | Ω | — |
| DC Input Impedance {DC common mode impedance} | ZRX-DC {RTX-DC, RRX-DC} | 36 {18} | — {—} | 60 {30} | Ω {Ω} | — |

Notes: 1. RXP, RXN: DC test
 2. TXP, TXN: DC test
 3. Need Reference Clock (Low Voltage Swing, Differential Clocks). The nominal single-ended swing for each clock is 0 to 0.7 V and a nominal frequency of 100 MHz ±100 PPM.

Table 73.29.2 USB 3.0 (Super-Speed 5 GT/s) External Clock Accuracy

Conditions: VDD09_USB30 = 0.82 V – 0.07 V / + 0.06 V,
 VDDQ18_USB30 = 1.8 V ± 0.1 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 VDDQ18_USB30 = 1.8 V ± 0.1 V, VDDD_USB30 = 1.03 ± 0.05 V [RZ/G2E], VSS = 0 V,
 Tc = –40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = –40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = –40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Test Item | Symbol | Min. | Typ. | Max. | Unit | Remarks |
|--|--------|------|---------|------|------|--------------------------------------|
| External clock accuracy (USB_XTAL, USB_EXTAL) * | — | — | 50.000 | — | MHz | Frequency accuracy: ±100 ppm or less |
| External clock accuracy (USB_3S0_CLK_P, USB_3S0_CLK_M) | — | — | 100.000 | — | MHz | |

Notes: For details about selecting clock sources, refer to Figure 63.1 in User's Manual (p. 63-1)

* Except for RZ/G2E

73.30 TMU

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.30.1 TMU Signal Timing

Conditions: VDDQ33 = 3.3 V ± 0.2 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]/± 0.3 V [RZ/G2E],
 GND = VSS = 0 V,
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3], Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|-------------------------|---------|------|------|-------|---------|----------------|
| TCLK clock cycle | tTCLKCY | 4 | — | 16.37 | tCYC | Figure 73.30.1 |
| Input clock pulse width | tTCLKW | 0.4 | — | 0.6 | tTCLKCY | |

Note: RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, RZ/G2E: tCYC is for one cycle of the S3D2φ clock.

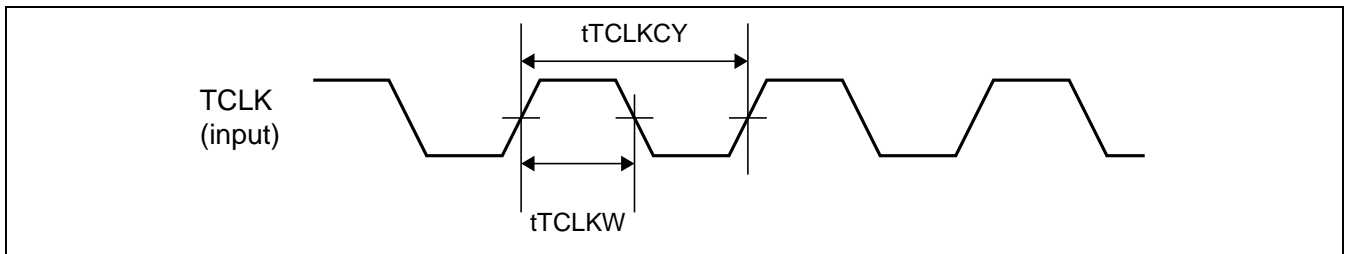


Figure 73.30.1 TMU Signal Timing

73.31 R-NANDC

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.31.1 R-NANDC Timing [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Conditions: VDDQ33 = VDDQVA_SDn = 3.3 ± 0.2 V, GND = VSS = 0 V, Tc = -40 to +115 °C [RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N], Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N], CL = 50 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|--|--------|-------------------------------|------|------|------|----------------|
| Output signal delay | tOD | 0 | — | 3.7 | ns | Figure 73.31.1 |
| rise-fall delay delta of each output signals | tRFD | 0 | — | 2.7 | ns | — |
| Read data setup time from NFRE# rise | tSD | 10 | — | — | ns | Figure 73.31.1 |
| Read data hold time from NFRE# rise | tHD | 0 | — | — | ns | |
| Read data setup time from NFRE# rise (fast mode) | tSD | 10-tpcyc *1 | — | — | ns | |
| Read data hold time from NFRE# rise (fast mode) | tHD | tpcyc *1 | — | — | ns | |
| NFWE# rise to NFRB# fall | tWB | (N+1) × tpcyc – 8 *1 *2 | — | — | ns | |
| Ready to RE# low | tRR | (M+1) × tpcyc + 0 *1 *3 | — | — | ns | |

Notes: 1. tpcyc is a cycle time of S3D1φ
 2. N is value of TIME_SEQ_1.TWB bit
 3. M is value of TIME_SEQ_1.TRR bit

Table 73.31.2 R-NANDC Timing [RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Conditions: VDDQ18 = 1.8 ± 0.1 V, GND = VSS = 0 V, Tc = -40 to +115 °C [RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N], CL = 30 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|--|--------|-------------------------------|------|------|------|----------------|
| Output signal delay | tOD | 0 | — | 3 | ns | Figure 73.31.1 |
| rise-fall delay delta of each output signals | tRFD | 0 | — | 1.9 | ns | — |
| Read data setup time from NFRE# rise | tSD | 8.5 | — | — | ns | Figure 73.31.1 |
| Read data hold time from NFRE# rise | tHD | 0 | — | — | ns | |
| Read data setup time from NFRE# rise (fast mode) | tSD | 8.5 – tpcyc *1 | — | — | ns | |
| Read data hold time from NFRE# rise (fast mode) | tHD | tpcyc *1 | — | — | ns | |
| NFWE# rise to NFRB# fall | tWB | (N+1) × tpcyc – 8 *1 *2 | — | — | ns | |
| Ready to RE# low | tRR | (M+1) × tpcyc + 0 *1 *3 | — | — | ns | |

Notes: 1. tpcyc is a cycle time of S3D1φ
 2. N is value of TIME_SEQ_1.TWB bit
 3. M is value of TIME_SEQ_1.TRR bit

Table 73.31.3 R-NANDC Timing [RZ/G2E]

Conditions: VDDQ33 = VDDQ_MMC = 3.3 ± 0.3 V, GND = VSS = 0 V, Ta = -40 to +85 °C, Tj = -40 to +115 °C,
CL = 30 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|--|--------|----------------------------|------|------|------|----------------|
| Output signal delay | tOD | 0 | — | 2.2 | ns | Figure 73.31.1 |
| rise-fall delay delta of each output signals | tRFD | 0 | — | 1 | ns | — |
| Read data setup time from NFRE# rise | tSD | 8.75 | — | — | ns | Figure 73.31.1 |
| Read data hold time from NFRE# rise | tHD | 0 | — | — | ns | |
| Read data setup time from NFRE# rise (fast mode) | tSD | 8.75 – tpcyc *1 | — | — | ns | |
| Read data hold time from NFRE# rise (fast mode) | tHD | tpcyc *1 | — | — | ns | |
| NFWE# rise to NFRB# fall | tWB | (N+1) × tpcyc – 8 *1 *2 | — | — | ns | |
| Ready to RE# low | tRR | (M+1) × tpcyc + 0 *1 *3 | — | — | ns | |

Notes: 1. tpcyc is a cycle time of S3D1Cφ
2. N is value of TIME_SEQ_1.TWB bit
3. M is value of TIME_SEQ_1.TRR bit

Table 73.31.4 R-NANDC Timing [RZ/G2E]

Conditions: VDDQ18 = VDDQ_MMC = 1.8 ± 0.1 V, GND = VSS = 0 V, Ta = -40 to +85 °C, Tj = -40 to +115 °C, CL = 30 pF

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|--|--------|-------------------------|------|------|------|----------------|
| Output signal delay | tOD | 0 | — | 2.2 | ns | Figure 73.31.1 |
| rise-fall delay delta of each output signals | tRFD | 0 | — | 1 | ns | — |
| Read data setup time from NFRE# rise | tSD | 8.75 | — | — | ns | Figure 73.31.1 |
| Read data hold time from NFRE# rise | tHD | 0 | — | — | ns | |
| Read data setup time from NFRE# rise (fast mode) | tSD | 8.75 – tpcyc *1 | — | — | ns | |
| Read data hold time from NFRE# rise (fast mode) | tHD | tpcyc *1 | — | — | ns | |
| NFWE# rise to NFRB# fall | tWB | (N+1) × tpcyc – 8 *1 *2 | — | — | ns | |
| Ready to RE# low | tRR | (M+1) × tpcyc + 0 *1 *3 | — | — | ns | |

- Notes: 1. tpcyc is a cycle time of S3D1Cφ
 2. N is value of TIME_SEQ_1.TWB bit
 3. M is value of TIME_SEQ_1.TRR bit

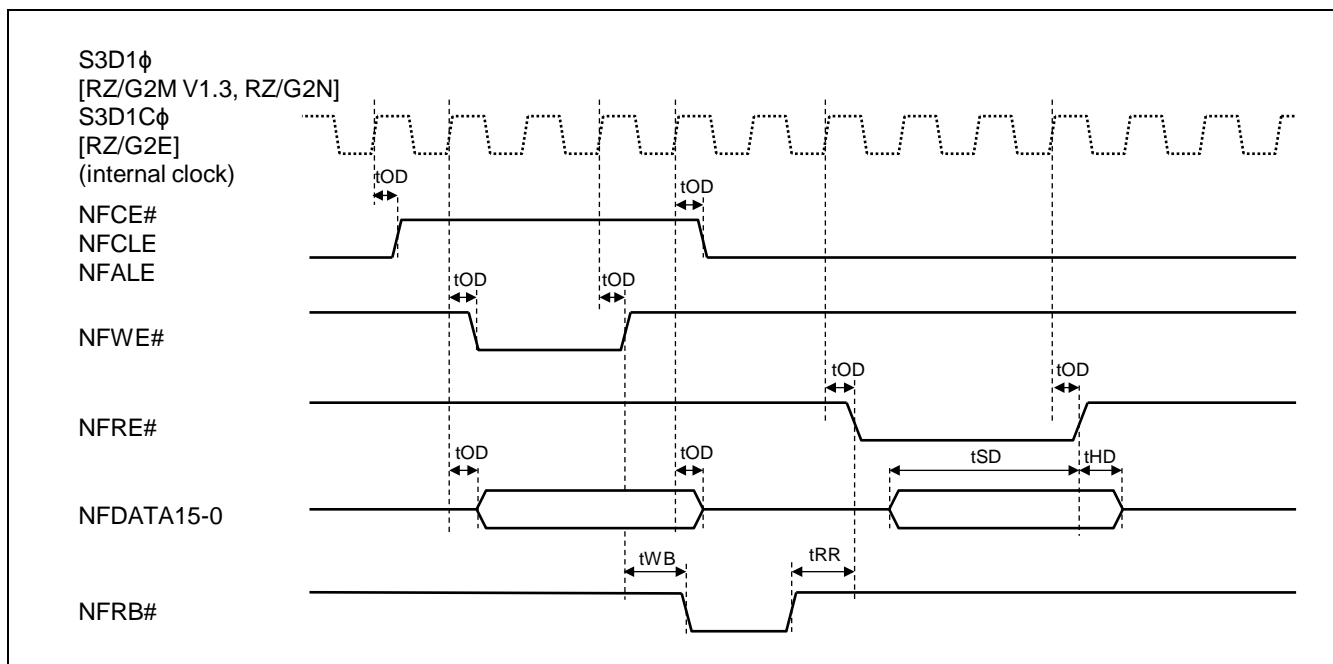


Figure 73.31.1 Input/output Timing in Synchronous Mode

73.32 DBSC4 Access Timing

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.32.1 DBSC4 Access Timing (DDR3L) [RZ/G2E]

Conditions for DDR3L: VDDQ_DDR = 1.35 V + 0.100/-0.067 V, GND = VSS = 0 V, Ta = -40 to +85 °C,

Tj = -40 to +115 °C

MDQ, MDQS, MDQS#, and MDM pins = output driver 34Ω setting, other pins = output driver 40Ω setting

| Item | Symbol | Min. | Max. | Unit | Figure |
|--|----------|-------|-------|----------|----------------|
| MCK average clock period | tCK(avg) | 1.074 | 1.266 | ns | Figure 73.32.2 |
| MCK absolute high pulse width | tCH(abs) | 0.44 | — | tCK(avg) | Figure 73.32.2 |
| MCK absolute low pulse width | tCL(abs) | 0.44 | — | tCK(avg) | Figure 73.32.2 |
| Command, Address, and Control output setup time to MCK, MCK# | tOS(1T) | 310 | — | ps | Figure 73.32.3 |
| | tOS(2T) | 890 | — | ps | |
| Command, Address, and Control output hold time from MCK, MCK# | tOH(1T) | 310 | — | ps | Figure 73.32.3 |
| | tOH(2T) | 320 | — | ps | |
| Command, Address, and Control pulse width for each output | tOPW(1T) | 620 | — | ps | Figure 73.32.3 |
| | tOPW(2T) | 1210 | — | ps | |
| Write Latency | WL | CWL | — | tCK(avg) | Figure 73.32.4 |
| MDQS, MDQS# rising edge to MCK, MCK# rising edge (write) | tWDQSS | -0.20 | 0.20 | tCK(avg) | Figure 73.32.4 |
| MDQS, MDQS# falling edge setup time to MCK, MCK# rising edge (write) | tWDSS | 0.25 | — | tCK(avg) | Figure 73.32.4 |
| MDQS, MDQS# falling edge hold time from MCK, MCK# rising edge (write) | tWDSH | 0.25 | — | tCK(avg) | Figure 73.32.4 |
| MDQS, MDQS# differential high pulse width (write) | tWDQSH | 0.45 | 0.55 | tCK(avg) | Figure 73.32.5 |
| MDQS, MDQS# differential low pulse width (write) | tWDQSL | 0.45 | 0.55 | tCK(avg) | Figure 73.32.5 |
| MDQS, MDQS# differential WRITE Preamble (write) | tWPRE | 0.9 | — | tCK(avg) | Figure 73.32.5 |
| MDQS, MDQS# differential WRITE Postamble (write) | tWPST | 0.3 | — | tCK(avg) | Figure 73.32.5 |
| MDQ and MDM output setup time to MDQS, MDQS# (write) | tWDS | 167 | — | ps | Figure 73.32.6 |
| MDQ and MDM output hold time from DQS, DQS# (write) | tWDH | 167 | — | ps | Figure 73.32.6 |
| MDQ and MDM output pulse width for each output (write) | tWDIPW | 334 | — | ps | Figure 73.32.6 |
| Read latency | RL | CL | — | tCK(avg) | Figure 73.32.7 |
| MDQS, MDQS# rising edge input access time from rising MCK, MCK# (read) | tRDQSCK | -225 | 1375 | ps | Figure 73.32.7 |
| MDQS, MDQS# differential input high pulse width (read) | tRQSH | 0.4 | — | tCK(avg) | Figure 73.32.8 |

| Item | Symbol | Min. | Max. | Unit | Figure |
|---|--------|------|------|----------|----------------|
| MDQS, MDQS# differential input low pulse width (read) | tRQSL | 0.4 | — | tCK(avg) | Figure 73.32.8 |
| MDQS, MDQS# differential READ Preamble (read) | tRPRE | 0.9 | — | tCK(avg) | Figure 73.32.8 |
| MDQS, MDQS# differential READ Postamble (read) | tRPST | 0.3 | — | tCK(avg) | Figure 73.32.8 |
| MDQS, MDQS# to DQ skew, per group, per access (read) | tRDQSQ | — | 124 | ps | Figure 73.32.9 |
| MDQ input hold time from MDQS, MDQS# (read) | tRQH | 0.35 | — | tCK(avg) | Figure 73.32.9 |

Note: The signal timing is based on the following electric potential:
 For MCK output, MDQS input/output: Differential input/output cross point voltage
 For MDQ input: MVREF
 For outputs other than MCK or MDQS: $0.5 \times VDDQ_DDR$

- Reference Load and VTT Termination of AC Timing
 Reference Load for AC Timing

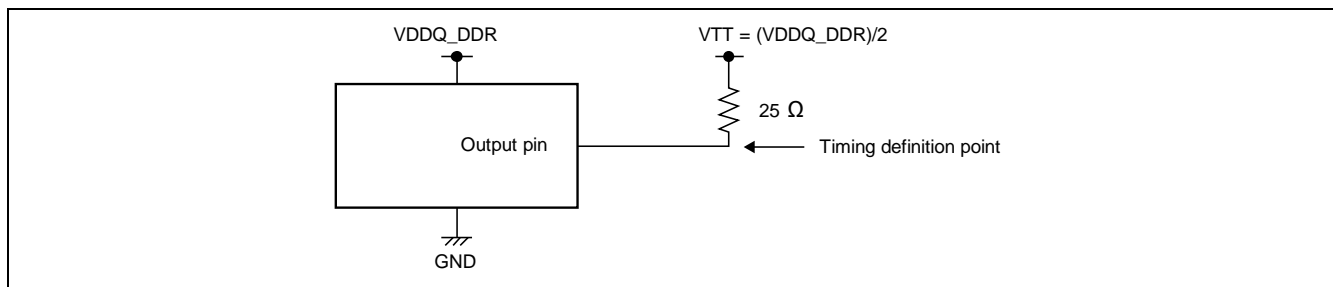


Figure 73.32.1 Reference Load for AC Timing

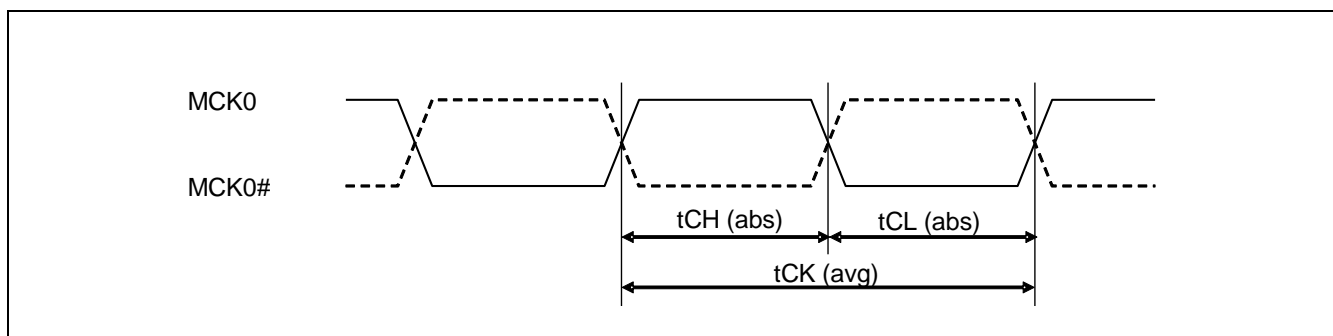


Figure 73.32.2 MCK Clock Output

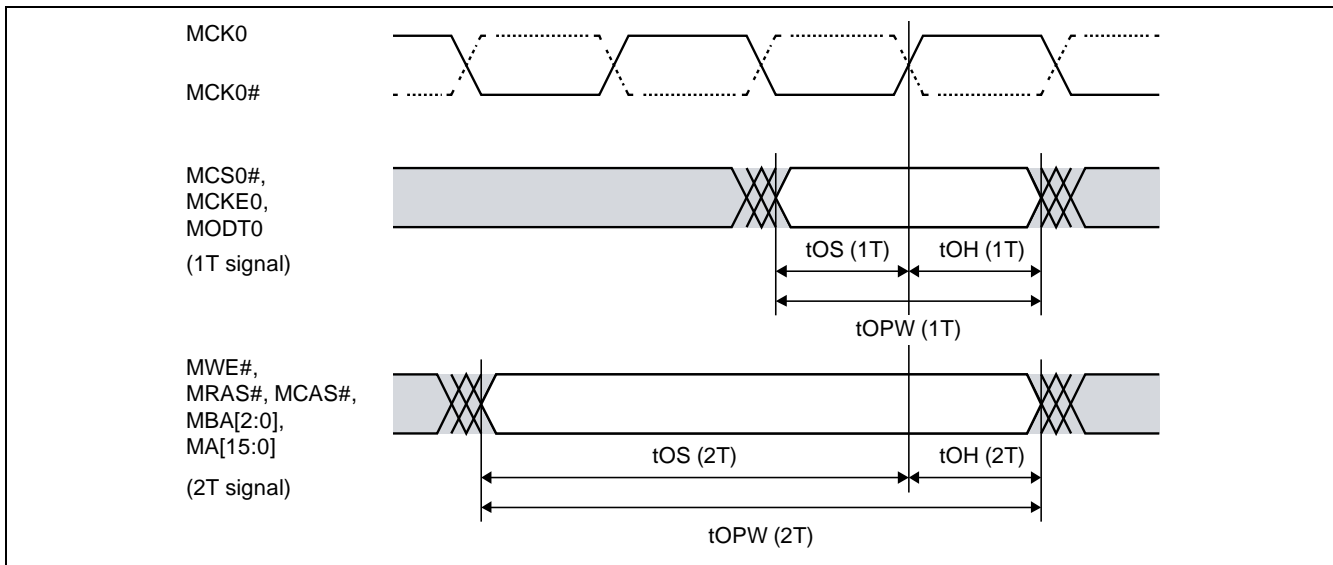


Figure 73.32.3 Command, Address, and Control Output Timing relative to MCK Output

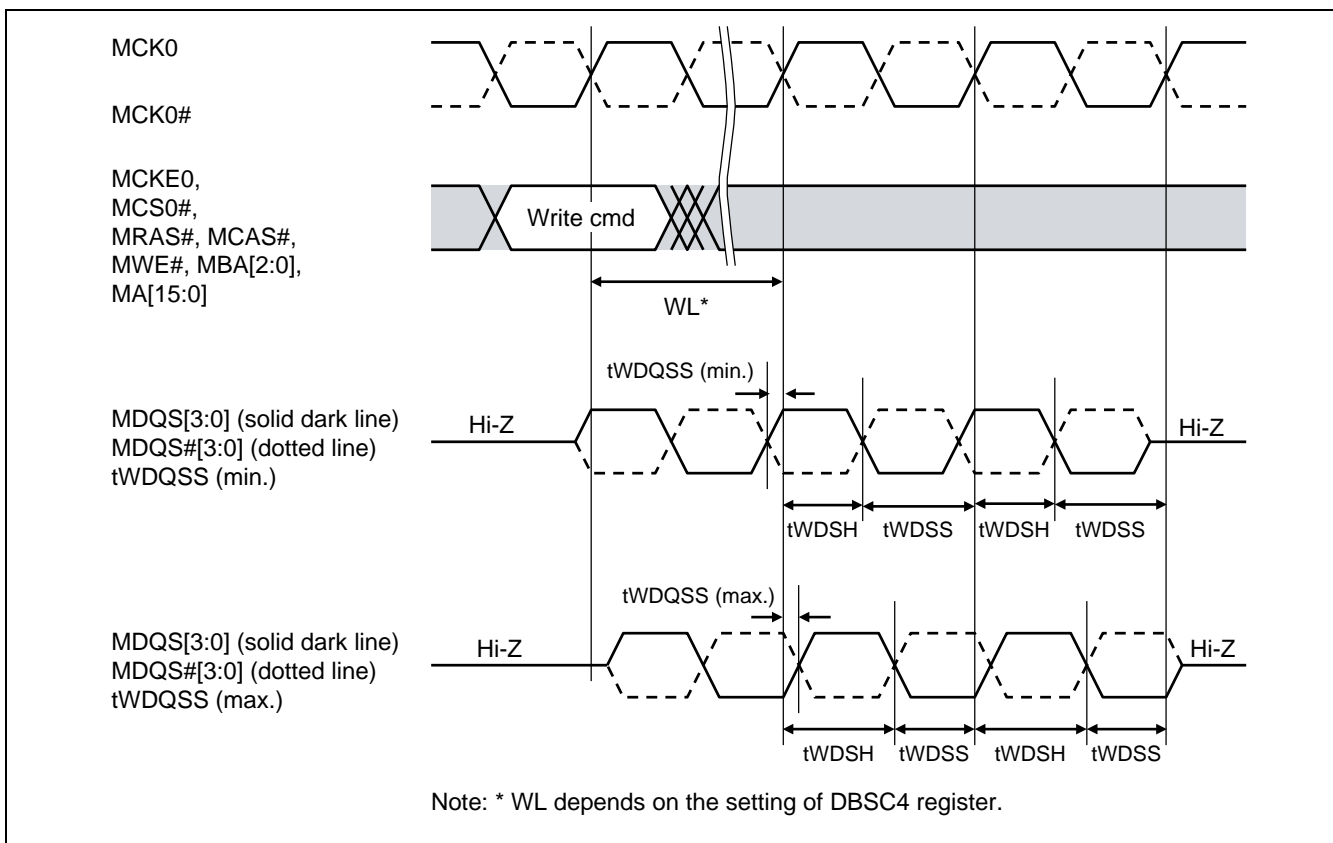


Figure 73.32.4 MDQS Output Timing relative to MCK Output (write)

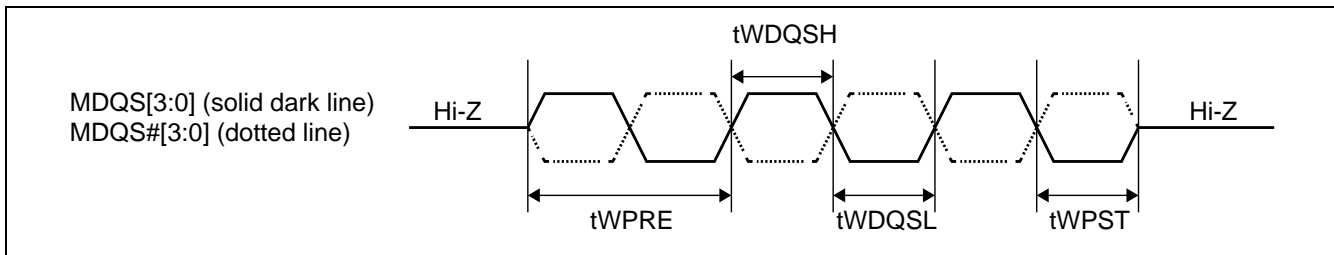


Figure 73.32.5 MDQS Output Timing (write)

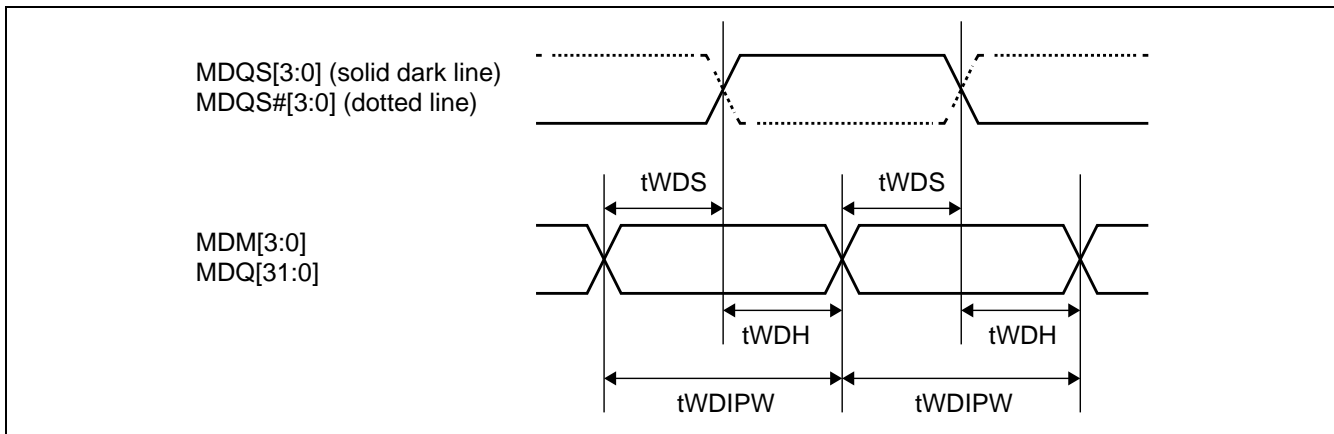


Figure 73.32.6 MDQ/MDM Output Timing relative to MDQ (write)

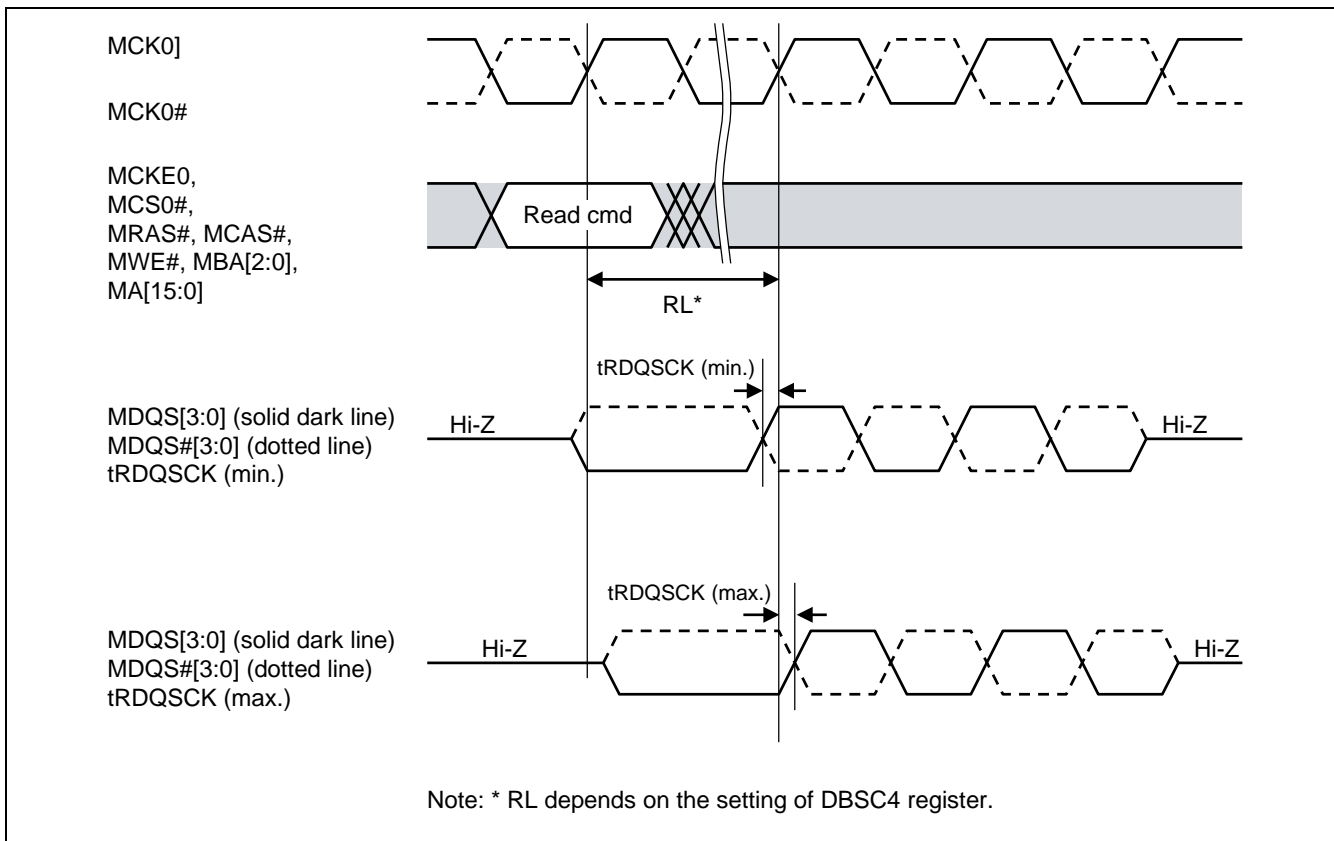


Figure 73.32.7 MDQS Input Timing relative to MCK Output (read)

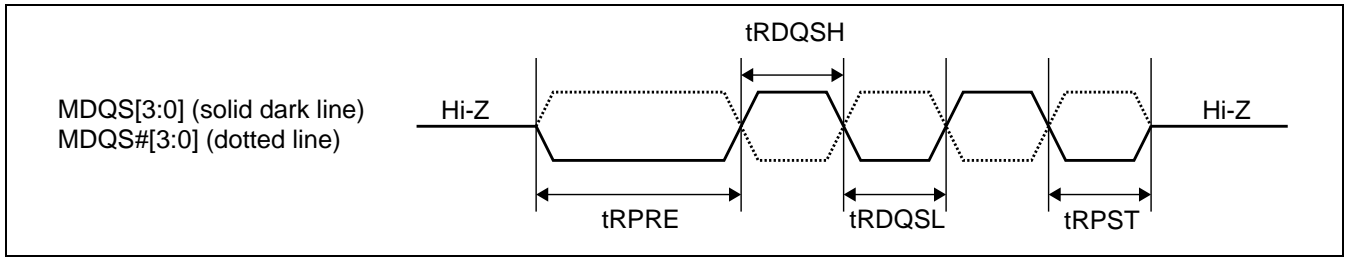


Figure 73.32.8 MDQS Input Timing (read)

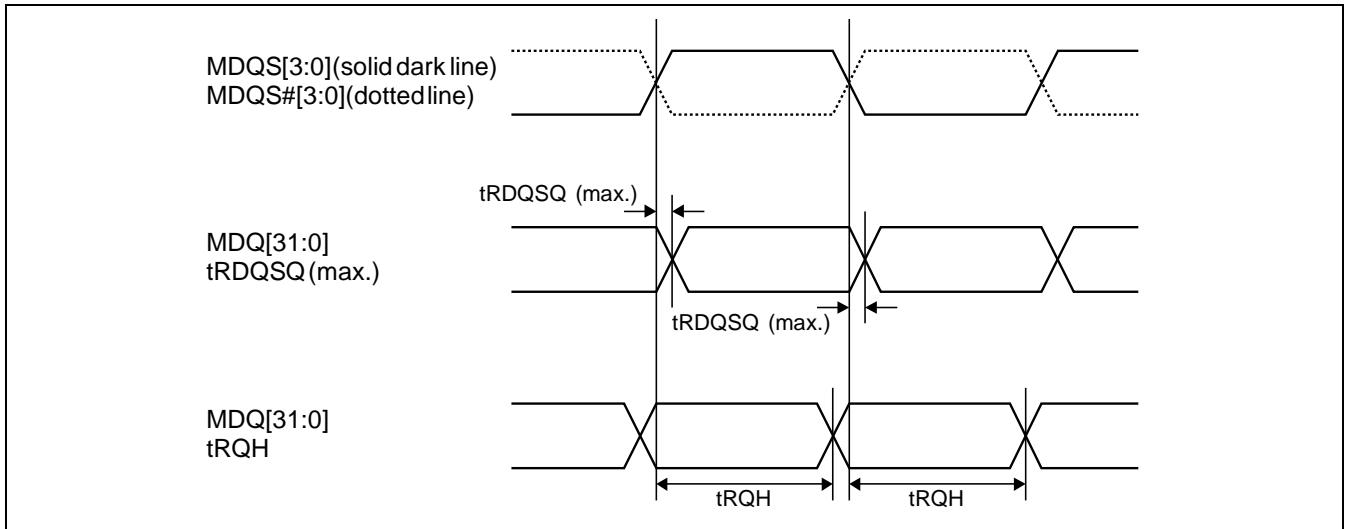


Figure 73.32.9 MDQ Input Timing relative to MDQS (read)

Table 73.32.2 DBSC4 Access Timing (LPDDR4) [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]

Conditions for LPDDR4: $VDDQ_DDR = 1.1\text{ V} + 0.07/-0.04\text{V}$, $GND = VSS = 0\text{ V}$,
 $T_c = -40\text{ to }+115\text{ }^\circ\text{C}$ [RZ/G2H, RZ/G2M V1.3],
 $T_a = -40\text{ to }+85\text{ }^\circ\text{C}$ [RZ/G2M V3.0, RZ/G2N],
 $T_j = -40\text{ to }+115\text{ }^\circ\text{C}$ [RZ/G2M V3.0, RZ/G2N]

| Item | Symbol | Min. | Max. | Unit | Figure |
|--|-------------|-------|------|----------|-----------------|
| MCK average clock period | tCK(avg) | 0.625 | 2.5 | ns | Figure 73.32.10 |
| MCK absolute high pulse width | tCH(abs) | 0.44 | 0.56 | tCK(avg) | |
| MCK absolute low pulse width | tCL(abs) | 0.44 | 0.56 | tCK(avg) | |
| Command, Address, and Control output setup time to MCK, MCK# | tOS(1T) | 191 | — | ps | Figure 73.32.11 |
| Command, Address, and Control output hold time from MCK, MCK# | tOH(1T) | 191 | — | ps | |
| Command, Address, and Control pulse width for each output | tOPW(1T) | 425 | — | ps | |
| MDQS, MDQS# rising edge to MCK, MCK# rising edge (write) | tWDQSS | 0.77 | 1.23 | tCK(avg) | Figure 73.32.12 |
| MDQS, MDQS# falling edge setup time to MCK, MCK# rising edge (write) | tWDSS | 0.22 | — | tCK(avg) | |
| MDQS, MDQS# falling edge hold time from MCK, MCK# rising edge (write) | tWDSH | 0.22 | — | tCK(avg) | |
| MDQS, MDQS# differential high pulse width (write) | tWDQSH | 0.4 | — | tCK(avg) | Figure 73.32.13 |
| MDQS, MDQS# differential low pulse width (write) | tWDQSL | 0.4 | — | tCK(avg) | |
| MDQS, MDQS# differential WRITE Preamble (write) | tWPRE | 1.8 | — | tCK(avg) | |
| MDQS, MDQS# differential WRITE Postamble (write) | tWPST | 0.4 | — | tCK(avg) | |
| MDQ and MDM output setup time to MDQS, MDQS# (write) | tWDS | 97 | — | ps | Figure 73.32.14 |
| MDQ and MDM output hold time from DQS, DQS# (write) | tWDH | 97 | — | ps | |
| MDQ and MDM output pulse width for each output (write) | tWDIPW | 194 | — | ps | |
| MDQS, MDQS# rising edge input access time from rising MCK, MCK# (read) | tRDQSCK | 1500 | 4600 | ps | Figure 73.32.15 |
| MDQS, MDQS# differential input high pulse width (read) | tRQSH | 0.38 | — | tCK(avg) | Figure 73.32.16 |
| MDQS, MDQS# differential input low pulse width (read) | tRQSL | 0.38 | — | tCK(avg) | |
| MDQS, MDQS# differential READ Preamble (read) | tRPRE | 1.8 | — | tCK(avg) | |
| MDQS, MDQS# differential READ Postamble (read) | tRPST | 0.4 | — | tCK(avg) | |
| Read Data Eye Mask (read) | TdIVW_total | — | 0.25 | UI | Figure 73.32.17 |
| | VdIVW_total | 140 | — | mV | |

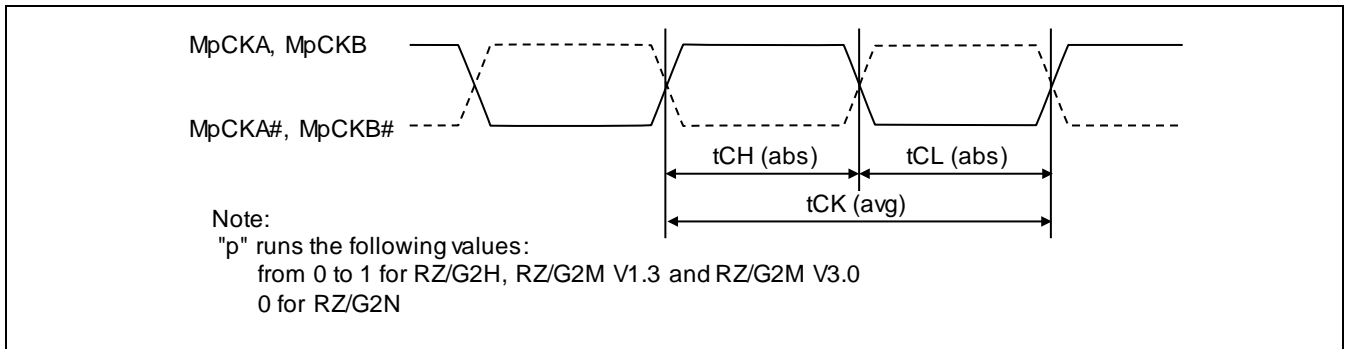


Figure 73.32.10 MCK Clock Output

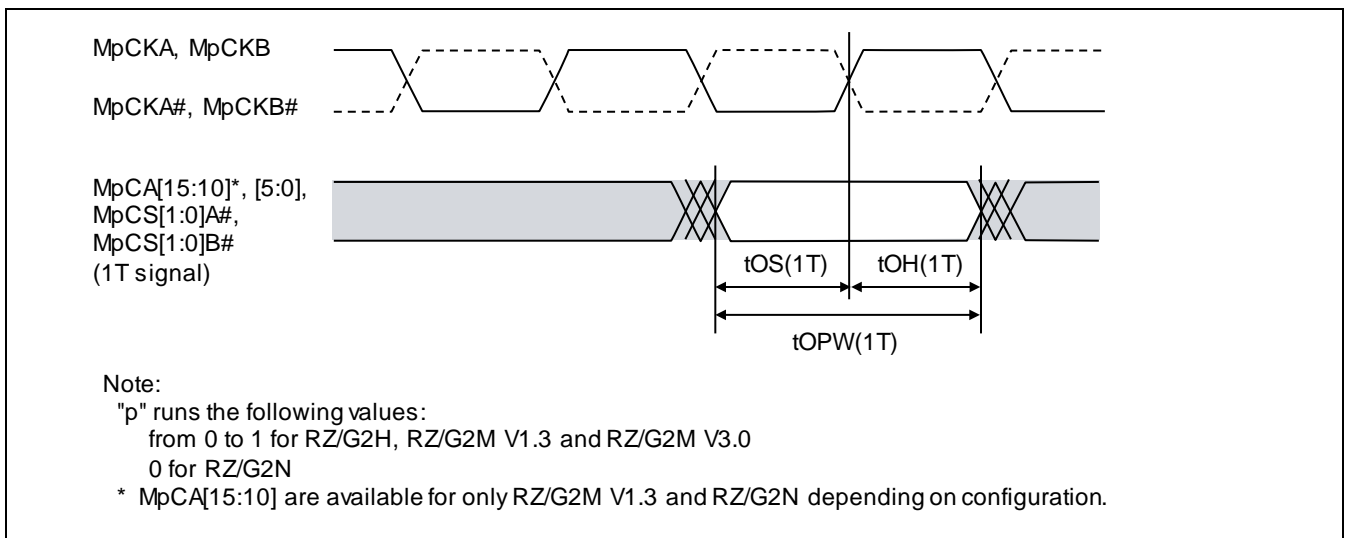


Figure 73.32.11 Command, Address, and Control Output Timing relative to MCK Output

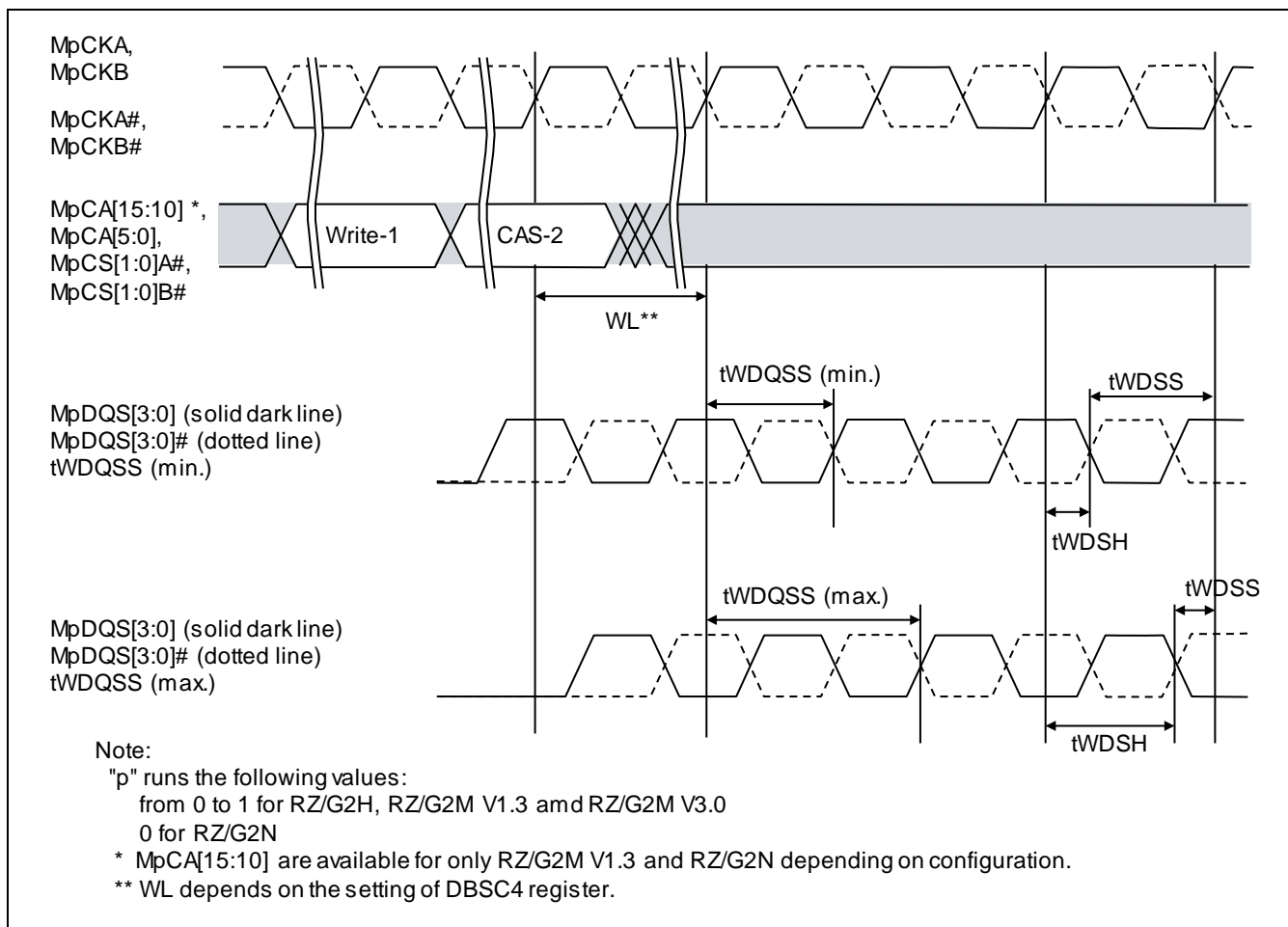


Figure 73.32.12 MDQS Output Timing relative to MCK Output (write)

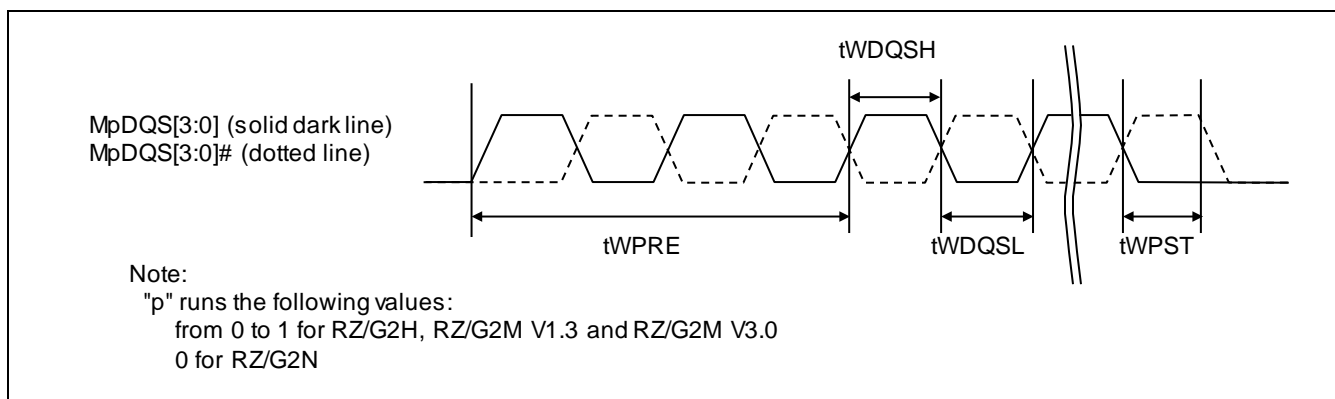


Figure 73.32.13 MDQS Output Timing (write)

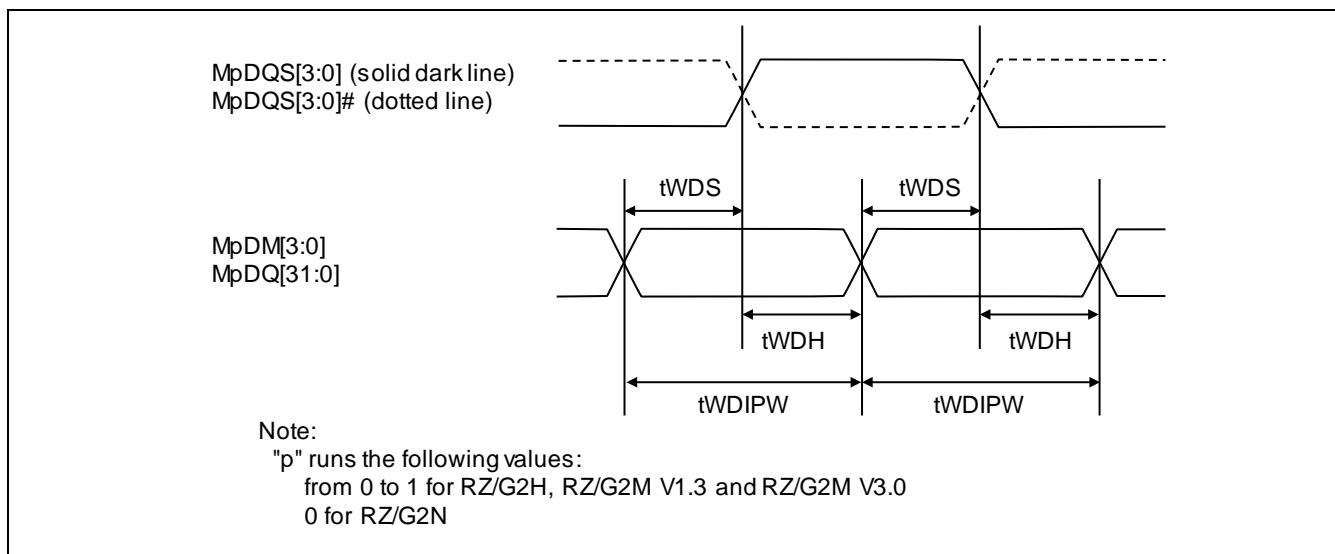


Figure 73.32.14 MDQ/MDM Output Timing relative to MDQ (write)

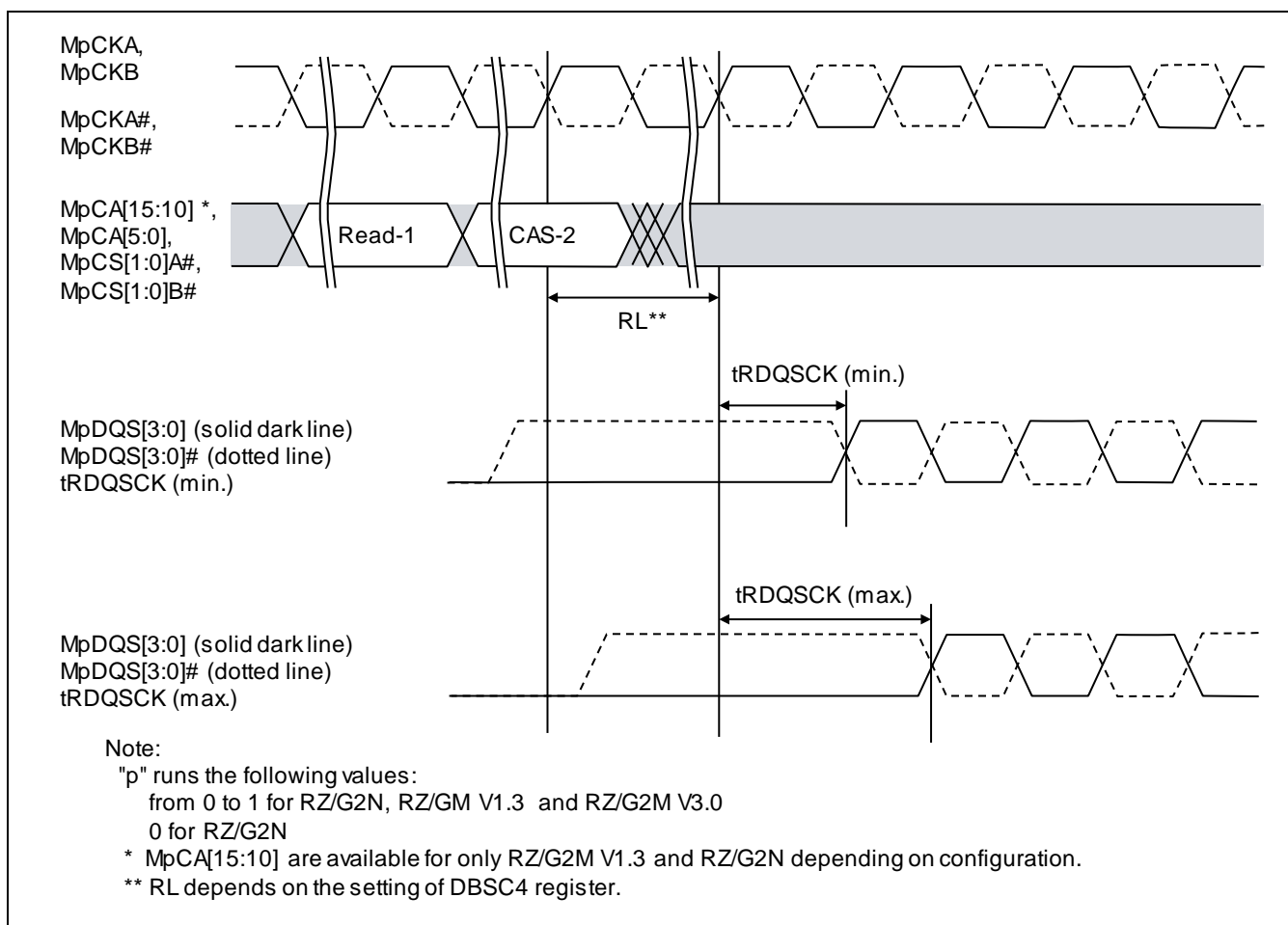


Figure 73.32.15 MDQS Input Timing relative to MCK Output (read)

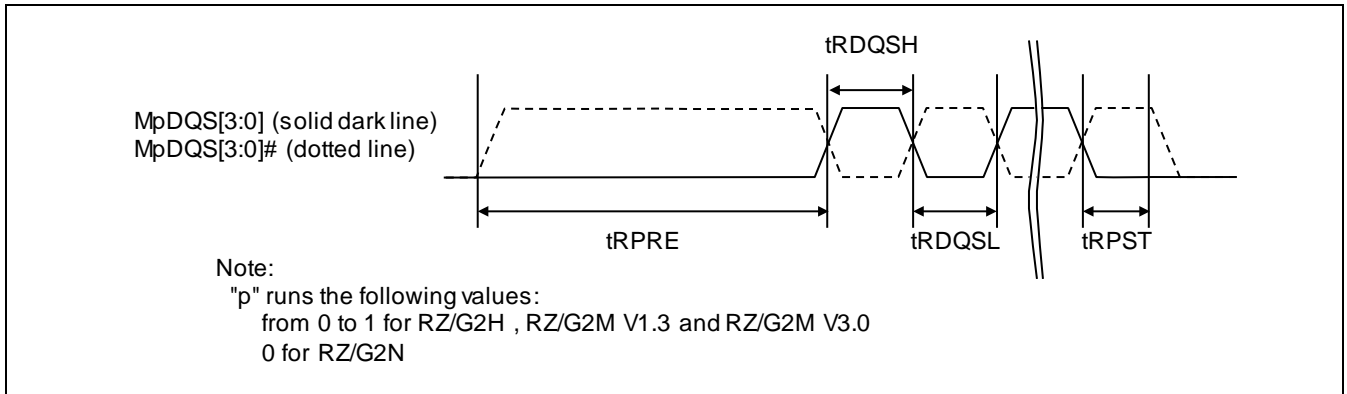


Figure 73.32.16 MDQS Input Timing (read)

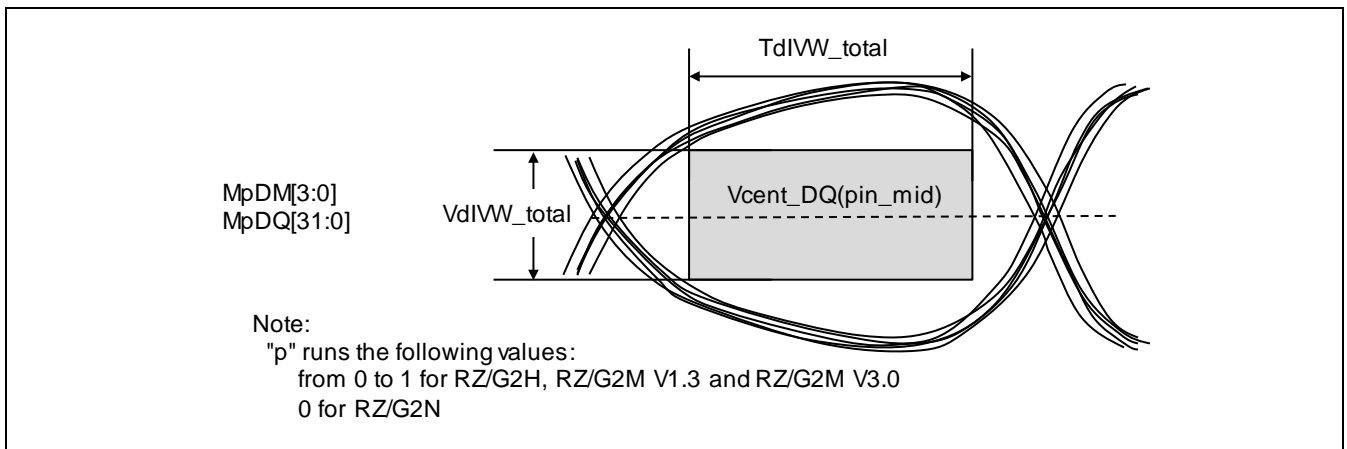


Figure 73.39.17 Read Data Eye Mask (read)

73.33 Debug and Trace

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

73.33.1 JTAG IF

Table 73.33.1 JTAG IF Signal Timing

Conditions: VDDQ18 = 1.8 V ± 0.1 V, GND = VSS = 0 V, Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E], CL = 30 pF

[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N, and RZ/G2E]

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|--|---------|------|------|------|------|----------------|
| TCK Input clock cycle | tTCKcyc | 50* | — | — | ns | Figure 73.33.1 |
| TCK Input clock pulse width (high level) | tTCKH | 20 | — | — | ns | |
| TCK Input clock pulse width (low level) | tTCKL | 20 | — | — | ns | |
| TDI/TMS setup time | tDIS | 15 | — | — | ns | Figure 73.33.2 |
| TDI/TMS hold time | tDIH | 15 | — | — | ns | |
| TDO output delay time | tDO | 0 | — | 14 | ns | |

[RZ/G2H]

| Item | Symbol | Min. | Typ. | Max. | Unit | Figures |
|--|---------|------|------|------|------|----------------|
| TCK Input clock cycle | tTCKcyc | 70* | — | — | ns | Figure 73.33.1 |
| TCK Input clock pulse width (high level) | tTCKH | 30 | — | — | ns | |
| TCK Input clock pulse width (low level) | tTCKL | 30 | — | — | ns | |
| TDI/TMS setup time | tDIS | 25 | — | — | ns | Figure 73.33.2 |
| TDI/TMS hold time | tDIH | 25 | — | — | ns | |
| TDO output delay time | tDO | 0 | — | 24 | ns | |

Note: * The cycle is 500 ns (2 MHz) during boundary scan operation.

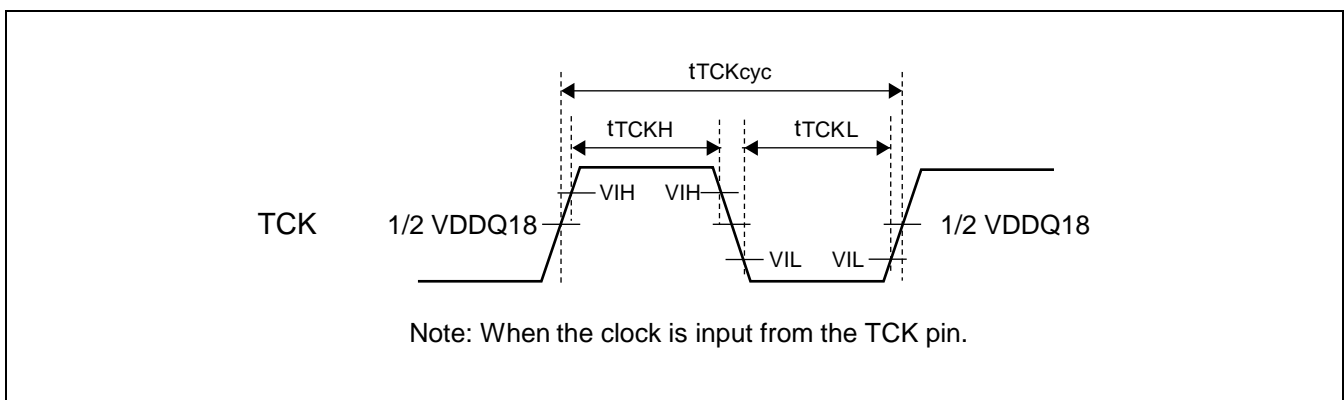
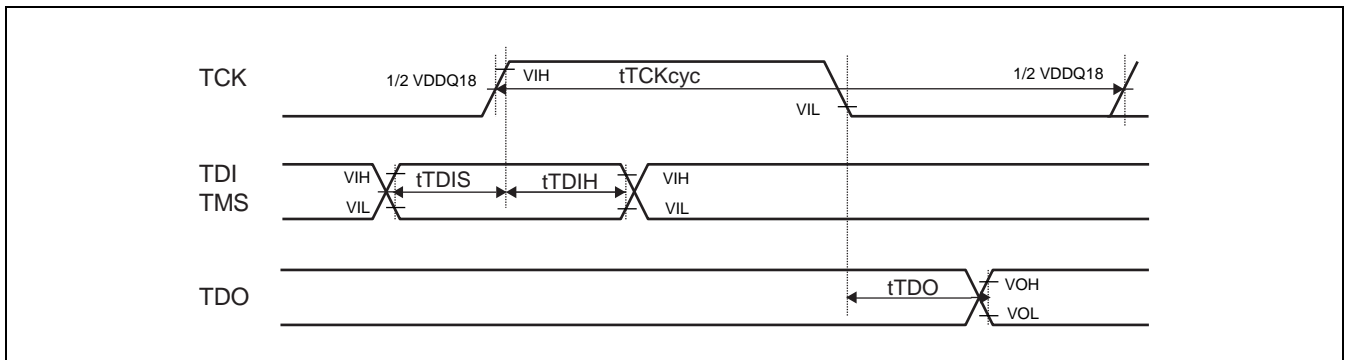


Figure 73.33.1 TCK Input Timing

**Figure 73.33.2 Data Transfer Timing**

73.34 GPIO

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.34.1 GPIO Module Signal Timing

Conditions: VDDQ33 = 3.3 V ± 0.2 V, VDDQVA_SDn = 3.3 V ± 0.2 V / 1.8 V ± 0.1 V [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]
 VDDQ33 = 3.3 V ± 0.3 V, VDD_QSPI = VDDQ_SDn = 3.3 V ± 0.3 V / 1.8 V ± 0.1 V, VDDQ25_AVB0 = 3.3 V ± 0.3 V / 2.5 V ± 0.2 V [RZ/G2E]
 Tc = -40 to +115 °C [RZ/G2H, RZ/G2M V1.3],
 Ta = -40 to +85 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E],
 Tj = -40 to +115 °C [RZ/G2M V3.0, RZ/G2N, RZ/G2E]

| Item | Symbol | Min. | Typ. | Max. | Unit | Figure |
|-------------------------|--------|------|------|------|--------|--------|
| GPIO input active width | tIOAW | 2 | — | — | tGPCYC | |

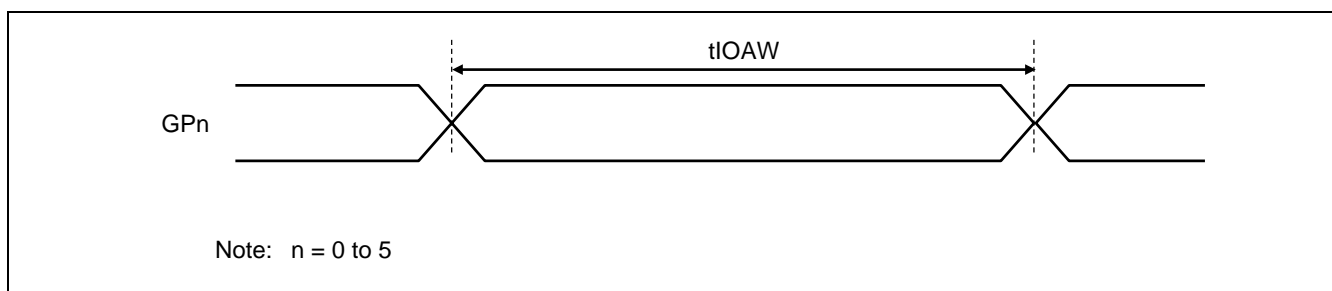


Figure 73.34.1 GPIO signal Timing

73.35 Internal PLL Characteristics (reference only)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

Table 73.35.1 Internal PLL Characteristics (reference only)

| Item | Symbol | Min. | Typ. | Max. | Unit | Remarks |
|----------------------|--------|------|------|------|------|------------|
| PLL long term jitter | — | -500 | — | +500 | ps | Term = 1μs |

Note: RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N : PLL1, RZ/G2E: PLL0

Appendix A. Product Register (PRR)

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

App.A.1 Overview

PRR indicates the product version and which of the Arm Cortex cores is present.

Notes: The Arm CPUs can read the values of all bits in this register.

App.A.1.1 Register Configuration

Table App.A.1 Register Configuration

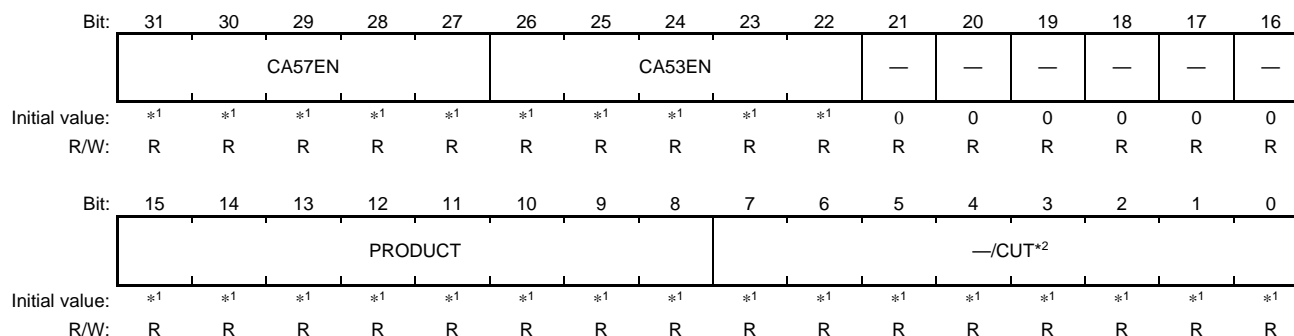
| Register Name | Abbreviation | R/W | Address | Initial Value (Power-On Reset by PRESET# Pin) | Initial Value (Software Reset) | Access Size | Second Generation RZ/G Series Products | | | |
|------------------|--------------|-----|-------------|---|--------------------------------|-------------|--|----------------------------|--------|--------|
| | | | | | | | RZ/G2H | RZ/G2M V1.3 RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| Product register | PRR | R | H'FFF0_0044 | * | * | 32 | √ | √ | √ | √ |

App.A.2 Register Description

App.A.2.1 Product register

The following describes the details of the product register (PRR).

| | | | |
|--------|--------------------------|--------|--------|
| RZ/G2H | RZ/G2M V1.3, RZ/G2M V3.0 | RZ/G2N | RZ/G2E |
| √ | √ | √ | √ |



- Notes:
- The read values of these bits are explained overleaf. Bits 31 to 22 for processors that are not present in a given product are read as 1.
 - RZ/G2M V1.3 and RZ/G2M V3.0 only.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---|-----|--|
| 31 to 27 | CA57EN | It depends on the product. See the description. | R | <p>Cortex-A57 CPU State</p> <p>[RZ/G2H]</p> <p>Bit 31 Cortex-A57 State</p> <p>0: The product has four Cortex-A57 CPUs.</p> <p>1: The product does not have four Cortex-A57 CPUs.</p> <p>Bit 30 Cortex-A57 CPU3 State</p> <p>0: The product has Cortex-A57 CPU3.</p> <p>1: The product does not have Cortex-A57 CPU3.</p> <p>Bit 29 Cortex-A57 CPU2 State</p> <p>0: The product has Cortex-A57 CPU2.</p> <p>1: The product does not have Cortex-A57 CPU2.</p> <p>Bit 28 Cortex-A57 CPU1 State</p> <p>0: The product has Cortex-A57 CPU1.</p> <p>1: The product does not have Cortex-A57 CPU1.</p> <p>Bit 27 Cortex-A57 CPU0 State</p> <p>0: The product has Cortex-A57 CPU0.</p> <p>1: The product does not have Cortex-A57 CPU0.</p> <p>[RZ/G2M V1.3, RZ/G2M V3.0, RZ/G2N]</p> <p>Bit 31 Cortex-A57 State</p> <p>0: The product has two Cortex-A57 CPUs.</p> <p>1: The product does not have two Cortex-A57 CPUs.</p> <p>Bit 30 Reserved. The read value is always 1.</p> <p>Bit 29 Reserved. The read value is always 1.</p> <p>Bit 28 Cortex-A57 CPU1 State</p> <p>0: The product has Cortex-A57 CPU1.</p> <p>1: The product does not have Cortex-A57 CPU1.</p> <p>Bit 27 Cortex-A57 CPU0 State</p> <p>0: The product has Cortex-A57 CPU0.</p> <p>1: The product does not have Cortex-A57 CPU0.</p> <p>[RZ/G2E]</p> <p>Bit 31 to 27 Reserved. The read value is always 1</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|----------|---|-----|--|
| 26 to 22 | CA53EN | It depends on the product. See the description. | R | <p>Cortex-A53 CPU State [RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0]</p> <p>Bit 26 Cortex-A53 State 0: The product has four Cortex-A53 CPUs. 1: The product does not have four Cortex-A53 CPUs.</p> <p>Bit 25 Cortex-A53 CPU3 State 0: The product has Cortex-A53 CPU3. 1: The product does not have Cortex-A53 CPU3.</p> <p>Bit 24 Cortex-A53 CPU2 State 0: The product has Cortex-A53 CPU2. 1: The product does not have Cortex-A53 CPU2.</p> <p>Bit 23 Cortex-A53 CPU1 State 0: The product has Cortex-A53 CPU1. 1: The product does not have Cortex-A53 CPU1.</p> <p>Bit 22 Cortex-A53 CPU0 State 0: The product has Cortex-A53 CPU0. 1: The product does not have Cortex-A53 CPU0.</p> <p>[RZ/G2E]</p> <p>Bit 26 Cortex-A53 State 0: The product has two Cortex-A53 CPUs. 1: The product does not have two Cortex-A53 CPUs.</p> <p>Bit 25 Reserved. The read value is always 1.</p> <p>Bit 24 Reserved. The read value is always 1.</p> <p>Bit 23 Cortex-A53 CPU1 State 0: The product has Cortex-A53 CPU1. 1: The product does not have Cortex-A53 CPU1.</p> <p>Bit 22 Cortex-A53 CPU0 State 0: The product has Cortex-A53 CPU0. 1: The product does not have Cortex-A53 CPU0.</p> <p>[RZ/G2N]</p> <p>Bit 26 to 22 Reserved. The read value is always 1.</p> |
| 21 | — | 0 | R | Reserved. The read value is always 0. |
| 20 to 16 | — | All 0 | R | Reserved |
| 15 to 8 | PRODUCT | See the description. | R | <p>Product ID Number B'0100_1111: and RZ/G2H B'0101_0010: RZ/G2M V1.3 and RZ/G2M V3.0 B'0101_0101: RZ/G2N B'0101_0111: RZ/G2E</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|----------------------|-----|--|
| 7 to 0 | — | — | R | [RZ/G2H, RZ/G2N, RZ/G2E] Reserved |
| | CUT | See the description. | R | [RZ/G2M V1.3, RZ/G2M V3.0] Cut Number B'0001_0001: RZ/G2M V1.3 B'0010_0000: RZ/G2M V3.0 |

Table App.A.2 Possible Configuration of the Arm Cortex Cores

| Product | RZ/G2H | RZ/G2M V1.3 | | RZ/G2N | RZ/G2E |
|-----------------|--------|-------------|---|--------|--------|
| | | RZ/G2M V3.0 | | | |
| Cortex-A57 CPU3 | √ | — | — | — | — |
| Cortex-A57 CPU2 | √ | — | — | — | — |
| Cortex-A57 CPU1 | √ | √ | √ | √ | — |
| Cortex-A57 CPU0 | √ | √ | √ | √ | — |
| Cortex-A53 CPU3 | √ | √ | — | — | — |
| Cortex-A53 CPU2 | √ | √ | — | — | — |
| Cortex-A53 CPU1 | √ | √ | — | — | √ |
| Cortex-A53 CPU0 | √ | √ | — | — | √ |

Appendix B. Sequence of Activation for the RZ/G series, 2nd generation products

| | |
|-------------|-------------|
| RZ/G2H | |
| RZ/G2M V1.3 | RZ/G2M V3.0 |
| RZ/G2N | RZ/G2E |

The following describes the initialization sequence of a second generation RZ/G series product.

(1) Selection of Master Boot Processor and External Boot Device

Select the master boot processor by placing the required levels on the MD6 and MD7 pins and external boot device by placing the required levels on the MD1, MD2, MD3 and MD4 pins before booting up the LSI. For details, see section 24, Booting.

Note: The following steps from (2) to (6) are not included in the sequence described in section 24, Booting because these steps are not relevant to booting. Note, however, that steps (2) to (6) must be executed before restoring the power supply to modules, including CPUs other than the master boot processor, to which the power has been cut off.

(2) Stopping the RWDT and SWDT

To prevent the watchdog timer issuing an unexpected reset or interrupt while the activation sequence is in progress, be sure to disable the RCLK watchdog timer (RWDT) and System watchdog timer (SWDT). Enable their operation at an appropriate time after the activation sequence is completed. For details, see section 17, Reset (RST), section 18, Interrupt Controller (INTC), section 64, RCLK Watchdog Timer (RWDT), and section 65, SystemWatchDog Timer (SWDT).

Notes: Use the RWTCRA register to disable the RWDT. Since the detailed specifications of the SWDT are not disclosed in this manual, contact us to inquire about the details or use the sample driver included in the board support package (BSP).

Currently, the reset and interrupt signals caused by the watchdog timer overflowing are initially masked in second generation RZ/G series products. Accordingly, this step of disabling the watchdog timers within the activation sequence is not mandatory.

(3) Selecting the Functions of Multiplexed Pins

When a pin function other than that selected by the initial setting is to be used, set the relevant registers in the pin function controller (PFC). Note that, before setting these registers, the LSI multiplexed pin setting register (PMMR) must be appropriately set to prevent incorrect writing to registers. For details, see section 6, Pin Function Controller (PFC).

Note: When a GPIO function different from that selected by the initial setting is to be used, start by setting the GPIO attributes before setting the pin function controller.

(4) Initial Settings for Operating Frequency Required by the Main Processor, Cache, MMU, and Interrupt Controller

Make initial settings for operating frequency required by the master processor, cache, MMU, and interrupt control. The operating frequencies are set by using the registers FRQCRC (for Cortex-A57 and Cortex-A53) and PLL0CR. The details of the functional specifications for the CPU required for setting the cache, MMU, and interrupt control are not included in this manual.

Notes: Cortex-A57 is supported only for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N.
Cortex-A53 is not supported for RZ/G2N.

(5) Initial Settings for Local Bus Control

When an external device is to be connected to the local bus (bits MD[4:1] are set to 0000), make initial settings for the registers including CSWCRn as required before starting access to the external device. For details, see section 25, External Bus Controller for EX-Bus (LBSC).

(6) Initial Settings for Memory Controllers

When a program is to be stored in and run from SDRAM, make initial settings for the required registers in the DBSC4. For activation sequence, see section 26, External Bus Controller for LPDDR4/ DDR3L SDRAM (DBSC4).

Note: Each product has different initial setting. Renesas is preparing the document of each product.

(7) Settings for Restoring Power Supply to Functional Modules to be Used Immediately after Activation (Other than Cortex-A57 and Cortex-A53)

In the initial state, power may not be supplied to some modules to be used. Accordingly, for modules to be used immediately after activation, set the control registers in the system controller (SYSC) as required and read the status registers to check that the power supply has been restored. For details on the sequence for restoring the power supply, see section 14, System Controller (SYSC).

(8) Settings for Release from the Module Reset State and the Supply of Clock Signals to Functional Modules to be Used Immediately after Activation (Other than CPUs)

Depends on module, in the initial state, clock signals may not be being supplied to all modules to be used. Accordingly, for modules to be used immediately after activation, set the xxxCKCR registers (xxx represents a module name) for the respective modules in the clock pulse generator (CPG) or set the RMSTPCRn/SMSTPCRn registers (n = 0 to 11) in the module standby and software reset control block of the CPG. For details, see section 11, Clock Pulse Generator (CPG) and section 12, Module Standby and Software Reset.

Notes: The settings for modules other than those to be used immediately after activation can be made separately before the modules are used.

All registers in CPG are protected from unexpected write operation, refer to section 11.2.1.

(9) Setting of Boot Address of the CPU (as Required)

To switch the boot address to an area in the DDR memory to which high-speed access is possible, start by downloading the required program to the specified area and set the starting address for execution in the boot address register for the relevant CPU. This setting is not required if the boot address is not to be changed. For details, see section 17, Reset (RST).

(10) Restoring the Power Supply to a CPU Other than the Master Boot Device

Before using a CPU (Cortex-A57 or Cortex-A53) other than the master boot processor, start by checking the PRR register described in Appendix A to confirm which CPU is available for use. If activation of a CPU disabled by the PRR register is attempted, the clock pulse generator may not operate properly.*¹

To use a CPU (Cortex-A57 or Cortex-A53) other than the master boot processor, set the Cortex-A53/Cortex-A57 debug resource reset control register (CA53DBGRCR/CA57DBGRCR) in the APMU module*² and set the relevant control register in the same module to restore the power supply. Then, read the status registers in the APMU and SYSC modules to check whether the power supply has actually been restored. This procedure must be followed for each CPU until restoration of the power supply is confirmed. For details, see section 13, Advanced Power Management Unit for AP-System Core (APMU), and section 14, System Controller (SYSC).*²

- Notes:
1. This step can be skipped if you already know which CPU is available for use.
 2. Cortex-A57 is supported only for , RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N.
Cortex-A53 is not supported for RZ/G2N.

(11) Releasing CPUs Other than the Master Boot Processor from the Reset State

When a CPU (Cortex-A57 or Cortex-A53) other than the master boot processor is to be used, clear the relevant bit in the Cortex-A57 or Cortex-A53 reset control register (CA57RESCNT/CA53RESCNT) to 0 to release it from the reset state. To reduce fluctuations in current, release the CPUs from the reset state one at a time.

This is the end of the required initial settings and all the CPUs to be used will have been activated.

- Notes:
- The settings in some steps of this sequence might be re-made by the OS.
- Cortex-A57 is supported only for RZ/G2H, RZ/G2M V1.3, RZ/G2M V3.0 and RZ/G2N.
- Cortex-A53 is not supported for RZ/G2N.

| | |
|---|---|
| Main Revisions and Additions in this Edition | RZ/G Series, 2nd Generation User's Manual: Hardware |
|---|---|

| Rev. | Date | Description | |
|------|---------------|-----------------|--|
| | | Page | Summary |
| 1.00 | Mar. 20, 2020 | — | First Edition issued |
| 1.01 | Jan. 29, 2021 | 24-13, 24-14 | Figure 24.5 eMMC using DMA boot sequence, Case of booting operation goes to NG, SCIF download mode will be started. Figure notes from 4) to 6) are removed, and note 3) is changed to be matching with the flow of revised Figure 24.5. |
| | | 29-101 ~105 | Changed figure number |
| | | 33-2 | Changed section number written in Notes no1 |

RZ/G Series, 2nd Generation
User's Manual: Hardware

Publication Date: Rev.1.00 Mar. 20.2020
Rev.1.01 Jan. 29, 2021

Published by: Renesas Electronics Corporation

**SALES OFFICES****Renesas Electronics Corporation**<http://www.renesas.com>Refer to "<http://www.renesas.com/>" for the latest and detailed information.**Renesas Electronics Corporation**

TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061, Japan

Renesas Electronics America Inc.1001 Murphy Ranch Road, Milpitas, CA 95035, U.S.A.
Tel: +1-408-432-8888, Fax: +1-408-434-5351**Renesas Electronics Canada Limited**9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004**Renesas Electronics Europe GmbH**Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327**Renesas Electronics (China) Co., Ltd.**Room 101-T01, Floor 1, Building 7, Yard No. 7, 8th Street, Shangdi, Haidian District, Beijing 100085, China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679**Renesas Electronics (Shanghai) Co., Ltd.**Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai 200333, China
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999**Renesas Electronics Hong Kong Limited**Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022**Renesas Electronics Taiwan Co., Ltd.**13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670**Renesas Electronics Singapore Pte. Ltd.**80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300**Renesas Electronics Malaysia Sdn.Bhd.**Unit No 3A-1 Level 3A Tower 8 UOA Business Park, No 1 Jalan Pengaturcara U1/51A, Seksyen U1, 40150 Shah Alam, Selangor, Malaysia
Tel: +60-3-5022-1288, Fax: +60-3-5022-1290**Renesas Electronics India Pvt. Ltd.**No.777C, 100 Feet Road, HAL 2nd Stage, Indiranagar, Bangalore 560 038, India
Tel: +91-80-67208700**Renesas Electronics Korea Co., Ltd.**17F, KAMCO Yangjae Tower, 262, Gangnam-daero, Gangnam-gu, Seoul, 06265 Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5338

RZ/G Series, 2nd Generation



Renesas Electronics Corporation

R01UH0808EJ0101